

T.C.
İSTANBUL AYDIN ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



PARKİNSON HASTALIKLARI VERİLERİNİN MAKİNE ÖĞRENMESİ
YÖNTEMLERİYLE ARAŞTIRILMASI

DOKTORA TEZİ

Fatih AYDIN

Bilgisayar Mühendisliği Ana Bilim Dalı

Bilgisayar Mühendisliği Bölümü

ARALIK 2017

T.C.
İSTANBUL AYDIN ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



PARKİNSON HASTALIKLARI VERİLERİNİN MAKİNE ÖĞRENMESİ
YÖNTEMLERİYLE ARAŞTIRILMASI

DOKTORA TEZİ

Fatih AYDIN
(Y1213.610001)

Bilgisayar Mühendisliği Ana Bilim Dalı

Bilgisayar Mühendisliği Bölümü

Tez Danışmanı: Prof.Dr. Zafer ASLAN

ARALIK 2017



21/12/2017

T.C.
İSTANBUL AYDIN ÜNİVERSİTESİ
FEN BİLİMLER ENSTİTÜSÜ MÜDÜRLÜĞÜ
DOKTORA TEZ ONAY BELGESİ

Enstitümüz Bilgisayar Mühendisliği Ana Bilim Dalı, Bilgisayar Mühendisliği Lisans Sonrası Doktora Programı Y1213.610001 numaralı öğrencisi Fatih AYDIN'in "PARKİNSON HASTALIKLARI VERİLERİNİN MAKİNE ÖĞRENMESİ YÖNTEMLERİYLE ARAŞTIRILMASI" adlı doktora tez çalışması Enstitümüz Yönetim Kurulunun 31/10/2017 tarih ve 2017/25 sayılı kararı ile oluşturulan jüri tarafından *21/12/2017* ile Doktora tezi olarak *kabul* edilmiştir.

	Unvan- Ad-Soyad	İmza
Danışman	Prof. Dr. Zafer ASLAN	<i>Zafer Aslan</i>
Üye (TİK)	Prof. Dr. Ahmad BABANLI	<i>Ahmad Babanlı</i>
Üye (TİK)	Prof. Dr. Mahir RASULOV	<i>Mahir Rasulov</i>
Üye	Doç. Dr. Taner ÇEVİK	<i>Taner Çevik</i>
Üye	Doç. Dr. Bahaddin SİNSOYSAL	<i>Bahaddin Sınoysal</i>

Tezin Savunulduğu Tarih :21/12/2017

Fen Bilimleri Enstitüsü Yönetim Kurulu'nun *21.02.2018* tarih ve *2018/03* sayılı kararı ile onaylanmıştır.

Prof. Dr. Haydar ÖZPINAR

Enstitü Müdür Vekili

YEMİN METNİ

Doktora tezi olarak sunduđum “Parkinson Hastalıkları Verilerinin Makine Öğrenmesi Yöntemleriyle Araştırılması” adlı çalışmanın, tezin proje safhasından sonuçlanmasına kadarki bütün süreçlerde bilimsel ahlak ve geleneklere aykırı düşecek bir yardıma başvurulmaksızın yazıldığını ve yararlandığım eserlerin Bibliyografya’da gösterilenlerden oluştuđunu, bunlara atıf yapılarak yararlanılmış olduğunu belirtir ve onurumla beyan ederim. (21/12/2017)

Fatih AYDIN



sevgili eşim Senem ve biricik kızım Zeynep'e ithafen,



ÖNSÖZ

Bu tez çalışmasındaki paha biçilmez rehberliği ve destekleri ile eğitimime olan katkılarından ötürü Prof. Dr. Zafer Aslan'a teşekkür etmek istiyorum. Bu tezin hazırlanması boyunca yaptıkları çok değerli yorum ve geri bildirimler için Prof. Dr. Ahmad Babanlı ve Prof. Dr. Mahir Rasulov'a teşekkür etmek istiyorum. Parkinson Hastalığı üzerine bilgi paylaşımında bulunan ve tez çalışmamda desteklerini esirgemeyen çok değerli hocalarım Prof. Dr. Pervin Kutluay İşeri, Doç Dr. Ayşe Kutlu, Yrd. Doç. Dr. Hülya Kodal Sevindir, Yrd. Doç. Dr. Cüneyt Yazıcı ve Yrd. Doç. Dr. Serap Mülâyim'e teşekkür etmek istiyorum. "Data Mining: Concepts and Techniques" isimli kitap çalışması ile Makine Öğrenmesi alanına yapmış olduğu katkılarından dolayı ve bu tez çalışmamda bana destek veren ve şans dileyen Prof. Dr. Jiawei Han'a teşekkür etmek istiyorum. "Improved complete ensemble EMD: A suitable tool for biomedical signal processing" isimli makale çalışması ile sinyal işleme alanına yapmış olduğu katkılarından dolayı ve bu tez çalışmamda yapmış oldukları değerli önerilerinden dolayı Dr. Marcelo Alejandro Colominas'a teşekkür etmek istiyorum. Ayrıca yaşamım boyunca ve özellikle doktora çalışmam süresince desteklerini hiçbir zaman esirgemeyen aileme, sevgili eşim Senem ve biricik kızım Zeynep'e çok teşekkür etmek istiyorum.

Aralık 2017

Fatih AYDIN

İÇİNDEKİLER

Sayfa

ÖNSÖZ.....	ix
İÇİNDEKİLER	xi
KISALTMALAR	xiii
ÇİZELGE LİSTESİ.....	xvii
ŞEKİL LİSTESİ.....	xxiii
SEMBOL LİSTESİ.....	xxix
ÖZET.....	xxxi
ABSTRACT	xxxiii
1. GİRİŞ.....	1
1.1 Tezin Amacı ve Yaklaşımı	5
1.2 Literatür Araştırması.....	6
1.2.1 Literatürde yer alan topluluk öğrenciler.....	6
1.2.2 Literatürde yer alan PH'nin teşhisi ile ilgili çalışmalar.....	17
1.3 Parkinson hastalığı.....	23
1.4 Yürüme	27
1.4.1 Yürümenin fazları.....	28
1.4.1.1 Topuk vuruşu (initial contact/heel strike)	29
1.4.1.2 Yüklenme/tabana vuruşu (loading response/footflat).....	29
1.4.1.3 Basma orta fazı (midstance).....	29
1.4.1.4 Basma sonu fazı/topuk kalkışı (terminal stance/heel off)	29
1.4.1.5 Salınım öncesi/parmak kalkışı (preswing/toe off)	30
1.4.1.6 Erken salınım/akselerasyon (initial swing)	30
1.4.1.7 Salınım ortası (midwing).....	30
1.4.1.8 Salınım sonu/deselerasyon (terminal swing)	30
1.4.2 Vücudun ilerletilmesi	30
1.4.3 Yürüme analizi	31
1.5 Tez İçeriği.....	34
2. VERİ VE YÖNTEM	37
2.1 Deneylerde Kullanılan Veri Kümeleri.....	37
2.1.1 Benchmark veri tabanı.....	37
2.1.2 Parkinson veri kümesi	40
2.2 Öznitelik Oluşturma ve Öznitelik Seçimi.....	40
2.2.1 Sinyal dönüşümleri ve öznitelik oluşturma	41
2.2.1.1 Fourier dönüşümü	41
2.2.1.2 Dalgacık dönüşümü.....	57
2.2.1.3 Hilbert-Huang dönüşümü.....	62
Empirik mod ayrışımı	63
Hilbert spektral analizi.....	68
2.2.2 Öznitelik seçimi.....	70
2.3 Yeni Bir Topluluk Öğrenme Algoritmasının Tasarımı: VIBES.....	76
2.3.1 VIBES algoritmasının zaman karmaşıklığı.....	80

2.3.2	VIBES algoritmasının bellek karmaşıklığı.....	83
2.3.3	İkili sınıflandırma problemleri için topluluk öğrencilerde çoğunluk oylama yönteminin uygunluğunun teorik gösterimi.....	85
2.3.4	Çoğunluk oylamalı topluluk öğrencilerde sınıflandırma hatası tahmini için önerilen yöntem	86
2.3.5	NFLT açısından VIBES algoritmasının teorik olarak değerlendirilmesi	92
2.4	VIBES Algoritmasında Özniteliklerin Enformasyon Ölçümü İçin Kullanılan Yöntemler	94
2.4.1	ReliefF algoritması	94
2.4.2	Shannon enformasyon kuramı	95
2.5	VIBES Algoritmasında Kullanılan Bazı Temel Öğreniciler.....	97
2.5.1	k-En yakın komşu algoritması	97
2.5.2	Naive bayes algoritması.....	99
2.5.3	Destek yöney makinası	100
2.5.4	Doğrusal diskriminant analiz.....	104
2.5.5	Karar ağaçları	106
2.5.6	Random forests	114
2.6	Sınıflandırıcı Performansını Ölçmek İçin Kullanılan Ölçüt ve Yöntemler .	114
2.6.1	F-Ölçütü (F_1).....	114
2.6.2	Ortalama karesel hata	115
2.6.3	Kappa değeri.....	116
2.6.4	k-Katlı çapraz doğrulama	117
3.	ANALİZ VE UYGULAMA	119
3.1	VIBES ve Diğer Makine Öğrenmesi Algoritmalarının Benchmark Veri Kümelerine Uygulanması	119
3.2	VIBES ve Diğer Makine Öğrenmesi Algoritmalarının Parkinson Veri Kümesine Uygulanması	120
4.	SONUÇLAR VE TARTIŞMA.....	131
4.1	Benchmark Veri Kümeleri Üzerinde VIBES ve Diğer Makine Öğrenmesi Algoritmalarının Karşılaştırmalı Sonuçları.....	131
4.2	Parkinson Veri Kümesi Üzerine Uygulanan Makine Öğrenmesi Algoritmalarının Sonuçları	150
4.2.1	Parkinson veri kümesine uygulanan WEKA'daki sınıflandırıcıların karşılaştırmalı sonuçları.....	151
4.2.2	Bir yapay sinir ağı modelinin parkinson veri kümesi üzerindeki sonuçları.....	155
4.2.3	VIBES algoritmasının parkinson veri kümesi üzerindeki sonuçları .	160
4.3	Bu Tez Çalışmasının Literatüre Katkıları ve Sonuçların Değerlendirilmesi	178
	KAYNAKLAR.....	181
	EKLER.....	197
	ÖZGEÇMİŞ.....	459

KISALTMALAR

123I-MIBG	: 123I-Meta-IodoBenzylGuanidine
1H-MRS	: MR Proton Spektroskopi
3D GA	: 3D Gait Analysis
A1DE	: Aggregating One-Dependence Estimators
A2DE	: Aggregating Two-Dependence Estimators
ADTree	: Alternating Decision Tree
AF	: Anlık Frekans
AH	: Alzheimer Hastalığı
ALS	: Amyotrophic Lateral Sclerosis
Bagging	: Bootstrap Aggregating
BEQ	: Biyomekaniksel Etkinlik Oranı
bGDK	: bölgesel Görünür Difüzyon Katsayısı
Bior3.3	: BiorSplines3.3 Dalgacığı
BLR	: Bayesian Logistic Regression
BN	: Bayesian Network
BTA	: Bayesian Topluluk Ağaçları
CHIRP	: Composite Hypercubes on Iterated Random Projections
CO	: Kontrol Grubu
Coif2	: Coiflet2 Dalgacığı
CP	: Cerebral Palsy
CvR	: Classification via Regression
ÇÖA	: Çoklu Öznitelik Altkümeleri
DA	: Diskriminant Analiz
DAG	: Difüzyon Ağırlıklı Görüntüleme
DaTSCAN	: Dopamin Transporter Scan
Db2	: Daubechies2 Dalgacığı
Db3	: Daubechies3 Dalgacığı
Db4	: Daubechies4 Dalgacığı
Db5	: Daubechies5 Dalgacığı
Db6	: Daubechies6 Dalgacığı
Db7	: Daubechies7 Dalgacığı
DDA	: Doğrusal Diskriminant Analiz
Dmey	: Dmeyer Dalgacığı
Düzenlenmiş-EÖM	: Düzenlenmiş-Ekstrem Öğrenme Makinesi
DYM	: Destek Yöney Makinaları
EÖML2	: L2-norm-optimize EÖM
EMA	: Empirik Mod Ayrışımı
EMG	: Electromyography
END	: Ensemble of Nested Dichotomies
ET	: Esansiyel Tremor
FDG-PET	: 18-Fluorodeoxyglucose Pozitron Emisyon Tomografi
FFD	: Fast Fourier Dönüşüm
FFT	: Fast Fourier Transform

FLR	: Fuzzy Lattice Reasoning Classifier
FT	: Functional Trees
FURIA	: Fuzzy Unordered Rule Induction Algorithm
GA	: Genetik Algoritmalar
GDI	: Gait Deviation Index
GGI	: Gillette Yürüme Göstergesi
GMM	: Gizli Markov Modeli
GPI	: Globus Pallidus Internus
GPS	: Gait Profile Score
GUTTEMA	: Gürültü Uyarlamalı TTEMA
GVS	: Yürüme Değişim Sapması
HD	: Hilbert Dönüşümü
HFI	: Hip Flexor Index
HH	: Huntington Hastalığı
HHD	: Hilbert-Huang Dönüşümü
HHT	: Hilbert-Huang Transform
HSA	: Hilbert Spektral Analizi
IBk	: Instance Based-k
IMC	: Input Mapped Classifier
İEÖMT	: İleri EÖM Topluluk
İGUTTEMA	: İyileştirilmiş GUTTEMA
İMF	: İçsel Mod Fonksiyonu
k-EYK	: k-En Yakın Komşuluk
k-KÇD	: k-Katlı Çapraz Doğrulama
KA	: Karar Ağacı
KBD	: Kortikobazal Dejenerasyon
KLR	: Kernel Logistic Regression
KZFD	: Kısa Zamanlı Fourier Dönüşümü
LC	: Lewy Cisimciği
LCD	: Lewy Cisimcikli Demans
LDA	: Linear Discriminant Analysis
LMT	: Logistic Model Trees
LOOÇD	: Leave-one-out Çapraz Doğrulama
LWL	: Locally Weighted Learning
MAE	: Mean Absolute Error
MAP	: Movement Analysis Profile
MRG	: Manyetik Rezonans Görüntüleme
MSA	: Multisistem Atrofi
MSE	: Mean Squared Error
MLP	: Multilayer Perceptron
NB	: Naive Bayes
NFLT	: No Free Lunch Teoremi
NI	: Normallik İndeksi
NNge	: Non-Nested Generalized Exemplars
OKH	: Ortalama Karesel Hata
OKHK	: Ortalama Karesel Hatanın Karekökü
PD	: Parkinsonlu Grup
PH	: Parkinson Hastalığı
PSP	: Progresif Supranükleer Palsi
QDA	: Quadratic Discriminant Analysis

RAE	: Relative Absolute Error
RC	: Random Committee
REPTree	: Reduced-Error Pruning Tree
RF	: Random Forests
RGDM	: Rastgele Genelleştirilmiş Doğrusal Model
Ridor	: Ripple-DOWn Rule Learner
RMSE	: Root Mean Squared Error
ROI	: Region Of Interest
RÖA	: Rastgele Öznitelik Ağırlıkları
RRSE	: Root Relative Squared Error
SGD	: Stochastic Gradient Descent
SimpleCART	: Simple Classification and Regression Trees
SN	: Substantia Nigra
Spegasos	: Stochastic Primal Estimated sub-GrAdient Solver for SVM
SVM	: Support Vector Machine
Sym2	: Symlets2 Dalgacı
TBA	: Temel Bileşen Analizi
TEMA	: Topluluk Empirik Mod Ayrışımı
TTEMA	: Tamamlayıcı TEMA
VAM	: Vücut Ağırlık Merkezi
VBM	: Voxel-Based Morphometry
VFI	: Voting Feature Intervals
VP	: Vasküler Parkinsonizm
WIHW	: Weighted Instances Handler Wrapper
YTKV	: Yer Tepkime Kuvvet Vektörü
ZeroR	: Zero Rule



ÇİZELGE LİSTESİ

Çizelge 2.1: Deneylerde Kullanılan Veri Kümelerinin Özellikleri	38
Çizelge 2.2: Parkinson Veri Kümesi İçin Tanımlayıcı Bilgiler (Goldberger et al. 2000).....	40
Çizelge 2.3: Fourier, dalgacık dönüşümü ve HHD nin karşılaştırmalı özeti (Huang & Wu 2008).....	63
Çizelge 2.4: Arama Yöntemlerinin Karşılaştırılması (De Silva & Leong 2015)	75
Çizelge 2.5: İki sınıflı bir tahmin için doğruluk tablosu (Stehman 1997).....	115
Çizelge 2.6: Kappa değeri için yorum tablosu (Landis & Koch 1977)	117
Çizelge 4.1: Normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları.....	159
Çizelge 4.2: Herbir Temel Öğrenici için temel parametrelerin olası durumları.....	161
Çizelge 4.3: VIBES (KA) algoritmasının LOOÇD yöntemine göre HHD'li veri kümesi üzerinde oluşturduğu modelin seçtiği öznitelik kümesi	172
Çizelge 4.4: VIBES (KA) algoritmasının LOOÇD yöntemine göre oluşturduğu modelin HHD'li Parkinson veri kümesi üzerindeki istatistiksel sonuçları	174
Çizelge 4.5: VIBES (KA) algoritmasının Hata Matrisi (Confusion Matrix).....	174
Çizelge 4.6: VIBES (KA) için HHD'li veri kümesi üzerindeki detaylı doğruluk tablosu	175
Çizelge 4.7: Aynı Parkinson veri kümesi üzerinde VIBES (KA) algoritması ve diğer çalışmaların sonuçlarının karşılaştırılması	176
Çizelge A.1: Deneklere ait tanımlayıcı bilgiler	199
Çizelge B.1: Deneylerde kullanılan bayesian öğrenme algoritmaları	205
Çizelge B.2: Deneylerde kullanılan fonksiyon öğrenme algoritmaları	206
Çizelge B.3: Deneylerde kullanılan örnek/tembel öğrenme algoritmaları	207
Çizelge B.4: Deneylerde kullanılan çeşitli öğrenme yaklaşımlarını kullanan algoritmalar	208
Çizelge B.5: Deneylerde kullanılan kural tabanlı öğrenme algoritmaları	209
Çizelge B.6: Deneylerde kullanılan karar ağacı öğrenme algoritmaları.....	210
Çizelge B.7: Deneylerde kullanılan meta/topluluk öğrenme algoritmaları	211
Çizelge B.8: Sinyallere uygulanan istatistiksel büyüklükler	249
Çizelge B.9: FD kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır).....	250
Çizelge B.10: Gaus2 dalgacığı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)	251
Çizelge B.11: BiorSplines3.3 dalgacığı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır).....	252
Çizelge B.12: Coiflets2 dalgacığı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)	257
Çizelge B.13: Daubechies2 dalgacığı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır).....	262

Çizelge B.14: Daubechies3 dalgacıđı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır).....	266
Çizelge B.15: Daubechies4 dalgacıđı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır).....	271
Çizelge B.16: Daubechies5 dalgacıđı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır).....	277
Çizelge B.17: Daubechies6 dalgacıđı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır).....	282
Çizelge B.18: Daubechies7 dalgacıđı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır).....	287
Çizelge B.19: DMeyer dalgacıđı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)	293
Çizelge B.20: Haar dalgacıđı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır).....	298
Çizelge B.21: Symlets2 dalgacıđı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)	302
Çizelge B.22: HHD kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)	306
Çizelge C.1: Baseline algoritmaların benchmark veri kümeleri üzerindeki sınıflandırma sonuçları	311
Çizelge C.2: Bayesian öğrenme algoritmalarının benchmark veri kümeleri üzerindeki sınıflandırma sonuçları	313
Çizelge C.3: Fonksiyon öğrenme algoritmalarının benchmark veri kümeleri üzerindeki sınıflandırma sonuçları	315
Çizelge C.4: Örnek tabanlı öğrenme algoritmalarının benchmark veri kümeleri üzerindeki sınıflandırma sonuçları	318
Çizelge C.5: Farklı öğrenme yaklaşımları kullanan algoritmaların benchmark veri kümeleri üzerindeki sınıflandırma sonuçları	320
Çizelge C.6: Kural öğrenme algoritmalarının benchmark veri kümeleri üzerindeki sınıflandırma sonuçları	322
Çizelge C.7: Karar ağacı algoritmalarının benchmark veri kümeleri üzerindeki sınıflandırma sonuçları	325
Çizelge C.8: Farklı öğrenme yaklaşımlarını benimseyen ve benchmark veri kümeleri üzerinde en iyi öğrenme sağlayan temel öğrencilerin global ölçekte karşılaştırma sonuçları (birinci grup)	328
Çizelge C.9: Farklı öğrenme yaklaşımlarını benimseyen ve benchmark veri kümeleri üzerinde en iyi öğrenme sağlayan temel öğrencilerin global ölçekte karşılaştırma sonuçları (ikinci grup).....	331
Çizelge C.10: Topluluk/meta öğrenme algoritmalarının (birinci grup) benchmark veri kümeleri üzerindeki sınıflandırma sonuçları.....	334
Çizelge C.11: Topluluk/meta öğrenme algoritmalarının (ikinci grup) benchmark veri kümeleri üzerindeki sınıflandırma sonuçları	337
Çizelge C.12: Topluluk/meta öğrenme algoritmalarının (üçüncü grup) benchmark veri kümeleri üzerindeki sınıflandırma sonuçları.....	340
Çizelge C.13: Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmalarının global ölçekte karşılaştırma sonuçları (birinci grup).....	343
Çizelge C.14: Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmalarının global ölçekte karşılaştırma sonuçları (ikinci grup)	346

Çizelge C.15: Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmalarının global ölçekte karşılaştırma sonuçları (üçüncü grup).....	349
Çizelge C.16: Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının temel öğrencilere göre benchmark veri kümeleri üzerindeki doğruluk oranları	352
Çizelge C.17: Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının temel öğrencilere göre benchmark veri kümeleri üzerindeki gerçek ve tahmini hata oranları.....	355
Çizelge C.18: Özniteliklerin bağımsızlığı varsayımına göre VIBES algoritmasının temel öğrencilere göre benchmark veri kümeleri üzerindeki doğruluk oranları	358
Çizelge C.19: Özniteliklerin bağımsızlığı varsayımına göre VIBES algoritmasının temel öğrencilere göre benchmark veri kümeleri üzerindeki gerçek ve tahmini hata oranları.....	361
Çizelge C.20: Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmaları ile VIBES algoritmasının karşılaştırma sonuçları (birinci grup).....	364
Çizelge C.21: Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmaları ile VIBES algoritmasının karşılaştırma sonuçları (ikinci grup)	367
Çizelge C.22: Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmaları ile VIBES algoritmasının karşılaştırma sonuçları (üçüncü grup).....	370
Çizelge C.23: Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının (Temel Öğrenici olarak RF algoritması seçildi) topluluk oluşturma yöntemi için kullandığı ileri yönlü arama ve bu aramanın optimize edilmiş halinin karşılaştırması.....	373
Çizelge C.24: Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının (Temel Öğrenici olarak RF algoritması seçildi) topluluk oluşturma yöntemi için kullandığı optimize edilmiş ileri yönlü arama ve Genetik Algoritmalar kullanılarak yapılan arama sonuçlarının karşılaştırması .	376
Çizelge C.25: Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının (Temel Öğrenici olarak RF algoritması seçildi) özniteliklerin enformasyon ölçümü yöntemlerine göre karşılaştırma sonuçları (deneyler 3 kere tekrarlandı)	379
Çizelge C.26: Baseline algoritmaların Parkinson veri kümesi üzerindeki sınıflandırma sonuçları	382
Çizelge C.27: Bayesian Öğrenme algoritmalarının normalize edilmiş Parkinson veri kümesi üzerindeki sınıflandırma sonuçları	383
Çizelge C.28: Fonksiyon Öğrenme algoritmalarının normalize edilmiş Parkinson veri kümesi üzerindeki sınıflandırma sonuçları	384
Çizelge C.29: Örnek Öğrenme algoritmalarının normalize edilmiş Parkinson veri kümesi üzerindeki sınıflandırma sonuçları	386
Çizelge C.30: Çeşitli öğrenme yaklaşımlarına sahip algoritmaların normalize edilmiş Parkinson veri kümesi üzerindeki sınıflandırma sonuçları	387
Çizelge C.31: Kural Tabanlı Öğrenme algoritmalarının normalize edilmiş Parkinson veri kümesi üzerindeki sınıflandırma sonuçları	388
Çizelge C.32: Karar Ağacı Öğrenme algoritmalarının normalize edilmiş Parkinson veri kümesi üzerindeki sınıflandırma sonuçları	390

Çizelge C.33: Temel Öğrencilerin Parkinson veri tabanı üzerindeki en iyi oldukları veri kümeleri ve sonuçları	392
Çizelge C.34: Topluluk/Meta Öğrenme algoritmalarının Parkinson veri kümesi üzerindeki sonuçları (Birinci grup)	394
Çizelge C.35: Topluluk/Meta Öğrenme algoritmalarının Parkinson veri kümesi üzerindeki sonuçları (İkinci grup)	396
Çizelge C.36: Makine Öğrenmesi algoritmalarının (VIBES hariç) Parkinson veri kümesi üzerindeki karşılaştırma sonuçları (Sadece temel öğrencilerden daha iyi bir öğrenme gerçekleştiren Topluluk Öğrenciler listede yer almaktadır).....	398
Çizelge C.37: VIBES algoritmasının (temel öğrenci olarak k-EYK kullanıldı) farklı parametrelerine göre Parkinson veri kümesi üzerindeki karşılaştırma sonuçları (Herbir deney 3 kere tekrarlandı)	400
Çizelge C.38: VIBES algoritmasının (temel öğrenci olarak NB kullanıldı) farklı parametrelerine göre Parkinson veri kümesi üzerindeki karşılaştırma sonuçları (Herbir deney 3 kere tekrarlandı)	402
Çizelge C.39: VIBES algoritmasının (temel öğrenci olarak DYM kullanıldı) farklı parametrelerine göre Parkinson veri kümesi üzerindeki karşılaştırma sonuçları (Herbir deney 3 kere tekrarlandı)	404
Çizelge C.40: VIBES algoritmasının (temel öğrenci olarak DDA kullanıldı) farklı parametrelerine göre Parkinson veri kümesi üzerindeki karşılaştırma sonuçları (Herbir deney 3 kere tekrarlandı)	406
Çizelge C.41: VIBES algoritmasının (temel öğrenci olarak KA kullanıldı) farklı parametrelerine göre Parkinson veri kümesi üzerindeki karşılaştırma sonuçları (Herbir deney 3 kere tekrarlandı)	408
Çizelge C.42: VIBES algoritmasının (temel öğrenci olarak RF kullanıldı) farklı parametrelerine göre Parkinson veri kümesi üzerindeki karşılaştırma sonuçları (Herbir deney 3 kere tekrarlandı)	410
Çizelge C.43: VIBES algoritmasının (temel öğrenci olarak k-EYK kullanıldı) HHD uygulanmış Parkinson veri kümesi üzerinde k-değerine göre sınıflandırma sonuçları (Herbir deney 10 kere gerçekleştirildi)	412
Çizelge C.44: VIBES algoritmasının (temel öğrenci olarak k-EYK kullanıldı) HHD uygulanmış Parkinson veri kümesi üzerinde k-KÇD'ye göre sınıflandırma sonuçları (Herbir deney 10 kere gerçekleştirildi)	413
Çizelge C.45: VIBES algoritmasının (temel öğrenci olarak KA kullanıldı) HHD uygulanmış Parkinson veri kümesi üzerinde k-KÇD'ye göre sınıflandırma sonuçları (Herbir deney 10 kere gerçekleştirildi)	414
Çizelge C.46: FFD uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)...	444
Çizelge C.47: Gaussian2 uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)...	445
Çizelge C.48: BiorSplines3.3 uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)	446

- Çizelge C.49:** Coiflets2 uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)... 447
- Çizelge C.50:** Daubechies2 uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi) 448
- Çizelge C.51:** Daubechies3 uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi) 449
- Çizelge C.52:** Daubechies4 uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi) 450
- Çizelge C.53:** Daubechies5 uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi) 451
- Çizelge C.54:** Daubechies6 uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi) 452
- Çizelge C.55:** Daubechies7 uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi) 453
- Çizelge C.56:** DMeyer uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)... 454
- Çizelge C.57:** Haar uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)... 455
- Çizelge C.58:** Symlets2 uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)... 456
- Çizelge C.59:** HHD uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)... 457



ŞEKİL LİSTESİ

Şekil 1.1: Parkinson sendromu klinik tablosu (Litvan 2005)	26
Şekil 1.2: Yürümenin fazları (Castermans et al. 2013)	29
Şekil 2.1: $[a, b]$ aralığının $[-\pi, \pi]$ aralığına dönüşmesi (Altın 2011)	51
Şekil 2.2: EMA yönteminin algoritması (Huang et al. 1998; Colominas et al. 2014)	64
Şekil 2.3: TEMA yönteminin algoritması (Wu & Huang 2009; Colominas et al. 2014).....	65
Şekil 2.4: TTEMA yönteminin algoritması (Yeh et al. 2010; Colominas et al. 2014)	66
Şekil 2.5: GUTTEMA yönteminin algoritması (Torres et al. 2011; Colominas et al. 2014).....	67
Şekil 2.6: İGUTTEMA yönteminin algoritması (Colominas et al. 2014)	68
Şekil 2.7: Özniteliklerin bağımsızlığı varsayımına göre hipotez oluşturma süreci ...	77
Şekil 2.8: Özniteliklerin bağımlılığı varsayımına göre hipotez oluşturma süreci	78
Şekil 2.9: VIBES algoritması	80
Şekil 2.10: Relief algoritması (Kira & Rendell 1992).....	95
Şekil 2.11: Doğrusal olarak birbirinden ayrılabilen veriler arasındaki hiperdüzlem (DFG Research Training Group n.d.).....	101
Şekil 2.12: w vektörü üzerine 2-boyutlu ve iki sınıflı verinin izdüşümü (Alpaydin 2014).....	105
Şekil 2.13: Örnek bir karar ağacı	106
Şekil 2.14: Temel bir karar ağacı algoritması (Han & Kamber 2006)	107
Şekil 2.15: Ayırma kriterine bağlı olarak karar ağacındaki üç olası durum ((Han & Kamber 2006)'dan izin alınarak değiştirilmiştir).....	110
Şekil 2.16: Budanmamış örnek bir karar ağacı (Han & Kamber 2006)	113
Şekil 2.17: Şekil 2.16'teki budanmamış karar ağacının budanmış biçimi (Han & Kamber 2006).....	113
Şekil 3.1: (a ve b) kontrol bireyi, (c ve d) ve (e ve f) parkinsonlu iki bireye ait sol ve sağ ayak için yere uygulanan kuvvetlerin zamana göre değişimi	121
Şekil 3.2: Parkinsonlu bireye ait sağ ayak için yere uygulanan kuvvet sinyalinin tepe analizi	123
Şekil 3.3: (a) Kontrol bireyi ve (b) Parkinsonlu bireye ait sağ ayak sinyalinin tek taraflı magnitüd spektrumu	124
Şekil 3.4: 2. Dereceden gaussian dalgacık.....	125
Şekil 3.5: Kontrol bireyi (Kontrol) ve iki parkinsonlu bireye (PH-1 ve PH-2) ait sağ ayak sinyalinin ölçekleme üstü - Q. moment grafiği	126
Şekil 3.6: Parkinsonlu bireye ait sağ ayak sinyalinin zamana göre dalgacık uyumu	126
Şekil 3.7: Kontrol bireyine ait sağ ayak sinyalinin zamana göre dalgacık uyumu..	127
Şekil 3.8: Parkinsonlu bireye ait sağ ayak sinyalinin Welch güç spektral yoğunluk tahmini.....	129
Şekil 4.1: Baseline algoritmaların benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu ve (b) rank sonuçları	132

- Şekil 4.2:** Bayesian öğrenme algoritmalarının benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu en yüksek ilk beş algoritma ve (b) rank sonuçları en yüksek ilk beş algoritma 133
- Şekil 4.3:** Fonksiyon öğrenme algoritmalarının benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu en yüksek ilk beş algoritma ve (b) rank sonuçları en yüksek ilk beş algoritma 134
- Şekil 4.4:** Örnek tabanlı öğrenme algoritmalarının benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu ve (b) rank sonuçları 135
- Şekil 4.5:** Çeşitli yaklaşımlara göre öğrenme gerçekleştiren algoritmaların benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu ve (b) rank sonuçları 136
- Şekil 4.6:** Kural tabanlı öğrenme algoritmalarının benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu en yüksek ilk beş algoritma ve (b) rank sonuçları en yüksek ilk beş algoritma 137
- Şekil 4.7:** Karar ağacı öğrenme algoritmalarının benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu en yüksek ilk beş algoritma ve (b) rank sonuçları en yüksek ilk beş algoritma 138
- Şekil 4.8:** Makine Öğrenmesi algoritmalarının (sadece temel öğrenciler) benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu en yüksek ilk beş algoritma ve (b) rank sonuçları en yüksek ilk beş algoritma..... 139
- Şekil 4.9:** Topluluk Öğrenme algoritmalarının benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu en yüksek ilk beş algoritma ve (b) rank sonuçları en yüksek ilk beş algoritma 140
- Şekil 4.10:** WEKA'daki tüm Makine Öğrenmesi algoritmalarının (*VIBES* hariç) benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu en yüksek ilk beş algoritma ve (b) rank sonuçları en yüksek ilk beş algoritma 141
- Şekil 4.11:** Özniteliklerin bağımlılığı varsayımına göre *VIBES* algoritmasının seçilen temel öğrencilere göre benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu değerleri ve (b) rank sonuçları 142
- Şekil 4.12:** Özniteliklerin bağımlılığı varsayımına göre *VIBES* algoritmasının seçilen temel öğrencilere göre benchmark veri kümeleri üzerindeki ortalama gerçek ve tahmini hata oranları 143
- Şekil 4.13:** Özniteliklerin bağımsızlığı varsayımına göre *VIBES* algoritmasının seçilen temel öğrencilere göre benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu değerleri ve (b) rank sonuçları 144
- Şekil 4.14:** Özniteliklerin bağımsızlığı varsayımına göre *VIBES* algoritmasının seçilen temel öğrencilere göre benchmark veri kümeleri üzerindeki ortalama gerçek ve tahmini hata oranları 145
- Şekil 4.15:** Tüm Makine Öğrenmesi algoritmalarının (*VIBES* dahil) benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu en yüksek ilk beş algoritma ve (b) rank sonuçları en yüksek ilk beş algoritma 146
- Şekil 4.16:** Özniteliklerin bağımlılığı varsayımına göre *VIBES* algoritmasının (Temel Öğrenici olarak RF algoritması seçildi) topluluk oluşturma yöntemi için kullandığı ileri yönlü arama ve bu aramanın optimize edilmiş halinin (a) Ortalama doğruluk oranına (b) Ortalama iterasyon sayısına ve (c) Ortalama gerçek ve tahmini hata oranına göre karşılaştırma sonuçları 147
- Şekil 4.17:** Özniteliklerin bağımlılığı varsayımına göre *VIBES* algoritmasının (Temel Öğrenici olarak RF algoritması seçildi) topluluk oluşturma

	yöntemi için kullandığı optimize edilmiş ileri yönlü arama ve Genetik Algoritmalar kullanılarak yapılan aramanın (a) Ortalama doğruluk oranına (b) Ortalama arama zamanına (c) Ortalama gerçek ve tahmini hata oranına ve (d) Rank sayısına göre karşılaştırma sonuçları	149
Şekil 4.18:	Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasında (Temel Öğrenici olarak RF algoritması seçildi) özniteliklerin enformasyonunu ölçmek için kullanılan Relief algoritması ve Shannon entropinin (a) Ortalama doğruluk oranına (b) İterasyon sayısına ve (c) Ortalama gerçek ve tahmini hata oranına göre karşılaştırma sonuçları	150
Şekil 4.19:	İki temel tahmin edicinin çeşitli sinyal dönüşümleri uygulanmış olan Parkinson veri kümesi üzerindeki sınıflandırma doğruluğu oranları	151
Şekil 4.20:	WEKA'daki Makine Öğrenmesi algoritmalarının, herbir Parkinson veri kümesi üzerinde en yüksek sınıflandırma doğruluğu ve ilgili Makine Öğrenmesi algoritması	155
Şekil 4.21:	Dmey dalgacığı uygulanmış Parkinson veri kümesi üzerinde bir Yapay Sinir Ağı modelinin eğitim, doğrulama, test ve tümünü içeren sınıflandırma sonuçları (Deney 100 kere gerçekleştirildi. Gizli katman sayısı 300'dür).....	160
Şekil 4.22:	VIBES algoritması Temel Öğrenici olarak k-EYK algoritmasını kullandığında yapılan 6 olası duruma göre veri kümeleri üzerindeki en yüksek sınıflandırma doğruluğu oranları	162
Şekil 4.23:	VIBES algoritması Temel Öğrenici olarak NB algoritmasını kullandığında yapılan 6 olası duruma göre veri kümeleri üzerindeki en yüksek sınıflandırma doğruluğu oranları	163
Şekil 4.24:	VIBES algoritması Temel Öğrenici olarak DYM algoritmasını kullandığında yapılan 6 olası duruma göre veri kümeleri üzerindeki en yüksek sınıflandırma doğruluğu oranları	164
Şekil 4.25:	VIBES algoritması Temel Öğrenici olarak DDA algoritmasını kullandığında yapılan 6 olası duruma göre veri kümeleri üzerindeki en yüksek sınıflandırma doğruluğu oranları	166
Şekil 4.26:	VIBES algoritması Temel Öğrenici olarak KA algoritmasını kullandığında yapılan 6 olası duruma göre veri kümeleri üzerindeki en yüksek sınıflandırma doğruluğu oranları	167
Şekil 4.27:	VIBES algoritması Temel Öğrenici olarak RF algoritmasını kullandığında yapılan 6 olası duruma göre veri kümeleri üzerindeki en yüksek sınıflandırma doğruluğu oranları	168
Şekil 4.28:	k değerinin değişimine göre VIBES (k-EYK) algoritmasının HHD'li veri kümesi üzerindeki sınıflandırma doğruluğunun değişim grafiği	169
Şekil 4.29:	Çapraz doğrulama yönteminin değişimine göre VIBES (1-EYK) algoritmasının sınıflandırma doğruluğunun ortalaması, standart sapması ve uç değerleri	170
Şekil 4.30:	Çapraz doğrulama yönteminin değişimine göre VIBES (KA) algoritmasının sınıflandırma doğruluğunun ortalaması, standart sapması ve uç değerleri	171
Şekil B.1:	VIBES algoritmasının benchmark veri kümeleri üzerine uygulanması için yazılan fonksiyon	212
Şekil B.2:	VIBES algoritmasının ana yapısı.....	213
Şekil B.3:	Modellerin oluşturulması süreci	214
Şekil B.4:	Topluluğun oluşturulma süreci	215

Şekil B.5: Topluluğun optimize edilerek oluşturulma süreci	216
Şekil B.6: Topluluğun Genetik Algoritmalar kullanılarak oluşturulma süreci.....	217
Şekil B.7: Topluluğun performans istatistiğinin hesaplanma süreci	218
Şekil B.8: Topluluğun performans çıktısı.....	220
Şekil B.9: Topluluk öğrenme algoritmalarında tahmini hata hesaplama.....	221
Şekil B.10: Sinyale ait istatistikler.....	222
Şekil B.11: Fourier dönüşümü kullanılarak her bir denek için öznitelik oluşturma	223
Şekil B.12: Gaus ana dalgacığı ile Dalgacık dönüşümü yapılarak her bir denek için öznitelik oluşturma	225
Şekil B.13: Ayrık Dalgacık dönüşümü yapılarak her bir denek için öznitelik oluşturma.....	227
Şekil B.14: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için BiorSplines3.3 dalgacığının 6. seviyeye kadar detay katsayıları	229
Şekil B.15: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için Coiflets2 dalgacığının 6. seviyeye kadar detay katsayıları	230
Şekil B.16: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için DMeyer dalgacığının 6. seviyeye kadar detay katsayıları	231
Şekil B.17: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için Haar dalgacığının 6. seviyeye kadar detay katsayıları.....	232
Şekil B.18: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için Symlets2 dalgacığının 6. seviyeye kadar detay katsayıları	233
Şekil B.19: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için Daubechies2 dalgacığının 6. seviyeye kadar detay katsayıları	234
Şekil B.20: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için Daubechies3 dalgacığının 6. seviyeye kadar detay katsayıları	235
Şekil B.21: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için Daubechies4 dalgacığının 6. seviyeye kadar detay katsayıları	236
Şekil B.22: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için Daubechies5 dalgacığının 6. seviyeye kadar detay katsayıları	237
Şekil B.23: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için Daubechies6 dalgacığının 6. seviyeye kadar detay katsayıları	238
Şekil B.24: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için Daubechies7 dalgacığının 6. seviyeye kadar detay katsayıları	239
Şekil B.25: HHD kullanılarak her bir denek için öznitelik oluşturma.....	240
Şekil B.26: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için ilk 6 İMF	243
Şekil B.27: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için ilk 6 İMF'ye ait genlik değerleri.....	244
Şekil B.28: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için ilk 6 İMF'ye ait AF değerleri.....	245
Şekil B.29: WEKA için arff dosya formatı oluşturma.....	246
Şekil B.30: Parkinson veri kümelerine uygulanan Yapay Sinir Ağı için benchmark programı	247
Şekil C.1: HHD'li Parkinson veri kümesi için 43. öznitelige göre oluşturulan karar ağacı modeli (Öznitelikler Shannon entropiye göre sıralandı).....	415
Şekil C.2: HHD'li Parkinson veri kümesi için 43. öznitelige göre oluşturulan karar ağacı kural modeli (Öznitelikler Shannon entropiye göre sıralandı)	416
Şekil C.3: HHD'li Parkinson veri kümesi için 68. öznitelige göre oluşturulan karar ağacı modeli (Öznitelikler Shannon entropiye göre sıralandı).....	417
Şekil C.4: HHD'li Parkinson veri kümesi için 68. öznitelige göre oluşturulan karar ağacı kural modeli (Öznitelikler Shannon entropiye göre sıralandı)	418



SEMBOL LİSTESİ

c	: Sınıf sayısı
\mathcal{D}	: Eğitim kümesi
\mathbf{e}	: Temel öğrencilerin hata olasılığı vektörü
ϵ	: Topluluk öğrencinin tahmini ortalama hatası
ϵ_{\max}	: Topluluk öğrencinin tahmini maksimum hatası
ϵ_{\min}	: Topluluk öğrencinin tahmini minimum hatası
E	: Topluluk
$E[X]$: X 'in beklenen değeri
$E[X Y]$: Y verildiğinde X 'in beklenen değeri
$EU(x)$: x 'in beklenen faydası
$EU(x, y)$: x ve y 'nin beklenen faydası
$EU(\alpha_i x)$: x özneliği bilindiğinde ve α_i kararı atandığında beklenen fayda
f	: Frekans
$f(n, m)$: Bellek karmaşıklık fonksiyonu
h_i	: i . Hipotez
$h(\omega)$: Marjinal spektrum
\mathcal{H}	: Topluluk hipotezi
K	: Kappa değeri
\mathcal{L}_i	: i . Temel öğrenci
m	: Veri sayısı
n	: Öznitelik sayısı
$P(X)$: X 'in olasılığı
$P(X Y)$: Y verildiğinde X 'in koşullu olasılığı
$s(t)$: t 'ye göre sinyal
T	: Temel öğrenci sayısı
$T(n)$: Çalışma zamanı fonksiyonu
$VIBES(X)$: VIBES algoritması temel öğrenci olarak X algoritmasını seçiyor
x	: Test örneği
\mathcal{X}	: Örnek uzayı
y	: Sınıf etiketleri
ψ	: Ana dalgacık fonksiyonu
Ψ	: Ana dalgacık fonksiyonunun Fourier dönüşümü
$\omega(t)$: Anlık frekans
+++	: Yetersiz bellekten dolayı deney gerçekleştirilemedi
???	: Uzun süren işlemlerden dolayı deney gerçekleştirilemedi
—	: Algoritma, veri kümesine uygun olmadığından deney yapılamadı



PARKİNSON HASTALIKLARI VERİLERİNİN MAKİNE ÖĞRENMESİ YÖNTEMLERİYLE ARAŞTIRILMASI

ÖZET

Bu tez çalışması Parkinson Hastalığı (PH) verilerine Makine Öğrenmesi algoritmalarının uygulanması üzerinedir. Bu amaçla özgün olarak yeni bir Makine Öğrenmesi algoritması tasarımına gidildi. Bu tasarım sonucunda; karar kaynaştırma ve sınıflandırıcı seçimi yöntemlerine dayalı olarak *VIBES* isminde yeni bir topluluk öğrenme algoritması geliştirilmiştir. *VIBES* algoritması ve diğer Makine Öğrenmesi algoritmaları UCI veritabanından seçilen 33 adet benchmark veri kümesine ve PhysioNet veritabanından elde edilen Parkinson verikümesine uygulandı.

VIBES algoritması iki varsayımdan birinin tercih edilmesine göre çalışmaktadır. Bu varsayımlar: Özniteliklerin birbirlerine bağımlı olduğu ve bağımsız olduğu varsayımlardır. Bu varsayımlara göre 33 benchmark veri kümesi üzerindeki ortalama doğruluk oranları sırasıyla %89.80 ve %88.22'dir. Benchmark veri kümeleri üzerindeki ikinci en iyi ortalamaya sahip öğrenme algoritması %87.76 doğruluk oranıyla Rotation Forest algoritmasıdır. Her iki topluluk öğrenme algoritması için kullanılan ortak temel öğrenici Random Forests (RF) algoritmasıdır. *VIBES* algoritmasında topluluk birleştirme yöntemi için çoğunluk oylama yöntemi kullanılmış ve sadece ikili sınıflandırma problemleri için çoğunluk oylama yönteminin diğer yöntemlere göre daha iyi olduğu sadece teorik olarak gösterilmiştir. *VIBES(RF)* algoritmasında hipotez arama algoritması olarak optimize edilmiş ileri yönlü bir arama yöntemi ve Genetik Algoritmalar (GA) yöntemi kullanılmıştır. 33 adet benchmark veri kümesi üzerindeki sonuçlara göre optimize edilmiş ileri yönlü arama algoritması ortalama %89.6963 doğruluk oranı ve ortalama 3.2359 saniyelik arama zamanı ile GA yöntemine göre daha iyi sonuç vermiştir. GA yöntemiyle, %89.4184 doğruluk oranı ve ortalama 127.6958 saniyelik arama zamanı elde edilmiştir. Buna göre doğruluk oranları arasında çok ciddi bir fark olmamasına rağmen optimize edilmiş ileri yönlü arama algoritması GA'ya göre daha az sürede aramalar gerçekleştirerek topluluğu oluşturmuştur. *VIBES(RF)* algoritmasında özniteliklerin enformasyon miktarını ölçmek için iki yöntem kullanılmıştır. Bunlar: ReliefF ve Shannon entropidir. Bu iki algoritmanın ortalama doğruluk oranları sırasıyla %89.8153 ve %89.0519'dur. Bu iki sonuç istatistiksel açıdan önemli bir farka sahip olmadığı ve Shannon entropi diğer yöntemle kıyasla daha hızlı çalıştığı için enformasyon ölçüm yöntemi olarak Shannon entropi, özniteliklerin bilgi miktarı ölçümü için varsayılan metot olarak ayarlanmıştır. Bu tez çalışmasında, çoğunluk oylamalı topluluk öğrenicilerde oluşturulan topluluğun gerçek hata oranını tahmin etmek için de veri uyarlamalı bir yöntem geliştirilmiştir. Bu yöntemle göre 33 adet benchmark veri kümesi üzerinde özniteliklerin bağımlılığı varsayımına göre gerçek ve tahmini hata oranları arasındaki fark %0.575'tir. Özniteliklerin bağımsızlığı varsayımına göre gerçek ve tahmini hata oranlarının farkı ise %1.655'tir. Böylelikle her iki varsayımın ortalama hata oranı %1.115 olur. Tüm bu veriler ışığında hata tahmin yönteminin gerçek hata oranına oldukça yakın sonuçlar verdiği söylenebilir.

Parkinson veri kümesi kullanılarak öznitelik oluşturmak için Fast Fourier Dönüşümü (FFD), Dalgacık dönüşümü (Haar, sym2, coif2, db2, db3, db4, db5, db6, db7, dmey, bio3.3 ve gaus2) ve Hilbert-Huang Dönüşümü (HHD) kullanılmıştır. Bu sinyal dönüşüm yöntemleri içerisinde HHD ile oluşturulmuş öznitelik kümesi üzerinde hem VIBES algoritması hem de diğer Makine Öğrenmesi algoritmaları yüksek doğruluk oranları vermiştir. Daubechies (db3) dalgacığı kullanılarak elde edilen öznitelik kümesi ikinci en iyi sınıflandırma doğruluğu veren sinyal dönüşüm yöntemi olmuştur. HHD kullanılarak oluşturulan Parkinson veri kümesine VIBES(KA) algoritmasının (özniteliklerin bağımsızlığı varsayımına); 2-katlı çapraz doğrulama, 5-katlı çapraz doğrulama, 10-katlı çapraz doğrulama ve Leave-One-Out-Çapraz-Doğrulama (LOOÇD) testlerinin 10 kere uygulanması sonucunda sırasıyla %95.4545±1.0785, %95.9394±1.3717, %95.4545±0.7693 ve %97.5758±0.0000 ortalama sınıflandırma doğrulukları elde edilmiştir. Bu testlerin ortalama sınıflandırma doğruluğu oranı ise %96.1061'dir. Bu deneylerin sonucunda LOOÇD yöntemi kullanılarak elde edilen topluluk hipotezi model olarak seçilmiştir. Bu model 14 temel hipotezden oluşmaktadır. Sonuç olarak literatürde daha önce yapılan 6 çalışmadan daha yüksek oranda sınıflandırma doğruluğu elde edilmiştir.

Anahtar Kelimeler: Makine Öğrenmesi, Topluluk Öğrenme, Hilbert-Huang Dönüşümü, Dalgacık Dönüşümü, Parkinson Hastalığı.

ANALYZING PARKINSON'S DISEASES DATA BY MACHINE LEARNING METHODS

ABSTRACT

This study is on Machine Learning algorithms applied to the Parkinson's Disease data. For this purpose, we planned developing a new and authentic Machine Learning algorithm. In this process; we developed a new Ensemble Learning algorithm, called *VIBES*, based on the Decision Fusion and the Classifier Selection methods. *VIBES* algorithm and other Machine Learning algorithms were applied to 33 benchmark datasets selected from UCI database and to Parkinson's Disease dataset selected from PhysioNet database.

VIBES algorithm operates according to two assumptions: that features are interdependent, and that features are independent of each other. According to these assumptions, average accuracy rates for 33 benchmark datasets are 89.80% and 88.22% respectively. The learning algorithm with the second best average accuracy is Rotation Forest algorithm, with 87.76% accuracy rate. The base learner used for both ensemble learning algorithms mentioned above is the Random Forests algorithm. We used the majority voting method as ensemble combination method for the *VIBES* algorithm, and only theoretically demonstrated that the majority voting method is better in comparison with other methods for only binary classification problems. We used an optimized forward search method and Genetic Algorithms as a hypothesis search algorithm for *VIBES(RF)* algorithm. According to the results regarding 33 benchmark datasets, the optimized forward search algorithm, with an average accuracy rate of 89.6963% and a search time of 3.2359 seconds, gave a better result in comparison with Genetic Algorithms. We got 89.4184% accuracy rate and 127.6958-second search time by using Genetic Algorithms. According to these results, the optimized forward search algorithm built the ensemble by doing the search in less time compared to Genetic Algorithms, although their accuracy rates were not significantly different. In measuring the amount of information of features in *VIBES(RF)* algorithm, we used two methods: ReliefF and Shannon entropy. The average accuracy rates of these algorithms are 89.8153% and 89.0519% respectively. We set Shannon entropy as a default method to measure the amount of information of features because of the fact that these two rates do not have a statistically significant difference and that Shannon entropy runs faster in comparison with ReliefF. In this thesis study, we developed a data-adaptive method to also predict the actual error rate of the ensemble built for ensemble learning with majority voting method. The difference between actual and predicted error rates is 0.575% in average for 33 benchmark datasets according to the method mentioned above and the assumption of feature interdependency, whereas it is 1.655% in average according to the assumption of feature independency. Thus, we have an average error rate of 1.115%. In light of these data, we can say that the error predicting method gives quite close results in relation to actual error rate.

We used Fast Fourier Transform, Wavelet Transform (haar, sym2, db2, db3, db4, db5, db6, db7, dmey, bio3.3, and gaus2), and Hilbert-Huang Transform so as to

create features for the Parkinson dataset. Both the *VIBES* algorithm and the other Machine Learning algorithms gave high classification accuracy for a set of features that was created by Hilbert-Huang Transform method. Besides, a set of features created through daubechies (db3) wavelet was a signal transform method that got the second best classification accuracy. When the *VIBES(KA)* algorithm (the assumption of the independency of features) was applied to the Parkinson dataset created using Hilbert-Huang Transform, and when 2-fold, 5-fold, 10-fold, and Leave-One-Out cross validations were applied to the dataset ten times, we received an average classification accuracy rate of $95.4545\% \pm 1.0785$, $95.9394\% \pm 1.3717$, $95.4545\% \pm 0.7693$, and $97.5758\% \pm 0.0000$ respectively. The average classification accuracy rate of these experiments is 96.1061%. Based on the results of our experiments, we selected as our model the ensemble hypothesis obtained using the Leave-One-Out cross validation method. This model consists of the hypotheses of 14 base learners. Consequently, we obtained a superior classification accuracy in comparison with six studies currently available in literature.

Keywords: Machine Learning, Ensemble Learning, Hilbert-Huang Transform, Wavelet Transform, Parkinson's Disease.

1. GİRİŞ

Geleneksel bilgi sistemleri ve uzman sistemler veri üzerinde derinlemesine bir analiz için yetersiz kalırken (Garcia 2014) bu kısıtlamanın üstesinden gelebilmek için daha sofistike sistemlere ihtiyaç duyulmaktadır. Sofistike bir sistemin oluşturulmasının ilk adımı bilgi keşfi yaklaşımıdır. Bilgi keşfi (veri madenciliği), veri üzerindeki örüntülerin ortaya çıkartılmasıyla ilişkili bir kavramdır. Bu örüntülerin ortaya çıkartılmasında çıkarımsal (tümevarımsal) algoritmalar kullanılmaktadır. Makine Öğrenmesi disiplini, tüm bu çıkarımsal algoritmaları içeren bir çalışma alanı olarak karşımıza çıkmaktadır (Kohavi & Provost 1998).

Makine Öğrenmesi bir dizi örnekten bir görev öğrenen mantıksal ya da ikili operasyonlara dayalı otomatik hesaplama yordamlarını kapsayan bir çalışma alanıdır (Michie et al. 2009). Makine Öğrenmesi; biyoloji, kimya, genetik, tıp, ekonomi, dil bilim ve finans gibi birbirinden oldukça farklı alanlarda kullanılmaktadır.

Tıp alanında Makine Öğrenmesinin uygulanması üzerine oldukça çok çalışma vardır. Bu çalışmalar özellikle “tıbbi teşhis” üzerine yoğunlaşmaktadır (Deo 2015). Tıp alanında geleneksel yöntemler yerine Makine Öğrenmesi tekniklerinin kullanımı çok güçlü sistemlerin oluşturulmasını sağlamıştır. Çünkü geleneksel sistemlerde bilgisayarlara kuralları biz veririz. Ancak Makine Öğrenmesi algoritmalarının kullanımıyla birlikte bilgisayarlara kurallar yerine sadece veriler verilir. Böylelikle verilerden kurallar çıkartma görevi algoritmalara bırakılmış olur. Sonuçta kural ezberleyen sistemler yerine kural öğrenen sistemler geliştirilmiş olur (Obermeyer & Emanuel 2016). Tıbbi teşhis problemini çözmek için geliştirilen bir Makine Öğrenmesi sisteminde şu özelliklerin olması beklenir: İyi performans, eksik verilerle başa çıkabilme, gürültülü verilerle başa çıkabilme, oluşturulan modelin saydam olması, sınırlı sayıda veriyle güvenilir sonuçların alınması (Kononenko 2001). Bu özellikler içerisinde, “oluşturulan modelin saydam olması” özelliği modelin türü ne olursa olsun doktorların üstesinden gelebileceği bir özellik olarak vurgulanmaktadır (Obermeyer & Emanuel 2016). Bu nedenle olmazsa olmaz bir özellik olmadığını belirtebiliriz. Bu özellik dışında geri kalan dört özelliğe sahip Makine Öğrenmesi

algoritmaları, yapılacak test ya da deneylerle ortaya çıkarılmalıdır. Diğer durumda, yeni Makine Öğrenmesi algoritması tasarımına gitmek ya da varolan bir yaklaşımı geliştirmek gerekmektedir.

Makine Öğrenmesi disiplininde, tüm gerçek dünya problemleri üzerinde en iyi öğrenmeyi gerçekleştirebilen tek bir Makine Öğrenmesi algoritması mevcut değildir. Bu nedenle algoritma seçimine, gerçekleştirilen deneylerle karar verilmelidir (Alpaydın 2014). Ayrıca algoritmaların eğitim kümesi üzerinde oluşturduğu model, bu eğitim kümesine bağlı olarak değişmektedir. Böylelikle, bu durumun bir sonucu olarak model seçiminde performans kriteri gündeme gelmektedir. Ayrıca en iyi öğrenme algoritmasının olmadığını, her algoritmanın çok başarılı olduğu veri kümeleri olduğu gibi çok başarısız olduğu veri kümeleri de vardır. Bu durum “no free lunch” (Wolpert 1996; Wolpert & Macready 1997) teoremi olarak bilinmektedir. Yani tüm veri kümeleri üzerinde en iyi performansı gösterecek olan evrensel bir öğrenme algoritması yoktur. Ayrıca “no free lunch” teoremi (NFLT) şunu da belirtmektedir: Herbir öğrenme algoritmasının tüm gerçek dünya problemleri üzerinde oluşturmuş olduğu modellerin ortalama performansları aynıdır. Bu teoremin sonuçları karşısında ilk akla gelen soru şudur: “Gerçek dünya problemleri üzerindeki tahmin başarımını arttırmak için klasik yaklaşımların aksine Makine Öğrenmesi alanında nasıl bir paradigma değişimine gitmek gerekir?”. Bu soruya verilebilecek en uygun cevap Topluluk Öğrenmesi yaklaşımıdır. Çünkü Topluluk Öğrenciler, klasik Makine Öğrenmesi algoritmalarından oldukça farklı bir öğrenme stratejisine sahiptir.

Topluluk Öğrenmesi, çoklu öğrencilerin aynı problemi çözmek için eğitildikleri bir Makine Öğrenmesi paradigmasıdır (Zhou 2009). Diğer bir ifadeyle Topluluk Öğrenmesi, çoklu öğrencileri eğitmek ve onların çıktılarını birleştirmek için uygulanan yöntem verilen addır. Topluluk Öğrenmesi yaklaşımının temel ilkesi şudur: Uygun bir biçimde birleştirilmiş bireysel tahmin edicilerin oluşturmuş olduğu topluluk kararının herhangi bir topluluk üyesinin kararından daha iyi olmasıdır. Birçok deneysel ve teorik çalışma da bu durumu kanıtlamaktadır. Her Makine Öğrenmesi algoritması belirli sınırlara sahiptir ve bu nedenle hatalar yapar. Topluluk Öğrenmesinin amacı en iyi kararı verirken algoritmaların güçlü ve zayıf yönlerini kontrol etmektir (Zhou 2009). Tahmin edicileri birleştirme yaklaşımı uzun bir süredir birçok disiplinin ilgisini çekmektedir. 200 yıldan uzun bir süre önce, az sayıda örnek gözlemi verilen bir olasılık dağılımının ortalamasının en iyi nasıl tahmin edileceği

üzerine tartışmalı bir soru ortaya çıktı. Pierre-Simon de Laplace (Laplace 1818) populasyon ortalamasını tahmin etmek için örnek ortalamasını kullanmanın daima en uygun yöntem olmadığını bunun yerine örnek medyanının daha iyi bir birleştirilmiş tahmin edici olduğunu gösterdi (Brown 2011). Bir Topluluk Öğrenmesinde topluluk oluşturulurken; topluluk modelinin nasıl oluşturulacağı, temel öğrencileri birleştirmek için kullanılacak yöntemler ve temel öğrencilerin özellik uzayını öğrenme stratejileri gibi tercihler topluluğun öğrenme başarımını ciddi bir biçimde belirlemektedir. Bu üç önemli aşama, aşağıda ana hatlarıyla açıklanmaktadır.

Topluluk Öğrenmesinin ilk aşaması topluluk modelinin nasıl oluşturulacağı ile ilgilidir. Topluluk Öğrenmesinde modeller oluşturulurken birçok öğrenci (Karar Ağaçları, Destek Yöney Makinaları vb.) tercih edilebilir. Bu öğrencilere temel ya da zayıf öğrenciler denir. Eğer Topluluk Öğrenmesinde tek bir temel öğrenci kullanılırsa buna homojen topluluk denir. Eğer Topluluk Öğrenmesinde birden fazla farklı temel öğrenci kullanılırsa buna da heterojen topluluk denir. Sonuç itibarıyla Topluluk Öğrenmesinin asıl amacı temel öğrencileri güçlendirmektir (Zhou 2012). Topluluk Öğrenmesi; güçlü varsayımları ve oluşturduğu basit topluluk modeliyle bilimsel araştırmanın temel varsayımlarından olan Occam'ın usturasını (Gauch 2002) ve Yunanlı filozof Epicurus'un (341 - 270 M.Ö.) çoklu açıklama ilkesini (Asmis 1984) karşılamaktadır. Ayrıca Hansen ve Salamon'un (Hansen & Salamon 1990) çalışmaları gürültülü veri karşısında bir topluluk öğrencisinin tek bir öğrenciye kıyasla daha dirençli olduğunu göstermiştir. Daha güçlü topluluk öğrencileri elde etmek için genellikle temel öğrencilerin mümkün olduğu kadar doğru tahminde bulunması ve oluşan her bir modelin birbirinden farklı olması gerektiğine inanılır. Ayrıca bir topluluk oluşturma maliyeti tek bir temel öğrenci oluşturma maliyetinden daha fazla değildir (Zhou 2012). Topluluk öğrencileri temel öğrencilerden daha güçlü kılan özelliklere kısaca değinecek olursak hiç şüphesiz bias, varyans, kovaryans ve gürültü arasındaki ilişkiye odaklanmak gerekmektedir (Ueda & Nakano 1996). Çünkü bir Topluluk Öğrencininin hata yapmasına bu dört etken neden olmaktadır. Topluluk öğrencilerin, temel öğrencilere kıyasla gürültüye karşı dirençli oldukları çeşitli çalışmalarda (Hansen & Salamon 1990) gösterilmiştir. Bunun yanı sıra Bagging (Breiman 1996) ve Boosting (Kearns 1988; Schapire 1990) gibi Topluluk Öğrenmesi yöntemleri sırasıyla varyans ve bias'ı azaltmak için

geliştirilmiş yöntemlerdir. Tüm bu özelliklerden dolayı Topluluk Öğrenmesinin temel öğrencilere göre daha güçlü olduğunu rahatlıkla söyleyebiliriz.

Temel öğrencilere ait modelleri birleştirmek için kullanılan yöntemler şunlardır: Doğrusal birleştirici, çarpımsal birleştirici ve oylamalı birleştiricidir. Bu birleştirme yöntemleri belirli eğitim kümeleri üzerinde en yüksek performansı elde etmek için seçilir. Ayrıca bu üç kuralın birçok problem üzerinde sürekli olarak iyi sonuç verdiği gösterilmiştir (Brown 2011).

Doğrusal birleştirici çıkışı sayısal değer olan modellerde ve sınıf-olasılık tahmini üreten sınıflandırma topluluklarında kullanılır. Sınıf-olasılık tahmini üreten topluluğun olasılık tahmini **(1.1)** denkleminde gösterilmektedir (Brown 2011).

$$\bar{f}(y|x) = \sum_{t=1}^T w_t f_t(y|x) \quad (1.1)$$

Bir diğer alternatif birleştirici çarpımsal birleştiricidir. Çarpımsal birleştirici için topluluğun olasılık tahmini **(1.2)** denkleminde gösterilmektedir (Brown 2011).

$$\bar{f}(y|x) = \frac{1}{Z} \prod_{t=1}^T f_t(y|x)^{w_t} \quad (1.2)$$

Bu denklemde Z sembolü \bar{f} 'in geçerli bir dağılım olmasını garantilemek için kullanılan bir normalizasyon faktörüdür. Burada dikkat edilmesi gereken husus, Z 'nin geçerli bir karar vermek için gerekli olmadığıdır. Çünkü sonsal tahminlerin sırası normalizasyondan önce ya da sonra değişmeden kalacaktır. Sınıf-koşullu olasılık tahminlerinin bağımsız olduğu varsayımı altında bu birleştirici, teorik olarak en uygun birleştirme stratejisidir. Ancak bu varsayım gerçek dünya problemleri ile örtüşmemektedir (Brown 2011). Doğrusal ve çarpımsal birleştiriciler gerçekte olası birleştirme stratejilerinin sürekliliğine izin veren genelleştirilmiş ortalamanın özel bir durumudur (Kuncheva 2004).

Oylamalı birleştirici çıkışı sayısal olmayan modeller için kullanılır. Herbir sınıflandırıcı belirli bir sınıfı oylar ve çoğunluk oyunu alan sınıf topluluğun çıkışını belirler. İkili bir sınıflandırma problemi için oylamalı birleştiricinin topluluk çıkışı **(1.3)** denkleminde gösterilmektedir (Brown 2011).

$$H(x) = \text{sign} \left(\sum_{t=1}^T w_t h_t(x) \right) \quad (1.3)$$

Temel öğreniciler özellik uzayından öğrenirken iki temel yöntemi uygular. Bunlar karar kaynaştırma (decision fusion) ve sınıflandırıcı seçimidir (classifier selection). Karar kaynaştırma da, her bir temel sınıflandırıcı tüm özellik uzayının bilgisine sahiptir. Sınıflandırıcı seçiminde ise, her bir temel sınıflandırıcı yerel bir uzmandır ve sadece özellik uzayının bir kısmını bilir. Ayrıca sınıflandırıcı seçim yaklaşımı, karar kaynaştırma yaklaşımı kadar üzerinde çok çalışılmış bir konu değildir (Kuncheva 2004).

1.1 Tezin Amacı ve Yaklaşımı

Bu tez çalışması Parkinson Hastalığı (PH) verileri üzerinde Makine Öğrenmesi yöntemlerinin uygulanması üzerinedir. Bu amaçla yeni bir Makine Öğrenmesi algoritması tasarımına gidildi. Bu tasarım sonucunda; karar kaynaştırma ve sınıflandırıcı seçimi yöntemlerine dayalı olarak geliştirilen yeni bir Topluluk Öğrenici, UCI veritabanından seçilen benchmark verikümesi ile PhysioNet veritabanından elde edilen Parkinson verikümesi üzerine uygulandı. Aynı zamanda geliştirilen algoritmanın sonuçları, diğer Makine Öğrenmesi algoritmalarıyla da karşılaştırıldı. Ayrıca çoğunluk oylamalı Topluluk Öğrenicilerde hata tahmini için bir yöntem önerildi. Böylelikle bir topluluğun tahmini hata oranı birleşmeden önce çok yakın bir biçimde tahmin edildi. Son olarak NFLT açısından geliştirilen algoritmanın teorik bir değerlendirilmesi gerçekleştirildi. Böylelikle herhangi bir veri kümesi üzerinde; geliştirilen algoritmanın, en kötü durumda bir temel öğrenici kadar iyi öğrenme gerçekleştirdiği gösterilmiş oldu.

Önerilen algoritmanın genel yaklaşımı şu şekildedir: Öncelikle veri kümesi için iki varsayımdan biri tercih edilir. Bu varsayımlar: Özniteliklerin birbirlerinden bağımsız olduğu ve özniteliklerin birbirlerine bağımlı olduğu varsayımlarıdır. Buna göre verilen bir $\mathcal{D} = \{\mathcal{X}, \mathcal{Y}\}$ $\mathcal{X} \in \mathbb{R}^{m \times n}$ eğitim kümesi için özniteliklerin birbirlerinden bağımsız olduğu varsayımına göre tüm \mathcal{L} temel öğrenicileri bu eğitim kümesine ait sadece bir özneliği öğrenir. Diğer varsayımda ise öncelikle öznitelikler Shannon entropi ya da ReliefF algoritmalarından herhangi biri ile en çok enformasyon verenden en az'a doğru sıralanır. Daha sonra ise temel öğreniciler sırasıyla; ilk bir,

ilk iki ve ilk n adet özniteliği öğrenir. Aynı zamanda her iki varsayım için de tüm veri kümesi girdi olarak kullanılır. Böylelikle bir \mathcal{X} girdisi için $n+1$ adet temel öğrenci elde edilir ve model çeşitliliği sağlanır. Daha sonra $n+1$ adet temel öğrenci içerisinde en yetkin topluluğu oluşturacak şekilde k adet temel öğrenci optimize edilmiş ileri yönlü bir arama algoritması kullanılarak seçilir ve bu temel öğrenciler çoğunluk oylama yöntemine göre birleştirilir.

1.2 Literatür Araştırması

Bu tez çalışmasının ana konusu iki alt kısımdan oluşmaktadır. Bunlar: yeni bir topluluk öğrencinin tasarlanarak benchmark veri kümesi üzerinde test edilmesi ve geliştirilen yeni algoritmanın parkinson veri kümesi üzerine uygulanmasıdır. Buna göre, literatürde bu tez çalışmasının kapsamı üzerine yapılmış olan çalışmalar aşağıda yer almaktadır.

1.2.1 Literatürde yer alan topluluk öğrenciler

Dasarathy ve Sheela (Dasarathy & Sheela 1979) tarafından yapılan çalışmada ilk olarak farklı girdiler için farklı sınıflandırıcılar kullanma fikri ortaya atıldı. Bu çalışmada temel sınıflandırıcı olarak doğrusal sınıflandırıcı ve bir k -en yakın komşu sınıflandırıcısı kullanılmıştır. Topluluk sınıflandırıcı, girdi alanındaki bir çatışma için k -en yakın komşu algoritmasını kullanırken; çatışmanın olmadığı alanlarda doğrusal sınıflandırıcı kullanılmaktadır.

Jacobs vd. (Jacobs et al. 1991) tarafından yapılan çalışmada her biri, eğitim kümesinin bir alt kümesini tutmayı öğrenen birçok ayrı ağdan oluşan sistemler için yeni bir gözetimli öğrenme metodu önerilmektedir. Yeni metot, çok katmanlı denetimli bir ağın modüler bir versiyonu olarak veya rekabetçi öğrenmenin birleştirici bir versiyonu olarak ortaya çıkmaktadır. Dolayısıyla bu iki farklı yaklaşım arasında yeni bir bağ oluşturulmuştur.

Pudil vd. (Pudil et al. 1992) tarafından yapılan çalışmada “reddetme seçeneği” olan çok kademeli bir model sınıflandırma sistemi oluşturma fikri sunulmuş ve çok kademeli bir sınıflandırıcı için bir üst kademeli ölçümlerin maliyetinin üst sınırları açısından, bir tek kademeli sınıflandırıcıya göre daha düşük karar riski verdiği gösterilmiştir.

Stacking (Stacked Generalization) algoritması (Wolpert 1992) öğrenme süreci boyunca *Bagging* ve *Boosting*'den farklı bir strateji izlemektedir ve Makine Öğrenmesi disiplinine meta öğrenici kavramını katmıştır. *Stacking* algoritması bir $D = \{x_i, y_i\}_{i=1}^m$ ($x_i \in \mathbb{R}^n, y_i \in Y$) eğitim kümesini girdi olarak alır ve üç aşamalı bir strateji izler. İlk aşamada D eğitim kümesi çapraz doğrulamaya benzer bir yöntemle iki parçaya ayrılır. Bu parçalardan biri olan $D_{training}$ eğitim kümesini, D_{test} ise test kümesini oluşturur. Data sonra $D_{training}$ kümesi kullanılarak T adet birbirinden farklı türde temel öğrenciler eğitilir. İkinci aşamada ise bu T adet temel öğrenci D_{test} kümesi üzerine uygulanarak D_{test} kümesinden D' veri kümesi oluşturulur. D' veri kümesi $D' = \{x'_i, y_i\}_{i=1}^m, x'_i = \{h_1(x_i), \dots, h_T(x_i)\}$ şeklindedir. Son aşamada ise bir meta öğrenci kullanılarak D' kümesinden sonuç hipotezi oluşturulur. *Stacking* algoritması çok sınıflı veri kümeleri üzerinde yüksek boyutlu meta veri kümesi oluşturduğundan dolayı yavaş bir öğrenme sağlamaktadır. Bu problemin üstesinden gelmek için Seewald (Seewald 2002) tarafından *StackingC* geliştirilmiştir.

Jordan ve Jacobs (Jordan & Jacobs 1994) tarafından yapılan çalışmada gözetimli öğrenme için ağaç yapılı bir mimari önerilmektedir. Mimarinin altında yatan istatistiksel model, hem karışım katsayıları hem de karışım bileşenleri genelleştirilmiş doğrusal modeller olan hiyerarşik bir karışım modelidir. Mimarinin parametrelerini ayarlamak için bir Beklenti Maksimizasyon algoritması önerilmiştir.

Bagging (**B**ootstrap **a**ggregating) (Breiman 1996) algoritması bir veri kümesinden tekrar örnekleme yapılarak yeni kümelerin oluşturulması ve bu kümelere temel öğrencilerin uygulanması esasına dayanır. *Bagging* algoritmasında veri kümesinin farklı birleştirmelerini oluşturma süreci *Bootstrap* (Efron & Tibshirani 1994) metoduna dayanır. Bu metod çapraz doğrulama yöntemine benzemektedir ve onun bir alternatifidir. Örneğin N adet örneğe sahip bir D eğitim kümesi için *Bagging*, M adet yeni D_i eğitim kümeleri oluşturur. Yeni oluşan bu eğitim kümelerinin örnek sayısı N adettir. Ayrıca yeni eğitim kümeleri orjinal eğitim kümesi olan D kümesinden oluşturulur. Bundan dolayı yeni oluşan eğitim kümeleri birbirinin aynısı olabilir. Şimdi N örnekten *Bootstrap* yöntemi ile N kez örnek seçtiğimizi düşünelim. N örnekten herhangi bir örneği seçme olasılığımız $\frac{1}{N}$, seçmeme olasılığı ise $1 - \frac{1}{N}$ olur. N kez seçim yaptıktan sonra bir örneğin hiç seçilmeme olasılığı (1.4) teki gibi olur.

$$\left(1 - \frac{1}{N}\right)^N \approx e^{-1} = 0.368 \quad (1.4)$$

Bu da, D_i kümelerinin orjinal küme olan D kümesinin yaklaşık olarak %63.2'sini içerdiğini ve verinin %36.8'inin öğrenmede hiç kullanılmadığını göstermektedir. Bunu engellemenin yolu örnek seçme sayısı N 'i arttırmaktır. Böylelikle örneklerin seçilmeme olasılığı düşecektir. *Bagging* yöntemi ile oluşturulan her bir model ağırlıklı ortalama kullanılarak birleştirilir ve nihai model oluşturulur. *Bagging* yöntemi özellikle stabil olmayan yani varyansı yüksek sınıflandırıcıların başarımını arttırmak için kullanılmaktadır (Breiman 1996). Ancak bazı örneklerin hiç seçilmemesi *Bagging*'in dezavantajı olarak görülmektedir. İterasyon sayısı N 'i arttırmak bu problemi çözsede sınıflandırıcının eğitim zamanını arttırmaktadır.

“*Boosting*” herhangi bir öğrenme algoritmasının performansını arttıran genel bir metottur. *Boosting*, eğitim kümesinden tekrar örneklenerek elde edilen çeşitli dağılımlar üzerinde zayıf bir öğrenme algoritmasının (temel öğrenici) uygulanması esasına dayanır. Daha sonra bu zayıf öğreniciler birleştirilerek tek bir tümleşik sınıflandırıcı elde edilir (Freund & Schapire 1996). İlk olarak kanıtlanabilir biçimde etkili *Boosting* algoritmaları Schapire (Schapire 1990) ve Freund (Freund 1995) tarafından sunuldu. Daha sonra ise birçok *Boosting* algoritması geliştirildi. Bu algoritmalarından biri olan *AdaBoost.M1* algoritması Schapire ve Freud (Freund & Schapire 1996) tarafından geliştirildi. *AdaBoost.M1* algoritması bir $S = \{x_i, y_i\}_{i=1}^m$ ($x_i \in \mathbb{R}^n, y_i \in Y$) eğitim kümesini girdi olarak alır. Daha sonra bu girdinin her örneği $t = 1, \dots, T$ adet iterasyon boyunca tekrar ağırlıklandırılır. Başlangıç durumunda $i = 1, \dots, m$ olacak şekilde tüm örnekler için ağırlıklar (1.5) deki gibi hesaplanır.

$$D_1(i) = \frac{1}{m} \quad (1.5)$$

Daha sonra temel sınıflandırıcı çağrılarak $h_t: X \rightarrow Y$ oluşturulur ve (1.6) daki gibi h_t nin hatası hesaplanır.

$$\varepsilon_t = \sum_{i: h_t(x_i) \neq y_i} D_t(i) \quad (1.6)$$

Eğer $\varepsilon_t > 1/2$ ise döngüden çıkılır. Diğer türlü döngüye devam edilir. Bir sonraki aşamada (1.7) deki gibi temel öğrenicilere ait katsayılar hesaplanır.

$$\beta_t = \varepsilon_t / (1 - \varepsilon_t) \quad (1.7)$$

Bir sonraki iterasyon için örneklerin yeni ağırlıklıkları Z_t ile normalize edilerek (1.8) deki gibi hesaplanır.

$$D_{t+1}(i) = \frac{D_t(i)}{Z_t} x \begin{cases} \beta_t & h_t(x_i) = y_i \\ 1 & \text{Diğer türlü} \end{cases} \quad (1.8)$$

Bu işlem bütün iterasyon boyunca aynı şekilde tekrarlanır. Böylelikle T adet iterasyon sonunda h_1, \dots, h_T hipotezleri elde edilir. Daha sonra bu hipotezler ağırlıklı oylama yöntemine göre birleştirilerek (1.9) da gösterildiği gibi sonuç hipotezi oluşturulur.

$$h_{fin}(x) = \arg \max_{y \in Y} \sum_{t: h_t(x)=y} \log_2 \frac{1}{\beta_t} \quad (1.9)$$

AdaBoost başta olmak üzere birçok *Boosting* algoritması gürültülü verilere karşı dirençli değildir. *MadaBoost* (Domingo & Watanabe 2000) ve *LogitBoost* (Friedman 1999) gibi algoritmalar *AdaBoost*'un bu gürültüye karşı zayıf olan toleransını yükseltmek için çeşitli yöntemler içermektedir. Ancak Konveks Potansiyel Boosting algoritmaları iterasyonu uygun bir noktada bitirme konusunda etkili çözümler geliştirememişlerdir (Long & Servedio 2010). Böylelikle iterasyon boyunca uygun durma noktasını aşarak gürültülü verileri öğrenmeye başlarlar.

Koppel ve Engelson (Koppel & Engelson 1996) tarafından yapılan çalışmada birkaç temel sınıflandırıcıyı tek bir sınıflandırıcı oluşturmak için nasıl entegre edilebileceği problemine çözüm olarak her sınıfın uzmanlık alanını analiz etmeye dayanan bir yaklaşım önerilmiştir.

Woods vd. (Woods et al. 1997) tarafından yapılan çalışmada bilinmeyen bir test örneği etrafındaki özellik alanının küçük bölgelerinde her bir sınıflandırıcının yerel doğruluğunun tahminlerini kullanarak sınıflandırıcıları birleştiren bir yöntem önerilmektedir.

Alpaydin (Alpaydin 1997) tarafından yapılan çalışmada k-en yakın komşu sınıflandırıcısı gibi tembel öğrenme yöntemlerinin dezavantajlarından biri olan tüm

eđitim setini saklama zorluđuna, bir özüm önerisi getirilmiřtir. oklu yođunlařtırılmıř en yakın komřu sınıflandırıcı, örneklerin bir alt kümesini ařamalı olarak saklar ve böylece örneklerin depolama ve hesaplama gereksinimlerini azaltır. Bu gibi alt grupları birden fazla eđitmek ve onlara oy vermek için iki oylama řeması kullanılmıřtır: oyların eřit ađırlıđa sahip olduđu basit oylama ve ađırlıkların, sınıflandırıcıların tahminlerine bađlı olarak sıralamaya bađlı olduđu ađırlıklı oylamadır.

Kittler vd. (Kittler et al. 1998) tarafından yapılan alıřmada bir eđitim kümesi üzerinde T adet farklı sınıflandırıcının yapmıř olduđu tahminler eřitli yöntemler kullanılarak birleřtirilmiřtir. Bu yöntemler yapılan tahminlerin olasılıklarının toplamı, arpımı, maksimumu, minimumu, medyanı ve ođunluk oylanmasından oluřmaktadır. Böylelikle bu alıřmada Topluluk Öđrenmede temel öđrenicilerin nasıl birleřtirileceđi sorusuna cevap verilmiřtir.

Frank vd. (Frank et al. 1998) tarafından yapılan alıřmada sınıflandırma için model ađaçlarının kullanımı önerilmektedir. Yapraklarda dođrusal regresyon fonksiyonlarına sahip bir karar ađacı türü olan model ađaçları, sürekli sayısal deđerleri tahmin etmek için kullanılmaktadır. Model ađaçları, bir sınıflandırma problemini bir fonksiyon yaklařımı problemine dönüřtürerek sınıflandırma problemlerini özer. Bunu yaparken Quinlan'ın $M5$ (Quinlan 1992) algoritmasına dayanan $M5'$ (Wang & Witten 1997) algoritmasını kullanır. Niteliklerin ođunun sayısal olması durumunda $C4.5$ algoritması (Quinlan 1993), kendisinin geliřtirilmiř ticari sürümü olan $C5.0$ algoritmasından daha iyi sonuçlar vermektedir (Frank et al. 1998).

Ho (Ho 1998) tarafından geliřtirilen *Random Subspace* algoritması birok karar ađacının biraraya getirilerek birleřtirilmesi esasına dayanmaktadır. Karar ađaçları üzerine yapılan alıřmalar daha ok bölme kriterleri ve ađaç boyutu üzerinedir. Az da olsa ařırı uyum ve maksimum dođruluk ikilemi üzerine odaklanan alıřmalar olmuřtur. Bu alıřmada bu ikilem üzerine odaklanılarak, eđitim verisi üzerinde en yüksek dođruluđa sahip ve karmařıklıđı arttıķça genelleme dođruluđu artan bir karar ađacı tabanlı sınıflayıcı oluřturmak için bir yöntem önerilmiřtir. *Random Subspace* algoritması, özellik vektörünün rasgele seilen bileřenlerinden oluřan yeni eđitim kümelerinin herbirinin bir karar ađacı tarafından öđrenilmesi ve oluřan modellerin ortalamasının nihai modeli oluřturması esasına dayanır. *Random Subspace*, rasgele

özellik seçimi ile yüksek boyutlu veri kümeleri üzerinde yüksek hızlarda çalışma özelliğine sahiptir.

Gama (Gama 1998a; Gama 1998b) tarafından yapılan çalışmada sınıflandırıcıları birleştirmek için *Cascade Generalization* isminde yeni bir genel yöntem sunulmuştur. *Cascade Generalization*'ın temel fikri, yeni nitelikler ekleyerek orijinal veri kümesinin bir uzantısını sıralı bir biçimde gerçekleştiren sınıflandırıcıların kümesini oluşturmaktır. Yeni özellikler bir temel sınıflandırıcı tarafından verilen olasılık sınıfı dağılımından türetilir. Bu adım, yüksek seviye sınıflandırıcılar için temsili dili genişletir ve bias'ı azaltır. *Cascade Generalization*, temel sınıflandırıcıların model sınıfı temsilini birleştiren veriler için tek bir yapılanmış model oluşturur.

Alpaydin ve Kaynak (Alpaydin & Kaynak 1998) tarafından yapılan çalışmada doğrusal parametrik model ve k-en yakın komşu parametrik olmayan sınıflandırıcısının ardışık olarak inşa edilmesini öneren bir yöntem önerilmektedir. Doğrusal model bir "kural" öğrenir ve k-En Yakın Komşuluk (k-EYK) "kural" tarafından reddedilen "istisnaları" öğrenir.

Bay (Bay 1999) tarafından yapılan çalışmada en yakın komşu sınıflandırıcısının doğruluğunu arttırmak için tasarlanmış bir birleştirme algoritması olan *Çoklu Öznitelik Altkümelere (ÇÖA)* tanıtılmaktadır. ÇÖA, her biri rastgele bir özellik alt kümesini kullanan birden çok en yakın komşu sınıflandırıcısını birleştirir. Deney sonuçları birkaç standart en yakın komşu algoritmasından belirgin biçimde daha iyi performans göstermekte ve artırılmış karar ağaçlarıyla rekabet etmektedir.

Liu ve Yao (Liu & Yao 1999) tarafından yapılan çalışmada sinir ağı toplulukları için negatif korelasyon öğrenimi isimli yeni bir öğrenme yaklaşımı sunulmaktadır. Sinir ağı toplulukları için önceki öğrenme yaklaşımlarının aksine, negatif korelasyon öğrenme, bir topluluktaki bireysel ağları eğitmeye çalışır ve onları aynı öğrenme sürecinde birleştirir. Negatif korelasyon öğrenmesinde, topluluktaki tüm bireysel ağlar, hata fonksiyonlarındaki korelasyon ceza terimleri yoluyla eşzamanlı ve etkileşimli olarak eğitilir. Aynı zamanda hataları birbiriyle ilişkisiz olan bireysel ağlar üretmek yerine bireysel ağlar arasındaki uzmanlaşmayı ve işbirliğini sağlamak için negatif ilişkili ağlar oluşturulur.

Merz (Merz 1999) tarafından yapılan çalışmada temel sınıflandırıcıların tahminlerini birleştirmek için bir yöntem önerilmektedir. Önerilen yöntem, eğitim kümesi ile onların sınıflandırılması arasındaki ilişkiyi modellemek için *Stacking* ve *Correspondence Analysis* stratejilerini kullanır. Ardından daha önce görülmeyen örnekleri sınıflandırmak için sonuç gösteriminde en yakın komşu yöntemi uygulanır.

Egmont-Petersen vd. (Egmont-Petersen et al. 1999) tarafından yapılan çalışmada geriye doğru arama yapan bir özellik seçimi prosedürü tanımlanmış ve her sınıflayıcının önceki iterasyondaki sınıflandırıcıdan daha küçük bir özellik alt kümesini kullanması sağlanmıştır. Daha sonra sınıflandırıcılar ardışık halde birleştirilir.

Frank ve Hall (Frank & Hall 2001) tarafından yapılan çalışmada gerçekte sınıf değerleri sıralı olan Makine Öğrenmesi problemlerinin çözüm performansını arttırmak için bir yöntem önerilmiştir. Genelde Makine Öğrenmesi algoritmaları sınıf değerlerini sırasız kabul etmektedir. Bu çalışmada standart sınıflandırma algoritmalarının sınıf niteliklerinde sıralama bilgisi kullanmasını sağlayan basit bir yöntem sunulmuştur.

Breiman (Breiman 2001) tarafından yapılan çalışmada ağaç tahmin edicilerin birleştirilmesine dayalı *Random Forests* isimli bir algoritma tanıtılmıştır. Algoritmada ormanlar için genelleme hatası ormandaki ağaç sayısına göre bir limite yakınsar. Sadece bir orman için genelleme hatası ormandaki bireysel ağaçların gücüne ve onlar arasındaki ilişkiye bağlıdır. Herbir düğümü ayırmak için rastgele bir özellik seçimi kullanma AdaBoost'a tercihen daha uygun hata oranları verir. Ayrıca AdaBoost'a kıyasla gürültüye karşı daha dirençlidir.

Seewald ve Fürnkranz (Seewald & Fürnkranz 2001) tarafından yapılan çalışmada temel seviyedeki doğru ve yanlış tahminleri tespit etmeye çalışan bir meta sınıflandırma tekniği olan *Grading* yöntemi tanıtılmaktadır. *Stacking* temel sınıflandırıcıların tahminlerini meta-seviye nitelikler olarak kullanırken, *Grading* meta-seviye sınıflar olarak "dereceli" tahminleri (yani doğru veya yanlış olarak işaretlenmiş tahminler) kullanır.

Frank vd. (Frank et al. 2002) tarafından yapılan çalışmada büyük veri kümeleri üzerinde öğrenme gerçekleştiren temel sınıflandırıcıların biraraya getirilmesiyle oluşturulan bir topluluk öğrenme stratejisi önerilmektedir. Önerilen yöntemde temel

sınıflandırıcı bunu etkili bir şekilde yapamasa bile büyük veri kümelerinin işlenmesine izin verir. Temel fikir, gelen verileri parçalara ayırmak ve bu parçalardan oluşan sınıflandırıcılara dayalı bir topluluk oluşturmaktır. Çalışmada, oluşturulan topluluğu uygun bir şekilde budamak için bir yöntem sunulmuştur. Bu algoritmayı pratikte uygularken çok önemlidir çünkü algoritmanın çalışma süresini ve bellek tüketimini önemli ölçüde azaltır.

Melville ve Mooney (Melville & Mooney 2003) tarafından yapılan çalışmada bir topluluğu oluşturan üyelerin çeşitliliğini arttırmaya yönelik *DECORATE* isimli bir yöntem önerilmiştir. Çeşitliliği arttırmak için yapay olarak oluşturulmuş eğitim örnekleri kullanılmıştır. Temel sınıflandırıcı asıl eğitim kümesi ile birlikte yapay eğitim kümesini de kullanarak bir model oluşturur. Daha sonra bu modeller birleştirilerek topluluk öğrenici elde edilir.

Ferri vd. (Ferri et al. 2004) tarafından yapılan çalışmada bir sınıflandırıcının sınıflandırma görevini sistematik bir şekilde başka bir sınıflandırıcıya devretme fikri analiz edilmiş ve geliştirilmiştir. İlk olarak, birinci sınıflandırıcı sınıflandıracağı örnekleri seçer ve zor örnekleri sınıflandırmak için ikinci bir sınıflandırıcıya bu görevi devreder. Ayrıca çalışmada, yinelemeli olarak keyfi sayıda zincir sınıflandırıcı içeren bir senaryoda sunulmuştur.

Frank ve Kramer (Frank & Kramer 2004) tarafından yapılan çalışmada çoklu sınıf problemleri için iç içe ikili ayrımlar (*Nested Dichotomies*) yöntemi önerilmiştir. Bu yöntem, L tane sınıfa sahip bir sınıflandırma problemini özyinelemeli bir biçimde daha küçük alt kümelere ayırarak bir ikili ağaç oluşturur. İkili ağaçların sayısı $2^L - 1$ dir. Sonuç modeli, bu ağaçların sınıf olasılık tahminlerinin ortalaması alınarak oluşturulur. Dong vd. (Dong et al. 2005) iç içe ikili ayrımlar yöntemini geliştirerek dengeli iç içe ikili ayrımlar yöntemini önermişlerdir. Bu yöntemin sınıflandırma doğruluğuna etkisi ya çok az ya da yoktur. Yalnız bu yöntem sınıflandırıcının eğitim zamanını düşürmektedir.

Oliveira vd. (Oliveira et al. 2005) tarafından yapılan çalışmada basamaklı sınıflandırıcı sistemleri için sınıfla ilişkili reddetme eşikleri sorunu ele alınmıştır. Literatürde, sınıfla ilgili reddetme eşiklerinin tek bir global eşikten daha iyi bir hata-reddetme dengesi sağladığı gösterilmiştir. Bu çalışmada, eşikleri bulmak için uygun bir algoritma kullanılabilirse, hata-reddetme değişiminin sınıfla ilişkili reddetme

eşikleriyle üretilebileceği iddia edilmektedir. Bunun ışığında, parçacık sürüsü optimizasyonu olarak adlandırılan bir optimizasyon algoritmasının kullanımı da önerilmektedir.

Ruta ve Gabrys (Ruta & Gabrys 2005) tarafından yapılan çalışmada sınıflandırıcı seçim metodolojisinin gözden geçirilmesi ve çoğunluk oylamasıyla sınıflandırıcıları birleştirme bağlamında çeşitlilik ölçümlerinin uygulanabilirliğinin değerlendirilmesi üzerinde durulmaktadır. Ayrıca çalışmada, çoğunluk oylama hatası ve çeşitlilik ölçümlerini de içeren bir dizi seçim kriteri ile düzgün çalışması için bir dizi arama algoritması önerilmektedir.

Rodriguez vd. (Rodriguez et al. 2006) tarafından yapılan çalışmada *Rotation Forest* isimli yeni bir Topluluk Öğrenme yöntemi tanıtılmıştır. Bu yöntem T adet sınıflandırıcının herbiri için şu adımları yerine getirir: Eğitim kümesi K adet yeni kümeye ayrılır. Bu ayırma işlemi özellik kümeleri oluşturularak gerçekleştirilir. Daha sonra bu özellik kümeleri için Temel Bileşen Analizi (Pearson 1901) çalıştırılarak katsayılar elde edilir. Eğitim kümesi olarak bu katsayılar kullanılır ve sınıflandırıcılar eğitilir. Sınıflandırıcıların birleştirilmesi ise temel sınıflandırıcıların herbir sınıfa göre olasılık tahminlerinin ortalaması alınarak gerçekleştirilir.

Wanas vd. (Wanas et al. 2006) tarafından yapılan çalışmada temel sınıflandırıcıları birleştirmeyi daha uyarlanabilir bir süreç haline getirmek için bir yöntem sunulmuştur. İlk olarak, karar birleştirme sürecini yönlendirmek için bir problemin özelliklerinden yararlanarak yeni bir mimari önerilmektedir. Hem özellikleri hem de sınıflandırıcı çıktılarını kullanarak, farklı sınıflandırıcıların güçlü ve zayıf yönleri tespit edilir. Bu bilgi, sistemin genelleme kabiliyetini arttırmak için kullanılır. Ayrıca, mimarinin bileşenleri üzerinde daha ileri eğitimin yapılması gerekip gerekmediğini belirlemek için nihai sınıflamaya olanak tanıyan bir kooperatif eğitim algoritması da önerilmektedir.

Zhang ve Street (Zhang & Street 2008) tarafından yapılan çalışmada bazı mevcut tekniklerin özelliklerini birleştiren yeni bir topluluk öğrenme algoritması önerilmektedir: *Bagging*, *Arcing* ve *Stacking*. Algoritmalarının temel yapısı *Bagging*'e benzemektedir. Bununla birlikte, her bir eğitim noktasının yanlış sınıflandırma maliyeti, gözlemlenen yanlış oylamalara göre tekrar tekrar ayarlanır.

Bu yolla yöntem, gürültülü verilere takılmadan topluluğun ihtiyaç duyduğu sınıflandırıcıları oluşturmak için *Arcing*'in avantajını kullanmaktadır.

Maudes vd. (Maudes et al. 2012) tarafından yapılan çalışmada karar ağaçları topluluklarından, *Rastgele Öznitelik Ağırlıkları (RÖA)* oluşturmak için bir yöntem önermektedir. Yöntem, *Random Forest*'a benzer. Ancak *Random Forest*'da yalnızca her düğüm için rasgele bir öznitelik altkümesi dikkate alınırken, *RÖA* hepsini dikkate alır. Rastsallığın kaynağı, her bir nitelikle ilişkili bir ağırlıktır. Ağacın tüm düğümleri aynı rastgele ağırlık gruplarını kullanır, ancak diğer ağaçlardaki ağırlık gruplarından farklıdır. Bu durumda, özelliklere verilen önem her ağaçta farklı olmaktadır. Böylelikle ağaçların inşası birbirinden farklı olmaktadır.

Song vd. (Song et al. 2013) tarafından yapılan çalışmada bir *Random Forests* sınıflandırıcının avantajlarını (mükemmel öngörü doğruluğu, özellik önem ölçütleri, doğruluğun eğitim kümesi dışındaki örnekleri tahmin edebilmesi gibi) ileri yönlü seçilen geliştirilmiş doğrusal modellerle (yorumlanabilirlik) paylaşan *Rastgele Genelleştirilmiş Doğrusal Model (RGDM)* isimli bir topluluk sınıflandırıcı önerilmiştir.

Abellán (Abellán 2013) tarafından yapılan çalışmada kesin olmayan olasılıklar ve belirsiz ölçümler ile oluşturulan karar ağaçlarını birleştirmek için farklı stratejiler arasından deneysel bir karşılaştırma sunulmaktadır. Sınıf değişkeninin her bir özelliğindeki bilgi derecesi kullanılarak kök düğümün değişimine dayanan *Bagging* ve *Boosting* olarak bilinen önceden geliştirilmiş şemalar kullanılmıştır. Bu amaçla, eksik veri ve sürekli değişkenlerle başa çıkmak için iki farklı yaklaşım uygulanmıştır.

Díez-Pastor vd. (Díez-Pastor et al. 2014) tarafından yapılan çalışmada ağaç topluluğu inşası için iki yeni yöntem önerilmektedir: *G-Forest* ve *GAR-Forest*. Bu çalışmada *Random Forest*'a benzer şekilde, ağaç yapım süreci bir derece rastgelelik içermektedir. Rastgele ve uyarlanabilir çözümler üretmek için *GRASP*'da kullanılan aynı strateji, ağaçların her bir düğümünde kullanılmaktadır. Topluluğun çeşitliliğinin kaynağı, *GRASP*'ın çözüm üretme yönteminin rastsallığında yatmaktadır. *GAR-Forest* için ağaç yapım yönteminin bir başka önemli özelliği, ağacın oluşturulması sürecinde azalan bir rastsallık seviyesidir: kökte maksimum rastsallık ve yapraklarda minimum rastsallık.

Kuncheva ve Rodriguez (Kuncheva & Rodríguez 2014) tarafından yapılan çalışmada; çoğunluk oylama, ağırlıklı çoğunluk oylama, recall birleştirici ve Naive Bayes (NB) birleştirici olmak üzere dört kombinasyon yöntemi için en uygunluk koşulları (minimum sınıflandırma hatası) veren sınıflayıcı kombinasyonu için olasılıklı bir çerçeve önerilmektedir. Çerçeve iki varsayım üzerine kurulmuştur: Sınıflandırıcı çıktılarının sınıf-koşullu bağımsızlığı ve bireysel doğruluklar hakkında bir varsayım.

Mayhua-López vd. (Mayhua-López et al. 2015) tarafından yapılan çalışmada Destek Yöney Makinelerinin gerçek bir *AdaBoost* topluluğu oluşturmak için temel sınıflandırıcılar olarak kullanılması önerilmekte ve sonuç topluluğunun, aynı çekirdeklere karşılık gelen ağırlıkları basitçe birleştirerek monolitik bir mimaride toplaması gerektiği ifade edilmiştir.

Sesmero vd. (Sesmero et al. 2015) tarafından yapılan çalışmada çok sayıda özelliğe sahip çok sınıflı problemlere odaklanan BCE isminde bir topluluk önerilmektedir. Önerilen yöntemde hesaplama maliyetini azaltmak için bir öznitelik seçim modülü vardır. BCE; temel öğrencilerin doğruluğuna, onların çeşitlilikleri üzerinden karar vermektedir.

Duan vd. (Duan et al. 2016) tarafından yapılan çalışmada Sınıflandırma ve regresyon ağaçları (*CART*) topluluğunu kullanan yeni bir "ağaç ortalaması" modeli önerilmektedir. Her bir kurucu ağaç, benzer verilerin bir alt kümesiyle tahmin edilir. Bu alt grupların gruplandırılmasını *Bayesian Topluluk Ağaçları (BTA)* olarak değerlendirilmekte ve bunlar bir Dirichlet süreci olarak modellenmektedir.

Gu ve Jin (Gu & Jin 2017) tarafından yapılan çalışmada farklı sınıflandırma modelleri ve/veya farklı özellikler kullanan bir dizi heterojen sınıflandırıcı üreten, *Multi-Train* olarak adlandırılan yeni bir yarı denetimli topluluk öğrenme algoritması önerilmektedir. Bu çalışmada, eğitim süreci boyunca her sınıflandırıcı geri kalan sınıflandırıcıların çoğunluk tahmini ile etiketlenmiş olan etiketsiz veriyi kullanarak rafine edilir. Ayrıca farklı modellerin ve farklı girdi özelliklerinin kullanılması topluluğun çeşitliliğini arttırmaktadır.

Abuassba vd. (Abuassba et al. 2017) tarafından yapılan çalışmada *Düzenlenmiş-Ekstrem Öğrenme Makinesi (Düzenlenmiş-EÖM)*, *L2-norm-optimize EÖM (ELML2)* ve *Kernel-EÖM*'yi içeren heterojen bir *İleri EÖM Topluluk (İEÖMT)* önerilmektedir.

Topluluk, rastgele seçilen *EÖM* sınıflandırıcısının; rasgele yeniden örnekleme yoluyla seçilen eğitim verilerinin bir alt kümesinde eğitilmesi suretiyle oluşturulmaktadır. Önerilen *İEÖMT*, sonuç topluluk arasında artan çeşitlilik ve doğruluğun bir hedef fonksiyonunu kullanarak geliştirilmiştir. Ayrıca, test verilerinin sınıf etiketi çoğunluk oyu yaklaşımı kullanılarak öngörülmektedir.

1.2.2 Literatürde yer alan PH'nin teşhisi ile ilgili çalışmalar

Lee ve Lim (Lee & Lim 2012) tarafından yapılan çalışmada idiyopatik Parkinson Hastalığı (PH) hastalarının yürüme özellikleri ve dalgacık tabanlı öznelik çıkarımı kullanılarak idiyopatik PH hastalarının sağlıklı bireylerden ayrılması için bir yöntem önerilmektedir. Çalışmada kullanılan veriler, bu tez çalışmasında da kullanılan Parkinson veri kümesi verileridir. Çalışmada sınıflandırma algoritması olarak Ağırlıklı Bulanık Üyelik Fonksiyonlu Sinir Ağları kullanılmıştır. Ayrıca istatistiksel yaklaşımlar kullanılarak 40 öznelik elde edilmiştir. Çalışmada uygulanan yöntemler içerisinde en yüksek sınıflandırma doğruluğu %77.33'tür.

Barut vd. (Barut et al. 2012) tarafından yapılan çalışmada PH hastalarının motor faaliyetlerini değerlendiren parmak vuru hızı bir bilgisayar sistemi yardımı ve geliştirilen bir yöntemle sayısal olarak ölçülmüştür. ROC analizi kullanılarak gerçekleştirilen değerlendirme de parmak vuru hızının; PH hastalarında kol-motor faaliyetinin ve hastalık şiddetinin değerlendirilmesinde kullanılabilecek bir yöntem olduğu gösterilmiştir. Sonuç olarak parmak vuru hızı yöntemin; günlük işlerde, pratik işlerde ve klinik araştırmalarda PH'nin değerlendirilmesinde kullanılabileceği belirtilmiştir.

Ishigami vd. (Ishigami et al. 2012) tarafından yapılan çalışmada PH ile Multisistem Atrofi (MSA) arasındaki ayırıcı tanı için bir proteomik profil stratejisi geliştirilmiştir. Bu yöntemde öncelikle cerebros spinal fluid peptitlerin ayrıştırılmasına dayalı olarak manyetik ayırıcı kullanılmıştır. Sonrasında kütle spektrometri analizi kullanılarak hastalara ait çok boyutlu özellik uzayı elde edilmiştir. Veri kümeleri yüksek boyutta olduğu için Temel Bileşen Analizi (TBA) kullanılarak yüksek boyutlu uzay, düşük boyutlu bir uzay ile temsil edilmiştir. Son olarak bu üç sınıfı (PH, MSA ve kontrol) birbirinden ayıracak model için Destek Yöney Makinaları (DYM) kullanılmıştır.

Jesse vd. (Jesse et al. 2012) tarafından yapılan çalışmada PH, MSA, Progresif Supranükleer Palsi (PSP) ve Kortikobazal Dejenerasyon (KBD) arasındaki ayırıcı

tanı için non-invazif bir yöntem olan Manyetik Rezonans Görüntüleme (MRG) yöntemi kullanılmıştır. Bu çalışmada MRG yöntemi belli başlı patolojileri dışarıda tutmak için tercih edilmiştir. Ayrıca çalışmada, Region of interest (ROI) için Substantia Nigra (SN) ve Globus Pallidus Internus (GPI) seçilmiş ve bu bölgelerin MRI çekimleri yapılmıştır. Farklı gruplar (SN ve GPI) arasındaki sinyal yoğunluklarının karşılaştırılmasında ANOVA testi uygulanmıştır.

Tsukamoto vd. (Tsukamoto et al. 2012) tarafından yapılan çalışmada PH'yi; MSA ve PSP'den ayıran ayırıcı tanı için 3-T MR görüntüleme tekniği kullanılmıştır. Çalışmada, 3-T MR kullanılarak her bir bireyin Eksen Difüzyon Ağırlıklı görüntüsü ve T2 ağırlıklı görüntüsü elde edilmiştir. Daha sonra bölgesel Görünür Difüzyon Katsayısı (bGDK) her bir ROI için hesaplanmıştır (ROI analizi beynin birçok lokasyonuna uygulanmıştır). Son olarakta MSA, PSP, PH ve kontrol grupları için ortalama bGDK değerlerini karşılaştırmak için Kruskal Wallis testi uygulanmıştır.

Daliri (Daliri 2013) tarafından yapılan çalışmada sağlıklı bireyler ile Parkinsonlu bireylerin birbirinden ayrılması için yeni bir yaklaşım önerilmektedir. Bu çalışmada kullanılan veriler, bu tez çalışmasında da kullanılan Parkinson veri kümesi ile aynıdır. Bu yöntem, deneklerin normal yürüyüşü esnasında yere uygulanan reaksiyon kuvvetinin sayısallaştırılarak ölçülmesi esasına dayanmaktadır. Bu ölçümler, her ayağın altına yerleştirilen 8 sensör kullanılarak hesaplanmıştır. Kuvvet ölçümleri arasındaki farkın mutlak değeri, her bir sensör ve her bir zaman için hesaplanmış ve bu sinyaller üzerine Kısa Zamanlı Fourier Dönüşümü (KZFD) uygulanmıştır. Böylelikle sinyallerin spektrumundan çeşitli özellikler çıkarılmıştır. Bu özelliklerin histogramı hesaplanmış ve sepet seçimi; özellik ayırma oranı yöntemi kullanılarak yapılmıştır. Daha sonra indirgenmiş histogramlar arasındaki ki-kare mesafesi hesaplanmış ve sınıflandırma için DYM için bir çekirdek oluşturulmuştur. Çalışmada sınıflandırma doğruluğu %91.20 olarak elde edilmiştir.

Nair vd. (Nair et al. 2013) tarafından yapılan çalışmada PH'yi MSA'dan ayıran ayırıcı tanı için 3-T MR görüntüleme tekniği kullanılmıştır. Görüntülerin değerlendirilmesi ve bu verilerin etiketlenmesi iki bağımsız nöroradyolojist tarafından gerçekleştirilmiştir. Son olarak bir karar ağacı modeli oluşturularak PH'nin, MSA'dan ayrılması sağlanmıştır.

Ota vd. (Ota et al. 2013) tarafından yapılan çalışmada PH'nin, MSA-P ve MSA-C'den ayırıcı tanısı üzerine bir çalışma gerçekleştirilmiştir. Çalışmada bireylerden T1 ağırlıklı veriler ve Yayınımsal MRG verileri elde edilmiştir. Daha sonra sırasıyla Voxel-based morphometry (VBM) ve Tract-based spatial yöntemler kullanılarak veriler analiz edilmiştir.

Drotár vd. (Drotár et al. 2014) tarafından yapılan çalışmada PH'li hastalarda ilk görülen özelliklerden olan el yazısının bozulmasının analiz edilmesi ile PH'li hastaların sağlıklı kontrol bireylerinden ayrılabilceği düşünülmüştür. PH'nin tespit edilmesinde el yazısının kinematiğindeki değişim kullanılmıştır. Bu açıdan öncelikle el yazısı özellikleri çıkarılmıştır. Özelliklerin çıkarılmasından sonra veri kümesi oluşturulmuş ve DYM ile sınıfların birbirlerinden ayrılması sağlanmıştır.

Akdemir vd. (Akdemir et al. 2014) tarafından yapılan çalışmada bazı parkinsonizmler'in (PH, MSA, PSP, KBD, Lewy cisimcikli demans (LCD)), kontrol bireylerinden ayırıcı tanısı için Brain 18F-FDG PET görüntüleme yöntemi kullanılmıştır. Görüntülerin analizi görsel olarak ve NeuroQ yazılımı kullanılarak gerçekleştirilmiştir.

Lee vd. (Lee et al. 2014) tarafından yapılan çalışmada PH'nin; PSP ve MSA'dan ayrılması için yürüme ve duruş özellikleri kullanılmıştır. Bunun için PH, PSP ve MSA hastalarının videoları analiz edilmiştir.

Pilleri vd. (Pilleri et al. 2014) tarafından yapılan çalışmada PH ve MSA'nın ayırıcı tanısı için kalp atım hızı ve kan basıncının sirkadiyen ritim özellikleri kullanılmıştır. İki hastalık grubunun karşılaştırılmasında t-testi ve ANCOVA testi kullanılmıştır.

Navarro-Otano vd. (Navarro-Otano et al. 2014) tarafından yapılan çalışmada PH ve Vasküler Parkinsonizm (VP) arasındaki ayırıcı tanı için 123I-meta-iodobenzylguanidine (123I-MIBG), kalp atımı, koku tanıma ve 123I-FP-CIT SPECT analizleri yapılmıştır. Bu analizler sonucunda elde edilen verilerin değerlendirilmesinde istatistiksel testler kullanılmıştır.

Salvatore vd. (Salvatore et al. 2014) tarafından yapılan çalışmada PH ve PSP arasındaki ayırıcı tanı için T1 ağırlıklı MRG verileri kullanılmıştır. Özellik çıkarımında Temel Bileşen Analizi kullanılmıştır. Hastalık gruplarının sınıflandırılmasında ise DYM kullanılmıştır.

Baudrexel vd. (Baudrexel et al. 2014) tarafından yapılan çalışmada PH, MSA ve PSP arasındaki ayırıcı tanı için 18-Fluorodeoxyglucose Pozitron Emisyon Tomografi (FDG-PET) ve Difüzyon Ağırlıklı Görüntüleme (DAG) yöntemleri karşılaştırılmış ve kullanılmıştır. Bu hastalık gruplarının sınıflandırılmasında ANOVA ve ROC analizi kullanılmıştır.

Khorasani ve Daliri (Khorasani & Daliri 2014) tarafından yapılan çalışmada sağlıklı bireyler ile Parkinsonlu bireylerin birbirinden ayrılması için yeni bir yaklaşım önerilmektedir. Bu çalışma da, PH hastalarını sağlıklı bireylerden ayırmak için Gaussian karışımı Gizli Markov Modeli (GMM) kullanılmıştır. Çalışmanın sonuçlarının; 16 sağlıklı ve 15 PH deneğine karşılık gelen yürüme ritim verilerinin sınıflandırılmasında kullanılan GMM sınıflandırıcısının performansının, En Küçük Kareler Destek Yöney Makinesi sınıflandırıcısından elde edilen sonuçlarla karşılaştırılabilir olduğu gösterilmiştir. Bu çalışmada, her bir sınıflandırıcının performansını değerlendirmek için Leave-One-Out-Çapraz-Doğrulama (LOOÇD) yöntemi kullanılmıştır. GMM yöntemi, sağlıklı deneklerden ve PH hastalarından elde edilen uzun adım aralığı bakımından (stride interval), yürüyüş verilerinin doğruluk oranı % 90.3 olarak elde edilmiştir.

Feng vd. (Feng et al. 2015) tarafından yapılan çalışmada PH ve MSA arasındaki ayırıcı tanı için 3.0T MRI verileri kullanılmıştır. Bu hastalık gruplarının karşılaştırılması için istatistiksel testler kullanılmıştır.

Huertas-Fernández vd. (Huertas-Fernández et al. 2015) tarafından yapılan çalışmada PH ve VP arasındaki ayırıcı tanı için [¹²³I]FP-CIT SPECT yöntemi kullanılmıştır. Verilerin sınıflandırılmasında Lojistik Regresyon, Doğrusal Ayırıcı Analiz ve DYM kullanılmıştır.

Zanigni vd. (Zanigni et al. 2015) tarafından yapılan çalışmada PH ve diğer parkinsonyen sendromlar (MSA-C, MSA-P ve Richardson sendromu (PSP-RS)) arasındaki ayırıcı tanı için MR Proton Spektroskopisi (1H-MRS) yöntemi kullanılmıştır. Diğer çalışmalardan farklı olarak bu çalışmada beyincik üzerine odaklanılmıştır. Çalışmada elde edilen verilerin analizinde istatistiksel analiz kullanılmıştır.

Bradvice vd. (Štenc Bradvice et al. 2015) tarafından yapılan çalışmada PH ve Esansiyel Tremor (ET) arasındaki ayırıcı tanı için Transrakniyal Sonografi ve koku

testi kullanılmıştır. Ayrıca Transrakniyal Sonografi sonuçları Dopamin Transporter Scan (DaTSCAN) sonuçları ile de karşılaştırılmış ve yüksek derecede uyumluluk bulunmuştur. Verilerin analizi istatistiksel analiz kullanılarak yapılmıştır.

Vranová vd. (Přikrylová Vranová et al. 2016) tarafından yapılan çalışmada nörodejeneratif hastalıkların (PH, LCD, Alzheimer hastalığı, MSA, PSP) ayırıcı tanısı için Clusterin proteininin beyin omirilik sıvısındaki seviyesi kullanılmıştır. Verilerin karşılaştırılmasında istatistiksel analiz kullanılmıştır.

Drotár vd. (Drotár et al. 2016) tarafından yapılan çalışmada PH'yi normal bireylerden ayıran ayırıcı bir tanı sistemi için el yazısının kinematik ve basınç özellikleri kullanılmıştır. Bu özelliklerden elde edilen verilerin sınıflandırılmasında k-en yakın komşuluk algoritması, AdaBoost ve DYM kullanılmıştır.

Ertuğrul vd. (Ertuğrul et al. 2016) tarafından yapılan çalışmada sağlıklı bireyler ile Parkinsonlu bireylerin birbirinden ayrılması için yeni bir yaklaşım önerilmektedir. Bu çalışmada kullanılan veriler, bu tez çalışmasında da kullanılan Parkinson veri kümesi ile aynıdır. Çalışmada yöntem olarak sinyaldeki yerel değişimleri yakalayabilmek için Kaydırılmış 1-Boyutlu Yerel İkili Örüntüler yöntemi ve Makine Öğrenmesi algoritmaları kullanılmıştır. Çalışmada sınıflandırma doğruluğu en yüksek %88.89 olarak elde edilmiştir.

Jane vd. (Jane et al. 2016) tarafından yapılan çalışmada sağlıklı bireyler ile Parkinsonlu bireylerin birbirinden ayrılması için yeni bir yaklaşım önerilmektedir. Bu çalışmada kullanılan veriler, bu tez çalışmasında da kullanılan Parkinson veri kümesi ile aynıdır. Çalışmada sınıflandırma algoritması olarak Q-geri yayımlı zaman gecikmeli sinir ağı kullanılmıştır. Çalışmada sınıflandırma doğruluğu ortalama %91.53 olarak elde edilmiştir.

Zeng vd. (Zeng et al. 2016) tarafından yapılan çalışmada sağlıklı bireyler ile Parkinsonlu bireylerin birbirinden ayrılması için yeni bir yaklaşım önerilmektedir. Bu çalışmada kullanılan veriler, bu tez çalışmasında da kullanılan Parkinson veri kümesi ile aynıdır. Çalışmada sınıflandırma algoritması olarak RBF Network algoritması kullanılmıştır. Çalışmada sınıflandırma doğruluğu %96.39 olarak elde edilmiştir.

Aydın ve Aslan (Aydın & Aslan 2017a) tarafından yapılan çalışmada yere uygulanan kuvvet sinyalleri kullanılarak nöro-dejeneratif hastalıkların (Amyotrophic Lateral

Sclerosis (ALS), Huntington Hastalığı (HH), PH) teşhisi ve sınıflandırılması gerçekleştirilmiştir. Sınıflandırmada beş Makine Öğrenmesi algoritması (A2DE, K*, Multilayer Perceptron, DECORATE, Random Forest) 10-katlı çapraz doğrulama yöntemi kullanılarak karşılaştırılmıştır. Özniteliklerin oluşturulmasında istatistiksel fonksiyonlar kullanılmıştır. Özellik çıkarımı için ise Temel Bileşen Analizi yöntemi kullanılmıştır. Bu algoritmalarından en iyi sonucu K* sınıflandırıcı vermiştir. Sınıflandırma sonucunda sınıflandırma doğruluğu %99.17 olarak elde edilmiştir.

Shrivastava vd. (Shrivastava et al. 2017) tarafından yapılan çalışmada sağlıklı bireyler ile Parkinsonlu bireylerin birbirinden ayrılması için yeni bir yaklaşım önerilmektedir. Bu çalışmada kullanılan veriler, bu tez çalışmasında da kullanılan Parkinson veri kümesi ile aynıdır. Çalışmada, PH'li bireylerle kontrol grubu bireylerin sınıflandırılmasına yardımcı olmak için ve gerekli en iyi öznitelikleri seçmek için doğadan esinlenen çeşitli algoritmaların karşılaştırmalı bir analizi yapılmıştır. Deneysel sonuçlar seçilen öznitelik kümesi üzerinde İkili Ayıklama Algoritması'nın (Binary Bat Algorithm); Parçacık Sürüsü Optimizasyonu (Particle Swarm Optimization), Genetik Algoritma ve Değiştirilmiş Güldürücü Arama Algoritması (Modified Cuckoo Search Algorithm) gibi geleneksel tekniklerden daha iyi performans sağladığını göstermiştir. Sonuç olarak bir Yapay Sinir Ağı modeli kullanılarak ortalama sınıflandırma doğruluğu %93.12 olarak elde edilmiştir.

Joshi vd. (Joshi et al. 2017) tarafından yapılan çalışmada sağlıklı bireyler ile Parkinsonlu bireylerin birbirinden ayrılması için yeni bir yaklaşım önerilmektedir. Çalışmada, Dalgacık analizi alternatif bir yöntem olarak sunulmuş ve DYM ile kombine edilen dalgacık analizinin etkili bir sınıflandırma doğruluğu üretebildiği gösterilmiştir. Hesaplamalı olarak basitleştirilmiş özellikler dalgacık dönüşümü kullanılarak çıkartılmış ve PH'nin teşhisi için DYM'yi desteklemekte kullanılmıştır. Bu tür sınıflandırma için en iyi tek parametrenin gözlemlenmesi için çeşitli yürüyüş parametrelerini: yani adım aralığı, salınım aralığı ve duruş aralığı (her iki bacadan) değerlendirilmiştir. Çalışmada sınıflandırma doğruluğu %90.32 olarak elde edilmiştir. Ancak sol bacadaki tüm yürüyüş parametreleri daha büyük bir özellik vektörü oluşturmak için bir araya getirildiğinde, sınıflandırma doğruluğu % 100'e yükselmiştir.

Aydın ve Aslan (Aydın & Aslan 2017b) tarafından yapılan çalışmada yere uygulanan kuvvet sinyalleri kullanılarak ALS, HH ve PH gibi nöro-dejeneratif hastalıkların

(NDH) sınıflandırılmasında kullanılabilmesi önerilmektedir. Deneyler 16 kontrol bireyi (KB), 13 ALS, 20 HH ve 15 PH'ye ait veriler kullanılarak gerçekleştirilmiştir. İlk olarak kuvvet sinyalleri, Discrete Meyer dalgacığı kullanılarak yedinci seviyeye kadar ayrıştırılmış ve yeni oluşan sinyallerden yedinci seviyedeki yaklaşım sinyali seçilmiştir. Daha sonra bu sinyal üzerinde tepe analizi gerçekleştirilerek sinyalin lokal maksimumları, tepe'nin x-ekseni değerleri, tepe genişliği ve tepe çıkıntıları elde edilmiştir. Daha sonra bu dört tepe özelliğinin her birinden 15 adet temel istatistiksel öznitelik elde edilmiştir. Böylelikle sol ayak için 60 ve sağ ayak için 60 olmak üzere toplamda 120 öznitelik elde edilmiştir. Daha sonra OneRules sınıflandırıcı kullanılarak bu öznitelikler içerisinde en çok enformasyon veren öznitelikler seçilmiştir. Bir sonraki aşamada ise Radyal Tabanlı Fonksiyon Ağı (RBFNetwork), Adaptif Yükseltme (Adaboost) ve Eklemeli Lojistik Regresyon (LogitBoost) algoritmaları kullanılarak ALS-KB için %93.1 doğruluk, HH-KB için %97.22 doğruluk, PH-KB için %83.87 doğruluk ve NDH-KB için %92.18 sınıflandırma doğruluğu elde edilmiştir.

1.3 Parkinson hastalığı

Sinir sistemi; beyin ve omurilikten yayılan silindirik lif demetleri olan sinir sözcüğünden adını almaktadır ve vücudun her bölümünü sinir sistemine bağlamak için tekrarlı bir biçimde dallanmaktadır (Kandel et al. 2012). Merkezi sinir sistemi ya da çevresel sinir sisteminin gördüğü iç/dış etkilerden ya da genetik faktörlerden dolayı beyinde çeşitli bozukluklar görülebilmektedir. Bu bozukluklardan biri de nörodejeneratif bozukluklardır. Nörodejenerasyon; nöronların ölümü de dahil, nöronların yapı ve işlev kaybını ifade eden genel bir ifadedir. ALS (Amyotrophic Lateral Sclerosis), Parkinson Hastalığı (PH), Alzheimer Hastalığı (AH) ve Huntington Hastalığı (HH) gibi birçok nörodejeneratif hastalık nörodejeneratif süreçlerin bir sonucu olarak ortaya çıkar. Nörodejeneratif hastalıklar tedavi edilemez ve bu ilerleyen bozulma sinir hücrelerinin ölümüne neden olmaktadır (JPND research 2017).

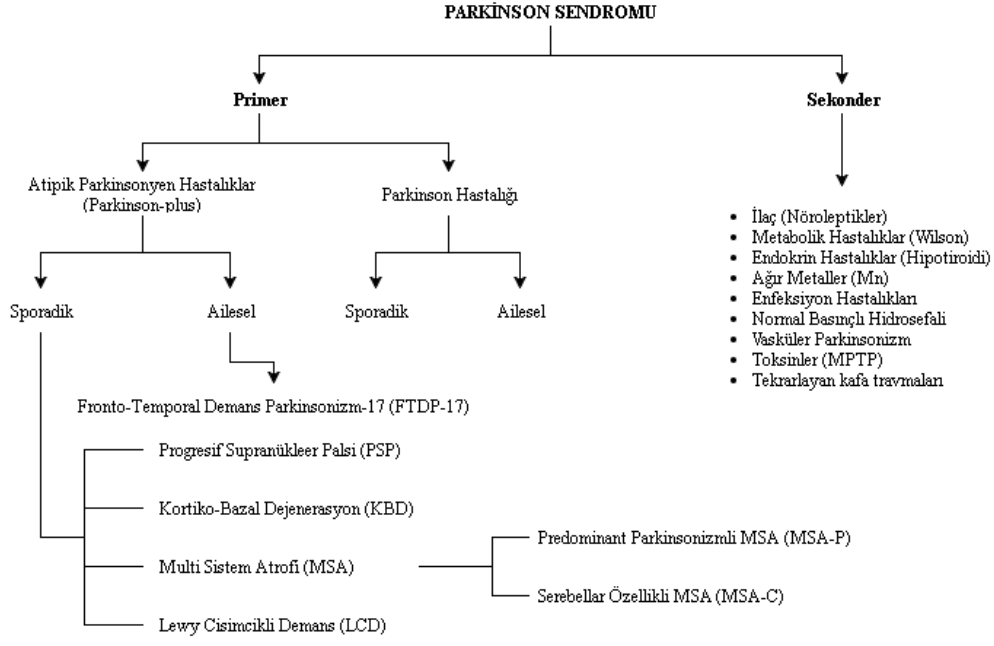
Beynin eferent sistemi motor sistem, motor sistemin son ortak noktası motor nöronlar, onların da yaptığı tek iş kasları aktive etmektir (Demirci 2010). Bazal ganglionların da özellikle amaca yönelik davranışların düzenlenmesinde karmaşık bir işlev gördüğü söylenebilir. Bazal ganglionlar hareketin sadece motor sistemi ile ilgili

değil aynı zamanda hareket ile ilişkili emosyonlar, motivasyon ve kognitif işlevlerde de yer aldığı söylenebilir (Bilgiç 2010). İstemli ya da istemsiz hareketlerin oluşumuna katkı sağlayan tüm bu sistemin tümü ya da bir kısmında meydana gelen bir sorun hareket bozukluklarına yol açmaktadır. Hareket bozuklukları deyimi Nöroloji’de belli bir grup elementer fenomeni ve bunların altında yatan hastalıkları tanımlamak için kullanılır. Bu hastalıkların temel ve ortak özelliği istemli veya otomatik hareketlerin yapılmasında güçlük, istemli hareketlerin istenilen hız, büyüklük ve koordinatlarda yapılamaması veya istem dışı oluşan hareketlerin tek başlarına veya değişik kombinasyonlar şeklinde ortaya çıkmasıdır (Emre & Hanağası 2010). Hareket bozuklukları yetersiz veya aşırı hareketlere yol açan durumlardır. Hareketlerde yavaşlık veya duraksamalara yol açan nörolojik bozukluklara hipokinetik bozukluklar denir ve bu kategorideki hastalıkların başında Parkinson Hastalığı (PH) ve Parkinsonizmin diğer türleri gelir.

PH, 1817 yılında İngiliz hekim James Parkinson tarafından “An Essay on the Shaking Palsy” adlı makalede ilk olarak tanımlanmıştır (Adams et al. 1997). Parkinson hastalığı; yaygın görülen, kronik ve ilerleyen bir nörodejeneratif beyin hastalığıdır. Görülme sıklığı 1000 kişide birdir. 60 yaşın üzerinde ise yaklaşık olarak 100 kişide bir görülmektedir (Adams et al. 1997; Playfer 1997; Maraganore 2000; Manciooco et al. 2009). Erkeklerde görülme sıklığı kadınlara oranla biraz daha fazladır (Waters 2008). Parkinson hastalığının ayırıcı özelliği ve engelleyici semptomu bradikinezidir. Bradikinezi hareketlerin yavaşlaması olarak tanımlanmaktadır. Hastalığıdaki erken dönemde bradikinezinin en önemli belirtisi yürüme, konuşma ve oturmada yaşanan zorluklardır (Gibberd 1986). PH’nin patolojik belirteci olan Lewy cisimciği (LC), ilk olarak 1912 yılında Friedrich Heinrich Lewy tarafından PH hastalarının beyin sapının üst bölgesi olan substansiya nigra (SN)’nin nöromelanin pigmenti içeren sinir hücrelerinin sitoplazmasında bulunan küre şeklindeki kalıntılar olarak tanımlanmıştır (Pollanen et al. 1993). Günümüzde PH’nin belirtilerinin görülmesinden sorumlu olan nigral nöronal kaybın nedeni bilinmemektedir. Ancak kalıtsal nedenler, çevresel toksinler ve yaşlanma etkisi gibi birçok faktörün neden olduğu düşünülmektedir (Piccini et al. 1999; LeWitt 2000). Ayrıca son yıllarda elde edilen genetik ve biyokimyasal veriler ışığında genetik ve/veya çevresel nedenlerle bozulan ubiquitin-proteozom sisteminin PH’nin

gelişimindeki asıl sorumlu mekanizma olduğu düşünülmektedir (Tanner & Goldman 1996).

PH hastalarındaki anormal hareketler; alışılmadık görünümleri, kompleksiteleri, hemen göze çarpmayabilmeleri veya çok değişken olabilmeleri gibi sebeplerle klinisyene tanıma ve kategorize etme zorlukları yaşatır. Hareket bozuklukları uzmanları bunu aşmak için bu anormal hareketleri izole ederek veya bileşenlerine ayırarak çalışma eğilimindedirler. Ancak esas önemli tanısal ipucunu sağlayan bu hareketlerin paterni ve vücutta gösterdiği dağılımdır. Parkinsonizmin kesin tanısı için istirahat tremoru veya bradikinezi olmak üzere bu bulgulardan ikisi mevcut olmalıdır. PH, Parkinsonizm vakalarının en yaygın formudur ve %80'nini oluşturur (Deligtisch et al. 2011). Parkinson Hastalığı'nda; Parkinsonizm bulgularına ilaveten hafif frontal lob bozukluklarından ağır demansa kadar değişebilen bilişsel bozukluklarda eşlik edebilir. PH dışında kalan primer dejeneratif parkinsonizme neden olan hastalıklarda PH'de görülen kardinal semptomlara ilaveten başka semptomların da görülmeleri nedeniyle bunlara Parkinson plus sendromu veya Atipik Parkinson sendromu gibi isimler verilir (Akyürekli & Akyürekli 2010). Toksinler, farmakolojik ajanlar, bazal gangliyonlar'ın fokal veya vasküler lezyonları gibi birçok farklı etken klinik olarak PH'den ayırt edilemeyen sendromlara neden olabilirler ve Sekonder Parkinsonizm olarak isimlendirilirler (Foltynie et al. 2003; Akyürekli & Akyürekli 2010). Şekil 1.1'de Parkinson sendromunun Litvan (Litvan 2005) tarafından yapılan klinik tablosu gösterilmektedir.



Şekil 1.1: Parkinson sendromu klinik tablosu (Litvan 2005)

PH'nin prevalansı 100.000 kişide yaklaşık 160 ve insidansı 100.000 kişide her yıl yaklaşık 20 vakadır. Prevalans ve insidans da yaşla birlikte artar. 70 yaşında prevalans 100.000 kişide yaklaşık 550, insidansıysa yaklaşık 100.000 kişide 120 vakadır. Bulguların ortalama ortaya çıkış yaşı her iki cinsiyet için de 56'dır. Ancak yaş aralığı geniştir ve erken başlangıçlı PH (<40 yaş) vakaları da pek seyrek değildir. Erkeklerde PH, aynı yaştaki kadınların 2 katı kadar sıktır. Ailesinde PH öyküsü olan bireylerde PH ortaya çıkma riski daha yüksektir ve tanımlanmış PH genlerindeki mutasyonların kişilerin soyuna da bağlı olarak vakaların %5-40'ından sorumlu olduğu düşünülmektedir. Kalan vakaların çoğunluğunda ise etyoloji daha karışıktır; gen-çevre ve gen-gen karşılıklı etkileşimi ön planda rol oynar (Deligtisch et al. 2011).

Nörodejeneratif hastalıklarda tedavi süreci, ancak nöron ölümünün nedenlerinin anlaşılmasıyla mümkün olacaktır. Nörodejeneratif hastalıklar konusunda yapılan son çalışmalar göstermektedir ki: Kliniklerde çalışan hekimler ve araştırmacılar nörodejeneratif hastalıklara neden olan etmenlere dayalı bir tedaviye karşı oldukça şüpheli yaklaşmaktadırlar. Son zamanlarda bu kötümser bakış açısı olumlu yönde olacak şekilde değişmektedir. Buna neden olan en önemli nedenler: hücre ve gelişim biyolojisi, model sistemler ve ilaç tedavisindeki gelişmelerdir. Ayrıca nörodejeneratif hastalıklarla ilgili araştırmalara en büyük katkıyı; moleküler genetik alanındaki

ilerlemeler, yüksek teknoloji kapsamına giren genomik arařtırmalar ve hızla gelişen biyolojik yöntemlerle birlikte canlı dokuların ürettiđi elektriksel işaretlerin elde edilmesi ve bu işaretleri işleme tekniklerindeki gelişmeler vermiştir (Beal et al. 2010).

PH ve HH'de bazal ganglionlardaki nöronların kaybıyla hareket kontrolünde anormallikler ortaya çıkmaktadır. AH'de hippokampal ve kortikal nöronların kaybına bađlı kognitif fonksiyonlarda bozulma olur. ALS'de ise spinal, bulbar ve kortikal motor nöronların bozulmaları nedeniyle kas güçsüzlüđü izlenir. ALS, AH, HH ve PH gibi nörodejeneratif hastalıkların bazı belirtileri ortaktır. Bu belirtilerden biri de fiziksel belirtilerdir. Yalnız AH'de bu belirti hastalığın son evrelerine dođru daha çok görülmekle beraber az da olsa hastalığın ilk evrelerinde görülebilmektedir (Selekler 2010). Ancak ALS, HH ve PH hastalarında ortak belirti hastalığın kendisini ilk olarak fiziksel olarak göstermesidir. Fiziksel bozuklukların kendisini gösterdiđi süreçlerden biri de yürüme eylemidir.

1.4 Yürüme

Yürüme, bir noktadan bir noktaya gidebilmek amacıyla gövdenin hareket ettirilmesidir. Yürüme, yaşamın çok basit bir parçası gibi görünmekle birlikte aslında son derece karmaşık bir hareketler zinciridir. Dik bir biçimde ve iki ayak üzerinde yürüme özelliđi memeliler arasında sadece insana özgüdür (Özaras & Yalçın 2002a). Ayrıca iki ayak üzerinde yürüme özelliđi memeliler arasında sadece kanguru ve jerboas'a özgüdür (Barnosky 2010).

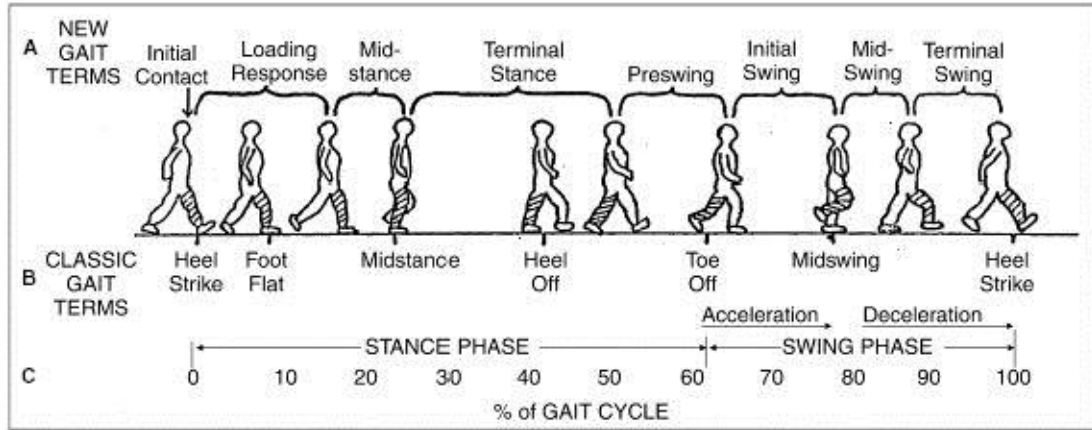
Son yıllarda hareket problemlerinin tanı ve tedavisinde önemli bir noktaya gelinmiştir. Günümüz yürüme analizi teknolojisi normal ve patolojik yürüyüş esnasında eklem ve kasların işlevlerini sayısal veriye dönüřtürerek analiz edilebilmesini sağlar. Böylelikle hasta üzerine uygulanan fizyoterapi, ilaç ve cerrahi gibi tedavilerin yürüme üzerine olan etkileri nesnel olarak deđerlendirilebilir. Yürüme analizi ile insan-uzmanın çözümleyemeyeceđi yürüme ve hareket sorunları kolay bir biçimde analiz edilebilmekte ve soruna neden olan kas ve ekleme yönelik müdahale yapılabilir (Özaras & Yalçın 2002a). Yürüme, sürekli kendini tekrar eden hareketlerden oluşur. Bu hareketler topluluđu, bir yürüme döngüsü olarak tanımlanır (Barr 2002). Aslında yürüme, beyinde başlar. Bu nedenle yürüme

sorunlarında değerlendirme; beyinden medulla spinalise, oradan da kas ve eklemlere doğru olmalıdır (Kanatlı et al. 2006).

Yürüme, stabil bir ağırlık taşıyıcı postürü devam ettirerek vücudu istenilen bir doğrultuda simültane olarak ilerleten alt ekstremitenin yaptığı bir hareket sekansının peşpeşe yapılmasına dayanır. Efektivite ise eklem mobilitesinin serbestliğine ve uygun yoğunlukta selektif olan kas hareketine dayanır. Normal fonksiyon aynı zamanda fizyolojik enerjiyi optimal şekilde korumaktadır. Patolojik durumlar yürümenin modunu ve etkinliğini değiştirir. Bazı hareketler kaybedildiğinde eğer ileriye gidiş ve duruş stabilitesi korunmak isteniyorsa kaybedilen hareketlerin yerine başkalarının konması gerekir. Normal fonksiyon ve çeşitli patolojik durumlardan kaynaklanan yürüme bozukluklarının tipleri ile ilgili detaylı bilgi sahibi olan bir klinisyen önemli kayıpları tespit edebilecek ve uygun düzeltici ölçümleri planlayabilecektir (Inman et al. 1989; Cochran 1982; Vaughan et al. 1999).

1.4.1 Yürümenin fazları

Alt ekstremitte hareketinin her sekansı (yürüme siklusu) bir ağırlık taşıyıcı periyot (stance/duruş) ve bir ilerleme intervali (swing/sallanma/ilerleme) içerir. Normal yürüme siklusu süresinin %60'ı duruşta, %40'ı sallanmada harcanır. Bu intervallerde harcanan süre yürüme hızına bağlı olarak değişir. Ayrıca bireyler arası farklılıklar da vardır. Her iki ekstremitenin resiprokal (değişmeli) hareketi bir çift duruş periyodunda (mesela her iki ayak yere değerken) ağırlık taşıma görevlerini paylaşmak üzere zamanlama yapmaktadır ve genellikle duruşun başlangıç ve bitiş %10'luk intervalini kapsar. Orta %40'luk periyot tek duruş periyodudur (tek bacak desteği) (Zengin 2007). Bu sırada diğer bacak sallanma yapmaktadır. Tüm bu yürüme fazları Şekil 1.2'de gösterilmektedir. Buna göre Basma fazı; başlangıç teması, yüklenme cevabı, orta duruş, son duruş ve salınım öncesi fazıdır. Salınım fazı ise; başlangıç salınımı, salınım ortası ve salınım sonu fazı olarak alt gruplara ayrılır (Sutherland & Valencia 1992; Özaras & Yalçın 2002b; Chambers & Sutherland 2002).



Şekil 1.2: Yürümenin fazları (Castermans et al. 2013)

1.4.1.1 Topuk vuruşu (initial contact/heel strike)

Yürüme siklusunun %0-2'si bu peryotta gerçekleşir. Basma fazının başlangıcı olup ayağın yere teması ile başlar. Bu temas ayağın önce topuk yere temas edecek şekilde yere indirilmesidir (Sutherland & Valencia 1992; Özaras & Yalçın 2002b; Chambers & Sutherland 2002).

1.4.1.2 Yüklenme/taban vuruşu (loading response/footflat)

Yürüme siklusunun %2-10'u bu peryotta gerçekleşir. Aynı zamanda birinci çift destek fazıdır. Diğer ayak yerden kaldırılana kadar gövde ağırlığı yerdeki ayağa aktarılır (Sutherland & Valencia 1992; Özaras & Yalçın 2002b; Chambers & Sutherland 2002).

1.4.1.3 Basma orta fazı (midstance)

Yürüme siklusunun %10-30'u bu peryotta gerçekleşir. Tek basma fazının başlangıcıdır. Salınım fazındaki bacak, yere basan bacağın yanından geçer. Bunu yapmaktadaki amaç yerde sabit duran ayağı kullanarak gövdeyi öne doğru ilerletmektir (Sutherland & Valencia 1992; Özaras & Yalçın 2002b; Chambers & Sutherland 2002).

1.4.1.4 Basma sonu fazı/topuk kalkışı (terminal stance/heel off)

Yürüme siklusunun %30-50'u bu peryotta gerçekleşir. Amaç bacağın yerden kesilmesidir (Sutherland & Valencia 1992; Özaras & Yalçın 2002b; Chambers & Sutherland 2002).

1.4.1.5 Salınım öncesi/parmak kalkışı (preswing/toe off)

Yürüme siklusunun %50-60'u bu peryotta gerçekleşir. Basma fazının sonlanıp salınım fazının başladığı evredir. Ayrıca ikinci çift destek evresini oluşturur. Karşı ayak yere değdiğinde başlar ve parmakların yerden kesilmesiyle sonlanır. Amaç bacağı salınımına hazırlamaktır (Sutherland & Valencia 1992; Özaras & Yalçın 2002b; Chambers & Sutherland 2002).

1.4.1.6 Erken salınım/akselerasyon (initial swing)

Yürüme siklusunun %60-73'ü bu peryotta gerçekleşir. Ayağın yerden kaldırılmasıyla başlar, ayak diğer ayağın hizasına geldiğinde biter. Amaç havadaki bacağı hızla öne doğru ilerletmektir (Sutherland & Valencia 1992; Özaras & Yalçın 2002b; Chambers & Sutherland 2002).

1.4.1.7 Salınım ortası (midwing)

Yürüme siklusunun %73-87'si bu peryotta gerçekleşir. Salınan bacak yere basan bacağın yanına gelir ve önüne geçer. Amaç ayağın yere değmeden aktarılmasıdır (Sutherland & Valencia 1992; Özaras & Yalçın 2002b; Chambers & Sutherland 2002).

1.4.1.8 Salınım sonu/deselerasyon (terminal swing)

Yürüme siklusunun %87-100'ü bu peryotta gerçekleşir. Salınan bacak basan bacağın önüne geçtiğinde başlar, ayağın yere değdiği ana kadar devam eder. Amaç ayağın yere basmaya hazırlanmasıdır (Sutherland & Valencia 1992; Özaras & Yalçın 2002b; Chambers & Sutherland 2002).

1.4.2 Vücutun ilerletilmesi

Vücutun ilerletilmesinde basma fazındaki ayağın bazı hareketleri olmaktadır. Bunlar: Topukta dönme (heel rocker), Ayak bileğinde dönme (ankle rocker) ve Önayakta dönme (forefoot rocker)'dir. Topukta dönme evresinde; ayak yere değdiğinde vücut topuk üzerinden öne doğru dönmeye başlar. Bu dönemde topuk bir kaldırıcının dayanak noktası gibi çalışır. Ayağın tamamı yere değdiğinde topukta dönme biter, ayak bileğinden dönme başlar. Ayak bileğinde dönme evresinde ayak yerde sabit durur. Yer Tepkime Kuvvet Vektörü (YTKV) ayak bileği eklemine önüne geçer ve pasif dorsifleksiyon oluşur. Ayak yerde sabit olduğu için tibia öne doğru gelir. Bu aşamada soleus kası eksantrik olarak kasılır ve tibianın öne gidişini kontrol eder. YTKV ön ayağa ulaştığında topuk yerden kalkar, ayak bileğindeki

dönme biter ve önyakta dönme başlar. Vücut ağırlık merkezi (VAM) destek alanı merkezinin önüne düşmeye başladığında ilerleme hızlanır. Triceps kasılarak ayak bileğinde plantar fleksiyon yaratır ve güçlü bir itici kuvvet sağlar. Burada artık gövde kütlesi uzun bir kaldıraç kolunun ucundaki pasif ağırlık gibidir ve diğer ayak yere basana kadar düşmesini önleyecek hiç bir kuvvet yoktur (Özaras & Yalçın 2002b).

Yer Tepkimesi Kuvveti, Newton'un üçüncü hareket ilkesine göre ayakta duran bir kişinin yerde oluşturduğu ağırlık kuvvet vektörüne karşılık, yer de eşit büyüklükte ve ters yönde bir YTKV oluşur. YTKV'nin yönü ve büyüklüğü yürüme esnasında sürekli bir biçimde değişir (Özaras & Yalçın 2002b).

1.4.3 Yürüme analizi

Yürüme eyleminin son derece karmaşık hareketler zinciri olması ve insan gözünün değerlendirip çözümleyememesinden dolayı yürüme eyleminin iyi bir biçimde analiz edilmesi gerekmektedir. Yürüme analizi; gözlemcinin göz ve beynini kullanarak ve aynı zamanda vücut hareketlerini, vücut mekaniğini ve kas aktivitesini ölçmek için çeşitli araçları kullanarak genelde hayvan hareketlerini özelde ise insan hareketlerini inceleyen sistematik bir çalışma alanıdır (Levine et al. 2012). Yürüme analizinin ilk değerlendirilmesi Aristo (384 B.C.-322 B.C.) zamanına uzanır (Aristotle 2004). Daha sonra 1890'da Alman anatomist Christian Wilhelm Braune ve Otto Fischer, yük altında ve yük olmadan insan yürümesinin biyomekaniği isimli bir dizi makale yayımladılar (Fischer & Braune 1895). Yalnız yürüme analizi açısından en önemli adım yürümenin sayısal olarak ölçülmesidir.

Yürüme analizinin ilk ölçümü Tibarewala ve Ganguli tarafından 1979 yılında yapılmıştır. Bu çalışmada, sağlıklı yetişkin erkeklerden alınan birçok normal yürüme eğrisi tanımlanmış ve bunlar kullanılarak insan performansının sayısal ölçümü olarak yürüme anormal indeksi geliştirilmiştir (Tibarewala & Ganguli 1979). Birkaç yıl sonra sağlıklı bireylerde serbest hız seviyesinde yürünürken soleus kaslarına (baldırın arkasında bulunan kaslar) göre Electromyography (EMG) profiline karar vermek için bir bilgisayar algoritması geliştirildi (Bogey et al. 1993). Bir sonraki çalışmada Kerrigan ve diğerleri, yürüme esnasında sacrum'un (kuyruk sokumu kemiğinin) dikey yer değişimini ve gövde örüntülerini iki belirteç olarak önerdiler ve yürümenin tüm biyomekaniksel performansının bir kestirimi olarak sacrum'u önerdiler (Kerrigan et al. 1996). Ayrıca adım uzunluğu, yürürken gövdenin dikey hareketi ve ayaktaiken sacral yüksekliği kullanarak Biyomekaniksel Etkinlik Oranını (BEQ) hesapladılar

(Kerrigan ve diğ., 1996). Ancak bu çalışmalardan sonra, 3D-GA (3D Gait Analysis) için özet ölçümlerle ilgili birçok çalışma yapıldı ve patolojik konularda onların uygulamaları normallik indeksiyle (NI) 2000'li yılların başında başladı.

Schutte'nin NI çalışması, yaygın bir klinik kabul ile global anlamda bir hastanın yürümesini karakterize etmeye izin veren ilk göstergedir. NI'nın diğer bir adı ise Gillette Yürüme Göstergesi (GGI)'dir (Schutte et al. 2000). Bu çalışma da, bir hastanın yürüme şeklinin hasta olmayan kontrol gruplarından ne kadar saptığının derecesini ölçmek için çok değişkenli istatistiksel metotlar kullanılmaktadır. NI, 16 adet 3D GA değişkene yani 3 adet zaman uzamsal parametreye (duruş fazı yüzdesi, normalize edilmiş hız ve ritim) ve 13 adet kinematik parametreye (ortalama pelvik tilt, pelvik tilt aralığı, ortalama pelvik dönüşü, minimum kalça esnemesi, kalça esneme aralığı, salınımdaki zirve abdüksiyonu, dururken ortalama kalça dönüşü, başlangıç temasında diz esnemesi, diz esnemesinin zirve zamanı, diz esnemesinin aralığı, dururken arkaya esneme zirvesi, salınım esnasında arkaya esneme zirvesi, ortalama ayak ilerleme açısı) uygulanarak standart çok değişkenli istatistiksel tekniklerle (örneğin Temel Bileşen Analizi) hesaplanmıştır. Bu bağımsız 16 değişkenin kareleri toplamı hesaplanarak; kişinin yürümesinin, normalden ne kadar saptığı yorumlanabilmektedir. Bu istatistiksel metot kullanılarak normal ortalama bir profilden patolojik bir yürüme örüntüsünün sapmasının temsili ve ölçümü mümkündür. Böylece NI; bir kişinin, ortalama normal bir profile göre yürüme sapmasının miktarını gösterir (Cimolin & Galli 2014). NI birçok araştırmada yaygın olarak kullanıldı (Schutte et al. 2000; Barton et al. 2007; Hillman et al. 2007; Trost et al. 2008; Thomason et al. 2011). Ancak bazı çalışmalarda da olumsuz sonuçlar ortaya çıkmıştır. Brehm tarafından yapılan çalışmada NI göstergesinin hedeflenen müdahalelerin etkilerini değerlendirirken kullanılan uygun özgüllük ve duyarlılık değerlerini vermediği gösterilmiştir (Brehm et al. 2008).

Hip Flexor Index (HFI) 5 kinematik ve kinetik değişkene (maksimum pelvik tilt, pelvik tilt aralığı, dururken maksimum kalça ekstansiyonu, ekstansiyon durumundan esneme durumuna geçişte kalça esneme anının son değişimi meydana geldiğinde durma fazının yüzdesi ve zirve geç duruş kalça esneme gücü) uygulanan Temel Bileşen Analizi kullanılarak geliştirildi. Yürürken tüm kalça fonksiyonlarını doğru bir biçimde tanımlayan tek bir gösterge sayısı türetildi (Schwartz & Rozumalski 2008). En önemlisi, bu gösterge tek bir anatomik düzeyde onun yalınlığından dolayı

GGI'nın aynı kategorisine ait değildir (Cimolin & Galli 2014). HFI kalça fonksiyonundaki bir değişimin klinik etkisini somutlaştırmak için kullanılan geçerli bir araçtır. Böylelikle HFI, birbirine bağımlı ve gereksiz yürüme değişkenlerinin istatistiksel ve çıkış analizleri ile ilgilenen araştırmacılara yardımcı olabilmektedir. Ancak HFI'nın dezavantajı sadece kalçadaki bir müdahaleyi değerlendirmek için uygun olmasıdır (Novacheck et al. 2002). Bu nedenle diğer uygulamalar bu parametrelerin doğasından dolayı uygun değildir (Cimolin & Galli 2014).

Gait Deviation Index (GDI), son zamanlarda geliştirilen ve tüm yürüme patolojilerini temsil ettiği düşünülen bir ölçüm tekniğidir. GDI; üç düzlem içinde alt karın ve kalçanın, sagittal düzlemde diz ve ayak bileğinin ve son olarak ayak ilerlemesinin 3D-GA kinematiklerinden ayrıştırılan tek bir değer kullanılarak 15 yürüme niteliğinin çıkartılmasına dayanır (Schutte et al. 2000). NI ve GDI'nin karşılaştırılmasında GDI'nın birkaç avantajının olduğu görülmektedir. Yürüme döngüsü boyunca kinematik değişkenlerdeki tüm değişkenlik az sayıda ayrı parametrenin yerine kullanılır. Böylece parametre seçiminde özneliliğin çoğu ortadan kaldırılır. NI için parametre seçimi Cerebral Palsy (CP)'li çocuklar için spesifiktir (McMulkin & MacWilliams 2008).

Gait Profile Score (GPS) ve Movement Analysis Profile (MAP) GDI'ya benzer yalnız GPS, hastanın kinematiklerinin tüm niteliğini özetleyen tek bir gösterge çıkış ölçümüdür. GDI'ya göre daha basit mesafe ölçümü yorumuna imkan tanır. Ayrıca 9 adet ilişkili kinematik değişken için tek bir Yürüme Değişim Sapması (GVS) sağlamak için analiz edilebilir (Baker et al. 2009).

GDI kinetik, bu ölçümde daha önceki ölçümlerden farklı olarak kinetik nitelikler de kullanılmıştır. Önceki ölçüm tekniklerinde uzamsal-zamansal parametreler ve kinematik niteliklerin sadece kullanılması yürüme örüntüsünün değerlendirilmesinde yeterli gelememektedir. Çünkü bu nitelikler hastanın yürüme örüntüsünün sınırlı bir değerlendirmesini sağlamaktadırlar. Kinetik parametrelerin entegrasyonu ile birlikte ortak reaksiyonların, momentlerin ve gücün daha iyi bir şekilde araştırılması önem kazanmaktadır. Böylece kinematik hareketin nasıl olduğu değil aynı zamanda kinetikle beraber neden olduğu daha kapsamlı bir biçimde anlaşılacaktır. İşte bu sınırlamaların üstesinden gelmek için GDI kinetik geliştirildi (Rozumalski & Schwartz 2011).

Sonuç olarak Cimolin ve Galli'ye göre klinik uygulamalar için özet ölçümlerde 3D-GA raporunda temsil edilen bilginin tümü (zaman uzamsal parametreler, kinematik, kinetik ve EMG verileri) birleştirilerek kullanılmalı (Cimolin & Galli 2014). Bu çalışmada kullanılan parametreler: kinetik, zaman uzamsal ve denek klinik bilgileridir. Kinetik, hareketi ve hareketten doğan kuvvetlerin (yer tepkimesi kuvvetleri, eklem momentleri, eklem güçleri) incelenmesidir. Yürüme eyleminin kinetik analizi esnasında ölçülebilen tek parametre YTKV'dir. YTKV, ayağın yere uyguladığı toplam kuvveti ölçen ayrıca basınca duyarlı olan, kuvvet platformu isimli plakalarla ölçülür. Platforma yerleştirilmiş olan transdüserler yardımıyla kuvvet platformu üzerindeki yükün her üç eksen üzerindeki bileşenleri ölçülür ve bu ölçülen veriler bilgisayar sistemine aktarılır. Böylelikle bir kişi kuvvet platformuna basarak ilerlediğinde basma fazında oluşan YTKV'ler belirli aralıklarla hesaplanır. Kuvvet platformları kinematik analiz yapan sistemlerle birlikte kullanıldığı takdirde; ayak bileğine, dize ve kalçaya etki eden momentler ve bu bölgelerde meydana gelen güçler hesaplanabilir. Bu kuvvetleri hesaplamak için link segment modeli ve ters dinamik (inverse dynamics) isimli analiz yöntemi kullanılır. Link segment modelinde vücudun; uyluk, baldır, ayak gibi bölgelerinin bağımsız kitlesi ve hareketi olduğu varsayılır. Daha sonra ters dinamik yöntemi için kuvvet platformundan elde edilen YTKV ve momentler, bacakların tüm kısımlarının; pozisyon, hız ve hızlanma verileri (kinematik veriler) ve kişinin antropometrik verileri bilgisayar sistemi aracılığıyla bir araya getirilir. Böylelikle kalça, diz ve ayak bileğine etki eden kuvvetler (dış momentler), dolaylı yollardan elde edilen iç momentler ve diğer güçler hesaplanabilir (Sutherland 1990).

1.5 Tez İçeriği

Bu tez çalışması dört ana bölümden oluşmaktadır. Bu bölümler sırasıyla şunlardır: Giriş, Veri ve Yöntem, Analiz ve Uygulama ve son olarak Sonuçlar ve Tartışma'dır. Bu bölümlerin içeriği hakkında aşağıda kısaca bilgiler verilmektedir.

2. Bölüm'de bu tez çalışmasında uygulanan yöntemleri ve kullanılan verileri tanıttık. Tez çalışmasında kullanılan verileri ikiye ayırdık: Benchmark veri kümeleri ve Parkinson veri kümesi. Daha sonra Parkinson veri kümesi için öznitelik oluşturma ve öznitelik seçim yöntemlerini tanıttık. Bu aşamadan sonra veri kümeleri üzerine uygulanan ve tarafımızdan önerilen *VIBES* isimli Topluluk Öğrenme algoritmasını

tanıttık. Ayrıca *VIBES* algoritmasının zaman ve bellek karmaşıklığını hesapladık. *VIBES* algoritmasında kullanılan çoğunluk oylama yönteminin diğer yöntemlere kıyasla ikili sınıflandırma problemlerinde daha etkili olduğunu teorik olarak gösterdik. Bunun yanısıra Çoğunluk Oylamalı Topluluk Öğrencilerde sınıflandırma hatasının tahmini oranını hesaplayan ve bu tez çalışmasında geliştirilen özgün bir yöntemi tanıttık. Ayrıca geliştirdiğimiz Topluluk Öğrenme algoritmasını NFLT açısından teorik olarak değerlendirdik. Bu tez çalışmasında *VIBES* algoritmasının temel öğrenci olarak kullandığı algoritmaları da tanıttık. Son olarak Makine Öğrenmesi algoritmalarının performanslarını analiz etmek için kullanılan istatistiksel ölçütleri ve yöntemleri tanıttık.

3. Bölümde *VIBES* ve diğer Makine Öğrenmesi algoritmalarının veri kümeleri üzerine uygulanmadan önce veri kümeleri üzerinde gerçekleştirilen işlemleri anlattık. Özellikle Parkinson veri kümesi üzerine uygulanan sinyal dönüşüm yöntemlerinin uygulanma sürecini ve bu süreç sonucunda oluşturulan öznitelikleri tanıttık.

4. Bölümde *VIBES* ve diğer Makine Öğrenmesi algoritmalarının benchmark ve Parkinson veri kümesi üzerindeki sonuçlarını karşılaştırarak tartıştık. Ayrıca bu tez çalışması sonucunda elde edilen sonuçlar ışığında çeşitli değerlendirmeler yaptık.



2. VERİ VE YÖNTEM

2.1 Deneylerde Kullanılan Veri Kümeleri

VIBES algoritmasının başarımını test etmek ve diğer Makine Öğrenmesi algoritmaları ile yarışabilme kapasitesini deneysel olarak ölçmek için UCI ve PhysioNET veritabanlarından çeşitli veri kümeleri kullanıldı. Bu veri kümeleri ile ilgili ayrıntılı bilgi aşağıda yer almaktadır.

2.1.1 Benchmark veri tabanı

VIBES algoritmasının başarımını test etmek ve diğer Makine Öğrenmesi algoritmaları ile yarışabilme kapasitesini deneysel olarak ölçmek için UCI (Lichman 2013) ve mlbench (Leisch 2012) veritabanlarından 33 veri kümesini kullandık. Bu veri kümelerinden 30 tanesi birbirinden farklı veri kümeleridir. Bu veri kümeleri ile ilgili tanımlayıcı bilgiler Çizelge 2.1’de gösterilmektedir.

Çizelge 2.1: Deneyleerde Kullanılan Veri Kümelerinin Özellikleri

Veri kümesi	Veri Sayısı	Öznitelik sayısı	Sınıf sayısı	Dengesizlik oranı	Kaynak
Banknote Authentication	1372	5	2	1.27	(Gillich & Lohweg 2010)
Blood Transfusion	748	5	2	3.20	(Yeh et al. 2009)
Boston Housing2	506	19	92	30.00	(Harrison & Rubinfeld 1978)
Breast Cancer	699	10	2	1.90	(Mangasarian & Wolberg 1990; Wolberg & Mangasarian 1990; Mangasarian et al. 1990; Bennett & Mangasarian 1992)
Breast Tissue	106	10	6	1.57	(Jossinet 1996; Estrela da Silva et al. 2000)
Cardiotocography-1	2126	22	3	9.40	(Ayres-de-Campos et al. 2000)
Cardiotocography-2	2126	22	10	10.92	(Ayres-de-Campos et al. 2000)
Climate Model	540	19	2	10.73	(Lucas et al. 2013)
DNA	3186	181	3	2.16	(Noordewier et al. 1990)
Glass	214	10	6	8.44	(Lichman 2013)
HTRU2	17898	9	2	9.92	(Lyon et al. 2016)
Ionosphere	351	35	2	1.78	(Sigillito et al. 1989)
Leaf	340	15	30	2.00	(Silva et al. 2013)
Letter Recognition	20000	17	26	1.10	(Frey & Slate 1991)
Libras Movement	360	91	15	1.00	(Dias et al. 2009)
LSVT Voice Rehabilitation	126	311	2	2.00	(Tsanas et al. 2014)
Madelon	2000	501	2	1.00	(Guyon et al. 2004)

Çizelge 2.1: (devam) Deneyleerde Kullanılan Veri Kümelerinin Özellikleri

MAGIC Gamma Telescope	19020	11	2	1.84	(Bock et al. 2004)
Parkinson Speech	1040	27	2	1.00	(Sakar et al. 2013)
QSAR biodegradation	1055	42	2	1.96	(Mansouri et al. 2013)
Satellite	6435	37	6	2.44	(Leisch 2012)
Seeds	210	8	3	1.00	(Charytanowicz et al. 2010)
Shuttle	58000	10	7	4558.60	(Leisch 2012)
Sonar	208	61	2	1.14	(Leisch 2012)
Vehicle	846	19	4	1.10	(Leisch 2012)
Vertebral Column	310	7	2	2.10	(Neto & Barreto 2009; da Rocha Neto et al. 2011)
Vowel	990	11	11	1.00	(Leisch 2012)
Wall Following Robot Navigation-1	5456	3	4	6.72	(Freire et al. 2009)
Wall Following Robot Navigation-2	5456	5	4	6.72	(Freire et al. 2009)
Wall Following Robot Navigation-3	5456	25	4	6.72	(Freire et al. 2009)
Wine Quality-red	1599	12	6	68.10	(Cortez et al. 2009)
Wine Quality-white	4898	12	7	439.60	(Cortez et al. 2009)
Zoo	101	17	7	10.25	(Lichman 2013)

2.1.2 Parkinson veri kümesi

VIBES algoritması, bu 33 veri kümesi dışında PhysioNET (Goldberger et al. 2000) veritabanından elde edilen bir veri kümesi üzerine de uygulandı (Frenkel-Toledo, Giladi, Peretz, Herman, Gruendlinger & Jeffrey M Hausdorff 2005; Yogeve et al. 2005; Frenkel-Toledo, Giladi, Peretz, Herman, Gruendlinger & Jeffrey M. Hausdorff 2005; Hausdorff et al. 2007). Bu veri kümesi, idiyopatik Parkinsonlu 93 hasta ve 72 sağlıklı kontrol bireyinden elde edilen yürüme ölçümlerini içermektedir. Ayrıca bu veri kümesi, yer seviyesinde yaklaşık olarak 2 dakika boyunca deneklerin yürürken yere uyguladıkları yer tepki kuvveti kayıtlarını da içerir. Deneklerin herbirinin ayaklarının altında, kuvveti (Newton cinsinden) zamanın bir fonksiyonu olarak ölçen 8 sensör (Ultraflex Computer Dyno Graphy, Infotronic Inc.) bulunmaktadır. Bu 16 algılayıcının her birinin çıktısı sayısallaştırılarak saniyede 100 örnek kaydedilmiştir. Ayrıca kayıtlar, her bir ayak için 8 algılayıcı çıkışının toplamını yansıtan iki sinyal içermektedir. Bu veri kümesi ile ilgili tanımlayıcı bilgiler Çizelge 2.2’de gösterilmektedir. Ayrıca deneklere ait ayrıntılı tanımlayıcı bilgiler Çizelge A.1’de gösterilmektedir. Bu veri kümesi üzerinde yüksek sınıflandırma doğruluğu elde etmek için *VIBES* algoritması kullanıldı ve *VIBES* algoritması diğer Makine Öğrenmesi algoritmaları ile karşılaştırıldı.

Çizelge 2.2: Parkinson Veri Kümesi İçin Tanımlayıcı Bilgiler (Goldberger et al. 2000)

Denekler	Veri Sayısı	Yaş	Ağırlık	Boy	Cinsiyet	
					E	K
Parkinsonlu	93	66.30 ± 9.50	72.40 ± 11.96	167.44 ± 8.59	58	35
Kontrol Bireyi	72	63.68 ± 8.70	72.69 ± 12.42	168.25 ± 8.58	40	32

2.2 Öznitelik Oluşturma ve Öznitelik Seçimi

Yapılandırılmamış ya da işlenmemiş veriye istatistiksel analiz uygulayarak öznitelik oluşturma süreci Makine Öğrenmesi alanı için önemli bir aşamadır. Aynı zamanda öznitelik oluşturma süreci Makine Öğrenmesi alanının zorlu yönlerinden biridir. Öznitelik oluşturma süreci dışında öznitelik seçimi de Makine Öğrenmesi alanı için

önemli bir alt başlıktır. Özellik seçimi, model yapımında kullanılacak ilgili özniteliklerin bir alt kümesini seçme işlemidir. Öznitelik seçme teknikleri üç nedenden dolayı kullanılır. Bunlar: Modellerin daha kolay yorumlanabilmesi için basitleştirilmesi (James et al. 2013), sınıflandırıcıların eğitim zamanını düşürmek ve varyanstan kaynaklanabilecek hatalardan kaçınmaktır (Bermingham et al. 2015).

Bu tez çalışmasında Parkinson veri kümesi üzerinde öznitelik oluşturma ve oluşturulan bu özniteliklerin seçilmesi işlemleri gerçekleştirildi. Bu süreçler aşağıda ayrıntılı bir şekilde anlatılmaktadır.

2.2.1 Sinyal dönüşümleri ve öznitelik oluşturma

Ham verinin Makine Öğrenmesi alanında etkili bir biçimde kullanılabilmesi için dönüştürülmesi gerekmektedir. Bunun için sinyal dönüşüm yöntemleri kullanılmaktadır. Özellikle zaman serilerinde sinyal dönüşüm yöntemleri etkili olmaktadır. Bu tez çalışmasında Parkinson veri kümesi üzerinde çeşitli sinyal dönüşüm yöntemleri uygulandı. Bu yöntemlerden aşağıda bahsedilmektedir. Bu yöntemler kullanılarak Parkinson veri kümesi için istatistiksel öznitelikler oluşturuldu.

2.2.1.1 Fourier dönüşümü

Doğada basit periyodik olaylar matematiksel olarak sinüs ve cosinüs fonksiyonları ile ifade edilirler. Örneğin bir sarkacın küçük genliklerle salınımı, bir diyapozonun titreşimleri ve buna benzer fiziksel olayların herbiri birer basit periyodik olaydır. Eğer olay saniyede n defa tekrarlanıyorsa basit titreşimleri gösteren fonksiyon $A \sin 2\pi nt$ ya da $A \cos 2\pi nt$ den biridir. Burada t saniye ile ölçülen zamanı, A genliği ve $w = 2\pi nt$ ifadesi de titreşim hareketinin açısal hız veya pülsasyonu olarak bilinmektedir. Basit sinüs ve cosinüs fonksiyonları ile belirtilen olaylara basit harmonik hareket adı verilir. Periyodik olayları ifade etmekte kullanılan periyodik fonksiyonlar trigonometrik serilerle ifade edilmektedir (Altın 2011). Periyodik fonksiyonlar reel değişkenli ve kompleks değişkenli fonksiyonlar olarak iki başlık altında incelenebilir.

Reel değişkenli periyodik fonksiyonlar; $D \subseteq R$ olmak üzere, $f: D \rightarrow R$ fonksiyonunda, $\forall x \in D$ için $x + T \in D$ ve $f(x + T) = f(x)$ şartlarını sağlayan sıfırdan farklı bir T reel sayısı varsa $f(x)$ fonksiyonuna periyodik fonksiyon denir. T sayısına da $f(x)$ fonksiyonunun bir periyodu denir (Aydın 1999).

$\forall x \in D$ için $f(x + \ell) = f(x)$ şartını sağlayan $\ell \neq 0$ sayılarından en küçük ve pozitif olanına $f(x)$ fonksiyonunun esas periyodu denir (Nasibov & Kaçar 2008).

Periyodik fonksiyonlar genellikle trigonometrik fonksiyonları çağrıştırır. Yalnız periyodik fonksiyonlar sadece trigonometrik fonksiyonlardan oluşmaz. Örneğin $f: [0,1] \rightarrow \mathbb{R}$ de tanımlı $f(x) = x^2$ fonksiyonu $\ell = 1$ periyodu ile periyodik fonksiyondur.

Fransız bilimadamı Joseph Fourier (1768-1830), *The Analytic Theory of Heat* (1822) isimli çalışmasında yakınsak ve integrallenebilir her fonksiyonun trigonometrik bir seri ile gösterilebileceğini belirtmiştir. Fourier serileri, fizik ve mühendislikte pek çok uygulama alanı bulunan önemli serilerden biridir. Fizikte elektromanyetik teori, kuantum teorisi, ısı teorisi, akustik, elektronik gibi çeşitli alanlardaki uygulamalarda, adi ve kısmi diferansiyel denklemlerin çözümlerinde (Brown & Churchill 1993) çoğu kez Fourier serileriyle karşılaşılır.

1, $\cos x$, $\sin x$, $\cos 2x$, $\sin 2x$, ... fonksiyonları $[-\pi, \pi]$ aralığında ortogonal bir sistem oluştururlar. 2π ortak periyoduna sahip olan bu fonksiyonlar yardımı ile oluşturulan trigonometrik serisi yakınsak ise onun toplamı da 2π periyotlu periyodik bir $f(x)$ fonksiyonu olacaktır (Altın 2011).

$$A_0 + \sum_{n=1}^N A_n \sin(nx + \phi_n) \quad (2.1)$$

$$\sin(nx + \phi_n) = \sin(\phi_n) \cos(nx) + \cos(\phi_n) \sin(nx) \quad (2.2)$$

$$a_0 + \sum_{n=1}^N \widetilde{a}_n \cos(nx) + \widetilde{b}_n \sin(nx) \quad (2.3)$$

(2.3) denklemindeki serinin yakınsadığı periyodik bir $f(x)$ fonksiyonunun bulunması halinde (2.4) serisine $f(x)$ 'in Fourier serisi ve a_0 , a_n , b_n sabitlerine de $f(x)$ 'in Fourier katsayıları adı verilir (Altın 2011).

$$f(x) = a_0 + \sum_{n=1}^{\infty} a_n \cos(nx) + b_n \sin(nx) \quad (2.4)$$

2π periyotlu bir $f(x)$ fonksiyonuna yakınsayan bir trigonometrik serinin bulunması için $f(x)$ 'in Dirichlet koşulları olarak bilinen aşağıdaki koşulları gerçeklemesi yeterlidir (Altın 2011).

1. $f(x)$ fonksiyonu 2π periyotlu periyodik bir fonksiyon olsun.
2. $f(x)$, $[-\pi, \pi]$ aralığında parçalı sürekli olsun.
3. $f(x)$, $[-\pi, \pi]$ aralığında sonlu sayıda ekstremuma sahip bulunsun.

Bu taktirde $f(x)$ fonksiyonu x 'in her değeri için yakınsak olan ve toplamı;

- a. x bir süreklilik noktası ise $f(x)$ 'e
- b. x bir düzgün süreksizlik noktası ise $\frac{f(x^+)+f(x^-)}{2}$ 'ye
- c. Aralığın uç noktalarında $f(-\pi) = f(\pi) = \frac{f(-\pi^+)+f(\pi^-)}{2}$ 'ye

eşit olan bir Fourier serisine açılabilir. Belirtelim ki herhangi bir trigonometrik seri yakınsak ya da ıraksak olabilir. Yakınsak olan her trigonometrik serinin bir Fourier serisi olması gerekmez. Aynı zamanda seriye karşılık gelen integrallenebilir herhangi bir $f(x)$ fonksiyonunun da olması gerekir (Altın 2011).

Bir $[a, b]$ aralığında integrallenebilen fonksiyonların bir dizisi $\{\phi_n(x)\}$ olsun. $x \in [a, b]$ için $q(x) \geq 0$ olan bir fonksiyon olmak üzere

$$\int_a^b q(x)\phi_n(x)\phi_m(x)dx = 0, m \neq n \quad (2.5)$$

Eğer (2.5) sağlanıyor ise $\{\phi_n(x)\}$ fonksiyonlar dizisine $[a, b]$ aralığında $q(x)$ ağırlık fonksiyonuna göre ortogonal (dik) bir sistem teşkil ediyor denir. Sinüs ve cosinüs gibi trigonometrik fonksiyonlar da ortogonal bir sistem oluşturmaktadırlar (Altın 2011). Bu durum şu şekilde gösterilmektedir.

i.

$$\int_{-\pi}^{\pi} \cos mx \cos nx dx = \begin{cases} 0, & m \neq n \\ \pi, & m = n \end{cases} \quad (2.6)$$

Eğer $m \neq n$ ise;

$$\int_{-\pi}^{\pi} \cos mx \cos nx dx$$

$$\begin{aligned}
&= \int_{-\pi}^{\pi} \left[\frac{\cos(m-n)x + \cos(m+n)x}{2} \right] dx \\
&= \frac{1}{2} \left[\frac{\sin(m-n)x}{m-n} + \frac{\sin(m+n)x}{m+n} \right] \Big|_{-\pi}^{\pi} \\
&= 0
\end{aligned}$$

Eğer $m = n$ ise;

$$\begin{aligned}
&\int_{-\pi}^{\pi} \cos mx \cos nx \, dx \\
&= \int_{-\pi}^{\pi} \cos^2 mx \, dx \\
&= \frac{1}{2} \int_{-\pi}^{\pi} (1 + \cos 2mx) \, dx \\
&= \frac{1}{2} \left[x + \frac{\sin 2mx}{2m} \right] \Big|_{-\pi}^{\pi} \\
&= \pi
\end{aligned}$$

ii.

$$\int_{-\pi}^{\pi} \sin mx \sin nx \, dx = \begin{cases} 0, & m \neq n \\ \pi, & m = n \end{cases} \quad (2.7)$$

Eğer $m \neq n$ ise;

$$\begin{aligned}
&\int_{-\pi}^{\pi} \sin mx \sin nx \, dx \\
&= \int_{-\pi}^{\pi} \left[\frac{\cos(m-n)x - \cos(m+n)x}{2} \right] dx \\
&= \frac{1}{2} \left[\frac{\sin(m-n)x}{m-n} - \frac{\sin(m+n)x}{m+n} \right] \Big|_{-\pi}^{\pi} \\
&= 0
\end{aligned}$$

Eğer $m = n$ ise;

$$\begin{aligned} & \int_{-\pi}^{\pi} \sin mx \sin nx \, dx \\ &= \int_{-\pi}^{\pi} \sin^2 mx \, dx \\ &= \frac{1}{2} \int_{-\pi}^{\pi} (1 - \cos 2mx) \, dx \\ &= \frac{1}{2} \left[x - \frac{\sin 2mx}{2m} \right]_{-\pi}^{\pi} \\ &= \pi \end{aligned}$$

iii.

$$\int_{-\pi}^{\pi} \sin mx \cos nx \, dx = 0, \quad \text{her } m, n \text{ için} \quad (2.8)$$

Eğer $m \neq n$ ise;

$$\begin{aligned} & \int_{-\pi}^{\pi} \sin mx \cos nx \, dx \\ &= \int_{-\pi}^{\pi} \left[\frac{\sin(m+n)x + \sin(m-n)x}{2} \right] \, dx \\ &= \frac{1}{2} \left[\frac{-\cos(m+n)x}{m+n} - \frac{\cos(m-n)x}{m-n} \right]_{-\pi}^{\pi} \\ &= 0 \end{aligned}$$

Eğer $m = n$ ise;

$$\int_{-\pi}^{\pi} \sin mx \cos nx \, dx$$

$$\begin{aligned}
&= \int_{-\pi}^{\pi} \sin mx \cos mx dx \\
&= \frac{\sin^2 mx}{2m} \Big|_{-\pi}^{\pi} \\
&= 0
\end{aligned}$$

(2.6), (2.7) ve (2.8) deki bağıntılar dikkate alınarak Fourier serisinin katsayıları aşağıdaki gibi bulunur:

i. (2.4) ifadesinin her iki yanını $-\pi$ den π ye integre edersek,

$$\begin{aligned}
\int_{-\pi}^{\pi} f(x) dx &= \int_{-\pi}^{\pi} \left[a_0 + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx) \right] dx \\
&= a_0 \int_{-\pi}^{\pi} dx + \sum_{n=1}^{\infty} \left[a_n \int_{-\pi}^{\pi} \cos nx dx + b_n \int_{-\pi}^{\pi} \sin nx dx \right] \\
&= \pi a_0 \\
a_0 &= \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x) dx
\end{aligned}$$

ii. (2.4) ifadesinin her iki yanını $\cos kx$ ile çarpar ve sonra $-\pi$ den π ye integre edersek,

$$\begin{aligned}
\int_{-\pi}^{\pi} f(x) \cos kx dx &= \int_{-\pi}^{\pi} \left[a_0 + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx) \right] \cos kx dx \\
&= \frac{a_0}{2} \int_{-\pi}^{\pi} \cos kx dx \\
&+ \sum_{n=1}^{\infty} \left[a_n \int_{-\pi}^{\pi} \cos nx \cos kx dx + b_n \int_{-\pi}^{\pi} \sin \frac{2\pi nx}{T} \cos \frac{2\pi kx}{T} dx \right]
\end{aligned}$$

$$\begin{aligned}
&= 0 + \left[a_1 \int_{-\pi}^{\pi} \cos 1x \cos kx + \dots + a_k \int_{-\pi}^{\pi} \cos kx \cos kx \right. \\
&\quad \left. + a_{k+1} \int_{-\pi}^{\pi} \cos(k+1)x \cos kx + \dots \right] \\
&\quad + \left[a_1 \int_{-\pi}^{\pi} \sin 1x \cos kx + \dots + a_k \int_{-\pi}^{\pi} \sin kx \cos kx \right. \\
&\quad \left. + a_{k+1} \int_{-\pi}^{\pi} \sin(k+1)x \cos kx + \dots \right] \\
&\quad = 0 + a_k \int_{-\pi}^{\pi} \cos^2 kx dx + 0 \\
&\quad = \pi a_k
\end{aligned}$$

$$a_k = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos kx dx, \quad k = 1, 2, \dots$$

k yerine n yazarsak

$$a_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos nx dx, \quad n = 1, 2, \dots$$

iii. (2.4) ifadesinin her iki yanını $\sin kx$ ile çarpar ve sonra $-\pi$ den π ye integre edersek

$$\begin{aligned}
&\int_{-\pi}^{\pi} f(x) \sin kx dx = \int_{-\pi}^{\pi} \left[a_0 + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx) \right] \sin kx dx \\
&= a_0 \int_{-\pi}^{\pi} \sin kx dx + \sum_{n=1}^{\infty} \left[a_n \int_{-\pi}^{\pi} \cos nx \sin kx dx + b_n \int_{-\pi}^{\pi} \sin nx \sin kx dx \right]
\end{aligned}$$

$$\begin{aligned}
&= 0 + \left[b_1 \int_{-\pi}^{\pi} \cos 1x \sin kx + \dots + b_k \int_{-\pi}^{\pi} \cos kx \sin kx \right. \\
&+ \left. b_{k+1} \int_{-\pi}^{\pi} \cos(k+1)x \sin kx + \dots \right] \\
&+ \left[b_1 \int_{-\pi}^{\pi} \sin 1x \sin kx + \dots + b_k \int_{-\pi}^{\pi} \sin kx \sin kx \right. \\
&+ \left. b_{k+1} \int_{-\pi}^{\pi} \sin(k+1)x \sin kx + \dots \right] \\
&= 0 + 0 + b_k \int_{-\pi}^{\pi} \sin^2 kx dx = \pi b_k \\
b_k &= \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin kx dx, \quad k = 1, 2, \dots
\end{aligned}$$

k yerine n yazarsak

$$b_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin nx dx, \quad n = 1, 2, \dots$$

elde edilir. Böylece (2.4) serisine ait Fourier katsayıları (2.9), (2.10) ve (2.11) deki gibi bulunur.

$$a_0 = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x) dx \quad (2.9)$$

$$a_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos nx dx \quad (2.10)$$

$$b_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin nx dx \quad (2.11)$$

Fourier serilerinin incelenmesinde bazen serinin kompleks üstel fonksiyonlar cinsinden ifade edilmesine ihtiyaç duyulur. Bunun önemli iki nedeni vardır. Birincisi, Fourier serilerinin mühendislikteki ve fizikteki bazı uygulamalarında fonksiyonların kompleks formdaki Fourier açılımlarında ve katsayılarda önemli sayılabilecek kısalık

ve netlikler ortaya çıkar. İkincisi de, kompleks Fourier serileri Fourier dönüşümlerinin ortaya çıkışına ve gelişimine alt yapı oluştururlar (Altın 2011). Bir $f(x)$ fonksiyonun kompleks Fourier serisi $f(x)$ fonksiyonunun (2.4) bağlantısındaki formunun, Fourier serisinden hareketle aşağıdaki şekilde elde edilir .

$$e^{inx} = \cos nx + i \sin nx$$

$$e^{-inx} = \cos nx - i \sin nx$$

yukarıdaki euler ifadeleri kullanılarak aşağıdaki ifadeler elde edilir.

$$\cos nx = \frac{e^{inx} + e^{-inx}}{2} \quad (2.12)$$

$$\sin nx = \frac{e^{inx} - e^{-inx}}{2i} \quad (2.13)$$

(2.12), (2.13) ve $i^2 = -1$ bağıntıları kullanılırsa, yani paydadaki i 'li ifadeyi ortadan kaldırmak için i/i ile çarpılırsa;

$$f(x) = a_0 + \sum_{n=1}^{\infty} \left[a_n \left(\frac{e^{inx} + e^{-inx}}{2} \right) + b_n \left(\frac{e^{inx} - e^{-inx}}{2i} \right) \right]$$

$$f(x) = a_0 + \sum_{n=1}^{\infty} \left[\left(\frac{a_n - ib_n}{2} \right) e^{inx} + \left(\frac{a_n + ib_n}{2} \right) e^{-inx} \right]$$

$$f(x) = c_0 + \sum_{n=1}^{\infty} [c_n e^{inx} + c_{-n} e^{-inx}]$$

$$f(x) = c_0 + \sum_{n=1}^{\infty} c_n e^{inx} + \sum_{n=-\infty}^{-1} c_n e^{inx}$$

$$f(x) = \sum_{n=-\infty}^{\infty} c_n e^{inx}$$

elde edilir. Burada,

$$c_0 = a_0 = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x) dx$$

$$c_n = \frac{a_n - ib_n}{2} = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x)[\cos nx - i \sin nx]dx = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x)e^{-inx}dx, \quad n > 0 \text{ için}$$

$$c_{-n} = \frac{a_n + ib_n}{2} = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x)[\cos nx + i \sin nx]dx = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x)e^{inx}dx, \quad n < 0 \text{ için}$$

olurlar. Böylece $f(x)$ 'in kompleks formda Fourier açılımı ve Fourier katsayıları (2.14) ve (2.15) te gösterilmektedir.

$$f(x) = \sum_{n=-\infty}^{\infty} c_n e^{inx} \quad (2.14)$$

$$c_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x)e^{-inx}dx \quad (2.15)$$

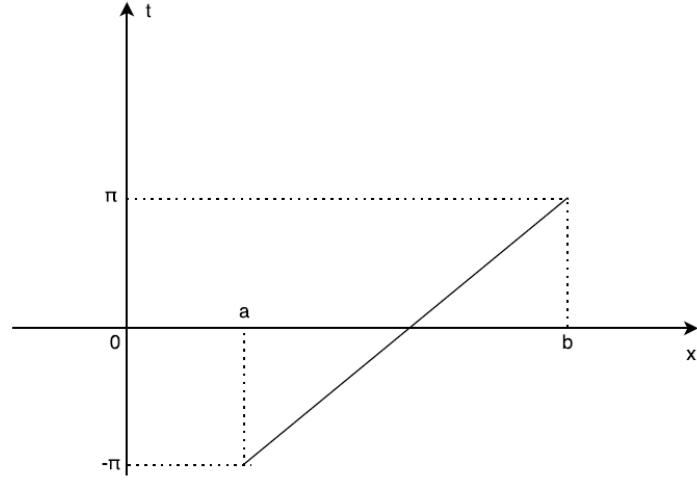
Uygulama alanlarında genellikle $f(x)$ 'in keyfi bir $[a, b]$ aralığında verilmiş olması halinde Fourier serisi gerekli olmaktadır. Böyle bir durumda yeni bir t değişkeni yardımıyla,

$$\frac{\pi - (-\pi)}{b - a} = \frac{t - \pi}{x - b}$$

$$x = \frac{b + a}{2} + \frac{b - a}{2\pi}t$$

$$t = \frac{\pi(2x - b - a)}{b - a}$$

dönüşümü yaparak $a \leq x \leq b$ aralığı $-\pi \leq t \leq \pi$ aralığına dönüştürülebilir. Bu dönüşüm Şekil 2.1'de gösterildiği gibi iki noktası bilinen bir doğru denklemdir.



Şekil 2.1: [a, b] aralığının [-π, π] aralığına dönüşmesi (Altın 2011)

Bu durumda,

$$f(x) = f\left(\frac{b+a}{2} + \frac{b-a}{2\pi}t\right) = F(t)$$

fonksiyonu, t 'ye göre 2π periyotlu bir Fourier serisine açılabilir. $F(t)$ 'nin Fourier serisi,

$$F(t) = A_0 + \sum_{n=1}^{\infty} (A_n \cos nt + B_n \sin nt)$$

Fourier katsayıları ise

$$A_n = \frac{1}{\pi} \int_{-\pi}^{\pi} F(t) \cos nt \, dt, \quad n = 0, 1, 2, \dots$$

$$B_n = \frac{1}{\pi} \int_{-\pi}^{\pi} F(t) \sin nt \, dt, \quad n = 0, 1, 2, \dots$$

olacaktır. t yerine x cinsinden olan değişkenin kullanılması bize $f(x)$ 'in $[a, b]$ aralığındaki Fourier açılımını verir. Bu durum aşağıda gösterilmektedir.

$$f(x) = A_0 + \sum_{n=1}^{\infty} \left[A_n \cos\left(\frac{n\pi(2x-b-a)}{b-a}\right) \right] + B_n \sin\left(\frac{n\pi(2x-b-a)}{b-a}\right) \quad (2.16)$$

$$A_n = \frac{2}{b-a} \int_a^b f(x) \cos\left(\frac{n\pi(2x-b-a)}{b-a}\right) dx, \quad n = 0, 1, 2, \dots \quad (2.17)$$

$$B_n = \frac{2}{b-a} \int_a^b f(x) \sin\left(\frac{n\pi(2x-b-a)}{b-a}\right) dx, \quad n = 0, 1, 2, \dots \quad (2.18)$$

Bu genel durumda eğer periyot büyüklüğü $b - a = 2L$ alınırsa $b = a + 2L$ olur ve $(a + b) / 2 = 0$ olduğundan dolayı (2.16), (2.17) ve (2.18) formülleri aşağıdaki formlere dönüşür.

$$f(x) = a_0 + \sum_{n=1}^{\infty} \left[a_n \cos\left(\frac{n\pi x}{L}\right) \right] + b_n \sin\left(\frac{n\pi x}{L}\right) \quad (2.19)$$

$$a_n = \frac{1}{L} \int_a^{a+2L} f(x) \cos\left(\frac{n\pi x}{L}\right) dx, \quad n = 0, 1, 2, \dots \quad (2.20)$$

$$b_n = \frac{1}{L} \int_a^{a+2L} f(x) \sin\left(\frac{n\pi x}{L}\right) dx, \quad n = 0, 1, 2, \dots \quad (2.21)$$

Bazen aralığı $[-L, L]$ olarak simetrik bir aralıkta inceleme yapmak gerekli olabilir. Bu durumda $T = 2L$ periyotlu $f(x)$ fonksiyonunun Fourier serisini ve Fourier katsayılarını elde etmek için (2.16), (2.17) ve (2.18) formüllerinde $a = -L$ ve $b = L$ almak ya da (2.19), (2.20) ve (2.21) formüllerinde $a = -L$ almak yeterlidir. Bu yapıldığında (2.22), (2.23) ve (2.24) formüller elde edilir.

$$f(x) = a_0 + \sum_{n=1}^{\infty} \left[a_n \cos\left(\frac{n\pi x}{L}\right) \right] + b_n \sin\left(\frac{n\pi x}{L}\right) \quad (2.22)$$

$$a_n = \frac{1}{L} \int_{-L}^L f(x) \cos\left(\frac{n\pi x}{L}\right) dx, \quad n = 0, 1, 2, \dots \quad (2.23)$$

$$b_n = \frac{1}{L} \int_{-L}^L f(x) \sin\left(\frac{n\pi x}{L}\right) dx, \quad n = 0, 1, 2, \dots \quad (2.24)$$

Bu son durumda aralık simetrik olduğundan $f(x)$ çift ya da tek fonksiyon ise aşağıdaki sonuçlar elde edilir.

a. $f(x)$ fonksiyonu $2L$ periyotlu çift bir fonksiyon ise

$$f(x) = a_0 + \sum_{n=1}^{\infty} a_n \cos\left(\frac{n\pi x}{L}\right)$$

$$a_n = \frac{2}{L} \int_0^L f(x) \cos\left(\frac{n\pi x}{L}\right) dx, \quad n = 0, 1, 2, \dots$$

b. $f(x)$ fonksiyonu $2L$ periyotlu tek bir fonksiyon ise

$$f(x) = a_0 + \sum_{n=1}^{\infty} a_n \sin\left(\frac{n\pi x}{L}\right)$$

$$b_n = \frac{2}{L} \int_0^L f(x) \sin\left(\frac{n\pi x}{L}\right) dx, \quad n = 0, 1, 2, \dots$$

Periyodik fonksiyonlarla ilgili problemlerin incelenmesinde Fourier serileri güçlü bir araçtır. Ancak pratikte karşılaşılan problemlerin bir çoğunda periyodik fonksiyonlar bulunmaz. Örneğin mekanikte impuls ya da elektrikte voltaj gibi olaylar periyodik değildir. Bir kez meydana gelen ve tekrarlanmayan bu gibi olayları açıklamak için Fourier serilerini kullanamayız. Ancak $[-L, L]$ keyfi aralığında verilen periyodik bir $f(x)$ fonksiyonunun periyodu sonsuza gittiğinde yani $L \rightarrow \infty$ için Fourier serisinin yaklaştığı limiti inceleyerek periyodik olmayan fonksiyonlar için de uygun bir gösterim elde edebiliriz. Bu durumda incelenen T boyundaki veri sonsuzda periyodikmiş gibi düşünülerek uygun bir açılım elde edilebilir. Bu tür problemlerin çözümü yani periyodik olmayan fonksiyonların Fourier serileri ile gösterilebilmesini Fourier integrali sağlar. Fourier integrali ise Fourier dönüşümünün temelini oluşturur (Altın 2011).

(2.22) ifadesindeki katsayıların yerine (2.23) ve (2.24) teki değerleri yazılırsa (2.25) teki ifadeyi elde ederiz.

$$f(t) = \frac{1}{2L} \int_{-L}^L f(t) dt + \sum_{n=1}^{\infty} \left\{ \left[\frac{1}{L} \int_{-L}^L f(t) \cos\left(\frac{n\pi t}{L}\right) dt \right] \cos\left(\frac{n\pi t}{L}\right) + \left[\frac{1}{L} \int_{-L}^L f(t) \sin\left(\frac{n\pi t}{L}\right) dt \right] \sin\left(\frac{n\pi t}{L}\right) \right\} \quad (2.25)$$

olur. $\omega_n = \frac{n\pi}{L}$ denirse $\Delta\omega_n = \omega_n - \omega_{n-1} = \frac{\pi}{L}$ olacağından (2.25) eşitliği

$$\begin{aligned}
f(t) &= \frac{1}{2\pi} \left(\int_{-L}^L f(t) dt \right) \Delta\omega \\
&+ \sum_{n=1}^{\infty} \left\{ \left[\frac{1}{\pi} \int_{-L}^L f(t) \cos(\omega_n t) dt \right] \cos(\omega_n t) \right. \\
&\left. + \left[\frac{1}{\pi} \int_{-L}^L f(t) \sin(\omega_n t) dt \right] \sin(\omega_n t) \right\} \Delta\omega
\end{aligned} \tag{2.26}$$

(2.26) eşitliğine dönüşür. $L \rightarrow \infty$ iken temel açısal frekans $\omega_0 \rightarrow 0$ olur. Yani sonsuz küçüklükte bir değer alır. Bu durumda $\omega_n = n.\omega_0$ sürekli bir değişkene dönüşür. Yani birbirini izleyen ardışık iki açısal frekans arasındaki fark $\Delta\omega \rightarrow 0$ olur. Yani $\Delta\omega$ çok küçük olur. Bu durumda $n.\omega = n.\Delta\omega$ çarpımı sürekli bir ω değişkenine yaklaşır. (2.26) ifadesi $\Delta\omega \rightarrow 0$ iken aşağıdaki şekle dönüşür.

$$\begin{aligned}
f(t) &= \lim_{\Delta\omega \rightarrow 0} \frac{1}{2\pi} \left(\int_{-L}^L f(t) dt \right) \Delta\omega \\
&+ \sum_{n=1}^{\infty} \left\{ \left[\frac{1}{\pi} \int_{-L}^L f(t) \cos(\omega_n t) dt \right] \cos(\omega_n t) \right. \\
&\left. + \left[\frac{1}{\pi} \int_{-L}^L f(t) \sin(\omega_n t) dt \right] \sin(\omega_n t) \right\} \Delta\omega
\end{aligned}$$

Bu ifade de $\Delta\omega \rightarrow 0$ için

$$\frac{1}{2\pi} \left(\int_{-L}^L f(t) dt \right) \Delta\omega \rightarrow 0$$

olur. Böylece (2.26) nın sağındaki Riemann toplamı $\Delta\omega \rightarrow 0$ için belirli bir integrale dönüşür ve f nin sağladığı belirli koşullar altında (2.26) eşitliği

$$f(t) = \int_0^{\infty} \left[\left(\frac{1}{\pi} \int_{-\infty}^{\infty} f(t) \cos(\omega t) dt \right) \cos(\omega t) + \left(\frac{1}{\pi} \int_{-\infty}^{\infty} f(t) \sin(\omega t) dt \right) \sin(\omega t) \right] d\omega \quad (2.27)$$

şeklinde ifade edilir. (2.27) ifadesini aşağıdaki şekilde gösterirsek

$$f(t) = \int_0^{\infty} [A(\omega) \cos(\omega t) + B(\omega) \sin(\omega t)] d\omega \quad (2.28)$$

$$A(\omega) = \frac{1}{\pi} \int_{-\infty}^{\infty} f(t) \cos(\omega t) dt \quad (2.29)$$

$$B(\omega) = \frac{1}{\pi} \int_{-\infty}^{\infty} f(t) \sin(\omega t) dt \quad (2.30)$$

(2.28), (2.29) ve (2.30) ifadelerinde sırasıyla Fourier integral gösterimi ve Fourier integral katsayıları elde edilir. (2.28) ifadesinde gösterilen Fourier integral gösterimini kompleks Fourier integraline dönüştürürsek

$$f(t) = \int_0^{\infty} \left[A(\omega) \left(\frac{e^{i\omega t} + e^{-i\omega t}}{2} \right) + B(\omega) \left(\frac{e^{i\omega t} - e^{-i\omega t}}{2i} \right) \right] d\omega$$

$$f(t) = \int_0^{\infty} \left\{ \frac{1}{2} [A(\omega) - iB(\omega)] e^{i\omega t} + \frac{1}{2} [A(\omega) + iB(\omega)] e^{-i\omega t} \right\} d\omega$$

$$f(t) = \int_0^{\infty} \{ C(\omega) e^{i\omega t} + \overline{C(\omega)} e^{-i\omega t} \} d\omega \quad (2.31)$$

elde edilir. (2.29), (2.30) ve (2.31) ifadeleri kullanılarak

$$C(\omega) = \frac{1}{2} [A(\omega) - iB(\omega)] = \frac{1}{2\pi} \int_{-\infty}^{\infty} f(t) [\cos(\omega t) - i \sin(\omega t)] dt$$

$$C(\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt \quad (2.32)$$

$$\overline{C(\omega)} = \frac{1}{2} [A(\omega) + iB(\omega)] = \frac{1}{2\pi} \int_{-\infty}^{\infty} f(t) [\cos(\omega t) + i \sin(\omega t)] dt$$

$$\overline{C(\omega)} = C(-\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} f(t) e^{i\omega t} dt$$

yazılabilecektir. $\overline{C(\omega)} = C(-\omega)$ olmasından dolayı,

$$\int_0^{\infty} \overline{C(\omega)} e^{-i\omega t} d\omega = \int_0^{\infty} C(-\omega) e^{-i\omega t} d\omega = \int_0^{-\infty} C(\omega) e^{i\omega t} d(-\omega) = \int_{-\infty}^0 C(\omega) e^{i\omega t} d\omega$$

olur ki bu durum (2.31) eşitliğinde dikkate alınır (2.31) eşitliği

$$f(t) = \int_0^{\infty} C(\omega) e^{i\omega t} d\omega + \int_{-\infty}^0 C(\omega) e^{i\omega t} d\omega = \int_{-\infty}^{\infty} C(\omega) e^{i\omega t} d\omega \quad (2.33)$$

şeklinde yazılabilir. (2.32) ifadesi (2.33) ifadesinde yeniden yazılırsa

$$f(t) = \int_{-\infty}^{\infty} \left\{ \frac{1}{2\pi} \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt \right\} e^{i\omega t} d\omega$$

elde edilir. İçteki integral (bütün t'ler üzerinde) sadece ω üzerinde bir fonksiyon bağımlılığı verecektir. İçteki integrali $F(\omega)$ olarak gösterirsek:

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) e^{i\omega t} d\omega \quad (2.34)$$

$$F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt \quad (2.35)$$

elde edilir. (2.35) ile tanımlanan F fonksiyonuna f 'nin Fourier dönüşümü denir.

(2.34) ile tanımlanan f fonksiyonuna da F 'nin ters Fourier dönüşümü denilmektedir.

(2.34) ve (2.35) ifadelerini simetrikliği sağlamak için aşağıdaki gibi de yazabiliriz:

$$f(t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} F(\omega) e^{i\omega t} d\omega \quad (2.36)$$

$$F(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt \quad (2.37)$$

(2.36) bağıntısı bir fonksiyon ya da sinyalin zaman domenini temsil ederken (2.37) bağıntısı frekans domenini temsil etmektedir.

2.2.1.2 Dalgacık dönüşümü

Fourier dönüşümü bir sinyalin frekans bileşenlerini analiz etmek için faydalı bir araçtır. Ancak Fourier dönüşümü tüm zaman eksenini boyunca alınırsa belirli bir frekansın hangi anda ortaya çıktığı belirlenemez. Bu sorunu gidermek için KZFD kullanılmaktadır. KZFD hem zaman hem de frekans bilgisini veren spektrogramı bulmak için kayan bir pencere kullanır. Ancak pencere uzunluğu frekans çözünürlüğünü sınırlandırmaktadır. Daha sonra Dennis Gabor tarafından KZFD'nin özel bir durumu olan Gabor dönüşümü geliştirilmiştir (Gabor 1946). Gabor dönüşümünde pencere fonksiyonu olarak Gaus fonksiyonu kullanılmaktadır. Gaus fonksiyonu sonsuz aralığa sahiptir ve bu nedenle uygulamalar için pratik değildir. Kısacası tüm bu problemlerin üstesinden gelebilmek için Dalgacık dönüşümü geliştirilmiştir. Dalgacık dönüşümü sınırlı süreli küçük dalgacıklardan oluşmaktadır.

Dalgacık dönüşümü ile ilgili ilk literatür Alfred Haar tarafından önerilen Haar dalgacıdır (Haar 1910). Yalnız “dalgacık” kavramı o zamana kadar bilinmemekteydi. “Dalgacık” kavramını ilk öneren kişi jeofizikçi Jean Morlet'tir (Morlet 1981). Daha sonra Morlet ve fizikçi Alex Grossman tarafından “dalgacık” kavramı kullanılmaya başlandı (Grossmann & Morlet 1984). 1985 yılından önce bilinen tek dalgacık Haar dalgacıydı. Hatta birçok araştırmacı Haar dalgacığına karşı başka ortogonal dalgacığın olmadığını düşünüyorlardı. Ancak matematikçi Yves Meyer, Meyer dalgacı isimli ikinci ortogonal dalgacı kurdu (Lemarié-Rieusset & Meyer 1986). Daha sonra Stephane Mallat, Meyer ile işbirliği yaparak çoklu çözünürlük kavramını önerdi (Mallat 1989b). Aynı yıllarda Ingrid Daubechies ortogonal dalgacık oluşturmak için sistematik bir metot geliştirdi (Daubechies 1988). Daha sonra Mallat hızlı Dalgacık Dönüşümü için algoritma önerdi (Mallat 1989a). Bu hızlı Dalgacık Dönüşümünün ortaya çıkmasıyla Dalgacık dönüşümü sinyal işleme alanında birçok uygulama alanı buldu.

Dalgacık dönüşümünün taban fonksiyonlarına “dalgacık” denilmekte olup, dalgacıklar bir ölçek ve dalgacık işlevi ile tanımlanmaktadır. Dalgacık dönüşümü,

Fourier dönüşümünün durağan olmayan sinyaller için eksik olan yönlerini kapatmak için geliştirilmiş bir dönüşüm yöntemidir. Bu analiz yöntemi; düşük frekanslardaki sinyaller için geniş, yüksek frekanslardaki sinyaller için dar olacak şekilde değişebilen pencere boyutlarına sahip olduğundan bütün frekans aralıklarında optimum zaman-frekans çözünürlüğü sağlayabilmektedir (Morlet 1981; Prochazka & Storek 1995). Ayrıca bu yöntem gürültüye karşı dirençli ve durağan olmayan sinyallere kolay bir biçimde uygulanabilmektedir.

Dalgacık sınırlı süresi olan ve ortalaması sıfır olan dalgacık şeklidir. Dalgacık analizinin altında yatan temel düşünce sinyali ölçeğe göre analiz etmektir ve Fourier tabanlı analiz de olduğu gibi sinüs temel fonksiyonları değil de dalgacık fonksiyonları kullanıldığından keskin süreksizlikleri içeren veriler için çok uygundur (Mallat 1989b; Miner 1998). Dalgacık dönüşümleri tüm sinyal boyunca frekans-zaman bilgisini korur. Bu sebeplerden dolayı durağan olmayan sinyallerin dalgacık temelli metotlarla işlenmesi geleneksel metotlardan daha iyi sonuçlar sağlar. Dalgacığın en önemli avantajlarından birisi, büyük sinyallerin yerel analizine olanak tanınmasıdır. Ayrıca süreksiz sinüzoidal bir sinyali ele alacak olursak, sinyalin Fourier dönüşümünde sadece sinyalin uç noktalarında pik değerleri oluşurken Dalgacık dönüşümü ile süreksizlik civarındaki değişim de gerçek zamanlı olarak gözlemlenebilmektedir (Huang 2005).

Dalgacık teorisindeki önemli bir ilerleme düzgün ana dalgacıkların keşfidir. Bu dalgacıkların bir dizi ayrık öteleme ve genişleme kümeleri $L^2(\mathbb{R})$ için bir ortonormal taban oluşturur. Burada L^2 , (2.38) de gösterildiği gibi sınırlı enerjiye sahip tüm f fonksiyonlarının kümesidir.

$$\int_{-\infty}^{\infty} |f(t)|^2 dt < \infty \quad (2.38)$$

Dalgacık dönüşümü ile Gabor dönüşümü arasındaki ana fark dalgacık dönüşümünde dalgacıkların sınırlı enerjiye sahip olmalarıdır. Gabor dönüşümünde ortonormal taban düzgün dalgacıklardan oluşmaz. Bu nedenle bu sorunun üstesinden gelebilmek için yeni bir yaklaşıma ihtiyaç vardır.

(2.39) ve (2.40) daki koşulları sağlayan kompleks değerli bir ana dalgacık olan ψ fonksiyonunu gözönüne alalım. Bu ifadelerde Ψ fonksiyonu, ψ fonksiyonunun

Fourier dönüşümüdür. (2.39) ifadesindeki koşul ψ fonksiyonunun sonlu enerjiye sahip olduğunu belirtir. (2.40) ifadesindeki kabul edilebilirlik koşulu ise $\Psi(\omega)$ fonksiyonu düzgün bir fonksiyon ise $\Psi(0) = 0$ olacağını belirtir.

$$\int_{-\infty}^{\infty} |\psi(t)|^2 dt < \infty \quad (2.39)$$

$$c_\psi = 2\pi \int_{-\infty}^{\infty} \frac{|\Psi(\omega)|^2}{|\omega|} d\omega < \infty \quad (2.40)$$

ψ ana dalgacık fonksiyonu (2.39) ve (2.40) daki koşulları karşıladığı takdirde $s(t)$ gibi gerçek bir sinyalin dalgacık dönüşümü $\psi(t)$ dalgacık fonksiyonuna göre (2.41) de gösterildiği gibi tanımlanır.

$$S(b, a) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} \psi' \left(\frac{t-b}{a} \right) s(t) dt \quad (2.41)$$

(2.41) ifadesinde ψ' ifadesi ψ fonksiyonunun kompleks eşleniğidir. ψ' fonksiyonu ($b \in \mathbb{R}, a > 0$) olmak üzere (b, a) açık aralığında tanımlıdır. (2.41) de b ifadesi zamanda kaydırma ile ilişkili bir değerdir. a parametresi ise analiz edilecek dalgacığın ölçeği ile ilişkili bir değerdir.

Eğer $\psi_{a,b}(t)$, (2.42) deki gibi tanımlanırsa bu ifade, dalgacık fonksiyonunun a kadar ölçeklenmesi ve b kadar kaydırılması anlamına gelecektir.

$$\psi_{a,b}(t) = a^{-1/2} \psi \left(\frac{t-b}{a} \right) \quad (2.42)$$

(2.41) eşitliğindeki ifade (2.42) de yer alan ifadeye göre tekrar düzenlenirse (2.43) teki yeni ifade elde edilir.

$$S(b, a) = \int_{-\infty}^{\infty} \psi'_{a,b}(t) s(t) dt \quad (2.43)$$

Ana dalgacık fonksiyonu (2.40) daki koşulu sağladığı zaman $S(b, a)$ dalgacık dönüşümünden $s(t)$ sinyali (2.44) te gösterildiği şekilde tekrar elde edilir.

$$s(t) = \frac{1}{c_\psi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} S(b, a) \psi_{a,b}(t) \frac{da db}{a^2} \quad (2.44)$$

Ayrık domende ölçekleme ve kaydırma parametreleri $a = a_0^m$ ve $b = nb_0$ şeklinde ayrıklaştırılır ve analiz edilecek dalgacıklar (2.45) teki gibi ayrıklaştırılır. Bu ifade de m ve n değerleri tamsayıdır.

$$\psi_{m,n}(t) = a_0^{-m/2} \psi\left(\frac{t - nb_0}{a_0^m}\right) \quad (2.45)$$

Buna göre ayrık dalgacık dönüşümü ve ayrık dalgacık dönüşümünün ters dönüşümü sırasıyla (2.46) ve (2.47) deki gibi gösterilmektedir. (2.47) de k_ψ ifadesi normalizasyon sabitidir.

$$S_{m,n} = \int_{-\infty}^{\infty} \psi'_{m,n}(t) s(t) dt \quad (2.46)$$

$$s(t) = k_\psi \sum_m \sum_n S_{m,n} \psi_{m,n}(t) \quad (2.47)$$

$\psi_{m,n}(t)$ fonksiyonu ölçek-zaman düzleminde örnekleme noktalarını verir. Örnekleme süreci, zaman (b-ekseni) yönünde doğrusal örnekleme; ölçek (a-ekseni) yönünde logaritmik örnekleme şeklindedir.

a_0 çok yaygın bir biçimde (2.48) deki gibi seçilir. Burada v bir tamsayı değerdir ve $\psi_{m,n}(t)$ nin v parçası bir grup olarak işlenir.

$$a_0 = 2^{1/v} \quad (2.48)$$

Bu adım *ses* (voice) olarak adlandırılır. v tamsayısı her *oktav* (octave) daki *ses*'lerin sayısıdır ve müzik anlamında iyi ayarlanmış ölçeği tanımlar. Bu durum konvansiyonel Fourier analizindeki bir dizi dar bant filtrelerinin kullanımına benzer. Dalgacık analizi diyadik ölçek analizine sınırlandırılmaz. Dalgacık analizi *oktav* başına bir dizi uygun *ses* sayısını kullanarak akustikte kullanılan 1/3 *oktav*, 1/6 *oktav* ya da 1/12 *oktav* analizlerini gerçekleştirir (Lee & Yamamoto 1994).

Literatürde birçok dalgacık fonksiyonu ailesi yer almaktadır. Bu dalgacık fonksiyonlarından bazıları aşağıda tanıtılmaktadır.

Haar dalgacıđı, dalgacık fonksiyonu türlerinden en basitidir. $\phi(t)$ (2.49) daki kořulları sađlayan bir kutu fonksiyonu olduđunu varsayalım.

$$\phi(t) = \begin{cases} 1, & 0 \leq t \leq 1 \\ 0, & \text{diđer} \end{cases} \quad (2.49)$$

Eđer $\psi(t)$ fonksiyonu $\psi(t) = \phi(2t) - \phi(2t - 1)$ řeklinde tanımlanırsa (2.50) deki fonksiyonu elde ederiz.

$$\psi(t) = \begin{cases} 1, & 0 < t \leq 1/2 \\ -1, & 1/2 < t \leq 1 \\ 0, & \text{diđer} \end{cases} \quad (2.50)$$

$\phi(t)$ fonksiyonu Haar ölçekleme fonksiyonudur ve $\psi(t)$ Haar dalgacıđıdır. Bu fonksiyon kendisinin kaydırma ve genişlemelerine diktir. Bu durum (2.51) de gösterilmektedir.

$$\psi_{m,n}(t) = 2^{-m/2} \psi(2^{-m}t - n), \quad m, n \in \mathbb{Z} \quad (2.51)$$

(2.51) de gösterildiđi gibi Haar dalgacık ailesi $L^2(\mathbb{R})$ için bir ortonormal taban oluşturur. Tarihsel olarak Haar fonksiyonu orjinal dalgacıktır. Bu dalgacık sürekli deđildir ve onun Fourier dönüşümü olan $\Psi(t)$ fonksiyonu sadece $|\omega|^{-1}$ olduđunda yani kötü frekans lokalizasyonuna denk geldiđinde çöker.

Meyer dalgacıđı, düzgün bir ortonormal dalgacık tabanı oluşturur. Hepsinden önce $\phi(t)$ ölçekleme fonksiyonunun Fourier dönüşümü olan $\Phi(\omega)$ (2.52) deki gibi tanımlanır.

$$\Phi(\omega) = \begin{cases} 1, & |\omega| \leq \frac{2}{3}\pi \\ \cos \left[\frac{\pi}{2} v \left(\frac{3}{4\pi} |\omega| - 1 \right) \right], & \frac{2}{3}\pi \leq |\omega| \leq \frac{4}{3}\pi \\ 0, & \text{diđer} \end{cases} \quad (2.52)$$

(2.52) de yer alan v ifadesi (2.53) ve (2.54) teki kořulları sađlayan düzgün bir fonksiyondur.

$$v(t) = \begin{cases} 0, & t \leq 0 \\ 1, & t \geq 1 \end{cases} \quad (2.53)$$

$$v(t) + v(1 - t) = 1 \quad (2.54)$$

Bu durumda ψ dalgacık fonksiyonu, Φ den kolay bir biçimde bulunabilir. Böylelikle ψ dalgacık fonksiyonunun Fourier dönüşümü (2.55) teki gibi bulunur.

$$\begin{aligned}\Psi(\omega) &= e^{i\omega/2} \sum_{l \in \mathbb{Z}} \Phi(\omega + 2\pi(2l + 1))\Phi(\omega/2) \\ &= e^{i\omega/2} [\Phi(\omega + 2\pi) + \Phi(\omega - 2\pi)]\Phi(\omega/2)\end{aligned}\quad (2.55)$$

Morlet dalgacığının Fourier dönüşümü kaydırılmış bir gausiyendir. Morlet dalgacığı (2.56) da gösterilmektedir. Morlet dalgacığının Fourier dönüşümü ise (2.57) de belirtilmektedir.

$$\psi(t) = (e^{-i\omega_0 t} - e^{-\omega^2/2})e^{-t^2/2} \quad (2.56)$$

$$\Psi(\omega) = e^{-(\omega-\omega_0)^2/2} - e^{-\omega^2/2}e^{-\omega_0^2/2} \quad (2.57)$$

ω_0 sıklıkla ψ nin en yüksek maksimumunun ikinci en yüksek maksimumuna oranının yaklaşık olarak 1/2 olmasına göre seçilir. Morlet dalgacığı gerçek değerli ve kompleks değerli bir dalgacıktır. Morlet dalgacığı, bir sinüs dalgasının bir gausiyen tarafından pencerelenmesinden (nokta nokta çarpılmasından) oluşur.

Daubechies dalgacığı, Ingrid Daubechies tarafından geliştirilmiş bir dalgacık ailesidir (Daubechies 1988; Daubechies 1993). Haar dalgacık tabanı hariç tüm ortonormal dalgacık tabanları sürekli destekli fonksiyonlardan oluşur. Daubechies dalgacığı da kompakt bir biçimde desteklidir. Daubechies dalgacığı dışında Symlets, Coiflets ve BiorSplines dalgacık aileleri de birçok uygulama alanında kullanılmaktadır. Symlets dalgacığı simetrikliği arttırılmış Daubechies dalgacıklarının değiştirilmiş bir türüdür.

2.2.1.3 Hilbert-Huang dönüşümü

Fourier dönüşümü gibi geleneksel veri analiz yöntemleri bir sinyalin doğrusal ve durağan olduğu varsayımı üzerine kuruludur. Dalgacık analizi ve Wagner-Wille dağılımı (Flandrin 1999; Gröchenig 2001) ise bir sinyalin durağan olmadığı ancak doğrusal olduğu varsayımı üzerine kuruludur. Bu yöntemlere ilaveten daha birçok doğrusal olmayan zaman serisi analiz yöntemleri (Tong 2002; Kantz & Schreiber 2004) geliştirildi. Ancak bu yöntemlerin de varsayımı sistemin durağan ve kararlı olduğudur. Yalnız çoğu gerçek sistemde (doğal ya da insan yapımı) sinyaller hem

doğrusal değil hem de durağan değildir. Doğrusal ve durağan olmayan sinyalleri bir varsayıma göre analiz etmek çözümden daha çok problem yaratabilmektedir. Doğrusal ve durağan olmayan veriyi temsil etmek için gerekli koşul, uyarlamalı bir temele sahip olmaktır. Önsel tanımlı bir fonksiyon güçlü bir temele dayanamaz. Bu sorunu çözmek için uyarlamalı -yani veriye bağımlı- sonsal tanımlı bir yöntemin kurulması gerekmektedir (Huang 2005). Bu sorun Huang vd. (Huang et al. 1996; Huang et al. 1998; Huang et al. 1999) tarafından geliştirilen Hilbert-Huang dönüşümü (HHD) isimli yöntemle gerçekleştirilerek hem doğrusal hem de durağan olmayan sinyallerin analiz edilmesi sağlanmıştır. Fourier dönüşümü, dalgacık dönüşümü ve HHD nin karşılaştırmalı özeti Çizelge 2.3’de gösterilmiştir (Huang & Wu 2008).

Çizelge 2.3: Fourier, dalgacık dönüşümü ve HHD nin karşılaştırmalı özeti (Huang & Wu 2008)

	Fourier	Dalgacık	HHD
Dayanak	önsel	önsel	uyarlamalı
Frekans	yayılm: global belirsiz	yayılm: bölgesel belirsiz	farklılaşım: yerel, belirli
Prezentasyon	enerji-frekans	enerji-zaman-frekans	enerji-zaman-frekans
Doğrusal sistem	evet	evet	hayır
Durağan sistem	hayır	evet	evet
Özellik çıkarımı	hayır	ayrık: hayır sürekli: evet	evet
Teorik düzlem	teori	teori	deneysel

HHD, iki bölümden oluşur: Empirik Mod Ayrışma (EMA) ve Hilbert Spektral Analizi (HSA). HHD özellikle zaman-frekans-enerji gösterimlerinde doğrusal olmayan ve durağan olmayan veri analizi için potansiyel olarak uygundur. Ayrıca HHD test edilmiş ve kapsamlı olarak geçerlilik kazanmıştır, ancak sadece empirik olarak (Huang 2005).

Empirik mod ayrışımı

EMA, durağan olmayan ve doğrusal olmayan süreçlerden gelen veriyle uğraşmak için gerekli bir yöntemdir. Önceki yöntemlerin hemen hemen hepsinin aksine, bu yeni yöntem verilere dayanan, sonsal-tanımlı, sezgisel ve uyarlanabilir. Modlara

ayırıştırma süreci, herhangi bir verinin farklı basit içsel salınım modlarından oluştuğu varsayımına dayanmaktadır. Doğrusal veya doğrusal olmayan her bir içsel mod, aynı sayıda ekstrem (sinyalin maksimum ve minimum değerleri) ve sıfır geçişli (bir matematiksel fonksiyonun işaretinin değiştiği nokta) olan basit bir salınımı temsil eder. Ayrıca salınım, "yerel ortalamaya" göre simetrik olabilmektedir ve herhangi bir anda veriler, birçok farklı salınım moduna sahip olabilmektedir. Bu salınım modlarının her biri, aşağıdaki tanıma sahip İçsel Mod Fonksiyonu (İMF) ile temsil edilir:

- Tüm veri kümesindeki ekstremum sayısı ve sıfır geçiş sayısı, en fazla bire eşit ya da en fazla birden farklı olmalıdır;
- Herhangi bir noktada, yerel maksimum ile tanımlanan zarfın ortalama değeri ve yerel minimum tarafından tanımlanan zarfın ortalama değeri sıfırdır.

Bir İMF, basit bir harmonik işleve karşılık gelen basit bir salınım modunu temsil eder, ancak daha genel bir yaklaşımla ifade edilirse: İMF, basit bir harmonik bileşen olarak sabit genlik ve frekans yerine, zamanın fonksiyonu olarak değişen genlik ve frekansa sahiptir. Orjinal sinyal, bu İMF lerin ve son monotonik eğilimin toplamı olarak ifade edilir. EMA yönteminin algoritması Şekil 2.2'de gösterilmektedir (Huang et al. 1998; Colominas et al. 2014).

1. $k = 0$ ve $r_0 = x$ in tüm ekstremumlarını bul.
2. En alt (e_{\min}) ve en üst (e_{\max}) zarfları elde etmek için r_k nın minimum ve maksimumları arasındaki ara değerleri hesapla.
3. Ortalama zarfı ($m = \frac{e_{\min} + e_{\max}}{2}$) hesapla.
4. İMF adayını ($d_{k+1} = r_k - m$) hesapla.
5. d_{k+1} bir İMF mi?
 - Evet ise; d_{k+1} i kaydet, artanı ($r_{k+1} = x - \sum_{i=1}^k d_i$) hesapla, $k = k + 1$ işlemini yap ve 2. adımda giriş verisi olarak r_k yı al.
 - Hayır ise; 2. adımda giriş verisi olarak d_{k+1} i al.
6. Son kalan r_k , durma kriterini sağlayana kadar devam et.

Şekil 2.2: EMA yönteminin algoritması (Huang et al. 1998; Colominas et al. 2014)

EMA'nın yerel doğası, bir modda çok farklı ölçeklerde salınımlar üretebilir ya da farklı modlarda benzer ölçeklerde salınımlar üretebilir. Bu durum arzu edilen bir

durum değildir ve herbir mod için benzer ölçeklerin tercih edilmesi arzu edilir. EMA'nın bu sonucu "mod karışımı" isimli bir probleme yol açmaktadır (Colominas et al. 2014). Bu sorunun üstesinden gelmek için Topluluk Empirik Mod Ayırıştırma (TEMA) (Wu & Huang 2009) isimli yeni bir yöntem önerildi. TEMA, orijinal sinyalin gürültülü kopyalarının oluşturduğu topluluk üzerinde ayrışmayı gerçekleştirir ve ortalamaya göre nihai sonuçları elde eder. Beyaz Gausiyen gürültünün eklenmesi, tüm zaman-frekans alanını doldurarak EMA'nın diyardik filtreleme davranışından yararlanarak mod karıştırmayı azaltır (Flandrin et al. 2004). Böylece tüm zaman eksenini boyunca benzer ölçeklerde çok daha düzenli modlar elde edilir. TEMA yönteminin algoritması Şekil 2.3'te gösterilmektedir (Wu & Huang 2009; Colominas et al. 2014).

1. x orjinal sinyal, $\beta > 0$ ve $w^{(i)}$ sıfır ortalamalı birim varyanslı beyaz gürültü olmak üzere;
 $x^{(i)} = x + \beta w^{(i)}$ yi hesapla ($i = 1, \dots, I$).
2. Herbir $x^{(i)}$, EMA yöntemiyle tamamen ayrıştırılır ve $d_k^{(i)}$ modları elde edilir ($k = 1, \dots, K$).
3. x 'in k . modu olarak \bar{d}_k yi ata ($\bar{d}_k = \frac{1}{I} \sum_{i=1}^I d_k^{(i)}$).

Şekil 2.3: TEMA yönteminin algoritması (Wu & Huang 2009; Colominas et al. 2014)

Tüm bu iyileştirmelere rağmen TEMA algoritması da yeni zorlukları beraberinde getirdi. Nitekim algoritmanın bir sonucu olarak; yeniden yapılandırılmış sinyal, modların toplamı ve nihai eğilim artık gürültü içermektedir. Ayrıca, sinyal ve gürültünün farklı gerçekleştirmeleri modların birbirinden farklı sayıda üretilmesine yol açmakta ve nihai ortalamayı zorlaştırmaktadır. TEMA algoritmasındaki zorlukların giderilmesi için Tamamlayıcı Topluluk Empirik Mod Ayırıştırma (TTEMA) algoritması geliştirildi (Yeh et al. 2010). TTEMA, tamamlayıcı (yani, ekleme ve çıkarma) gürültü çiftlerini kullanarak sinyali tekrar oluşturma problemini önemli ölçüde hafifletti. TTEMA yönteminin algoritması Şekil 2.4'te gösterilmektedir (Yeh et al. 2010; Colominas et al. 2014).

1. x orjinal sinyal, $\beta > 0$ ve $w^{(i)}$ sıfır ortalamalı birim varyanslı beyaz gürültü olmak üzere;
 $y_1^{(i)} = x + \beta w^{(i)}$ yi hesapla ($i = 1, \dots, I$) ve
 $y_2^{(i)} = x - \beta w^{(i)}$ yi hesapla ($i = 1, \dots, I$).
2. Herbir $y_1^{(i)}$, EMA yöntemiyle tamamen ayrıştırılır ve $d_k^{(i)}$ modları elde edilir ($k = 1, \dots, K$).

Herbir $y_2^{(i)}$, EMA yöntemiyle tamamen ayrıştırılır ve $d_l^{(i)}$ modları elde edilir ($l = 1, \dots, L$).
3. y_1 'in k . modu olarak \bar{d}_k yı ata ($\bar{d}_k = \frac{1}{I} \sum_{i=1}^I d_k^{(i)}$),
 y_2 'nin l . modu olarak \bar{d}_l yı ata ($\bar{d}_l = \frac{1}{I} \sum_{i=1}^I d_l^{(i)}$),
 $x = \bar{d}_k + \bar{d}_l$.

Şekil 2.4: TTEMA yönteminin algoritması (Yeh et al. 2010; Colominas et al. 2014)

Bununla birlikte, TTEMA algoritmasının tamlık özelliği kanıtlanamaz ve sinyalin farklı gürültülü kopyaları farklı sayıda mod üretebildiğinden dolayı nihai ortalama problemi çözülmemiş olarak kalır (Colominas et al. 2014). Tüm bu sorunların üstesinden gelebilmek için Torres vd. tarafından Gürültü Uyarlamalı Tamamlayıcı Topluluk Empirik Mod Ayrıştırma (GUTTEMA) (Torres et al. 2011) algoritması geliştirildi. GUTTEMA, ihmal edilebilir bir yeniden yapılandırma hatasının üstesinden gelerek ve sinyal artı gürültünün farklı gerçekleştirmeleri için farklı mod sayıları problemini çözerek TEMA üzerinde önemli bir gelişme sağladığını gösterdi. Bu tekniğin uygulamaları; biyomedikal mühendisliği (Navarro et al. 2012), sismoloji (Hooshmand et al. 2012; Han & van der Baan 2013) ve bina enerji tüketimi (Fontugne et al. 2013) gibi alanlarda bulunabilir. GUTTEMA yönteminin algoritması Şekil 2.5'te gösterilmektedir (Torres et al. 2011; Colominas et al. 2014). Şekil 2.5'te yer alan $E(\cdot)$ sembolü, EMA tarafından k . modun üretimini ifade eden bir operatördür.

1. EMA'yı kullanarak herbir $i = 1, \dots, I$ için $x^{(i)} = x + \beta_0 w^{(i)}$ hesapla:

$$\tilde{d}_1 = \frac{1}{I} \sum_{i=1}^I d_1^{(i)} = \bar{d}_1$$

2. İlk aşamada ($k = 1$) ilk arta kalanı hesapla: ($r_1 = x - \tilde{d}_1$)

3. EMA'yı kullanarak $i = 1, \dots, I$ için $r_1 + \beta_1 E_1(w^{(i)})$ nin ilk modunu elde et ve ikinci GUTTEMA modunu tanımla:

$$\tilde{d}_2 = \frac{1}{I} \sum_{i=1}^I E_1 \left(r_1 + \beta_1 E_1(w^{(i)}) \right)$$

4. $k = 2, \dots, K$ için k . arta kalanı hesapla:

$$r_k = r_{(k-1)} - \tilde{d}_k$$

5. EMA'yı kullanarak $i = 1, \dots, I$ için $r_k + \beta_k E_k(w^{(i)})$ nin ilk modunu elde et ve $(k + 1)$. GUTTEMA modunu tanımla:

$$\tilde{d}_{(k+1)} = \frac{1}{I} \sum_{i=1}^I E_k \left(r_k + \beta_k E_k(w^{(i)}) \right)$$

6. Bir sonraki k değeri için 4. adıma git.

Şekil 2.5: GUTTEMA yönteminin algoritması (Torres et al. 2011; Colominas et al. 2014)

EMA'nın gürültüye bağlı varyasyonlarındaki ana fikir, yeni ekstremumlar yaratmak için sinyale bazı kontrollü gürültüler eklemektir. Bu yolla yerel ortalama; sinyalin geri kalanında (ekstremumların oluşturulmadığı yerde) değiştirilmeden kalırken, yeni ekstremumların oluşturulduğu bölümlerde orjinal sinyale sadık kalmak için zorlanıldı. Yani algoritma, ölçek-enerji alanının belirli bazı değerlerine odaklanmaya zorlanıldı. Ortalamanın anlamı, sinyal artı gürültü gerçekleştirmelerinde biraz farklı olan bu yerel ortalamayı daha iyi tahmin etmektir. Ancak, TEMA bunun yerine yerel ortalamaları değil modları tahmin etmektedir. Bunun nedeni, sinyal artı gürültünün herbir gerçekleşimini bağımsız olarak ayrıştırmasıdır. Böylece her ayrıştırmanın ilk aşamasında bir yerel ortalama ve bir mod vardır. TEMA'da farklı bir şekilde ilerlemek imkansızdır, o zaman gerçek mode, biraz artan gürültü içeren orjinal sinyalin gürültülü kopyalarının modlarının bir ortalamasıdır. Öte yandan, GUTTEMA bir sonraki modeli hesaplamak için herbir nihai modu kullanır. Herbir mod, bir deflasyon şemasında sıralı olarak hesaplanır. Ardından, TEMA'dan, sinyal artı gürültünün her bir gerçekleşiminin yerel ortalamasını tahmin etmede TEMA'dan farklı bir biçimde ilerlenir. Daha sonra arta kalan sinyal ve yerel ortalamanın ortalaması arasındaki fark olarak gerçek mod tanımlanır.

Buna rağmen, GUTTEMA'da halen iyileştirilmesi gereken bazı yönler vardır. Bu yönler şunlardır:

- Modlar artık gürültü içeriyor (Colominas et al. 2014).
- Sinyal bilgisi, ayrışmanın ilk evrelerinde bazı "suni" modlarla birlikte TEMA'dan "daha sonra" görünür (Colominas et al. 2014).

Buna göre GUTTEMA'daki bu eksikleri gidermek için Colominas vd. tarafından İyileştirilmiş GUTTEMA (İGUTTEMA) (Colominas et al. 2014) algoritması geliştirilmiştir. İGUTTEMA yönteminin algoritması Şekil 2.6'da gösterilmektedir (Colominas et al. 2014). Şekil 2.6'da yer alan $M(\cdot)$ sembolü, sinyalin yerel ortalamasını sağlayan bir operatördür.

1. İlk artık sinyali ($r_1 = \langle M(x^{(i)}) \rangle$) elde etmek için $x^{(i)} = x + \beta_0 E_1(w^{(i)})$ EMA tarafından hesaplanır.
2. İlk adımda ($k = 1$) ilk modu ($\tilde{d}_1 = x - r_1$) hesapla.
3. İkinci artık sinyal ($r_1 + \beta_1 E_2(w^{(i)})$) için kestirim yap ve ikinci modu tanımla: $\tilde{d}_2 = r_1 - r_2 = r_1 - \langle M(r_1 + \beta_1 E_2(w^{(i)})) \rangle$
4. $k = 3, \dots, K$ için k . artanı hesapla: $r_k = \langle M(r_{k-1} + \beta_{k-1} E_k(w^{(i)})) \rangle$
5. k . modu hesapla: $\tilde{d}_k = r_{k-1} - r_k$
6. Bir sonraki k için 4. adıma git.

Şekil 2.6: İGUTTEMA yönteminin algoritması (Colominas et al. 2014)

Bu tez çalışmasında kullanılan İGUTTEMA ve EMA yöntemlerine ait kullandığımız MATLAB fonksiyonlarını şu adreslerden indirdik: <http://perso.ens-lyon.fr/patrick.flandrin/emd.html> ve <http://www.bioingenieria.edu.ar/grupos/ldnlys/>.

Hilbert spektral analizi

HHD'nün amacı, zaman serisi verilerinin zaman-frekans-enerji ilişkisini çıkartmaktır. Ayrıca yöntem, durağan olmayan verileri yerel olarak tanımlamaya çalışmaktadır. Anlık frekansları (AF) ve genlikleri hesaplamak ve sinyali daha yerel olarak tarif etmek için bir Fourier veya dalgacık tabanlı dönüşüm değil, Hilbert dönüşümü (HD) kullanılmıştır. HD'yi tam olarak anlayabilmek için AF kavramını iyi bilmek gerekmektedir. Bu kavramdan da önce frekans kavramına öncelikle göz atalım.

Frekans herhangi bir salınım hareketinin incelenmesi için gerekli bir niceliktir. Frekansın çok temel ve doğrudan tanımı yapılacak olursa; **(2.58)** de gösterildiği gibi periyodun tersidir.

$$f = \frac{1}{T} \quad (2.58)$$

Bu tanıma göre frekans sadece tam bir dalga hareketi varsa vardır ve frekans, hassas bir zamansal çözünürlük olmadan bu uzunluk boyunca sabit olacaktır. Aslında, araştırmacıların önemli bir kısmı, frekansın bütün bir dalga profili olmadan tanımlanamayacağını düşünmektedir (Huang et al. 2009). **(2.58)** de verilen frekans tanımına dayanarak frekansı belirlemenin yolu, ardışık dalgalanmalar arasındaki zaman aralıklarını veya ardışık dalgalar üzerindeki faza karşılık gelen noktaları ölçmektir. Bu, periyodun sabit olduğu basit bir sinüzoidal dalga için çok kolay bir biçimde uygulanır. Yalnız gerçek veriler için bu durum birkaç zorluk içermektedir: İlk olarak tüm dalga boyunca frekans değeri sabitse frekans nasıl türevlenebilir olur? İkinci olarak, durağan olmayan ve frekans modülasyonu içeren doğrusal olmayan sistemlerde her değişen frekansla detaylı frekans modülasyonu açığa çıkmaz. Son olarak karmaşık bir titreşimde, birbirini izleyen iki sıfır-geçiş arasında çok fazla ekstremum olabilir. Kısacası böyle sistemlerin üretmiş olduğu sinyaller için frekans kavramı açık değildir. Frekansın tanımından dolayı “anlık” sözcüğünü frekans kavramı ile ilişkilendirmede belirgin bir çatışma vardır. Bu nedenle AF’in tanımı, tartışmalı ve empirik olarak değerlendirilir (Boashash 1992).

AF’in hesaplanması için HD kullanılır. Bir $x(t)$ sinyalinin Hilbert dönüşümü **(2.59)** da gösterildiği gibi hesaplanır.

$$H(x(t)) = \hat{x}(t) = \frac{1}{\pi} P \int_{-\infty}^{\infty} \frac{x(\tau)}{t - \tau} d\tau \quad (2.59)$$

Bu dönüşüm sonucunda $x(t)$ sinyalinin faz açısı $\pm 90^\circ$ kaydırılarak $x(t)$ sinyali kompleks düzleme genişletilmiş olur. Daha sonra $x(t)$ sinyali ve bu sinyalin Hilbert dönüşümünün biraraya gelmesiyle sinyalin **(2.60)** da gösterildiği gibi analitik temsili oluşur.

$$z(t) = x(t) + i\hat{x}(t) = a(t)e^{i\theta(t)} \quad (2.60)$$

Bu analitik fonksiyonun anlık genliđi ve anlık fazı sırasıyla (2.61) ve (2.62) de gösterilmektedir.

$$a(t) = \{x^2(t) + \hat{x}^2(t)\}^{1/2} \quad (2.61)$$

$$\theta(t) = \tan^{-1} \frac{\hat{x}(t)}{x(t)} \quad (2.62)$$

Sonuç olarak anlık frekans (2.63) te gösterildiđi gibi hesaplanır. Yani anlık frekans, bir sinyalin anlık fazının zamana göre deđişimidir.

$$\omega(t) = \frac{d\theta(t)}{dt} \quad (2.63)$$

Genlik ve frekans fonksiyonları zaman fonksiyonları olarak ifade edildiđinden, belirli bir zamanda belirli bir frekanstaki bađıl genliđi veya enerjiyi (genliđin karesi) gösteren Hilbert spektrumu $H(\omega, t)$ olarak oluşturulabilir. Daha sonra, marjinal bir spektrum, denklem (2.64) te olduđu gibi hesaplanabilir. Marjinal spektrum, tüm veri aralıđı boyunca tüm genliklerin (enerjilerin) toplamını temsil eder.

$$h(\omega) = \int_0^T H(\omega, t) dt \quad (2.64)$$

Tüm fonksiyonların fiziksel anlık frekanslar üretme anlamında "iyi" Hilbert dönüşümü vermediđi çeşitli çalışmalarda (Huang et al. 1998; Huang et al. 1999) gösterilmektedir. Örneđin, sıfır ortalamaya sahip olmayan fonksiyonlar Hilbert dönüşümü kullanılarak negatif frekans katkısı verecektir (Huang et al. 1998; Huang et al. 1999).

2.2.2 Öznitelik seçimi

Bir sınıflandırma modelinin karmaşıklığı girdi sayısına bađlıdır. Bu deđer, böyle bir modeli eđitmenin zaman ve uzay karmaşıklığını ve gerekli öğrenme örneđi sayısını belirler. Bir modeli eđitmek için tüm girdiler yerine sadece önemli olanlarını almak zaman ve uzay karmaşıklığını azaltmamıza olanak sađlar. Bu nedenle boyut azaltma önemli bir süreç olarak karşımıza çıkmaktadır (Alpaydin 2014). Boyut azaltmayı istememizin birkaç nedeni vardır. Bunlar:

- Bellek ve zaman karmaşıklığını azaltmak.

- Bir girdinin gereksiz olduğunu anladığımızda onu elde etmek ya da ölçmek için gereken zamandan tasarruf ederiz.
- Küçük veri kümelerinde daha basit modeller daha güvenilir olur. Böylesi modellerin varyansı düşüktür.
- Veri kümesi daha az değişkenle açıklandığında bu verileri oluşturan süreci daha iyi anlayabiliriz.
- Veriyi daha iyi görselleştirebiliriz.

Boyut azaltmak için iki temel yöntem vardır. Bunlar: öznitelik seçimi ve öznitelik çıkarımıdır. Öznitelik seçimi birçok örüntü tanıma ve Yapay Zeka problemlerinin çok önemli adımlarından biridir (Zhang & Deng 2007). Öznitelik seçiminde, n adet değişkenden en çok bilgi içeren k tanesi bulunur ve öteki $(n-k)$ tanesi atılır. Öznitelik çıkarımında ise asıl n adet değişken birleştirilerek $k < n$ tane yeni değişken oluştururuz (Alpaydin 2014). Bu yöntemler çıktı bilgisini kullanıp kullanmadıklarına göre gözetimli ya da gözetimsiz olabilirler. Öznitelik çıkarımı için en iyi bilinen ve sık kullanılan yöntemler temel bileşen analizi ve doğrusal ayırtaç çözümleridir. Bu yöntemlerin ikisi de doğrusaldır ve ilki gözetimsiz, ikincisi gözetimlidir. Bunların dışında gözetimsiz yöntemlerden birkaçı şunlardır: etmen çözümlenmesi ve çok boyutlu ölçeklemedir. Doğrusal olmayan boyut azaltmaya örnek olarak eşölçümsel öznitelik eşleme ve yerel doğrusal gömme yöntemleri verilebilir.

Makine Öğrenmesi algoritmalarını etkili bir biçimde kullanabilmek için verinin önceden işlenmiş olması gereklidir. Öznitelik seçimi de verinin işlenmesi noktasında önemli yöntemlerden biridir. Öznitelik seçimi yöntemlerinin amacı ilişkisiz, gereksiz ve gürültülü özniteliklerin elenerek Makine Öğrenmesi algoritmalarının hızını ve sınıflandırma doğruluğunu arttırmaktır (Kalousis et al. 2007). İlgisiz öznitelikler, yararlı bilgi sağlamayan özelliklerdir. Ayrıca gereksiz öznitelikler, seçilen özniteliklerden daha fazla bilgi sağlamazlar. Gözetimli öğrenme açısından öznitelik seçimi üç yaklaşımdan birini kullanarak bir dizi aday özellik verir (Molina et al. 2002):

- Değerlendirme ölçütünü optimize eden özniteliklerin alt kümesinin boyutu
- Değerlendirme ölçümleri üzerinde belirli bir kısıtlamayı sağlayan alt kümenin küçük boyutu

- Genel olarak, boyut ve değerlendirme ölçümleri arasında en iyi uyuma sahip alt küme

Öznitelik seçimi, Makine Öğrenmesi algoritmalarının genelleme yeteneğini ve öğrenme hızını arttırırken model karmaşıklığını da azaltmaktadır. Öznitelik seçimi boyunca; verideki ilgisiz, gereksiz veya gürültülü öznitelikler birçok durumda engellenebilmektedir (Dash & Liu 1997). Örneklerin sayısı özniteliklerden çok daha az olduğunda, Makine Öğrenmesi algoritmaları veriyi öğrenme konusunda özellikle zorlanır. Çünkü arama uzayı yeterli büyüklükte değildir. Böylelikle model, gürültü ve ilgili veriler arasında doğru bir şekilde ayırım yapamamaktadır (Provost 2000). Öznitelik seçimi için iki önemli yaklaşım vardır. İlki Bireysel Değerlendirme, ikincisi Alt Küme Değerlendirmedir. Özniteliklerin sıralanması Bireysel Değerlendirme olarak bilinir (Guyon & Elisseeff 2003). Bireysel Değerlendirmede, tek bir özneliğin ağırlığı, uygunluğunun derecesine göre atanır. Alt Küme Değerlendirmesinde ise aday öznitelik alt kümeleri, arama stratejisi kullanılarak oluşturulmuştur (Kumar 2014).

Öznitelik seçimi için genel yöntem dört anahtar adımdan oluşmaktadır. Bunlar: Alt küme oluşturma, alt küme değerlendirme, durma kriteri ve sonuç değerlendirmedir.

Alt küme oluşturma, her bir aşamanın arama alanındaki değerlendirme için bir aday alt küme belirlediği sezgisel bir arama yöntemidir. İki temel konu alt küme oluşturma sürecinin doğasına karar verir: Bir sonraki alt küme oluşturma ve arama organizasyonudur. Birincisi; arama yönünü etkileyen arama başlangıç noktasına karar verir. Her aşamada arama başlangıç noktalarına karar vermek için ileri, geri, bileşik, ağırlıklandırma ve rastgele yöntemler kullanılır (Doak 1992). İkincisi; ardışık arama, üssel arama (Narendra & Fukunaga 1977; Pearl 1984) veya rasgele arama (Liu & Motoda 1998) gibi belirli bir stratejiyle öznitelik seçimi sürecinden sorumludur. Yeni oluşturulan bir alt küme belirli bir değerlendirme kriteri ile değerlendirilmelidir. Bu nedenle, özniteliklerin aday alt kümesinin performansını belirlemek için literatürde birçok değerlendirme kriteri önerilmiştir. Makine Öğrenmesi algoritmaları üzerindeki bağımlılıklarına dayanarak değerlendirme ölçütleri iki gruba ayrılır: bağımsız ve bağımlı kriterler (Liu & Yu 2005). Bağımsız kriterler, bir öznitelik kümesinin veya bir özneliğin performansını değerlendirmek için herhangi bir Makine Öğrenmesi algoritması içermeksizin eğitim verisinin temel özelliklerini kullanır. Bağımlı kriterler ise Makine Öğrenmesi algoritmasının

performansına dayalı öznitelikleri seçmek için önceden belirlenmiş Makine Öğrenmesi algoritmalarını içerir. Son olarak, seçim sürecini durdurmak için durdurma ölçütleri belirlenmelidir. Öznitelik seçimi işlemi doğrulama işleminde durur. Doğrulama süreci öznitelik seçimi sürecinin parçası değildir. Ancak öznitelik seçim yöntemi, farklı testler ve karşılaştırmaları daha önceden belirlenen sonuçlarla ya da yapay veri kümeleri, gerçek dünya veri kümeleri ya da her ikisini birden kullanan rakip yöntemlerin sonuçlarıyla karşılaştırarak doğrulanmalıdır (Kumar 2014).

Makine Öğrenmesi algoritmalarının veriden model oluşturma sürecini önemli bir biçimde belirleyen etmenlerden biri de öznitelik kümesidir. Öznitelik seçimi için üç genel yaklaşım vardır. Bunlar: Filtre Yaklaşımı, Sarmalama Yaklaşımı ve Gömülü Yaklaşımdır. Filtre Yaklaşımı, eğitim verisinin genel özelliklerini Makine Öğrenmesi algoritmasından bağımsız olarak kullanır yani özniteliklerin değerlendirilmesinde istatistiksel testler kullanılır (Bolón-Canedo et al. 2013). Sarmalama Yaklaşımı, ilgililik düzeyi ile en iyi öznitelik alt kümesi seçimi arasındaki ilişkiyi keşfeder ve Makine Öğrenmesi algoritmasına uyarlanmış optimal bir öznitelik alt kümesini arar (Kohavi & John 1997). Özellik seçimindeki yaklaşımlardan sonuncusu ise deneylerde kullanılan sınıflandırma algoritmalarının seçilen öznitelik gruplarından hangisi üzerinde iyi bir öğrenme sağladığı üzerinedir. Bu yaklaşım Sarmalama Yaklaşımına benzemektedir. Ancak Gömülü Yaklaşımda, arama sürecine öğrenme süreci etki etmektedir. Bu da hesaplamasal maliyeti azaltmada ve veriye olan aşırı eğilimi azaltmaktadır.

Gömülü Yaklaşım ve Sarmalama Yaklaşımı aynı zamanda fayda teorisi yaklaşımını kullanmaktadır. Fayda Teorisi bir durum hakkında belirsizlik sözkonusu olduğu zaman akılcı kararlar verme ile ilgili bir yaklaşımdır (Alpaydin 2014). Fayda teorisine göre yapılan seçimler sonucunda en iyi sonucu veren öznitelik kümesi aynı zamanda öznitelik vektörümüzü oluşturmaktadır. Bu durumu matematiksel olarak ifade edelim: Elimizde bulunan ilk öznitelik x olsun ve S durumu ayrık durumlardan oluşsun; S_k , $k = 1, \dots, n$. Buna göre x özniteliğini bildiğimizde S_k durumunun gerçekleşme olasılığı $P(S_k|x)$ olarak hesaplanır. x özniteliğini S_k durumuna atayan karar hareketi α_i ve fayda fonksiyonumuzu da U_{ik} olarak tanımlayalım. Böylelikle beklenen fayda (2.65) te gösterildiği gibidir (Alpaydin 2014).

$$EU(\alpha_i|x) = \sum_k U_{ik} \mathcal{P}(S_k|x) \quad (2.65)$$

Beklenen faydayı en yüksek yapacak hareketin α_i olduğunu varsayarsak; x özniteliği için α_i hareketinin beklenen faydası (2.66) eşitliğindeki gibi hesaplanır (Alpaydin 2014).

$$EU(x) = \max_i \sum_k U_{ik} \mathcal{P}(S_k|x) \quad (2.66)$$

Mevcut öznitelik kümesine eklenen yeni bir özniteliğimiz y olsun. Buna göre beklenen fayda (2.67) eşitliğindeki gibi tekrar hesaplanır (Alpaydin 2014).

$$EU(x, y) = \max_i \sum_k U_{ik} \mathcal{P}(S_k|x, y) \quad (2.67)$$

Böylece $EU(x, y) > EU(x)$ ise y özniteliğinin seçilen Makine Öğrenmesi algoritması için yüksek enformasyon sağlayan bir öznitelik olduğu söylenebilir. Yani öznitelik kümesine eklenen yeni özniteliğin beklenen faydası önceki öznitelik kümesinin beklenen faydasından büyük ise yeni özniteliğin yararlı bir öznitelik olduğu söylenebilir.

Literatürde birçok öznitelik seçimi yöntemi önerilmiştir. Ancak bu yöntemlerin karşılaştırmalı olarak çalışılması çok zor bir görevdir. Gerçek veri setinden önce ilişkili öznitelikleri bilmeden, öznitelik seçim yöntemlerinin etkinliğini bulmak çok zordur. Zira veri setleri çok sayıdaki ilişkisiz öznitelik, gereksiz öznitelik, gürültülü veri ya da özelliklerden veya örneklerden dolayı yüksek boyutluluk gibi birçok zorluk içerebilir. Bu nedenle öznitelik seçimi yönteminin performansı, öğrenme yönteminin performansına bağlıdır. Literatürde doğruluk, bilgisayar kaynakları, öznitelik seçimi hızı gibi birçok performans ölçütü vardır. Çoğu araştırmacı, öznitelik seçiminde "en iyi yöntem" in bulunmadığını kabul etmektedir (Bolón-Canedo et al. 2013). Bu nedenle, yeni öznitelik seçme yöntemleri, farklı stratejiler kullanarak özel problemlerin üstesinden gelmeye çalışmaktadır. Bu stratejiler aşağıda belirtilmektedir:

- Bir topluluk yöntemi kullanarak öznitelik seçiminin daha iyi bir şekilde yapılmasını sağlamak (Saeys et al. 2008; Bolón-Canedo et al. 2012)

- Ağaç topluluğu (Tuv et al. 2009) gibi başka yöntemlerle birleştirme ve öznitelik çıkarımı (Vainer 2009)
- Mevcut algoritmaları yeniden yorumlama (Sun 2007; Sun et al. 2008)
- Çözülmemiş problemlerle başa çıkmak için yeni bir yöntem oluşturma (Chidlovskii & Lecerf 2008; Loscalzo et al. 2009)
- Birkaç öznitelik seçim yöntemini birleştirmek (Zhang et al. 2008; El Akadi et al. 2011)

Öznitelik seçiminde değerlendirme kriteri, en uygun özellik alt kümesini elde etmek için kullanılır. Yüksek boyutlu verilerde (örnek sayısı $<$ özellik sayısı), optimal özellik alt kümesini bulmak zor bir iştir (Kohavi & John 1997). Bu problem NP-hard bir problem olarak bilinmektedir (Kohavi & John 1997; Blum & Rivest 1992). Yani N adet öznitelik sayısı için 2^N adet öznitelik alt kümesi bulunmaktadır. Öznitelik alt kümesinin bulunması için birçok arama yöntemi vardır. Bu yöntemler ve birbirleriyle karşılaştırmaları Çizelge 2.4’de gösterilmektedir (De Silva & Leong 2015).

Çizelge 2.4: Arama Yöntemlerinin Karşılaştırılması (De Silva & Leong 2015)

Arama Yöntemi	Karmaşıklık	Avantajlar	Dezavantajlar
Kapsamlı	$O(2^N)$	<ul style="list-style-type: none"> • Yüksek doğruluk • Optimal öznitelik kümesi 	Hesaplama maliyeti çok yüksek
Ardışık	$O(N^2)$	<ul style="list-style-type: none"> • Uygulanması basittir 	Geriye doğru iz sürme daha az esnek
Rastgele	$O(N \log N)$	<ul style="list-style-type: none"> • Kullanıcılar doğruluk ve hız arasında tercih yapabilir • Yerel optimumdan kaçınır 	Düşük doğruluk

VIBES algoritması Parkinson veri kümesine uygulanmadan önce veri kümesine ait özniteliklerin seçilmesi gerekmektedir. Çünkü onbinlerce öznitelik üzerinde Veri Madenciliği gerçekleştirmek oldukça zahmetli bir süreçtir. Veri Madenciliği sürecini hızlı bir şekilde gerçekleştirmek için Parkinson veri kümesi için oluşturulan özniteliklerden en fazla enformasyon veren (ya da sınıflandırma doğruluğu en yüksek) özniteliklerin seçilmesi gerekmektedir. Öznitelik seçimi için kullanılan öznitelik seçim algoritması *OneRAttributeEval* algoritmasıdır. Böylelikle daha yüksek enformasyon veren öznitelikler seçilirken düşük enformasyon veren öznitelikler elenmiş oldu. Özniteliklerin seçimi esnasında *OneRAttributeEval* algoritması 10-katlı çapraz doğrulama yöntemi kullanılarak çalıştırıldı. Bunun

sonucunda özniteliklerin her birinin veriyi doğru sınıflandırma yüzdeleri elde edilir. 10-katlı çapraz doğrulama yöntemi uygulandığı için doğru sınıflandırma oranı bu 10-kat'ın ortalamasıdır. Böylelikle her bir öznitelik için ($ort \pm sapma$) şeklinde sonuçlar elde edilmiş olur. Daha sonra öznitelikler seçilirken ($ort - sapma \geq 70$) şartına uyan öznitelikler seçilir. Böylelikle onbinlerce öznitelik arasından sadece sayısı yüzlerle ifade edilebilecek kadar öznitelik seçilir. Bölüm 3.2'de bu süreç ayrıntılı bir biçimde anlatılmaktadır.

2.3 Yeni Bir Topluluk Öğrenme Algoritmasının Tasarımı: VIBES

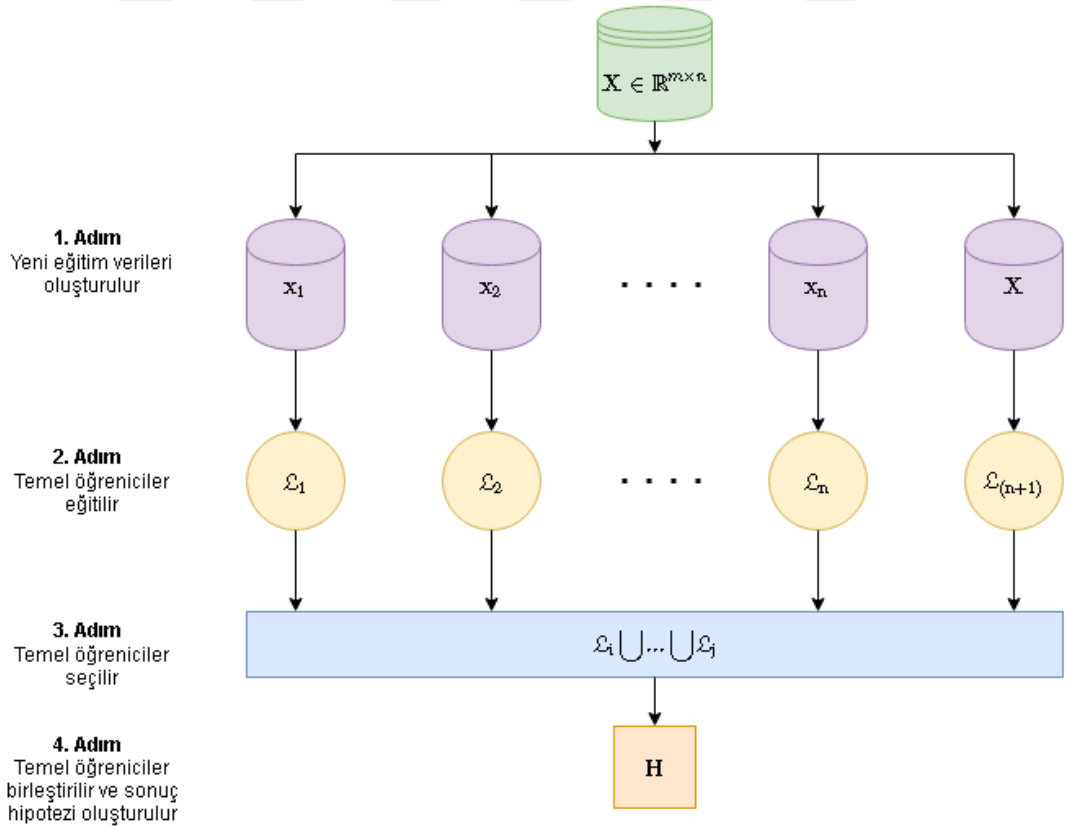
Topluluk Öğrenmede hipotez oluşturma süreci topluluğun optimize edilmesi ölçütüne bağlı olarak değişmektedir. Optimize etme ölçütünün amacı birbirinden farklı çeşitlilikte modeller elde etmek ve elde edilen modellerden uygun olanları biraraya getirmektir. Özetle Topluluk Öğrenmede model çeşitliliği ve farklı modelleri biraraya getirme stratejisi sınıflandırma doğruluğunu doğrudan etkilemektedir.

Model çeşitliliğini arttırmak için kullanılan çeşitli yöntemler vardır. Bu yöntemler şu şekildedir:

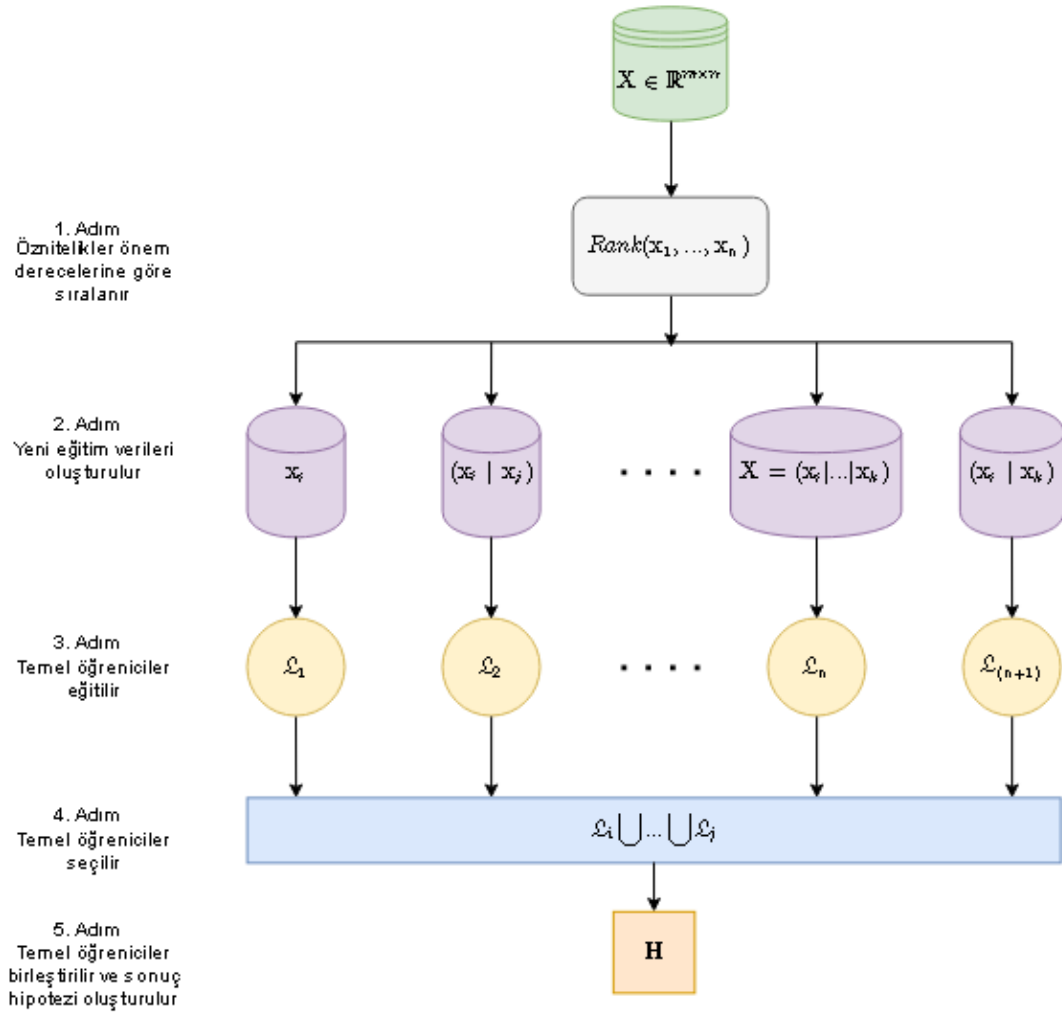
- Eğitim kümesinin parçalara ayrılarak herbir parçanın temel sınıflandırıcılar tarafından öğrenilmesidir.
- Öznitelik alt kümelerinin oluşturularak herbir alt kümenin bir temel sınıflandırıcı tarafından öğrenilmesidir.
- Sınıfların bazılarının bir temel sınıflandırıcı tarafından öğrenilirken bazılarının da başka temel sınıflandırıcılar tarafından öğrenilmesidir.

Oluşan modelleri uygun bir biçimde biraraya getirmek sınıflandırma doğruluğunu önemli ölçüde arttırmaktadır. Bunun için iki yöntemi aynı anda uygulamak gerekmektedir. Bu yöntemler: Model birleştirme yöntemi ve model seçim yöntemidir. Model birleştirme yöntemi; sınıflara göre yapılan tahminlerin olasılıklarının maksimumu, minimumu, çarpımı ve ağırlıklı çoğunluk oylaması alınarak hesaplanmaktadır. Model seçim yöntemi ise; ileri yönlü (forward), geriye doğru (backward), ileri-geri yönlü (floating), sezgisel (heuristic) ve tam arama (exhaustive) yöntemlerinden oluşmaktadır.

Bu tez çalışmasında özgün olarak geliştirilen Topluluk Öğrenme algoritması iki varsayım üzerine kuruldu: Öznitelikler birbirlerinden bağımsızdır ve öznitelikler birbirlerine bağımlıdır. Verilen bir $\mathcal{D} = \{\mathcal{X}, \mathcal{y}\}$ $\mathcal{X} \in \mathbb{R}^{m \times n}$ eğitim kümesi için; ilk varsayıma göre her bir $\mathcal{L}_i, i \in \{1, \dots, n\}$ temel öğrenici, bu eğitim kümesine ait sadece bir özneliği ve orjinal eğitim kümesinin kendisini öğrenir. Bu süreç Şekil 2.7’de gösterilmektedir. Özniteliklerin bağımlılığı varsayımına göre; ilk olarak öznitelikler enformasyon sağlama ölçütüne göre en yüksekten en düşüğe doğru sıralanır. Enformasyon ölçümü için ReliefF ve Shannon enformasyonu yöntemlerinden biri tercih edilir. Daha sonra her bir temel öğrenici sırasıyla ilk bir, ilk iki ilk n ve son olarak birinci ve n . niteliklerin oluşturduğu yeni veri kümelerini öğrenir. Bu süreç Şekil 2.8’de gösterilmektedir. Sonuç olarak bir \mathcal{X} girdisi için $n+1$ adet farklı desende temel öğrenici elde edilir (Bu desen farklılığından esinlenerek ve müzikal ritme bir analogi yaparak geliştirilen algoritmanın adı *VIBES* olarak konuldu). Daha sonra $n+1$ adet temel öğrenici içerisinde en yetkin topluluğu oluşturacak şekilde k adet temel öğrenici seçilir ve çoğunluk oylama yöntemine göre birleştirilir.



Şekil 2.7: Özniteliklerin bağımsızlığı varsayımına göre hipotez oluşturma süreci



Şekil 2.8: Özniteliklerin bağımlılığı varsayımına göre hipotez oluşturma süreci

Topluluk Öğrenmede temel öğrencilerin seçimi ve birleştirme yöntemi çok önemlidir. Bu çalışmada temel öğrenci seçiminde optimize edilmiş ileri yönlü bir algoritma ve Genetik Algoritmaları kullandık. Yapılan testler sonucunda optimize edilmiş ileri yönlü arama, Genetik Algoritmalar kullanılarak yapılan aramaya göre daha iyi sonuç vermiştir. Ayrıca Ruta ve Gabrys (Ruta & Gabrys 2005) tarafından yapılan çalışmada da ileri yönlü aramanın Genetik Algoritma ve diğer arama türlerine göre daha iyi sonuç verdiği görülmüştür. Temel öğrencilerin birleştirilmesinde ise çoğunluk oylama yöntemini kullandık. Ayrıca bu yöntemlerin tercih edilmesinde yapılan deneylerde bu parametrelerin iyi sonuçlar vermesi etkili oldu.

VIBES algoritmasının pseudo kodları Şekil 2.9'de gösterilmektedir. Buna göre

$\mathcal{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n] \in \mathbb{R}^{m \times n}$ örnek uzayı, $\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix}$ sınıf etiketleri ve \mathcal{L} bir temel

sınıflandırıcı olmak üzere; herbir temel öğrencinin oluşturduğu $h_i, i \in \{1, \dots, n\}$ hipotezi, özneliklerin birbirlerine bağımlı ya da birbirlerinden bağımsız olmaları varsayımına göre oluşturulur. Temel öğrencilerden hipotezler oluşturulduktan sonra hipotez seçimi aşamasına geçilir. Temel öğrencilerin seçim süreci ilk temel öğrencinin seçimi ile başlar. Bu süreç boyunca tüm temel öğrenciler gezilir ve en yüksek doğruluğu veren temel öğrenci seçilir. Daha sonra seçilen ilk temel öğrenci ile ikinci aday-öğrencilerin tahminleri çoğunluk oylamaya göre oylanır ve topluluğun doğruluk oranı hesaplanır. Doğruluk oranını maksimum yapan temel öğrenci seçilir. Bu işlem, tüm temel öğrenciler gezilene kadar devam eder. Bu aşamadan sonra, seçilen temel öğrencilere ait hipotezler çoğunluk oylama yöntemine göre birleştirilir ve sonuç hipotezi oluşturulur.

Algoritma vibes

Giriş: \mathcal{X} örnek uzayı,
 \mathcal{Y} sınıf etiketleri,
 \mathcal{L} temel öğrenici,
 x test örneği,
isDependent özniteliklerin bağımlılığı,
infoMeasure özniteliklerin enformasyon ölçümü
Çıkış: \mathcal{H} sonuç hipotezi

Metot:

```
if(infoMeasure is ReliefF) :
    rank ← Relieff Algorithm( $\mathcal{X}$ ,  $\mathcal{Y}$ )
else
    rank ← Shannon Entropy ( $\mathcal{X}$ ,  $\mathcal{Y}$ )

S, E ← {}
for i to rank:
    if(isDependent) :
        if( $i \neq n + 1$ ) :           S ← S  $\cup$   $\mathcal{X}^i$ 
        else                         S ←  $\mathcal{X}^{\text{rank}[1]} \cup \mathcal{X}^{\text{rank}[\text{end}]}$ 
    else
        if( $i \neq n + 1$ ) :           S ←  $\mathcal{X}^i$ 
        else                         S ←  $\mathcal{X}$ 
     $h_i \leftarrow \mathcal{L}_i(S, \mathcal{Y})$ 

for i ← 1 to n + 1 :
    C, Q ← {}
    for j ← 1 to n + 1 :
        if( $\mathcal{L}_j \notin E$ ) :
            T ←  $\mathcal{L}_j \cup E$ 
            if(Q < T) :
                C ←  $\mathcal{L}_j$ 
                Q ← T
        else
            break
    E ← C  $\cup$  E

 $\mathcal{H}(x) \leftarrow \max_{y \in \mathcal{Y}} h: x \xrightarrow{E} y$ 
```

Şekil 2.9: VIBES algoritması**2.3.1 VIBES algoritmasının zaman karmaşıklığı**

VIBES algoritması için çalışma zamanı fonksiyonu $T(n)$ iki kısımdan oluşmaktadır. İlk kısım, temel öğrencilerin eğitildiği kısmı; ikinci kısım ise temel öğrencilerin seçildiği kısmı içermektedir. Bu durum (2.68) de gösterilmektedir.

$$T(n) = T_1(n) + T_2(n) \quad (2.68)$$

İlk kısım için çalışma zamanı fonksiyonu (2.69) da gösterilmektedir.

$$T_1(n) = a \cdot (n + 1) \quad (2.69)$$

(2.69) eşitliğinde a değeri temel öğrencilerin eğitim zamanını ifade etmek için kullanılan bir sabittir. İkinci kısım için çalışma zamanı fonksiyonu (2.70)'te gösterilmektedir.

$$T_2(n) = b \cdot \left(\frac{(n + 1)(n + 2)}{2} \right) \quad (2.70)$$

(2.70) eşitliğinde b değeri temel öğrencilerin seçimi esnasında yapılan işlemlerin zamanını ifade etmektedir. Yalnız (2.70)'deki ifade optimize edilmiş olan temel öğrenci seçim zamanını göstermemektedir. Bunun yerine tüm temel öğrencilerin denendiği durumları oluşturan algoritmanın zamanını ifade etmektedir. Optimize edilmiş olan algoritmanın çalışma zamanını ifade edecek genel bir fonksiyon yoktur. Çünkü her sınıflandırma problemi için oluşturulan modeller değişmektedir. Aynı zamanda oluşan bu modeller, temel öğrencilere bağlı olarakta değişmektedir. Kısacası optimize edilmiş böyle bir Greedy algoritmasının çalışma zamanı için genel bir fonksiyon bulunmamaktadır. Bu nedenle optimize edilmiş algoritmanın çalışma zamanı genel bir fonksiyon olan $f(\cdot)$ fonksiyonu ile ifade edilecektir. Böylelikle (2.70)'deki ifade (2.71)'deki ifadeye dönüşür ve optimize edilmiş temel öğrenci oluşturma algoritması ile optimize edilmemiş algoritma arasında $T_3(n) \leq T_2(n)$ ilişkisi oluşur.

$$T_3(n) = b \cdot f \left(\frac{(n + 1)(n + 2)}{2} \right) \quad (2.71)$$

Buna göre VIBES algoritmasının toplam çalışma zamanı (2.72)'deki gibi olur. Bu ifade de yer alan a ve b katsayıları arasında her zaman $a \geq b$ ilişkisi vardır. Çünkü bir temel öğrenciyi eğitmek için geçen süre, bir temel öğrenciyi seçmek için geçen süreden daha fazla olmaktadır. Bu durumları dikkate alarak VIBES algoritmasının çalışma zamanını en iyi ve en kötü duruma göre analiz edeceğiz.

$$T(n) = a \cdot (n + 1) + b \cdot f \left(\frac{(n + 1)(n + 2)}{2} \right) \quad (2.72)$$

Şimdi (2.72)'deki ifadeyi en kötü duruma göre analiz edelim. Buna göre çalışma zamanı fonksiyonunu girdi büyüklüğü olan n değerine göre yazıp a ve b değerlerini 1 olarak alırsak ve $f(\cdot)$ fonksiyonunu birim fonksiyon olarak kabul edersek (2.74)'teki ifadeyi elde ederiz.

$$T(n) = (n + 1) + \frac{(n + 1)(n + 2)}{2} \quad (2.73)$$

$$T(n) = \frac{n^2 + 5n + 4}{2} \quad (2.74)$$

Buna göre algoritmanın zaman karmaşıklığını üst sınır olan *big - O* notasyonuna göre yazarsak; ilk önce $c > 0$ ve $\forall n \geq n_0$ için $f(n) \leq c \cdot g(n)$ eşitsizliğini sağlayan c ve n_0 değerlerini bulmamız gerekmektedir. Bu eşitsizlik için uygun değerler $c = 1$ ve $n_0 = 6$ olarak bulunur ve $g(n) = n^2$ için zaman karmaşıklığı $O(n^2)$ olarak bulunur.

Şimdi (2.72)'deki ifadeyi en iyi duruma göre analiz edelim. Buna göre çalışma zamanı fonksiyonunu girdi büyüklüğü olan n değerine göre yazıp a ve b değerlerini 1 olarak alırsak ve $f(\cdot)$ fonksiyonunu optimize edilmiş topluluk oluşturma algoritmasının en iyi durum fonksiyonu olarak kabul edersek (2.76)'daki ifadeyi elde ederiz.

$$T(n) = (n + 1) + (1 + \ln n) \quad (2.75)$$

$$T(n) = 2 + \ln n + n \quad (2.76)$$

Buna göre algoritmanın zaman karmaşıklığını alt sınır olan *big - Ω* notasyonuna göre yazarsak; ilk önce $c > 0$ ve $\forall n \geq n_0$ için $0 \leq c \cdot g(n) \leq f(n)$ eşitsizliğini sağlayan c ve n_0 değerlerini bulmamız gerekmektedir. Bu eşitsizlik için uygun değerler $c = 3$ ve $n_0 = 1$ olarak bulunur ve $g(n) = n$ için zaman karmaşıklığı $\Omega(n)$ olarak bulunur.

Böylelikle *VIBES* algoritmasının zaman karmaşıklığı en iyi ve en kötü durumlar için sırasıyla $\Omega(n)$ ve $O(n^2)$ olarak bulunur.

2.3.2 VIBES algoritmasının bellek karmaşıklığı

VIBES algoritması için bellek karmaşıklığı fonksiyonu $f(n, m)$ iki kısımdan oluşmaktadır. İlk kısım, temel öğrencilerin eğitildiği kısmı; ikinci kısım ise temel öğrencilerin seçildiği kısmı içermektedir. Bu durum (2.77) da gösterilmektedir.

$$f(n, m) = f_1(n, m) + f_2(n, m) \quad (2.77)$$

İlk kısım için bellek karmaşıklığı fonksiyonu (2.78) de gösterilmektedir.

$$f_1(n, m) = a + nm \quad (2.78)$$

Bu eşitlikte a değeri bir sabit değeri ifade etmektedir. İkinci kısım için bellek karmaşıklığı fonksiyonu (2.79) de gösterilmektedir.

$$f_2(n, m) = b + (n + 1)m \quad (2.79)$$

Bu eşitlikte b değeri bir sabit değeri ifade etmektedir. Toplam bellek karmaşıklığı ise (2.80) daki gibi olur.

$$f(n, m) = a + nm + b + (n + 1)m \quad (2.80)$$

Çalışma zamanı fonksiyonunu girdi büyüklüğü olan n, m değerine göre yazıp a ve b değerlerini 0 olarak alırsak (2.81) daki ifadeyi elde ederiz.

$$f(n, m) = 2 \cdot n \cdot m + m \quad (2.81)$$

Bu durumda n ve m değerlerinin birbirlerine göre durumları en kötü durum senaryosuna ($n \sim m$ ya da $n \cong m$) göre oluşturulursa (2.82) de gösterilen durum elde edilir.

$$f(n, m) = \begin{cases} 2(m - 1)m + m, & n < m \\ 2mm + m, & n = m \\ 2n(n - 1) + (n - 1), & n > m \end{cases} \quad (2.82)$$

(2.82) deki ifadeyi düzenlersek (2.83) de gösterilen ifadeyi elde ederiz.

$$f(n, m) = \begin{cases} 2m^2 - m, & n < m \\ 2m^2 + m, & n = m \\ 2n^2 - n - 1, & n > m \end{cases} \quad (2.83)$$

Buna göre algoritmanın bellek karmaşıklığını her bir durum için en üst sınır olan *big - O* notasyonuna göre yazarsak bellek karmaşıklığı sırasıyla (2.84), (2.85) ve (2.86) de gösterildiği gibi olur.

$$c = 2 \text{ ve } m_0 = 1 \text{ için } O(m^2) \quad (2.84)$$

$$c = 3 \text{ ve } m_0 = 1 \text{ için } O(m^2) \quad (2.85)$$

$$c = 2 \text{ ve } n_0 = 1 \text{ için } O(n^2) \quad (2.86)$$

(2.84), (2.85) ve (2.86) teki ifadeleri birleştirecek olursak bellek karmaşıklığı (2.87) deki gibi olur.

$$f(n, m) = O\left(\left(\max_{n, m \in \mathbb{N}^+} \{n, m\}\right)^2\right) \quad (2.87)$$

(2.81)'deki ifadeye göre n ve m değerlerinin birbirlerine göre durumları en iyi durum senaryosuna ($\frac{n}{m} \cong n$ ya da $\frac{m}{n} \cong m$) göre oluşturulursa (2.88) de gösterilen durum elde edilir.

$$f(n, m) = \begin{cases} 2n, & \frac{n}{m} \cong n \\ 3m, & \frac{m}{n} \cong m \end{cases} \quad (2.88)$$

Buna göre algoritmanın bellek karmaşıklığını her bir durum için en alt sınır olan *big - Ω* notasyonuna göre yazarsak bellek karmaşıklığı sırasıyla (2.89) ve (2.90) da gösterildiği gibi olur.

$$c = 2 \text{ ve } n_0 = 1 \text{ için } \Omega(n) \quad (2.89)$$

$$c = 3 \text{ ve } m_0 = 1 \text{ için } \Omega(m) \quad (2.90)$$

(2.89) ve (2.90) ifadelerini birleştirecek olursak bellek karmaşıklığı (2.91) deki gibi olur.

$$f(n, m) = \Omega\left(\max_{n, m \in \mathbb{N}^+} \{n, m\}\right) \quad (2.91)$$

Böylelikle *VIBES* algoritmasının bellek karmaşıklığı en iyi ve en kötü durumlar için sırasıyla $\Omega(\max_{n,m \in \mathbb{N}^+} \{n, m\})$ ve $O((\max_{n,m \in \mathbb{N}^+} \{n, m\})^2)$ olarak bulunur.

2.3.3 İkili sınıflandırma problemleri için topluluk öğrencilerde çoğunluk oylama yönteminin uygunluğunun teorik gösterimi

Bir Topluluk Öğrenci için en uygun tahmin kararı (yani minimum yanlış tahmin kararı), temel öğrencilerin yapmış olduğu $\mathcal{L}(X) = p$ için olası gerçek a değerini bulmaktır. n tane temel öğrenci tarafından yapılan $\mathbf{p} = p_1 \cdots p_n$ tahminlerini ve gerçek değer a 'yı ($a = f + \varepsilon$) gözönüne alalım. Buna göre Bayes Teoremi yoluyla a 'nın sonsal olasılığı (2.92)'de gösterilmektedir.

$$P(a|p_1 \cdots p_n) = \frac{P(p_1 \cdots p_n|a)P(a)}{P(p_1 \cdots p_n)} \quad (2.92)$$

Böylece iki sınıflı (1 ve 0) bir problem için sonsal olasılık (2.93) ve (2.94)'deki gibi ifade edilir.

$$P(a = 1|p_1 \cdots p_n) = \frac{P(p_1 \cdots p_n|a = 1)P(a = 1)}{P(p_1 \cdots p_n)} \quad (2.93)$$

$$P(a = 0|p_1 \cdots p_n) = \frac{P(p_1 \cdots p_n|a = 0)P(a = 0)}{P(p_1 \cdots p_n)} \quad (2.94)$$

(2.93) ve (2.94)'deki sonsal olasılık değerleri iki çarpanla hesaplanır. Bu çarpanlardan biri önsel olasılık olan $P(a)$ ve a 'nın olabirliği (likelihood) olarak adlandırılan veri-bağımlı ifade $P(p_1 \cdots p_n|a)$ 'dır. Normalizasyon sabiti $P(p_1 \cdots p_n)$ ise en uygun tahmin kararını hesaplamada kullanılmaz.

$P(a = 1|\mathbf{p})$ ve $P(a = 0|\mathbf{p})$ 'yi bulmak için $a=0$ ve $a=1$ gibi iki hipotezin önsel olasılıkları hakkında bir varsayım yapmak gerekmektedir. Aynı zamanda verilen bir a için \mathbf{p} 'nin olasılığı hakkında bir varsayım yapmak gerekmektedir. Bunun için öncelikle $P(a = 1) = P(a = 0) = 0.5$ şeklindeki önsel olasılıkların eşit olduğunu varsayıyoruz. Ayrıca $P(a|\mathbf{p})$ sonsal olasılığını maksimize etmek $P(\mathbf{p}|a)$ 'nin olabirliğini maksimize etmeye eşittir. Bunun yanı sıra her bir temel öğrencinin hata oranını $e < 0.5$ olarak varsayıyoruz. Çünkü en kötü senaryo da bile çoğunluk tahmin edicisi örneklerin en az yarısını doğru olarak tahmin etmektedir. Yani hata oranı en kötü durumda $e = 0.5$ olmaktadır. Ancak sınıflandırma problemlerine uygulanan Makine Öğrenmesi algoritmaları en kötü durumda bile çoğunluk tahmin edicisinin

yapmış olduğu tahmin doğruluğuna eşit olmaktadır. Bu nedenle temel öğrencilerin yaptıkları tahminler için hata oranları $e < 0.5$ olarak varsayılmaktadır. Böylece $P(p|a)$ olabilirliği (2.95)'deki gibi hesaplanır. Herbir temel öğrencinin olabilirliği ise (2.96)'deki gibi hesaplanır.

$$P(p|a) = \prod_{i=1}^n P(p_i|a) \quad (2.95)$$

$$P(p_i|a) = \begin{cases} (1-e), & p_i = a \\ e, & p_i \neq a \end{cases} \quad (2.96)$$

Böylece iki hipotezin olabilirlik oranı (2.97)'te gösterildiği gibi hesaplanır.

$$\frac{P(p|a=1)}{P(p|a=0)} = \prod_{i=1}^n \frac{P(p_i|a=1)}{P(p_i|a=0)} \quad (2.97)$$

$\frac{P(p_i|a=1)}{P(p_i|a=0)}$ ifadesi $p_i = 1$ ise $\left(\frac{1-e}{e}\right)^n$ 'e eşittir yok eğer $p_i = 0$ ise $\left(\frac{e}{1-e}\right)^n$ 'e eşittir ve $e < 0.5$ için $\left(\frac{1-e}{e}\right)^n$ oranı her zaman 1'den daha büyük olur. Çoğunluk oylamalı yöntem yerine minimum ya da maksimum olasılıklı tahminin sonucuna karar verme yöntemlerin de ise bu oran 1'e eşit olacaktır. Çünkü olasılığın frekans tanımından faydalanacak olursak ve önsel olasılıkları dikkate alacak olursak maksimum ya da minimum olasılıklı durumun 1 ya da 0 gelme olasılığı eşit olacaktır. Yani tek bir temel öğrencinin yapmış olduğu tahmin ya 0 ya da 1'dir. Bu durum tüm diğer temel öğrenci tahminleri için de aynıdır. Böylelikle yapılan bir tahminde, 1 ya da 0 gelme olasılığı için maksimum ya da minimum olasılık gelme olasılıkları da eşit olmaktadır. Sonuç olarak, çoğunluk oylamalı yöntemin topluluk oluşturma sürecinde diğer yöntemlere kıyasla teorik olarak en uygun yöntem olduğu görülmektedir.

2.3.4 Çoğunluk oylamalı topluluk öğrencilerde sınıflandırma hatası tahmini için önerilen yöntem

Bir Topluluk öğrenci hatalı tahminler yaptığında bazı sorular ortaya çıkmaktadır. Bu sorulardan en önemli ikisi şudur: “Bireysel tahmin edicilerin tahmin doğrulukları topluluğun tahminlerini ne kadar etkilemektedir?” ve “bireysel tahmin edicilerin etkileşimleri topluluğun tahminlerinde ne kadar etkilidir?”. Bu iki soru iki kavramı gündeme getirmektedir. Bu kavramlar: Doğruluk ve çeşitliliktir. Hata fonksiyonunun türüne ve birleştirme yöntemine bağlı olarak, topluluk hatasındaki doğruluk-çeşitlilik

kırılmasının her zaman mümkün olmadığı anlaşılmaktadır (Brown 2011). Bu iki terim arasında bir değişinti fikri Bias-Varyans ayrışmasını (Geman et al. 1992) anımsatmaktadır.

Bias-Varyans (underfitting-overfitting) ikilemi Makine Öğrenmesi algoritmalarının çalışma mantığını anlamada önemli bir araçtır. Son yıllarda yapılan deneysel çalışmalarda bias-varyans ikileminin kullanımı bir hayli artmaktadır. Bias ve varyans kavramları çok basit öğrenme algoritmalarının çok kompleks olanlar karşısında ne kadar performanslı olabileceğini ve model topluluklarının basit modeller karşısında ne kadar performanslı olabileceğini göstermeye yardımcı olmaktadır. Bias-Varyans ikilemi çeşitli istatistiksel hata fonksiyonları için türetildiği gibi karesel kayıplar (squared loss) için de türetilmiştir (Domingos 2000). Bu türetme Geman vd. (Geman et al. 1992) tarafından yapılan çalışmada gösterilmiştir. Bu gösterimde Makine Öğrenmesi algoritmasının ortalama karesel hatası, bias'ın karesi ile varyans'ın toplamı olarak ifade edilmiştir. Bu gösterim dışında, Bias-Varyans ikilemi farklı biçimlerde de gösterilmektedir. Örneğin yapılan bu çalışmalardan bir tanesi 0-1 kaybı içindir. Yalnız gerçekleştirilen bu gösterimlerin bir kısmı önemli derecede eksiklikler içermektedir (Domingos 2000; James 2003). Örneğin Kong ve Dietterich'in (Kong & Dietterich 1995) 0-1 kaybı için yapmış olduğu gösterim de varyans negatif çıkmakta ve gürültü kaynaklı sorunlar dikkate alınmamaktadır (Domingos 2000). Hata fonksiyonu olarak karesel kayıplara dayalı hata fonksiyonunu kullanmak çok uygundur. Bunun nedeni ise karesel hatanın Bias-Varyans ikilemi gibi iyi bilinen matematiksel özelliklere sahip olmasıdır (James 2003). Buna göre bireysel bir tahmin edici için Bias-Varyans ikilemi **(2.101)** de gösterildiği gibi karesel kayıp fonksiyonu türünden ifade edilir (Geman et al. 1992).

$$bias^2 = E[(f(x) - E[d])^2] \quad (2.98)$$

$$varyans(d) = E[(E[d] - d)^2] \quad (2.99)$$

$$varyans(gürültü) = E[\varepsilon^2] \quad (2.100)$$

$$E[(f(x) - d)^2] = bias^2 + var(d) + var(gürültü) \quad (2.101)$$

Yalnız topluluk öğrencilerin hata analizinde Bias-Varyans ayrışımı tek başına yeterli olmamaktadır. Çünkü topluluk öğrenciler birbirleriyle etkileşim içerisinde olan

bireysel tahmin edicilerden oluşmaktadır. Bu nedenle (2.101) deki ifadenin topluluk öğrenciler için tekrar düzenlenmesi gerekmektedir. Bu düzenleme ile topluluk öğrencileri için Bias-Varyans-Kovaryans ayrışımı (Ueda & Nakano 1996) gerçekleştirilir. Bu ayrışımın genel ifadesi (2.102) de gösterilmektedir (Ueda & Nakano 1996).

$$E \left[(\bar{f}(x) - d)^2 \right] = \overline{bias}^2 + \frac{1}{T} \overline{var}(d) + \left(1 - \frac{1}{T}\right) \overline{kov} + var(gürültü) \quad (2.102)$$

Çoğunluk oylamalı topluluk öğrencilerde sınıflandırma hatasını önceden kestirme, tüm diğer birleştirme yöntemlerinden daha zordur. Genel olarak, topluluk sınıflandırma hatasının, düzgün doğruluk ve çeşitlilik bileşenlerini içerecek biçimde bilinen bir dökümü yoktur. Modeller arasındaki korelasyonun performansı etkilediğini gösteren en basit sezgi, Binom teoremi tarafından verilir (Brown 2011). Buna göre her biri aynı hata olasılığına sahip T adet modele sahipsek ve istatistiksel olarak bağımsız hatalar yaptıklarını varsayarsak, çoğunluk oylama topluluğu (2.103) de gösterilen hata ihtimaline sahiptir (Brown 2011).

$$P(H(x) \neq y) = \sum_{k>(T/2)}^T \binom{T}{k} p^k (1-p)^{(T-k)} \quad (2.103)$$

Örneğin $T=10$ ve $p=0.5$ için çoğunluk oylama hatası 0.205 olacaktır. Ancak bu durum sadece istatistiksel olarak bağımsız hatalar olduğunda söz konusu olmaktadır. Yani birbiriyle ilişkili durumlar büyük bir problem olarak karşımıza çıkmaktadır. Bunun yerine çeşitli araştırmacılar, çoğunluk oylama topluluklarında kendi sezgisel çeşitlilik tanımlamalarını önerdiler. Bu araştırmacıların önde gelen isimlerinden Kuncheva (Kuncheva 2004), önerilen birkaç çeşitlilik ölçümü çalışmalarına önderlik etti. Yalnız sonuç şu idi: "Herhangi bir ölçüm çoğunluk oyunun doğruluğu ile tutarlı bir şekilde ilişkilendirilmemektedir". Buna rağmen, bazı araştırmacılar topluluk yönteminin performansını karakterize eden yaklaşık bir ölçüm olarak bu çalışmaları yararlı buldu. Ancak çeşitlilik üzerine "son söz" olarak bu ölçümlere güvenilmemelidir. Bu durumda Kuncheva'nın önerisi (Kuncheva 2004) basitlik ve hesaplama kolaylığı nedeniyle Q-istatistiğidir.

Topluluk öğrencilerin hata analizlerini yaparken ortaya çıkan en önemli sorun temel öğrencileri birleştirme yönteminin karmaşıklığıdır. Örneğin çoğunluk oylama yöntemi ile farklı çeşitlilikteki modeller birleştirilirken çok farklı topluluklar ortaya

çıkabilmektedir. Bu da topluluğun tahmin hatasını belirlemeyi güçleştirmektedir. Topluluğun tahmin hatasını belirlemek yerine topluluğun en düşük ve en yüksek olası tahmin hatası belirleyebilmek daha olasıdır. Bu tez çalışmasında da çoğunluk oylamalı topluluk öğrencilerin hata analizlerini yapmak için bir yöntem önerildi. Bu yöntem çoğunluk oylamalı bir topluluk öğrencinin en düşük ve en yüksek olası tahmin hatasını vermektedir. Ayrıca topluluğu oluştururken uygulanan yöntemin optimize edilmiş olması durumunda; topluluğun tahmin hatası, geliştirilen yöntemin en düşük olası tahmin hatasına yakındır. Ancak optimize edilmemiş bir birleştirme de; topluluğun hata oranı, geliştirilen yöntemin ürettiği en yüksek olası tahmin hatasına yakın olmaktadır.

Geliştirilen yöntem empirik olarak geliştirildi ve sezgilerimizle uyumluluk göstermektedir. Yapılan deneysel sonuçlar da yöntemin başarılı olduğunu göstermektedir. 4. Bölümde bu sonuçlar gösterilmektedir.

$e = [e_1, e_2, \dots, e_T]$ vektörü herbir temel öğrencinin yapmış olduğu hata olasılığı olsun. Buna göre çoğunluk oylamalı topluluk öğrencilerin minimum ve maksimum hata tahmini T değerinin çift ya da tek olmasına bağlı olarak aşağıdaki adımlar uygulanarak hesaplanır.

1. Eğer minimum hata tahmini hesaplanacaksa e vektörü büyükten küçüğe doğru sıralanır ve e_{min} vektörü oluşturulur. Diğer türlü e vektörü küçükten büyüğe doğru sıralanır ve e_{max} vektörü oluşturulur.
2.
 - a. T değeri çift ise;
 - i. Minimum hata tahmini hesaplanacaksa, e_{min} vektörünün ilk elemanı (en büyük değeri) diziden çıkarılır ve e_{min} vektörünün son elemanı (en küçük değeri) dizinin sonuna eklenir.
 - ii. Maksimum hata tahmini hesaplanacaksa, e_{max} vektörünün ilk elemanı (en küçük değeri) diziden çıkarılır ve e_{max} vektörünün son elemanı (en büyük değeri) dizinin sonuna eklenir.
 - b. T değeri tek ise; e_{min} ve e_{max} vektörlerinde herhangi bir değişiklik yapılmaz.

3.

- a. e_{min} vektörü için minimum hata tahmini ε_{min} şu şekilde hesaplanır: Vektörün ilk iki elemanının harmonik ortalaması hesaplanır. Daha sonra bu sonuç ile vektörün üçüncü elemanının harmonik ortalaması hesaplanır. Bu süreç son elemana kadar aynı şekilde gerçekleştirilerek ε_{min} değeri hesaplanır. Herhangi bir değerın sıfır olması durumunda sıfır olan elemanın değeri $\frac{1}{m}$ olacak şekilde alınır.
- b. e_{max} vektörü için maksimum hata tahmini ε_{max} şu şekilde hesaplanır: Vektörün ilk iki elemanının aritmetik ortalaması hesaplanır. Daha sonra bu sonuç ile vektörün üçüncü elemanının aritmetik ortalaması hesaplanır. Bu süreç son elemana kadar aynı şekilde gerçekleştirilerek ε_{max} değeri hesaplanır.

e_{max} vektörü yukarıda anlatıldığı şekliyle oluşturulduktan sonra ε_{max} in değeri adım adım (2.104) teki gibi hesaplanır. Bu ifadeyi düzenlersek (2.105) teki ifadeyi elde ederiz. Bu ifadeyi en genel biçimde yazarsak (2.106) teki ifadeye dönüşür. Bu ifadenin matematiksel gösterimi (2.107) da belirtilmektedir.

$$\varepsilon_{max} = \frac{\left(\frac{\left(\frac{\left(\frac{e_1 + e_2}{2} \right) + e_3}{2} \right) + e_4}{2} \right) + \dots}{2} + e_T \quad (2.104)$$

$$\varepsilon_{max} = \frac{e_1 + e_2 + 2^1 e_3 + 2^2 e_4 + \dots + 2^{(T-2)} e_T}{2^{(T-1)}} \quad (2.105)$$

$$\varepsilon_{max} = \frac{e_1 + \dots + 2^{(T-2)} e_T}{2^{(T-1)}} \quad (2.106)$$

$$\varepsilon_{max} = \frac{1}{2^{(T-1)}} \left(e_1 + \sum_{t=2}^T 2^{(T-2)} e_t \right), \quad T \geq 2 \quad (2.107)$$

e_{min} vektörü yukarıda anlatıldığı şekliyle oluşturulduktan sonra ε_{min} in değeri adım adım (2.108) deki gibi hesaplanır. Bu ifadeyi düzenlersek (2.109) deki ifadeyi elde

ederiz. Bu ifadeyi en genel biçimde yazarsak (2.110) daki ifadeye dönüşür. Bu ifadenin matematiksel gösterimi (2.111) de belirtilmektedir.

$$\varepsilon_{min} = \frac{2^{(T-1)} e_1 e_2 e_3 e_4 \cdots e_T}{\left(\frac{e_1 e_2 e_3 e_4 \cdots e_T}{e_1} + \frac{e_1 e_2 e_3 e_4 \cdots e_T}{e_2} + \cdots + \frac{2^{(T-2)} e_1 e_2 e_3 e_4 \cdots e_T}{e_T} \right)} \quad (2.108)$$

$$\varepsilon_{min} = \frac{2^{(T-1)}}{\left(\frac{1}{e_1} + \frac{1}{e_2} + \frac{2^1}{e_3} + \frac{2^2}{e_4} + \cdots + \frac{2^{(T-2)}}{e_T} \right)} \quad (2.109)$$

$$\varepsilon_{min} = \frac{2^{(T-1)}}{\left(\frac{1}{e_1} + \cdots + \frac{2^{(T-2)}}{e_T} \right)} \quad (2.110)$$

$$\varepsilon_{min} = 2^{(T-1)} \left(\frac{1}{\frac{1}{e_1} + \sum_{t=2}^T \frac{2^{(T-2)}}{e_t}} \right), \quad T \geq 2 \quad (2.111)$$

Buna göre çoğunluk oylamalı bir topluluk öğrenci için en kötü durum senaryosuna göre ε_{min} ve ε_{max} değerlerini hesaplayalım. En kötü durum senaryosuna göre $e = [c, c, c, \dots, p]$ ve $c > p$ olsun. Bu senaryoda temel öğrencilerin en kötü tahmini c olarak belirlenmiştir. Çünkü rastgele bir tahmin ediciden daha iyi bir tahmin edici çoğunluk tahmin edicisidir. Çoğunluk tahmin edicisi en fazla sayıya sahip sınıfı tüm tahminlerinde kullanır. Buna göre en kötü senaryoya göre (2.112) de ε_{max} ve (2.113) de ise ε_{min} gösterilmektedir.

$$\varepsilon_{max} = \begin{cases} c, & \text{Eğer } T \text{ çift ise} \\ c - \frac{c+p}{2^{(T-1)}} & \text{Eğer } T \text{ tek ise} \end{cases} \quad (2.112)$$

$$\varepsilon_{min} = \begin{cases} \frac{4cp}{3c+p} & \text{Eğer } T \text{ çift ise} \\ \frac{2cp}{c+p} & \text{Eğer } T \text{ tek ise} \end{cases} \quad (2.113)$$

(2.112) deki $c - \frac{c+p}{2^{(T-1)}}$ ifadesi için temel öğrenci sayısı olan T değerini ortadan kaldıracak şekilde limit alırsak (2.114) deki ifadeyi elde ederiz. Böylelikle (2.112) ifadesi, (2.115) deki ifade gibi yazılabilir.

$$\lim_{T \rightarrow \infty} c - \frac{c + p}{2^{(T-1)}} = c \quad (2.114)$$

$$\varepsilon_{max} \approx c, \text{ Eğer } T \text{ çok büyük bir deęer ise} \quad (2.115)$$

2.3.5 NFLT aısından VIBES algoritmasının teorik olarak deęerlendirilmesi

NFLT'ye iliřkin ierimler henüz ortaya konulmadan ok nce bu teoriye ait ilk izlere İsko filozof David Hume'da rastlanılmaktadır. David Hume 1739 yılında yayınladıęı eseri "A Treatise of Human Nature"da (Hume 1739), "Nesnelerin sık sık veya srekli birleřiminin gzlemlenmesinden sonra bile, deneyimlediğimiz nesnelerin tesinde herhangi bir nesne hakkında herhangi bir ıkarım yapmak iin hibir sebebimiz olmadıęını" belirtmiřtir. İnsan ęrenmesine dair bu bakıř aısından ok sonra; Tom Mitchell (Mitchell 1980), Schaffer (Schaffer 1994) ve Kohavi-Wolpert (Kohavi & Wolpert 1996) gittike artan bir titizlikle biassız ęrenmenin faydasız olduęunu yapmıř oldukları alıřmalarla gstermiřlerdir.

David Wolpert, "The Lack of A Priori Distinctions Between Learning Algorithms" (Wolpert 1996) isimli alıřmasında ęrenme algoritmaları arasında bir ayırmda bulunmak iin yeterli nsel (a priori) bilgiye sahip olmadıęımızı gstermiřtir. Daha sonra David Wolpert, William Macready ile birlikte yapmıř oldukları "No Free Lunch Theorems for Optimization" (Wolpert & Macready 1997) isimli alıřma ile; bu teoriyi 'arama ve optimizasyon' problemlerine de uygulamıřlardır.

Bylelikle birok bilimsel alıřma alanında olduęu gibi (Örneęin matematik alanında Gdel kanıtlanması (Gdel 1931), herhangi bir aritmetik dizgesinde aksiyomlardan yola ıkararak kanıtlanamayan doęruların var olduęunu ortaya koyarken, ekonomi alanında Arrow'un imkansızlık teoremi (Arrow 1950) mkemmел bir oylama sisteminin olmadıęını belirtmiřtir) Makine ęrenmesi alanında da bir eřit imknsızlık teorisi ortaya ıkmıřtır. İmkansızlık teorileri bir dizgenin sınırlarını izerek dizgelerin evrensellik iddialarını ortadan kaldırmaktadır. Bu aıdan bakacak olursak; Gdel kanıtlanması matematik'in saęlam temellere dayanmadıęını deęil, matematik'in saęlam temellere dayandırmak iin bir dizge oluřturulmasının mmkn olmayacaęını gstermiřtir. Makine ęrenmesi alanında ise NFLT; herhangi bir gerek dnya problemi zerinde herhangi bir ęrenme algoritmasını tercih etmemiz iin nsel bir bilgiye sahip olmadıęımızı sylemektedir. Yani NFLT, tm ęrenme algoritmalarının performanslarının nsel ayrımların eksikliklerinden dolayı

birbirleriyle aynı olduğunu ve bu nedenle evrensel en iyi öğrenme algoritmasının olamayacağını belirtmektedir. Ancak, sadece herhangi bir veri kümesi üzerinde sadece bazı öğrenme algoritmaları en iyi performansı gösterecektir.

NFLT'nin öğrenme algoritmaları için çizdiği sınırı matematiksel bir notasyonla ifade edelim. Öncelikle (x_i, y_i) ayrık değerli ikililerinden oluşan ve $i = 1, \dots, m$ ve $y_i = \pm 1$ şeklinde tanımlı \mathcal{D} eğitim kümesine sahip olduğumuzu düşünelim. Ayrıca verilerimizin bilinmeyen bir $f(x)$ fonksiyonundan oluştuğunu ve sıfır olmayan Bayes hata oranına (EK A.1'e bakınız) sahip olduğumuzu varsayalım. Buna göre algoritmanın yapmış olduğu hatayı hesaplayalım. Ancak $f(x)$ fonksiyonumuzu bilemeyeceğimiz için hatayı da doğrudan hesaplayamayız. Bu nedenle \mathcal{D} eğitim kümesi için hatanın beklenen değerini hesaplamalıyız. (2.116)'da hatanın beklenen değeri gösterilmektedir (Wolpert 1996). Bu ifade de $P(I)$ tüm olası girişlerin olasılığını ifade etmektedir.

$$E[e|\mathcal{D}] = \sum_{h,f} \sum_{I \notin \mathcal{D}} P(I)[1 - \delta(f(I), h(I))] P(h|\mathcal{D})P(f|\mathcal{D}) \quad (2.116)$$

(2.116)'daki ifadeden yola çıkarak örnekleme dağılımı $P(I)$ 'dan bağımsız olarak ve eğitim kümesinin örnek sayıları eşit olacak şekilde; $P_k(h|\mathcal{D})$ şeklinde tanımlı k adet öğrenme algoritmasının tüm $f(\cdot)$ hedef fonksiyonları üzerindeki beklenen hata değerleri arasındaki ilişki $\forall k, E_k[e|f, m] = c_1$ 'dir ve rastgele tahmin ediciden daha iyi en az bir hedef fonksiyonu vardır. Ayrıca sabit herhangi bir \mathcal{D} eğitim kümesi için tüm $f(\cdot)$ hedef fonksiyonları üzerindeki beklenen hata değerleri arasındaki ilişki $\forall k, E_k[e|f, \mathcal{D}] = c_2$ 'dir. Bunun yanı sıra tüm $P(f)$ öncülleri üzerindeki beklenen hata değerleri arasındaki ilişki $\forall k, E_k[e|m] = c_3$ 'tür. Son olarak sabit herhangi bir \mathcal{D} eğitim kümesi için tüm $P(f)$ öncülleri üzerindeki beklenen hata değerleri arasındaki ilişki $\forall k, E_k[e|\mathcal{D}] = c_4$ 'tür.

Herhangi bir öğrenme algoritması için verilen bir veri kümesi için hipotez oluşturma olasılığı $P(h|\mathcal{D})$ 'dir. VIBES algoritması için oluşturulan hipotezlerin olasılıkları sırasıyla $P(h_1|d_1), P(h_2|d_2), \dots, P(h_n|d_n)$ şeklindedir. Bu olasılıklar içerisinde $P(h_n|d_n)$ olasılığı $P(h|\mathcal{D})$ olasılığı ile aynıdır. Yani VIBES algoritmasının her zaman en az bir hipotezi, seçilen temel öğrencinin oluşturduğu hipotez ile aynıdır. Çünkü her zaman asıl veri kümesi her iki algoritma için de ortaktır. Buna göre

(2.111)'deki ifadeden faydalanarak bir temel öğrenici/topluluk öğrenici ile VIBES algoritması arasındaki hatanın beklenen değeri ilişkisi (2.117)'de gösterilmektedir.

$$E[e|\mathcal{D}] \geq 2^{(T-1)} \left(\frac{1}{\frac{1}{E[e|\mathcal{D}]} + \sum_{t=2}^T \frac{2^{(T-2)}}{E[e|d_t]}} \right), \quad T \geq 2 \quad (2.117)$$

(2.117)'deki ifadeye göre VIBES algoritması NFLT açısından bir başka öğrenme algoritması ile kıyaslandığında beklenen hata değeri en kötü durumda bu öğrenme algoritmasına eşit olacaktır.

2.4 VIBES Algoritmasında Özniteliklerin Enformasyon Ölçümü İçin Kullanılan Yöntemler

Bu tez çalışmasında özniteliklerin enformasyon ölçümünde iki yöntem kullanıldı. Bu yöntemlere aşağıda değinilmektedir.

2.4.1 ReliefF algoritması

ReliefF algoritması (Kononenko et al. 1997) Relief algoritmasının (Kira & Rendell 1992) geliştirilmiş bir halidir. ReliefF algoritması örnek-tabanlı öğrenme yaklaşımından esinlenerek geliştirilmiş bir öznitelik seçim algoritmasıdır. Relief algoritması; veri kümesinden rastgele aldığı bir örneğin, veri kümesindeki her bir sınıfa ait en yakın örnekler içerisinde rastgele seçtiği bir örneğe olan uzaklığına göre bir öznitelik ağırlık ölçümü yapar. Relief algoritmasına ait pseudo kod Şekil 2.10'da gösterilmektedir. Şekil 2.10'daki pseudo kodda yer alan $\text{diff}(\cdot)$ fonksiyonu uzaklık ölçüm fonksiyonudur. Nominal değerler için $\text{diff}(\cdot)$ fonksiyonu (2.118)'deki gibi tanımlanmaktadır. Nümerik değerler için ise $\text{diff}(\cdot)$ fonksiyonu (2.119)'deki gibi tanımlanmaktadır. (2.119)'deki ifadede yer alan nu_k ifadesi normalizasyon birimidir.

$$\text{diff}(x_k, y_k) = \begin{cases} 0, & x_k \text{ ve } y_k \text{ aynı ise} \\ 1, & x_k \text{ ve } y_k \text{ farklı ise} \end{cases} \quad (2.118)$$

$$\text{diff}(x_k, y_k) = (x_k - y_k) / nu_k \quad (2.119)$$

Algoritma: *Relief*

Giriş: S , eğitim kümesi.

m , veri sayısı.

Çıkış: w , özniteliklerin ağırlıkları

Metot:

$w \leftarrow (0, 0, \dots, 0)$

for $i \leftarrow 1$ to m

Rastgele bir $x \in S$ örneğini al;

x 'e en yakın pozitif örneklerden rastgele birini al, $Z^+ \in S^+$

x 'e en yakın negatif örneklerden rastgele birini al, $Z^- \in S^-$

if $x \in S^+$ **then**

Near-hit $\leftarrow Z^+$

Near-miss $\leftarrow Z^-$

else

Near-hit $\leftarrow Z^+$

Near-miss $\leftarrow Z^-$

for $j \leftarrow 1$ to #all_features

$w_j \leftarrow w_j - \text{diff}(x_j, \text{Near-hit}_j)^2 + \text{diff}(x_j, \text{Near-miss}_j)^2$

endfor

endfor

Şekil 2.10: Relief algoritması (Kira & Rendell 1992)

Relief algoritması iki eksikliği içermektedir. Bu eksiklikler: Relief algoritmasının eksik verilerle ilgili yetersizliği ve sadece iki sınıflı problemlere uygulanmasıdır (Kononenko et al. 1997). Bu eksiklikleri gidermek için Relief algoritmasının bir uzatımı olan ReliefF algoritması geliştirilmiştir.

2.4.2 Shannon enformasyon kuramı

Entropi kelimesi ilk olarak Rudolf Clausius tarafından kullanılmıştır (Baray 2003). Clausius, kapalı bir sistemdeki enerji miktarının ölçüsünü bu sisteme giren ısının ve sistemin sıcaklığının bir fonksiyonu olarak tanımlamış ve buna da Almandadaki

“energie” kelimesine benzediğinden dolayı Yunancada “dönüşüm” anlamına gelen “entropi” adını vermiştir (Fang et al. 1997).

Enformasyon Kuramı, Claude Shannon tarafından geliştirilmiş bir kuramdır. Bu kuram ile “iletişim”, matematiksel bir temele oturtulmuş ve iletişimin teorik sınırları çizilmiştir (MacKay 2003). Shannon (Shannon 1948) bu kuramını “A Mathematical Theory of Communication” isimli çalışmasında naif bir biçimde açıklamaktadır. Shannon, John von Neumann’ın da desteği ile “entropi” kelimesini bilgedeki belirsizliğin ölçüsünü göstermede kullanmıştır (Fang et al. 1997). Entropi kavramı günümüzde istatistiksel mekanik, termodinamik, finans, parametre tahmini ve Makine Öğrenmesi gibi birçok alanda kullanılmaktadır. Makine Öğrenmesi alanında özellikle özniteliklerin enformasyon miktarını hesaplamada kullanılmaktadır.

Entropi kavramı, eldeki bilginin sayısallaştırılması ve beklentisizliğin maksimumlaşmasıdır. Dunham (Dunham 2002), entropinin bir veri kümesi içindeki belirsizlik, şaşkınlık ve rastgeleliği ölçmek için kullanıldığını belirtmiştir. Eldeki bütün veriler tek bir sınıfa ait olsaydı, örneğin herkes aynı futbol takımını tutsaydı, herhangi bir kişiye tuttuğu takımı sorduğumuzda alacağımız yanıt bizi şaşırtmayacaktır; bu durumda entropi sıfır olacaktır.

Buna göre bir eğitim kümesindeki özniteliklerin enformasyon miktarını şu şekilde hesaplayabiliriz: Eğitim kümesi çıkışlarının y_1, y_2, \dots, y_n olduğu ve bu çıkışlara karşılık mevcut olasılıkların $P(y_1), P(y_2), \dots, P(y_n)$ olduğu durumlarda enformasyon kazancı (Information Gain) (2.120)’deki formüle göre özniteliklerin enformasyonunu hesaplamaktadır. Bu ölçüm bize; eğitim kümesinin girişlerini oluşturan öznitelikler tarafından sağlanan bilginin beklenen miktarını vermektedir (Shannon 1948).

$$I(P(y_1), P(y_2), \dots, P(y_n)) = - \sum_{i=1}^n P(y_i) \log_2 P(y_i) \quad (2.120)$$

Y sınıf niteliğine göre verilen bir X niteliğinin bilgi kazancı, X’in değeri bilindiğinde yani $I(Y; X)$ bilindiğinde Y’nin değeri hakkındaki belirsizliği azaltır. Y’nin değeri hakkındaki belirsizlik onun entropisi ile yani $H(Y)$ ile ölçülür. X’in değerini bildiğimizde Y’nin değeri hakkındaki belirsizlik Y’nin koşullu entropisi tarafından verilir. Yani $H(Y|X)$ tarafından verilir. Bu durumda bilgi kazancı formülü (2.121)’deki gibi hesaplanır (Shannon 1948).

$$I(Y; X) = H(Y) - H(Y|X) \quad (2.121)$$

$Y=\{y_1, \dots, y_n\}$ ve $X=\{x_1, \dots, x_n\}$ değerlerini alan ayrık değişkenler olsun. Bu durumda Y 'nin entropisi ve Y 'nin koşullu entropisi sırasıyla (2.122) ve (2.123)'deki gibi hesaplanır (Shannon 1948).

$$H(Y) = - \sum_{i=1}^k P(Y = y_i) \log_2 P(Y = y_i) \quad (2.122)$$

$$H(Y|X) = - \sum_{j=1}^l P(X = x_j) H(Y|X = x_j) \quad (2.123)$$

Bilgi kazancı formülünün alternatif formülü ise (2.124)'de gösterilmektedir. Bu durumda X ve Y 'nin ortak entropisini hesaplamamız gerekmektedir. Buna göre ortak entropi (2.125)'daki gibi hesaplanır (Shannon 1948).

$$I(Y; X) = H(X) + H(Y) - H(X, Y) \quad (2.124)$$

$$H(X, Y) = - \sum_{i,j} P(X = x_i, Y = y_j) \log_2 P(X = x_i, Y = y_j) \quad (2.125)$$

2.5 VIBES Algoritmasında Kullanılan Bazı Temel Öğreniciler

Bu tez çalışmasında VIBES algoritmasının temel öğrenicileri olarak altı temel öğrenici kullanıldı. Bu temel öğrenicilere aşağıda değinilmektedir.

2.5.1 k-En yakın komşu algoritması

M -boyutlu bir uzayda tanımlı olan bir veri kümesi ve bir sorgu noktası verilmiş olsun. Buna göre bu sorgu noktasına en yakın veriyi bulma adımı en yakın komşuluk problemi olarak tanımlanır (Beyer et al. 1999). Ancak, k -En Yakın Komşuluk (k -EYK) algoritması ise bir sorgu noktasının bir veri kümesi içerisindeki en yakın k tane veriye olan uzaklığını esas almaktadır. Uzaklıkların hesaplanmasında (2.126), (2.127), (2.128) ve (2.129)'de gösterildiği gibi sırasıyla Euclidean, Manhattan, Chebyshev ve Minkowski uzaklık formülleri (Deza & Deza 2009) kullanılmaktadır (Witten et al. 2011). Minkowski uzaklık formülünde $m=1$ ve $m=2$ için hesaplanan

uzaklıklar sırasıyla Manhattan ve Euclidean uzaklığı vermektedir. p ve q veri noktaları olmak üzere $d(p,q)$ iki nokta arasındaki uzaklık olarak tanımlansın.

$$d_{Euclidean}(p, q) = \sqrt{\sum_{i=1}^n (p_i - q_i)^2} \quad (2.126)$$

$$d_{Manhattan}(p, q) = \sum_{i=1}^n |p_i - q_i| \quad (2.127)$$

$$d_{Chebyshev}(p, q) = \max_i (|p_i - q_i|) \quad (2.128)$$

$$d_{Minkowski}(p, q) = \left(\sum_{i=1}^n |p_i - q_i|^m \right)^{1/m} \quad (2.129)$$

k-EYK algoritmasında k değerinin optimal değerden yüksek olması birbirine benzemeyen noktaların bir araya toplanmasına, optimal değerden küçük seçilmesiyle birbirine benzeyen sınıfların ayrı sınıflara konmasına neden olmaktadır. k-EYK algoritması için en uygun k değerinin belirlenmesi önemli bir adımdır; çünkü k değerine göre sorgu noktasının hangi sınıfa atanacağı belirlenmektedir.

k-EYK sınıflandırıcı yapılan k gözlem içinden en fazla tekrar eden sınıfa göre karar vermektedir. Yalnız bu karar sınıfı, sadece k komşuya göre alınırsa her zaman iyi performans vermeyebilir. Bu durumda k komşu arasında en çok tekrar eden sınıfın seçilmesi yerine ağırlıklı oylama adlı bir yöntemin uygulanması söz konusu olabilir. Ağırlıklı oylama yöntemi gözlenen değerler için ağırlıklı uzaklıkların hesaplanmasına göre çalışır. Ağırlıklı oylama yöntemi için iki farklı ağırlık hesaplama yöntemi vardır. Bunlar (2.130) ve (2.131) eşitliğinde gösterilmektedir (Witten et al. 2011). (2.130) eşitliği ağırlıklı inverse olarak bilinir. (2.131) eşitliği ise ağırlıklı benzerlik olarak bilinir. Bu eşitliklerde yer alan N ; eğitim verisini, n ise; eğitim kümesinde yer alan nitelik sayısını göstermektedir.

$$weight_{inverse} = \sum_{i=1}^N \frac{1}{\sqrt{\frac{d(p, q)^2}{n}}} \quad (2.130)$$

$$weight_{similarity} = \sum_{i=1}^N \left(1 - \sqrt{\frac{d(p,q)^2}{n}} \right) \quad (2.131)$$

(2.130) ve (2.131)'de yer alan $d(p,q)$ ifadesi p ve q noktaları arasındaki uzaklığı ifade etmektedir. Her bir sınıf için ağırlıklar hesaplanarak ağırlıklı oylama değeri belirlenir. En büyük değere sahip olan sınıf, yeni gözlemin sınıfı olarak kabul edilir.

k-EYK gibi doğrusal olmayan algoritmalar doğaları gereği düşük biaslı ve yüksek varyanslıdır (Manning & Raghavan 2009). Bir öğrenme algoritmasının bias'ı, algoritmanın yapacağı tahminler ile verinin gerçek değeri arasındaki fark olarak bilinmektedir. Bu farkın çok büyük olması algoritmanın düşük isabetli tahminler yaptığını göstermektedir. Bir öğrenme algoritmasının varyansı ise algoritmanın yapmış olduğu tahminler arasındaki değişimin ortalamadan ne kadar saptığı olarak bilinmektedir. Diğer bir deyişle bir sorgu noktası için gerçekleştirilen tahminlerin ortalamadan ne kadar saptığıdır. Sonuç olarak hem bias hem de varyans öğrenme algoritmaları için tahmin hatalarına sebep olmaktadır.

2.5.2 Naive bayes algoritması

Naive Bayes (NB) sınıflandırıcı, güçlü bağımsız varsayımlarıyla Bayes'in teoremine dayalı basit bir olasılık sınıflandırıcıdır. Bir NB sınıflandırıcı, bir sınıfın bir özelliğinin herhangi bir diğer özelliği ile ilişkili olmaması esasına dayanır. Yani NB sınıflandırıcı tüm özelliklerin birbirlerinden bağımsız olduğunu varsayar. NB sınıflandırıcı birçok kompleks gerçek dünya problemlerinde oldukça iyi sonuç vermektedir. 2004'te, Bayes sınıflandırma problemlerinin analizi NB sınıflandırıcıların nedensiz görünen etkileri için bazı teorik nedenlerin var olduğunu gösterdi (Zhang 2004). NB sınıflandırıcının bir avantajı da sınıflandırma için gerekli olan parametreleri tahmin etmek için sadece küçük miktarda eğitim verisinin yeterli olmasıdır.

NB sınıflandırıcılar makine öğrenmesi alanındaki önemli araştırma sınırlarını gösterirler. Bunun amacı sadece gerçek dünya verisi üzerinde iyi çalışan öğrenme metotlarını keşfetmek değil aynı zamanda açık anlamları da keşfetmektir. NB sınıflandırıcılar olasılıksal bilgiyi öğrenme, kullanma ve göstermede açık anlamlara sahip basit bir yaklaşım sağlarlar. NB metodu gözetimli tümevarımsal görevlerde

kullanılmak için tasarlanılır. Bu metodun performans amacı test örneklerinin sınıfını doğru bir biçimde tahmin etmektir (John & Langley 1995).

Bir NB sınıflandırıcı için olasılık model bir durumsal modeldir. Bu olasılık model için F_1, F_2, \dots, F_n özellik değişkenleri ve sınıfın ya da çıkışın bağımlı değişkeni C olmak üzere durumsal model Bayes teoremi kullanılarak (2.132)'de gösterilmektedir (Witten et al. 2011).

$$p(C|F_1, \dots, F_n) = \frac{p(C)p(F_1, \dots, F_n|C)}{p(F_1, \dots, F_n)} \quad (2.132)$$

Özelliklerin ayrık değerlere değil de sürekli değerlere sahip olduğu durumlarda, x sürekli nitelik olmak üzere; ilk olarak veri sınıflara göre ayrılır. Sonra herbir sınıftaki x 'in ortalaması ve varyansı hesaplanır. Son olarak verilen $C=c$ sınıfına ait değerlerin olasılığı bir normal dağılım için $X=x$ için (2.133)'daki formülle hesaplanır (Witten et al. 2011).

$$p(X = x|C = c) = \frac{1}{\sqrt{2\pi\sigma_c^2}} e^{-\frac{(x-\mu_c)^2}{2\sigma_c^2}} \quad (2.133)$$

2.5.3 Destek yöney makinası

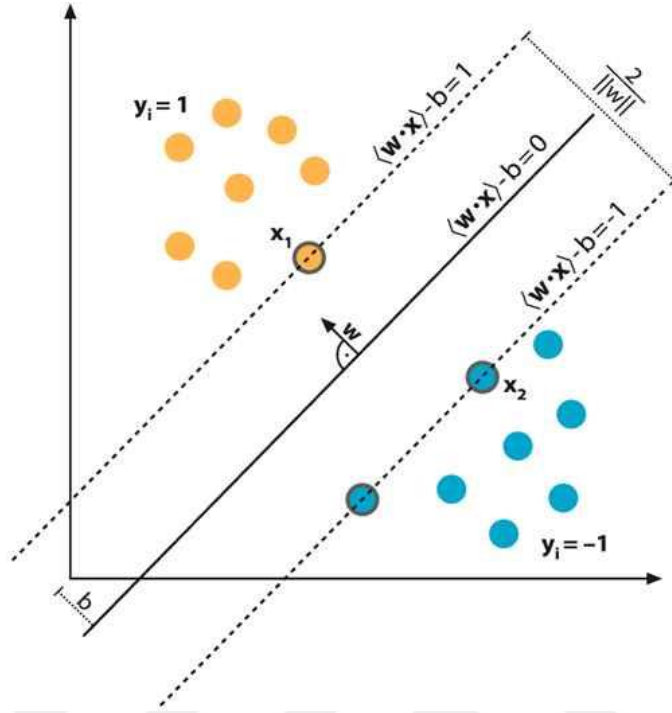
Bu yöntem düşük boyutta doğrusal olarak ayırlamayacak bir veri kümesini, çoğu zaman daha yüksek boyuta taşıyarak bir düzlem yardımıyla ayırmaya dayanır. Eğer veri kümesi doğrusal olarak ayrılabilirse boyut değişimine gerek duyulmaz. Destek Yöney Makinesi (DYM), sonsuz sayıdaki düzlemsel ayırıcılardan, birbirinden ayrılacak sınıf örneklerinden maksimum uzaklıkta yer alanı seçer.

İki sınıflı bir sınıflandırma probleminde $-1/+1$ bu iki sınıfın etiketlerini gösterebilir. $X=\{x^t, r^t\}$ örneğinde $x^t \in C_1$ ise $r^t = +1$ ve $x^t \in C_2$ ise $r^t = -1$ değerini alsın. w ve w_0 parametrelerinin saptanması gereken koşullar (2.134)'daki gibi tanımlanırsa ve

$$\begin{aligned} w^T x^t + w_0 &\geq +1, & \text{Eğer } r^t &= +1 \\ w^T x^t + w_0 &\leq -1, & \text{Eğer } r^t &= -1 \end{aligned} \quad (2.134)$$

bu koşulları tek satırda özetlersek $r^t(w^T x^t + w_0) \geq +1$ ifadesi elde edilir. Koşulun $r^t(w^T x^t + w_0) \geq 0$ olmaması önemlidir. Çünkü örneklerin üstdüzlemin yalnızca doğru yanında değil daha iyi genelleme için ona belli bir uzaklıkta kalması gerekmektedir. Sınırın her iki yanında en yakın örneklerin sınıra uzaklığına kenar

payı denir ve iyi bir genelleme için bu kenar payını olabildiğince büyük yapmak gerekmektedir (Alpaydin 2014).



Şekil 2.11: Doğrusal olarak birbirinden ayrılabilen veriler arasındaki hiperdüzlem (DFG Research Training Group n.d.)

x^t örneğinin destek vektöre uzaklığının formülü (2.135)'de yer almaktadır (Alpaydin 2014).

$$\frac{|w^T x^t + w_0|}{\|w\|} \quad (2.135)$$

$r^t \in \{-1, +1\}$ olduğunda bunu her iki yandaki örnekler için yazarsak (2.136) ifadesi elde edilir (Alpaydin 2014).

$$\frac{r^t(w^T x^t + w_0)}{\|w\|} \quad (2.136)$$

Bu değer en az p gibi bir değer olmasını istediğimizi düşünürsek (2.137) ifadesi elde edilir (Alpaydin 2014).

$$\frac{r^t(w^T x^t + w_0)}{\|w\|} \geq p, \quad \forall t \quad (2.137)$$

Buna göre p değerini olabildiğince büyütmek isteriz. Çünkü destek vektörlerin birbirlerinden olabildiğince uzak olmasını isteriz. Ancak bunu w vektörünü

ölçekleyerek yapmak istenen çözüm olmaz. Tek çözüm olarak $p\|w\| = 1$ ifadesini sabitleyip kenar payını olabildiğince büyütmek için $\|w\|$ değerini yani vektörün boyunu olabildiğince küçültürüz (Cortes & Vapnik 1995; Vapnik 2000). Bu durumda maksimum boşluğu bulmaya yarayan problem quadratik programlama modeline dönüşür (Cristianini & Shawe-Taylor 2000). Buna göre minimum olmasını istediğimiz vektör boyu ve kısıtları (2.138)'teki gibi belirtebiliriz. Bu ifadede 1/2 ifadesi hesaplamayı kolaylaştırmak için eklenmiştir (Alpaydin 2014).

$$\min \frac{w^T w}{2} \quad \text{kısıt: } r^t(w^T x^t + w_0) \geq +1, \quad \forall t \quad (2.138)$$

(2.138)'teki problem doğrusal olmayan bir optimizasyon problemidir. Bu tür problemleri çözmek için Lagrange fonksiyonundan faydalanılır. α^t Lagrange çarpanları kullanılarak kısıtlanmamış biçimde (2.139)'deki gibi yazılır. Bu ifade düzenlenirse (2.140)'daki ifadeyi elde ederiz (Alpaydin 2014).

$$L_p = \frac{1}{2}(w^T w) - \sum_{t=1}^N \alpha^t [r^t(w^T x^t + w_0) - 1] \quad (2.139)$$

$$L_p = \frac{1}{2}(w^T w) - \sum_{t=1}^N \alpha^t r^t (w^T x^t + w_0) + \sum_t \alpha^t \quad (2.140)$$

Bu işlevi w , w_0 değerlerine göre en küçük ve $\alpha^t \geq 0$ değişkenlerine göre en büyük noktayı arıyoruz. Bu dışbükey, ikinci dereceden bir iyileme problemidir. Çünkü hem ana terim hem de doğrusal kısıtlar dışbükeydir. Bu yüzden yerine Karush-Kuhn-Tucker koşulları kullanılarak eşlek problem çözülebilir. Eşlek, L_p problemini α^t değişkenlerine göre büyütmektir ve kısıtlar, L_p işlevinn w ve w_0 parametrelerine göre eğiminin 0 ve $\alpha^t \geq 0$ olmasıdır. Bu kısıtlar (2.141) ve (2.142)'de gösterilmektedir (Alpaydin 2014).

$$\frac{\partial L_p}{\partial w} = 0 \implies w = \sum_t \alpha^t r^t x^t \quad (2.141)$$

$$\frac{\partial L_p}{\partial w_0} = 0 \implies \sum_t \alpha^t r^t = 0 \quad (2.142)$$

Bu kısıtları (2.140) ifadesinde yerine koyarsak (2.143)'daki ifadeyi elde ederiz (Alpaydin 2014).

$$\begin{aligned}
L_d &= \frac{1}{2}(w^T w) - w^T \sum_t \alpha^t r^t x^t - w_0 \sum_t \alpha^t r^t + \sum_t \alpha^t \\
&= -\frac{1}{2}(w^T w) + \sum_t \alpha^t \\
&= -\frac{1}{2} \sum_t \sum_s \alpha^t \alpha^s r^t r^s (x^t)^T x^s + \sum_t \alpha^t
\end{aligned} \tag{2.143}$$

Bu işlevi α^t değişkenlerine göre büyütürken kısıtları yazarsak (2.144) ifadesi elde edilir (Alpaydin 2014).

$$\sum_t \alpha^t r^t = 0 \text{ ve } \alpha^t \geq 0, \forall t \tag{2.144}$$

Bu ifade ikinci dereceden optimizasyon problemidir. Bu durumda dual model matris formunda (2.145)'deki gibi ifade edilir (Alpaydin 2014).

$$\max \alpha^T I - \frac{1}{2} \alpha^T H \alpha \text{ ve kısıtlar: } \alpha^t \geq 0 \tag{2.145}$$

Bu ifade de H , Hessian matrisini göstermektedir. I ise birim matristir ve $I = [1 \ 1 \ \dots \ 1]$ şeklinde gösterilir. H matrisi (2.146)'deki gibi ifade edilir (Alpaydin 2014).

$$H = r^t r^s (x^t x^s) = r^t r^s (x^r)^T x^s \tag{2.146}$$

Bu optimizasyon probleminin (w^* , b^* , α^*) optimal noktası için $(\alpha^t)^*$ çözümleri yani Lagrange çarpanları (2.141) ve (2.142)'de belirtildiği gibi optimal hiper düzlemin (2.147)'de gösterildiği gibi w^* ve (2.148)'te gösterildiği gibi b^* parametrelerini belirler. Söz konusu parametreler destek vektörleri ve bu destek vektörlerin sayısını göstermek üzere aşağıdaki şekilde yazılabilir. Burada N ifadesi destek vektör sayısıdır (Alpaydin 2014).

$$w^* = \sum_t (\alpha^t)^* r^t x^t \tag{2.147}$$

$$b^* = \frac{1}{N} \sum_t \frac{1}{r^t} - (x^t)^T w^* \tag{2.148}$$

(2.148) ifadesinde görüldüğü gibi b^* in hesaplanmasında sadece destek vektörler kullanılmaktadır. Çünkü destek vektör olmayan tüm Lagrange çarpanları sıfıra

eşittir (Özkan 2016). Bu durumda yeni bir örneğe göre sınıflandırma için (2.149)'teki fonksiyon kullanılabilir.

$$f(x) = \text{sign} \left(\sum_t (\alpha^t)^* r^t ((x^t)^T x) b^* \right) \quad (2.149)$$

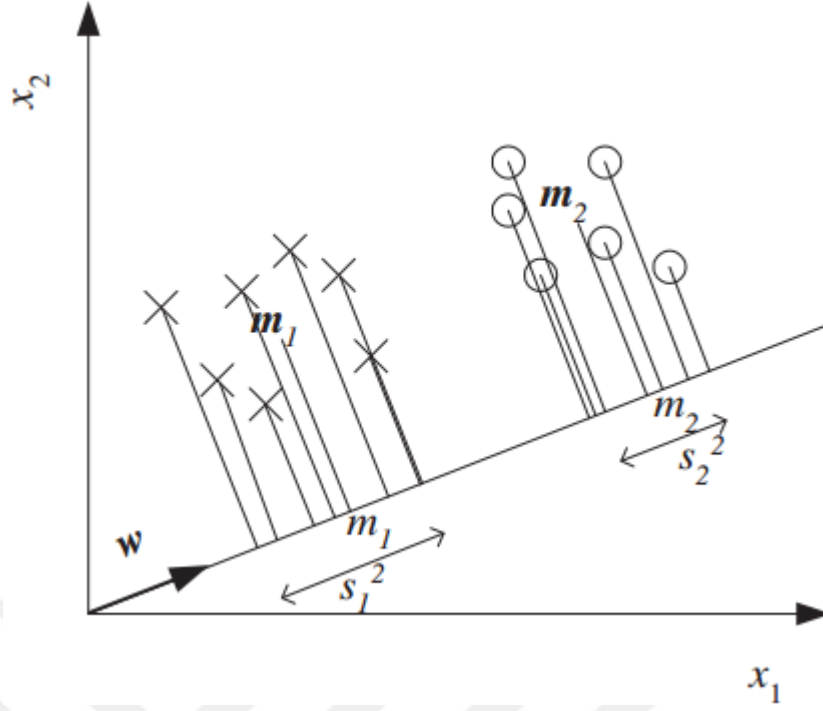
(2.149) ifadesi yerine sadece destek vektörler gözönüne alınacak olursa yeni ifademiz (2.150)'daki gibi olur.

$$f(x) = \text{sign}(w^* x + b^*) \quad (2.150)$$

2.5.4 Doğrusal diskriminant analiz

Doğrusal diskriminant analiz (DDA) sınıflandırma problemlerinde boyut azaltmak için kullanılan gözetimli bir yöntemdir. Şekil 2.12'de gösterildiği gibi iki sınıflı (C_1 ve C_2) bir sınıflandırma problemi için DDA şu şekilde çalışır:

Öncelikle verinin izdüşümünün üzerine düştüğü w vektörü tanımlanır. Böylelikle x 'in w üzerine düştüğü izdüşümü $z = w^T x$ şeklinde hesaplanır. Böylelikle boyut azaltımı gerçekleştirilmiş olur. m_1 ve m_2 , sırasıyla izdüşümden önce ve sonra C_1 'den örneklerin ortalamasıdır. Ayrıca $m_1 \in \mathcal{R}^d$ ve $m_2 \in \mathcal{R}$ olarak tanımlıdır. m_1 ve m_2 ile m_1 ve m_2 arasındaki ilişki sırasıyla (2.151) ve (2.152)'de gösterilmektedir (Alpaydın 2014). $x^t \in C_1$ ise $r^t = 1$ ve $x^t \in C_2$ ise $r^t = 0$ şeklinde tanımlı bir $X = \{x^t, r^t\}$ veri kümesi için,



Şekil 2.12: w vektörü üzerine 2-boyutlu ve iki sınıflı verinin izdüşümü (Alpaydin 2014)

$$m_1 = \frac{\sum_t w^T x^t r^t}{\sum_t r^t} = w^T m_1 \quad (2.151)$$

$$m_2 = \frac{\sum_t w^T x^t (1 - r^t)}{\sum_t (1 - r^t)} = w^T m_2 \quad (2.152)$$

İzdüşümden sonra C_1 ve C_2 'den örneklerin dağılımı (2.153) ve (2.154)'da sırasıyla gösterilmektedir (Alpaydin 2014).

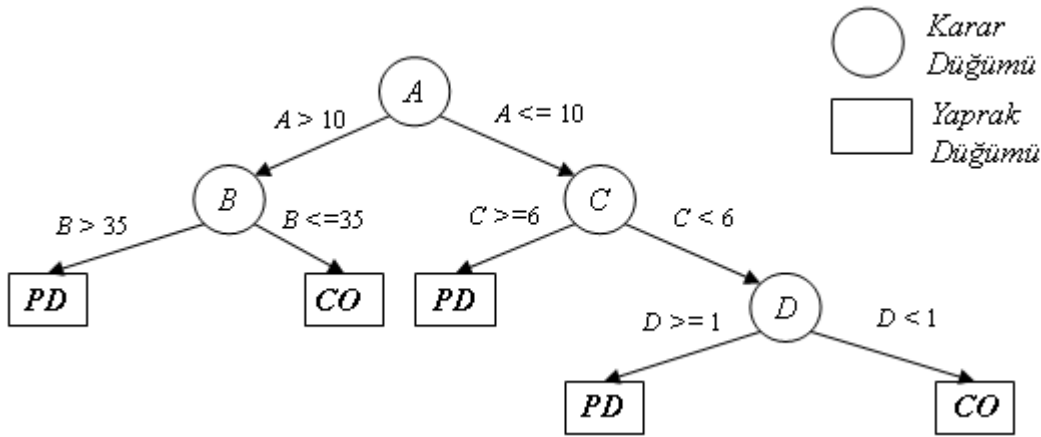
$$s_1^2 = \sum_t (w^T x^t - m_1)^2 r^t \quad (2.153)$$

$$s_2^2 = \sum_t (w^T x^t - m_2)^2 (1 - r^t) \quad (2.154)$$

İzdüşümden sonra iyi bir biçimde birbirinden ayrılmış iki sınıf için ortalamaların mümkün olabildiğince uzak olması ve sınıfların örneklerinin mümkün olabildiğince küçük bir bölgeye yayılması istenir (Alpaydin 2014). Diğer bir ifadeyle $|m_1 - m_2|$ 'nin büyük olması ve $s_1^2 + s_2^2$ 'nin de küçük olması istenir.

2.5.5 Karar ağaçları

Bir karara varırken oluşturulacak karar ağacının kökünden başlayıp yaprak düğümlere doğru hareket ederek örnek üzerinden sınıflama yapan bir algoritmadır. Karar ağaçları basit bir anlayışa dayanmakla birlikte, bazı veriler üzerinde çok başarılı modeller oluşturabilmektedirler. Şekil 2.13’de örnek bir karar ağacı yer almaktadır. Bu karar ağacının en altında karar düğümleri yer almaktadır. Bu örnekte karar ağacı sınıflandırıcısı; *PD* ve *CO* ile etiketlenmiş olan örnekleri sınıflandırmaktadır.



Şekil 2.13: Örnek bir karar ağacı

1970'lerin sonları ve 1980'lerin başında J. Ross Quinlan Makine Öğrenmesi alanında yeni bir algoritma olan ve ID3 (İterative Dichotomiser) olarak bilinen bir karar ağacı algoritması geliştirdi. Bu çalışma; E. B. Hunt, J. Marin ve P. T. Stone tarafından tanımlanan Kavram Öğrenme Sistemleri üzerine yapılan daha önceki çalışmalara genişletildi. Daha sonra Quinlan, yeni geliştirilmiş gözetimli öğrenme algoritmalarının genellikle karşılaştırıldığı bir benchmark haline gelen C4.5'i (ID3'in ardılı) sundu. 1984'te bir grup istatistikçi (L. Breiman, J. Friedman, R. Olshen ve C. Stone), ikili karar ağaçlarının oluşturulmasını tanımlayan Sınıflama ve Regresyon Ağaçlarını (CART) sundular. Böylelikle ID3 ve CART aynı anda birbirinden bağımsız olarak geliştirildi. Ancak bu iki algoritma, eğitim kümelerinden Karar Ağaçlarını öğrenme için benzer bir yaklaşım izledi (Han & Kamber 2006).

ID3, C4.5 ve CART karar ağaçlarının yukarıdan aşağıya yinelemeli bir bölme ve yönet biçiminde oluşturulduğu açgözlü (yani geri izleme olmayan) bir yaklaşımı benimser. Karar ağacı çıkarımı için çoğu algoritma aynı zamanda bir takım eğitim kümesi ve bunlarla ilişkili sınıf etiketleri ile başlayan yukarıdan aşağıya doğru bir

yaklaşımı izler. Ağaç oluşturulurken, eğitim kümesi tekrarlanan bir şekilde küçük alt gruplara bölünür. Temel bir karar ağacı algoritması Şekil 2.14'de özetlenmiştir (Han & Kamber 2006).

Algoritma: *Generate_decision_tree*

Giriş: D , eğitim kümesi.

attribute_list, aday nitelik kümesi.

Attribute_selection_method, ayırma kriterine karar veren metot; bu kriter *splitting_attribute*'dan oluşur.

Çıkış: Karar Ağacı modeli

Metot:

- (1) N düğümünü oluştur;
- (2) **if** D veri kümesindeki verilerin tümü C sınıfı **then**
- (3) **return** C sınıfı olarak etiketlenmiş bir yaprak düğümü olarak N_i döndür;
- (4) **if** *attribute_list* = \emptyset **then**
- (5) **return** D deki çoğunluk sınıfına göre etiketlenmiş bir yaprak düğümü olarak N_i döndür;
- (6) “En iyi” ayırma kriterini bulmak için *Attribute_selection_method*(D , *attribute_list*) metodunu uygula;
- (7) *splitting_criterion* ile N düğümünü etiketle;
- (8) **if** *splitting_attribute* ayırık değerli **then**
- (9) *attribute_list* \leftarrow *attribute_list* - *splitting_attribute*;
- (10) **for each** *splitting_criterion*'ın j . çıkışı
- (11) D_j , j . çıkışı ifade eden D deki veri gruplarının kümesi olsun;
- (12) **if** $D_j = \emptyset$ **then**
- (13) N düğümüne D deki çoğunluk sınıfı olarak etiketlenmiş bir yaprağı ata;
- (14) **else** N düğümüne *Generate_decision_tree*(D_j , *attribute_list*) tarafından döndürülen düğümü ata;
- endfor**
- (15) **return** N ;

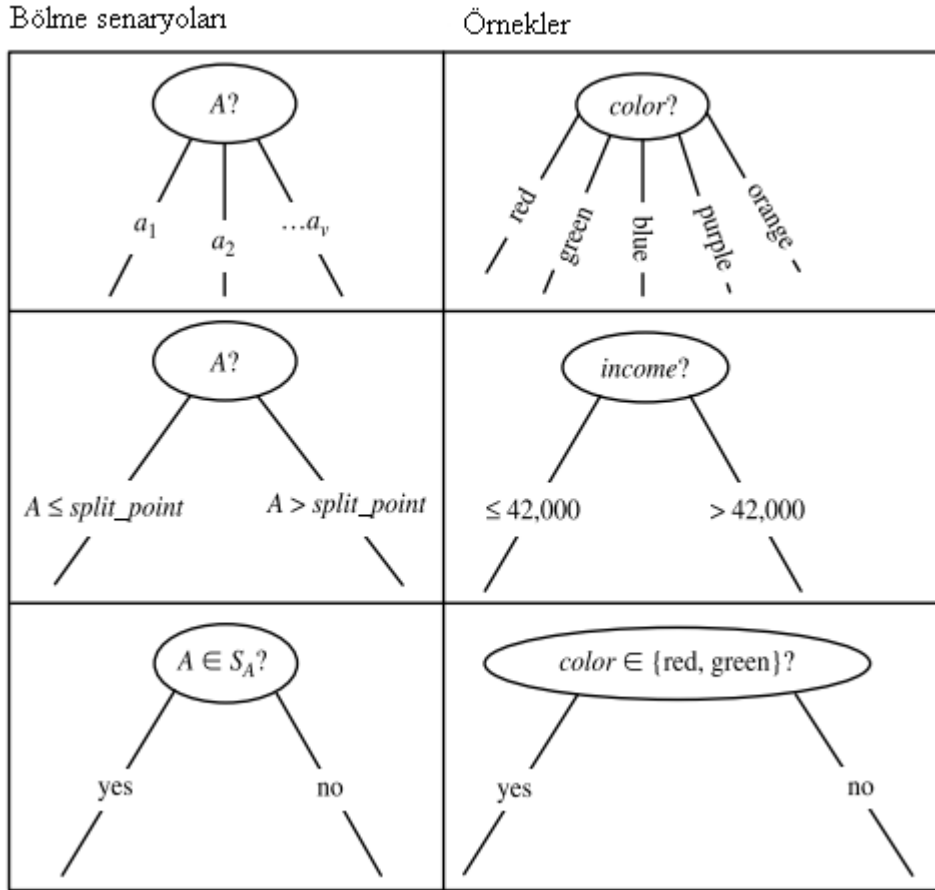
Şekil 2.14: Temel bir karar ağacı algoritması (Han & Kamber 2006)

Şekil 2.14'deki algoritma şu stratejide çalışmaktadır:

- Öncelikle algoritma üç parametreyle çağrılır. Bu parametreler şunlardır: D , $attribute_list$ ve $Attribute_selection_method$. Burada D veri kümesi olarak ifade edilmektedir. $attribute_list$ parametresi, veri kümelerini tanımlayan öz niteliklerin bir listesidir. $Attribute_selection_method$, verilen veri gruplarını sınıflara göre “en iyi” ayıran özelliği seçmek için sezgisel bir prosedürü belirtir. Bu yordam, bilgi kazancı veya Gini dizini gibi bir öznitelik seçimi ölçütü kullanır. Ağacın kesinlikle ikili olup olmadığı genellikle özellik seçimi ölçütü tarafından yönlendirilir. Gini dizini gibi bazı öznitelik seçim ölçümleri, elde edilen ağacın ikili olmasını zorlar. Diğerleri (bilgi kazancı gibi) ise, çoklu bölmelere (yani, bir düğümden iki veya daha fazla dalın çıkması) izin vermez.
- Ağaç, D 'deki eğitim verilerini temsil eden tek bir düğüm olan N ile başlar (1. adım).
- D 'deki veri grupları tümüyle aynı sınıfa ait ise, N düğümü bir yaprak haline gelir ve bu sınıfla etiketlenir (2. ve 3. adımlar). 4. ve 5. adımlar ise karar ağacını sonlandırma koşullarıdır.
- Aksi takdirde algoritma, bölme kriterini belirlemek için $Attribute_selection_method$ 'u çağırır. $splitting_criterion$, D 'deki veri gruplarını sınıflara göre ayırmak için “en iyi” yola karar veren N düğümünü test eden özneliği ifade etmektedir (6. adım). ayrıca $splitting_criterion$, seçilen testin sonuçlarına göre hangi dalların N düğümünden çıkacağını belirtir.
- N düğümü, düğümde bir test görevi gören $splitting_criterion$ ile etiketlenmiştir (7. adım). $splitting_criterion$ her bir sonucu için N düğümünden bir dal çıkar. D 'deki gözlemler buna göre bölümlendirilir (10. ve 11. adımlar). Böylelikle Şekil 2.15'te gösterildiği gibi üç olası senaryo vardır. Buna göre A , bölme niteliği olsun. A , eğitim verilerine dayanan v ayrı değeri $\{a_1, a_2, \dots, a_v\}$ içerir.
 1. A ayrık değerli ise bu durumda, N düğümündeki testin sonuçları A 'nın bilinen değerlerine doğrudan karşılık gelir. Bir dal, A 'nın bilinen her değeri olan a_j için oluşturulur ve Şekil 2.15 (a)'da gösterildiği gibi bu değerle etiketlenir. D_j parçası, A 'nın a_j değerine D 'deki sınıf-etiketli veri gruplarının alt kümesidir. Verilen bir parçadaki tüm veri gruplarının hepsi, A için aynı değere sahip olduğu için, A 'nın gelecekte herhangi bir

bölme işleminde dikkate alınmasına gerek yoktur. Bu nedenle, *attribute_list*'ten kaldırılır (8. ve 9. adımlar).

2. A sürekli değerli ise bu durumda, N düğümündeki test $A \leq split_point$ ve $A \geq split_point$ koşullarına tekabül eden iki olası sonuca sahiptir; burada *split_point*, *splitting_criterion*'ın bir parçası olarak *Attribute_selection_method* tarafından döndürülen bölme noktasıdır (Pratikte *split_point* yani a , genellikle A 'nın bilinen iki değerinin orta noktası olarak alınır ve bu nedenle eğitim verilerinden A 'nın önceden mevcut olan bir değeri olmayabilir). Böylelikle iki dal, N 'den ayrılır ve Şekil 2.15 (b)'ye göre etiketlenir. Sonuçta veri grupları şu şekilde bölünür: D_1 , $A \leq split_point$ için D 'deki sınıf-etiketli veri gruplarının alt kümesini tutarken, D_2 gerisini tutar.
3. A ayrık değerli ise bu durumda ikili bir ağaç üretilmelidir (öznitelik seçimi ölçütü veya kullanılan algoritma tarafından belirlendiği gibi): N düğümündeki test " $A \in S_A$?" şeklindedir. S_A , A için bölme alt kümesidir ve bölme kriterinin bir parçası olarak *Attribute_selection_method* tarafından döndürülür. S_A , bilinen A değerlerinin bir alt kümesidir. Verilen bir veri grubu A 'nın a_j değerine sahipse ve $a_j \in S_A$ ise, N düğümündeki test yerine getirilir. Böylelikle Şekil 2.15 (c)'de gösterildiği gibi iki dal, N 'den çıkar. N 'nin sol dalı *yes* olarak etiketlenmiştir. Böylelikle D_1 , testi karşılayan D 'deki sınıf-etiketli veri gruplarının alt kümesi ile ilişkilendirilir. N 'den çıkan sağ dal *no* olarak etiketlenir. Böylelikle D_2 , testi karşılamayan D 'nin sınıf-etiketli veri gruplarının alt kümesi ile ilişkilendirilir.



Şekil 2.15: Ayırma kriterine bağlı olarak karar ağacındaki üç olası durum ((Han & Kamber 2006)'dan izin alınarak değiştirilmiştir)

- Algoritma, D 'nin D_j veri grupları için bir karar ağacı oluşturmak için aynı işlemi özyinelemeli olarak kullanır (14. adım).
- Özyinelemeli bölme süreci aşağıdaki durma koşullarından herhangi biri doğru olduğunda durur.
 1. D veri kümesindeki veri gruplarının tümü aynı sınıfa ait ise (2. ve 3. adımlar),
 2. Veri gruplarının daha fazla bölünebileceği kalan nitelik olmadığı (4. adım), çoğunluk oyu kullanılır (5. adım). Bu, N düğümünü bir yaprağa dönüştürmek ve D 'deki en yaygın sınıfla etiketlemek içindir.
 3. Belirli bir dalda veri grubu yoksa. Yani, D_j veri parçası boşsa (12. adım). Bu durumda bir yaprak D 'deki çoğunluk sınıfına göre oluşturulur (13. adım).
- Sonuç olarak karar ağacı döndürülür (15. adım).

Verilen bir D veri parçasını “en iyi” ayıran ayırma kriterini seçmek için öznitelik seçim ölçümü kullanılır. Ayırma kriterinin sonuçlarına göre D , daha küçük parçalara ayrılabilseydi, ideal olarak her bölüm saf olurdu (yani, belirli bir bölüme düşen tüm

veri grupları aynı sınıfa ait). Kavramsal olarak, "en iyi" bölme kriteri, böyle bir senaryoda en çok yakın sonuç veren kriterdir. Öznitelik seçim ölçütleri bölünmüş kurallar olarak da bilinir. Çünkü belirli bir düğümdeki veri gruplarının nasıl bölüneceğini belirler. Öznitelik seçme ölçütü, verilen eğitim verilerini tanımlayan her nitelik için bir sıralama sağlar. Ölçüm için en iyi skora sahip olan öznitelik, verilen veri grupları için bölme niteliği olarak seçilir. Aynı öznitelik sürekli değerli ise veya ikili ağaçlarla sınırlandırılmışsa, sırasıyla bölme noktası veya bölme alt kümesi de bölme kriterinin bir parçası olarak belirlenmelidir. D verisi için oluşturulan ağaç düğümü bölme kriteri ile etiketlenir, dallar kriterin her sonucu için oluşturulur ve gruplar buna göre bölünür. Üç popüler öznitelik seçme ölçütü vardır. Bunlar: Bilgi kazancı, kazanç oranı ve gini dizinidir (Han & Kamber 2006).

ID3, öznitelik seçim ölçütü olarak bilgi kazancı yöntemini kullanır. Bu ölçüt, mesajların değerini veya "bilgi içeriğini" inceleyen bilgi teorisi üzerine Claude Shannon'un öncülüğünü yaptığı çalışmaya dayanmaktadır. D 'deki bir veri grubunu sınıflandırmak için ihtiyaç duyulan beklenen bilgi (2.155)'deki formüle göre hesaplanmaktadır. Bu formülde p_i ifadesi D 'deki C_i sınıfına ait veri gruplarının olasılığıdır. D 'den A niteliği ile ayrılan bir veri grubunu sınıflandırmak için gerekli beklenen bilgi (2.156)'deki formüle göre hesaplanmaktadır. Bilgi kazancı, orijinal bilgi gereksinimi (yani, yalnızca sınıfların oranına bağlı olarak) ile yeni bilgi gereksinim arasındaki fark olarak (yani, A üzerinde bölümlenmeden sonra elde edilen) tanımlanır. (2.157)'da bu denklem gösterilmektedir (Han & Kamber 2006).

$$Info(D) = - \sum_{i=1}^m p_i \log_2(p_i) \quad (2.155)$$

$$Info_A(D) = \sum_{j=1}^v \frac{|D_j|}{|D|} \times Info(D_j) \quad (2.156)$$

$$Gain(A) = Info(D) - Info_A(D) \quad (2.157)$$

Başka bir deyişle, $Gain(A)$ bize A 'ya göre dallanılırken ne kadar kazanç sağlandığını anlatır. Bu, A 'nın değerini bilmekten kaynaklanan bilgi gereksiniminde beklenen azalmadır. En yüksek bilgi kazanımına sahip A özelliği ($Gain(A)$), N düğümündeki ayırma özneliği olarak seçilmiştir. Bu, "en iyi sınıflandırma" yapacak olan A özneliğini ayırmayı istediğimizi söylemekle eşdeğerdir. Böylelikle, veri gruplarını

sınıflandırmayı bitirmek için gerekli bilgi miktarı minimal olmaktadır (yani minimum $Info_A(D)$) (Han & Kamber 2006).

Bilgi kazancı ölçütü çok sayıda çıktıyla test edildiğinde eğilimli olmaktadır. Yani büyük sayıda değerlere sahip öznitelikleri seçmeye daha çok eğilimlidir. Bu nedenle ID3'ün geliştirilmiş versiyonu olan C4.5 algoritması bilgi kazancı ölçütü yerine kazanç oranı ölçütünü kullanmaktadır. Kazanç oranı ölçütü, bilgi kazancı ölçütünün eğilimli olan özelliğinin üstesinden gelmektedir (Han & Kamber 2006). Kazanç oranı, $Info(D)$ ile benzer bir biçimde tanımlı (2.158)'daki "ayırma bilgisi" değerini kullanarak bilgi kazancını bir tür normalize ederek (2.159)'deki formüle göre ölçüm yapar (Han & Kamber 2006).

$$SplitInfo_A(D) = - \sum_{j=1}^v \frac{|D_j|}{|D|} \times \log_2 \left(\frac{|D_j|}{|D|} \right) \quad (2.158)$$

$$GainRatio(A) = \frac{Gain(A)}{SplitInfo(A)} \quad (2.159)$$

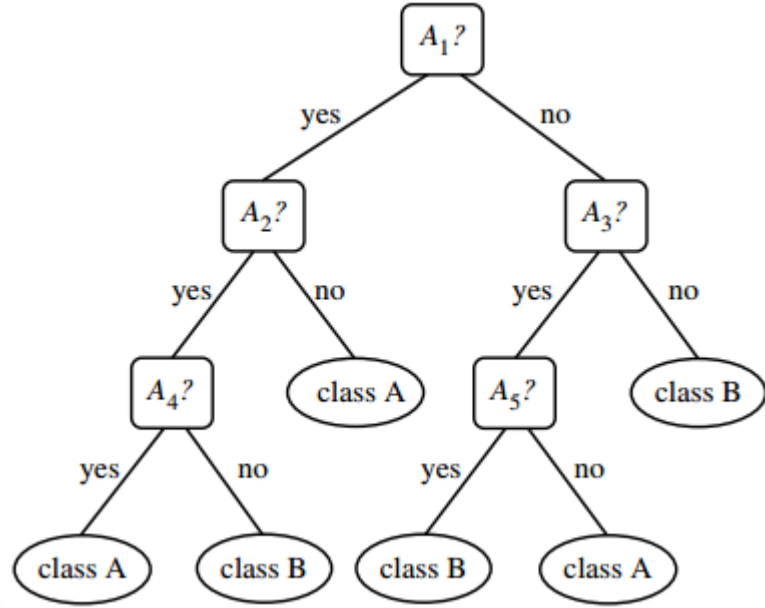
Gini indeksi CART algoritması tarafından kullanılır. Gini indeks ile D'nin katışıklığı ölçülmektedir. Gini indeks (2.160)'deki formül kullanılarak hesaplanır. D verisi A özniteliği kullanılarak D1 ve D2 gibi ikili ayrılırsa Gini indeksi (2.161)'deki formül kullanılarak hesaplanır. Ayrıca ayrık veya sürekli değerli bir özellik üzerinde bir ikili bölünme tarafından oluşacak olan katışıklıktaki azalma (2.162)'deki formül kullanılarak hesaplanır (Han & Kamber 2006).

$$Gini(D) = 1 - \sum_{i=1}^m p_i^2 \quad (2.160)$$

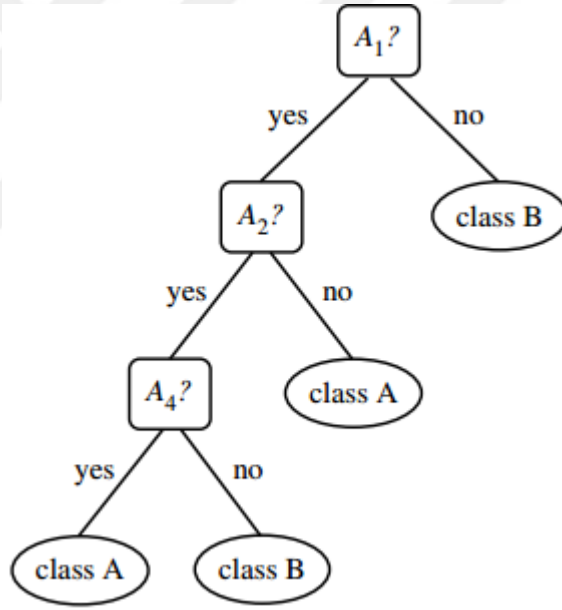
$$Gini_A(D) = \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2) \quad (2.161)$$

$$\Delta Gini(A) = Gini(D) - Gini_A(D) \quad (2.162)$$

Bir karar ağacı oluşturulduğunda dalların birçoğu eğitim verilerindeki anormallikleri gürültüye veya belirsizliklere bağlı olarak yansıtacaktır. Ağaç budama yöntemleri, verilere aşırı uyma sorununu giderir. Bu tür yöntemler, en az güvenilir dalları kaldırmak için tipik olarak istatistiksel yöntemler kullanır (Han & Kamber 2006).



Şekil 2.16: Budanmamış örnek bir karar ağacı (Han & Kamber 2006)



Şekil 2.17: Şekil 2.16'teki budanmamış karar ağacının budanmış biçimi (Han & Kamber 2006)

Şekil 2.16'te hazırlanmamış bir ağaç ve Şekil 2.17'te budanmış biçimi gösterilmektedir. Budamalı ağaçlar daha küçük ve daha az karmaşıktır ve dolayısıyla anlaşılması daha kolaydır. Bağımsız test verilerini (yani daha önce görülmemiş olanların) düzensiz ağaçlara göre doğru şekilde sınıflandırılmasında genellikle daha hızlı ve daha iyidirler (Han & Kamber 2006).

2.5.6 Random forests

Breiman (Breiman 2001) tarafından yapılan çalışmada ağaç tahmin edicilerin birleştirilmesine dayalı *Random Forests* isimli bir algoritma tanıtılmıştır. Algoritmada ormanlar için genelleme hatası ormandaki ağaç sayısına göre bir limite yakınsar. Sadece bir orman için genelleme hatası ormandaki bireysel ağaçların gücüne ve onlar arasındaki ilişkiye bağlıdır. Herbir düğümü ayırmak için rastgele bir özellik seçimi kullanma AdaBoost'a tercihen daha uygun hata oranları verir. Ayrıca AdaBoost'a kıyasla gürültüye karşı daha dirençlidir (Breiman 2001).

2.6 Sınıflandırıcı Performansını Ölçmek İçin Kullanılan Ölçüt ve Yöntemler

Bir öğrenme algoritmasının öğrenme süreci sonunda performansına karar verilebilmesi için çeşitli istatistiksel kriterlere gerek vardır. Bu tez çalışmasına konu olan öğrenme algoritmalarının performanslarını ölçmek için kullanılacak kriterler şunlardır: F-Ölçütü, Ortalama Kareysel Hatanın Karekökü (Root Mean Squared Error (RMSE)) ve Kappa değeridir. Bu tez çalışması kapsamında yapılan deneylerin sonuçlarının doğruluğunu garanti altına almak için ise tüm deneylerde k-Katlı Çapraz Doğrulama (k-KÇD) metodu uygulanacaktır.

2.6.1 F-Ölçütü (F_1)

F-Ölçütü (Van Rijsbergen 1979); (2.163)'te ifade edilen *Kesinlik* (precision) ve (2.164)'te ifade edilen *Hassasiyet* (recall) değerlerinin harmonik ortalaması olarak ifade edilir. (2.165)'te dengeli F-Ölçütünün denklemi gösterilmektedir. Ancak gerçekte F-Ölçütünün tam ifadesi (2.166)'da gösterilmektedir (Chinchor 1992). Bu ifadede β sembolü *Kesinlik* ve *Hassasiyet* arasındaki dengeyi ifade etmektedir. İkili bir sınıflandırma problemi (pozitif ve negatif) için *kesinlik*, pozitif olarak tahmin edilen örnekler arasından doğru olarak sınıflandırılan pozitiflerin oranıdır. *Hassasiyet* ise gerçek sınıfı pozitif olan örnekler arasından doğru olarak sınıflandırılan pozitif örneklerin oranıdır (Dumais et al. 2009; Witten et al. 2011). Çizelge 2.5'te ikili bir sınıflandırma problemi için doğruluk tablosu gösterilmektedir.

Çizelge 2.5: İki sınıflı bir tahmin için doğruluk tablosu (Stehman 1997)

		Tahmini Sınıf	
		Evet	Hayır
Gerçek Sınıf	Evet	Gerçek pozitif (GP)	Hatalı pozitif (HP)
	Hayır	Hatalı negatif (HN)	Gerçek negatif (GN)

$$Kesinlik = \frac{GP}{GP + HP} \quad (2.163)$$

$$Hassasiyet = \frac{GP}{GP + HN} \quad (2.164)$$

$$F_1 = \frac{2 * Kesinlik \times Hassasiyet}{Kesinlik + Hassasiyet} \quad (2.165)$$

$$F_\beta = \frac{(\beta^2 + 1) \times Kesinlik \times Hassasiyet}{\beta^2 \times Kesinlik + Hassasiyet}, (0 \leq \beta \leq +\infty) \quad (2.166)$$

F-ölçütü özellikle eğitim kümesinin hazırlanması boyunca öğrenme algoritmasının performansını arttırmak için kullanılmaktadır. Bunun için F-ölçütü ve veri büyüklüğü arasında çizilen öğrenme eğrisinden bir mutluluk grafiği (Russell & Norvig 2003) elde edilmeye çalışılmaktadır. Böylelikle gereksiz veriler ayıklanarak öğrenme algoritmasının performansı artırılır. Ayrıca F-ölçütü sınıfların tanı için yeterli olup olmadığına karar vermek amacıyla kullanılmaktadır. Genel olarak bu tez çalışmasında kabul edilebilir F-ölçütü değeri minimum 0.5 olarak kabul edildi.

2.6.2 Ortalama karesel hata

Ortalama Karesel Hatanın Karekökü (OKHK), Ortalama Karesel Hata (Mean Squared Error (MSE)) değerinin karekökünün alınmasıyla hesaplanır. Ortalama Karesel Hata (OKH) yaygın bir biçimde birçok alanda kullanılmaktadır. Birçok matematiksel yöntem (lineer regresyon gibi) matematiksel olarak ele alınması kolay olduğu için OKH'yi kullanılmaktadır. Makine Öğrenmesi alanında kullanımı ise öğrenme algoritmalarının performanslarını ölçmektir (Witten et al. 2011).

Bir tahmin edicinin hata oranı rastgele bir tahmin yapmasından ya da doğru bir tahmin yapamamasından kaynaklanmaktadır (Lehmann & Casella 1998). OKH ve OKHK değeri sıfıra ne kadar yakınsa bir tahmin edicinin yapmış olduğu hata da o kadar az ya da düşüktür. Ayrıca her öğrenme problemi için OKH ya da OKHK'nin

kabul edilebilir hata değeri farklı olabilmektedir. OKH ve OKHK'nin formülleri (2.167) ve (2.168) ifadelerinde sırasıyla yer almaktadır (Witten et al. 2011). Bu formüllerde yer alan semboller için; p , tahmini değeri; a ise gerçek değeri ifade etmektedir.

$$OKH = \frac{(p_1 - a_1)^2 + \dots + (p_n - a_n)^2}{n} \quad (2.167)$$

$$OKHK = \sqrt{\frac{(p_1 - a_1)^2 + \dots + (p_n - a_n)^2}{n}} \quad (2.168)$$

2.6.3 Kappa değeri

Kappa değeri bir eğitim kümesi üzerinde gerçekleştirilen öğrenme sonucunda yapılan tahminler ve gerçek sınıflar arasındaki uyumu ölçmek için kullanılmaktadır (Witten et al. 2011). Kappa değeri ilk olarak Cohen (Cohen 1960) tarafından geliştirildi. Cohen'e göre Kappa değeri, iki gözlemci arasındaki uyumu ölçmek için kullanılan bir ölçüttür (Cohen 1960). Kappa değeri geliştirildiği andan beri birkaç çalışma alanında kullanıldı. Makine öğrenmesi alanı da bu disiplinlerden bir tanesidir. Kappa değeri, Makine öğrenmesi alanında bir öğrenme algoritmasının doğruluğunu arttırmaya yardımcı bir ölçüt olarak kullanılmaktadır. Kappa değeri (2.169)'de gösterildiği gibi hesaplanmaktadır (Cohen 1960).

$$K = \frac{P(a) - P(e)}{1 - P(e)} \quad (2.169)$$

$P(a)$ öğrenme algoritmasının doğruluğunu ve $P(e)$ aynı eğitim kümesi üzerinde tahmin yapan öğrenme algoritmasının elde etmiş olduğu beklenen doğruluğu ifade etmektedir. Yani yapılan tahminin ortaya çıkma olasılığıdır. Cohen'in Kappa değerine yapılan en önemli eleştiri bazen sezgilerimizden uzak olacak şekilde beklenenden farklı sonuçlar vermesidir (Gwet 2010). Örneğin iki sınıftan oluşmuş ve veri sayısı aynı iki eğitim kümesi üzerinde gerçekleştirilen öğrenme sonucunda öğrenme algoritmasının gerçekleştirmiş olduğu doğruluk oranları eşit olmasına rağmen Kappa değeri iki eğitim kümesi için de farklı olabilmektedir. Ancak birçok eğitim kümesi üzerinde sınıflandırıcıların öğrenme yeteneklerini ölçmek için başarılı bir ölçüt olarak kullanılmaktadır.

Kappa değeri -1 ve 1 arasında değişmektedir. -1 değeri tam bir uyumsuzluğu ya da ters yönde bir ilişki durumunu belirtmektedir. 1 ise kusursuz bir uyumu belirtmektedir. Kappa değeri 0.4 ve üzeri bir değer ise şansın ötesinde bir uyumluluktan bahsedilebilir (Landis & Koch 1977). Landis ve Koch tarafından Kappa değerine yapılan yorum Çizelge 2.6’da gösterilmektedir.

Çizelge 2.6: Kappa değeri için yorum tablosu (Landis & Koch 1977)

Kappa değeri	Yorum
<0.00	İlişki yok yada ters ilişki
0.00 – 0.20	Önemsiz ilişki
0.21 – 0.40	Orta derecede ilişki
0.41 – 0.60	Kabul edilebilir ilişki
0.61 – 0.80	Önemli derecede ilişki
0.81 – 1.00	Mükemmel ilişki

2.6.4 k-Katlı çapraz doğrulama

Eğitim kümesinin bir öğrenme algoritması tarafından yeterince iyi öğrenilip öğrenilmediğini anlamının çeşitli yöntemleri vardır. Bu yöntemlerden bir tanesi de k-KÇD’dir. k-KÇD verinin hemen hemen k eşit parçaya bölünerek bir öğrenme algoritması tarafından öğrenilebilmesi için bir yöntem sunar. Böylece öğrenme algoritması k defa hem test hem de eğitilmiş olur. Ayrıca öğrenme algoritması daima eğitim kümesi üzerinde bir eksik katla eğitilir (Kohavi & Provost 1998). Yani k alt parçadan $(k-1)$ ’i eğitim amacıyla kullanılırken geriye kalan kat ise test kümesi olarak kullanılır (Alpaydın 2014). Öğrenme algoritmasının doğruluk oranı ise k adet kat için ortalama doğruluk hesaplanarak bulunur (Kohavi & Provost 1998).

k-KÇD kullanılarak eğitim kümesi için veri sayısında sınırlamalara gidilebilir (Witten et al. 2011). Böylece gerekli olmayan verinin eğitim kümesi içerisinde çıkarılması sağlanabilir. Bu da k-EYK gibi örnek tabanlı yaklaşımları kullanan algoritmalar için hem eğitim hem de test sürecinde performans sağlayacaktır. Ayrıca k-KÇD yöntemi kullanılarak bias-varyans ikilemi dengelenebilmektedir (Mitchell 1997).

k-KÇD yöntemi hesaplamasal maliyet açısından yüksektir. Yalnız bir öğrenme algoritmasının hata oranını tahmin etmek istediğimizde çok faydalı bir yöntemdir.

Aynı zamanda az sayıda veriye sahip iki öğrenme algoritmasını birbiriyle kıyaslamak için de k-KÇD yönteminden faydalanılır (Mitchell 1997). k-KÇD’de önemli bir problem eğitim ve test kümesinin her ikisine dağılmış olan örneklerin sınıfları arasındaki dengesizliktir (Witten et al. 2011). Bu sınıf dengesizliğinden dolayı öğrenme algoritmaları sağlıklı bir öğrenme süreci gerçekleştirememektedirler. Bu nedenle öğrenme algoritmalarının hata oranları olması gerekenden daha yüksek çıkabilmektedir. Bu sorunu çözenin yolu; her bir sınıfın hem eğitim hem de test kümesi içerisinde dengeli bir biçimde yer almasıdır. Böylelikle sorunun çözümü sağlanabilmektedir. Sorunun çözümünde uygulanan bu teknik “stratification” olarak isimlendirilmektedir. “Stratification” tekniği, eğitim ve test kümelerindeki dengesizliğe karşı bir koruma sağlamaktadır (Witten et al. 2011). WEKA, k-KÇD yönteminde “stratification” tekniğini uygulamaktadır. Bu da öğrenme süreçleri için güvenilir doğruluk oranları vermektedir.

Bu tez çalışmasında k değeri yapılan deneylerin çok büyük bir kısmında 10 olarak alındı. Bunun nedeni ise: farklı öğrenme yöntemleri kullanılarak gerçekleştirilen çeşitli eğitim kümeleri üzerindeki yaygın testlerde hata oranının en doğru tahminini yapmak için k 'nın en doğru değerinin 10 civarında olduğunun görülmesidir. Ayrıca yapılan teorik çalışmalarda da k değerinin yaklaşık olarak 10 olduğu gösterilmektedir (Witten et al. 2011).

Makine Öğrenmesi algoritmalarının performanslarını test ederken bazen k-KÇD yönteminin özelleşmiş hali olan Leave-one-out Çapraz Doğrulama (LOOÇD) kullanılır. Bu yöntem de doğrulama verisi olarak orijinal örnekten tek bir gözlem örneği kullanma esasına dayanır. Yani orijinal örnekler içerisinde bir örnek doğrulama örneği olurken geriye kalan örnekler eğitim örneği olarak kullanılır. Bu durum tüm örnekler doğrulama örneği olarak kullanılana kadar tekrarlanır. Bu yöntem özellikle sınıflandırıcıların oluşturmuş oldukları modelleri birbirlerinden ne kadar farklı olduklarını ortaya koymak için kullanılır. Eğer her bir durumda modeller birbirinden oldukça farklı sonuçlar üretiyorsa orijinal örneklerin gözden geçirilmesi gerekmektedir. Çünkü bu verilerin gürültülü veriler olma ihtimali vardır (Witten et al. 2011).

3. ANALİZ VE UYGULAMA

Bu tez çalışmasında gerçekleştirilen tüm deneyler aşağıda belirtilen özellikteki bir bilgisayar üzerinde gerçekleştirildi: Intel Core i7 CPU Q740 1.73 GHZ, 8 GB RAM ve 64-bit Windows 7 Ultimate Service Pack 1.

Yapılan deneylerde kullanılan Makine Öğrenmesi algoritmaları WEKA aracıda yer alan algoritmalarıdır. Bu algoritmalar WEKA aracı içerisinde 7 kategori altında gruplanmıştır. Bu kategoriler şunlardır: bayes (bayesian öğrenme), functions (fonksiyon öğrenme), lazy (örnek ya da tembel öğrenme), misc (farklı öğrenme yaklaşımları), rules (kural öğrenme), trees (karar ağacı öğrenme) ve meta (meta ya da topluluk öğrenme)'dir. Bu kategorilere göre deneylerde kullanılan algoritmaların listesi sırasıyla Çizelge B.1, Çizelge B.2, Çizelge B.3, Çizelge B.4, Çizelge B.5, Çizelge B.6 ve Çizelge B.7'de gösterilmektedir.

3.1 VIBES ve Diğer Makine Öğrenmesi Algoritmalarının Benchmark Veri Kümelerine Uygulanması

VIBES algoritmasının performansı Çizelge 2.1'de belirtilen veri kümeleri üzerinde uygulandı. Bu tez çalışmasındaki tüm deneylerde *VIBES* algoritması temel öğrenici olarak 6 temel öğrenici kullandı. Bu temel öğreniciler şunlardır: k-EYK, NB, DYM, Diskriminant Analiz (DA), Karar Ağacı (KA) ve Random Forest (RF)'dir. *VIBES* algoritmasının kodlanması ve veri kümeleri üzerine uygulanması MATLAB R2016b aracı kullanılarak gerçekleştirildi. *VIBES* algoritması dört ana kısımdan oluşmaktadır. Bunlar:

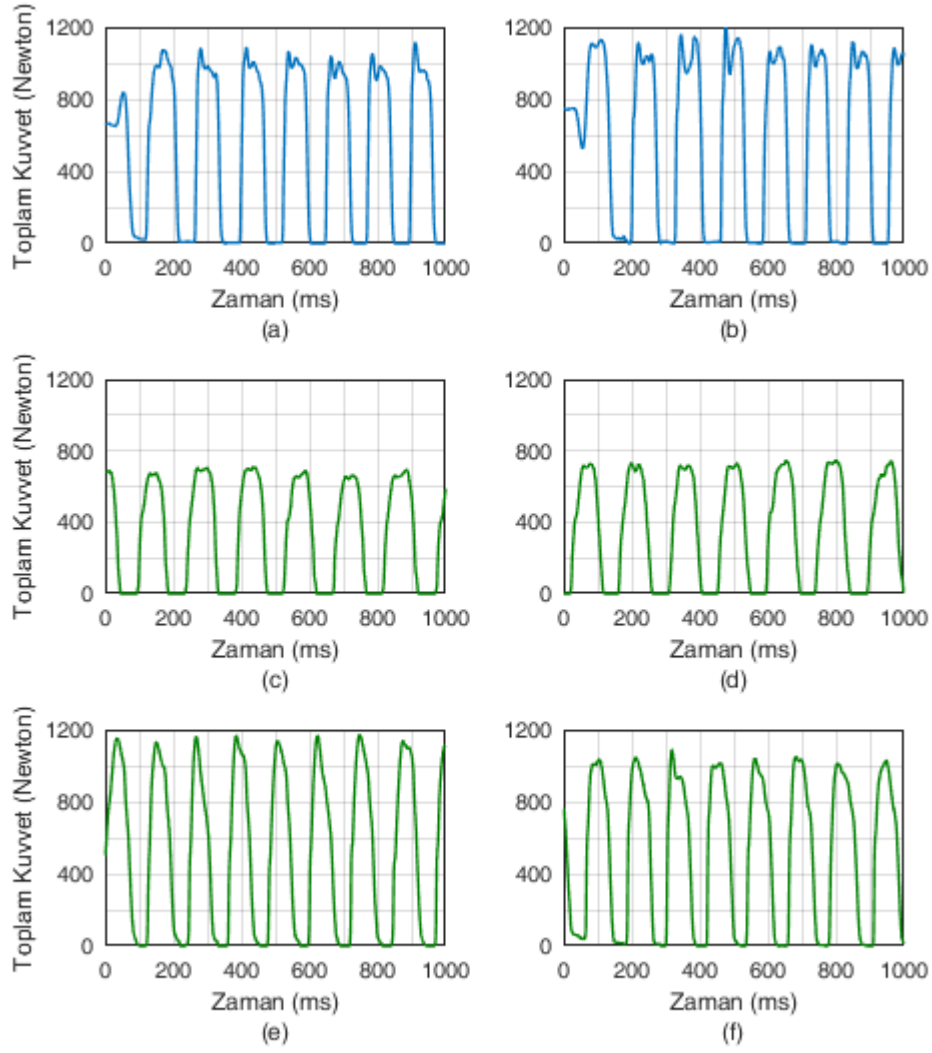
- Özniteliklerin enformasyon ölçütüne göre sıralanmalarını içeren kısımdır. Bu kısımda iki yöntem kullanılmıştır. Bunlar ReliefF ve Shannon enformasyonudur. ReliefF algoritması MATLAB R2016b versiyonunda tanımlıdır. Shannon enformasyonunun hesaplanması için Pereira (Pereira 2015) tarafından geliştirilen hazır fonksiyon kullanılmıştır.
- Temel öğreniciler kullanılarak veri üzerinde hipotezlerin oluşturulduğu kısım.

- Birbiriyle uyumlu hipotezlerin biraraya getirildiği kısım.
- Oluşan bu topluluğun performansının istatistiksel özetinin hesaplandığı kısımdır.

Bu dört ana kısma ait kodlar Şekil B.2, Şekil B.3, Şekil B.4, Şekil B.7 ve Şekil B.8’te gösterilmektedir. *VIBES* algoritması tüm veri kümeleri üzerinde 10-katlı çapraz doğrulama yöntemi uygulanarak 3 kere tekrarlandı. Bununla ilgili kodlar Şekil B.1’de gösterilmektedir. Böylelikle bu 3 denemenin ortalaması *VIBES* algoritmasının her bir veri kümesi için sınıflandırma doğruluğu olarak atandı. Çoğunluk oylamalı topluluk öğrencilerde hata analizi için geliştirilen ve 2.3.4 bölümünde tanıtılan yöntem *VIBES* algoritması üzerinde uygulandı. Böylelikle gerçek sonuçlar ile teorik sonuçların karşılaştırması yapıldı. Bu yöntemle ait kodlar Şekil B.9’da gösterilmektedir. Tüm bu yapılan deneylerin sonuçları 4.1. bölümde yer almaktadır. Ayrıca *VIBES* algoritması diğer Makine Öğrenmesi algoritmaları ile de karşılaştırıldı. Bu Makine Öğrenmesi algoritmaları Benchmark veri kümelerine WEKA 3.9.1 versiyonu üzerinde uygulanarak sonuçlar elde edildi (Sonuçlar elde edilirken 10-katlı çapraz doğrulama yöntemi kullanıldı). Daha sonra bu sonuçlar *VIBES* algoritması ile karşılaştırıldı.

3.2 VIBES ve Diğer Makine Öğrenmesi Algoritmalarının Parkinson Veri Kümesine Uygulanması

VIBES algoritması, PhysioNET veritabanından elde edilen Parkinson veri kümesi üzerinde de uygulandı. Bu veri kümesi, idiyopatik Parkinsonlu 93 hasta ve 73 sağlıklı kontrol bireyinden elde edilen yürüme ölçümlerini içermektedir. Yani bu veri kümesi, yer seviyesinde yaklaşık olarak 2 dakika boyunca deneklerin yürürken yere uyguladıkları yer tepki kuvveti kayıtlarını da içerir. Bu veri kümesi için örnek bir tepki kuvveti kaydı Şekil 3.1’de gösterilmektedir.



Şekil 3.1: (a ve b) kontrol bireyi, (c ve d) ve (e ve f) parkinsonlu iki bireye ait sol ve sağ ayak için yere uygulanan kuvvetlerin zamana göre değişimi

VIBES algoritması Parkinson veri kümesine uygulanmadan önce veri kümesine ait özniteliklerin oluşturulması gerekmektedir. Ayrıca öznitelik oluştururken sinyalin ham hali ve her bir sensöre ait sinyallerin birbirleriyle olan etkileşimleri de kullanıldı. Daha sonra deneklere ait ham sinyaller ve etkileşimli sinyaller, sinyal dönüşüm yöntemleri ile dönüştürüldü. Sinyallerin dönüştürülmesinde Fourier dönüşümü, Dalgacık dönüşümü ve HHD kullanıldı. Her bir dönüşüm işlemi sonrasında elde edilen öznitelikler daha sonra *OneRAttributeEval* öznitelik seçim algoritması kullanılarak seçildi. Böylelikle daha yüksek enformasyon veren öznitelikler seçilirken düşük enformasyon veren öznitelikler elenmiş oldu. Özniteliklerin seçimi esnasında *OneRAttributeEval* algoritması 10-katlı çapraz

doğrulama yöntemi kullanılarak çalıştırıldı. Bunun sonucunda özniteliklerin her birinin veriyi doğru sınıflandırma yüzdeleri elde edilir. 10-katlı çapraz doğrulama yöntemi uygulandığı için doğru sınıflandırma oranı bu 10-kat'ın ortalamasıdır. Böylelikle her bir öznitelik için ($ort \pm sapma$) şeklinde sonuçlar elde edilir. Öznitelikler seçilirken (3.1)'deki kurala uyan öznitelikler seçilir.

$$(ort - sapma \geq 70) \quad (3.1)$$

Böylelikle onbinlerce öznitelik arasından sadece sayısı yüzlerle ifade edilebilecek kadar öznitelik seçilir. Parkinson veri kümesinden öznitelik oluşturma ve bu öznitelikler arasından öznitelik seçme süreci ayrıntılı bir biçimde aşağıda açıklanmaktadır.

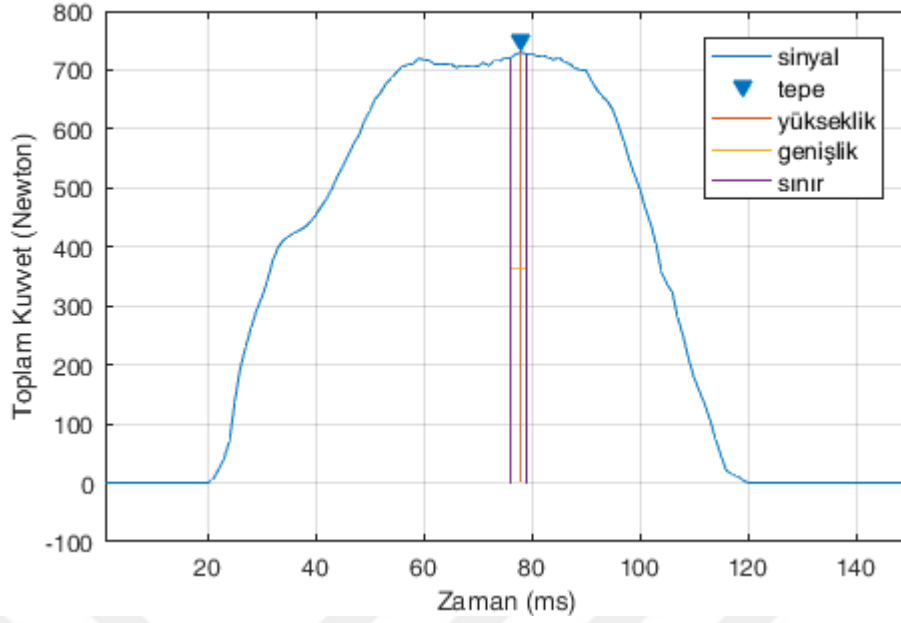
Deneklerin her bir ayağından elde edilen 16 adet sinyali şu şekilde etiketleyelim: $L_1, L_2, L_3, L_4, L_5, L_6, L_7, L_8, R_1, R_2, R_3, R_4, R_5, R_6, R_7$ ve R_8 . Daha sonra her bir ayağa ait sinyallerin toplamını ifade eden 2 adet sinyali şu şekilde etiketleyelim: T_1 ve T_2 . Böylelikle her bir denek için toplam 18 sinyalimiz oldu. Son olarak bu 18 sinyalin kendi aralarında çeşitli fonksiyonlara tabi tutulmasıyla yeni sinyaller elde ettik. Sinyaller üzerinde uygulanan fonksiyonlar (3.2), (3.3) ve (3.4)'te gösterilmektedir.

$$f_1(x, y) = x + y \quad (3.2)$$

$$f_2(x, y) = x - y \quad (3.3)$$

$$f_3(x, y) = |x - y| \quad (3.4)$$

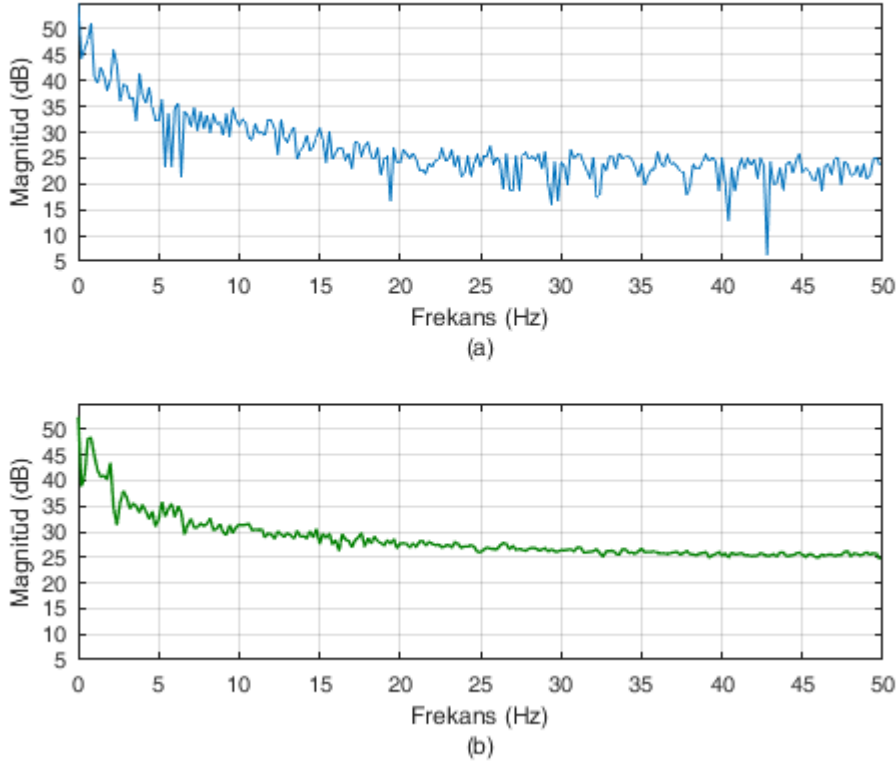
Örneğin L_1 ve L_2 sinyalleri için bu fonksiyonlar uygulandığında yeni oluşacak 3 sinyal sırasıyla şu şekilde olacaktır: L_1+L_2, L_1-L_2 ve $|L_1-L_2|$. Bu 18 temel sinyalin kendi aralarında tekrara yer vermeksizin bu 3 fonksiyona göre etkileşmeleri sonucunda 459 yeni sinyal elde edilir. Böylelikle 18 temel sinyal ve 459 yeni sinyalle birlikte her bir denek için toplamda 477 sinyal elde etmiş olduk. Sinyaller elde edildikten sonra sinyal dönüşüm yöntemleri kullanılarak sinyallerin dönüştürülmesi gerçekleştirildi. Son olarak 477 adet ham sinyal ve bu sinyallerin dönüştürülmesiyle elde edilen dönüşüm sinyallerinin tepe analizleri yapıldı. Şekil 3.2'de parkinsonlu bireye ait sağ ayak sinyaline ait tepe analizi gösterilmektedir.



Şekil 3.2: Parkinsonlu bireye ait sağ ayak için yere uygulanan kuvvet sinyalinin tepe analizi

Tepe analizi sonucunda sinyallerin; maksimum ekstrem değerleri, bu ekstrem değerlerine ait konumlar, tepelere ait genişlikler ve tepenin yükseklik değerlerine ait 4 adet yöney elde edilir. Tepe analizleri yapıldıktan sonra her bir sinyale ait bu 4 adet yöney üzerinde istatistiksel fonksiyonlar uygulanarak istatistiksel öznitelikler elde edilir. Ayrıca bazı dönüşüm sinyallerinin ham halleri kullanılarak doğrudan istatistiksel öznitelikler de elde edilir. Bu sinyallere uygulanan istatistiksel fonksiyonlar Çizelge B.8’de gösterilmektedir. Ayrıca Şekil B.10’de sinyale ait istatistiklerin hesaplanması için yazılan MATLAB fonksiyonu gösterilmektedir. Bu çizelgedeki fonksiyonların bazıları matematiksel notasyonla gösterilirken geri kalan diğerleri ise MATLAB fonksiyonu olarak ifade edildi.

Sinyaller üzerinde uygulanan ilk dönüşüm yöntemi Fourier dönüşümüdür. Fourier dönüşümü ile zaman domenindeki sinyal frekans domenine dönüştürülür. Bu duruma ilişkin olarak kontrol bireyi ve parkinsonlu bireye ait frekans domeni Şekil 3.3’te gösterilmektedir. Her bir denek için 477 adet sinyal ve bu sinyallerin Fourier dönüşümü kullanılarak elde edilen öznitelik sayısı 85860’tır. Bu öznitelikleri elde etmek için Şekil B.11’deki MATLAB kodları yazılmıştır. Daha sonra *OneRAttributeEval* algoritması kullanılarak (3.1)’deki koşula göre 85860 öznitelik içerisinde sadece 24 öznitelik seçilmiştir. Bu özniteliklerin listesi Çizelge B.9’de gösterilmektedir.

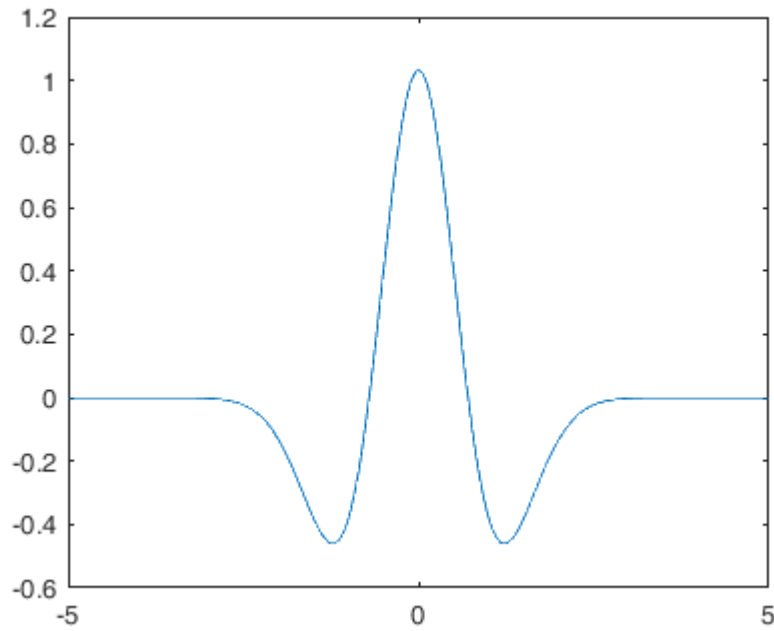


Şekil 3.3: (a) Kontrol bireyi ve (b) Parkinsonlu bireye ait sağ ayak sinyalinin tek taraflı magnitüd spektrumu

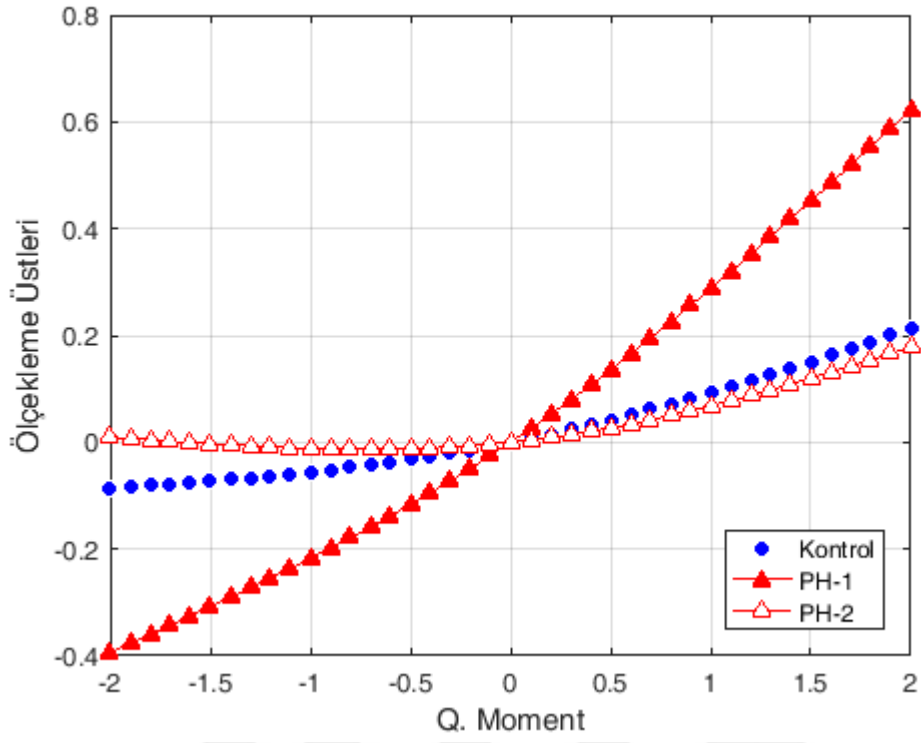
Sinyaller üzerinde uygulanan ikinci dönüşüm yöntemi Dalgacık dönüşümüdür. Dalgacık dönüşümü ile bir sinyalin zaman-frekans-ölçek çözünürlüğü elde edilebilmektedir. Bu tez çalışmasında sinyallerin Dalgacık dönüşümü için yedi dalgacık ailesi kullanıldı. Bu dalgacık aileleri şunlardır: Gaus, BiorSplines, Coiflets, Daubechies, DMeyer, Haar ve Symlets'dir. Bu dalgacık ailelerinden Gaus dalgacığı sürekli dalgacık dönüşümü yaparken diğer dalgacık aileleri ayrık dalgacık dönüşümü gerçekleştirmektedirler.

2. dereceden Gaussian dalgacığı ile, her bir denek için 477 adet sinyal ve bu sinyallerin dalgacık dönüşümü kullanılarak elde edilen öznelilik sayısı 92061'dir. Bu öznelilikleri elde etmek için Şekil B.12'deki MATLAB kodları yazılmıştır. Daha sonra *OneRAttributeEval* algoritması kullanılarak (3.1)'deki koşula göre 92061 öznelilik içerisinde sadece 23 öznelilik seçilmiştir. Bu özneliliklerin listesi Çizelge B.10'te gösterilmektedir. Şekil 3.4'te 2. Dereceden gaussian dalgacık gösterilmektedir. Ayrıca deneklere ait kuvvet sinyallerinin ölçekleme kuvvetleri de hesaplanmıştır. Ölçekleme kuvveti bir sinyalin monofraktal ya da multifraktal olduğunu belirten bir ölçektir. Şekil 3.5'te kontrol bireyi ve iki parkinsonlu bireye ait

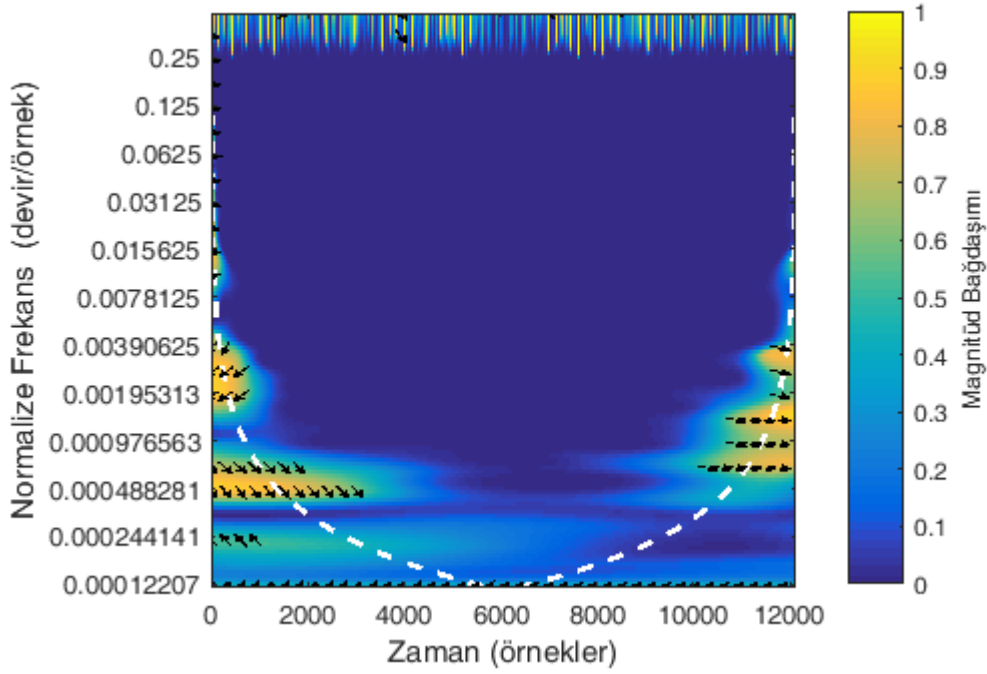
kuvvet sinyallerinin; ölçekleme üstleri-Q. moment grafiđi gösterilmektedir. Buna göre sekildeki bireylere ait kuvvet sinyalinin ölçekleme kuvvetinin Q. Momente göre grafiđi eğrisel bir fonksiyon olduđu için bu sinyaller multifraktaldır. Diđer türlü monofraktal bir sinyal olmaktadır. Şekil 3.6'da parkinsonlu bir bireye ait kuvvet sinyalinin zaman sinyaline göre dalgacık uyumu gösterilmektedir. Şekil 3.7'de kontrol bireyine ait kuvvet sinyalinin zaman sinyaline göre dalgacık uyumu gösterilmektedir. Dalgacık uyumu iki sinyal arasındaki ilişkiyi gösteren bir ölçüttür. Burada her bir deneđe ait kuvvet sinyali deđişirken zaman sinyali deđişmemektedir. Buradaki amaç deđişmeyen bir sinyale göre diđer sinyallerin ilişkilerini ölçmektir.



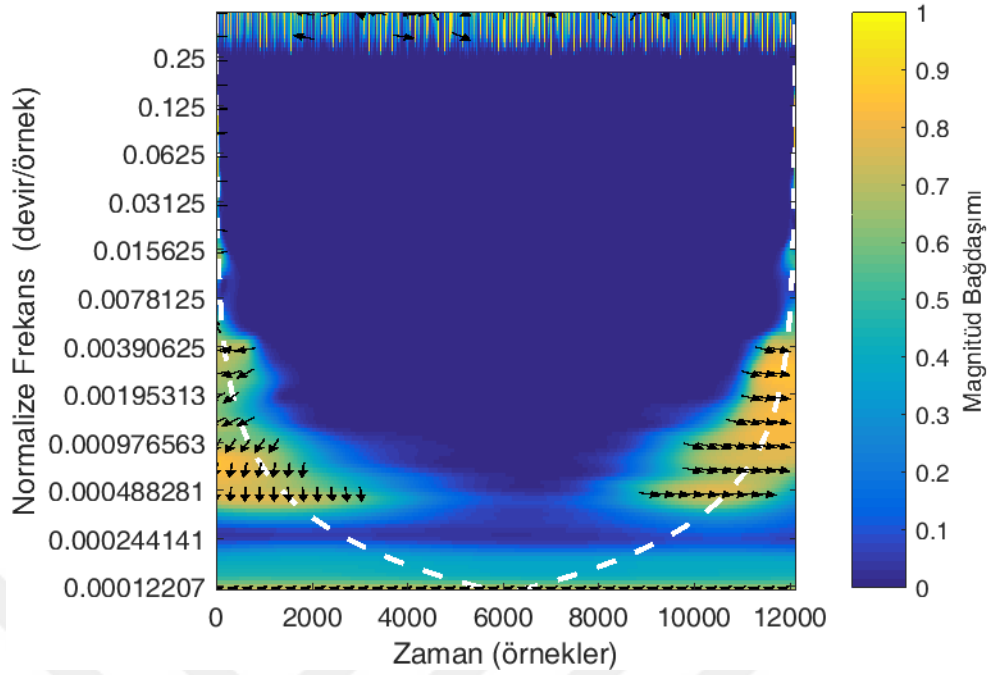
Şekil 3.4: 2. Dereceden gaussian dalgacık



Şekil 3.5: Kontrol bireyi (Kontrol) ve iki parkinsonlu bireye (PH-1 ve PH-2) ait sağ ayak sinyalinin ölçekleme üstü - Q. moment grafiği



Şekil 3.6: Parkinsonlu bireye ait sağ ayak sinyalinin zamana göre dalgacık uyumu



Şekil 3.7: Kontrol bireyine ait sağ ayak sinyalinin zamana göre dalgacık uyumu

Her bir deneğe karşılık her bir BiorSplines3.3, Coiflets2, Dmeyer, Haar, Symlets2, Daubechies2, Daubechies3, Daubechies4, Daubechies5, Daubechies6 ve Daubechies7 dalgacıkları için elde edilen öznitelik sayısı 278091'dir. Bu öznitelikleri elde etmek için Şekil B.13'deki MATLAB kodları yazılmıştır. Daha sonra *OneRAttributeEval* algoritması kullanılarak (3.1)'deki koşula göre 278091 öznitelikten her bir dalgacık türü için öznitelikler seçilmiştir.

BiorSplines3.3 dalgacığı kullanarak oluşturulan öznitelikler içerisinde sadece 146 öznitelik seçilmiştir. Bu özniteliklerin listesi Çizelge B.11'te gösterilmektedir. Şekil B.14'de parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için BiorSplines3.3 dalgacığına ait 6. seviyeye kadar detay katsayıları gösterilmektedir.

Coiflets2 dalgacığı kullanılarak oluşturulan öznitelikler içerisinde sadece 147 öznitelik seçilmiştir. Bu özniteliklerin listesi Çizelge B.12'te gösterilmektedir. Şekil B.15'de parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için Coiflets2 dalgacığına ait 6. seviyeye kadar detay katsayıları gösterilmektedir.

DMeyer dalgacığı kullanarak oluşturulan öznitelikler içerisinde sadece 149 öznitelik seçilmiştir. Bu özniteliklerin listesi Çizelge B.19'de gösterilmektedir. Şekil B.16'te parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için DMeyer dalgacığına ait 6. seviyeye kadar detay katsayıları gösterilmektedir.

Haar dalgacıđı kullanılarak oluşturulan öznitelikler içerisinde sadece 114 öznitelik seçilmiştir. Bu özniteliklerin listesi Çizelge B.20’te gösterilmektedir. Şekil B.17’te parkinsonlu bireyin sağ ayađına ait kuvvet sinyali için Haar dalgacıđına ait 6. seviyeye kadar detay katsayıları gösterilmektedir.

Symlets2 dalgacıđı kullanılarak oluşturulan öznitelikler içerisinde sadece 126 öznitelik seçilmiştir. Bu özniteliklerin listesi Çizelge B.21’te gösterilmektedir. Şekil B.18’te parkinsonlu bireyin sağ ayađına ait kuvvet sinyali için Symlets2 dalgacıđına ait 6. seviyeye kadar detay katsayıları gösterilmektedir.

Daubechies2 dalgacıđı kullanılarak oluşturulan öznitelikler içerisinde sadece 125 öznitelik seçilmiştir. Bu özniteliklerin listesi Çizelge B.13’da gösterilmektedir. Şekil B.19’da parkinsonlu bireyin sağ ayađına ait kuvvet sinyali için Daubechies2 dalgacıđına ait 6. seviyeye kadar detay katsayıları gösterilmektedir.

Daubechies3 dalgacıđı kullanılarak oluşturulan öznitelikler içerisinde sadece 142 öznitelik seçilmiştir. Bu özniteliklerin listesi Çizelge B.14’de gösterilmektedir. Şekil B.20’de parkinsonlu bireyin sağ ayađına ait kuvvet sinyali için Daubechies3 dalgacıđına ait 6. seviyeye kadar detay katsayıları gösterilmektedir.

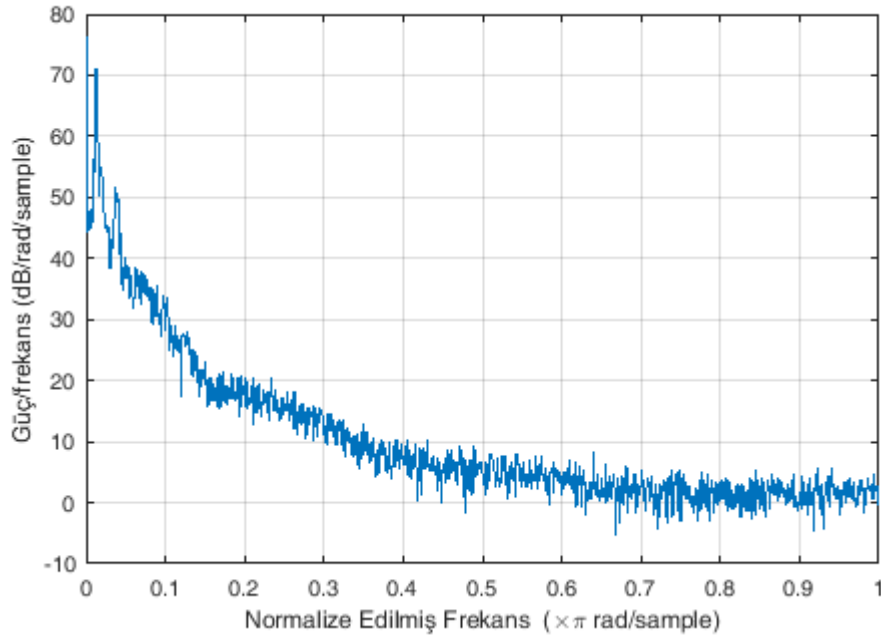
Daubechies4 dalgacıđı kullanılarak oluşturulan öznitelikler içerisinde sadece 172 öznitelik seçilmiştir. Bu özniteliklerin listesi Çizelge B.15’de gösterilmektedir. Şekil B.21’de parkinsonlu bireyin sağ ayađına ait kuvvet sinyali için Daubechies4 dalgacıđına ait 6. seviyeye kadar detay katsayıları gösterilmektedir.

Daubechies5 dalgacıđı kullanılarak oluşturulan öznitelikler içerisinde sadece 132 öznitelik seçilmiştir. Bu özniteliklerin listesi Çizelge B.16’da gösterilmektedir. Şekil B.22’de parkinsonlu bireyin sağ ayađına ait kuvvet sinyali için Daubechies5 dalgacıđına ait 6. seviyeye kadar detay katsayıları gösterilmektedir.

Daubechies6 dalgacıđı kullanılarak oluşturulan öznitelikler içerisinde sadece 130 öznitelik seçilmiştir. Bu özniteliklerin listesi Çizelge B.17’de gösterilmektedir. Şekil B.23’de parkinsonlu bireyin sağ ayađına ait kuvvet sinyali için Daubechies6 dalgacıđına ait 6. seviyeye kadar detay katsayıları gösterilmektedir.

Daubechies7 dalgacıđı kullanılarak oluşturulan öznitelikler içerisinde sadece 168 öznitelik seçilmiştir. Bu özniteliklerin listesi Çizelge B.18’de gösterilmektedir. Şekil B.24’de parkinsonlu bireyin sağ ayađına ait kuvvet sinyali için Daubechies7 dalgacıđına ait 6. seviyeye kadar detay katsayıları gösterilmektedir.

Sinyaller üzerinde uygulanan son dönüşüm yöntemi HHD'dir. HHD ile öznelik oluşturmadan önce kuvvet sinyallerinin Welch Güç Spektral Yoğunluk Tahminleri hesaplanmış ve bu yoğunluk tahminlerinden öznelikler oluşturulmuştur. Şekil 3.8'de Parkinsonlu bireye ait sağ ayak sinyalinin Welch güç spektral yoğunluk tahmini gösterilmektedir. HHD ile bir sinyal İMF'lerine ayrıştırılır. Şekil B.26'te parkinsonlu bir bireye ait sağ ayak kuvvet sinyalinin ilk 6 İMF'si gösterilmektedir. Şekil B.27'te bu ilk 6 İMF'ye ait genlik değerleri yer almaktadır. Daha sonra bu İMF'ler kullanılarak AF'ler elde edilir. Şekil B.28'te bu ilk 6 İMF'ye ait AF'ler gösterilmektedir. Her bir denek için 477 adet sinyal ve bu sinyallerin HHD'si kullanılarak elde edilen öznelik sayısı 694512'dir. Bu öznelikleri elde etmek için Şekil B.25'deki MATLAB kodları yazılmıştır. Daha sonra *OneRAttributeEval* algoritması kullanılarak (3.1)'deki koşula göre 694512 öznelik içerisinde sadece 135 öznelik seçilmiştir. Bu özneliklerin listesi Çizelge B.22'te gösterilmektedir.



Şekil 3.8: Parkinsonlu bireye ait sağ ayak sinyalinin Welch güç spektral yoğunluk tahmini

Sinyal dönüşüm yöntemleri kullanılarak elde edilen öznelikler ve bu öznelikler içerisinde yüksek enformasyon veren özneliklerin seçilmesi işleminin tamamlanmasının ardından WEKA aracının kullanabileceği formatta eğitim kümeleri oluşturulur. Bu dosyaları oluşturmak için yazılan MATLAB kodları Şekil B.29'da gösterilmektedir. VIBES ve WEKA'da yer alan diğer Makine Öğrenmesi

algoritmalarının parkinson veri kümesi üzerindeki karşılaştırmalı sonuçları 4.2. bölümde gösterilmektedir. Parkinson veri kümesine uygulanan tüm Makine Öğrenmesi algoritmaları aksi belirtilmedikçe 10-katlı çapraz doğrulama yöntemi kullanılarak uygulandı.



4. SONUÇLAR VE TARTIŞMA

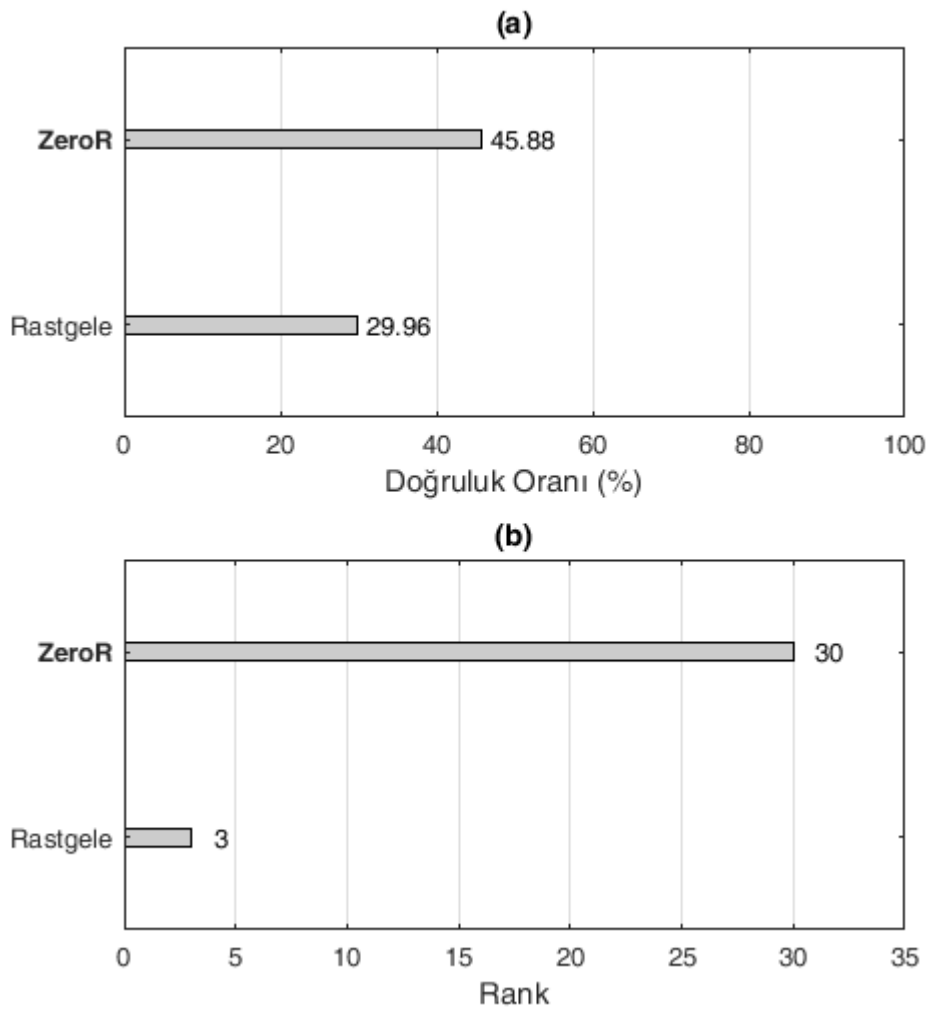
4.1 Benchmark Veri Kümeleri Üzerinde VIBES ve Diğer Makine Öğrenmesi Algoritmalarının Karşılaştırmalı Sonuçları

VIBES algoritmasını Parkinson veri kümesi üzerine uygulamadan önce benchmark veri kümeleri üzerinde diğer Makine Öğrenmesi algoritmaları ile birlikte uygulanıp karşılaştırılmıştır. Böylelikle *VIBES* algoritmasının sadece spesifik bir veri kümesine değil, aynı zamanda tüm diğer sınıflandırma problemlerine başarıyla uygulanabileceği sonuçlarıyla gösterilmiştir.

Benchmark veri kümeleri üzerine Makine Öğrenmesi algoritmalarının uygulanması ile ilgili deneyler WEKA 3.9.1 ortamında gerçekleştirilmiştir. Deneylerde kullanılan sınıflandırma algoritmaları WEKA'da 7 ana kategoriye (Bayes, functions, lazy, meta, misc, rules ve trees) ayrılmıştır. Bu tez çalışmasında da sınıflandırma algoritmaları bu kategoriler altında karşılaştırılmıştır. Ayrıca benchmark veri kümeleri üzerine uygulanan tüm Makine Öğrenmesi algoritmaları için varsayılan parametreler kullanıldı. WEKA'da herbir Makine Öğrenmesi algoritması için bu parametre değerleri sınıflandırma problemleri üzerinde ortalaması alınmış değerlerdir.

Bir veri kümesi üzerine uygulanan herhangi bir Makine Öğrenmesi algoritmasının en temel iki tahmin ediciden daha iyi tahminler üretmesi beklenir (Kılıçaslan et al. 2009). Bu tahmin ediciler: *Çoğunluk tahmin edicisi (ZeroR)* ve *rastgele tahmin edicidir*. Eğer bir Makine Öğrenmesi algoritması bu iki tahmin ediciyle aynı ya da daha düşük doğrulukta tahminler veriyorsa bu Makine Öğrenmesi algoritmasının yapmış olduğu tahminler kabul edilebilir değildir. Aynı zamanda bir veri kümesi üzerinde yapılabilecek en kötü tahmini rastgele tahmin edici yapmaktadır. Bu tahmin ediciden sonraki en iyi tahmin edici *ZeroR*'dur. Tüm bu nedenlerle yapılan deneylerde bu iki tahmin edicinin yapmış oldukları tahminler diğer Makine Öğrenmesi algoritmalarının tahminleriyle karşılaştırılmıştır. 33 adet veri kümesinden oluşan benchmark veri kümelerinin tümü üzerinde bu iki temel sınıflandırıcı uygulanmıştır. Şekil 4.1'de bu iki tahmin edicinin sonuçları gösterilmektedir. Şekil

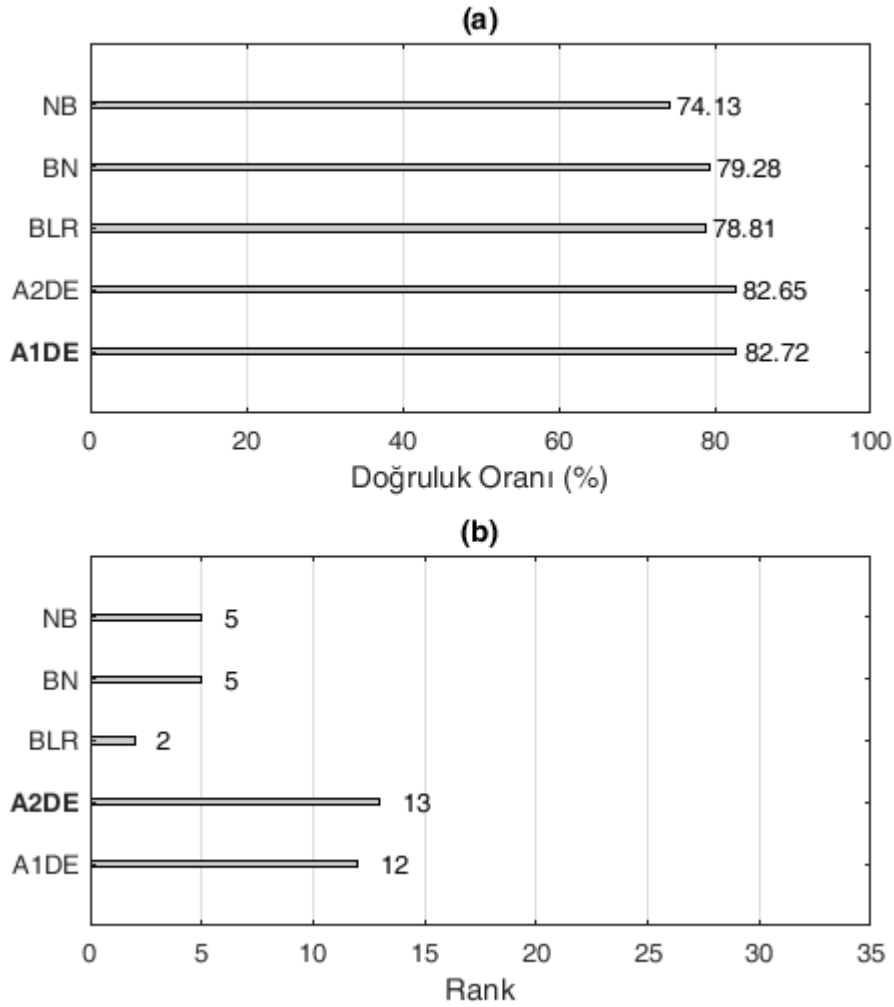
4.1 (a)'da bu iki tahmin edicinin benchmark veri kümeleri üzerindeki doğruluk oranlarının ortalaması gösterilmektedir. Bu sonuca göre ve daha öncede belirttiğimiz gibi *ZeroR*, rastgele tahmin ediciye göre daha yüksek sınıflandırma doğruluk oranına sahiptir. Şekil 4.1 (b)'de ise bu iki temel sınıflandırıcının 33 adet benchmark veri kümesinin kaç tanesinde en yüksek tahminde bulunduğu sayısı (rank) gösterilmektedir (Eğer algoritmalar için en yüksek sınıflandırma doğruluğu aynı ise her bir algoritmanın rank değeri bir arttırılır). Bu iki baseline tahmin edicinin benchmark veri kümeleri üzerindeki ayrıntılı sınıflandırma sonuçları Çizelge C.2'de gösterilmektedir.



Şekil 4.1: Baseline algoritmaların benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu ve (b) rank sonuçları

İlk olarak Bayesian Öğrenme yaklaşımına sahip sınıflandırıcılar kendi aralarında benchmark veri kümeleri üzerine uygulandı. Şekil 4.2 (a)'da bu algoritmalarından en

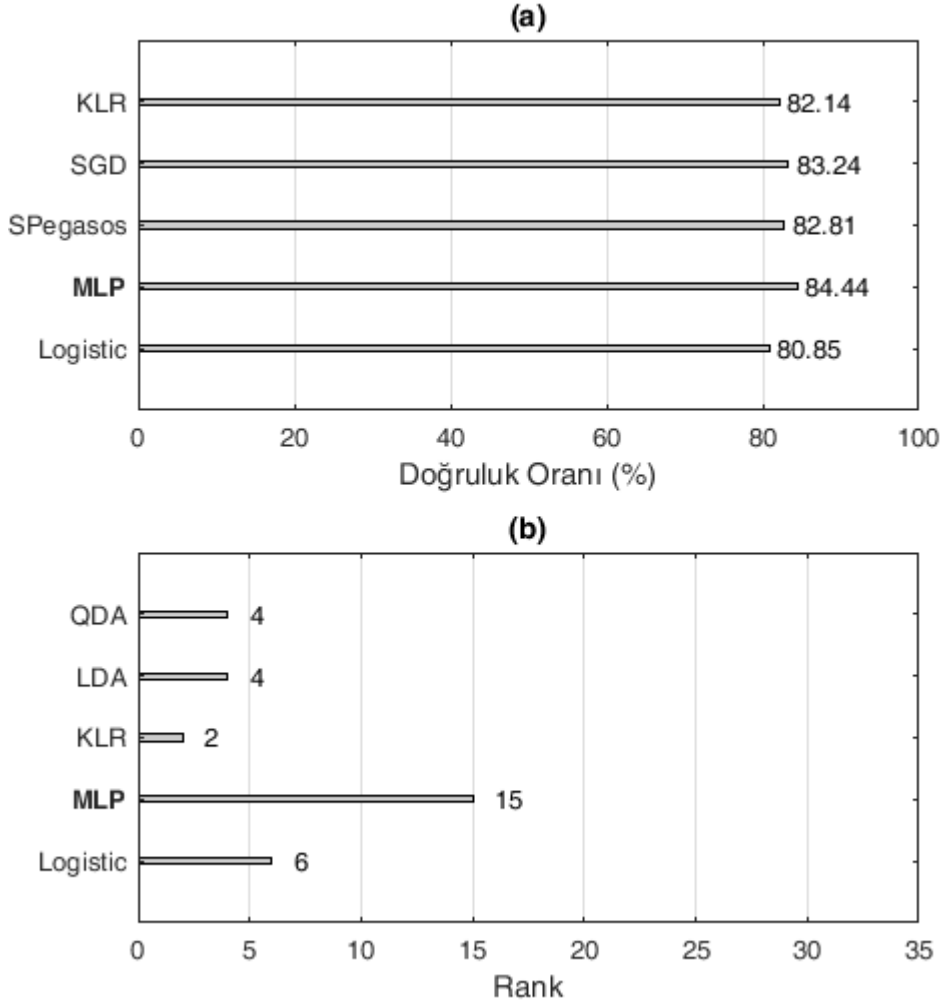
yüksek ortalama doğruluk oranına sahip ilk beş algoritma gösterilmektedir. Buna göre en yüksek sınıflandırma doğruluk oranını *Aggregating One-Dependence Estimators (A1DE)* algoritması verdi. Şekil 4.2 (b)'de ise benchmark veri kümeleri üzerindeki en yüksek ranka sahip ilk beş algoritma gösterilmektedir. Buna göre en yüksek rank değerini *Aggregating Two-Dependence Estimators (A2DE)* algoritması verdi. Bayesian Öğrenme algoritmalarının benchmark veri kümeleri üzerindeki ayrıntılı sınıflandırma sonuçları Çizelge C.2'de gösterilmektedir.



Şekil 4.2: Bayesian öğrenme algoritmalarının benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu en yüksek ilk beş algoritma ve (b) rank sonuçları en yüksek ilk beş algoritma

İkinci olarak Fonksiyon Öğrenme yaklaşımına sahip sınıflandırıcılar kendi aralarında benchmark veri kümeleri üzerine uygulandı. Şekil 4.3 (a)'da bu algoritmalarından en yüksek ortalama doğruluk oranına sahip ilk beş algoritma gösterilmektedir. Buna göre en yüksek sınıflandırma doğruluk oranını *Multilayer Perceptron (MLP)*

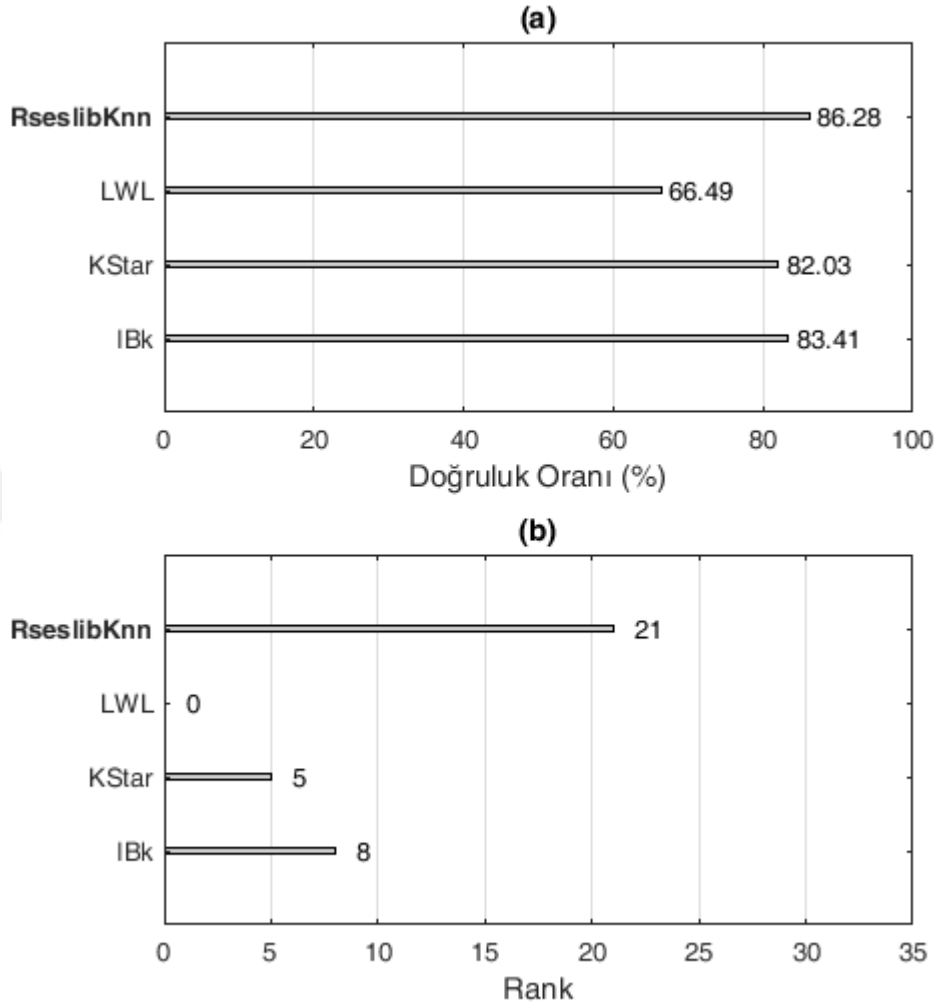
algoritması verdi. Şekil 4.3 (b)'de ise benchmark veri kümeleri üzerindeki en yüksek ranka sahip ilk beş algoritma gösterilmektedir. Buna göre en yüksek rank değerini yine *MLP* algoritması verdi. Fonksiyon Öğrenme algoritmalarının benchmark veri kümeleri üzerindeki ayrıntılı sınıflandırma sonuçları Çizelge C.3'te gösterilmektedir.



Şekil 4.3: Fonksiyon öğrenme algoritmalarının benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu en yüksek ilk beş algoritma ve (b) rank sonuçları en yüksek ilk beş algoritma

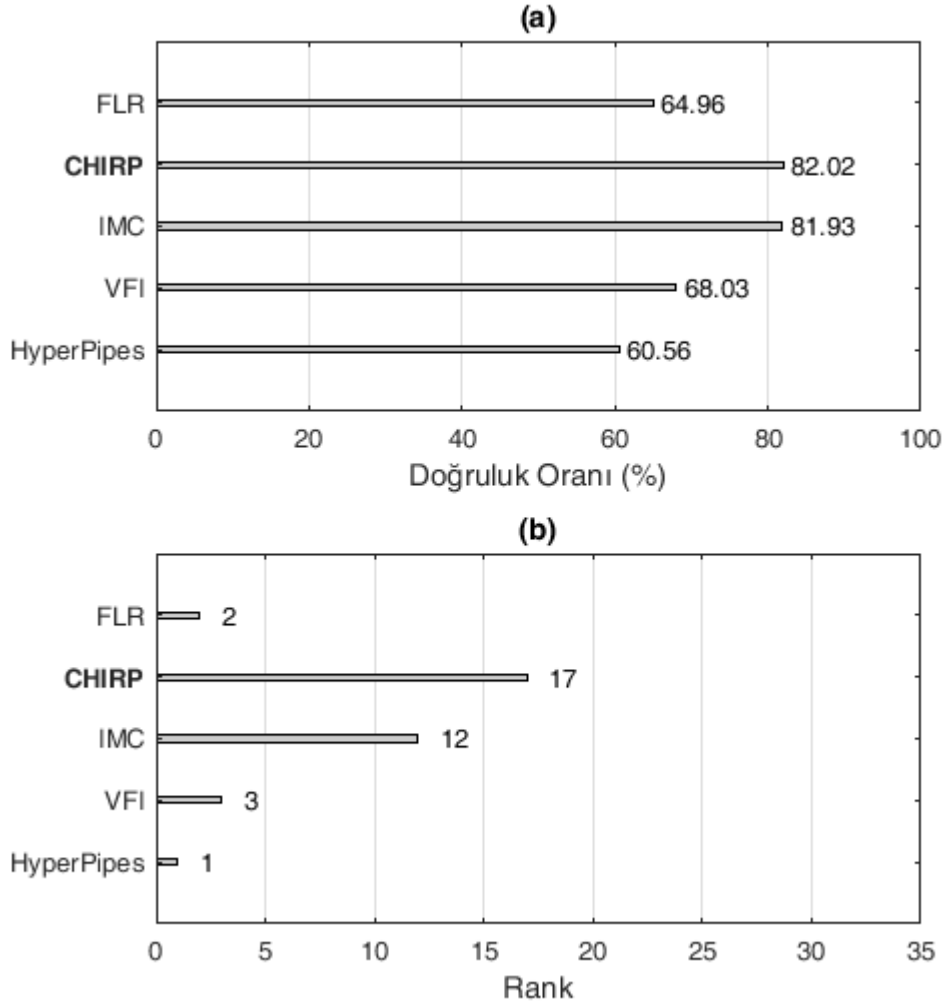
Üçüncü olarak Örnek Tabanlı Öğrenme yaklaşımına sahip sınıflandırıcılar kendi aralarında benchmark veri kümeleri üzerine uygulandı. Şekil 4.4 (a)'da bu algoritmalarından en yüksek ortalama doğruluk oranına sahip algoritmalar gösterilmektedir. Buna göre en yüksek sınıflandırma doğruluk oranını *RseslibKnn* algoritması verdi. Şekil 4.4 (b)'de ise benchmark veri kümeleri üzerindeki en yüksek ranka sahip algoritmalar gösterilmektedir. Buna göre en yüksek rank değerini yine *RseslibKnn* algoritması verdi. Örnek Tabanlı Öğrenme algoritmalarının benchmark

veri kümeleri üzerindeki ayrıntılı sınıflandırma sonuçları Çizelge C.4'te gösterilmektedir.



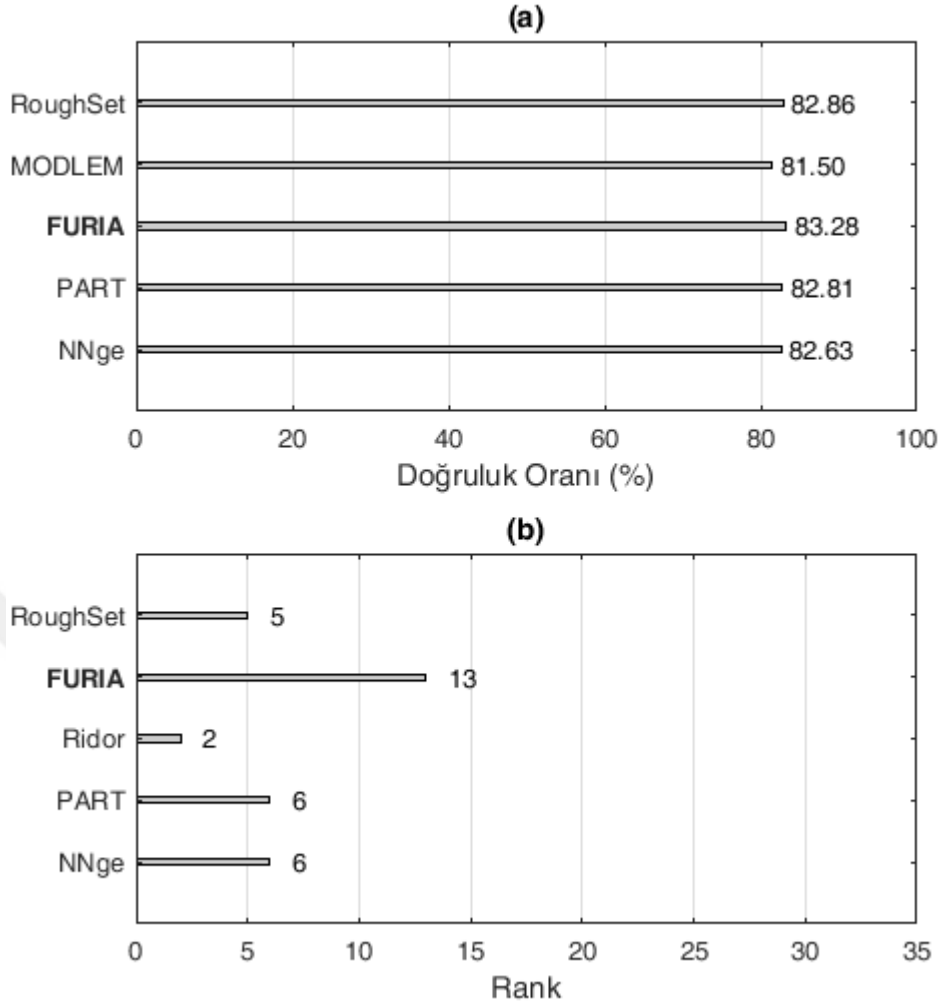
Şekil 4.4: Örnek tabanlı öğrenme algoritmalarının benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu ve (b) rank sonuçları

Dördüncü olarak çeşitli yaklaşımlara göre öğrenme gerçekleştiren sınıflandırıcılar kendi aralarında benchmark veri kümeleri üzerine uygulandı. Şekil 4.5 (a)'da bu algoritmalarından en yüksek ortalama doğruluk oranına sahip algoritmalar gösterilmektedir. Buna göre en yüksek sınıflandırma doğruluk oranını *Composite Hypercubes on Iterated Random Projections (CHIRP)* algoritması verdi. Şekil 4.5 (b)'de ise benchmark veri kümeleri üzerindeki en yüksek ranka sahip algoritmalar gösterilmektedir. Buna göre en yüksek rank değerini yine *CHIRP* algoritması verdi. Şekil 4.5'deki algoritmaların benchmark veri kümeleri üzerindeki ayrıntılı sınıflandırma sonuçları Çizelge C.5'te gösterilmektedir.



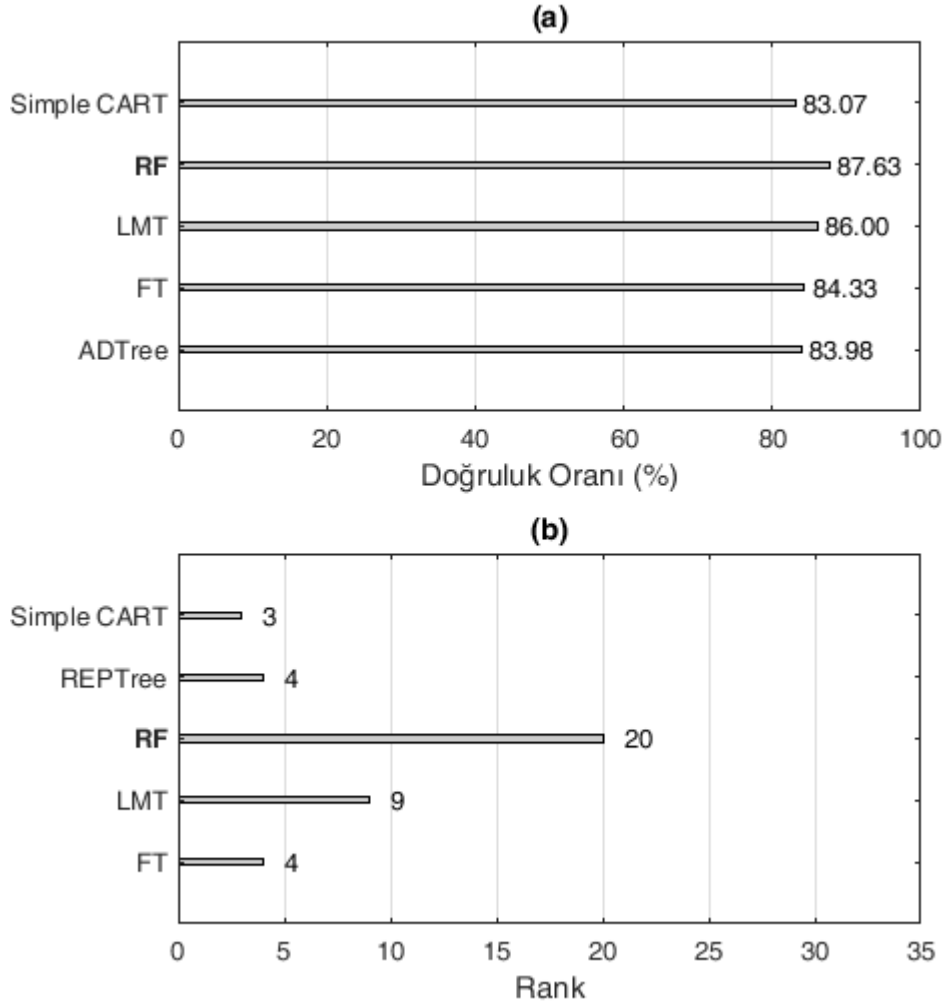
Şekil 4.5: Çeşitli yaklaşımlara göre öğrenme gerçekleştiren algoritmaların benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu ve (b) rank sonuçları

Beşinci olarak Kural Tabanlı Öğrenme yaklaşımına sahip sınıflandırıcılar kendi aralarında benchmark veri kümeleri üzerine uygulandı. Şekil 4.6 (a)'da bu algoritmalardan en yüksek ortalama doğruluk oranına sahip ilk beş algoritma gösterilmektedir. Buna göre en yüksek sınıflandırma doğruluk oranını *Fuzzy Unordered Rule Induction Algorithm (FURIA)* algoritması verdi. Şekil 4.6 (b)'de ise benchmark veri kümeleri üzerindeki en yüksek ranka sahip ilk beş algoritma gösterilmektedir. Buna göre en yüksek rank değerini yine *FURIA* algoritması verdi. Kural Tabanlı Öğrenme algoritmalarının benchmark veri kümeleri üzerindeki ayrıntılı sınıflandırma sonuçları Çizelge C.6'da gösterilmektedir.



Şekil 4.6: Kural tabanlı öğrenme algoritmalarının benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu en yüksek ilk beş algoritma ve (b) rank sonuçları en yüksek ilk beş algoritma

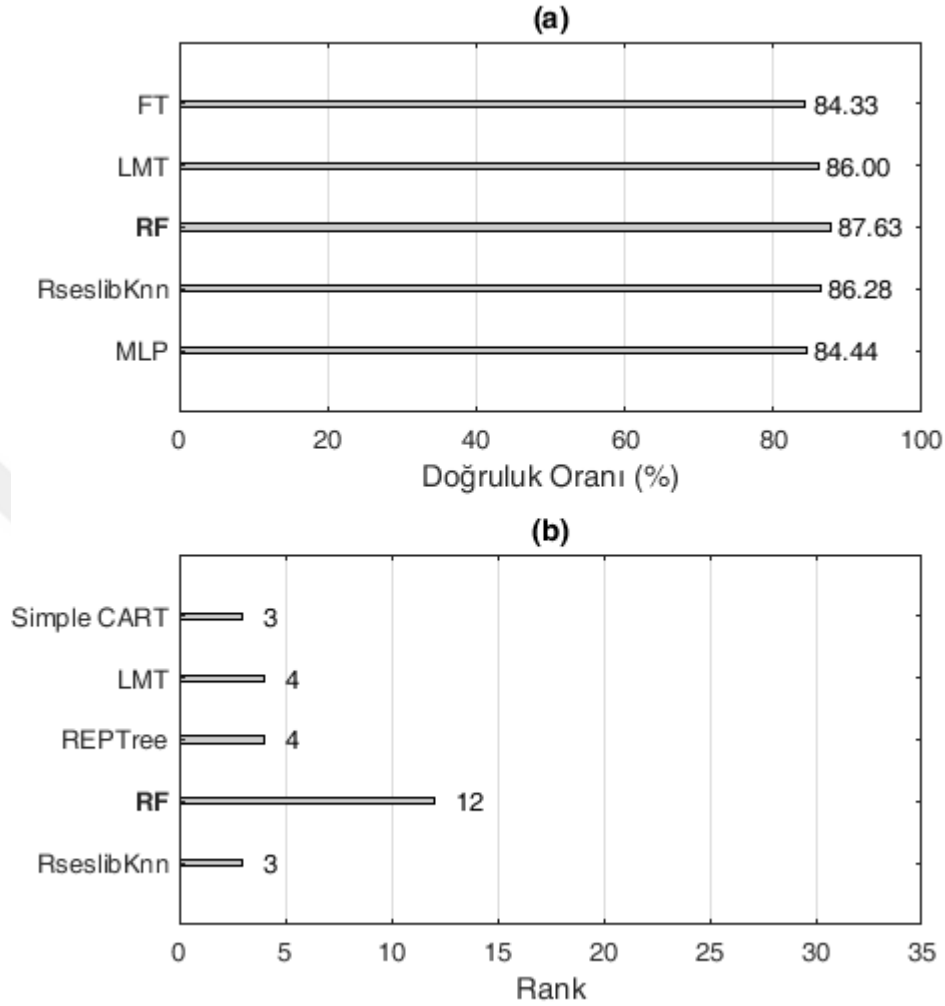
Altıncı olarak Karar Ağacı Öğrenme yaklaşımına sahip sınıflandırıcılar kendi aralarında benchmark veri kümeleri üzerine uygulandı. Şekil 4.7 (a)'da bu algoritmalarından en yüksek ortalama doğruluk oranına sahip ilk beş algoritma gösterilmektedir. Buna göre en yüksek sınıflandırma doğruluk oranını *Random Forests (RF)* algoritması verdi. Şekil 4.7 (b)'de ise benchmark veri kümeleri üzerindeki en yüksek ranka sahip ilk beş algoritma gösterilmektedir. Buna göre en yüksek rank değerini yine *RF* algoritması verdi. Karar Ağacı Öğrenme algoritmalarının benchmark veri kümeleri üzerindeki ayrıntılı sınıflandırma sonuçları Çizelge C.7'de gösterilmektedir.



Şekil 4.7: Karar ağacı öğrenme algoritmalarının benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu en yüksek ilk beş algoritma ve (b) rank sonuçları en yüksek ilk beş algoritma

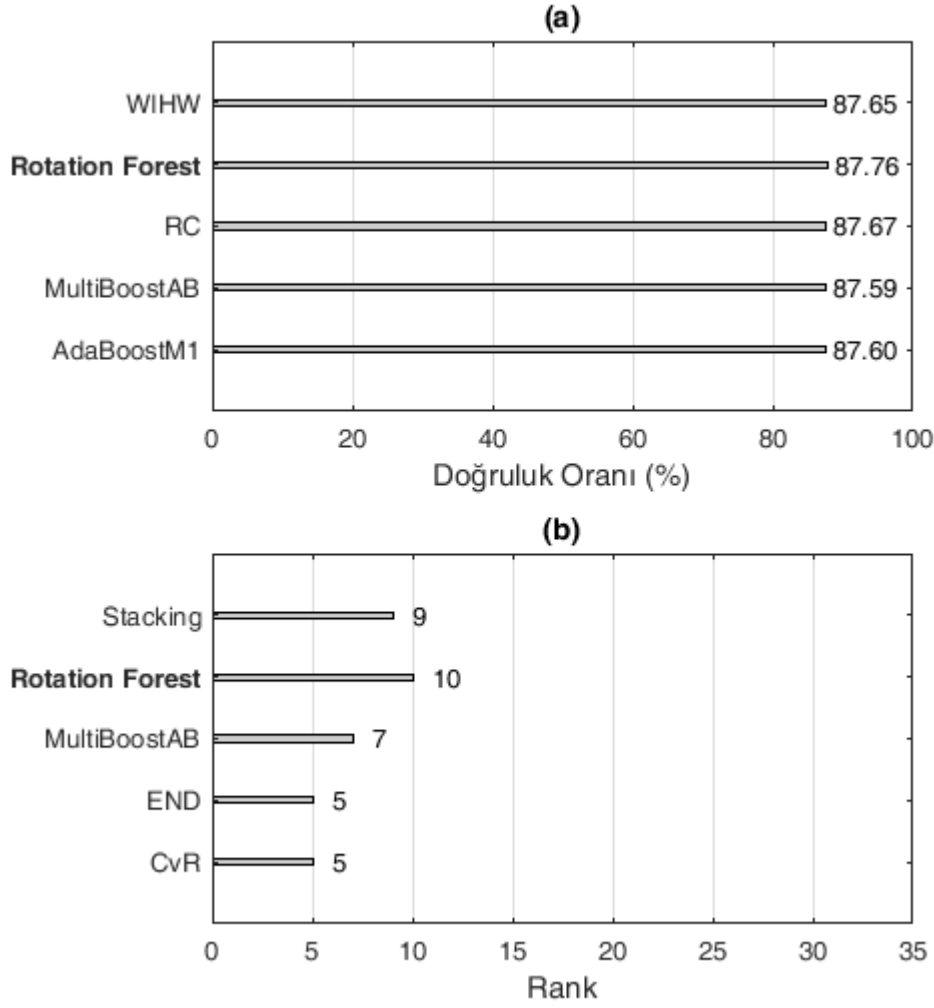
Şu ana kadar yapılan deneylerde temel öğrenciler öğrenme yaklaşımlarına göre kendi aralarında karşılaştırıldı. Şimdi ise herbir öğrenme yaklaşımı içerisinde en yüksek sınıflandırma doğruluğuna sahip sınıflandırıcılar karşılaştırılacak. Şekil 4.8’de bu sınıflandırıcıların (sadece temel öğrenciler) karşılaştırılması gösterilmektedir. Şekil 4.8 (a)’daki sonuçlara göre bu sınıflandırıcılar içerisinde Karar Ağacı Öğrenme yaklaşımını benimseyen *RF* sınıflandırıcı en yüksek ortalama sınıflandırma doğruluğuna sahiptir. Aynı zamanda Şekil 4.8 (b)’deki sonuçlara göre *RF* sınıflandırıcı rank bakımından da en yüksek değere sahiptir. Bu nedenle Topluluk Öğrenme ya da Meta Öğrenme yaklaşımlarını içeren sınıflandırıcılarda benchmark veri kümeleri üzerinde temel öğrenci olarak *RF* sınıflandırıcı kullanılacaktır. Şekil

4.8'deki sonuçların ayrıntılı karşılaştırmalı sonuçları Çizelge C.8 ve Çizelge C.9'da gösterilmektedir.



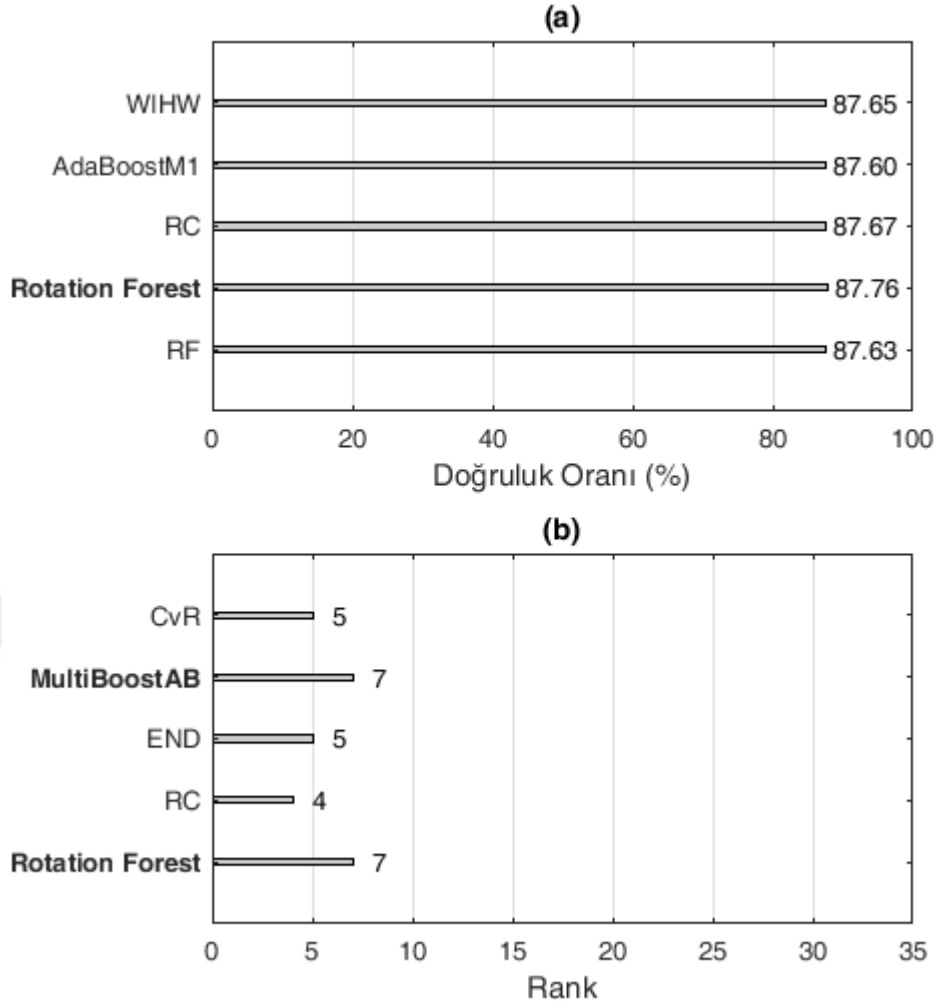
Şekil 4.8: Makine Öğrenmesi algoritmalarının (sadece temel öğrenciler) benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu en yüksek ilk beş algoritma ve (b) rank sonuçları en yüksek ilk beş algoritma

Son olarak Topluluk/Meta Öğrenme yaklaşımına sahip sınıflandırıcılar kendi aralarında benchmark veri kümeleri üzerine uygulandı. Şekil 4.9 (a)'da bu algoritmalarından en yüksek ortalama doğruluk oranına sahip ilk beş algoritma gösterilmektedir. Buna göre en yüksek sınıflandırma doğruluk oranını *Rotation Forest* algoritması verdi. Şekil 4.9 (b)'de ise benchmark veri kümeleri üzerindeki en yüksek ranka sahip ilk beş algoritma gösterilmektedir. Buna göre en yüksek rank değerini yine *Rotation Forest* algoritması verdi. Topluluk/Meta Öğrenme algoritmalarının benchmark veri kümeleri üzerindeki ayrıntılı sınıflandırma sonuçları Çizelge C.10, Çizelge C.11 ve Çizelge C.12'de gösterilmektedir.



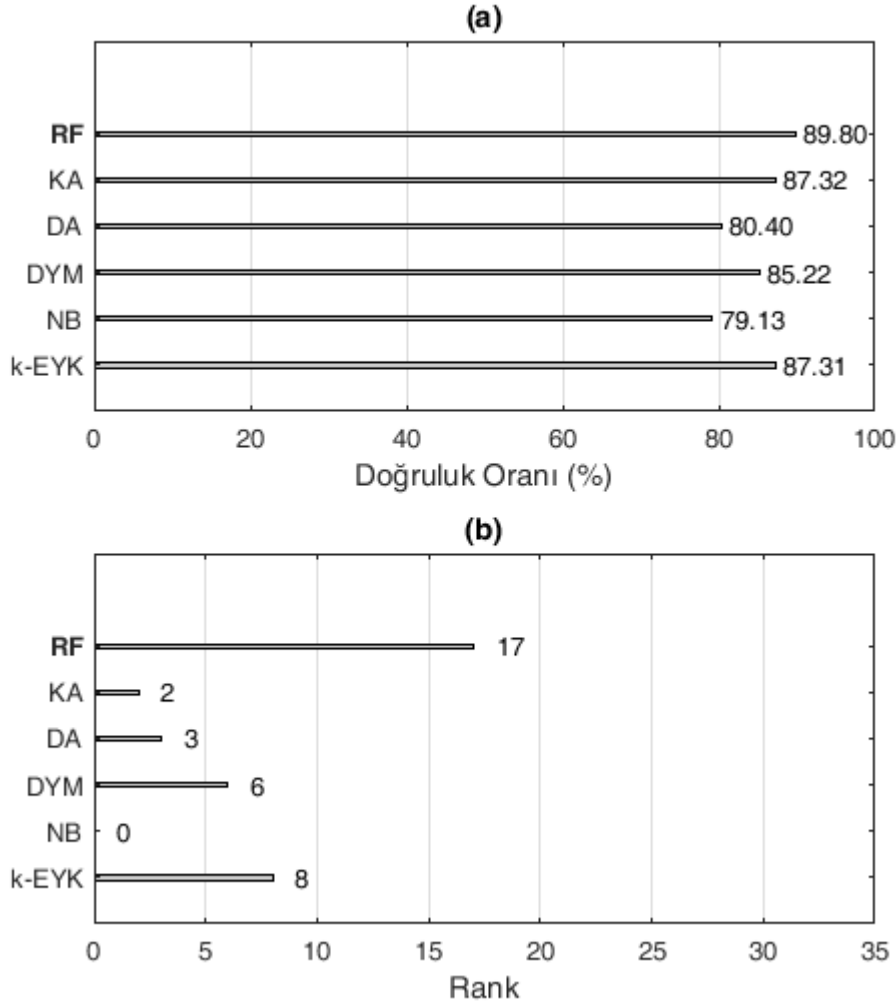
Şekil 4.9: Topluluk Öğrenme algoritmalarının benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu en yüksek ilk beş algoritma ve (b) rank sonuçları en yüksek ilk beş algoritma

Son olarak WEKA’da yer alan tüm Makine Öğrenmesi algoritmaları (88 algoritma) birbirleriyle karşılaştırıldı. Karşılaştırma sonuçları Şekil 4.10’da gösterilmektedir. Bu karşılaştırmaya *VIBES* algoritması dahil edilmedi. *VIBES* algoritması daha sonra bu algoritmalarla da kıyaslanmaktadır. Şekil 4.10 (a)’daki sonuçlara göre benchmark veri kümeleri üzerinde *Rotation Forest* algoritması en yüksek ortalama doğruluğa sahip algoritmadır. Şekil 4.10 (b)’de ise en yüksek rank değerine sahip iki algoritma vardır. Bu algoritmalar: *Rotation Forest* ve *MultiBoostAB*’dir. Benchmark veri kümeleri üzerinde tüm Makine Öğrenmesi algoritmalarının ayrıntılı karşılaştırma sonuçları Çizelge C.13, Çizelge C.14 ve Çizelge C.15’te gösterilmektedir.



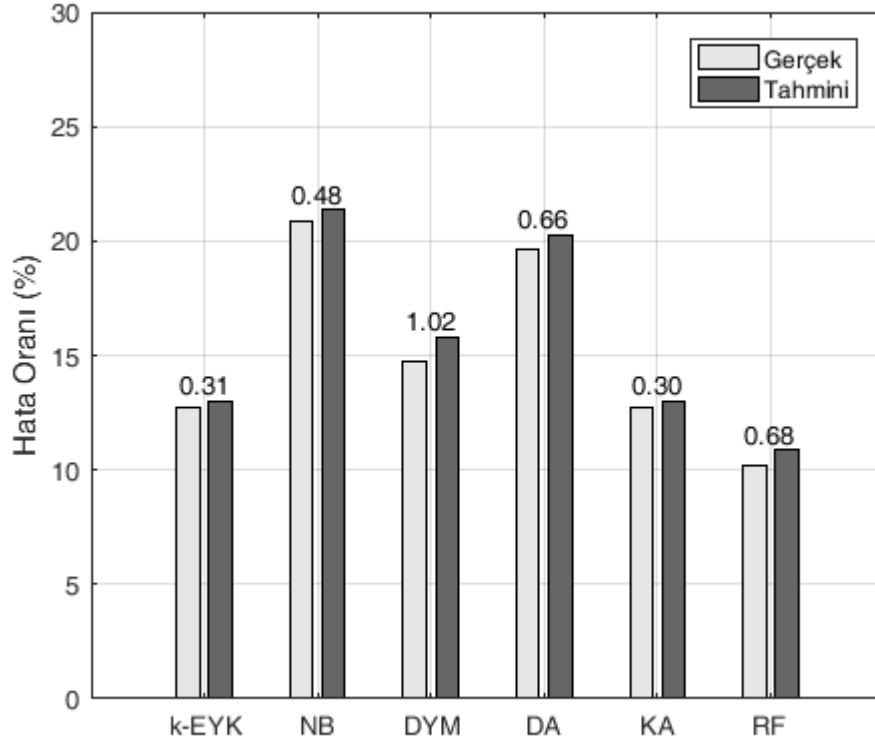
Şekil 4.10: WEKA'daki tüm Makine Öğrenmesi algoritmalarının (*VIBES* hariç) benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu en yüksek ilk beş algoritma ve (b) rank sonuçları en yüksek ilk beş algoritma

VIBES algoritması iki varsayım üzerine çalışmaktadır: Özniteliklerin birbirlerine bağımlı olduğu ve bağımsız olduğu'dur. Benchmark veri kümeleri üzerine *VIBES* algoritmasını uygularken her iki varsayıma göre deneyler tasarlayacağız. İlk deneyimiz özniteliklerin bağımlılığı varsayımına göre gerçekleştirildi. Bu deneyin sonuçları Şekil 4.11'de gösterilmektedir. Bu sonuçlara göre temel öğrenici *RF* seçildiğinde *VIBES* algoritmasının ortalama sınıflandırma doğruluğu %89.80 ve rank değeri 17 ile en yüksek olmaktadır. Ayrıca diğer Topluluk Öğrenme algoritmalarında da temel öğrenici olarak *RF* seçilmişti. Şekil 4.11'deki sonuçların ayrıntılı listesi Çizelge C.16'da gösterilmektedir.



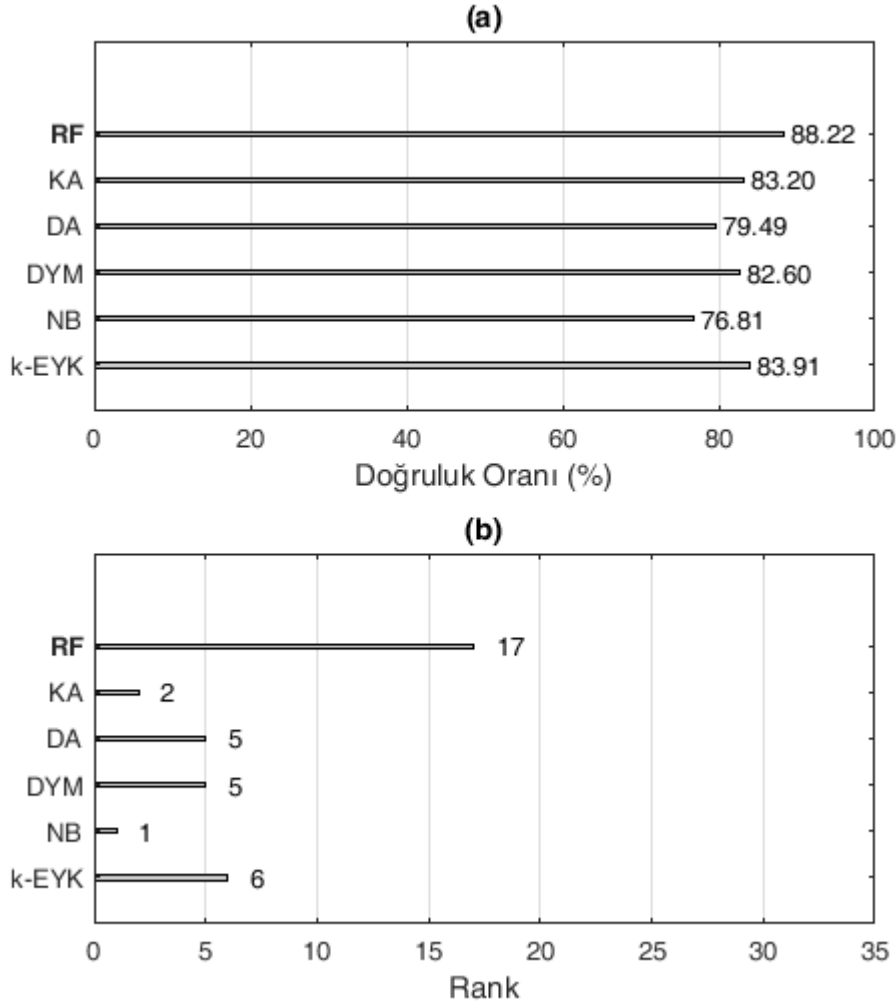
Şekil 4.11: Özniteliklerin bağımlılığı varsayımına göre *VIBES* algoritmasının seçilen temel öğrenicilere göre benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu değerleri ve (b) rank sonuçları

Çoğunluk Oylamalı Topluluk Öğrenicilerde topluluğun tahmini ve gerçek hatası arasındaki farkın düşük olması önemlidir. Geliştirmiş olduğumuz çoğunluk oylamalı sınıflandırıcılarda hata oranı tahmin algoritması gerçek sonuçlara oldukça yakın değerler vermiştir. Özniteliklerin bağımlılığı varsayımına göre *VIBES* algoritmasının temel öğrenicilere göre tahmini ve gerçek hata oranları arasındaki fark oldukça düşüktür. Bu karşılaştırmalara ilişkin sonuçlar Şekil 4.12’de gösterilmektedir. Buna göre ortalama hata oranı farkı %0.575’tir. Bu sonuca göre hata tahmin algoritması oldukça iyi çalışmaktadır. Bu hata değerlerinin herbir benchmark veri kümesine göre ayrıntılı listesi Çizelge C.17’de gösterilmektedir.



Şekil 4.12: Özniteliklerin bağımlılığı varsayımına göre *VIBES* algoritmasının seçilen temel öğrencilere göre benchmark veri kümeleri üzerindeki ortalama gerçek ve tahmini hata oranları

Özniteliklerin bağımsızlığı varsayımına göre gerçekleştirilen deneyin sonuçları Şekil 4.13'te gösterilmektedir. Bu sonuçlara göre temel öğrenci *RF* seçildiğinde *VIBES* algoritmasının ortalama sınıflandırma doğruluğu %88.22 ve rank değeri 17 ile en yüksek olmaktadır. Ayrıca Şekil 4.13'deki sonuçların ayrıntılı listesi Çizelge C.18'de gösterilmektedir.

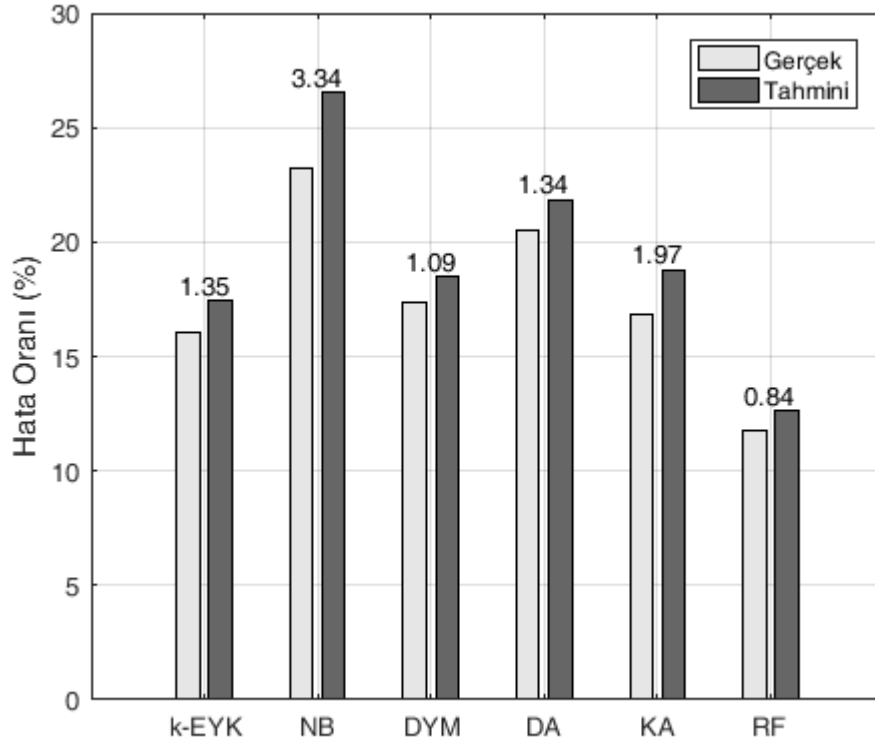


Şekil 4.13: Özniteliklerin bağımsızlığı varsayımına göre *VIBES* algoritmasının seçilen temel öğrencilere göre benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu değerleri ve (b) rank sonuçları

Özniteliklerin bağımsızlığı varsayımına göre *VIBES* algoritmasının temel öğrencilere göre tahmini ve gerçek hata oranları arasındaki fark oldukça düşüktür. Bu karşılaştırmalara ilişkin sonuçlar Şekil 4.13'te gösterilmektedir. Buna göre ortalama hata oranı farkı %1.655'tir. Bu sonuca göre hata tahmin algoritması oldukça iyi çalışmaktadır. Bu hata değerlerinin her bir benchmark veri kümesine göre ayrıntılı listesi Çizelge C.19'da gösterilmektedir.

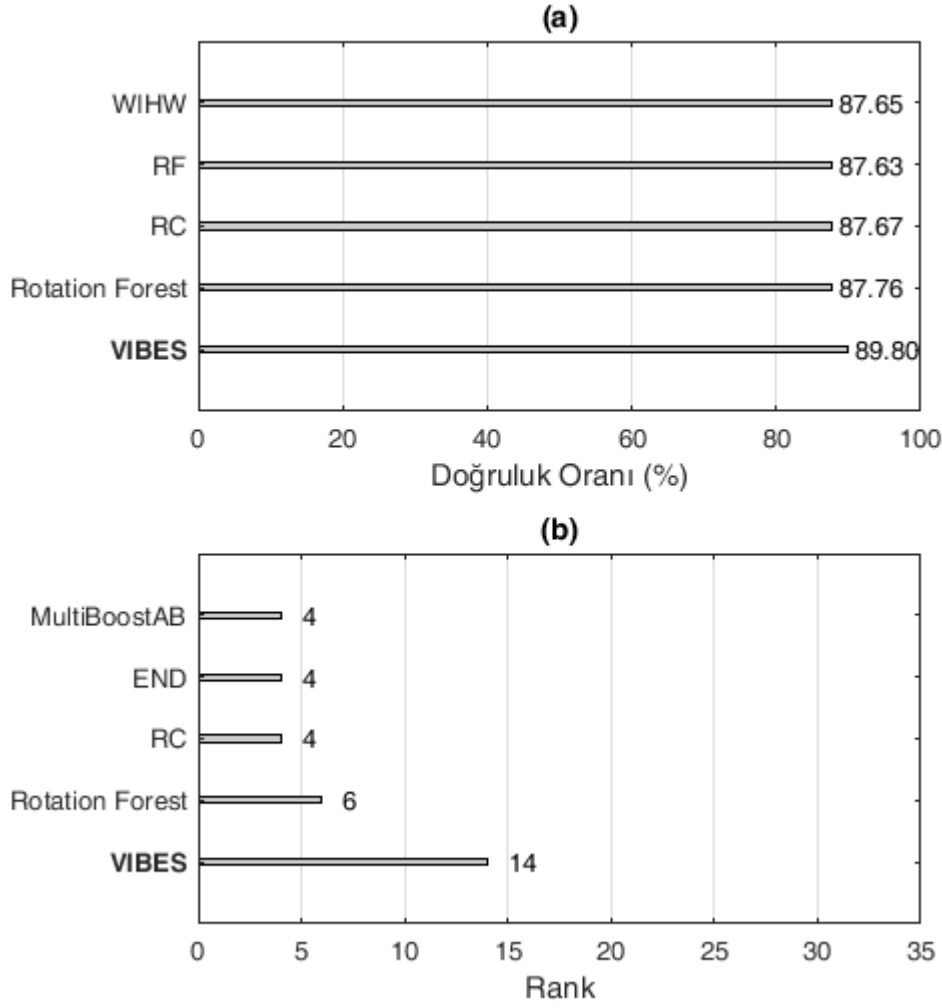
Her iki varsayıma göre hata oranı farklarını kıyasladığımızda doğruluk oranı yükseldikçe hata oranı farkının düştüğü görülmektedir. Sınıflandırma doğruluğu düştükçe de hata oranı farkının yükseldiği görülmektedir. Sonuç olarak her iki varsayıma göre ortalama hata oranı farkı %1.115 olmaktadır. Bu da kabul edilebilir bir sonuç olarak karşımıza çıkmaktadır. Ayrıca tahmin değerinin gerçek hata

değerine göre her zaman büyük çıkması bir üst sınır olması açısından önemlidir. Böylelikle bir topluluğun gerçek hatasının her zaman tahmini hatadan daha düşük olacağı söylenebilecektir.



Şekil 4.14: Özniteliklerin bağımsızlığı varsayımına göre *VIBES* algoritmasının seçilen temel öğrenicilere göre benchmark veri kümeleri üzerindeki ortalama gerçek ve tahmini hata oranları

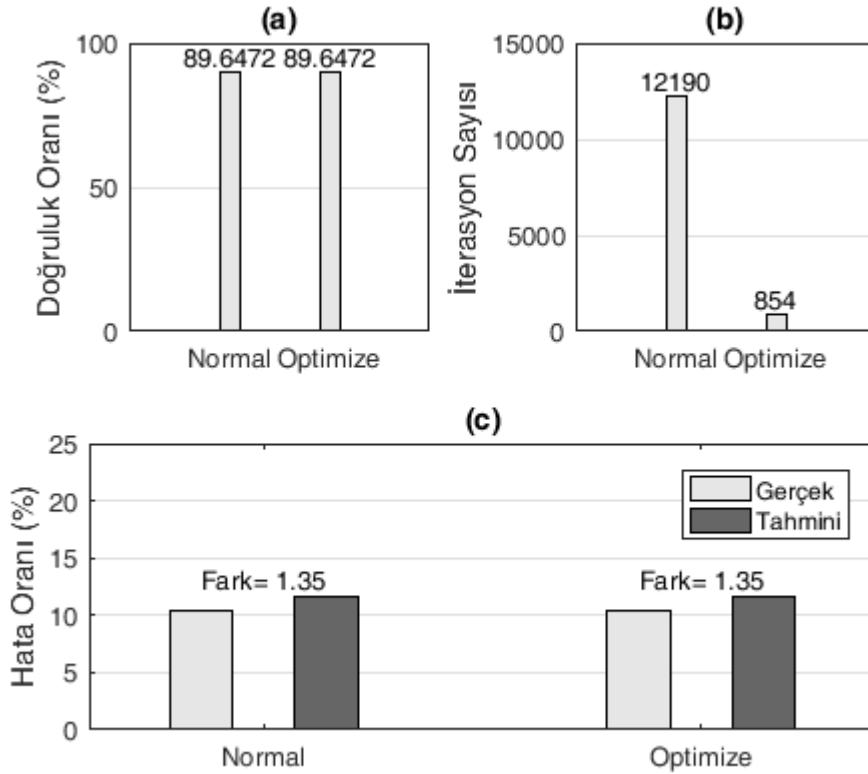
Son olarak WEKA’da yer alan tüm Makine Öğrenmesi algoritmaları (88 algoritma) ile *VIBES* algoritması karşılaştırıldı. Karşılaştırma sonuçları Şekil 4.15’de gösterilmektedir. Şekil 4.15 (a)’daki sonuçlara göre benchmark veri kümeleri üzerinde *VIBES* algoritması %89.80 ile en yüksek ortalama doğruluğa sahip algoritmadır. Şekil 4.15 (b)’de ise en yüksek rank değerine sahip algoritma 14 rank değeri ile yine *VIBES* algoritması olmuştur. Benchmark veri kümeleri üzerinde *VIBES* algoritması dahil tüm Makine Öğrenmesi algoritmalarının ayrıntılı karşılaştırma sonuçları Çizelge C.20, Çizelge C.21 ve Çizelge C.22’de gösterilmektedir. Tüm bu sonuçlara göre *VIBES* algoritması ve en yakın rakibi olan Rotation Forest algoritması arasındaki fark %2.04’tür. Bu sonuç sınıflandırma doğruluğu açısından önemli bir sonuçtur.



Şekil 4.15: Tüm Makine Öğrenmesi algoritmalarının (VIBES dahil) benchmark veri kümeleri üzerindeki (a) ortalama sınıflandırma doğruluğu en yüksek ilk beş algoritma ve (b) rank sonuçları en yüksek ilk beş algoritma

VIBES algoritmasında topluluğu oluştururken veri kümesinin boyutuna bağlı olarak ciddi bir zaman maliyeti oluşmaktadır. Bu nedenle topluluğu oluştururken sınıflandırma doğruluğunu düşürmeden ileri yönlü aramayı uygun bir noktada kesmek önemlidir. Bu açıdan ileri yönlü aramayı optimize etmek için arama algoritmasında değişiklikler gerçekleştirdik. Bu değişiklikler optimize edilmiş ve normal (optimize edilmemiş) VIBES algoritmasının karşılaştırma sonuçları Şekil 4.16'da gösterilmektedir. Bu sonuçlara göre Şekil 4.16 (a)'da gösterildiği gibi sınıflandırma doğruluğu değişmeden Şekil 4.16 (b)'de gösterildiği gibi ortalama iterasyon sayısı yaklaşık olarak 14 kat düşürülmüştür. Bu durum yüksek boyutlu veri kümeleri için topluluğu hızlı bir biçimde oluşturmamıza olanak vermektedir. Ayrıca Şekil 4.16 (c)'de tahmini ve gerçek hata oranı arasındaki farkta aynı kalmaktadır.

Böylelikle sınıflandırma doğruluğunu değiştirmeden zaman maliyetini düşürmüş olduk. Şekil 4.16'daki sonuçlar özniteliklerin bağımlılığı varsayımına göre yapılmış olsa da özniteliklerin bağımsızlığı varsayımı için de sonuç aynı olacaktır. Yani sınıflandırma doğruluğu değişmezken topluluğu oluşturan zaman maliyeti düşmektedir. VIBES algoritmasının benchmark veri kümeleri üzerinde optimize edilmiş ve edilmemiş versiyonlarının ayrıntılı karşılaştırma sonuçları Çizelge C.23'te gösterilmektedir.

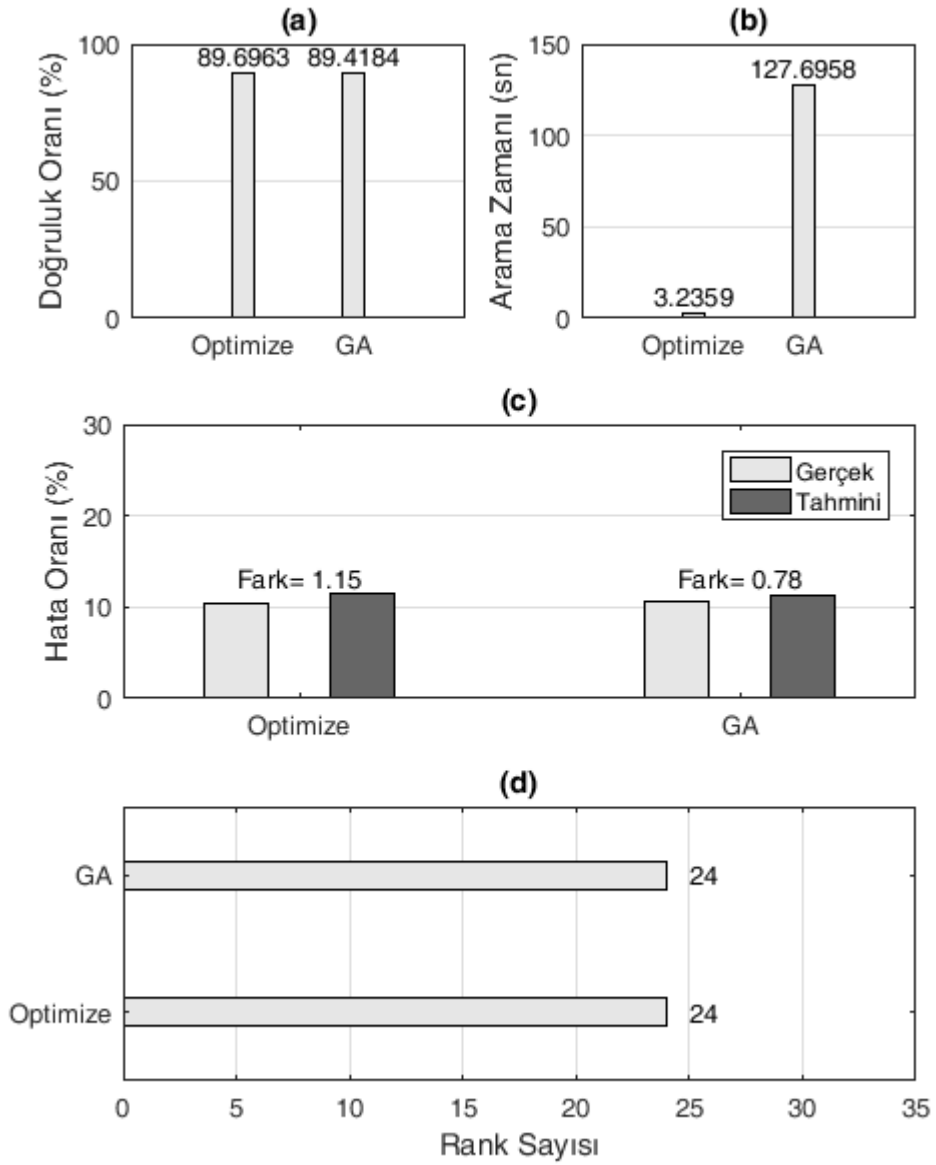


Şekil 4.16: Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının (Temel Öğrenici olarak RF algoritması seçildi) topluluk oluşturma yöntemi için kullandığı ileri yönlü arama ve bu aramanın optimize edilmiş halinin (a) Ortalama doğruluk oranına (b) Ortalama iterasyon sayısına ve (c) Ortalama gerçek ve tahmini hata oranına göre karşılaştırma sonuçları

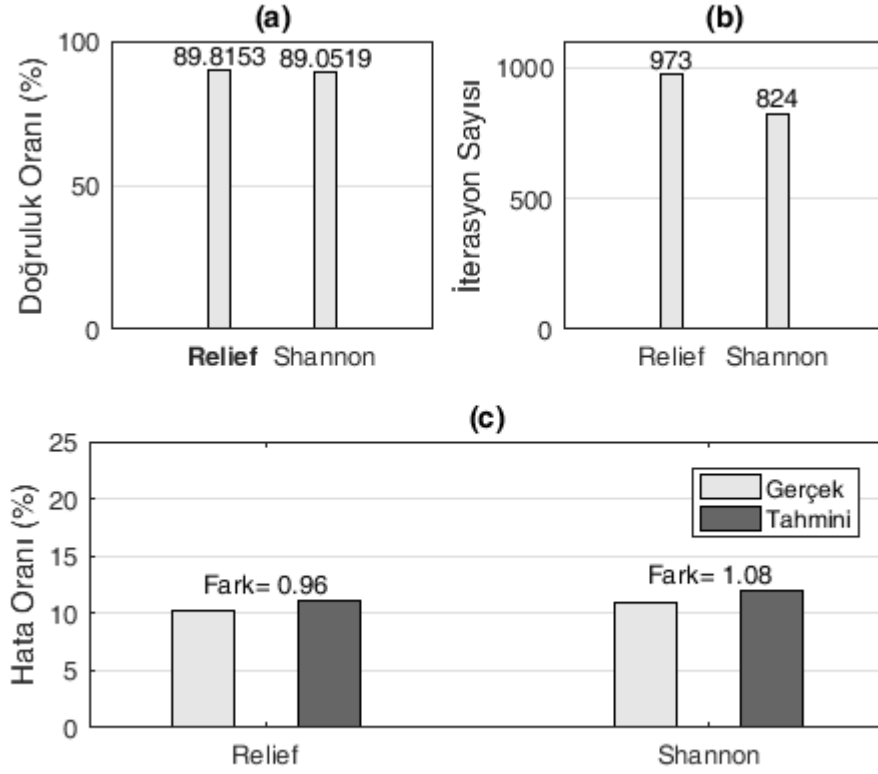
VIBES algoritmasının topluluk oluştururken kullandığı arama yöntemleri: Optimize edilmiş ileri yönlü arama ve Genetik Algoritmalar (GA)'dır. Bu iki arama yöntemine göre yapılan deneylerin sonuçları Şekil 4.17'de gösterilmektedir. Şekil 4.17 (a)'daki sonuçlara göre ileri yönlü arama, GA'ya göre bir nebze daha fazla doğru sınıflandırma oranına sahiptir. Aradaki fark %0.2779'dur. Bu fark önemli bir fark değildir. Yalnız Şekil 4.17 (b)'deki sonuçlara göre optimize edilmiş ileri yönlü

arama, GA'ya göre zaman maliyeti açısından oldukça iyidir. Buna göre optimize edilmiş ileri yönlü arama GA'ya göre yaklaşık 39 kat daha hızlıdır. Şekil 4.17 (c)'de optimize edilmiş ileri yönlü arama ile GA için gerçek ve tahmini hata oranları gösterilmektedir. Bu fark değerlerinin, yukarıda belirtilen %1.115 değeri ile uyumlu olduğu görülmektedir. Şekil 4.17 (d)'de ise optimize edilmiş ileri yönlü aramayla GA ile yapılan arama sonucunda oluşturulmuş modellerin rank değerleri karşılaştırılmıştır. Bu sonuçlara göre benchmark veri kümeleri üzerinde eşit sayıda rank değeri elde edilmiştir. Sonuç olarak optimize edilmiş ileri yönlü arama ile GA arasında sınıflandırma doğruluğu açısından bir fark olmadığı görülmektedir. Yalnız optimize edilmiş ileri yönlü aramanın zaman maliyeti açısından daha iyi olduğu açıktır. Bu nedenle topluluk oluşturmada optimize edilmiş ileri yönlü arama algoritması daha etkilidir. Bu iki arama yöntemine göre benchmark veri kümeleri üzerindeki ayrıntılı sonuçlar Çizelge C.24'te gösterilmektedir.

Özniteliklerin bağımlılığı varsayımına göre VIBES algoritması ilk önce özniteliklerin enformasyon miktarını hesaplar. Enformasyon miktarını hesaplarken iki yöntem kullanılır. Bu yöntemler: ReliefF ve Shannon entropi'dir. Şekil 4.18'de bu iki öznitelik enformasyon hesaplama yönteminin sonuçları gösterilmektedir. Şekil 4.18 (a)'da her iki yöntemin sınıflandırma doğruluklarının karşılaştırması gösterilmektedir. Buna göre ReliefF algoritmasına göre yapılan sınıflandırma doğruluğu Shannon entropiye göre %0.7634 oranında daha yüksek doğruluk oranına sahiptir. Şekil 4.18 (b)'de her iki yöntem optimize edilmiş ileri yönlü aramada gerçekleştirilen iterasyon sayısına göre karşılaştırılmışlardır. Buna göre ReliefF algoritması ortalama iterasyon sayısı bakımından Shannon entropiye göre 149 iterasyon daha fazla çalışmıştır. Yalnız bu değer istatistiksel açıdan çok önemli değildir. Şekil 4.18 (c)'de ReliefF ile Shannon entropi için gerçek ve tahmini hata oranları gösterilmektedir. Bu fark değerlerinin, yukarıda belirtilen %1.115 değeri ile uyumlu olduğu görülmektedir. Yani gerçek ve tahmini hata oranı arasındaki fark oldukça düşüktür. Bu iki enformasyon ölçüm yöntemine göre benchmark veri kümeleri üzerindeki ayrıntılı sonuçlar Çizelge C.25'te gösterilmektedir. Sonuç olarak ReliefF ve Shannon entropi algoritmaları istatistiksel açıdan birbirlerine oldukça yakın sonuçlar vermektedir. Bu nedenle sınıflandırma problemlerine göre her iki algoritmayı da denemek gerekmektedir.



Şekil 4.17: Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının (Temel Öğrenici olarak RF algoritması seçildi) topluluk oluşturma yöntemi için kullandığı optimize edilmiş ileri yönlü arama ve Genetik Algoritmalar kullanılarak yapılan aramanın (a) Ortalama doğruluk oranına (b) Ortalama arama zamanına (c) Ortalama gerçek ve tahmini hata oranına ve (d) Rank sayısına göre karşılaştırma sonuçları



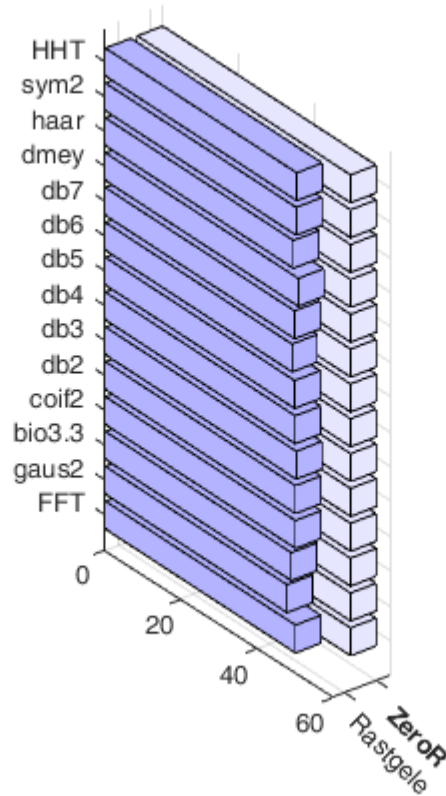
Şekil 4.18: Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasında (Temel Öğrenici olarak RF algoritması seçildi) özniteliklerin enformasyonunu ölçmek için kullanılan Relief algoritması ve Shannon entropinin (a) Ortalama doğruluk oranına (b) İterasyon sayısına ve (c) Ortalama gerçek ve tahmini hata oranına göre karşılaştırma sonuçları

Sonuç olarak VIBES algoritması benchmark veri kümeleri üzerinde diğer 88 Makine Öğrenmesi algoritmasına göre en yüksek sınıflandırma doğruluğuna sahiptir. Ayrıca VIBES algoritmasının bazı parametre değerleri yapılan deneyler sonucunda daha iyi sonuç vermiştir. Bu parametre değerleri VIBES algoritmasının varsayılan parametreleri olarak ayarlandı. Buna göre VIBES algoritmasının varsayılan parametre değerleri: Özniteliklerin Durumu: Öznitelikler birbirlerine bağımlı, Arama Algoritması: Optimize edilmiş ileri yönlü arama, Enformasyon Ölçümü: ReliefF olarak belirlenmiştir.

4.2 Parkinson Veri Kümesi Üzerine Uygulanan Makine Öğrenmesi Algoritmalarının Sonuçları

Parkinson veri kümesine Makine Öğrenmesi algoritmaları uygulanmadan önce iki temel tahmin edici (Rastgele tahmin edici ve ZeroR) bu veri kümelerine uygulandı. İki temel tahmin edicinin sonuçları Şekil 4.19'da gösterilmektedir. Ayrıca ayrıntılı

sonular izelge C.26’da gsterilmektedir. Bu sonulara gre sinyal dnsm uygulanmıř tm Parkinson veri kmelerinde ZeroR sınıflandırıcı, rastgele tahmin ediciye gre daha yksek sınıflandırma doęruluęuna sahiptir. Bu durum iki tahmin edicinin doęasına uygun bir sonutur. Dięer Makine ęrenmesi algoritmalarının bu veri kmeleri zerindeki sınıflandırma doęruluęunun bu iki tahmin edicinin tahminlerinden daha yksek olması gerekmektedir. Dięer trl yapılan tahminler kabul edilebilir olmayacaktır.



řekil 4.19: İki temel tahmin edicinin eřitli sinyal dnsmleri uygulanmıř olan Parkinson veri kmesi zerindeki sınıflandırma doęruluęu oranları

4.2.1 Parkinson veri kmesine uygulanan WEKA’daki sınıflandırıcıların karřılařtırmalı sonuları

WEKA’da yer alan 91 adet Makine ęrenmesi algoritması, 0-1 normalizasyonu uygulanmıř Parkinson veri kmesi zerine tatbik edilmiřtir. Makine ęrenmesi algoritmaları bu veri kmelerine uygulanmadan nce znitelik seim sreci gerekleřtirildi. znitelik seimi iin WEKA’da yer alan WrapperSubsetEval znitelik seim yntemi kullanıldı (WEKA’da verinin tm eęitim kmesi olarak seildi. Yani apraz doęrulama gerekleřtirilmedi. Ancak WrapperSubsetEval

algoritması 10-katlı çapraz doğrulama yöntemini kullandığı içintekrardan çapraz doğrulama yapılmasına gerek görülmemiştir). Bu yöntem, seçilen Makine Öğrenmesinin en yüksek oranda doğruluk verdiği ve en düşük hata oranına sahip olduğu öznitelik kümesini seçmektedir. Bu yöntem öznitelik seçimi için çeşitli yöntemler kullanmaktadır. Bu tez çalışmasında öznitelik arama yöntemi olarak BestFirst algoritması tercih edildi. Bu seçimin yapılmasında BestFirst algoritmasının daha yüksek doğruluk veren öznitelik kümelerini oluşturması etkili oldu. Ayrıca arama işlemi için ileri yönlü arama gerçekleştirildi. Sarmalama (wrapper) yaklaşımı seçilen Makine Öğrenmesi için en yüksek doğrulukta öznitelik kümesini oluşturmasına rağmen zaman maliyeti açısından oldukça kötüdür. Özellikle eğitim süreci uzun süren algoritmalar için sarmalama yöntemi oldukça zaman harcamaktadır. Bu durumda genel yaklaşımımız; ilgili Parkinson veri kümesi üzerinde en yüksek doğruluğu veren öznitelik kümesinin, eğitim ve arama zamanı uzun süren Makine Öğrenmesi algoritmaları için de kullanılması yönündedir.

Benchmark veri kümelerinde olduğu gibi Parkinson veri kümeleri için de Makine Öğrenmesi algoritmaları öğrenme yaklaşımlarına göre karşılaştırıldı. WEKA'da Makine Öğrenmesi algoritmaları 7 ana kategoriye ayrılmaktadır. Bunlar: Bayes, functions, lazy, meta, misc, rules ve trees'dir.

Parkinson veri kümelerine ilk uygulanan Makine Öğrenmesi algoritmaları Bayesian öğrenme yaklaşımına sahip algoritmalarlardır. Bu algoritmalar içerisinde Bayesian Network (BN) algoritması HHD uygulanmış Parkinson veri kümesi üzerinde %93.94 doğruluk oranıyla en yüksek sınıflandırma doğruluğuna sahiptir. Aynı zamanda Aggregating One-Dependence Estimators (A1DE) algoritması Daubechies3 dalgacığı uygulanmış Parkinson veri kümesi üzerinde %93.94 doğruluk oranıyla en yüksek sınıflandırma doğruluğuna sahiptir. Diğer Bayesian Öğrenme yaklaşımına sahip algoritmalar Parkinson veri kümeleri üzerinde daha düşük sınıflandırma doğruluğu elde etmişlerdir. Bu sonuçlarla ilgili ayrıntılar Çizelge C.27'de gösterilmektedir.

Parkinson veri kümelerine ikinci uygulanan Makine Öğrenmesi algoritmaları Fonksiyon öğrenme yaklaşımına sahip algoritmalarlardır. Bu algoritmalar içerisinde Linear Discriminant Analysis (LDA) algoritması Daubechies3 dalgacığı uygulanmış Parkinson veri kümesi üzerinde %93.33 doğruluk oranıyla en yüksek sınıflandırma doğruluğuna sahiptir. Diğer Fonksiyon Öğrenme yaklaşımına sahip algoritmalar

Parkinson veri kümeleri üzerinde daha düşük sınıflandırma doğruluğu elde etmişlerdir. Bu sonuçlarla ilgili ayrıntılar Çizelge C.28’de gösterilmektedir.

Parkinson veri kümelerine üçüncü uygulanan Makine Öğrenmesi algoritmaları Örnek öğrenme yaklaşımına sahip algoritmalarıdır. Bu algoritmalar içerisinde Instance Based-k (IBk) algoritması HHD uygulanmış Parkinson veri kümesi üzerinde %91.52 doğruluk oranıyla en yüksek sınıflandırma doğruluğuna sahiptir. Aynı zamanda IBk algoritması Daubechies4 dalgacığı uygulanmış Parkinson veri kümesi üzerinde %91.52 doğruluk oranıyla en yüksek sınıflandırma doğruluğuna sahiptir. Diğer Örnek Öğrenme yaklaşımına sahip algoritmalar Parkinson veri kümeleri üzerinde daha düşük sınıflandırma doğruluğu elde etmişlerdir. Bu sonuçlarla ilgili ayrıntılar Çizelge C.29’da gösterilmektedir.

Parkinson veri kümelerine dördüncü uygulanan Makine Öğrenmesi algoritmaları çeşitli öğrenme yaklaşımlarına sahip algoritmalarıdır. Bu algoritmalar içerisinde HyperPipes algoritması Daubechies4 dalgacığı uygulanmış Parkinson veri kümesi üzerinde %89.70 doğruluk oranıyla en yüksek sınıflandırma doğruluğuna sahiptir. Diğer çeşitli öğrenme yaklaşımlarına sahip algoritmalar Parkinson veri kümeleri üzerinde daha düşük sınıflandırma doğruluğu elde etmişlerdir. Bu sonuçlarla ilgili ayrıntılar Çizelge C.30’da gösterilmektedir.

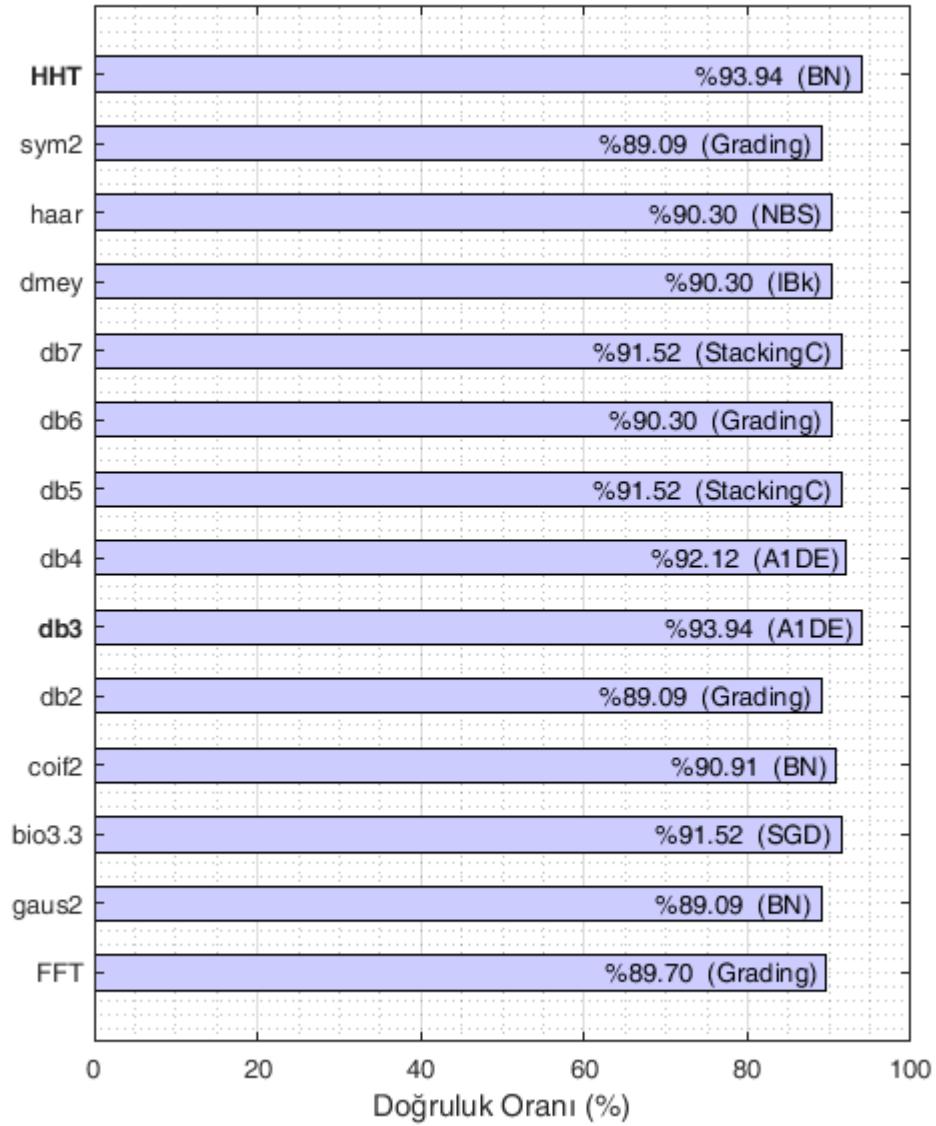
Parkinson veri kümelerine beşinci uygulanan Makine Öğrenmesi algoritmaları Kural Tabanlı Öğrenme yaklaşımına sahip algoritmalarıdır. Bu algoritmalar içerisinde Decision Table algoritması HHD uygulanmış Parkinson veri kümesi üzerinde %90.30 doğruluk oranıyla en yüksek sınıflandırma doğruluğuna sahiptir. Diğer Kural Tabanlı Öğrenme yaklaşımına sahip algoritmalar Parkinson veri kümeleri üzerinde daha düşük sınıflandırma doğruluğu elde etmişlerdir. Bu sonuçlarla ilgili ayrıntılar Çizelge C.31’de gösterilmektedir.

Parkinson veri kümelerine altıncı uygulanan Makine Öğrenmesi algoritmaları Karar Ağacı Öğrenme yaklaşımına sahip algoritmalarıdır. Bu algoritmalar içerisinde J48 (C4.5) algoritması Daubechies3 dalgacığı uygulanmış Parkinson veri kümesi üzerinde %90.30 doğruluk oranıyla en yüksek sınıflandırma doğruluğuna sahiptir. Diğer Karar Ağacı Öğrenme yaklaşımına sahip algoritmalar Parkinson veri kümeleri üzerinde daha düşük sınıflandırma doğruluğu elde etmişlerdir. Bu sonuçlarla ilgili ayrıntılar Çizelge C.32’de gösterilmektedir.

Parkinson veri kümelerine uygulanan Temel Öğreniciler içerisinde en yüksek doğruluk oranına BN ve A1DE algoritmaları sahiptir. Bayesian Network (BN) algoritması HHD uygulanmış Parkinson veri kümesi üzerinde %93.94 doğruluk oranıyla en yüksek sınıflandırma doğruluğuna sahiptir. Aynı zamanda Aggregating One-Dependence Estimators (A1DE) algoritması Daubechies3 dalgacığın uygulanmış Parkinson veri kümesi üzerinde %93.94 doğruluk oranıyla en yüksek sınıflandırma doğruluğuna sahiptir. Diğer Temel Öğreniciler Parkinson veri kümeleri üzerinde daha düşük sınıflandırma doğruluğu elde etmişlerdir. Bu sonuçlarla ilgili ayrıntılar Çizelge C.33’de gösterilmektedir.

Parkinson veri kümelerine son uygulanan Makine Öğrenmesi algoritmaları Topluluk/Meta Öğrenme yaklaşımına sahip algoritmalarlardır. Bu algoritmalar içerisinde END, Filtered Classifier, MultiClass Classifier, OrdinalClass Classifier, Weighted Instances Handler Wrapper (WIHW), MultiScheme ve StackingC algoritmaları Daubechies3 dalgacığın uygulanmış Parkinson veri kümesi üzerinde %93.94 doğruluk oranıyla en yüksek sınıflandırma doğruluğuna sahiptirler. Ayrıca END, Filtered Classifier, MultiClass Classifier, OrdinalClass Classifier, WIHW ve Grading algoritmaları HHD uygulanmış Parkinson veri kümesi üzerinde %93.94 doğruluk oranıyla en yüksek sınıflandırma doğruluğuna sahiptirler. Diğer Topluluk/Meta Öğrenme yaklaşımına sahip algoritmalar Parkinson veri kümeleri üzerinde daha düşük sınıflandırma doğruluğu elde etmişlerdir. Bu sonuçlarla ilgili ayrıntılar Çizelge C.34 ve Çizelge C.35’de gösterilmektedir.

WEKA’da yer alan Makine Öğrenmesi algoritmalarının (91 algoritma) herbir Parkinson veri kümesi üzerinde en yüksek sınıflandırma doğruluğu ve bu doğruluk oranını elde eden Makine Öğrenmesi algoritması Şekil 4.20’de gösterilmektedir.



Şekil 4.20: WEKA'daki Makine Öğrenmesi algoritmalarının, her bir Parkinson veri kümesi üzerinde en yüksek sınıflandırma doğruluğu ve ilgili Makine Öğrenmesi algoritması

4.2.2 Bir yapay sinir ağı modelinin parkinson veri kümesi üzerindeki sonuçları

Parkinson veri kümeleri üzerinde Ölçeklenmiş Eşlenik Dereceli Geri Yayılım Ağı (Scaled conjugate gradient backpropagation) kullanılarak bir sinir ağı modeli oluşturulmaya çalışıldı. Sinir ağı modeli oluşturulmadan önce veri kümeleri normalize edildi. Ayrıca WEKA'da yer alan CfsSubsetEval algoritması ve BestFirst arama yöntemi kullanılarak öznelik seçimi gerçekleştirildi (Verinin tümü eğitim kümesi olarak seçildi. Yani çapraz doğrulama gerçekleştirilmedi. Ancak

algoritmanın kendisi çapraz doğrulama yöntemini kullanmaktadır). Sinir ağı modeli eğitilirken verinin %70'i eğitim, %15'i doğrulama ve %15'i de test için kullanılmıştır. Ayrıca herbir deney 30 kere tekrarlanarak ortalama sınıflandırma doğruluğu elde edilmiştir. Deneylerin birçok kez tekrarlanmasının nedeni her deneyde çapraz doğrulama yöntemi ile oluşturulan veri kümesinin farklı olmasıdır. Aynı zamanda her deneyde sinir ağının başlangıç ağırlıkları farklı olmaktadır. Bu nedenle kararlı bir model oluşturmak için deneyler 30 defa gerçekleştirildi. Böylelikle gerçek bir test kümesi ile karşılaşıldığında sınıflandırma doğruluğu yapılan deneylerin ortalama sonucuna yakın çıkabilecektir. Sinir ağı modeli oluşturulurken tüm parametreler varsayılan değerlerde bırakıldı. Yalnız gizli katman sayısı 10, 50, 100, 150, 200, 250, 300, 350 ve 400 olarak değiştirilerek sınıflandırma doğruluğu (test kümesi sonucu) ve diğer doğruluk oranları (eğitim kümesi ve doğrulama kümesi sonuçları) analiz edildi. Bu işlemlerin gerçekleştirilmesi için yazılan MATLAB kodu Şekil B.30'da gösterilmektedir.

FFD uygulanmış Parkinson veri kümesine, Ölçeklenmiş Eşlenik Dereceli Geri Yayılım Ağı uygulandığında ve ağ yapısında 200 gizli katman kullanıldığında sinir ağı modeli test kümesi için en yüksek doğruluk oranı olan ortalama %83.73 doğruluk oranı vermiştir. Gizli katman sayısının değişimine göre diğer doğruluk oranları Çizelge C.46'da gösterilmektedir.

Gaussian2 dalgacığı uygulanmış Parkinson veri kümesine, Ölçeklenmiş Eşlenik Dereceli Geri Yayılım Ağı uygulandığında ve ağ yapısında 250 gizli katman kullanıldığında sinir ağı modeli test kümesi için en yüksek doğruluk oranı olan ortalama %82.13 doğruluk oranı vermiştir. Gizli katman sayısının değişimine göre diğer doğruluk oranları Çizelge C.47'de gösterilmektedir.

BiorSplines3.3 dalgacığı uygulanmış Parkinson veri kümesine, Ölçeklenmiş Eşlenik Dereceli Geri Yayılım Ağı uygulandığında ve ağ yapısında 200 gizli katman kullanıldığında sinir ağı modeli test kümesi için en yüksek doğruluk oranı olan ortalama %83.73 doğruluk oranı vermiştir. Gizli katman sayısının değişimine göre diğer doğruluk oranları Çizelge C.48'de gösterilmektedir.

Coiflets2 dalgacığı uygulanmış Parkinson veri kümesine, Ölçeklenmiş Eşlenik Dereceli Geri Yayılım Ağı uygulandığında ve ağ yapısında 50 gizli katman kullanıldığında sinir ağı modeli test kümesi için en yüksek doğruluk oranı olan

ortalama %83.60 doğruluk oranı vermiştir. Gizli katman sayısının değişimine göre diğer doğruluk oranları Çizelge C.49'da gösterilmektedir.

Daubechies2 dalgacığı uygulanmış Parkinson veri kümesine, Ölçeklenmiş Eşlenik Dereceli Geri Yayılım Ağı uygulandığında ve ağ yapısında 350 gizli katman kullanıldığında sinir ağı modeli test kümesi için en yüksek doğruluk oranı olan ortalama %82.53 doğruluk oranı vermiştir. Gizli katman sayısının değişimine göre diğer doğruluk oranları Çizelge C.50'de gösterilmektedir.

Daubechies3 dalgacığı uygulanmış Parkinson veri kümesine, Ölçeklenmiş Eşlenik Dereceli Geri Yayılım Ağı uygulandığında ve ağ yapısında 10 gizli katman kullanıldığında sinir ağı modeli test kümesi için en yüksek doğruluk oranı olan ortalama %84.67 doğruluk oranı vermiştir. Gizli katman sayısının değişimine göre diğer doğruluk oranları Çizelge C.51'de gösterilmektedir.

Daubechies4 dalgacığı uygulanmış Parkinson veri kümesine, Ölçeklenmiş Eşlenik Dereceli Geri Yayılım Ağı uygulandığında ve ağ yapısında 200 gizli katman kullanıldığında sinir ağı modeli test kümesi için en yüksek doğruluk oranı olan ortalama %82.40 doğruluk oranı vermiştir. Gizli katman sayısının değişimine göre diğer doğruluk oranları Çizelge C.52'de gösterilmektedir.

Daubechies5 dalgacığı uygulanmış Parkinson veri kümesine, Ölçeklenmiş Eşlenik Dereceli Geri Yayılım Ağı uygulandığında ve ağ yapısında 150 gizli katman kullanıldığında sinir ağı modeli test kümesi için en yüksek doğruluk oranı olan ortalama %83.33 doğruluk oranı vermiştir. Gizli katman sayısının değişimine göre diğer doğruluk oranları Çizelge C.53'te gösterilmektedir.

Daubechies6 dalgacığı uygulanmış Parkinson veri kümesine, Ölçeklenmiş Eşlenik Dereceli Geri Yayılım Ağı uygulandığında ve ağ yapısında 50 gizli katman kullanıldığında sinir ağı modeli test kümesi için en yüksek doğruluk oranı olan ortalama %84.27 doğruluk oranı vermiştir. Gizli katman sayısının değişimine göre diğer doğruluk oranları Çizelge C.54'te gösterilmektedir.

Daubechies7 dalgacığı uygulanmış Parkinson veri kümesine, Ölçeklenmiş Eşlenik Dereceli Geri Yayılım Ağı uygulandığında ve ağ yapısında 100 gizli katman kullanıldığında sinir ağı modeli test kümesi için en yüksek doğruluk oranı olan ortalama %84.00 doğruluk oranı vermiştir. Gizli katman sayısının değişimine göre diğer doğruluk oranları Çizelge C.55'de gösterilmektedir.

Dmeyer dalgacıđı uygulanmıř Parkinson veri kümesine, Ölçeklenmiř Eřlenik Dereceli Geri Yayılım Ađı uygulandıđında ve ađ yapısında 300 gizli katman kullanıldıđında sinir ađı modeli test kümesi için en yüksek dođruluk oranı olan ortalama %85.07 dođruluk oranı vermiřtir. Gizli katman sayısının deđiřimine göre diđer dođruluk oranları Çizelge C.56'da gösterilmektedir.

Haar dalgacıđı uygulanmıř Parkinson veri kümesine, Ölçeklenmiř Eřlenik Dereceli Geri Yayılım Ađı uygulandıđında ve ađ yapısında 10 gizli katman kullanıldıđında sinir ađı modeli test kümesi için en yüksek dođruluk oranı olan ortalama %84.13 dođruluk oranı vermiřtir. Gizli katman sayısının deđiřimine göre diđer dođruluk oranları Çizelge C.57'de gösterilmektedir.

Symlets2 dalgacıđı uygulanmıř Parkinson veri kümesine, Ölçeklenmiř Eřlenik Dereceli Geri Yayılım Ađı uygulandıđında ve ađ yapısında 100 gizli katman kullanıldıđında sinir ađı modeli test kümesi için en yüksek dođruluk oranı olan ortalama %82.80 dođruluk oranı vermiřtir. Gizli katman sayısının deđiřimine göre diđer dođruluk oranları Çizelge C.58'de gösterilmektedir.

HHD uygulanmıř Parkinson veri kümesine, Ölçeklenmiř Eřlenik Dereceli Geri Yayılım Ađı uygulandıđında ve ađ yapısında 10 gizli katman kullanıldıđında sinir ađı modeli test kümesi için en yüksek dođruluk oranı olan ortalama %81.07 dođruluk oranı vermiřtir. Gizli katman sayısının deđiřimine göre diđer dođruluk oranları Çizelge C.59'da gösterilmektedir.

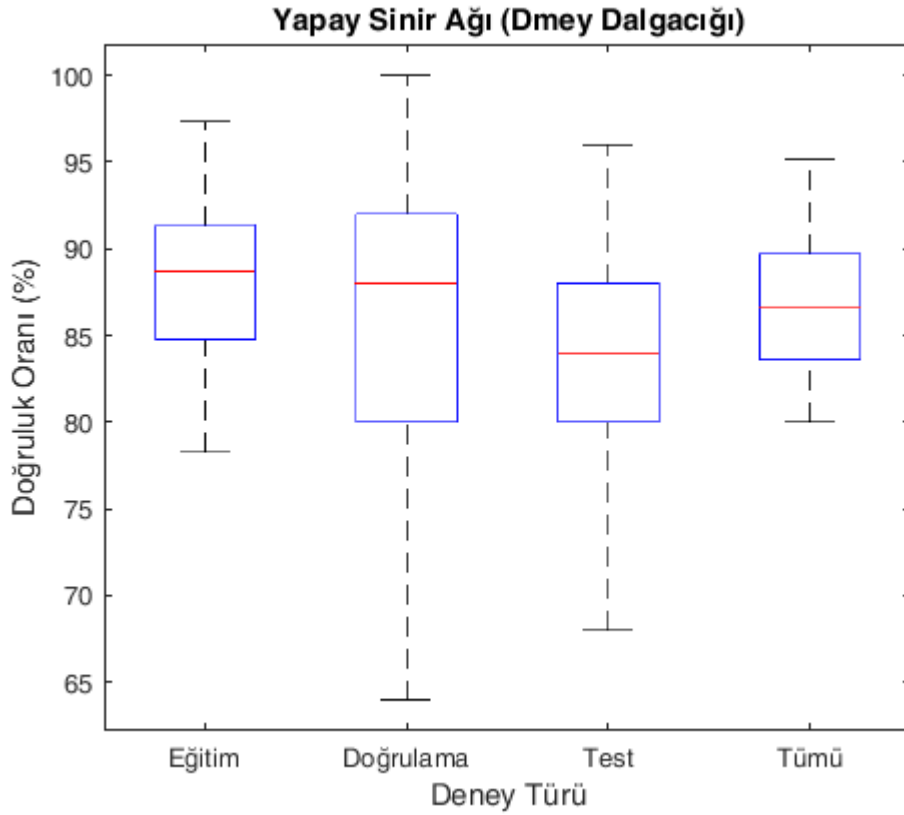
Tüm bu sonuçlara göre sinyal dönüşümlerine göre en yüksek sonuçlar Çizelge 4.1'de gösterilmektedir. Bu sonuçlara göre en yüksek sonucu Dmeyer dalgacıđı uygulanmıř veri kümesi üzerindeki sinir ađı modeli elde etmiřtir.

Çizelge 4.1: Normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları

Sinyal Dönüşümü	Gizli Katman Sayısı	Eğitim (%)	Doğrulama (%)	Test (%)	Tümü (%)
FFD	200	82.20±2.50	82.93±9.15	83.73±6.96	82.55±2.34
Gaussian2	250	83.19±5.73	80.53±6.10	82.13±9.26	82.63±4.85
BiorSplines3.3	200	85.19±4.22	85.20±7.51	83.73±5.65	84.97±2.85
Coiflets2	50	84.55±3.34	83.20±7.31	83.60±6.92	84.20±2.91
Daubechies2	350	84.75±3.39	82.93±8.40	82.53±8.17	84.14±1.99
Daubechies3	10	86.17±2.84	84.40±6.84	84.67±6.97	85.68±2.38
Daubechies4	200	84.12±4.27	82.27±7.33	82.40±6.94	83.58±3.33
Daubechies5	150	85.57±3.57	86.00±5.83	83.33±7.21	85.29±2.34
Daubechies6	50	84.52±3.28	83.73±6.03	84.27±8.27	84.36±2.13
Daubechies7	100	86.06±3.09	84.67±7.21	84.00±7.20	85.54±2.14
DMeyer	300	86.81±5.02	82.80±7.73	85.07±7.27	85.94±3.77
Symlets2	100	85.25±5.79	81.73±7.25	82.80±7.21	84.34±4.14
Haar	10	87.07±2.95	86.13±8.90	84.13±6.85	86.48±2.43
HHD	10	84.70±3.53	81.60±7.09	81.07±8.59	83.68±2.81

Dmeyer dalgacığı uygulanmış Parkinson veri kümesi üzerine uygulanan 300 gizli katmanlı sinir ağı modeli 30 deneyin ardından test kümesi üzerinde ortalama %85.07 doğruluk oranı vermiştir. Bu ortalama sonucun hassasiyetini arttırmak için deney sayısı 30'dan 100'e çıkartıldı. Böylelikle test kümesi üzerinde daha hassas bir ortalama sınıflandırma doğruluğu elde edilmiş oldu. Bu deneylerin sonucunda test kümesi üzerinde ortalama doğruluk oranı %83.00 olarak elde edildi. Aynı zamanda 100 deney içerisinde test kümesi üzerinde en yüksek sınıflandırma doğruluğu %96.00 olarak elde edilmiştir. En düşük sınıflandırma doğruluğu ise %68 olarak gözlemlendi. Bu sonuçların tümü Şekil 4.21'de ayrıntılı olarak gösterilmektedir. Bu sonuçlara göre eğitim ve doğrulama kümeleri üzerindeki ortalama sınıflandırma

doğruluğu yaklaşık olarak %88 olarak gözlenirken test kümesinde bu oran %83.00 olarak gözlemlendi.



Şekil 4.21: Dmey dalgacı uygulanmış Parkinson veri kümesi üzerinde bir Yapay Sinir Ağı modelinin eğitim, doğrulama, test ve tümünü içeren sınıflandırma sonuçları (Deney 100 kere gerçekleştirildi. Gizli katman sayısı 300'dür)

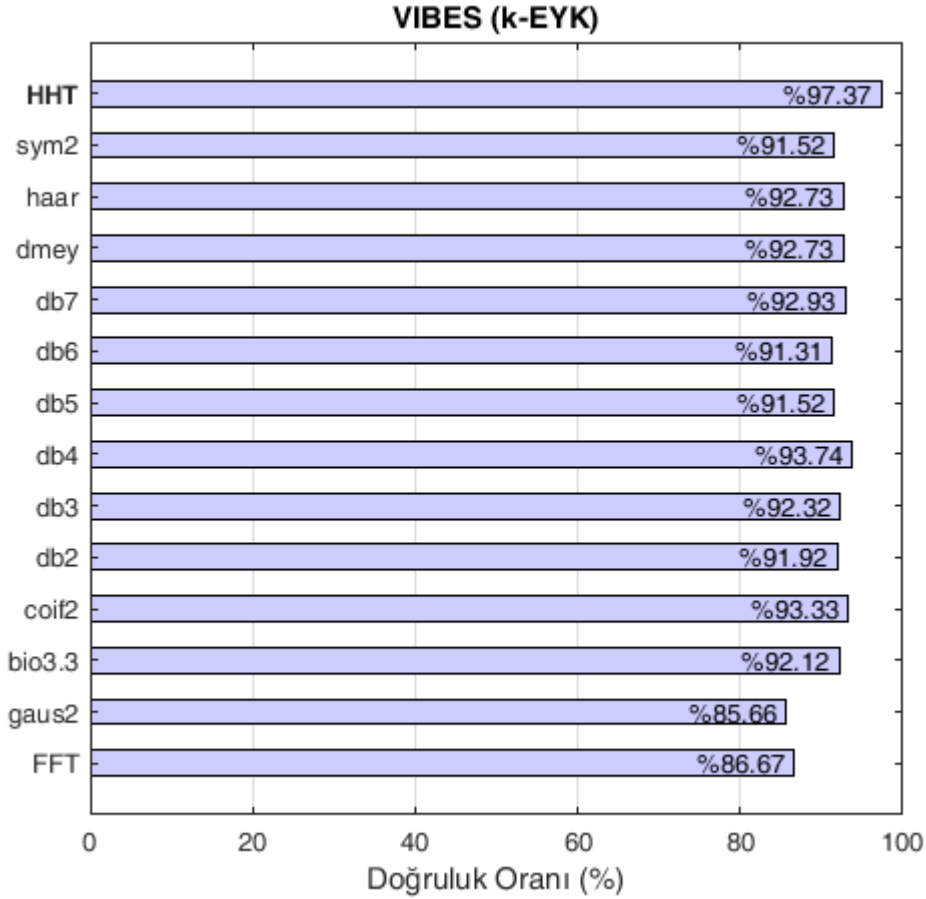
4.2.3 VIBES algoritmasının parkinson veri kümesi üzerindeki sonuçları

VIBES algoritması Parkinson veri kümesi üzerine uygulanırken yapılan tüm deneylerde VIBES algoritmasının tüm olası parametreleri dikkate alındı. Örneğin özneliklerin birbirlerine bağımlı ya da bağımsız olduğu, özneliklerin bağımlı olduğu varsayıma göre özneliklerin hangi enformasyon ölçüm yöntemine (ReliefF ya da Shannon entropi) göre sıralanacağı, topluluğu oluştururken optimize edilmiş ileri yönlü arama mı yoksa GA'nın mı kullanılacağı ve son olarak 6 temel öğrenciden hangisinin seçileceği gibi olası durumlar yapılan deneylerle belirlendi. Böylelikle Parkinson veri kümeleri üzerinde bu 36 olası senaryonun tümü gerçekleştirildi ve en iyi durum VIBES algoritmasının bu sınıflandırma problemi için parametre değerleri olarak ayarlandı. Herbir temel öğrenci için olası 6 durum Çizelge 4.2'de gösterilmektedir. Ayrıca yapılan herbir deney 3 kere tekrarlanmış ve sonuçların ortalaması alınmıştır.

Çizelge 4.2: Herbir Temel Öğrenici için temel parametrelerin olası durumları

No	Öznitelikler Bağımlı mı	Öznitelik Enformasyon Ölçüm Yöntemi	Topluluk Oluşturma Yöntemi
1	Evet	ReliefF	Optimize
2	Evet	ReliefF	GA
3	Evet	Shannon	Optimize
4	Evet	Shannon	GA
5	Hayır	Shannon	Optimize
6	Hayır	Shannon	GA

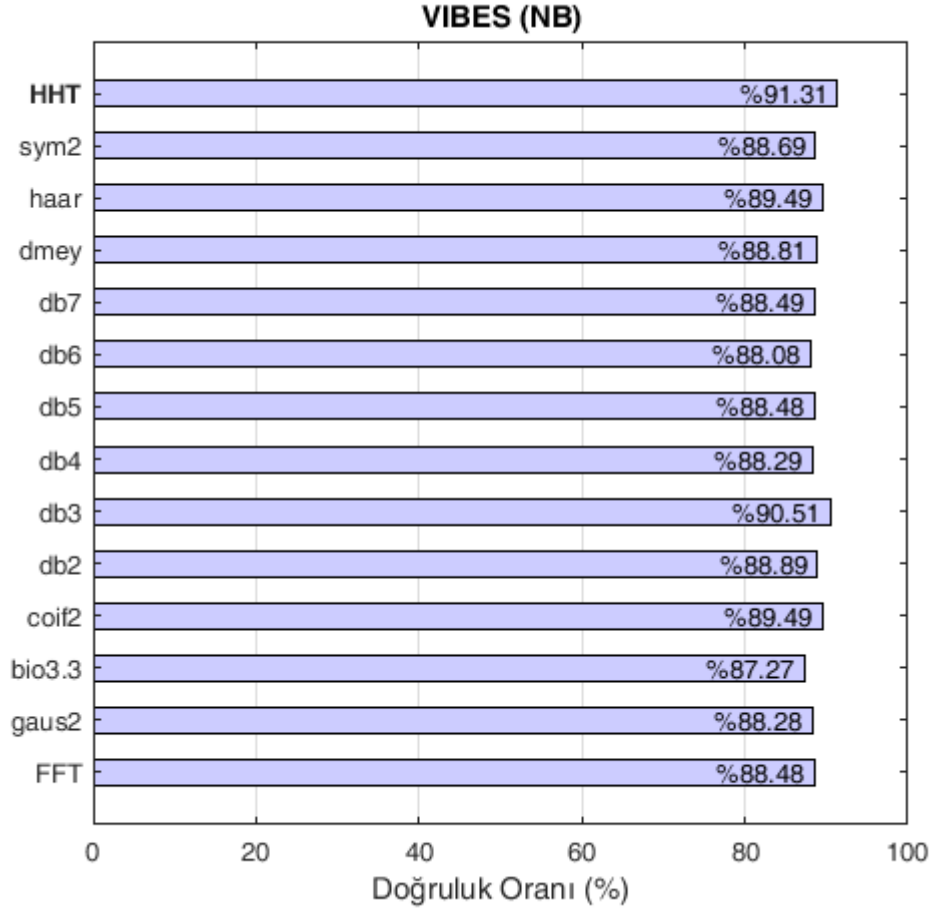
VIBES algoritması temel öğrenici olarak k-EYK algoritmasını kullandığında olası 6 durumun deneyi gerçekleştirildi. Yapılan deneylere göre Çizelge 4.2'deki 6 numaralı parametre değerlerine göre VIBES algoritması sadece 2 veri kümesi (FFD ve Gaussian2) üzerinde en yüksek sınıflandırma doğruluğuna sahip olurken; 5 numaralı parametre değerlerine göre geri kalan 12 veri kümesi üzerinde en yüksek sınıflandırma doğruluğuna sahip olmuştur. Buna göre VIBES algoritması özniteliklerin bağımsız olduğu varsayımına ve topluluk oluşturma yöntemi olarak optimize edilmiş ileri yönlü arama algoritmasına göre ayarlanmış parametre değerlerinde en yüksek sınıflandırma doğruluğunu vermiştir. Yapılan deneylerle ilgili ayrıntılı sonuçlar Çizelge C.37'de gösterilmektedir. Buna göre VIBES algoritması Temel Öğrenici olarak k-EYK algoritmasını kullandığında yapılan 6 olası duruma göre veri kümeleri üzerindeki en yüksek sınıflandırma doğruluğu oranları Şekil 4.22'de gösterilmektedir. Bu sonuçlara göre HHD'li Parkinson veri kümesi %97.37 sınıflandırma doğruluğu ile en yüksek sınıflandırma doğruluğuna sahiptir. En düşük sınıflandırma doğruluğu ise Gaussian2'li Parkinson veri kümesi için %85.66 olarak gözlenmiştir.



Şekil 4.22: VIBES algoritması Temel Öğrenici olarak k-EYK algoritmasını kullandığında yapılan 6 olası duruma göre veri kümeleri üzerindeki en yüksek sınıflandırma doğruluğu oranları

VIBES algoritması temel öğrenici olarak NB algoritmasını kullandığında olası 6 durumun deneyi gerçekleştirildi. Yapılan deneylere göre Çizelge 4.2'deki 6 numaralı parametre değerlerine göre VIBES algoritması sadece 3 veri kümesi (FFD, Gaussian2 ve Dmeyer) üzerinde en yüksek sınıflandırma doğruluğuna sahip olurken; 5 numaralı parametre değerlerine göreyse geri kalan 11 veri kümesi üzerinde en yüksek sınıflandırma doğruluğuna sahip olmuştur. Buna göre VIBES algoritması özneliklerin bağımsız olduğu varsayımına ve topluluk oluşturma yöntemi olarak optimize edilmiş ileri yönlü arama algoritmasına göre ayarlanmış parametre değerlerinde en yüksek sınıflandırma doğruluğunu vermiştir. Yapılan deneylerle ilgili ayrıntılı sonuçlar Çizelge C.38'de gösterilmektedir. Buna göre VIBES algoritması Temel Öğrenici olarak NB algoritmasını kullandığında yapılan 6 olası duruma göre veri kümeleri üzerindeki en yüksek sınıflandırma doğruluğu oranları Şekil 4.23'te gösterilmektedir. Bu sonuçlara göre HHD'li Parkinson veri kümesi

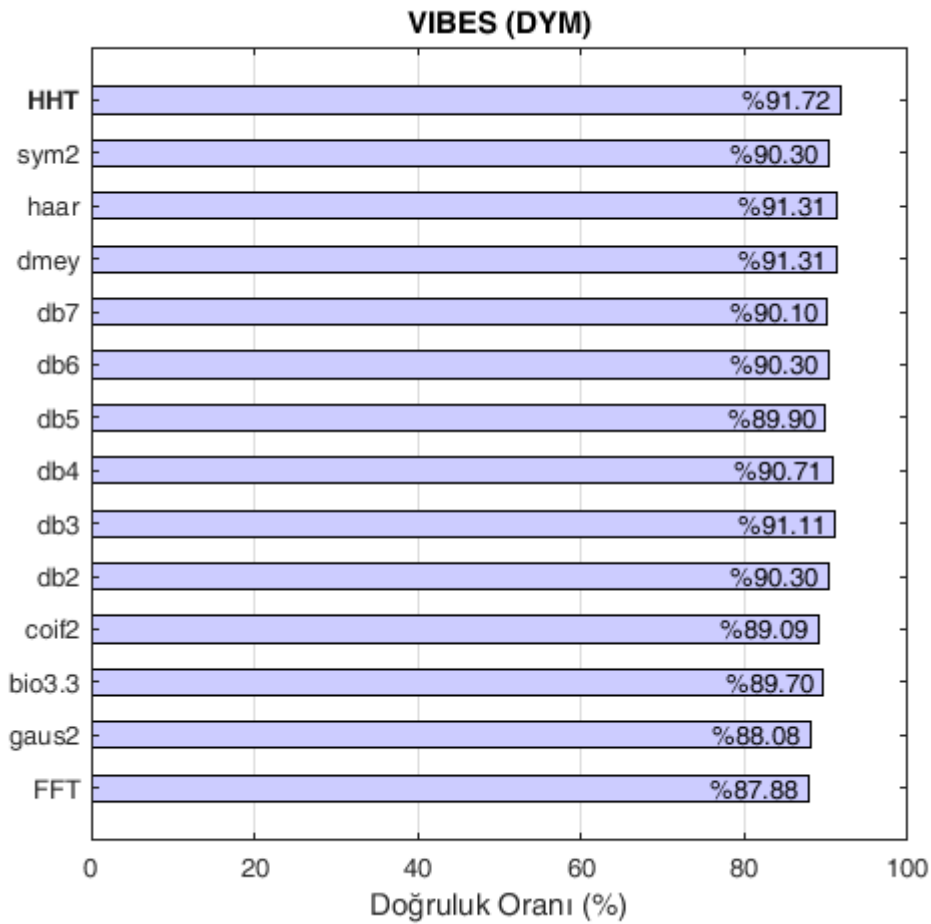
%91.31 sınıflandırma doğruluğu ile en yüksek sınıflandırma doğruluğuna sahiptir. En düşük sınıflandırma doğruluğu ise BiorSplines3.3'lü Parkinson veri kümesi için %87.27 olarak gözlenmiştir.



Şekil 4.23: VIBES algoritması Temel Öğrenici olarak NB algoritmasını kullandığında yapılan 6 olası duruma göre veri kümeleri üzerindeki en yüksek sınıflandırma doğruluğu oranları

VIBES algoritması temel öğrenici olarak DYM algoritmasını kullandığında olası 6 durumun deneyi gerçekleştirildi. Yapılan deneylere göre Çizelge 4.2'deki 1 numaralı parametre değerlerine göre VIBES algoritması sadece 3 veri kümesi (Daubechies5, Daubechies6 ve Dmeyer) üzerinde en yüksek sınıflandırma doğruluğuna sahip olurken; 3 numaralı parametre değerleri için 1 veri kümesi (Symlets2), 6 numaralı parametre değerleri için 2 veri kümesi (FFD ve Gaussian2) ve son olarak 5 numaralı parametre değerleri için de geri kalan 8 veri kümesi üzerinde en yüksek sınıflandırma doğruluğuna sahip olmuştur. Buna göre VIBES algoritması özneliliklerin bağımsız olduğu varsayımına ve topluluk oluşturma yöntemi olarak optimize edilmiş ileri

yönlü arama algoritmasına göre ayarlanmış parametre değerlerinde en yüksek sınıflandırma doğruluğunu vermiştir. Yapılan deneylerle ilgili ayrıntılı sonuçlar Çizelge C.39’da gösterilmektedir. Buna göre VIBES algoritması Temel Öğrenici olarak DYM algoritmasını kullandığında yapılan 6 olası duruma göre veri kümeleri üzerindeki en yüksek sınıflandırma doğruluğu oranları Şekil 4.24’te gösterilmektedir. Bu sonuçlara göre HHT’li Parkinson veri kümesi %91.72 sınıflandırma doğruluğu ile en yüksek sınıflandırma doğruluğuna sahiptir. En düşük sınıflandırma doğruluğu ise FFT’li Parkinson veri kümesi için %87.88 olarak gözlenmiştir.

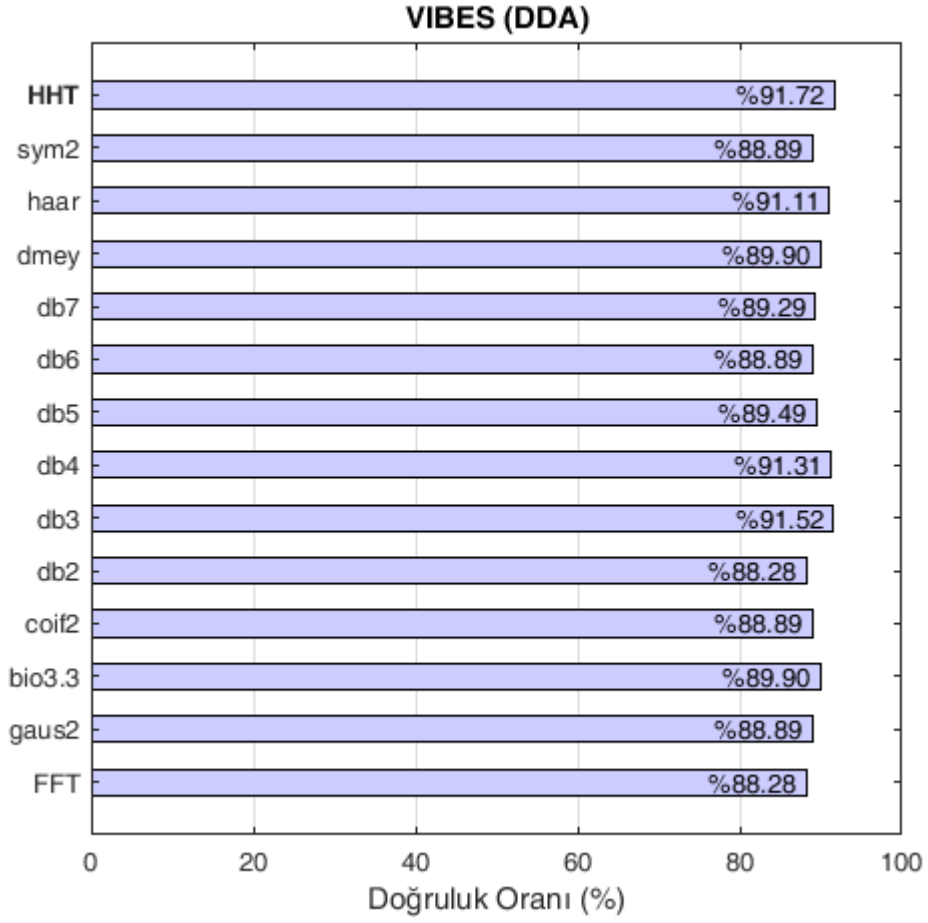


Şekil 4.24: VIBES algoritması Temel Öğrenici olarak DYM algoritmasını kullandığında yapılan 6 olası duruma göre veri kümeleri üzerindeki en yüksek sınıflandırma doğruluğu oranları

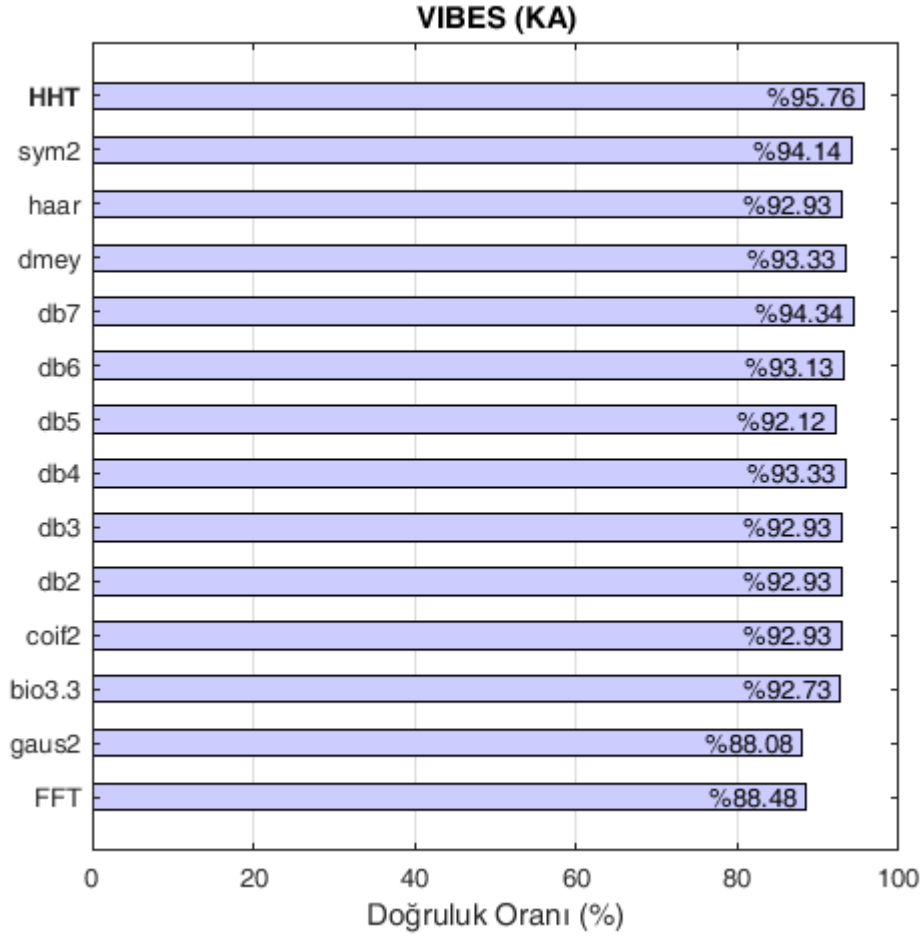
VIBES algoritması temel öğrenici olarak DDA algoritmasını kullandığında olası 6 durumun deneyi gerçekleştirildi. Yapılan deneylere göre Çizelge 4.2’deki 1 numaralı parametre değerlerine göre VIBES algoritması 5 veri kümesi (Coiflets2, Daubechies3, Daubechies4, Daubechies6 ve Dmeyer) üzerinde en yüksek sınıflandırma

doğruluğuna sahip olurken; 6 numaralı parametre değerleri için 2 veri kümesi (FFD ve Gaussian2) ve son olarak 5 numaralı parametre değerleri için de geri kalan 9 veri kümesi üzerinde en yüksek sınıflandırma doğruluğuna sahip olmuştur (Bazı veri kümeleri üzerinde farklı parametre değerlerinin aynı sınıflandırma doğruluğuna sahip oldukları görülmüştür). Buna göre VIBES algoritması özniteliklerin bağımsız olduğu varsayımına ve topluluk oluşturma yöntemi olarak optimize edilmiş ileri yönlü arama algoritmasına göre ayarlanmış parametre değerlerinde en yüksek sınıflandırma doğruluğunu vermiştir. Yapılan deneylerle ilgili ayrıntılı sonuçlar Çizelge C.40'ta gösterilmektedir. Buna göre VIBES algoritması Temel Öğrenici olarak DDA algoritmasını kullandığında yapılan 6 olası duruma göre veri kümeleri üzerindeki en yüksek sınıflandırma doğruluğu oranları Şekil 4.25'te gösterilmektedir. Bu sonuçlara göre HDD'li Parkinson veri kümesi %91.72 sınıflandırma doğruluğu ile en yüksek sınıflandırma doğruluğuna sahiptir. En düşük sınıflandırma doğruluğu ise FFD'li ve Daubechies2'li Parkinson veri kümeleri için %88.28 olarak gözlenmiştir.

VIBES algoritması temel öğrenici olarak KA algoritmasını kullandığında olası 6 durumun deneyi gerçekleştirildi. Yapılan deneylere göre Çizelge 4.2'deki 6 numaralı parametre değerlerine göre VIBES algoritması sadece 2 veri kümesi (FFD ve Gaussian2) üzerinde en yüksek sınıflandırma doğruluğuna sahip olurken; 5 numaralı parametre değerlerine göreyse geri kalan 12 veri kümesi üzerinde en yüksek sınıflandırma doğruluğuna sahip olmuştur. Buna göre VIBES algoritması özniteliklerin bağımsız olduğu varsayımına ve topluluk oluşturma yöntemi olarak optimize edilmiş ileri yönlü arama algoritmasına göre ayarlanmış parametre değerlerinde en yüksek sınıflandırma doğruluğunu vermiştir. Yapılan deneylerle ilgili ayrıntılı sonuçlar Çizelge C.41'de gösterilmektedir. Buna göre VIBES algoritması Temel Öğrenici olarak KA algoritmasını kullandığında yapılan 6 olası duruma göre veri kümeleri üzerindeki en yüksek sınıflandırma doğruluğu oranları Şekil 4.26'da gösterilmektedir. Bu sonuçlara göre HDD'li Parkinson veri kümesi %95.76 sınıflandırma doğruluğu ile en yüksek sınıflandırma doğruluğuna sahiptir. En düşük sınıflandırma doğruluğu ise Gaussian2'li Parkinson veri kümesi için %88.08 olarak gözlenmiştir.



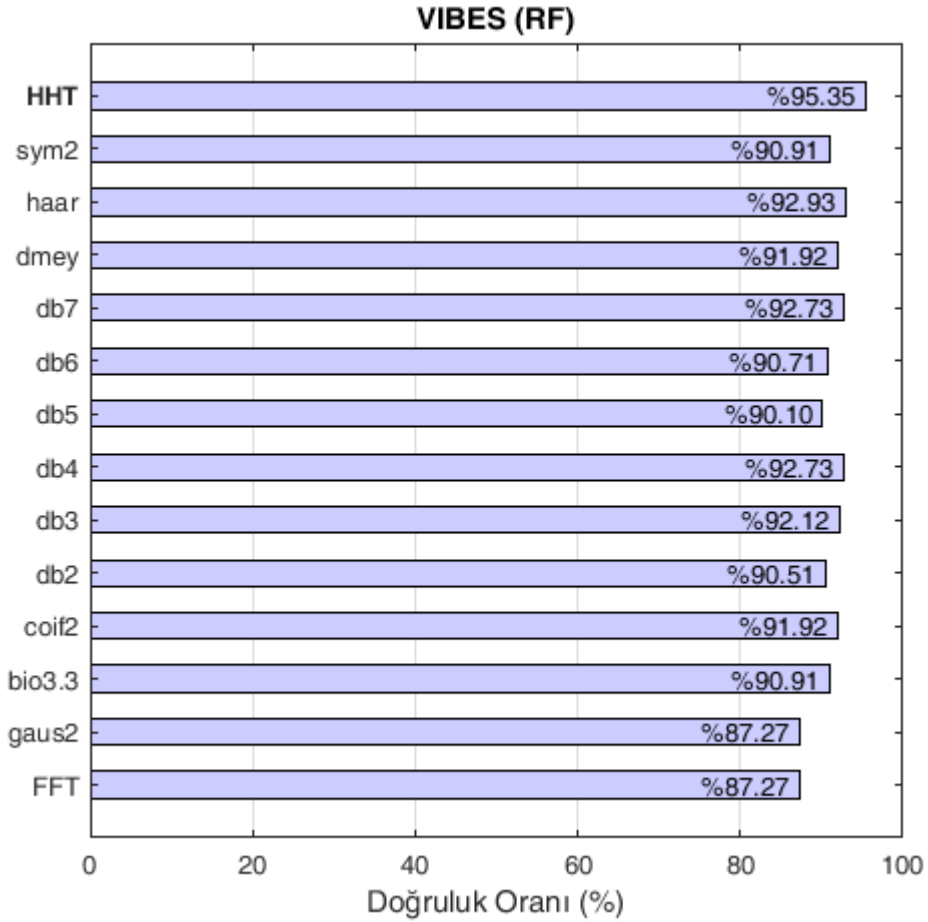
Şekil 4.25: VIBES algoritması Temel Öğrenici olarak DDA algoritmasını kullandığında yapılan 6 olası duruma göre veri kümeleri üzerindeki en yüksek sınıflandırma doğruluğu oranları



Şekil 4.26: VIBES algoritması Temel Öğrenici olarak KA algoritmasını kullandığında yapılan 6 olası duruma göre veri kümeleri üzerindeki en yüksek sınıflandırma doğruluğu oranları

VIBES algoritması temel öğrenici olarak RF algoritmasını kullandığında olası 6 durumun deneyi gerçekleştirildi. Yapılan deneylere göre Çizelge 4.2'deki 2 numaralı parametre değerlerine göre VIBES algoritması 1 veri kümesi (Gaussian2) üzerinde en yüksek sınıflandırma doğruluğuna sahip olurken; 3 numaralı parametre değerleri için 1 veri kümesi (FFD) ve son olarak 5 numaralı parametre değerleri için de geri kalan 12 veri kümesi üzerinde en yüksek sınıflandırma doğruluğuna sahip olmuştur. Buna göre VIBES algoritması özneliliklerin bağımsız olduğu varsayımına ve topluluk oluşturma yöntemi olarak optimize edilmiş ileri yönlü arama algoritmasına göre ayarlanmış parametre değerlerinde en yüksek sınıflandırma doğruluğunu vermiştir. Yapılan deneylerle ilgili ayrıntılı sonuçlar Çizelge C.42'de gösterilmektedir. Buna göre VIBES algoritması Temel Öğrenici olarak RF algoritmasını kullandığında yapılan 6 olası duruma göre veri kümeleri üzerindeki en yüksek sınıflandırma

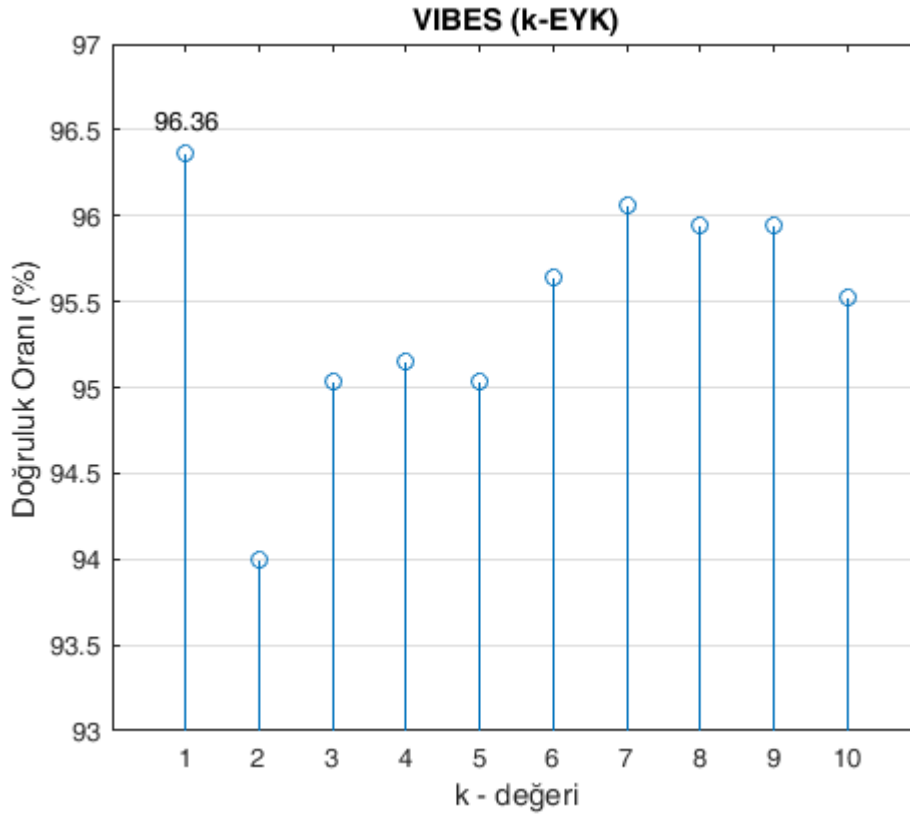
doğruluğu oranları Şekil 4.27’de gösterilmektedir. Bu sonuçlara göre HHD’li Parkinson veri kümesi %95.35 sınıflandırma doğruluğu ile en yüksek sınıflandırma doğruluğuna sahiptir. En düşük sınıflandırma doğruluğu ise FFD’li ve Gaussian2’li Parkinson veri kümeleri için %87.27 olarak gözlenmiştir.



Şekil 4.27: VIBES algoritması Temel Öğrenici olarak RF algoritmasını kullandığında yapılan 6 olası duruma göre veri kümeleri üzerindeki en yüksek sınıflandırma doğruluğu oranları

Yapılan bu deneyler sonucunda HHD’li Parkinson veri kümesinin VIBES algoritmasının tüm Temel Öğrenicileri tarafından en yüksek doğruluk oranıyla sınıflandırıldığı görülmüştür. Ayrıca bu veri kümesi üzerinde en yüksek sınıflandırma doğruluğunu iki temel öğrenici elde etmiştir. Bu temel öğreniciler k-EYK ve KA algoritmalarıdır (Deney parametreleri: özneliklerin bağımsızlığı varsayımı ve topluluk oluşturma yöntemi olarak optimize edilmiş ileri yönlü arama). Bu temel öğrenicilerden k-EYK algoritmasının parametre değerleri varsayılan olarak kullanıldı. Yalnız k-EYK algoritmasının k değeri bu sınıflandırıcının modelinin

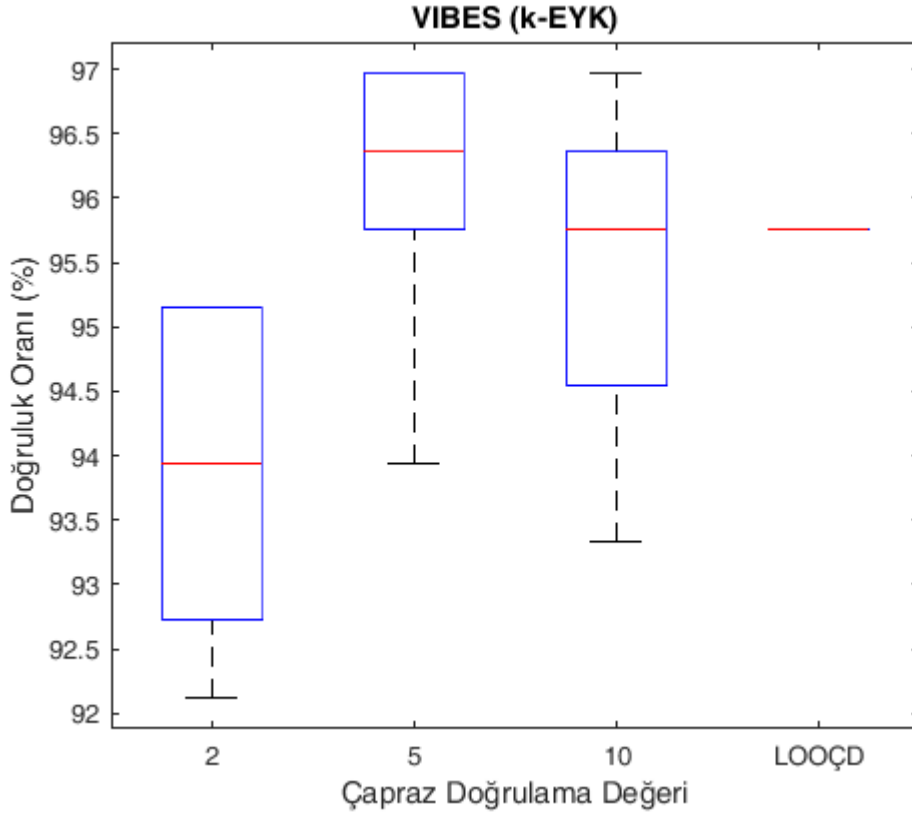
kompleksliğini oldukça belirleyici kılmaktadır. Yani k değeri küçüldükçe daha basit modeller oluşurken k değerinin artmasıyla k-EYK sınıflandırıcının oluşturmuş olduğu modelde kompleksleşmektedir. Bu durum bias-varyans ikilemine neden olmaktadır. Bu nedenle k değerinin 1'den 10'a kadar olan değişiminin sınıflandırma doğruluğuna olan etkisi analiz edildi. Bu analiz sonucunda k'nın 1 değerinin optimal olduğu görüldü. Böylelikle çok basit bir modelin HHD'li veri kümesi üzerinde oldukça iyi bir sınıflandırma sağladığı söylenebilir. Yani doğrusal bir sınıflandırıcının HHD'li veri kümesini sınıflandırmada daha iyi sonuç vereceği söylenebilir. Ayrıca yapılan her bir deney 10 kere tekrarlanarak kararlı bir sonucunun elde edilmesi sağlandı. Yapılan deneylerin sonuçları Şekil 4.28'de gösterilmektedir. Ayrıntılı sonuçlar ise Çizelge C.43'te gösterilmektedir.



Şekil 4.28: k değerinin değişimine göre VIBES (k-EYK) algoritmasının HHD'li veri kümesi üzerindeki sınıflandırma doğruluğunun değişim grafiği

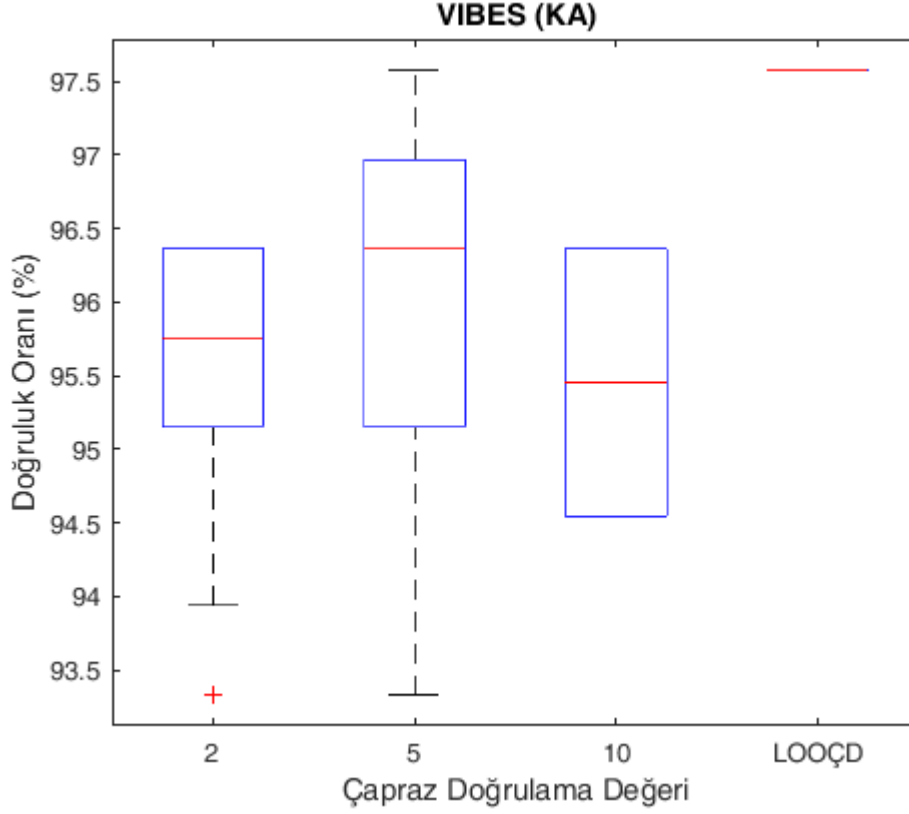
VIBES (k-EYK) algoritması için optimal k değeri 1 olarak bulunduktan sonra VIBES (1-EYK) için çapraz doğrulama yöntemine göre sınıflandırma doğruluğunun değişimleri incelenmiştir. Çapraz doğrulama yöntemi için yapılan deneyler şunlardır: 2-katlı çapraz doğrulama, 5-katlı çapraz doğrulama, 10-katlı çapraz doğrulama ve

LOOÇD'dir. Herbir çapraz doğrulama yöntemi için 10 deney gerçekleştirildi. Bu deneylerin sonuçları Şekil 4.29'da gösterilmektedir. Sonuçların ayrıntıları ise Çizelge C.44'te gösterilmektedir. Buna göre bu dört yöntemin ortalama sınıflandırma doğruluğu %95.32 olarak elde edilmiştir.



Şekil 4.29: Çapraz doğrulama yönteminin değişimine göre VIBES (1-EYK) algoritmasının sınıflandırma doğruluğunun ortalaması, standart sapması ve uç değerleri

VIBES (1-EYK) algoritması için için çapraz doğrulama yöntemine göre sınıflandırma doğruluğunun değişimleri incelendikten sonra VIBES (KA) algoritmasının çapraz doğrulama yöntemine göre sınıflandırma doğruluğunun değişimleri incelenmiştir. Yapılan deneyler sonucunda temel öğrenici olarak KA algoritması 1-EYK algoritmasına göre daha yüksek doğrulukta sınıflandırma gerçekleştirmiştir. VIBES (KA) algoritmasının çapraz doğrulama yönteminin değişimine göre sonuçları Şekil 4.30'da gösterilmektedir. Sonuçların ayrıntıları ise Çizelge C.45'te gösterilmektedir. Buna göre bu dört yöntemin ortalama sınıflandırma doğruluğu %96.11 olarak elde edilmiştir.



Şekil 4.30: Çapraz doğrulama yönteminin değişimine göre VIBES (KA) algoritmasının sınıflandırma doğruluğunun ortalaması, standart sapması ve uç değerleri

Bu sonuçlar ışığında Parkinson veri kümeleri içerisinde VIBES algoritmasının en yüksek sınıflandırma doğruluğu elde ettiği veri kümesi HHD'li Parkinson veri kümesidir. Bu veri kümesi üzerinde basit ve en yüksek sınıflandırma doğruluğunu veren temel öğrenici KA'dır. Ayrıca HHD'li veri kümesi üzerinde VIBES algoritmasının performansını arttıran parametreler şunlardır: Özniteliklerin bağımsızlığı varsayımı ve topluluk oluşturma yöntemi olarak optimize edilmiş ileri yönlü aramadır. Ayrıca çapraz doğrulama yöntemindeki değişime göre sınıflandırma doğruluğu oranı %95.45'in altına düşmemiştir. En yüksek sınıflandırma doğruluğu ise LOOÇD yönteminde %97.58 olarak gözlenmiştir. Ayrıca bu yöneme göre deneyler 10 kere tekrarlanmasına rağmen bu sonuç hiç değişmemiş ve standart sapma değeri sıfır olmuştur. Tüm bu nedenlerden dolayı Parkinson veri kümesi için en performanslı modelin VIBES (KA) algoritmasının LOOÇD yöntemine göre oluşturmuş olduğu model olduğu söylenebilir. Bu modele göre seçilen özniteliklerin sayısı 14'tür. Bu 14 öznitelik şunlardır: 43, 68, 127, 3, 111, 99, 126, 13, 35, 131, 45, 129, 81 ve 87. özniteliklerdir (Öznitelik kümesi Shannon entropiye göre sıralanma

durumuna göre verilmiştir). Bu öznitelikler ayrıntılı olarak Çizelge 4.3'te gösterilmektedir.

Çizelge 4.3: VIBES (KA) algoritmasının LOOÇD yöntemine göre HDD'li veri kümesi üzerinde oluşturduğu modelin seçtiği öznitelik kümesi

Öznitelik No	Shannon Entropiden Önce Öznitelik No	Ayak Sinyali	Sinyale Uygulanan İstatistiksel Fonksiyon	Sınıflandırma Doğruluğu (%)
43	3	L1 + L2	mad(modes_3, 1)	77.58
68	15	L1 - L4	min(modes_1)	74.55
127	18	L1 - R4	mad(pks3_1, 0)	72.12
3	41	L1 - L7	median(p2_3)	75.76
111	45	L1 - R2	max(pks3_2)	68.48
99	48	L1 - L5	mad(p3_3, 0)	76.36
126	70	R1 - R6	mean(w4_2)	75.76
13	87	L8 + R1	median(pks3_5)	70.30
35	98	L1 - L3	norm(p2_2, 1)	66.67
131	102	L1 - L3	wentropy(amplitude_2, 'shannon')	63.03
45	105	L1 - L6	mean(pxx)	72.73
129	107	L1 - L6	mad(p3_2, 0)	63.03
81	121	L4 - R4	std(modes_2)/mean(modes_2)	72.73
87	133	R1 - R6	mad(pks2_1, 1)	72.12

43. özniteliğe göre oluşturulan karar ağacı yapısı ve kural yapısı sırasıyla Şekil C.1 ve Şekil C.2'de gösterilmektedir. 68. özniteliğe göre oluşturulan karar ağacı yapısı ve kural yapısı sırasıyla Şekil C.3 ve Şekil C.4'te gösterilmektedir. 127. özniteliğe göre oluşturulan karar ağacı yapısı ve kural yapısı sırasıyla Şekil C.5 ve Şekil C.6'da gösterilmektedir. 3. özniteliğe göre oluşturulan karar ağacı yapısı ve kural yapısı sırasıyla Şekil C.7 ve Şekil C.8'de gösterilmektedir. 111. özniteliğe göre oluşturulan karar ağacı yapısı ve kural yapısı sırasıyla Şekil C.9 ve Şekil C.10'da gösterilmektedir. 99. özniteliğe göre oluşturulan karar ağacı yapısı ve kural yapısı sırasıyla Şekil C.11 ve Şekil C.12'de gösterilmektedir. 126. özniteliğe göre oluşturulan karar ağacı yapısı ve kural yapısı sırasıyla Şekil C.13 ve Şekil C.14'te

gösterilmektedir. 13. özniteliğe göre oluşturulan karar ağacı yapısı ve kural yapısı sırasıyla Şekil C.15 ve Şekil C.16'da gösterilmektedir. 35. özniteliğe göre oluşturulan karar ağacı yapısı ve kural yapısı sırasıyla Şekil C.17 ve Şekil C.18'de gösterilmektedir. 131. özniteliğe göre oluşturulan karar ağacı yapısı ve kural yapısı sırasıyla Şekil C.19 ve Şekil C.20'de gösterilmektedir. 45. özniteliğe göre oluşturulan karar ağacı yapısı ve kural yapısı sırasıyla Şekil C.21 ve Şekil C.22'de gösterilmektedir. 129. özniteliğe göre oluşturulan karar ağacı yapısı ve kural yapısı sırasıyla Şekil C.23 ve Şekil C.24'de gösterilmektedir. 81. özniteliğe göre oluşturulan karar ağacı yapısı ve kural yapısı sırasıyla Şekil C.25 ve Şekil C.26'da gösterilmektedir. 87. özniteliğe göre oluşturulan karar ağacı yapısı ve kural yapısı sırasıyla Şekil C.27 ve Şekil C.28'de gösterilmektedir.

VIBES (KA) algoritmasının LOOÇD'ye göre oluşturduğu modelin HHD'li Parkinson veri kümesi üzerindeki sınıflandırma sonuçları Çizelge 4.4'te gösterilmektedir. Bu sonuçlara göre 165 verinin 161 tanesi doğru olarak sınıflandırılmış ve sınıflandırma doğruluğu %97.58 olarak elde edilmiştir. Ayrıca modelin öğrenme düzeyini ölçen Kappa değeri 0.9507 olarak elde edilmiştir. Böylelikle modelin mükemmel bir öğrenme gerçekleştirdiğini söyleyebiliriz. Root Mean Squared Error (RMSE) değeri 0.1557 olarak elde edilmiştir. Bu değer in sıfıra çok yakın olması VIBES (KA) algoritmasının modelinin yüksek doğrulukta tahminler yaptığını göstermektedir.

Çizelge 4.4: VIBES (KA) algoritmasının LOOÇD yöntemine göre oluşturduğu modelin HHD’li Parkinson veri kümesi üzerindeki istatistiksel sonuçları

İstatistiksel Ölçüt	Değeri
Örnek sayısı	165
Doğru sınıflandırılan örnek sayısı	161
Yanlış sınıflandırılan örnek sayısı	4
Sınıflandırma Doğruluğu	%97.5758
Kappa	0.9507
Mean Absolute Error (MAE)	0.0242
Root Mean Absolute Error (RMSE)	0.1557
Relative Absolute Error (RAE)	20.6250
Root Relative Squared Error (RRSE)	6.4226

VIBES (KA) algoritmasının HHD’li Parkinson veri kümesi üzerindeki Hata Matrisi (Confusion Matrix) Çizelge 4.5’te gösterilmektedir. Bu sonuca göre 2 parkinsonlu birey yanlış sınıflandırılarak sağlıklı birey olarak tahmin edilmiştir. Aynı zamanda 2 sağlıklı birey yanlış sınıflandırılarak parkinsonlu birey olarak tahmin edilmiştir. Bunun 4 bireyin dışındaki 161 birey için yapılan sınıflandırma tahmini doğru bir biçimde gerçekleştirilmiştir.

Çizelge 4.5: VIBES (KA) algoritmasının Hata Matrisi (Confusion Matrix)

Hata Matrisi	CO (tahmin sınıfı)	PD (tahmin sınıfı)
CO (gerçek sınıf)	70	2
PD (gerçek sınıf)	2	91

VIBES (KA) algoritmasının detaylı doğruluk tablosu Çizelge 4.6’te gösterilmektedir. Bu tabloya göre VIBES (KA) algoritması Parkinsonlu (PD) ve Kontrol (CO) sınıfı için de yüksek oranda sınıflandırma doğruluğu elde etmiştir. Bunun için F-Ölçütü ve ROC değerlerini incelemek yeterlidir. Bu sonuçların 1’e çok yakın olması her iki sınıfta (PD ve CO) VIBES (KA) algoritması tarafından yeterince iyi öğrenildiğini

ortaya koymaktadır. Yalnız bu iki sınıf içerisinde en iyi öğrenmenin PD’li sınıfa ait öğrenme olduğunu F-Ölçütü değerinin 0.9785 olmasından çıkarmaktayız. Çünkü CO sınıfı için bu değer 0.9722’dir. Ayrıca Kesinlik ve Hassasiyet sonuçlarında da PD sınıfı CO sınıfına göre VIBES (KA) algoritması tarafından bir nebze daha iyi öğrenilmiştir.

Çizelge 4.6: VIBES (KA) için HHD’li veri kümesi üzerindeki detaylı doğruluk tablosu

	GP	HP	Kesinlik	Hassasiyet	F-Ölçütü	ROC	Sınıflar
	0.9722	0.0215	0.9722	0.9722	0.9722	0.9513	CO
	0.9785	0.0278	0.9785	0.9785	0.9785	0.9513	PD
Ağırlıklı Ortalama	0.9754	0.0246	0.9754	0.9754	0.9754	0.9513	

PhysioNET (Goldberger et al. 2000) veritabanından elde edilen Parkinson veri kümesi (Frenkel-Toledo, Giladi, Peretz, Herman, Gruendlinger & Jeffrey M Hausdorff 2005; Yogeve et al. 2005; Frenkel-Toledo, Giladi, Peretz, Herman, Gruendlinger & Jeffrey M. Hausdorff 2005; Hausdorff et al. 2007) üzerinde *VIBES (KA)* algoritması ve diğer çalışmaların karşılaştırmalı sonuçlarını içeren tablo Çizelge 4.7’de gösterilmektedir. Bu sonuçlara göre VIBES (KA) algoritması diğer tüm çalışmalardan daha yüksek oranda sınıflandırma doğruluğu veren bir model oluşturmuştur. Önerdiğimiz yonteme en yakın sonuçlar Zeng vd. (Zeng et al. 2016) tarafından yapılan çalışmaya aittir. Bu tez çalışmasının önerdiği yöntem (HHD+VIBES (KA)), 2-katlı çapraz doğrulama ve LOOÇD yöntemlerine göre yapılan sınıflandırma tahminlerinde Zeng vd.’nin önerdiği yöntemden daha yüksek doğruluk oranı elde etmiştir. Yalnızca 5-katlı çapraz doğrulama yöntemine göre Zeng vd.’nin önerdiği yöntem bu tez çalışmasının önerdiği yöntemden %0.4506 gibi küçük bir doğruluk oranı farkıyla yüksek çıkmıştır. Sonuç olarak bu tez çalışmasının önerdiği yöntemin ortalama sınıflandırma doğruluğu oranı tüm diğer çalışmaların önerdiği yöntemlerin ortalama sınıflandırma doğruluğundan daha yüksektir.

Çizelge 4.7: Aynı Parkinson veri kümesi üzerinde VIBES (KA) algoritması ve diğer çalışmaların sonuçlarının karşılaştırılması

Yapılan Çalışmalar	Veri Türü	Sinyal Dönüşüm/Öznitelik Oluşturma Yöntemi	Sınıflandırma Algoritması	Çapraz Doğrulama	Doğruluk Oranı	Ortalama Doğruluk Oranı
Önerilen Yöntem	Yere uygulanan dikey kuvvet sinyalleri	HHD ve İstatistiksel Yöntemler	VIBES (KA)	2-katlı çapraz doğrulama	95.4545±1.0785	%96.1061
				5-katlı çapraz doğrulama	95.9394±1.3717	
				10-katlı çapraz doğrulama	95.4545±0.7693	
				LOOÇD	97.5758±0.0000	
(Zeng et al. 2016)	Yere uygulanan dikey kuvvet sinyalleri	İstatistiksel Yöntemler	Radyal Tabanlı Fonksiyon Sinir Ağı	Tümü Eğitim ve Tümü Test	%99.40	%93.7767
				2-katlı çapraz doğrulama	%91.57	
				5-katlı çapraz doğrulama	%96.39	
(Shrivastava et al. 2017)	Yere uygulanan dikey kuvvet sinyalleri	İkili Ayıklama Algoritması	Yapay Sinir Ağı	LOOÇD	%93.37	%93.1200
				Çapraz doğrulama	%93.08 (Ga verisi)	
					%96.35 (Ju verisi)	
					%89.93 (Si verisi)	

Çizelge 4.7: (devam) Aynı Parkinson veri kümesi üzerinde VIBES (KA) algoritması ve diğer çalışmaların sonuçlarının karşılaştırılması

(Jane et al. 2016)	Yere uygulanan dikey kuvvet sinyalleri	-	Q-Geri Yayılımlı Zaman Gecikmeli Sinir Ağı	10-katlı çapraz doğrulama	%93.10 (Ga verisi) %91.67 (Si verisi) %89.66 (Ju verisi)	%91.5300
(Daliri 2013)	Yere uygulanan dikey kuvvet sinyalleri	Kısa Zamanlı Fourier Dönüşümü ve İstatistiksel Yöntemler	DYM (Ki-Kare uzaklık çekirdeği)	Veri kümesinin %50'si eğitim, %50'si test kümesi (20 kere çalıştırıldı)	%91.20	%91.2000
(Ertuğrul et al. 2016)	Yere uygulanan dikey kuvvet sinyalleri	İstatistiksel Yöntemler	Kaydırılmış Bir-Boyutlu İkili Örüntüler	10-katlı çapraz doğrulama	%88.8889	%88.8889
(Lee & Lim 2012)	Yere uygulanan dikey kuvvet sinyalleri	Dalgacık Dönüşümü ve İstatistiksel Yöntemler	Ağırlıklandırılmış Bulanık Üyelik Fonksiyonlu Sinir Ağı	-	%77.33	%77.3333

4.3 Bu Tez Çalışmasının Literatüre Katkıları ve Sonuçların Değerlendirilmesi

Bu tez çalışmasının literatüre yapmış olduğu katkılar, değerlendirmeler ve sonuçlar aşağıda maddeler halinde belirtilmektedir.

- Bu tez çalışmasında VIBES isminde yeni bir topluluk öğrenme algoritması geliştirilmiştir. Bu algoritma 33 adet benchmark veri kümesi ve Parkinson veri kümesi üzerinde diğer algoritmalara göre en iyi sonuçları vermiştir.
- VIBES algoritmasının 33 adet benchmark veri kümesi üzerindeki ortalama doğruluk oranı %89.80'dir. Bu veri kümeleri üzerindeki ikinci en iyi ortalamaya sahip öğrenme algoritması %87.76 doğruluk oranıyla Rotation Forest algoritmasıdır. Her iki algoritma için kullanılan temel öğrenici Random Forests algoritmasıdır.
- VIBES (RF) algoritmasında topluluk birleştirme yöntemi için çoğunluk oylama yöntemi kullanılmıştır. 2.3.4. Bölümde çoğunluk oylama yönteminin diğer yöntemlere göre daha iyi olduğu teorik olarak gösterilmiştir.
- VIBES (RF) algoritmasında hipotez arama algoritması olarak optimize edilmiş ileri yönlü bir arama yöntemi ve GA yöntemi kullanılmıştır. 33 adet benchmark veri kümesi üzerindeki sonuçlara göre optimize edilmiş ileri yönlü arama algoritması ortalama %89.6963 doğruluk oranı ve ortalama 3.2359 saniyelik arama zamanı ile GA yöntemine göre daha iyi sonuç elde edilmiştir. GA yöntemiyle ise %89.4184 doğruluk oranı ve ortalama 127.6958 saniyelik arama zamanı elde edilmiştir. Buna göre doğruluk oranları arasında çok ciddi bir fark olmasa dahi optimize edilmiş ileri yönlü arama algoritması GA'ya göre daha kısa sürede aramalar gerçekleştirerek topluluğu oluşturmuştur.
- VIBES (RF) algoritmasında özniteliklerin enformasyon miktarını ölçmek için iki yöntem kullanılmıştır. Bunlar: ReliefF ve Shannon entropidir. Bu iki algoritmanın ortalama doğruluk oranları sırasıyla %89.8153 ve %89.0519'dur. Bu nedenle enformasyon ölçüm yöntemi olarak Shannon entropi varsayılan algoritma olarak ayarlanmıştır.
- VIBES (RF) algoritması iki varsayımdan birinin tercih edilmesine göre çalışmaktadır. Bu varsayımlar: Özniteliklerin birbirlerine bağımlı olduğu ve bağımsız olduğu varsayımlardır. Bu varsayımlara göre 33 benchmark veri kümesi üzerindeki ortalama doğruluk oranları sırasıyla %89.80 ve %88.22'dir.

- Bu tez çalışmasında, çoğunluk oylamalı topluluk öğrencilerde tahmini hata oranını bulmak için bir yöntem geliştirilmiştir. Bu yönteme göre 33 adet benchmark veri kümesi üzerinde özniteliklerin bağımlılığı varsayımına göre gerçek ve tahmini hata oranlarının farkı %0.575'tir. Özniteliklerin bağımsızlığı varsayımına göre gerçek ve tahmini hata oranlarının farkı ise %1.655'tir. Sonuç olarak her iki varsayıma göre ortalama hata oranı farkı %1.115'dir. Bu sonuca göre hata tahmin yöntemi gerçek hata oranına oldukça yakın sonuçlar vermiştir.
- Parkinson veri kümesi kullanılarak öznitelik oluşturmak için FFD, Dalgacık dönüşümü ve HHD kullanılmıştır. Bu sinyal dönüşüm yöntemleri içerisinde HHD ile oluşturulmuş öznitelik kümesi üzerinde hem VIBES algoritması hem de diğer Makine Öğrenmesi algoritmaları yüksek doğruluk oranları vermiştir. HHD dışında daubechies (db3 ve db4) dalgacıkları kullanılarak elde edilen öznitelik kümeleri de ikinci en iyi sınıflandırma doğruluğu veren sinyal dönüşüm yöntemleri olmuştur.
- HHD kullanılarak oluşturulan Parkinson veri kümesi VIBES (KA) algoritmasına ve özniteliklerin bağımsızlığı varsayımına göre LOOÇD testi sonucunda %97.5758 sınıflandırma doğruluğu elde edilmiştir. Böylelikle literatürde daha önce yapılan 6 çalışmadan daha yüksek oranda sınıflandırma doğruluğu elde edilmiştir.



KAYNAKLAR

- Abellán, J.**, 2013. Ensembles of decision trees based on imprecise probabilities and uncertainty measures. *Information Fusion*, 14(4), pp.423–430.
- Abuassba, A.O.M. et al.**, 2017. Improving Classification Performance through an Advanced Ensemble Based Heterogeneous Extreme Learning Machines. *Computational Intelligence and Neuroscience*, 2017, pp.1–11.
- Adams, R.D., Victor, M. & Ropper, A.H.**, 1997. *Principles of Neurology* 6th ed., New York, NY: McGraw-Hill.
- El Akadi, A. et al.**, 2011. A two-stage gene selection scheme utilizing MRMR filter and GA wrapper. *Knowledge and Information Systems*, 26(3), pp.487–500.
- Akdemir, Ü.Ö. et al.**, 2014. Brain 18F-FDG PET Imaging in the Differential Diagnosis of Parkinsonism. *Clinical Nuclear Medicine*, 39(3), pp.e220–e226.
- Akyürekli, Ö. & Akyürekli, O.**, 2010. Parkinson Sendromu: Klinik Belirtiler ve Sınıflama. In M. Emre, ed. *Parkinson Hastalığı*. Ankara: Ayrıntı Basımevi, pp. 69–74.
- Alpaydin, E.**, 2014. *Introduction to Machine Learning* 3rd ed., MIT Press.
- Alpaydin, E.**, 1997. Voting over Multiple Condensed Nearest Neighbors. *Artificial Intelligence Review*, 11(1/5), pp.115–132.
- Alpaydin, E. & Kaynak, C.**, 1998. Cascading Classifiers. *Kybernetika*, 34(4), pp.369–374.
- Altın, A.**, 2011. *Fourier Analizi*, Ankara: Gazi Kitabevi.
- Aristotle**, 2004. *On the Gait of Animals*, Kessinger Publishing.
- Arrow, K.J.**, 1950. A Difficulty in the Concept of Social Welfare. *Journal of Political Economy*, 58(4), pp.328–346.
- Asmis, E.**, 1984. *Epicurus' Scientific Method*, Cornell University Press.
- Aydin, F. & Aslan, Z.**, 2017a. Classification of Neurodegenerative Diseases using Machine Learning Methods. *International Journal of Intelligent Systems and Applications in Engineering*, 1(5), pp.1–9.
- Aydin, F. & Aslan, Z.**, 2017b. Diagnosis of neuro degenerative diseases using machine learning methods and wavelet transform. *Journal of the Faculty of Engineering and Architecture of Gazi University*, 32(3), pp.749–766.
- Aydın, S.**, 1999. *Analize Giriş: Birinci Cilt* 7th ed., İstanbul: Beta Yayınları.
- Ayres-de-Campos, D. et al.**, 2000. Sisporto 2.0: A program for automated analysis of cardiocograms. *The Journal of Maternal-Fetal Medicine*, 9(5), pp.311–318.
- Baker, R. et al.**, 2009. The Gait Profile Score and Movement Analysis Profile. *Gait & Posture*, 30(3), pp.265–269.
- Baray, A.**, 2003. Entropi ve Karar Verme. *Yönetim Dergisi*, 14(44), pp.7–21.
- Barnosky, A.D.**, 2010. *Mammal Anatomy: An Illustrated Guide* third., Cavendish Square Publishing.
- Barr, A.E.**, 2002. Biomechanics and Gait. In K. J. Koval, ed. *Orthopaedic Knowledge Update 7: Home Study Syllabus*. Rosemont, Illinois: American Academy Orthopedic Surgeons, pp. 31–38.
- Barton, G. et al.**, 2007. Gait quality assessment using self-organising artificial neural networks. *Gait & Posture*, 25(3), pp.374–379.

- Barut, B.Ö. et al.**, 2012. Quantitative Assessment of Finger-Tapping Performance in Patients With Parkinson's Disease. *Journal of Neurological Sciences (Turkish)*, 29(4), pp.689–697.
- Baudrexel, S. et al.**, 2014. The value of putaminal diffusion imaging versus 18-fluorodeoxyglucose positron emission tomography for the differential diagnosis of the Parkinson variant of multiple system atrophy. *Movement Disorders*, 29(3), pp.380–387.
- Bay, S.D.**, 1999. Nearest neighbor classification from multiple feature subsets. *Intelligent Data Analysis*, 3(3), pp.191–209.
- Beal, M.F., Lang, A.E. & Ludolph, A.C.**, 2010. *Neurodegenerative Diseases: Neurobiology, Pathogenesis and Therapeutics*, Cambridge University Press.
- Bennett, K. & Mangasarian, O.L.**, 1992. Robust linear programming discrimination of two linearly inseparable sets. *Optimization Methods and Software*, 1(1), pp.23–34.
- Bermingham, M.L. et al.**, 2015. Application of high-dimensional feature selection: evaluation for genomic prediction in man. *Scientific Reports*, 5(1), p.10312.
- Beyer, K.S. et al.**, 1999. When Is Nearest Neighbor'' Meaningful? In *ICDT '99 Proceedings of the 7th International Conference on Database Theory*. London, UK: Springer-Verlag, pp. 217–235.
- Bilgiç, B.**, 2010. Bazal Ganglionlar: Anatomi, Fizyoloji ve İşlevleri. In M. Emre, ed. *Parkinson Hastalığı*. Ankara: Ayrıntı Basımevi, pp. 27–35.
- Blum, A.L. & Rivest, R.L.**, 1992. Training a 3-node neural network is NP-complete. *Neural Networks*, 5(1), pp.117–127.
- Boashash, B.**, 1992. Estimating and interpreting the instantaneous frequency of a signal. I. Fundamentals. *Proceedings of the IEEE*, 80(4), pp.520–538.
- Bock, R.K. et al.**, 2004. Methods for multidimensional event classification: a case study using images from a Cherenkov gamma-ray telescope. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 516(2–3), pp.511–528.
- Bogey, R.A., Barnes, L.A. & Perry, J.**, 1993. A computer algorithm for defining the group electromyographic profile from individual gait profiles. *Archives of physical medicine and rehabilitation*, 74(3), pp.286–91.
- Bolón-Canedo, V., Sánchez-Marroño, N. & Alonso-Betanzos, A.**, 2013. A review of feature selection methods on synthetic data. *Knowledge and Information Systems*, 34(3), pp.483–519.
- Bolón-Canedo, V., Sánchez-Marroño, N. & Alonso-Betanzos, A.**, 2012. An ensemble of filters and classifiers for microarray data classification. *Pattern Recognition*, 45(1), pp.531–539.
- Brehm, M., Harlaar, J. & Schwartz, M.**, 2008. Effect of ankle-foot orthoses on walking efficiency and gait in children with cerebral palsy. *Journal of Rehabilitation Medicine*, 40(7), pp.529–534.
- Breiman, L.**, 1996. Bagging predictors. *Machine Learning*, 24(2), pp.123–140.
- Breiman, L.**, 2001. Random Forests. *Machine Learning*, 45(1), pp.5–32.
- Brown, G.**, 2011. Ensemble Learning. In *Encyclopedia of Machine Learning*. Boston, MA: Springer US, pp. 312–320.
- Brown, J.W. & Churchill, R. V.**, 1993. *Fourier Series and Boundary Value Problems* 50th ed., Mc Graw-Hill.
- Castermans, T. et al.**, 2013. Towards Effective Non-Invasive Brain-Computer Interfaces Dedicated to Gait Rehabilitation Systems. *Brain Sciences*, 4(1), pp.1–48.

- Chambers, H.G. & Sutherland, D.H.**, 2002. A practical guide to gait analysis. *The Journal of the American Academy of Orthopaedic Surgeons*, 10(3), pp.222–31.
- Charytanowicz, M. et al.**, 2010. Complete Gradient Clustering Algorithm for Features Analysis of X-Ray Images. In E. Pietka & J. Kawa, eds. *Information Technologies in Biomedicine*. Springer-Verlag Berlin Heidelberg, pp. 15–24.
- Chidlovskii, B. & Lecerf, L.**, 2008. Scalable Feature Selection for Multi-class Problems. In *Machine Learning and Knowledge Discovery in Databases*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 227–240.
- Chinchor, N.**, 1992. MUC-4 evaluation metrics. In *Proceedings of the 4th conference on Message understanding - MUC4 '92*. Morristown, NJ, USA: Association for Computational Linguistics, p. 22.
- Cimolin, V. & Galli, M.**, 2014. Summary measures for clinical gait analysis: A literature review. *Gait & Posture*, 39(4), pp.1005–1010.
- Cochran, G.V.B.**, 1982. *A Primer of Orthopaedic Biomechanics*, Churchill Livingstone.
- Cohen, J.**, 1960. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1), pp.37–46.
- Colominas, M.A., Schlotthauer, G. & Torres, M.E.**, 2014. Improved complete ensemble EMD: A suitable tool for biomedical signal processing. *Biomedical Signal Processing and Control*, 14, pp.19–29.
- Cortes, C. & Vapnik, V.**, 1995. Support-Vector Networks. *Machine Learning*, 20(3), pp.273–297.
- Cortez, P. et al.**, 2009. Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems*, 47(4), pp.547–553.
- Cristianini, N. & Shawe-Taylor, J.**, 2000. *An Introduction to Support Vector Machines and Other Kernel based Learning Methods*, New York, NY, USA: Cambridge University Press.
- Daliri, M.R.**, 2013. Chi-square distance kernel of the gaits for the diagnosis of Parkinson's disease. *Biomedical Signal Processing and Control*, 8(1), pp.66–70.
- Dasarathy, B.V. & Sheela, B.V.**, 1979. A composite classifier system design: Concepts and methodology. *Proceedings of the IEEE*, 67(5), pp.708–713.
- Dash, M. & Liu, H.**, 1997. Feature selection for classification. *Intelligent Data Analysis*, 1(1–4), pp.131–156.
- Daubechies, I.**, 1988. Orthonormal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics*, 41(7), pp.909–996.
- Daubechies, I.**, 1993. Orthonormal Bases of Compactly Supported Wavelets II. Variations on a Theme. *SIAM Journal on Mathematical Analysis*, 24(2), pp.499–519.
- Deligtisch, A. et al.**, 2011. Movement Disorders. In J. C. M. Brust, ed. *Current Diagnosis & Treatment Neurology*. New York, New York, USA: Mc Graw Hill, pp. 201–229.
- Demirci, M.**, 2010. Motor Fonksiyonların Organizasyonu. In M. Emre, ed. *Parkinson Hastalığı*. Ankara: Ayrıntı Basımevi, pp. 1–13.
- Deo, R.C.**, 2015. Machine learning in medicine. *Circulation*, 132(20), pp.1920–1930.
- Deza, E. & Deza, M.M.**, 2009. *Encyclopedia of Distances* first., Berlin, Heidelberg: Springer Berlin Heidelberg.
- DFG Research Training Group**, Use of Information Technologies for Precision Crop Protection. Available at: http://www.precision-crop-protection.uni-bonn.de/gk_research/project.php?project=3_06 [Accessed June 5, 2015].

- Dias, D.B. et al.**, 2009. Hand movement recognition for Brazilian Sign Language: A study using distance-based neural networks. In *2009 International Joint Conference on Neural Networks*. IEEE, pp. 697–704.
- Díez-Pastor, J.-F., García-Osorio, C. & Rodríguez, J.J.**, 2014. Tree ensemble construction using a GRASP-based heuristic and annealed randomness. *Information Fusion*, 20, pp.189–202.
- Doak, J.**, 1992. *An Evaluation of Feature Selection Methods and Their Application to Computer Security*,
- Domingo, C. & Watanabe, O.**, 2000. MadaBoost: A Modification of AdaBoost. *Conference on Computational Learning Theory (COLT)*, pp.180–189.
- Domingos, P.**, 2000. A Unified Bias-Variance Decomposition and its Applications. In *In Proceeding 17th International Conference on Machine Learning*. San Mateo: Morgan Kaufmann, pp. 231–238.
- Dong, L., Frank, E. & Kramer, S.**, 2005. Ensembles of Balanced Nested Dichotomies for Multi-class Problems. In G. J. Jorge A.M., Torgo L., Brazdil P., Camacho R., ed. *Knowledge Discovery in Databases: PKDD 2005*. Springer, Berlin, Heidelberg, pp. 84–95.
- Drotár, P. et al.**, 2014. Analysis of in-air movement in handwriting: A novel marker for Parkinson’s disease. *Computer Methods and Programs in Biomedicine*, 117(3), pp.405–411.
- Drotár, P. et al.**, 2016. Evaluation of handwriting kinematics and pressure for differential diagnosis of Parkinson’s disease. *Artificial Intelligence in Medicine*, 67, pp.39–46.
- Duan, L.L., Clancy, J.P. & Szczesniak, R.D.**, 2016. Bayesian Ensemble Trees (BET) for Clustering and Prediction in Heterogeneous Data. *Journal of Computational and Graphical Statistics*, 25(3), pp.748–761.
- Dumais, S. et al.**, 2009. F-Measure. In *Encyclopedia of Database Systems*. Boston, MA: Springer US, pp. 1147–1147.
- Dunham, M.H.**, 2002. *Data Mining: Introductory and Advanced Topics* 1st ed., Pearson.
- Efron, B. & Tibshirani, R.J.**, 1994. *An Introduction to the Bootstrap* 1st., Chapman and Hall/CRC.
- Egmont-Petersen, M., Dassen, W.R.M. & Reiber, J.H.C.**, 1999. Sequential selection of discrete features for neural networks – A Bayesian approach to building a cascade. *Pattern Recognition Letters*, 20(11–13), pp.1439–1448.
- Emre, M. & Hanağası, H.A.**, 2010. Hareket Bozuklukları: Kavram ve Sınıflama. In M. Emre, ed. *Parkinson Hastalığı*. Ankara: Ayrıntı Basımevi, pp. 55–58.
- Ertuğrul, Ö.F. et al.**, 2016. Detection of Parkinson’s disease by Shifted One Dimensional Local Binary Patterns from gait. *Expert Systems with Applications*, 56, pp.156–163.
- Estrela da Silva, J., Marques de Sá, J.P. & Jossinet, J.**, 2000. Classification of breast tissue by electrical impedance spectroscopy. *Medical & Biological Engineering & Computing*, 38(1), pp.26–30.
- Fang, S.-C., Rajasekera, J.R. & Tsao, H.-S.J.**, 1997. *Entropy Optimization and Mathematical Programming* 1st ed., Boston, MA: Springer US.
- Feng, J. -y. et al.**, 2015. The putaminal abnormalities on 3.0T magnetic resonance imaging: can they separate parkinsonism-predominant multiple system atrophy from Parkinson’s disease? *Acta Radiologica*, 56(3), pp.322–328.
- Ferri, C., Flach, P. & Hernández-Orallo, J.**, 2004. Delegating classifiers. In *Twenty-first international conference on Machine learning - ICML '04*. New

- York, New York, USA: ACM Press, p. 37.
- Fischer, O. & Braune, W.**, 1895. *Der Gang des Menschen: Versuche am unbelasteten und belasteten Menschen* 1st ed., Hirzel Verlag.
- Flandrin, P.**, 1999. *Time-Frequency/Time-Scale Analysis*, Academic Press.
- Flandrin, P., Rilling, G. & Goncalves, P.**, 2004. Empirical Mode Decomposition as a Filter Bank. *IEEE Signal Processing Letters*, 11(2), pp.112–114.
- Foltynie, T., Levwis, S. & Barker, R.A.**, 2003. *Your Questions Answered Parkinson's Disease* 1st ed., London: Churchill Livingstone.
- Fontugne, R. et al.**, 2013. Strip, bind, and search. In *Proceedings of the 12th international conference on Information processing in sensor networks - IPSN '13*. New York, New York, USA: ACM Press, p. 129.
- Frank, E. et al.**, 2002. Racing Committees for Large Datasets. In S. Lang, K. Satoh, & C. H. Smith, eds. *Lecture Notes in Computer Science*. Berlin, Heidelberg: Springer, Berlin, Heidelberg, pp. 153–164.
- Frank, E. et al.**, 1998. Using Model Trees for Classification. *Machine Learning*, 32(1), pp.63–76.
- Frank, E. & Hall, M.**, 2001. A Simple Approach to Ordinal Classification. In L. De Raedt & P. Flach, eds. *Machine Learning: ECML 2001*. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg, pp. 145–156.
- Frank, E. & Kramer, S.**, 2004. Ensembles of nested dichotomies for multi-class problems. In *Twenty-first international conference on Machine learning - ICML '04*. New York, New York, USA: ACM Press, p. 39.
- Freire, A.L. et al.**, 2009. Short-term memory mechanisms in neural network learning of robot navigation tasks: A case study. In *2009 6th Latin American Robotics Symposium (LARS 2009)*. IEEE, pp. 1–6.
- Frenkel-Toledo, S., Giladi, N., Peretz, C., Herman, T., Gruendlinger, L. & Hausdorff, J.M.**, 2005. Effect of gait speed on gait rhythmicity in Parkinson's disease: variability of stride time and swing time respond differently. *Journal of NeuroEngineering and Rehabilitation*, 2(1), p.23.
- Frenkel-Toledo, S., Giladi, N., Peretz, C., Herman, T., Gruendlinger, L. & Hausdorff, J.M.**, 2005. Treadmill walking as an external pacemaker to improve gait rhythm and stability in Parkinson's disease. *Movement Disorders*, 20(9), pp.1109–1114.
- Freund, Y.**, 1995. Boosting a Weak Learning Algorithm by Majority. *Information and Computation*, 121(2), pp.256–285.
- Freund, Y. & Schapire, R.R.E.**, 1996. Experiments with a New Boosting Algorithm. *International Conference on Machine Learning*, pp.148–156.
- Frey, P.W. & Slate, D.J.**, 1991. Letter recognition using Holland-style adaptive classifiers. *Machine Learning*, 6(2), pp.161–182.
- Friedman, J.**, 1999. Additive Logistic Regression: A Statistical View of Boosting. , 10(2), pp.509–524.
- Gabor, D.**, 1946. Theory of communication. Part 1: The analysis of information. *Journal of the Institution of Electrical Engineers - Part III: Radio and Communication Engineering*, 93(26), pp.429–441.
- Gama, J.**, 1998a. Combining classifiers by constructive induction. In *Machine Learning: ECML-98*. pp. 178–189.
- Gama, J.**, 1998b. Local cascade generalization. In *Proceedings of the International Conference on Machine Learning, ICML '98*. pp. 206–214.
- Garcia, J.**, 2014. Machine Learning and Cognitive Systems: The Next Evolution of

- Enterprise Intelligence (Part I). *WIRED Magazine*.
- Gauch, H.G.**, 2002. *Scientific Method in Practice* 1st ed., Cambridge University Press.
- Geman, S., Bienenstock, E. & Doursat, R.**, 1992. Neural Networks and the Bias/Variance Dilemma. *Neural Computation*, 4(1), pp.1–58.
- Gibberd, F.B.**, 1986. The management of Parkinson's disease. *The Practitioner*, 230(1412), pp.139–46.
- Gillich, E. & Lohweg, V.**, 2010. Banknote Authentication. In *1. Jahreskolloquium Bildverarbeitung in der Automation*. pp. 1–8.
- Gödel, K.**, 1931. Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I. *Monatshefte für Mathematik und Physik*, 38–38(1), pp.173–198.
- Goldberger, A.L. et al.**, 2000. PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals. *Circulation*, 101(23), pp.e215–e220.
- Gröchenig, K.**, 2001. *Foundations of Time-Frequency Analysis*, Boston, MA: Birkhäuser Boston.
- Grossmann, A. & Morlet, J.**, 1984. Decomposition of Hardy Functions into Square Integrable Wavelets of Constant Shape. *SIAM Journal on Mathematical Analysis*, 15(4), pp.723–736.
- Gu, S. & Jin, Y.**, 2017. Multi-train: A semi-supervised heterogeneous ensemble classifier. *Neurocomputing*, 249, pp.202–211.
- Guyon, I. et al.**, 2004. Result analysis of the NIPS 2003 feature selection challenge. In *NIPS'04 Proceedings of the 17th International Conference on Neural Information Processing Systems*. pp. 545–552.
- Guyon, I. & Elisseeff, A.**, 2003. An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research*, 3(2003), pp.1157–1182.
- Gwet, K.L.**, 2010. *Handbook of Inter-Rater Reliability* 2nd ed., Advanced Analytics.
- Haar, A.**, 1910. Zur Theorie der orthogonalen Funktionensysteme. *Mathematische Annalen*, 69, pp.331–371.
- Han, J. & van der Baan, M.**, 2013. Empirical mode decomposition for seismic time-frequency analysis. *Geophysics*, 78(2), pp.O9–O19.
- Han, J. & Kamber, M.**, 2006. *Data Mining: Concepts and Techniques* 2nd ed., San Francisco: Morgan Kaufmann.
- Hansen, L.K. & Salamon, P.**, 1990. Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(10), pp.993–1001.
- Harrison, D. & Rubinfeld, D.L.**, 1978. Hedonic housing prices and the demand for clean air. *Journal of Environmental Economics and Management*, 5(1), pp.81–102.
- Hausdorff, J.M. et al.**, 2007. Rhythmic auditory stimulation modulates gait variability in Parkinson's disease. *European Journal of Neuroscience*, 26(8), pp.2369–2375.
- Hillman, S.J. et al.**, 2007. Correlation of the Edinburgh Gait Score With the Gillette Gait Index, the Gillette Functional Assessment Questionnaire, and Dimensionless Speed. *Journal of Pediatric Orthopaedics*, 27(1), pp.7–11.
- Ho, T.K.**, 1998. The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8), pp.832–844.
- Hooshmand, A., Nasser, J. & Siahkoobi, H.R.**, 2012. Seismic data denoising based on the Complete Ensemble Empirical Mode Decomposition. In *Istanbul*

- 2012 - *International Geophysical Conference and Oil & Gas Exhibition*. Society of Exploration Geophysicists and The Chamber of Geophysical Engineers of Turkey, pp. 1–4.
- Huang, N.E.**, 2005. Introduction to the Hilbert-Huang Transform and Its Related Mathematical Problems. In N. E. Huang & S. S. P. Shen, eds. *Hilbert-Huang Transform and Its Applications*. Singapore: World Scientific Publishing, pp. 1–26.
- Huang, N.E. et al.**, 2009. On Instantaneous Frequency. *Advances in Adaptive Data Analysis*, 1(2), pp.177–229.
- Huang, N.E. et al.**, 1998. The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 454(1971), pp.903–995.
- Huang, N.E., Long, S.R. & Shen, Z.**, 1996. The Mechanism for Frequency Downshift in Nonlinear Wave Evolution. *Advances in Applied Mechanics*, 32, p.59–117C.
- Huang, N.E., Shen, Z. & Long, S.R.**, 1999. A New View of Nonlinear Water Waves: The Hilbert Spectrum. *Annual Review of Fluid Mechanics*, 31(1), pp.417–457.
- Huang, N.E. & Wu, Z.**, 2008. A review on Hilbert-Huang transform: Method and its applications to geophysical studies. *Reviews of Geophysics*, 46(2), p.RG2006.
- Huertas-Fernández, I. et al.**, 2015. Machine learning models for the differential diagnosis of vascular parkinsonism and Parkinson's disease using [123I]FP-CIT SPECT. *European Journal of Nuclear Medicine and Molecular Imaging*, 42(1), pp.112–119.
- Hume, D.**, 1739. Book I: The Understanding. In L. A. Selby-Bigge, ed. *A Treatise of Human Nature*. Liberty Fund.
- Inman, V.T., Ralston, H.J. & Todd, F.**, 1989. *Human walking*, Edwin Mellen Pr.
- Ishigami, N. et al.**, 2012. Cerebrospinal fluid proteomic patterns discriminate Parkinson's disease and multiple system atrophy. *Movement Disorders*, 27(7), pp.851–857.
- Jacobs, R.A. et al.**, 1991. Adaptive Mixtures of Local Experts. *Neural Computation*, 3(1), pp.79–87.
- James, G. et al.**, 2013. *An Introduction to Statistical Learning*, New York, NY: Springer New York.
- James, G.M.**, 2003. Variance and Bias for General Loss Functions. *Machine Learning*, 51(2), pp.115–135.
- Jane, Y.N., Nehemiah, H.K. & Arputharaj, K.**, 2016. A Q-backpropagated time delay neural network for diagnosing severity of gait disturbances in Parkinson's disease. *Journal of Biomedical Informatics*, 60, pp.169–176.
- Jesse, S. et al.**, 2012. Signal alterations of the basal ganglia in the differential diagnosis of Parkinson's disease: a retrospective case-controlled MRI data bank analysis. *BMC Neurology*, 12(1), p.163.
- John, G.H. & Langley, P.**, 1995. Estimating continuous distributions in Bayesian classifiers. In *UAI'95 Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*. Montréal, Qué, Canada: Morgan Kaufmann, pp. 338–345.
- Jordan, M.I. & Jacobs, R.A.**, 1994. Hierarchical Mixtures of Experts and the EM Algorithm. *Neural Computation*, 6(2), pp.181–214.
- Joshi, D., Khajuria, A. & Joshi, P.**, 2017. An automatic non-invasive method for

- Parkinson's disease classification. *Computer Methods and Programs in Biomedicine*, 145, pp.135–145.
- Jossinet, J.**, 1996. Variability of impedivity in normal and pathological breast tissue. *Medical & Biological Engineering & Computing*, 34(5), pp.346–350.
- JPND research**, 2017. What is Neurodegenerative Disease? Available at: <http://www.neurodegenerationresearch.eu/about/what/> [Accessed June 18, 2017].
- Kalousis, A., Prados, J. & Hilario, M.**, 2007. Stability of feature selection algorithms: a study on high-dimensional spaces. *Knowledge and Information Systems*, 12(1), pp.95–116.
- Kanatlı, U. et al.**, 2006. Yürüme Analizinin Ortopedik Uygulamaları. *Türk Ortopedi ve Travmatoloji Birliği Derneği Dergisi*, 5(1–2), pp.53–59.
- Kandel, E.R. et al.**, 2012. *Principles of Neural Science* 5th ed., McGraw-Hill Education.
- Kantz, H. & Schreiber, T.**, 2004. *Nonlinear Time Series Analysis* 2nd ed., Cambridge University Press.
- Kearns, M.**, 1988. *Thoughts on Hypothesis Boosting*,
- Kerrigan, D.C. et al.**, 1996. A tool to assess biomechanical gait efficiency; a preliminary clinical study. *American journal of physical medicine & rehabilitation*, 75(1), pp.3–8.
- Khorasani, A. & Daliri, M.R.**, 2014. HMM for Classification of Parkinson's Disease Based on the Raw Gait Data. *Journal of Medical Systems*, 38(12), p.147.
- Kira, K. & Rendell, L.A.**, 1992. The feature selection problem: traditional methods and a new algorithm. In *AAAI'92 Proceedings of the tenth national conference on Artificial intelligence*. San Jose, California, pp. 129–134.
- Kittler, J. et al.**, 1998. On Combining Classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3), pp.226–239.
- Kılıçaslan, Y., Güner, E.S. & Yıldırım, S.**, 2009. Learning-based pronoun resolution for Turkish with a comparative evaluation. *Computer Speech & Language*, 23(3), pp.311–331.
- Kohavi, R. & John, G.H.**, 1997. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1–2), pp.273–324.
- Kohavi, R. & Provost, F.**, 1998. Glossary of Terms. *Machine Learning*, 30(2/3), pp.271–274.
- Kohavi, R. & Wolpert, D.H.**, 1996. Bias plus variance decomposition for zero-one loss functions. In *ICML'96 Proceedings of the Thirteenth International Conference on Machine Learning*. Bari, Italy: Morgan Kaufmann, pp. 275–283.
- Kong, E.B. & Dietterich, T.G.**, 1995. Error-Correcting Output Coding Corrects Bias and Variance. In *In Proceedings of the Twelfth International Conference on Machine Learning*. Morgan Kaufmann, pp. 313–321.
- Kononenko, I.**, 2001. Machine learning for medical diagnosis: history, state of the art and perspective. *Artificial Intelligence in Medicine*, 23(1), pp.89–109.
- Kononenko, I., Šimec, E. & Robnik-Šikonja, M.**, 1997. Overcoming the Myopia of Inductive Learning Algorithms with RELIEFF. *Applied Intelligence*, 7(1), pp.39–55.
- Koppel, M. & Engelson, S.P.**, 1996. Integrating Multiple Classifiers By Finding Their Areas of Expertise. In *AAAI-96 Workshop On Integrating Multiple Learning Models*. pp. 53–58.
- Kumar, V.**, 2014. Feature Selection: A literature Review. *The Smart Computing*

Review, 4(3).

- Kuncheva, L.I.**, 2004. *Combining Pattern Classifiers: Methods and Algorithms* 1st ed., Wiley-Interscience.
- Kuncheva, L.I. & Rodríguez, J.J.**, 2014. A weighted voting framework for classifiers ensembles. *Knowledge and Information Systems*, 38(2), pp.259–275.
- Landis, J.R. & Koch, G.G.**, 1977. The Measurement of Observer Agreement for Categorical Data. *Biometrics*, 33(1), p.159.
- Laplace, P.-S.**, 1818. *Deuxieme supplement a la theorie analytique des probabilites*, Paris: Gauthier-Villars.
- Lee, D.T.L. & Yamamoto, A.**, 1994. Wavelet Analysis: Theory and Applications. *hp journal*, 45(6), pp.44–52.
- Lee, S.-H. & Lim, J.S.**, 2012. Parkinson's disease classification using gait characteristics and wavelet-based feature extraction. *Expert Systems with Applications*, 39(8), pp.7338–7344.
- Lee, S.-M. et al.**, 2014. Differential diagnosis of parkinsonism with visual inspection of posture and gait in the early stage. *Gait & Posture*, 39(4), pp.1138–1141.
- Lehmann, E.L. & Casella, G.**, 1998. *Theory of Point Estimation*, New York: Springer-Verlag.
- Leisch, F.**, 2012. Machine Learning Benchmark Problems. Available at: <https://www.rdocumentation.org/packages/mlbench/versions/2.1-1> [Accessed May 19, 2017].
- Lemarié-Rieusset, P.G. & Meyer, Y.**, 1986. Ondelettes et Bases Hilbertiennes. *Revista Matematica Iberoamericana*, 2, pp.1–18.
- Levine, D., Richards, J. & Whittle, M.W.**, 2012. *Whittle's Gait Analysis* 6th ed., Churchill Livingstone.
- LeWitt, P.A.**, 2000. Parkinson's Disease: Etiologic Considerations. In C. H. Adler & J. E. Ahlskog, eds. *Parkinson's Disease and Movement Disorders*. Totowa, NJ: Humana Press, pp. 91–100.
- Lichman, M.**, 2013. UCI Machine Learning Repository. Available at: <http://archive.ics.uci.edu/ml> [Accessed May 29, 2017].
- Litvan, I.**, 2005. What is an Atypical Parkinsonian Disorder? In I. Litvan, ed. *Atypical Parkinsonian Disorders*. Totowa, NJ: Humana Press, pp. 1–9.
- Liu, H. & Motoda, H.**, 1998. *Feature Selection for Knowledge Discovery and Data Mining*, Boston, MA: Springer US.
- Liu, H. & Yu, L.**, 2005. Toward integrating feature selection algorithms for classification and clustering. *IEEE Transactions on Knowledge and Data Engineering*, 17(4), pp.491–502.
- Liu, Y. & Yao, X.**, 1999. Ensemble learning via negative correlation. *Neural Networks*, 12(10), pp.1399–1404.
- Long, P.M. & Servedio, R.A.**, 2010. Random classification noise defeats all convex potential boosters. *Machine Learning*, 78(3), pp.287–304.
- Loscalzo, S., Yu, L. & Ding, C.**, 2009. Consensus Group Based Stable Feature Selection. In *In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-09)*. Paris, pp. 567–576.
- Lucas, D.D. et al.**, 2013. Failure analysis of parameter-induced simulation crashes in climate models. *Geoscientific Model Development Discussions*, 6(1), pp.585–623.
- Lyon, R.J. et al.**, 2016. Fifty years of pulsar candidate selection: from simple filters to a new principled real-time classification approach. *Monthly Notices of the*

- Royal Astronomical Society*, 459(1), pp.1104–1123.
- MacKay, D.J.C.**, 2003. *Information Theory, Inference and Learning Algorithms*, Cambridge University Press.
- Mallat, S.G.**, 1989a. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7), pp.674–693.
- Mallat, S.G.**, 1989b. Multiresolution approximations and wavelet orthonormal bases of L₂. *Transactions of The American Mathematical Society*, 315, pp.69–87.
- Manciocco, A. et al.**, 2009. The application of Russell and Burch 3R principle in rodent models of neurodegenerative disease: The case of Parkinson’s disease. *Neuroscience & Biobehavioral Reviews*, 33(1), pp.18–32.
- Mangasarian, O.L., Setiono, R. & Wolberg, W.H.**, 1990. Pattern Recognition Via Linear Programming: Theory And Application To Medical Diagnosis. In T. F. Coleman & Y. Li, eds. *In Proceedings of the Workshop on Large-Scale Numerical Optimization*. Philadelphia: SIAM Publications, pp. 22–31.
- Mangasarian, O.L. & Wolberg, W.H.**, 1990. Cancer Diagnosis via Linear Programming. *SIAM News*, 23, pp.1–18.
- Manning, C.D. & Raghavan, P.**, 2009. An Introduction to Information Retrieval. In *Online*. p. 1.
- Mansouri, K. et al.**, 2013. Quantitative Structure–Activity Relationship Models for Ready Biodegradability of Chemicals. *Journal of Chemical Information and Modeling*, 53(4), pp.867–878.
- Maraganore, D.M.**, 2000. Epidemiology and Genetics of Parkinson’s Disease. In C. H. Adler & J. E. Ahlskog, eds. *Parkinson’s Disease and Movement Disorders*. Totowa, NJ: Humana Press, pp. 85–90.
- Maudes, J. et al.**, 2012. Random feature weights for decision tree ensemble construction. *Information Fusion*, 13(1), pp.20–30.
- Mayhua-López, E., Gómez-Verdejo, V. & Figueiras-Vidal, A.R.**, 2015. A new boosting design of Support Vector Machine classifiers. *Information Fusion*, 25, pp.63–71.
- McMulkin, M.L. & MacWilliams, B.A.**, 2008. Intersite variations of the Gillette Gait Index. *Gait & Posture*, 28(3), pp.483–487.
- Melville, P. & Mooney, R.J.**, 2003. Constructing diverse classifier ensembles using artificial training examples. In *In Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI-2003)*. Acapulco, Mexico, pp. 505–510.
- Merz, C.J.**, 1999. Using Correspondence Analysis to Combine Classifiers. *Machine Learning*, 36(1/2), pp.33–58.
- Michie, D., Spiegelhalter, D.J. & Taylor, C.C.**, 2009. *Machine Learning, Neural and Statistical Classification* 2009th ed., Overseas Press.
- Miner, N.E.**, 1998. *An introduction to Wavelet Theory and Analysis*, California.
- Mitchell, T.M.**, 1997. *Machine Learning* 1st ed., McGraw-Hill Education.
- Mitchell, T.M.**, 1980. The Need for Biases in Learning Generalizations. In J. W. Shavlik & T. G. Dietterich, eds. *Readings in Machine Learning*. Morgan Kaufman, pp. 184–191.
- Molina, L.C., Belanche, L. & Nebot, A.**, 2002. Feature selection algorithms: a survey and experimental evaluation. In *2002 IEEE International Conference on Data Mining, 2002. Proceedings*. IEEE Comput. Soc, pp. 306–313.
- Morlet, J.**, 1981. Sampling Theory and Wave Propagation. In *Proc. 51st Annual International Meeting of the Society of Exploration Geophysicists*. Los Angeles.

- Nair, S.R. et al.**, 2013. A decision tree for differentiating multiple system atrophy from Parkinson's disease using 3-T MR imaging. *European Radiology*, 23(6), pp.1459–1466.
- Narendra, P. & Fukunaga, K.**, 1977. A Branch and Bound Algorithm for Feature Subset Selection. *IEEE Transactions on Computers*, C-26(9), pp.917–922.
- Nasibov, F.H. & Kaçar, A.**, 2008. *Analize Giriş: Fonksiyonlar Ve Fonksiyonların Sınıflandırılması*, Ankara: Palme Yayıncılık.
- Navarro-Otano, J. et al.**, 2014. 123I-MIBG cardiac uptake, smell identification and 123I-FP-CIT SPECT in the differential diagnosis between vascular parkinsonism and Parkinson's disease. *Parkinsonism & Related Disorders*, 20(2), pp.192–197.
- Navarro, X., Poree, F. & Carrault, G.**, 2012. ECG removal in preterm EEG combining empirical mode decomposition and adaptive filtering. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 661–664.
- Neto, A.R.R. & Barreto, G.A.**, 2009. On the Application of Ensembles of Classifiers to the Diagnosis of Pathologies of the Vertebral Column: A Comparative Analysis. *IEEE Latin America Transactions*, 7(4), pp.487–496.
- Noordewier, M.O., Towell, G.G. & Shavlik, J.W.**, 1990. Training knowledge-based neural networks to recognize genes in DNA sequences. In *NIPS-3 Proceedings of the 1990 conference on Advances in neural information processing systems 3*. pp. 530–536.
- Novacheck, T.F., Trost, J.P. & Schwartz, M.H.**, 2002. Intramuscular psoas lengthening improves dynamic hip function in children with cerebral palsy. *Journal of pediatric orthopedics*, 22(2), pp.158–64.
- Obermeyer, Z. & Emanuel, E.J.**, 2016. Predicting the Future — Big Data, Machine Learning, and Clinical Medicine. *New England Journal of Medicine*, 375(13), pp.1216–1219.
- Oliveira, L.S., Britto, A.S. & Sabourin, R.**, 2005. Improving cascading classifiers with particle swarm optimization. In *Eighth International Conference on Document Analysis and Recognition (ICDAR '05)*. IEEE, p. 570–574 Vol. 2.
- Ota, M. et al.**, 2013. Differential Diagnosis Tool for Parkinsonian Syndrome Using Multiple Structural Brain Measures. *Computational and Mathematical Methods in Medicine*, 2013, pp.1–10.
- Özaras, N. & Yalçın, S.**, 2002a. Normal Yürüme ve Yürüme Analizi. *Turkish Journal of Physical Medicine and Rehabilitation*, 48(3).
- Özaras, N. & Yalçın, S.**, 2002b. *Yürüme Analizi* 2nd ed., İstanbul: Avrupa Tıp Kitapçılık.
- Özkan, Y.**, 2016. *Veri Madenciliği Yöntemleri* 3rd., Papatya Yayıncılık.
- Pearl, J.**, 1984. *Heuristics: Intelligent Search Strategies for Computer Problem Solving* 1st ed., Addison-Wesley.
- Pearson, K.**, 1901. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine Series 6*, 2(11), pp.559–572.
- Pereira, C.M.M.**, 2015. INFOgain. Available at: <https://github.com/cmmp/infogain> [Accessed June 6, 2017].
- Piccini, P. et al.**, 1999. The role of inheritance in sporadic Parkinson's disease: evidence from a longitudinal study of dopaminergic function in twins. *Annals of neurology*, 45(5), pp.577–82.
- Pilleri, M. et al.**, 2014. Heart rate circadian profile in the differential diagnosis between Parkinson disease and multiple system atrophy. *Parkinsonism &*

- Related Disorders*, 20(2), pp.217–221.
- Playfer, J.R.**, 1997. Parkinson's disease. *Postgraduate Medical Journal*, 73(859), pp.257–264.
- Pollanen, M.S., Dickson, D.W. & Bergeron, C.**, 1993. Pathology and biology of the Lewy body. *Journal of neuropathology and experimental neurology*, 52(3), pp.183–91.
- Příkrylová Vranová, H. et al.**, 2016. Clusterin CSF levels in differential diagnosis of neurodegenerative disorders. *Journal of the Neurological Sciences*, 361, pp.117–121.
- Prochazka, A. & Storek, M.**, 1995. Wavelet transform use for signal classification by self-organizing neural networks. In *1995 Fourth International Conference on Artificial Neural Networks*. Cambridge, UK: The Institution of Engineering and Technology, pp. 209–213.
- Provost, F.**, 2000. Distributed Data Mining: Scaling up and Beyond. In H. Kargupta & P. Chan, eds. *Advances in Distributed and Parallel Knowledge Discovery*. San Francisco: Morgan Kaufmann, pp. 3–28.
- Pudil, P. et al.**, 1992. Multistage pattern recognition with reject option. In *Proceedings., 11th IAPR International Conference on Pattern Recognition. Vol.II. Conference B: Pattern Recognition Methodology and Systems*. IEEE Comput. Soc. Press, pp. 92–95.
- Quinlan, J.R.**, 1993. *C4.5: Programs for Machine Learning* 1st., Morgan Kaufmann.
- Quinlan, J.R.**, 1992. Learning with Continuous Classes. In *Proceedings of Australian Joint Conference on Artificial Intelligence*. Hobart, pp. 343–348.
- Van Rijsbergen, C.J.**, 1979. *Information Retrieval* 2nd ed., Butterworth-Heinemann.
- da Rocha Neto, A.R. et al.**, 2011. Diagnostic of Pathology on the Vertebral Column with Embedded Reject Option. In *Pattern Recognition and Image Analysis*. Springer, Berlin, Heidelberg, pp. 588–595.
- Rodriguez, J.J., Kuncheva, L.I. & Alonso, C.J.**, 2006. Rotation Forest: A New Classifier Ensemble Method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10), pp.1619–1630.
- Rozumalski, A. & Schwartz, M.H.**, 2011. The GDI-Kinetic: A new index for quantifying kinetic deviations from normal gait. *Gait & Posture*, 33(4), pp.730–732.
- Russell, S.J. & Norvig, P.**, 2003. *Artificial Intelligence: A Modern Approach* 2nd ed., New Jersey: Prentice Hall.
- Ruta, D. & Gabrys, B.**, 2005. Classifier selection for majority voting. *Information Fusion*, 6(1), pp.63–81.
- Saeyns, Y., Abeel, T. & Van de Peer, Y.**, 2008. Robust Feature Selection Using Ensemble Feature Selection Techniques. In *Machine Learning and Knowledge Discovery in Databases*. Springer-Verlag Berlin Heidelberg, pp. 313–325.
- Sakar, B.E. et al.**, 2013. Collection and Analysis of a Parkinson Speech Dataset With Multiple Types of Sound Recordings. *IEEE Journal of Biomedical and Health Informatics*, 17(4), pp.828–834.
- Salvatore, C. et al.**, 2014. Machine learning on brain MRI data for differential diagnosis of Parkinson's disease and Progressive Supranuclear Palsy. *Journal of Neuroscience Methods*, 222, pp.230–237.
- Schaffer, C.**, 1994. A Conservation Law for Generalization Performance. In W. W. Cohen & H. Hirsch, eds. *Proceedings of the Eleventh International Machine*

- Learning Conference*. New Brunswick, NJ, pp. 259–265.
- Schapiro, R.E.**, 1990. The strength of weak learnability. *Machine Learning*, 5(2), pp.197–227.
- Schutte, L.M. et al.**, 2000. An index for quantifying deviations from normal gait. *Gait & posture*, 11(1), pp.25–31.
- Schwartz, M.H. & Rozumalski, A.**, 2008. The gait deviation index: A new comprehensive index of gait pathology. *Gait & Posture*, 28(3), pp.351–357.
- Seewald, A.K.**, 2002. How to Make Stacking Better and Faster While Also Taking Care of an Unknown Weakness. In *Nineteenth International Conference on Machine Learning*. pp. 554–561.
- Seewald, A.K. & Fürnkranz, J.**, 2001. An Evaluation of Grading Classifiers. In *Advances in Intelligent Data Analysis*. Springer, Berlin, Heidelberg, pp. 115–124.
- Selekler, K.**, 2010. Alois Alzheimer ve Alzheimer Hastalığı. *Türk Geriatri Dergisi*, 13, pp.9–14.
- Sesmero, M.P. et al.**, 2015. An ensemble approach of dual base learners for multi-class classification problems. *Information Fusion*, 24, pp.122–136.
- Shannon, C.E.**, 1948. A Mathematical Theory of Communication. *Bell System Technical Journal*, 27(3), pp.379–423.
- Shrivastava, P. et al.**, 2017. A survey of nature-inspired algorithms for feature selection to identify Parkinson’s disease. *Computer Methods and Programs in Biomedicine*, 139, pp.171–179.
- Sigillito, V.G. et al.**, 1989. Classification of radar returns from the ionosphere using neural networks. *Johns Hopkins APL Technical Digest*, 10, pp.262–266.
- De Silva, A.M. & Leong, P.H.W.**, 2015. Feature Selection. In *Grammar-Based Feature Generation for Time-Series Prediction*. pp. 13–24.
- Silva, P.F.B., Marçal, A.R.S. & da Silva, R.M.A.**, 2013. Evaluation of Features for Leaf Discrimination. In *Image Analysis and Recognition*. pp. 197–204.
- Song, L., Langfelder, P. & Horvath, S.**, 2013. Random generalized linear model: a highly accurate and interpretable ensemble predictor. *BMC Bioinformatics*, 14(1), p.5.
- Stehman, S. V.**, 1997. Selecting and interpreting measures of thematic classification accuracy. *Remote Sensing of Environment*, 62(1), pp.77–89.
- Štenc Bradvica, I. et al.**, 2015. Transcranial sonography and the pocket smell test in the differential diagnosis between parkinson’s disease and essential tremor. *Neurological Sciences*, 36(8), pp.1403–1410.
- Sun, Y.**, 2007. Iterative RELIEF for Feature Weighting: Algorithms, Theories, and Applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6), pp.1035–1051.
- Sun, Y., Todorovic, S. & Goodison, S.**, 2008. A Feature Selection Algorithm Capable of Handling Extremely Large Data Dimensionality. In *Proceedings of the 2008 SIAM International Conference on Data Mining*. Philadelphia, PA: Society for Industrial and Applied Mathematics, pp. 530–540.
- Sutherland, D.H.**, 1990. Gait analysis in neuromuscular disease. *Instructional course lectures*, 39, pp.333–41.
- Sutherland, D.H. & Valencia, F.**, 1992. Padiatric gait. In J. C. Drennan, ed. *The child’s foot and ankle*. New York: Raven Press, pp. 19–35.
- Tanner, C.M. & Goldman, S.M.**, 1996. Epidemiology of Parkinson’s disease. *Neurologic clinics*, 14(2), pp.317–35.
- Thomason, P. et al.**, 2011. Single-Event Multilevel Surgery in Children with Spastic

- Diplegia. *The Journal of Bone and Joint Surgery-American Volume*, 93(5), pp.451–460.
- Tibarewala, D.N. & Ganguli, S.**, 1979. Use of a gait abnormality index for locomotion efficiency evaluation. *Journal of biomedical engineering*, 1(4), pp.263–4.
- Tong, H.**, 2002. Nonlinear Time Series Analysis Since 1990: Some Personal Reflections. *Acta Mathematicae Applicatae Sinica, English Series*, 18(2), pp.177–184.
- Torres, M.E. et al.**, 2011. A complete ensemble empirical mode decomposition with adaptive noise. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 4144–4147.
- Trost, J.P. et al.**, 2008. Comprehensive short-term outcome assessment of selective dorsal rhizotomy. *Developmental Medicine & Child Neurology*, 50(10), pp.765–771.
- Tsanas, A. et al.**, 2014. Objective Automatic Assessment of Rehabilitative Speech Treatment in Parkinson’s Disease. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 22(1), pp.181–190.
- Tsukamoto, K. et al.**, 2012. Significance of apparent diffusion coefficient measurement for the differential diagnosis of multiple system atrophy, progressive supranuclear palsy, and Parkinson’s disease: evaluation by 3.0-T MR imaging. *Neuroradiology*, 54(9), pp.947–955.
- Tuv, E. et al.**, 2009. Feature Selection with Ensembles, Artificial Variables, and Redundancy Elimination. *Journal of Machine Learning Research*, 10(Jul), pp.1341–1366.
- Ueda, N. & Nakano, R.**, 1996. Generalization error of ensemble estimators. In *Proceedings of International Conference on Neural Networks (ICNN’96)*. IEEE, pp. 90–95.
- Vainer, I.**, 2009. *Obtaining Scalable and Accurate Classification in Large Scale Spatiotemporal Domains*. Bar Ilan University.
- Vapnik, V.N.**, 2000. *The Nature of Statistical Learning Theory*, New York, NY: Springer New York.
- Vaughan, C.L., Davis, B.L. & O’Connor, J.C.**, 1999. *Dynamics of Human Gait* 2nd ed., Cape Town: Kiboho Publishers.
- Wanas, N.M., Dara, R.A. & Kamel, M.S.**, 2006. Adaptive fusion and co-operative training for classifier ensembles. *Pattern Recognition*, 39(9), pp.1781–1794.
- Wang, Y. & Witten, I.H.**, 1997. Induction of Model Trees for Predicting Continuous Classes. In *Proceedings of the poster papers of the European Conference on Machine Learning*. Prague, pp. 128–137.
- Waters, C.H.**, 2008. *Diagnosis and Management of Parkinson’s Disease* 6th ed., New York: Professional Communications.
- Witten, I.H., Frank, E. & Hall, M.A.**, 2011. *Data Mining: Practical Machine Learning Tools and Techniques* 3rd ed., Morgan Kaufmann.
- Wolberg, W.H. & Mangasarian, O.L.**, 1990. Multisurface method of pattern separation for medical diagnosis applied to breast cytology. *Proceedings of the National Academy of Sciences*, 87(23), pp.9193–9196.
- Wolpert, D.H.**, 1992. Stacked generalization. *Neural Networks*, 5(2), pp.241–259.
- Wolpert, D.H.**, 1996. The Lack of A Priori Distinctions Between Learning Algorithms. *Neural Computation*, 8(7), pp.1341–1390.
- Wolpert, D.H. & Macready, W.G.**, 1997. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1), pp.67–82.

- Woods, K., Kegelmeyer, W.P. & Bowyer, K.**, 1997. Combination of multiple classifiers using local accuracy estimates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4), pp.405–410.
- Wu, Z. & Huang, N.E.**, 2009. Ensemble Empirical Mode Decomposition: A Noise-Assisted Data Analysis Method. *Advances in Adaptive Data Analysis*, 1(1), pp.1–41.
- Yeh, I.-C., Yang, K.-J. & Ting, T.-M.**, 2009. Knowledge discovery on RFM model using Bernoulli sequence. *Expert Systems with Applications*, 36(3), pp.5866–5871.
- Yeh, J.-R., Shieh, J.-S. & Huang, N.E.**, 2010. Complementary Ensemble Empirical Mode Decomposition: A Novel Noise Enhanced Data Analysis Method. *Advances in Adaptive Data Analysis*, 2(2), pp.135–156.
- Yogev, G. et al.**, 2005. Dual tasking, gait rhythmicity, and Parkinson's disease: Which aspects of gait are attention demanding? *European Journal of Neuroscience*, 22(5), pp.1248–1256.
- Zanigni, S. et al.**, 2015. The contribution of cerebellar proton magnetic resonance spectroscopy in the differential diagnosis among parkinsonian syndromes. *Parkinsonism & Related Disorders*, 21(8), pp.929–937.
- Zeng, W. et al.**, 2016. Parkinson's disease classification using gait analysis via deterministic learning. *Neuroscience Letters*, 633, pp.268–278.
- Zengin, S.**, 2007. *İhmal Edilmiş Aşıl Tendon Rüptürlerinde Lindholm Yöntemi ve Sonuçlarının Yürüme Analizi ile Değerlendirilmesi*. T.C. Sağlık Bakanlığı Okmeydanı Eğitim ve Araştırma Hastanesi.
- Zhang, H.**, 2004. The Optimality of Naive Bayes. In V. Barr & Z. Markov, eds. *Proceedings of the Seventeenth International Florida Artificial Intelligence Research Society Conference (FLAIRS 2004)*. Miami Beach, Florida, USA: AAAI Press.
- Zhang, J.-G. & Deng, H.-W.**, 2007. Gene selection for classification of microarray data based on the Bayes error. *BMC Bioinformatics*, 8(1), p.370.
- Zhang, Y., Ding, C. & Li, T.**, 2008. Gene selection algorithm by combining reliefF and mRMR. *BMC Genomics*, 9(Suppl 2), p.S27.
- Zhang, Y. & Street, W.N.**, 2008. Bagging with Adaptive Costs. *IEEE Transactions on Knowledge and Data Engineering*, 20(5), pp.577–588.
- Zhou, Z.-H.**, 2009. Ensemble Learning. In *Encyclopedia of Biometrics*. Boston, MA: Springer US, pp. 270–273.
- Zhou, Z.-H.**, 2012. *Ensemble Methods: Foundations and Algorithms* 1st ed., Chapman and Hall/CRC.



EKLER

EK A: Veri ve Yöntem

EK B: Analiz ve Uygulama

EK C: Sonuçlar ve Tartışma





EK A**Çizelge A.1:** Deneklere ait tanımlayıcı bilgiler

ID	Cinsiyet	Yaş	Boy (m)	Ağırlık (kg)	HoehnYahr	UPDRS	UPDRSM	TUAG	Hız (m/sn)
GaPt03	K	82	1.45	50	3	20	10	36.34	-
GaPt04	E	68	1.71	-	2.5	25	8	11	0.642
GaPt05	K	82	1.53	51	2.5	24	5	14.5	0.908
GaPt06	E	72	1.7	82	2	16	13	10.47	0.848
GaPt07	K	53	1.67	54	3	44	22	18.34	0.677
GaPt08	K	68	1.63	57	2	15	8	10.11	1.046
GaPt09	E	69	1.6	68	3	34	17	12.7	0.894
GaPt12	K	59	1.63	67	2	25	7	8.37	1.261
GaPt13	E	70	1.68	53	2	38	21	15.51	0.726
GaPt14	E	56	1.95	105	2	29	19	-	1.369
GaPt15	E	81	-	80	2	33	20	10.32	0.948
GaPt16	E	79	1.7	72	2	18	10	8.74	1.048
GaPt17	E	71	1.82	85	3	44	27	14.81	0.731
GaPt18	K	76	1.52	74	2	29	20	12.72	0.889
GaPt19	K	76	-	55	2	30	22	13.2	1.124
GaPt20	E	81	1.65	75	2	63	36	13.71	0.722
GaPt21	E	80	-	63	3	46	27	14.06	1.166
GaPt22	E	78	1.78	65	2	49	29	-	0.802
GaPt23	K	71	1.6	-	3	51	25	25.01	0.36
GaPt24	E	68	1.72	73	2.5	42	15	11.42	1.255
GaPt25	E	81	1.76	90	2.5	31	18	15.22	1.128
GaPt26	K	78	1.52	60	2.5	24	5	7.27	1.244
GaPt27	E	72	1.8	95	2	21	10	7.88	1.423
GaPt28	E	61	1.79	101	2.5	54	29	13.02	0.987
GaPt29	E	68	1.63	80	2	27	16	10.16	1.092
GaPt30	E	69	1.78	93	2	20	12	9.91	1.064
GaPt31	E	67	1.76	90	2.5	27	13	12.6	0.876
GaPt32	E	63	1.69	75	2	33	24	11.22	1.242
GaPt33	E	63	1.86	80	2.5	42	31	11.97	0.825
JuPt01	E	77	1.83	85	2	15	11	15.5	1.013

Çizelge A.1: (devam) Deneklere ait tanımlayıcı bilgiler

JuPt02	K	72	160	68	2.5	32	21	12.47	0.906
JuPt03	K	74	164	65	2.5	14	10	9.59	0.993
JuPt04	E	70	173	80	2.5	34	19	13.75	0.807
JuPt05	E	74	170	61	2	15	14	11.16	1.234
JuPt06	E	78	175	82	2.5	21	12	16.29	0.832
JuPt07	K	51	160	65	3	24	14	9.65	1.185
JuPt08	K	54	159	70	2	26	16	8.5	1.246
JuPt09	E	61	170	82	2.5	27	10	10.81	1.146
JuPt10	E	69	170	80	3	28	20	10.96	1.218
JuPt11	E	74	175	80	2	24	16	12.81	0.785
JuPt12	K	58	156	56	2.5	21	14	9.85	1.052
JuPt13	E	80	164	60	3	39	23	16.09	0.664
JuPt14	K	67	158	63	2	13	8	10.5	0.906
JuPt15	K	64	162	66	2.5	18	10	10	1.099
JuPt16	K	64	150	63	2	19	16	12.22	0.755
JuPt17	E	64	178	75	2.5	24	18	10.03	1.112
JuPt18	E	82	160	65	3	33	16	25.25	0.413
JuPt19	K	68	160	60	2	17	13	10	1.183
JuPt20	E	64	168	80	2	33	24	11.22	1.212
JuPt21	E	84	173	75	2.5	23	18	15.31	0.616
JuPt22	E	69	170	70	2.5	38	18	8.81	1.178
JuPt23	E	68	160	69	2	19	16	7.53	1.247
JuPt24	E	68	163	75	2.5	40	23	10.66	0.918
JuPt25	E	45	174	68	2	23	14	10.63	1.087
JuPt26	K	66	167	84	2	18	11	12.47	0.94
JuPt27	K	58	170	65	2	16	13	12.67	1.135
JuPt28	K	57	175	65	2.5	43	25	11.43	1.118
JuPt29	K	69	160	61	2.5	25	15	10.49	1.044
SiPt02	K	77	1.58	67	2.5	-	-	-	0.77
SiPt04	E	73	1.7	70	2	27	19	14.26	1.147
SiPt05	E	56	1.72	76	2.5	50	32	10.38	1.276
SiPt07	K	54	1.66	65	2	-	-	10.99	1.088
SiPt08	E	67	1.75	100	2	56	33	10.83	1.187
SiPt09	E	63	1.74	80	2	21	12	7.94	1.352
SiPt10	K	77	1.6	60	2	41	27	14.41	0.935
SiPt12	K	64	1.67	55	2	44	34	10.28	0.988
SiPt13	E	52	1.65	75	2	42	27	10.38	1.107
SiPt14	E	60	1.65	65	2.5	46	32	8.13	1.218

Çizelge A.1: (devam) Deneklere ait tanımlayıcı bilgiler

SiPt15	E	62	1.65	82	2	32	19	10.89	1.031
SiPt16	E	50	1.73	69	2	41	26	9.09	1.139
SiPt17	E	64	1.65	80	2	36	24	11.56	1.215
SiPt18	E	57	1.6	88	2	48	30	14.39	1.117
SiPt19	E	53	1.8	76	2	40	29	11.07	1.298
SiPt20	E	59	1.76	77	2	26	18	13.16	1.032
SiPt21	E	58	1.67	66	2	24	15	10.48	1.151
SiPt22	E	71	1.68	67	2	31	19	11.99	1.021
SiPt23	K	50	1.54	82	2	26	20	11.08	1.088
SiPt24	K	36	1.64	52	2	24	17	9.53	1.192
SiPt25	E	63	1.68	80	2	33	24	11.22	1.167
SiPt27	E	50	1.72	75	2	46	26	11.32	1.259
SiPt28	K	70	1.58	72	2.5	33	18	10.98	1.204
SiPt29	K	72	1.73	85	2.5	70	44	15.51	0.856
SiPt30	E	60	1.64	89	2	15	12	11.55	1.162
SiPt31	K	56	1.6	74	2	39	24	9.97	0.903
SiPt32	E	64	1.66	75	2	32	23	12.83	1.12
SiPt33	E	67	1.76	74	2	28	15	10.57	1.08
SiPt34	E	71	1.8	68	2	37	22	11.63	1.386
SiPt35	E	57	1.74	95	2	42	29	8.77	1.28
SiPt36	K	53	1.58	62	2	52	32	11.27	0.97
SiPt37	K	66	1.7	62	2.5	27	21	7.56	1.01
SiPt38	K	65	1.59	60	2	22	14	10.13	1.07
SiPt39	K	69	1.68	53	2	33	20	13.97	0.88
SiPt40	E	69	1.6	81	2.5	37	24	9.7	1.07
GaCo01	E	66	1.8	83	0	0	0	-	1.075
GaCo02	E	74	1.74	70	0	1	1	-	1.04
GaCo03	E	69	1.8	101	0	0	0	-	1.051
GaCo04	E	86	1.66	65	0	3	3	9.57	1.175
GaCo05	K	75	1.5	88	0	0	0	-	0.92
GaCo06	E	82	1.68	82	0	1	1	9.7	1.121
GaCo07	E	79	1.68	95	0	1	1	-	1.282
GaCo08	K	78	1.58	65	0	1	1	-	0.975
GaCo09	K	78	1.63	64	0	1	1	-	1.249
GaCo10	E	69	1.72	70	0	0	0	10.12	1.164
GaCo11	E	70	1.75	80	0	3	3	7.16	1.515
GaCo12	K	65	1.72	60	0	1	1	7.49	1.389
GaCo13	K	67	1.58	72	0	0	0	8.1	1.211

Çizelge A.1: (devam) Deneklere ait tanımlayıcı bilgiler

GaCo14	K	68	1.7	47	0	2	2	8.11	1.298
GaCo15	K	67	1.7	70	0	0	0	8.37	1.344
GaCo16	K	63	1.55	58	0	0	0	8.84	1.346
GaCo17	E	67	1.62	58	0	0	0	7.19	1.415
GaCo22	E	65	1.72	72	0	0	0	-	1.542
JuCo01	E	78	167	75	-	0	0	10.19	1.089
JuCo02	E	71	175	80	-	3	3	7.16	1.329
JuCo03	K	66	172	60	-	1	1	7.49	1.322
JuCo04	K	68	170	80	-	1	1	9.27	1.239
JuCo05	K	55	159	52	-	0	0	9.09	1.248
JuCo06	K	74	155	74	-	0	0	9.91	0.954
JuCo07	K	62	160	58	-	2	2	8.31	1.332
JuCo08	K	77	170	65	-	0	0	9.22	1.286
JuCo10	K	62	167	78	-	0	0	8.82	1.375
JuCo09	E	65	182	83	-	0	0	7.07	1.051
JuCo11	E	72	171	70	-	0	0	10.6	1.203
JuCo12	K	72	160	61	-	0	0	8.94	1.143
JuCo13	E	61	177	72	-	0	0	9.97	1.17
JuCo14	E	55	180	90	-	0	0	-	1.42
JuCo15	K	53	150	56	-	0	0	-	1.164
JuCo16	E	66	172	83	-	0	0	7.13	1.484
JuCo17	K	65	170	60	-	0	0	7.06	1.54
JuCo18	K	64	163	59	-	0	0	9.85	1.231
JuCo19	K	59	168	59	-	0	0	8.91	1.253
JuCo20	K	67	160	67	-	0	0	13.4	1.031
JuCo21	E	60	185	88	-	0	0	10.45	1.13
JuCo22	K	62	162	75	-	0	0	11.98	1.144
JuCo23	E	68	182	87	-	0	0	12.02	1.17
JuCo24	E	64	170	72	-	0	0	8.84	1.391
JuCo25	E	53	172	85	-	0	0	8.44	1.265
JuCo26	E	60	167	70	-	0	0	8.28	1.349
SiCo01	K	53	1.53	56	-	-	-	12.24	0.906
SiCo03	K	65	1.72	60	-	-	-	7.49	1.371
SiCo04	E	71	1.77	80	-	-	-	7.16	1.427
SiCo05	K	55	1.56	90	-	-	-	10.01	1.07
SiCo06	K	70	1.56	54	-	-	-	9.69	1.263
SiCo07	K	57	1.59	55	-	-	-	9.99	1.348
SiCo08	K	65	1.57	69	-	-	-	12.19	1.191

Çizelge A.1: (devam) Deneklere ait tanımlayıcı bilgiler

SiCo09	E	56	1.72	75	-	-	-	9.47	1.31
SiCo10	E	50	1.7	68	-	-	-	8.79	1.442
SiCo11	K	60	1.6	80	-	-	-	8.97	1.465
SiCo12	K	37	1.61	54	-	-	-	6.23	1.538
SiCo13	E	61	1.73	69	-	-	-	9.97	1.299
SiCo14	E	52	1.64	64	-	-	-	9.77	1.151
SiCo15	K	54	1.69	73	-	-	-	7.68	1.469
SiCo16	E	60	1.85	88	-	-	-	9.28	1.073
SiCo17	K	64	1.65	-	-	-	-	10.44	1.202
SiCo18	K	61	1.56	56	-	-	-	10.41	1.27
SiCo19	E	51	1.67	80	-	-	-	10.04	1.086
SiCo20	E	62	1.72	86	-	-	-	9.7	1.25
SiCo21	E	56	1.78	98	-	-	-	7.63	1.458
SiCo22	E	52	1.66	88	-	-	-	7.63	1.418
SiCo23	E	53	1.7	76	-	-	-	8.94	1.16
SiCo24	E	54	1.7	73	-	-	-	11.05	1.03
SiCo25	E	57	1.76	80	-	-	-	9.16	1.13
SiCo26	E	60	1.74	72	-	-	-	9.2	1
SiCo27	E	67	1.9	95	-	-	-	12.52	1.12
SiCo28	E	61	1.7	72	-	-	-	12.65	0.99
SiCo29	E	53	1.7	87	-	-	-	11.41	1.29
SiCo30	E	63	1.74	82	-	-	-	8.68	1.42

EK A.1 Bayes Hata Oranı

İstatistiksel sınıflandırmada Bayes Hata Oranı, herhangi bir sınıflandırıcı için rastgele bir sonucun mümkün olan en düşük hata oranını verir. Bu hata oranı aynı zamanda indirgenemez hata oranı olarakta isimlendirilir.

(A.1)'de Bayes kuralı gösterilmektedir. Ayrıca x örneklerinin olasılık yoğunluk fonksiyonu (A.2)'de gösterilmektedir. Sonuç olarak bir x örneğini en yüksek sonsal olasılığa sahip bir sınıfa atayan Bayes sınıflandırıcının hata oranı (A.3)'te gösterilmektedir.

$$P(y_i|x) = \frac{P(x|y_i)P(y_i)}{P(x)} \quad (\text{A.1})$$

$$P(x) = \sum_{i=1}^c P(x|y_i)P(y_i) \quad (\text{A.2})$$

$$\text{Bayes Hata Oranı} = 1 - \sum_{i=1}^c \int_{y_i} P(x|y_i)P(y_i) dx \quad (\text{A.3})$$

EK B

Çizelge B.1: Deneylerde kullanılan bayesian öğrenme algoritmaları

No	Algoritmanın WEKA'daki Adı
1	A1DE
2	A2DE
3	BayesNet
4	BayesianLogisticRegression
5	NaiveBayes
6	NaiveBayesSimple
7	NaiveBayesUpdateable

Çizelge B.2: Deneylerde kullanılan fonksiyon öğrenme algoritmaları

No	Algoritmanın WEKA'daki Adı
1	FLDA
2	KernelLogisticRegression
3	LDA
4	libLINEAR
5	libSVM
6	Logistic
7	MLPClassifier
8	MultilayerPerceptron
9	MultilayerPerceptronCS
10	QDA
11	RBFClassifier
12	RBFNetwork
13	SGD
14	SimpleLogistic
15	SMO
16	SPegasos
17	VotedPerceptron

Çizelge B.3: Deneylerde kullanılan örnek/tembel öğrenme algoritmaları

No	Algoritmanın WEKA'daki Adı
1	IBk
2	IbkLG
3	Kstar
4	LocalKnn
5	LWL
6	RseslibKnn



Çizelge B.4: Deneylerde kullanılan çeşitli öğrenme yaklaşımlarını kullanan algoritmalar

No	Algoritmanın WEKA'daki Adı
1	CHIRP
2	FLR
3	HyperPipes
4	VFI



Çizelge B.5: Deneylerde kullanılan kural tabanlı öğrenme algoritmaları

No	Algoritmanın WEKA'daki Adı
1	Conjunctive Rule
2	Decision Table
3	DTNB
4	FURIA
5	JRip
6	MODLEM
7	NNge
8	OLM
9	OneR
10	PART
11	Ridor
12	RoughSet
13	Vfdr

Çizelge B.6: Deneylerde kullanılan karar ağacı öğrenme algoritmaları

No	Algoritmanın WEKA'daki Adı
1	ADTree
2	BFTree
3	CDT
4	Decision Stump
5	Extra Tree
6	FT
7	Hoeffding Tree
8	J48
9	J48 Consolidated
10	J48graft
11	LADTree
12	LMT
13	NBTree
14	Random Forest
15	RandomTree
16	REPTree
17	SimpleCart

Çizelge B.7: Deneylerde kullanılan meta/topluluk öğrenme algoritmaları

No	Algoritmanın WEKA'daki Adı
1	AdaBoostM1
2	Bagging
3	Classification via Clustering
4	Classification via Regression
5	Dagging
6	Decorate
7	END
8	Filtered Classifier
9	Iterative Classifier Optimizer
10	LogitBoost
11	MultiBoostAB
12	MultiClass Classifier
13	MultiClass Classifier Updateable
14	OrdinalClass Classifier
15	RacedIncremental LogitBoost
16	Random Committee
17	Randomizable Filtered Classifier
18	Random SubSpace
19	RealAdaBoost
20	Rotation Forest
21	Threshold Selector
22	Weighted InstancesHandler Wrapper
23	Grading
24	MultiScheme
25	Stacking
26	StackingC
27	Vote

```

function [] = benchmark(data, step, baseLearner, isDependent)
    acc = zeros(step,1);
    err = zeros(step,1);
    est_err1 = zeros(step,1);
    est_err2 = zeros(step,1);
    est_err3 = zeros(step,1);
    est_err4 = zeros(step,1);
    est_err5 = zeros(step,1);

    X = table2array(data(:,1:end-1));
    Y = categorical(data.Class);

    if (strcmp(attrMeasure, 'relief'))
        [ranked, ~] = relieff( X, Y, 1 );
    elseif (strcmp(attrMeasure, 'shannon'))
        [gains, attorder] = computegains(data(:,1:end-1),
data(:,end));
        A = sortrows([gains attorder]);
        ranked = flip(A(:,2));
    else
        return;
    end

    for i=1 : step
        fprintf('%d\n',i);
        [output, e1, e2, e3, eMax, eMin] = VIBES(X, Y, ranked,
baseLearner, 10, isDependent);
        acc(i) = output.CCI*100/output.NOI;
        err(i) = output.ICI*100/output.NOI;
        est_err1(i) = e1;
        est_err2(i) = e2;
        est_err3(i) = e3;
        est_err4(i) = eMax;
        est_err5(i) = eMin;
    end

    fprintf('%2.4f %2.4f %2.4f %2.4f %2.4f %2.4f
%2.4f\n', mean(acc), mean(err), mean(est_err1),...
        mean(est_err2), mean(est_err3), mean(est_err4),
mean(est_err5));
    fprintf('%2.4f %2.4f %2.4f %2.4f %2.4f %2.4f
%2.4f\n', std(acc), std(err), std(est_err1),...
        std(est_err2), std(est_err3), std(est_err4), std(est_err5));
end

```

Şekil B.1: VIBES algoritmasının benchmark veri kümeleri üzerine uygulanması için yazılan fonksiyon

```

% INPUTS:
% data = MxN matrix, where M = observations and N = features
% labels = Mx1 matrix with the true labels of each observation
% k = nearest neighbor value for relieff algorithm
% kFold = fold value for Cross Validation

% OUTPUTS: the classifier's performance summary

%% Version 1.0
function [output, e1, e2, e3, eMax, eMin] = VIBES(X, Y, ranked,
base, fold, isDependent)
    [row, ~] = size(X);
    [hPredictions, predictionError] = generateModels(X, Y, ranked,
base, fold, isDependent );
    [ePrediction, index] = generateEnsemble( Y, hPredictions);
    output = generateStatistics( Y, ePrediction, index );
    %performanceSummary( output );
    [e1, e2, e3, eMax, eMin] = estimatedError(predictionError,
index, row);
end

```

Şekil B.2: VIBES algoritmasının ana yapısı

```

function [prediction, predictionError] = generateModels(X, Y,
ranked, base, fold, isDependent)
    [row, col] = size(X);

    prediction = repmat(Y, 1, col+1);
    predictionError = zeros(1, col+1);

    for i=1 : col+1
        disp(i);
        if(isDependent)
            if(i~=col+1)
                indices = ranked(1:i);
            else
                indices = [ranked(1); ranked(end)];
            end
        else
            if(i~=col+1)
                indices = i;
            else
                indices = 1:col;
            end
        end
        if(strcmp(base, 'knn'))
            Mdl = fitcknn(X(:,indices), Y, 'NumNeighbors', 1);
        elseif(strcmp(base, 'nb'))
            Mdl = fitcnb(X(:,indices), Y, 'DistributionNames',
'normal'); % normal, mn, kernel, mvnm
        elseif(strcmp(base, 'svm'))
            t = templateSVM('Standardize', true, 'KernelFunction',
'linear'); % linear, gaussian, rbf, polynomial
            Mdl = fitcecoc(X(:,indices), Y, 'Learners', t);
        elseif(strcmp(base, 'disc'))
            Mdl = fitcdiscr(X(:,indices), Y, 'DiscrimType',
'linear'); % 'linear', 'pseudolinear', 'diaglinear', 'quadratic',
'diagquadratic', 'pseudoquadratic'
        elseif(strcmp(base, 'tree'))
            Mdl = fitctree(X(:,indices), Y);
        elseif(strcmp(base, 'rf'))
            Mdl = TreeBagger(100, X(:,indices), Y, 'OOBPrediction',
'on');

            elabel = oobPredict(Mdl);
            prediction(:,i) = elabel;
            predictionError(i) = 100-(sum(Y==elabel)*100/row);
            continue;
        end
        CVMdl = crossval(Mdl, 'KFold', fold);
        [elabel, ~] = kfoldPredict(CVMdl);
        prediction(:,i) = elabel;
        predictionError(i) = 100-(sum(Y==elabel)*100/row);
    end
end
end

```

Şekil B.3: Modellerin oluşturulması süreci


```

function [prediction, index] = generateEnsemble( actual,
hPredictions )

    function [ePrediction] = finalHypotesis(prediction)
        class = categories(actual);
        predictedClassCount = countcats(prediction,2);
        [~, I] = max(predictedClassCount, [],2);
        ePrediction = class(I(1:end));
    end

    [~, b] = size(hPredictions);
    index = zeros(1, b);
    accuracyValue = zeros(1, b);

    for i=1 : b
        res = -1;
        for j=1 : b
            if (sum(index == j) > 0)
                continue;
            end

            prediction = finalHypotesis([hPredictions(:, index(1:(i-
1))) hPredictions(:, j)]);
            acc = sum(prediction == actual);
            if (res < acc)
                res = acc;
                accuracyValue(i) = acc;
                index(i) = j;
            end
        end
    end

    [~, ind] = max(accuracyValue);
    index = index(1:ind);
    prediction = categorical(finalHypotesis(hPredictions(:,
index)));
end

```

Şekil B.4: Topluluğun oluşturulma süreci

```

function [prediction, index] = generateEnsemble_Opt( actual,
hPredictions )

function [ePrediction] = finalHypotesis(prediction)
    class = categories(actual);
    predictedClassCount = countcats(prediction,2);
    [~, I] = max(predictedClassCount,[],2);
    ePrediction = class(I(1:end));
end

[~, b] = size(hPredictions);
index = zeros(1, b);
accuracyValue = zeros(1, b);
%% OPTİMİZASYON
% Optimizasyon sınırı belirleniyor
waitCount = 0;
cycle = log(b);

for i=1 : b
    res = -1;
    for j=1 : b
        if (sum(index == j) > 0)
            continue;
        end

        prediction = finalHypotesis([hPredictions(:, index(1:(i-
1))) hPredictions(:, j)]);
        acc = sum(prediction == actual);
        if (res < acc)
            res = acc;
            accuracyValue(i) = acc;
            index(i) = j;
        end
    end

    %% OPTİMİZASYON
    % Optimizasyon Kodları
    if (max(accuracyValue) > accuracyValue(i))
        waitCount = waitCount + 1;
        if (waitCount > cycle)
            break;
        end
    else
        waitCount = 0;
    end
end

[~, ind] = max(accuracyValue);
index = index(1:ind);
prediction = categorical(finalHypotesis(hPredictions(:,
index)));
end

```

Şekil B.5: Topluğun optimize edilerek oluşturulma süreci

```

function [acc, prediction, index] = generateEnsemble_GA( index,
actual, hPredictions )

    function [ePrediction] = finalHypotesis(prediction)
        class = categories(actual);
        predictedClassCount = countcats(prediction,2);
        [~, I] = max(predictedClassCount, [],2);
        ePrediction = class(I(1:end));
    end

    if (sum(index) == 0)
        acc = 0;
    else
        prediction = finalHypotesis(hPredictions(:, find(index)));
        acc = -sum(prediction == actual);
    end
end

options = gaoptimset('PopulationType', 'bitstring');
FitnessFcn=@(x)generateEnsemble_GA(x, Y, hPredictions);
[index2, acc, exitflag, Output] = ga(FitnessFcn, col+1,
options);

```

Şekil B.6: Topluluğun Genetik Algoritmalar kullanılarak oluşturulma süreci

```

function [ output ] = generateStatistics( actual, ePrediction,
Indices )
    Class = categories(actual);
    NumberOfInstanceAsClass = countcats(actual);
    NumberOfClass = length(Class);
    ROC = zeros(NumberOfClass, 1);
    NumberOfInstance = length(actual);
    %Number of Instances
    output.NOI = NumberOfInstance;

    %Set Confusion Matrix
    ConfusionMatrix = confusionmat(actual, ePrediction);
    %Calculated True Positive
    TP = diag(ConfusionMatrix) ./ sum(ConfusionMatrix,2);
    %Calculated False Positive
    FP = (sum(ConfusionMatrix,1)' - diag(ConfusionMatrix)) ./
(length(actual) - sum(ConfusionMatrix,2));
    %Calculated Precision
    Precision = diag(ConfusionMatrix) ./ sum(ConfusionMatrix,1)';
    %Calculated Recall
    Recall = TP;
    %Calculated F-Measure
    F_Measure = (2*Precision.*Recall)./(Precision+Recall);

    %Calculated Cohen's Kappa Statistic
    pA = trace(ConfusionMatrix) / NumberOfInstance;
    pE = sum((sum(ConfusionMatrix,2).*sum(ConfusionMatrix,1)') ./
(NumberOfInstance*NumberOfInstance));
    output.Kappa = (pA-pE)/(1-pE);

    predict = (actual == ePrediction);
    meanPredict = mean(predict);
    sumPredict = sum(predict);
    %Calculated Mean Absolute Error
    output.MAE = 1-meanPredict;
    %Calculated Root Mean Squared Error
    output.RMSE = sqrt(1-meanPredict);
    %Calculated Relative Absolute Error
    output.RAE = sumPredict / sum(abs(meanPredict-double(predict)));
    %Calculated Root Relative Squared Error
    output.RRSE = sqrt(sumPredict / sum((meanPredict-
double(predict)).^2));

    %Calculated TPR as each Classes
    last = 0;
    TPR = zeros(length(actual), NumberOfClass);
    for i = 1 : NumberOfClass
        first = last + 1;
        last = last + NumberOfInstanceAsClass(i);
        row = 1;
        for j = first : last
            TPR(row, i) = sum(ePrediction(first:j) == Class(i)) /
NumberOfInstanceAsClass(i);
            row = row + 1;
        end
        TPR(row:end, i) = TPR(row-1, i);
    end
end

```

Şekil B.7: Topluluğun performans istatistiğinin hesaplanma süreci

```

%Calculated Area Under ROC
for i = 1 : NumberOfClass
    ROC(i) = TP(i) * (1-FP(i)); %trapz(sort(1-
(sum(TPR(:,setdiff(1:NumberOfClass,i)),2) ./ (NumberOfClass-1))),
TPR(:,i));
end

%Detailed Accuracy By Class
Class{NumberOfClass+1} = '';
RowNames = cell(NumberOfClass+1,1);
for i = 1 : NumberOfClass
    RowNames(i) = cellstr(int2str(i));
end
RowNames(end) = cellstr('Weighted Avg. ');

output.DetailedAccuracyByClass = table([TP; mean(TP)], [FP;
mean(FP)], [Precision; mean(Precision)], ...
[Recall; mean(Recall)],
[F_Measure; mean(F_Measure)], [ROC; mean(ROC)], ...
cellstr(Class), ...
'VariableNames', {'TP', 'FP',
'Precision', 'Recall', 'FMeasure', 'ROC', 'Class'}, ...
'RowNames', RowNames); %{'..',
'..', 'Weighted Avg.'});

%Feature Indices
output.Indices = Indices;
%Predicted Output
output.Prediction = ePrediction;
%Correctly Classified Instances
output.CCI = trace(ConfusionMatrix);
%Incorrectly Classified Instances
output.ICI = length(actual) - trace(ConfusionMatrix);
%Confusion Matrix
CM = table;
for i = 1 : NumberOfClass
    CM(:,i) = num2cell(ConfusionMatrix(:,i));
end
CM.Properties.VariableNames = Class(1:end-1);
CM.Properties.RowNames = Class(1:end-1);
output.ConfusionMatrix = CM;
end

```

Şekil B.7: Topluluğun performans istatistiğinin hesaplanma süreci

```

function [] = performanceSummary(output)
    disp('=== Stratified cross-validation ===');
    disp('=== Summary ===');
    disp(' ');
    fprintf('Correctly Classified Instances           %8d           %13.4f
%%\n', output.CCI, output.CCI*100/output.NOI);
    fprintf('Incorrectly Classified Instances       %8d           %13.4f
%%\n', output.ICI, output.ICI*100/output.NOI);
    fprintf('Kappa statistic                           %13.4f\n',
output.Kappa);
    fprintf('Mean absolute error                           %13.4f\n',
output.MAE);
    fprintf('Root mean squared error                       %13.4f\n',
output.RMSE);
    fprintf('Relative absolute error                       %13.4f %%\n',
output.RAE);
    fprintf('Root relative squared error                   %13.4f %%\n',
output.RRSE);
    fprintf('Total Number Of Instances                     %8d\n',
output.NOI);
    disp(' ');
    disp('=== Detailed Accuracy By Class ===');
    disp(' ');
    disp(output.DetailedAccuracyByClass);
    disp(' ');
    disp('=== Confusion Matrix ===');
    disp(' ');
    disp(output.ConfusionMatrix);
end

```

Şekil B.8: Topluluğun performans çıktısı

```

function [e1, e2, e3, eMax, eMin] = estimatedError(predictionError,
index, row)
    error = predictionError(index);
    errorMin = sort(error, 'descend');
    errorMax = sort(error);

    if(mod(length(index),2)==0)
        errorMin = [ errorMin(2:end) errorMin(end)];
        errorMax = [ errorMax(2:end) errorMax(end)];
    end

    for i=1 : length(index)
        if(errorMin(i) == 0)
            errorMin(i) = 1/row;
        end
        if(errorMax(i) == 0)
            errorMax(i) = 1/row;
        end
    end

    eMax = errorMax(1);
    eMin = 1/errorMin(1);
    for i=2 : length(index)
        eMax = eMax + 2^(i-2)*errorMax(i);
        eMin = eMin + 2^(i-2)/errorMin(i);
    end
    eMax = eMax / (2^(length(index)-1));
    eMin = (2^(length(index)-1)) / eMin;

    e1 = mean([eMax; eMin]);
    e2 = geomean([eMax; eMin]);
    e3 = harmmean([eMax; eMin]);
end

```

Şekil B.9: Topluluk öğrenme algoritmalarında tahmini hata hesaplama

```
function [ output_args ] = Stat( signal )
    if(isempty(signal))
        output_args = [NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN
NaN NaN NaN NaN NaN];
    else
        output_args = [std(signal)/mean(signal) max(signal)
min(signal) mean(signal) std(signal) skewness(signal)
kurtosis(signal) median(signal) range(signal) mad(signal, 0)
mad(signal, 1) sum(signal) wentropy(signal, 'shannon')
wentropy(signal, 'log energy') norm(signal, 1) norm(signal)];
    end;
end
```

Şekil B.10: Sinyale ait istatistikler




```

data_files = ...
{'GaCo01_01', 'GaCo02_01', 'GaCo03_01', 'GaCo04_01', 'GaCo05_01',
'GaCo06_01', 'GaCo07_01', 'GaCo08_01', ...
'GaCo09_01', 'GaCo10_01', 'GaCo11_01', 'GaCo12_01', 'GaCo13_01',
'GaCo14_01', 'GaCo15_01', 'GaCo16_01', ...
'GaCo17_01', 'GaCo22_01', ...
'JuCo01_01', 'JuCo02_01', 'JuCo03_01', 'JuCo04_01', 'JuCo05_01',
'JuCo06_01', 'JuCo07_01', 'JuCo08_01', ...
'JuCo09_01', 'JuCo11_01', 'JuCo12_01', 'JuCo13_01', 'JuCo14_01',
'JuCo15_01', 'JuCo16_01', 'JuCo17_01', ...
'JuCo18_01', 'JuCo19_01', 'JuCo20_01', 'JuCo21_01', 'JuCo22_01',
'JuCo23_01', 'JuCo24_01', 'JuCo25_01', ...
'JuCo26_01', ...
'SiCo01_01', 'SiCo03_01', 'SiCo04_01', 'SiCo05_01', 'SiCo06_01',
'SiCo07_01', 'SiCo08_01', 'SiCo09_01', ...
'SiCo10_01', 'SiCo11_01', 'SiCo12_01', 'SiCo13_01', 'SiCo14_01',
'SiCo15_01', 'SiCo16_01', 'SiCo17_01', ...
'SiCo18_01', 'SiCo19_01', 'SiCo20_01', 'SiCo21_01', 'SiCo22_01',
'SiCo23_01', 'SiCo24_01', 'SiCo25_01', ...
'SiCo26_01', 'SiCo27_01', 'SiCo28_01', 'SiCo29_01', 'SiCo30_01',
...
'GaPt03_01', 'GaPt04_01', 'GaPt05_01', 'GaPt06_01', 'GaPt07_01',
'GaPt08_01', 'GaPt09_01', 'GaPt12_01', ...
'GaPt13_01', 'GaPt14_01', 'GaPt15_01', 'GaPt16_01', 'GaPt17_01',
'GaPt18_01', 'GaPt19_01', 'GaPt20_01', ...
'GaPt21_01', 'GaPt22_01', 'GaPt23_01', 'GaPt24_01', 'GaPt25_01',
'GaPt26_01', 'GaPt27_01', 'GaPt28_01', ...
'GaPt29_01', 'GaPt30_01', 'GaPt31_01', 'GaPt32_01', 'GaPt33_01',
...
'JuPt01_01', 'JuPt02_01', 'JuPt03_01', 'JuPt04_01', 'JuPt05_01',
'JuPt06_01', 'JuPt07_01', 'JuPt08_01', ...
'JuPt09_01', 'JuPt10_01', 'JuPt11_01', 'JuPt12_01', 'JuPt13_01',
'JuPt14_01', 'JuPt15_01', 'JuPt16_01', ...
'JuPt17_01', 'JuPt18_01', 'JuPt19_01', 'JuPt20_01', 'JuPt21_01',
'JuPt22_01', 'JuPt23_01', 'JuPt24_01', ...
'JuPt25_01', 'JuPt26_01', 'JuPt27_01', 'JuPt28_01', 'JuPt29_01',
...
'SiPt02_01', 'SiPt04_01', 'SiPt05_01', 'SiPt07_01', 'SiPt08_01',
'SiPt09_01', 'SiPt10_01', 'SiPt12_01', ...
'SiPt13_01', 'SiPt14_01', 'SiPt15_01', 'SiPt16_01', 'SiPt17_01',
'SiPt18_01', 'SiPt19_01', 'SiPt20_01', ...
'SiPt21_01', 'SiPt22_01', 'SiPt23_01', 'SiPt24_01', 'SiPt25_01',
'SiPt27_01', 'SiPt28_01', 'SiPt29_01', ...
'SiPt30_01', 'SiPt31_01', 'SiPt32_01', 'SiPt33_01', 'SiPt34_01',
'SiPt35_01', 'SiPt36_01', 'SiPt37_01', ...
'SiPt38_01', 'SiPt39_01', 'SiPt40_01'};

fs = 100; % 100 samples per second

F1 = zeros(1, 80);
F2 = zeros(1, 80);
F_total = zeros(1, 85860);
features = zeros(length(data_files), 85860);

```

Şekil B.11: Fourier dönüşümü kullanılarak her bir denek için öznitelik oluşturma

```

for i=1 : length(data_files)
    load (strcat('C:\...\', data_files{i}, '.mat'));

    abs_diff = zeros(size(data,1), 153);
    diff = zeros(size(data,1), 153);
    add = zeros(size(data,1), 153);

    m=1;
    for k=2 : 18
        for t=k+1 : 19
            df = data(:, k) - data(:, t);
            abs_diff(:, m) = abs(df);
            diff(:, m) = df;
            add(:, m) = data(:, k) + data(:, t);
            m = m + 1;
        end
    end

    signals = [data(:, 2:end) abs_diff diff add];

    [row, col] = size(signals);
    for k=1 : col
        % Single-sided magnitude spectrum with frequency axis in
Hertz
        % Each bin frequency is separated by fs/N Hertz.
        X_mags = abs(fft(signals(:, k)));
        bin_vals = (0 : row-1);
        fax_Hz = bin_vals*fs/row;
        N_2 = ceil(row/2);

        [pks, locs, w, p] = findpeaks(signals(:, k));
        F1 = [Stat(signals(:, k)) Stat(pks) Stat(locs) Stat(w)
Stat(p)];

        [pks2, locs2, w2, p2] = findpeaks(X_mags(1:N_2));
        F2 = [Stat(X_mags(1:N_2)) Stat(pks2) Stat(locs2) Stat(w2)
Stat(p2)];

        [~, I] = sort(X_mags(1:N_2), 'descend');
        fax_Hz = fax_Hz(I(1:20));

        F_total((k-1)*180+1 : (k-1)*180+180) = [F1 F2 fax_Hz(1)
fax_Hz(2) fax_Hz(3) fax_Hz(20) Stat(fax_Hz)];
    end

    features(i, :) = F_total;
end;

```

Şekil B.11: Fourier dönüşümü kullanılarak her bir denek için öznelik oluşturma

```

clc;
clear all;

data_files = ...
{'GaCo01_01', 'GaCo02_01', 'GaCo03_01', 'GaCo04_01', 'GaCo05_01',
'GaCo06_01', 'GaCo07_01', 'GaCo08_01', ...
'GaCo09_01', 'GaCo10_01', 'GaCo11_01', 'GaCo12_01', 'GaCo13_01',
'GaCo14_01', 'GaCo15_01', 'GaCo16_01', ...
'GaCo17_01', 'GaCo22_01', ...
'JuCo01_01', 'JuCo02_01', 'JuCo03_01', 'JuCo04_01', 'JuCo05_01',
'JuCo06_01', 'JuCo07_01', 'JuCo08_01', ...
'JuCo09_01', 'JuCo11_01', 'JuCo12_01', 'JuCo13_01', 'JuCo14_01',
'JuCo15_01', 'JuCo16_01', 'JuCo17_01', ...
'JuCo18_01', 'JuCo19_01', 'JuCo20_01', 'JuCo21_01', 'JuCo22_01',
'JuCo23_01', 'JuCo24_01', 'JuCo25_01', ...
'JuCo26_01', ...
'SiCo01_01', 'SiCo03_01', 'SiCo04_01', 'SiCo05_01', 'SiCo06_01',
'SiCo07_01', 'SiCo08_01', 'SiCo09_01', ...
'SiCo10_01', 'SiCo11_01', 'SiCo12_01', 'SiCo13_01', 'SiCo14_01',
'SiCo15_01', 'SiCo16_01', 'SiCo17_01', ...
'SiCo18_01', 'SiCo19_01', 'SiCo20_01', 'SiCo21_01', 'SiCo22_01',
'SiCo23_01', 'SiCo24_01', 'SiCo25_01', ...
'SiCo26_01', 'SiCo27_01', 'SiCo28_01', 'SiCo29_01', 'SiCo30_01',
...
'GaPt03_01', 'GaPt04_01', 'GaPt05_01', 'GaPt06_01', 'GaPt07_01',
'GaPt08_01', 'GaPt09_01', 'GaPt12_01', ...
'GaPt13_01', 'GaPt14_01', 'GaPt15_01', 'GaPt16_01', 'GaPt17_01',
'GaPt18_01', 'GaPt19_01', 'GaPt20_01', ...
'GaPt21_01', 'GaPt22_01', 'GaPt23_01', 'GaPt24_01', 'GaPt25_01',
'GaPt26_01', 'GaPt27_01', 'GaPt28_01', ...
'GaPt29_01', 'GaPt30_01', 'GaPt31_01', 'GaPt32_01', 'GaPt33_01',
...
'JuPt01_01', 'JuPt02_01', 'JuPt03_01', 'JuPt04_01', 'JuPt05_01',
'JuPt06_01', 'JuPt07_01', 'JuPt08_01', ...
'JuPt09_01', 'JuPt10_01', 'JuPt11_01', 'JuPt12_01', 'JuPt13_01',
'JuPt14_01', 'JuPt15_01', 'JuPt16_01', ...
'JuPt17_01', 'JuPt18_01', 'JuPt19_01', 'JuPt20_01', 'JuPt21_01',
'JuPt22_01', 'JuPt23_01', 'JuPt24_01', ...
'JuPt25_01', 'JuPt26_01', 'JuPt27_01', 'JuPt28_01', 'JuPt29_01',
...
'SiPt02_01', 'SiPt04_01', 'SiPt05_01', 'SiPt07_01', 'SiPt08_01',
'SiPt09_01', 'SiPt10_01', 'SiPt12_01', ...
'SiPt13_01', 'SiPt14_01', 'SiPt15_01', 'SiPt16_01', 'SiPt17_01',
'SiPt18_01', 'SiPt19_01', 'SiPt20_01', ...
'SiPt21_01', 'SiPt22_01', 'SiPt23_01', 'SiPt24_01', 'SiPt25_01',
'SiPt27_01', 'SiPt28_01', 'SiPt29_01', ...
'SiPt30_01', 'SiPt31_01', 'SiPt32_01', 'SiPt33_01', 'SiPt34_01',
'SiPt35_01', 'SiPt36_01', 'SiPt37_01', ...
'SiPt38_01', 'SiPt39_01', 'SiPt40_01'};

wname = 'gaus2';
level = 6;

fs = 100; % 100 samples per second

```

Şekil B.12: Gaus ana dalgacığı ile Dalgacık dönüşümü yapılarak her bir denek için öznitelik oluşturma

```

F1 = zeros(1, 80);
F2 = zeros(1, 80);
F_total = zeros(1, 92061);
features = zeros(length(data_files), 92061);

for i=1 : length(data_files)
    load (strcat('C:\...\', data_files{i}, '.mat'));

    abs_diff = zeros(size(data,1), 153);
    diff = zeros(size(data,1), 153);
    add = zeros(size(data,1), 153);

    m=1;
    for k=2 : 18
        for t=k+1 : 19
            df = data(:, k) - data(:, t);
            abs_diff(:, m) = abs(df);
            diff(:, m) = df;
            add(:, m) = data(:, k) + data(:, t);
            m = m + 1;
        end
    end

    signals = [data(:, 2:end) abs_diff diff add];

    [row, col] = size(signals);
    for k=1 : col
        C = cwt(signals(:, k), 1:fs, wname);
        D = mean(C);

        [hexp, tauq] = wtmm(signals(:, k));
        wcoh = wcoherence(signals(:, k), data(:, 1));
        wcoh2 = mean(wcoh);

        [pks, locs, w, p] = findpeaks(signals(:, k));
        F1 = [Stat(signals(:, k)) Stat(pks) Stat(locs) Stat(w)
Stat(p)];

        [pks2, locs2, w2, p2] = findpeaks(D);
        F2 = [Stat(D) Stat(pks2) Stat(locs2) Stat(w2) Stat(p2)];

        F_total((k-1)*193+1 : (k-1)*193+193) = [F1 F2 hexp
Stat(tauq) Stat(wcoh2)];
    end

    features(i, :) = F_total;
end;

```

Şekil B.12: Gaus ana dalgacı ile Dalgacık dönüşümü yapılarak her bir denek için öznitelik oluşturma

```

clc;
clear all;

data_files = ...
{'GaCo01_01', 'GaCo02_01', 'GaCo03_01', 'GaCo04_01', 'GaCo05_01',
'GaCo06_01', 'GaCo07_01', 'GaCo08_01', ...
'GaCo09_01', 'GaCo10_01', 'GaCo11_01', 'GaCo12_01', 'GaCo13_01',
'GaCo14_01', 'GaCo15_01', 'GaCo16_01', ...
'GaCo17_01', 'GaCo22_01', ...
'JuCo01_01', 'JuCo02_01', 'JuCo03_01', 'JuCo04_01', 'JuCo05_01',
'JuCo06_01', 'JuCo07_01', 'JuCo08_01', ...
'JuCo09_01', 'JuCo11_01', 'JuCo12_01', 'JuCo13_01', 'JuCo14_01',
'JuCo15_01', 'JuCo16_01', 'JuCo17_01', ...
'JuCo18_01', 'JuCo19_01', 'JuCo20_01', 'JuCo21_01', 'JuCo22_01',
'JuCo23_01', 'JuCo24_01', 'JuCo25_01', ...
'JuCo26_01', ...
'SiCo01_01', 'SiCo03_01', 'SiCo04_01', 'SiCo05_01', 'SiCo06_01',
'SiCo07_01', 'SiCo08_01', 'SiCo09_01', ...
'SiCo10_01', 'SiCo11_01', 'SiCo12_01', 'SiCo13_01', 'SiCo14_01',
'SiCo15_01', 'SiCo16_01', 'SiCo17_01', ...
'SiCo18_01', 'SiCo19_01', 'SiCo20_01', 'SiCo21_01', 'SiCo22_01',
'SiCo23_01', 'SiCo24_01', 'SiCo25_01', ...
'SiCo26_01', 'SiCo27_01', 'SiCo28_01', 'SiCo29_01', 'SiCo30_01',
...
'GaPt03_01', 'GaPt04_01', 'GaPt05_01', 'GaPt06_01', 'GaPt07_01',
'GaPt08_01', 'GaPt09_01', 'GaPt12_01', ...
'GaPt13_01', 'GaPt14_01', 'GaPt15_01', 'GaPt16_01', 'GaPt17_01',
'GaPt18_01', 'GaPt19_01', 'GaPt20_01', ...
'GaPt21_01', 'GaPt22_01', 'GaPt23_01', 'GaPt24_01', 'GaPt25_01',
'GaPt26_01', 'GaPt27_01', 'GaPt28_01', ...
'GaPt29_01', 'GaPt30_01', 'GaPt31_01', 'GaPt32_01', 'GaPt33_01',
...
'JuPt01_01', 'JuPt02_01', 'JuPt03_01', 'JuPt04_01', 'JuPt05_01',
'JuPt06_01', 'JuPt07_01', 'JuPt08_01', ...
'JuPt09_01', 'JuPt10_01', 'JuPt11_01', 'JuPt12_01', 'JuPt13_01',
'JuPt14_01', 'JuPt15_01', 'JuPt16_01', ...
'JuPt17_01', 'JuPt18_01', 'JuPt19_01', 'JuPt20_01', 'JuPt21_01',
'JuPt22_01', 'JuPt23_01', 'JuPt24_01', ...
'JuPt25_01', 'JuPt26_01', 'JuPt27_01', 'JuPt28_01', 'JuPt29_01',
...
'SiPt02_01', 'SiPt04_01', 'SiPt05_01', 'SiPt07_01', 'SiPt08_01',
'SiPt09_01', 'SiPt10_01', 'SiPt12_01', ...
'SiPt13_01', 'SiPt14_01', 'SiPt15_01', 'SiPt16_01', 'SiPt17_01',
'SiPt18_01', 'SiPt19_01', 'SiPt20_01', ...
'SiPt21_01', 'SiPt22_01', 'SiPt23_01', 'SiPt24_01', 'SiPt25_01',
'SiPt27_01', 'SiPt28_01', 'SiPt29_01', ...
'SiPt30_01', 'SiPt31_01', 'SiPt32_01', 'SiPt33_01', 'SiPt34_01',
'SiPt35_01', 'SiPt36_01', 'SiPt37_01', ...
'SiPt38_01', 'SiPt39_01', 'SiPt40_01'};

wname = 'bior3.3'; %
haar, dmey, db2, db3, db4, db5, db6, db7, sym2, coif2, bior3.3
level = 6;

```

Şekil B.13: Ayrık Dalgacık dönüşümü yapılarak her bir denek için öznelilik oluşturma

```

fs = 100; % 100 samples per second

F1 = zeros(1, 80);
F2 = zeros(1, 80*level);
F_total = zeros(1, 278091);
features = zeros(length(data_files), 278091);

for i=1 : length(data_files)
    load (strcat('C:\...\', data_files{i}, '.mat'));

    abs_diff = zeros(size(data,1), 153);
    diff = zeros(size(data,1), 153);
    add = zeros(size(data,1), 153);

    m=1;
    for k=2 : 18
        for t=k+1 : 19
            df = data(:, k) - data(:, t);
            abs_diff(:, m) = abs(df);
            diff(:, m) = df;
            add(:, m) = data(:, k) + data(:, t);
            m = m + 1;
        end
    end

    signals = [data(:, 2:end) abs_diff diff add];

    [row, col] = size(signals);
    for k=1 : col
        [C,L] = wavedec(signals(:, k), level, wname);
        D = detcoef(C,L,1:level);

        [pks, locs, w, p] = findpeaks(signals(:, k));
        F1 = [Stat(signals(:, k)) Stat(pks) Stat(locs) Stat(w)
Stat(p)];

        for j=1 : level
            D_level = cell2mat(D(:, j));
            [pks2, locs2, w2, p2] = findpeaks(D_level);
            F2((j-1)*80+1 : (j-1)*80+80) = [Stat(D_level)
Stat(pks2) Stat(locs2) Stat(w2) Stat(p2)];
        end

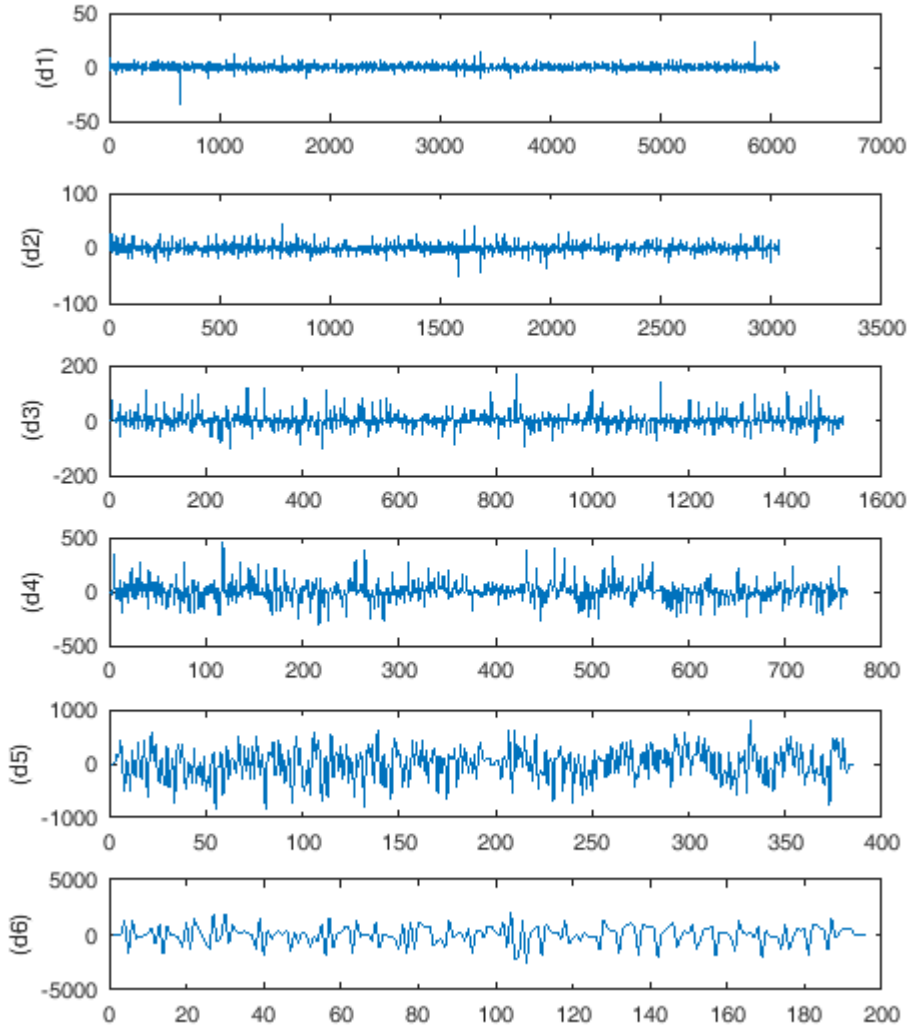
        [Ea,Ed] = wenergy(C,L);

        F_total((k-1)*583+1 : (k-1)*583+583) = [F1 F2 Ea Ed(1)
Ed(2) Ed(3) Ed(4) Ed(5) Ed(6) Stat(Ed)];
    end

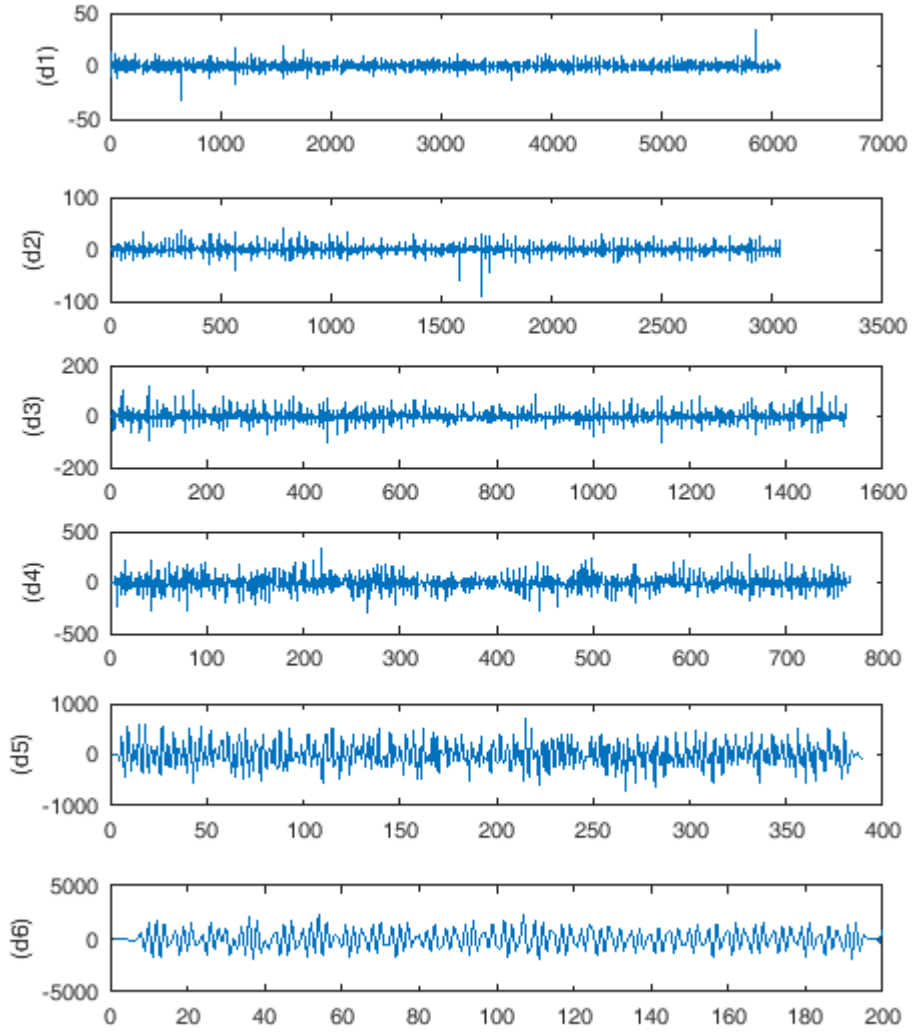
    features(i, :) = F_total;
end;

```

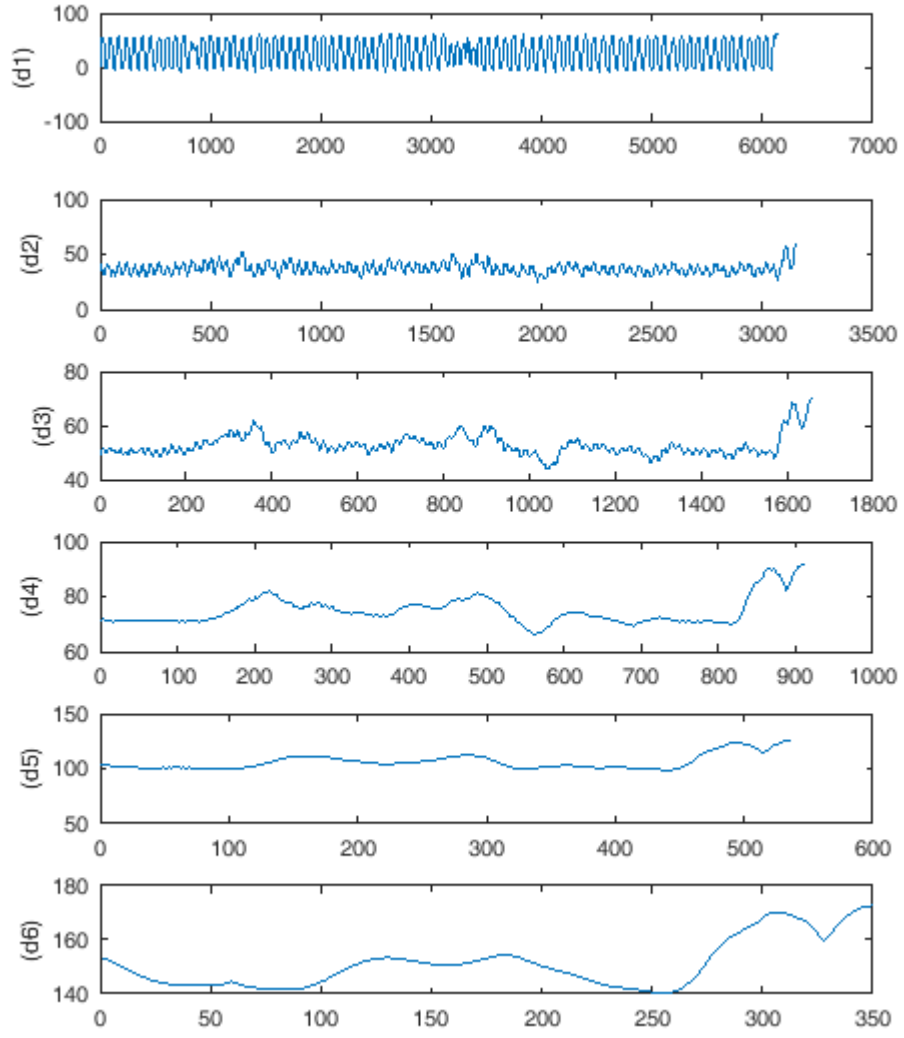
Şekil B.13: Ayrık Dalgacık dönüşümü yapılarak her bir denek için öznelilik oluşturma



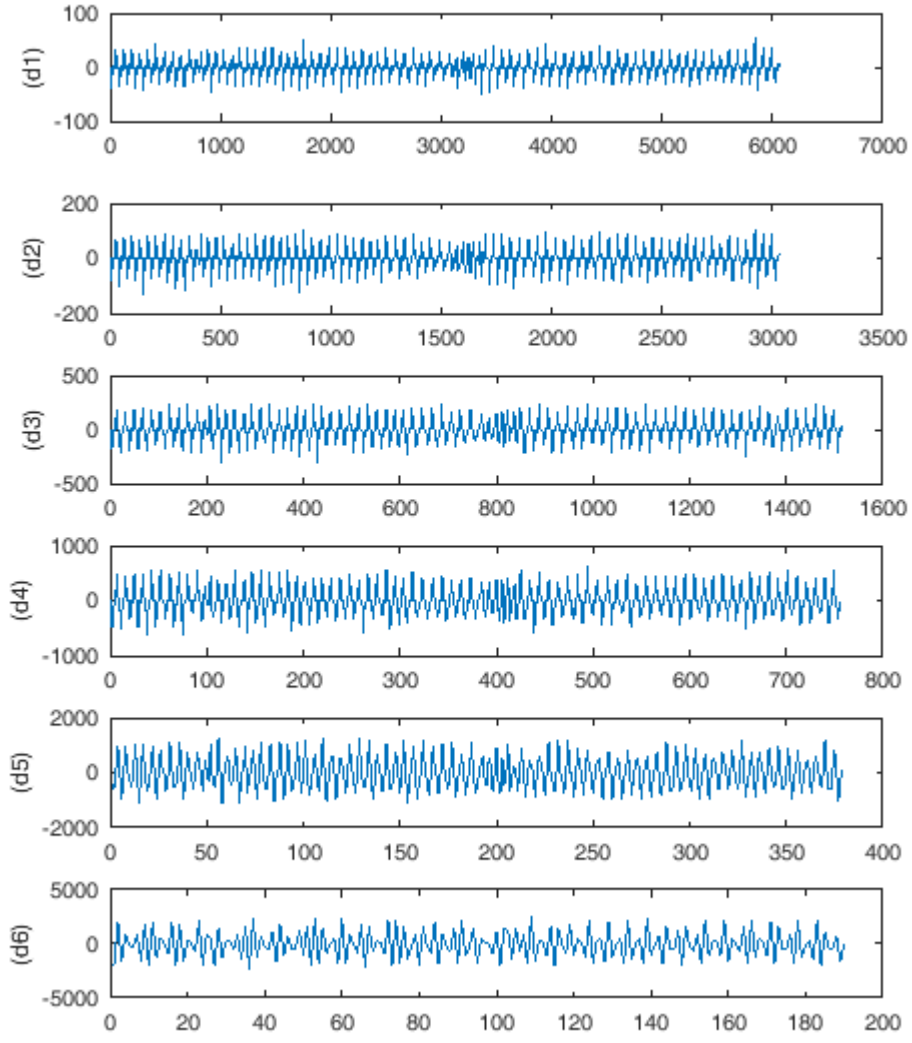
Şekil B.14: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için BiorSplines3.3 dalgacığının 6. seviyeye kadar detay katsayıları



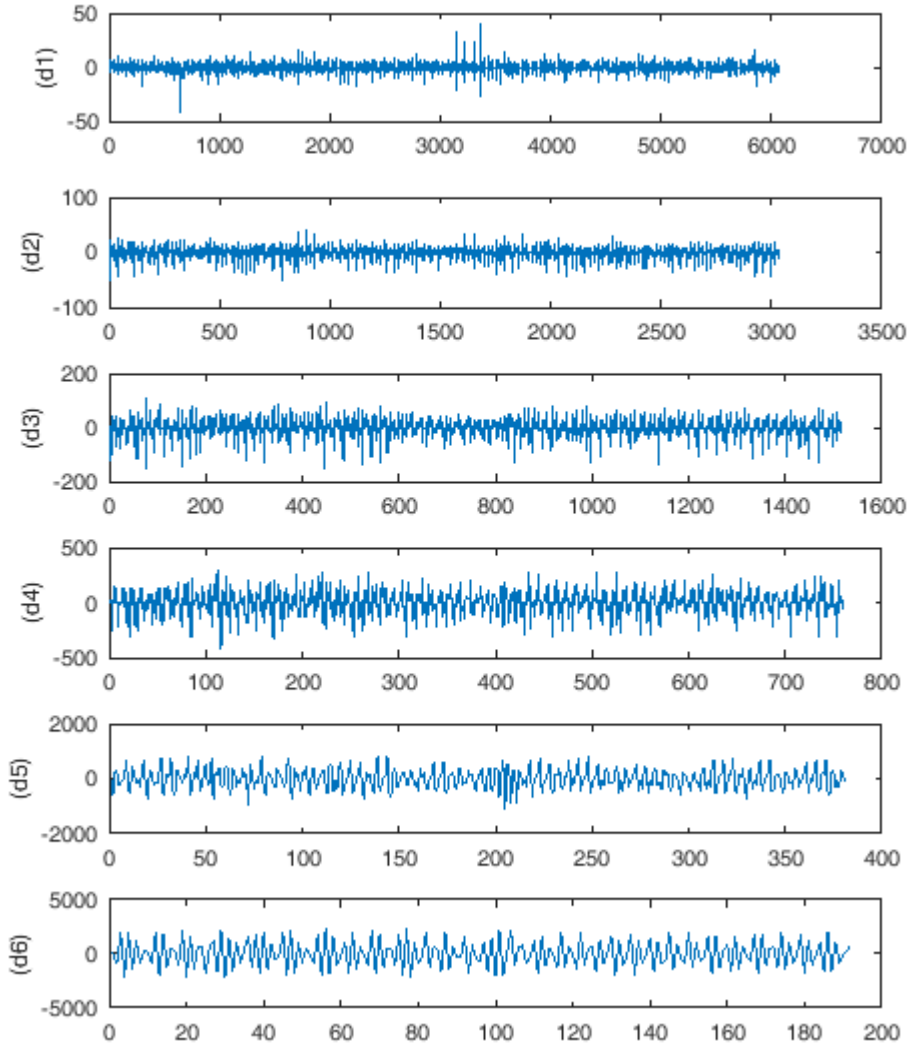
Şekil B.15: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için Coiflets2 dalgacığının 6. seviyeye kadar detay katsayıları



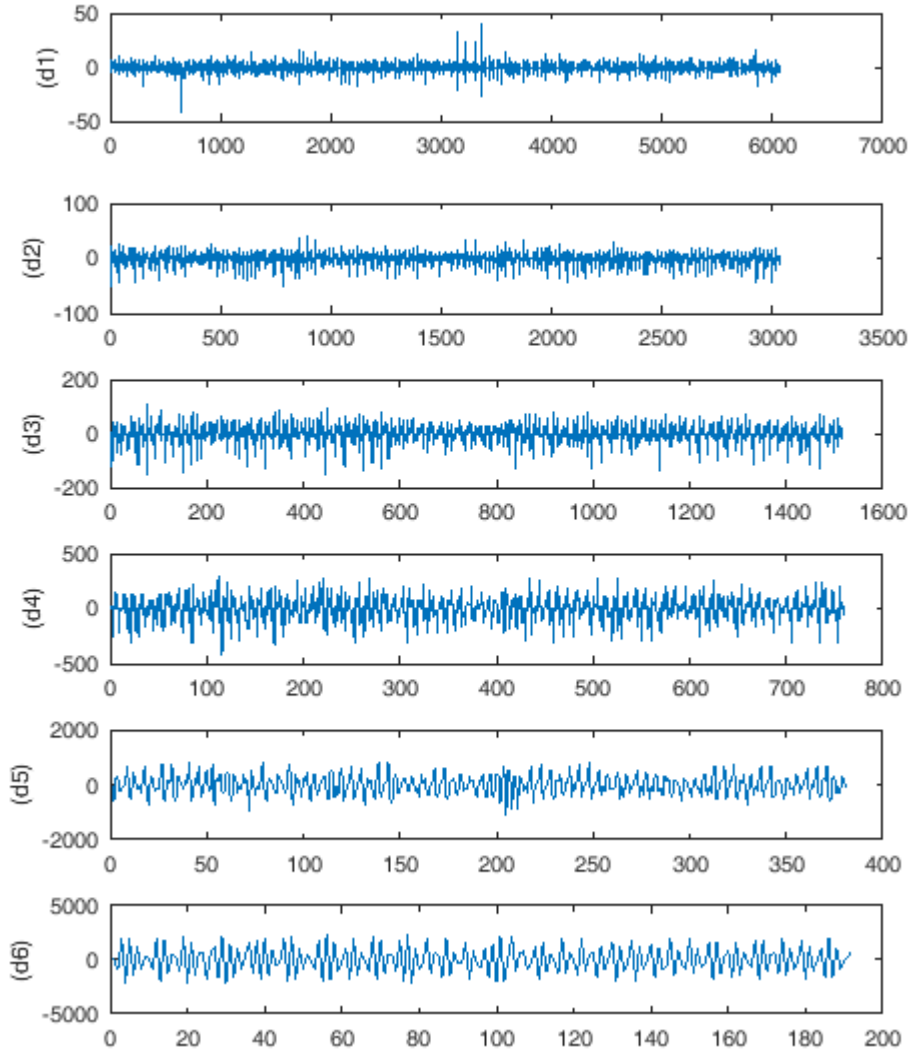
Şekil B.16: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için DMeyer dalgacığının 6. seviyeye kadar detay katsayıları



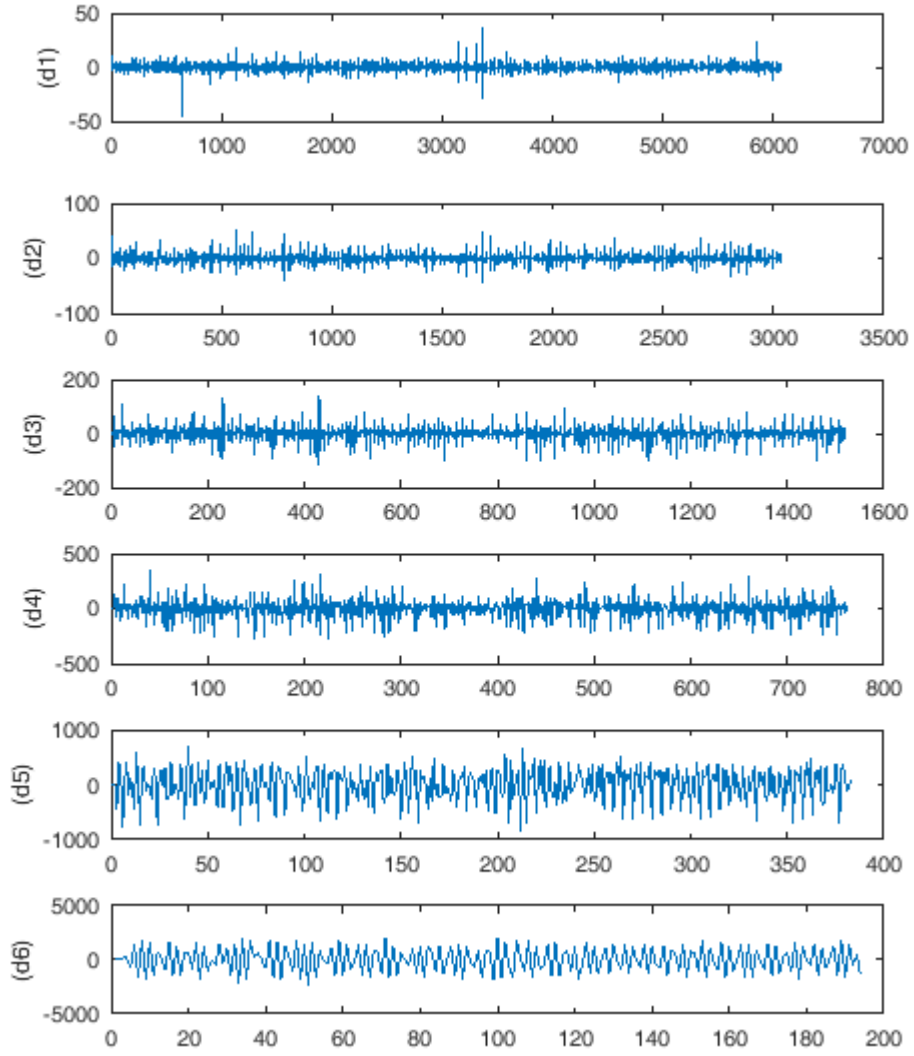
Şekil B.17: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için Haar dalgacığının 6. seviyeye kadar detay katsayıları



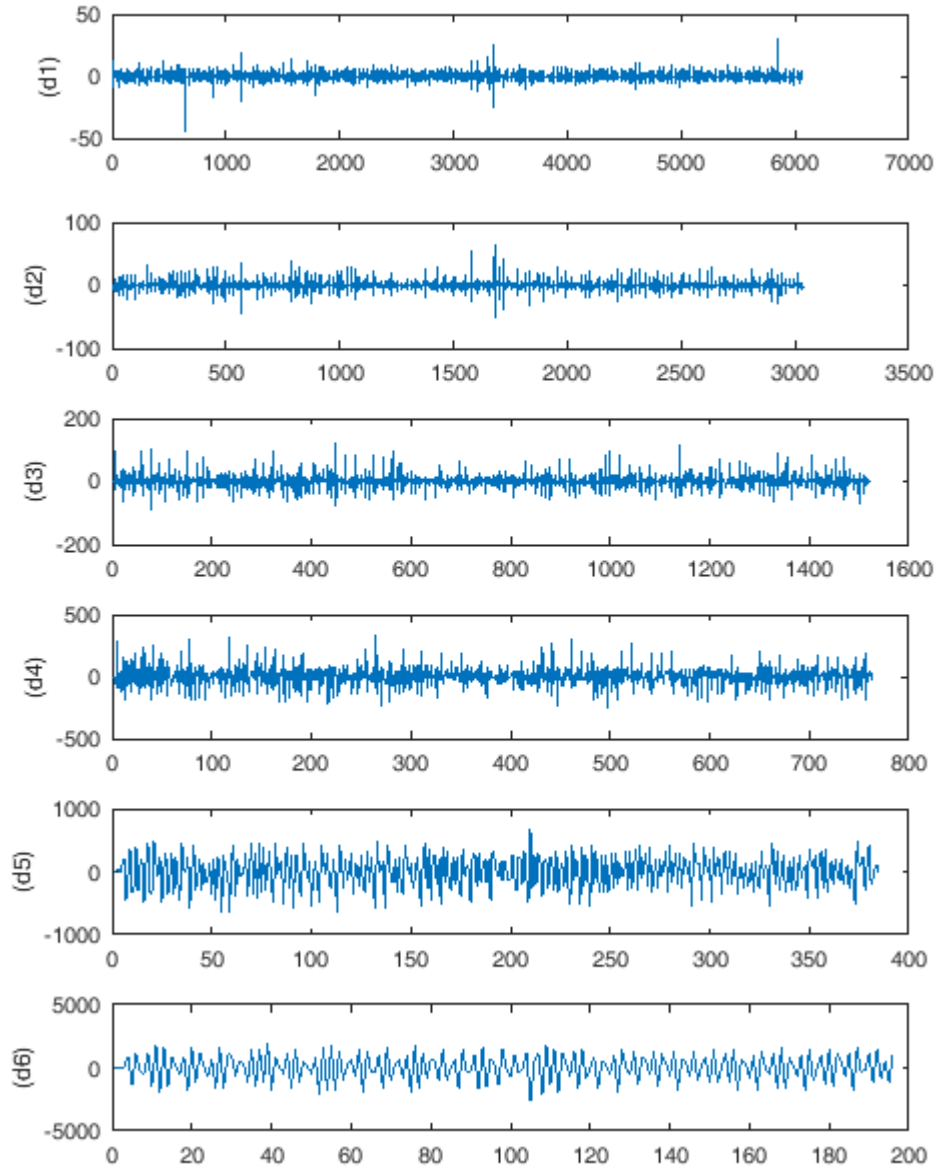
Şekil B.18: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için Symlets2 dalgacığının 6. seviyeye kadar detay katsayıları



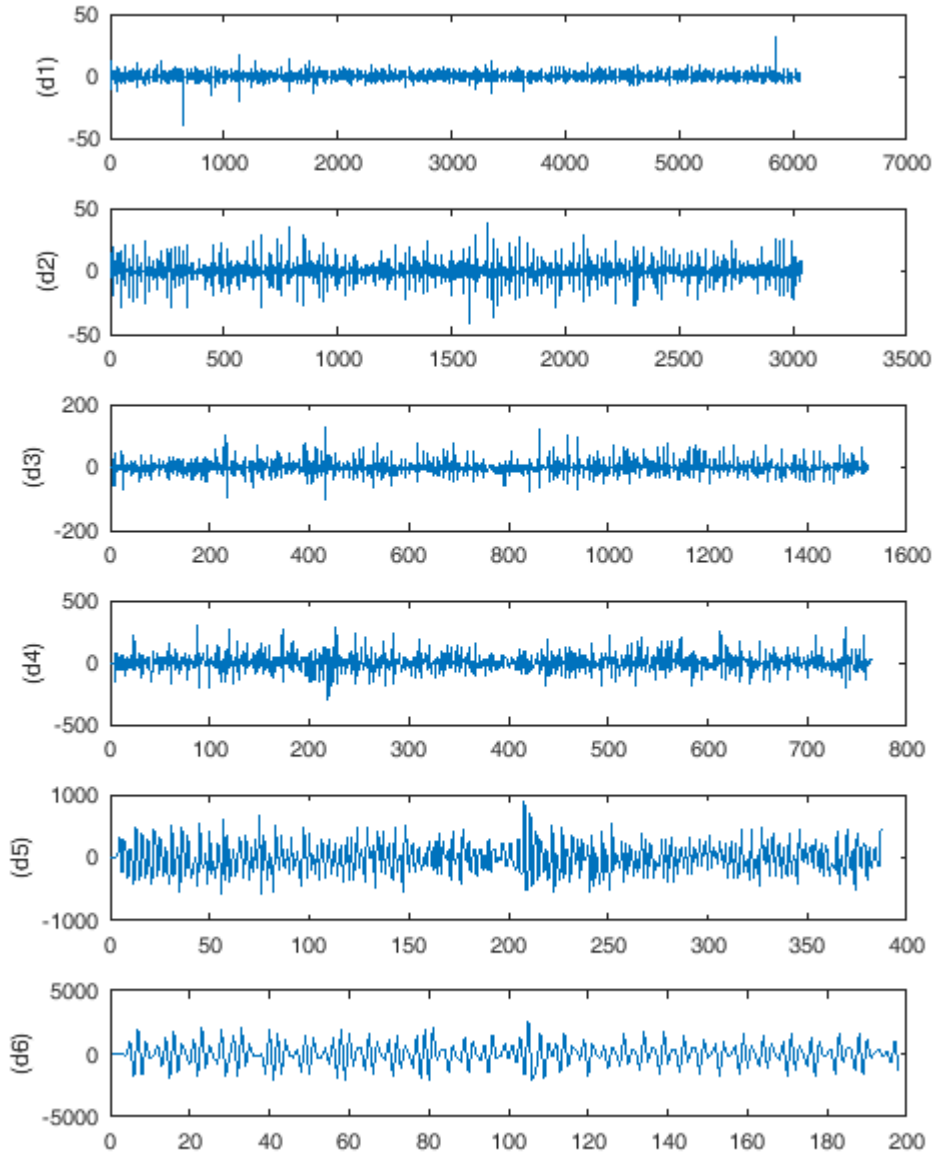
Şekil B.19: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için Daubechies2 dalgacığının 6. seviyeye kadar detay katsayıları



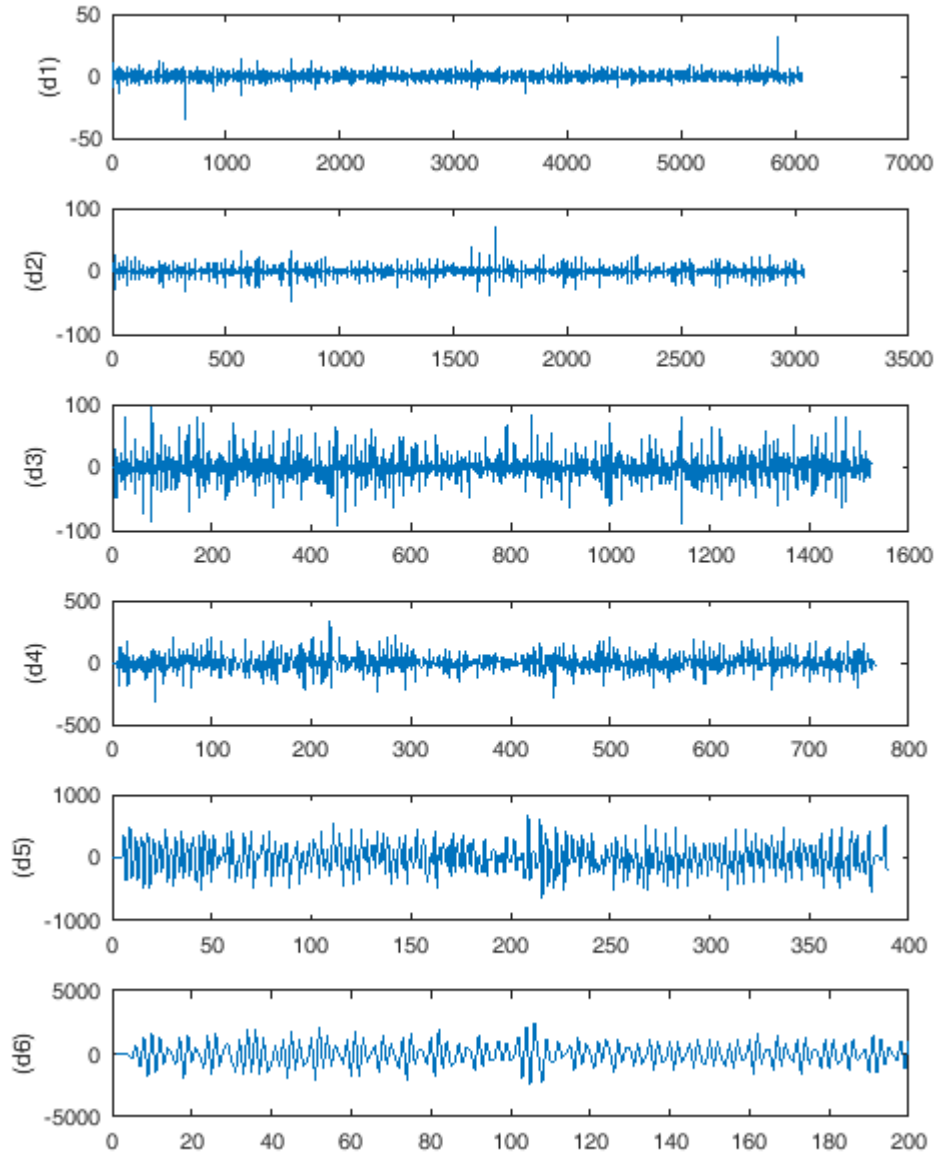
Şekil B.20: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için Daubechies3 dalgacığının 6. seviyeye kadar detay katsayıları



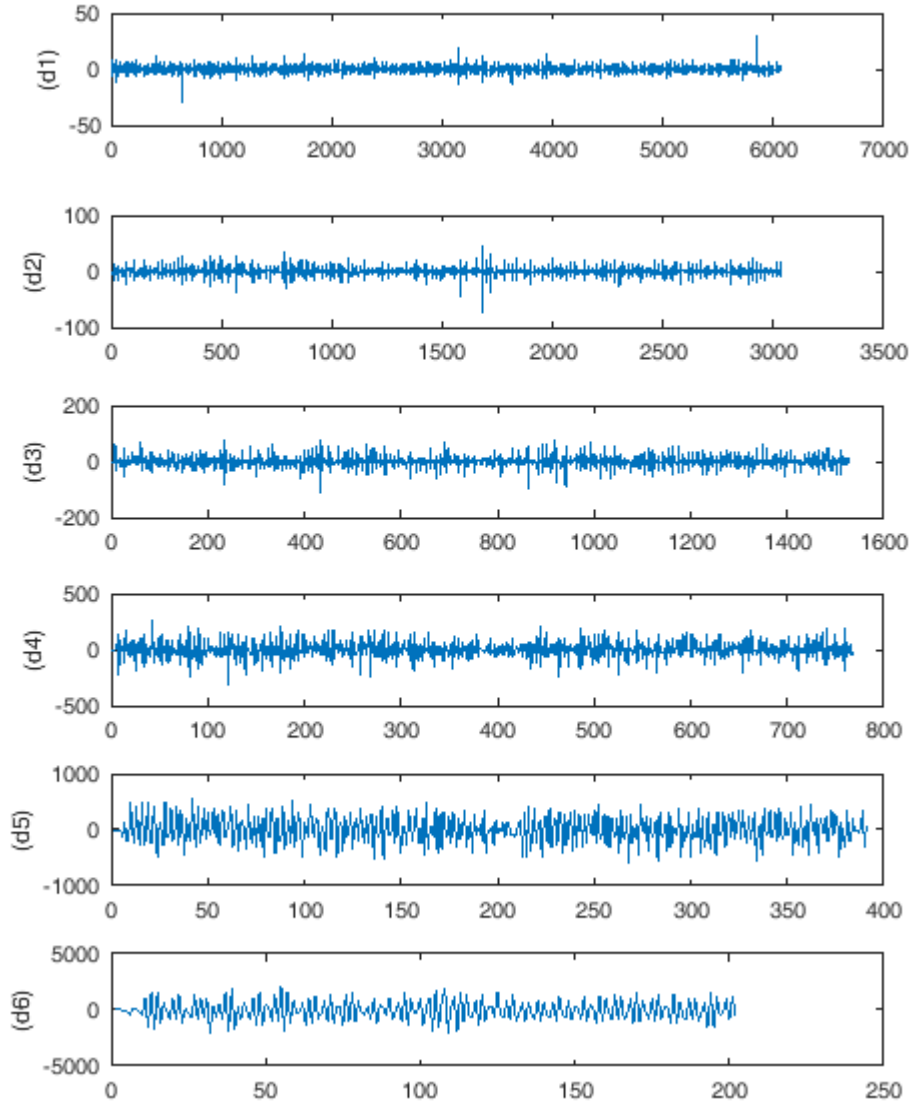
Şekil B.21: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için Daubechies4 dalgacığının 6. seviyeye kadar detay katsayıları



Şekil B.22: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için Daubechies5 dalgacığının 6. seviyeye kadar detay katsayıları



Şekil B.23: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için Daubechies6 dalgacığının 6. seviyeye kadar detay katsayıları



Şekil B.24: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için Daubechies7 dalgacığının 6. seviyeye kadar detay katsayıları

```

clc;
clear all;

data_files = ...
{'GaCo01_01', 'GaCo02_01', 'GaCo03_01', 'GaCo04_01', 'GaCo05_01',
'GaCo06_01', 'GaCo07_01', 'GaCo08_01', ...
'GaCo09_01', 'GaCo10_01', 'GaCo11_01', 'GaCo12_01', 'GaCo13_01',
'GaCo14_01', 'GaCo15_01', 'GaCo16_01', ...
'GaCo17_01', 'GaCo22_01', 'GaPt03_01', 'GaPt04_01', 'GaPt05_01',
'GaPt06_01', 'GaPt07_01', 'GaPt08_01', ...
'GaPt09_01', 'GaPt12_01', 'GaPt13_01', 'GaPt14_01', 'GaPt15_01',
'GaPt16_01', 'GaPt17_01', 'GaPt18_01', ...
'GaPt19_01', 'GaPt20_01', 'GaPt21_01', 'GaPt22_01', 'GaPt23_01',
'GaPt24_01', 'GaPt25_01', 'GaPt26_01', ...
'GaPt27_01', 'GaPt28_01', 'GaPt29_01', 'GaPt30_01', 'GaPt31_01',
'GaPt32_01', 'GaPt33_01', ...
'JuCo01_01', 'JuCo02_01', 'JuCo03_01', 'JuCo04_01', 'JuCo05_01',
'JuCo06_01', 'JuCo07_01', 'JuCo08_01', ...
'JuCo09_01', 'JuCo11_01', 'JuCo12_01', 'JuCo13_01', 'JuCo14_01',
'JuCo15_01', 'JuCo16_01', 'JuCo17_01', ...
'JuCo18_01', 'JuCo19_01', 'JuCo20_01', 'JuCo21_01', 'JuCo22_01',
'JuCo23_01', 'JuCo24_01', 'JuCo25_01', ...
'JuCo26_01', 'JuPt01_01', 'JuPt02_01', 'JuPt03_01', 'JuPt04_01',
'JuPt05_01', 'JuPt06_01', 'JuPt07_01', ...
'JuPt08_01', 'JuPt09_01', 'JuPt10_01', 'JuPt11_01', 'JuPt12_01',
'JuPt13_01', 'JuPt14_01', 'JuPt15_01', ...
'JuPt16_01', 'JuPt17_01', 'JuPt18_01', 'JuPt19_01', 'JuPt20_01',
'JuPt21_01', 'JuPt22_01', 'JuPt23_01', ...
'JuPt24_01', 'JuPt25_01', 'JuPt26_01', 'JuPt27_01', 'JuPt28_01',
'JuPt29_01', ...
'SiCo01_01', 'SiCo03_01', 'SiCo04_01', 'SiCo05_01', 'SiCo06_01',
'SiCo07_01', 'SiCo08_01', 'SiCo09_01', ...
'SiCo10_01', 'SiCo11_01', 'SiCo12_01', 'SiCo13_01', 'SiCo14_01',
'SiCo15_01', 'SiCo16_01', 'SiCo17_01', ...
'SiCo18_01', 'SiCo19_01', 'SiCo20_01', 'SiCo21_01', 'SiCo22_01',
'SiCo23_01', 'SiCo24_01', 'SiCo25_01', ...
'SiCo26_01', 'SiCo27_01', 'SiCo28_01', 'SiCo29_01', 'SiCo30_01',
'SiPt02_01', 'SiPt04_01', 'SiPt05_01', ...
'SiPt07_01', 'SiPt08_01', 'SiPt09_01', 'SiPt10_01', 'SiPt12_01',
'SiPt13_01', 'SiPt14_01', 'SiPt15_01', ...
'SiPt16_01', 'SiPt17_01', 'SiPt18_01', 'SiPt19_01', 'SiPt20_01',
'SiPt21_01', 'SiPt22_01', 'SiPt23_01', ...
'SiPt24_01', 'SiPt25_01', 'SiPt27_01', 'SiPt28_01', 'SiPt29_01',
'SiPt30_01', 'SiPt31_01', 'SiPt32_01', ...
'SiPt33_01', 'SiPt34_01', 'SiPt35_01', 'SiPt36_01', 'SiPt37_01',
'SiPt38_01', 'SiPt39_01', 'SiPt40_01'};

Nstd = 0.001;
NR = 1;
MaxIter = 15;
fs = 100;
mode_number = 6;
inc = 80;
t_inc = 80*6 + 80*6 + 80*5 + 96;

```

Şekil B.25: HHD kullanılarak her bir denek için öznitelik oluşturma

```

F1 = zeros(1, 96);
F2 = zeros(1, 80);
F3 = zeros(1, 80);
F4 = zeros(1, 80);
F_total = zeros(1, t_inc*477);
features = zeros(length(data_files), t_inc*477);

for i=1 : length(data_files)
    load (strcat('C:\...\', data_files{i}, '.mat'));

    abs_diff = zeros(size(data,1), 153);
    diff = zeros(size(data,1), 153);
    add = zeros(size(data,1), 153);

    m=1;
    for k=2 : 18
        for t=k+1 : 19
            df = data(:, k) - data(:, t);
            abs_diff(:, m) = abs(df);
            diff(:, m) = df;
            add(:, m) = data(:, k) + data(:, t);
            m = m + 1;
        end
    end

    signals = [data(:, 2:end) abs_diff diff add];

    [row, col] = size(signals);

    for j=1 : a
        pxx = pwelch(package(j, :));
        [pks, locs, w, p] = findpeaks(package(j, :));
        F1 = [Stat(package(j, :)) Stat(pxx) Stat(pks) Stat(locs)
Stat(w) Stat(p)];

        [modes] = ceemdan(package(j, :), Nstd, NR, MaxIter, 1);
        for t=1 : mode_number
            [pks2, locs2, w2, p2] = findpeaks(modes(t, 1:end));
            F2(1, ((t-1)*inc + 1):((t-1)*inc + inc)) =
[Stat(modes(t, 1:end)) Stat(pks2) Stat(locs2) Stat(w2) Stat(p2)];
        end;

        y = hilbert(modes);
        amplitude = abs(y);
        instfreq = (fs/(2*pi))*diff(unwrap(angle(y)));

        for t=1 : mode_number
            [pks3, locs3, w3, p3] = findpeaks(amplitude(t,
1:end));
            F3(1, ((t-1)*inc + 1):((t-1)*inc + inc)) =
[Stat(amplitude(t, 1:end)) Stat(pks3) Stat(locs3) Stat(w3)
Stat(p3)];
        end;
    end;
end;

```

Şekil B.25: HHD kullanılarak her bir denek için öznelik oluşturma

```

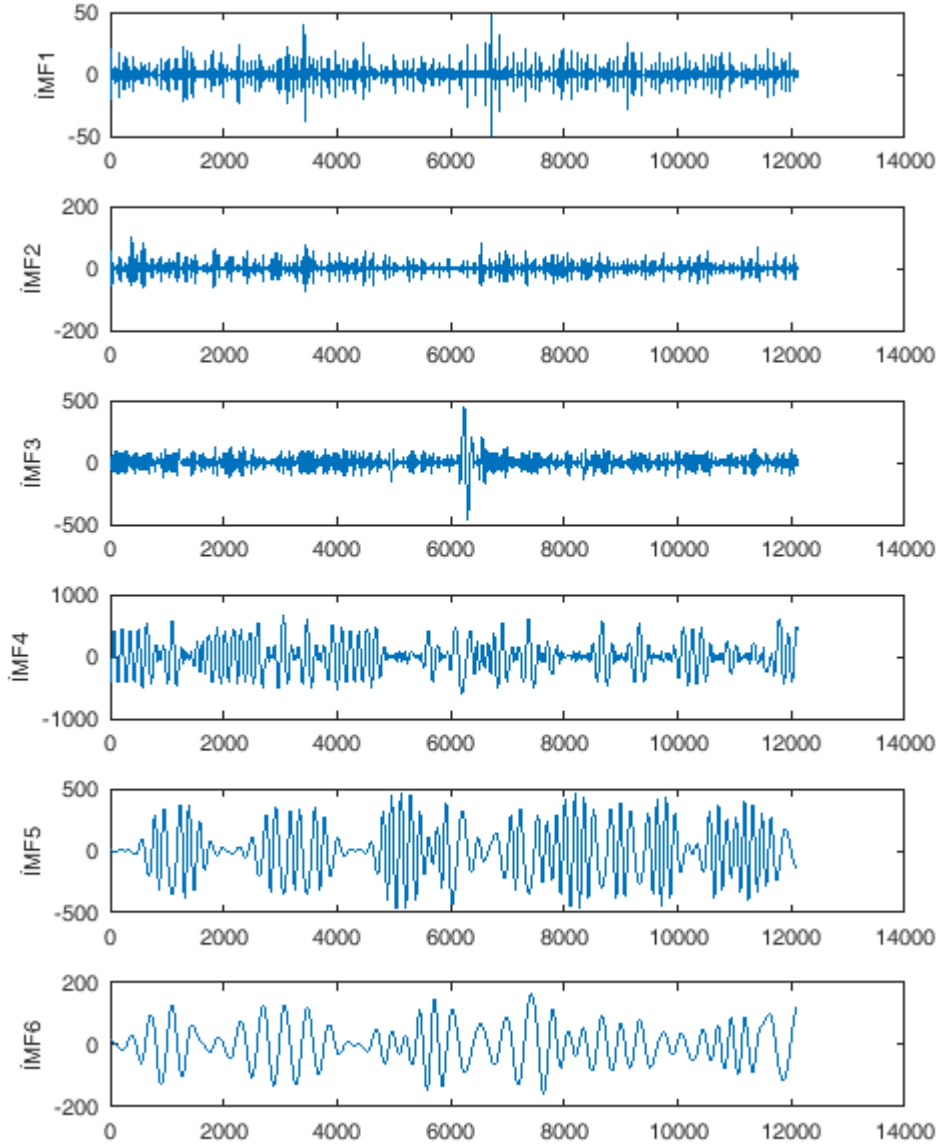
        for t=1 : mode_number - 1
            [pks4, locs4, w4, p4] = findpeaks(instfreq(t, 1:end));
            F4(1, ((t-1)*inc + 1):((t-1)*inc + inc)) =
[Stat(instfreq(t, 1:end)) Stat(pks4) Stat(locs4) Stat(w4)
Stat(p4)];
            end;

            F_total(1, ((j-1)*t_inc + 1):((j-1)*t_inc + t_inc)) = [F1
F2 F3 F4];
            end;
            features(i, :) = F_total;
        end;

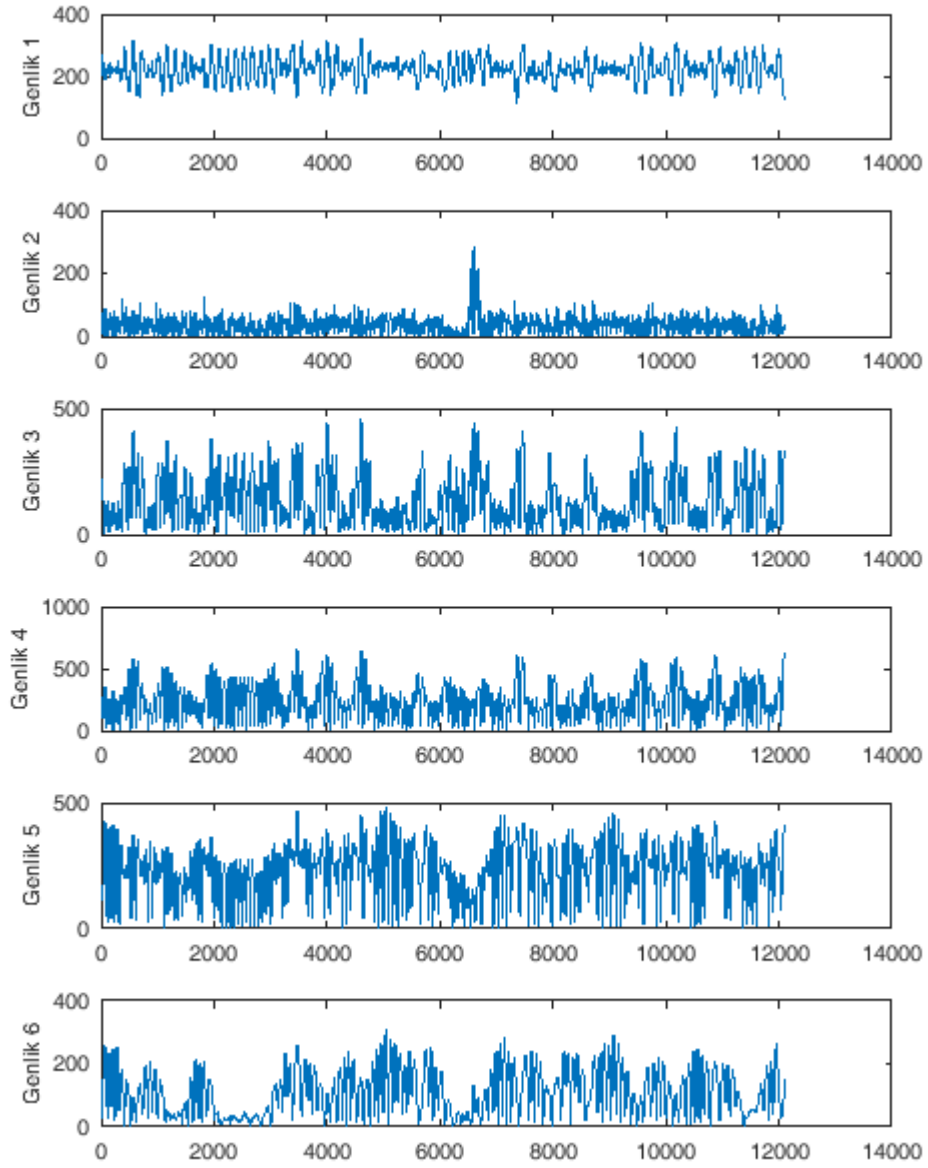
```

Şekil B.25: HHD kullanılarak her bir denek için öznelik oluşturma

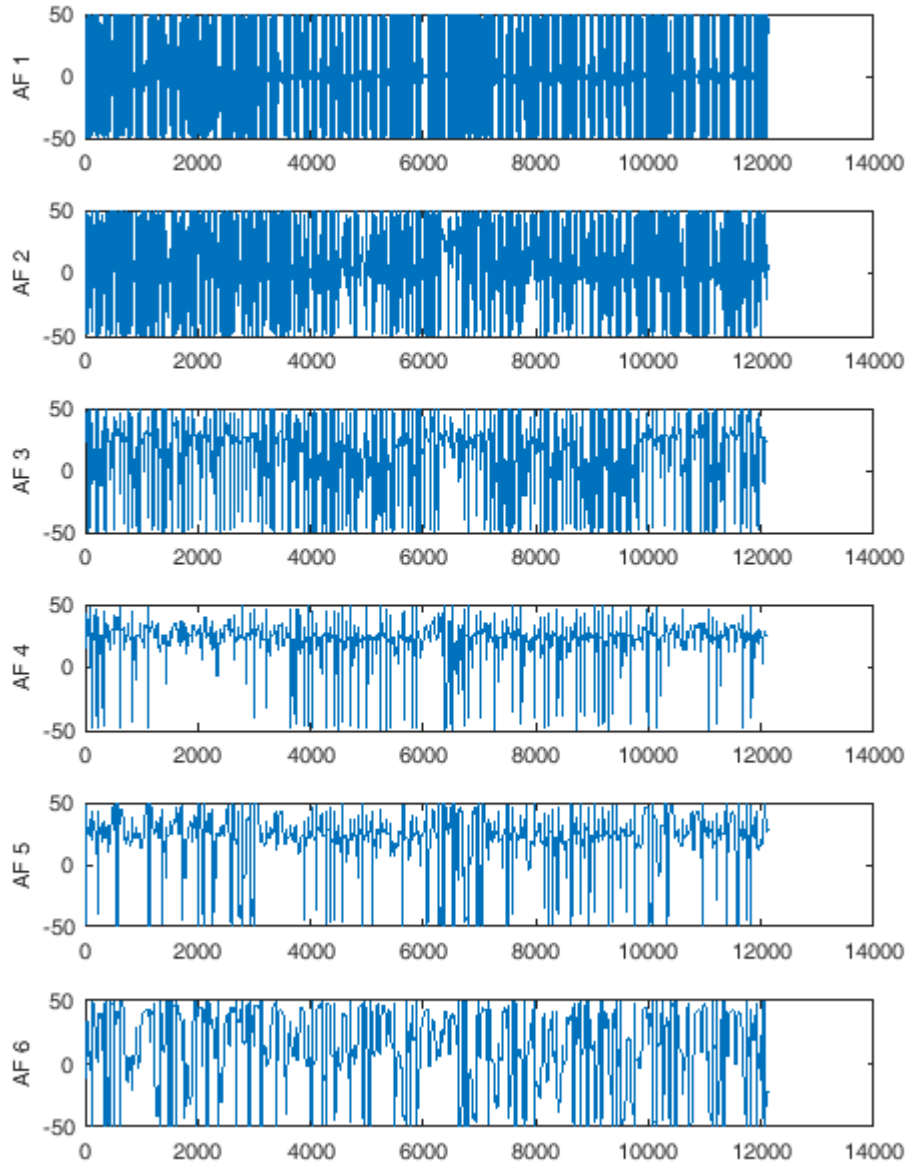




Şekil B.26: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için ilk 6 İMF



Şekil B.27: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için ilk 6 İMF'ye ait genlik değerleri



Şekil B.28: Parkinsonlu bireyin sağ ayağına ait kuvvet sinyali için ilk 6 İMF'ye ait AF değerleri

```

clc;
clear all;

outcome = ...
{ 'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO',
'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO', ...
'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO',
'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO', ...
'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO',
'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO', ...
'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO', 'CO',
'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD',
'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', ...
'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD',
...
'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD',
'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', ...
'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD',
...
'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD',
'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', ...
'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD',
'PD', 'PD', 'PD', 'PD', 'PD', 'PD', 'PD'};

path = 'C:\...\';
fileName = 'p1';
load (strcat(path, fileName, '.mat'));
features = p1;
[row, column]=size(features);

fid = fopen( strcat(path, fileName, '.arff'), 'wt');
fprintf(fid, '@relation parkinson\n\n');
for j=1 : column
    disp(j);
    fprintf(fid, strcat('@attribute', ' f_', num2str(j), '
numeric\n'));
end
fprintf(fid, '@attribute outcome {CO, PD}\n');
fprintf(fid, '\n@data\n');
for k=1 : row
    disp(k);
    for t=1 : column
        fprintf(fid, strcat(num2str(features(k, t)), ', ');
    end
    fprintf(fid, strcat(outcome{k}, '\n'));
end
fclose(fid);
fclose('all');

```

Şekil B.29: WEKA için arff dosya formatı oluşturma


```

function Benchmark_NN(input, output, hiddenLayerSize, step)

    for i=1:size(input,2)
        input(:,i) = mat2gray(input(:,i));
    end

    function [accuracyTrain, accuracyVal, accuracyTest, accuracyAll]
= BackPropagationNeuralNetwork()
        x = input';
        t = output';

        % Choose a Training Function
        % For a list of all training functions type: help nntain
        % 'trainlm' is usually fastest. (Levenberg-Marquardt)
        % 'trainbr' takes longer but may be better for challenging
problems.
        % 'trainscg' uses less memory. Suitable in low memory
situations.
        trainFcn = 'trainscg';

        % Create a Pattern Recognition Network
        net = patternnet(hiddenLayerSize, trainFcn);
        net.trainParam.showWindow = 0;

        % Setup Division of Data for Training, Validation, Testing
        net.divideParam.trainRatio = 70/100;
        net.divideParam.valRatio = 15/100;
        net.divideParam.testRatio = 15/100;

        % Train the Network
        [net, tr] = train(net,x,t);

        % Training Confusion Plot Variables
        yTrn = net(x(:,tr.trainInd));
        tTrn = t(:,tr.trainInd);
        [c,~] = confusion(tTrn,yTrn);
        accuracyTrain = 100*(1-c);

        % Validation Confusion Plot Variables
        yVal = net(x(:,tr.valInd));
        tVal = t(:,tr.valInd);
        [c,~] = confusion(tVal,yVal);
        accuracyVal = 100*(1-c);

        % Test Confusion Plot Variables
        yTst = net(x(:,tr.testInd));
        tTst = t(:,tr.testInd);
        [c,~] = confusion(tTst,yTst);
        accuracyTest = 100*(1-c);
    end
end

```

Şekil B.30: Parkinson veri kümelerine uygulanan Yapay Sinir Ağı için benchmark programı

```

    % Overall Confusion Plot Variables
    yAll = net(x);
    tAll = t;
    [c,~] = confusion(tAll,yAll);
    accuracyAll = 100*(1-c);

end

acc1 = zeros(step,1);
acc2 = zeros(step,1);
acc3 = zeros(step,1);
acc4 = zeros(step,1);

for i=1 : step
    %disp(i);
    [accuracyTrain, accuracyVal, accuracyTest, accuracyAll] =
BackPropagationNeuralNetwork();
    acc1(i) = accuracyTrain;
    acc2(i) = accuracyVal;
    acc3(i) = accuracyTest;
    acc4(i) = accuracyAll;
end

fprintf('%2.2f±%2.2f\t%2.2f±%2.2f\t%2.2f±%2.2f\t%2.2f±%2.2f\n',
...
        mean(acc1), std(acc1), mean(acc2), std(acc2), mean(acc3),
std(acc3), ...
        mean(acc4), std(acc4));
end

```

Şekil B.30: Parkinson veri kümelerine uygulanan Yapay Sinir Ağı için benchmark programı

Çizelge B.8: Sinyallere uygulanan istatistiksel büyüklükler

İstatistiksel Büyüklükler	Formül / Fonksiyon
Maksimum	$\max(x)$
Minimum	$\min(x)$
Ortalama	$\frac{1}{N} \sum_{i=1}^N x_i$
Standart Sapma	$\sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - \mu)^2}$
Çarpıklık	$\frac{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^3}{\left(\sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2}\right)^3}$
Basıklık	$\frac{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^4}{\left(\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2\right)^2}$
Ortanca	$\text{median}(x)$
Aralık	$\max(x) - \min(x)$
Ortalamadan Mutlak Sapma	$\text{mad}(x, 0);$
Ortancadan Mutlak Sapma	$\text{mad}(x, 1)$
Toplam	$\text{sum}(x)$
Shannon entropisi	$\text{wentropy}(x, 'shannon')$
Logaritmik entropi	$\text{wentropy}(x, 'log energy')$
Yöneynin 1. normu	$\text{norm}(x, 1)$
Yöneynin 2. normu	$\text{norm}(x)$

Çizelge B.9: FD kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

Öznelik No	Ayak Sinyali	Sinyale Uygulanan İstatistiksel Fonksiyon	Özneliğin Tek Başına Sınıflandırma Doğruluğu (%)
1	$ L_1 - L_3 $	norm(p)	72.125±2.085
2	$ L_5 - R_6 $	wentropy(w, 'shannon')	73.402±2.178
3	$ R_1 - R_7 $	std(signals)/mean(signals)	78.789±1.903
4	$ R_3 - R_6 $	norm(w)	75.219±0.790
5	$ R_1 - R_7 $	kurtosis(X_mags(1:N_2))	73.741±3.457
6	$ R_3 - R_6 $	wentropy(w, 'shannon')	72.186±1.850
7	$L_1 - L_6$	mean(pks)	74.416±2.425
8	$L_1 - L_6$	std(signals)	73.397±1.564
9	$L_1 - L_6$	median(pks)	73.200±1.950
10	$L_1 - R_7$	sum(pks)	73.067±1.904
11	$L_1 - L_6$	skewness(p)	72.396±2.390
12	$L_1 - R_6$	range(p)	71.448±1.281
13	$L_6 - R_5$	kurtosis(signals)	73.806±1.438
14	$R_1 - R_7$	std(signals)/mean(signals)	77.983±1.718
15	$R_1 - R_7$	kurtosis(X_mags(1:N_2))	73.741±3.457
16	$L_4 + L_7$	kurtosis(signals)	73.670±0.834
17	$L_4 + L_8$	skewness(signals)	73.941±1.728
18	$L_4 + L_5$	kurtosis(signals)	74.141±1.843
19	$L_5 + L_8$	std(signals)/mean(signals)	73.130±2.749
20	$L_5 + L_8$	median(signals)	72.725±1.815
21	$L_5 + L_8$	mad(signals, 1)	72.455±1.855
22	$L_4 + L_5$	skewness(signals)	71.920±1.876
23	$R_5 + R_8$	std(signals)/mean(signals)	75.356±2.582
24	$R_5 + R_6$	kurtosis(signals)	72.728±2.142

Çizelge B.10: Gaus2 dalgacığı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

Öznitelik No	Ayak Sinyali	Sinyale Uygulanan İstatistiksel Fonksiyon	Özniteliğin Tek Başına Sınıflandırma Doğruluğu (%)
1	$ L_1 - L_3 $	norm (p)	72.125±2.085
2	$ L_5 - R_6 $	wentropy (w, 'shannon')	73.402±2.178
3	$ R_1 - R_7 $	std(signals)/mean(signals)	78.789±1.903
4	$ R_3 - R_6 $	norm (w)	75.219±0.790
5	$ R_3 - R_6 $	wentropy (w, 'shannon')	72.186±1.850
6	$L_1 - L_6$	mean (pks)	74.416±2.425
7	$L_1 - L_6$	std(signals)	73.397±1.564
8	$L_1 - L_6$	median (pks)	73.200±1.950
9	$L_1 - R_7$	sum (pks)	73.067±1.904
10	$L_1 - L_6$	std (p)	72.396±2.390
11	$L_1 - R_6$	range (p)	71.448±1.281
12	$L_6 - R_5$	kurtosis (signals)	73.806±1.438
13	$R_1 - R_7$	std(signals)/mean(signals)	77.983±1.718
14	$L_4 + L_8$	skewness (signals)	73.941±1.728
15	$L_3 + L_7$	kurtosis (signals)	73.670±0.834
16	$L_3 + L_5$	kurtosis (signals)	74.141±1.843
17	$L_5 + L_8$	median (signals)	72.725±1.815
18	$L_5 + L_8$	std(signals)/mean(signals)	73.130±2.749
19	$L_5 + L_8$	mad(signals, 1)	72.455±1.855
20	$L_4 + L_5$	skewness (signals)	71.920±1.876
21	$R_5 + R_8$	std(signals)/mean(signals)	75.356±2.582
22	$R_5 + R_6$	kurtosis (signals)	72.728±2.142
23	$R_5 + R_8$	skewness (D)	72.254±1.879

Çizelge B.11: BiorSplines3.3 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

Öznelik No	Ayak Sinyali	Sinyale Uygulanan İstatistiksel Fonksiyon	Özneliğin Tek Başına Sınıflandırma Doğruluğu (%)
1	L_1	mad(pks2_4, 0)	71.449±1.392
2	$ L_1 - L_6 $	mean(pks2_5)	77.576±1.907
3	$ L_1 - L_2 $	mad(p2_4, 0)	76.973±2.068
4	$ L_1 - R_6 $	std(p2_2)	76.772±2.350
5	$ L_1 - L_3 $	range(p2_4)	76.902±2.465
6	$ L_1 - L_3 $	max(p2_4)	76.633±2.325
7	$ L_1 - L_2 $	std(p2_4)	76.094±1.658
8	$ L_1 - L_3 $	std(pks2_4)	75.761±1.931
9	$ L_1 - L_6 $	mean(pks2_4)	75.423±1.594
10	$ L_1 - L_2 $	range(D_level_4)	76.901±2.726
11	$ L_1 - L_2 $	norm(D_level_4)	74.547±1.523
12	$ L_1 - L_3 $	std(p2_4)	75.624±2.610
13	$ L_1 - L_3 $	mad(pks2_4, 0)	74.549±1.992
14	$ L_1 - L_3 $	mean(pks2_4)	75.017±2.582
15	$ L_1 - L_3 $	mad(p2_4, 0)	74.677±1.588
16	$ L_1 - L_4 $	mad(p2_4, 0)	75.222±2.776
17	$ L_1 - L_6 $	std(D_level_4)	74.075±1.396
18	$ L_1 - L_4 $	wentropy(pks2_4, 'shannon')	74.074±1.809
19	$ L_1 - L_3 $	std(D_level_4)	74.819±2.238
20	$ L_1 - L_2 $	mean(pks2_4)	74.751±2.730
21	$ L_1 - L_4 $	mad(pks2_4, 0)	74.879±2.349
22	$ L_1 - L_2 $	wentropy(p2_4, 'shannon')	73.672±1.538
23	$ L_1 - R_2 $	range(p2_4)	73.942±1.987
24	$ L_1 - L_2 $	range(p2_4)	73.537±1.762
25	$ L_1 - L_2 $	mean(p2_4)	73.470±1.642
26	$ L_1 - L_2 $	wentropy(pks2_4, 'shannon')	73.468±1.414
27	$ L_1 - L_2 $	range(p2_3)	73.601±1.635
28	$ L_1 - L_2 $	max(p2_3)	73.601±1.635

Çizelge B.11: (devam) BiorSplines3.3 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

29	$ L_1 - L_3 $	range(pks2_4)	73.742±2.163
30	$ L_1 - L_3 $	norm(pks2_4)	73.877±2.578
31	$ L_1 - L_4 $	std(p2_4)	73.536±1.948
32	$ L_1 - R_2 $	max(p2_4)	73.807±2.398
33	$ L_1 - R_8 $	std(p2_1)	73.131±1.142
34	$ L_1 - L_3 $	norm(D_level_4)	73.938±2.462
35	$ L_1 - L_2 $	max(p2_4)	73.334±1.868
36	$ L_1 - R_6 $	range(D_level_3)	73.403±1.594
37	$ L_1 - R_4 $	wentropy(p2_3, 'shannon')	73.134±2.013
38	$ L_1 - L_3 $	wentropy(pks2_4, 'shannon')	73.742±2.800
39	$ L_1 - R_6 $	std(D_level_3)	73.467±2.384
40	$ L_1 - R_1 $	std(pks2_4)	73.334±1.947
41	$ L_1 - L_6 $	std(p2_4)	73.469±2.620
42	$ L_1 - L_6 $	std(D_level_5)	74.207±3.025
43	$ L_1 - L_6 $	mad(pks2_4, 0)	73.405±2.492
44	$ L_1 - L_2 $	wentropy(D_level_4, 'shannon')	72.998±1.920
45	$ L_1 - L_3 $	norm(p2_4)	73.271±2.351
46	$ L_1 - L_6 $	norm(pks2_5)	73.064±1.706
47	$ L_1 - R_4 $	std(D_level_3)	73.205±2.727
48	$ L_1 - L_2 $	mean(p2_3)	72.797±2.082
49	$ L_1 - L_6 $	std(pks2_3)	74.006±3.035
50	$ L_1 - L_3 $	range(D_level_4)	72.929±1.955
51	$ L_1 - R_6 $	range(pks2_3)	72.663±2.182
52	$ L_1 - R_6 $	std(D_level_2)	72.325±1.244
53	$ L_1 - L_3 $	max(D_level_4)	72.798±2.709
54	$ L_1 - L_3 $	max(pks2_4)	72.798±2.709
55	$ L_1 - L_3 $	wentropy(p2_4, 'shannon')	72.460±1.715
56	$ L_1 - L_3 $	wentropy(D_level_4, 'shannon')	73.401±2.864
57	$ L_1 - L_2 $	norm(pks2_4)	72.459±1.749
58	$ L_1 - R_6 $	norm(D_level_3)	73.333±3.170
59	$ L_1 - L_3 $	mad(D_level_4, 0)	72.595±1.989
60	$ L_1 - L_4 $	norm(pks2_4)	72.457±1.978
61	$ L_1 - L_4 $	range(pks2_4)	72.661±2.360

Çizelge B.11: (devam) BiorSplines3.3 dalgacığı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

62	$ L_1 - L_3 $	norm(p)	72.125±2.085
63	$ L_1 - L_2 $	std(pks2_4)	73.467±3.079
64	$ L_1 - L_2 $	std(D_level_4)	72.253±1.663
65	$ L_1 - L_2 $	norm(D_level_4, 1)	72.321±1.878
66	$ L_5 - R_6 $	wentropy(w, 'shannon')	73.402±2.178
67	$ L_6 - R_1 $	max(pks2_3)	72.997±1.710
68	$ L_6 - R_1 $	max(D_level_3)	72.997±1.710
69	$ L_6 - R_1 $	std(D_level_3)	71.717±1.317
70	$ R_1 - R_7 $	std(signals)/mean(signals)	78.789±1.903
71	$ R_1 - R_6 $	mad(D_level_5, 1)	76.230±1.863
72	$ R_1 - R_6 $	std(p2_4)	74.683±2.955
73	$ R_1 - R_3 $	mad(p2_4, 0)	73.063±1.959
74	$ R_1 - R_3 $	wentropy(p2_4, 'shannon')	73.335±2.699
75	$ R_1 - R_3 $	norm(p2_4)	72.928±2.363
76	$ R_1 - R_3 $	wentropy(D_level_4, 'shannon')	73.334±2.344
77	$ R_3 - R_6 $	norm(w)	75.219±0.790
78	$ R_3 - R_6 $	wentropy(w, 'shannon')	72.186±1.850
79	$L_1 - L_4$	mean(pks2_4)	76.635±2.711
80	$L_1 - L_3$	wentropy(pks2_4, 'shannon')	75.356±1.528
81	$L_1 - L_6$	mad(pks2_4, 0)	76.028±2.204
82	$L_1 - L_4$	mad(pks2_4, 0)	74.881±1.259
83	$L_1 - L_4$	wentropy(D_level_4, 'shannon')	75.019±1.577
84	$L_1 - L_3$	mad(pks2_4, 0)	74.546±2.054
85	$L_1 - L_3$	std(pks2_4)	75.153±2.622
86	$L_1 - L_6$	mean(pks)	74.416±2.425
87	$L_1 - L_3$	norm(pks2_4)	74.277±2.236
88	$L_1 - L_2$	std(D_level_4)	74.138±2.485
89	$L_1 - L_6$	std(signals)	73.397±1.564
90	$L_1 - L_3$	max(D_level_4)	73.270±2.199
91	$L_1 - L_2$	mean(p2_3)	73.334±2.198
92	$L_1 - L_6$	range(p2_6)	72.996±1.729
93	$L_1 - L_3$	max(pks2_4)	73.270±2.199
94	$L_1 - L_6$	median(pks)	73.200±1.950

Çizelge B.11: (devam) BiorSplines3.3 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

95	$L_1 - L_2$	range (p2_4)	72.528±1.912
96	$L_1 - L_4$	std(pks2_4)	72.997±2.752
97	$L_1 - L_6$	std(p)	72.396±2.390
98	$L_1 - L_2$	std(pks2_4)	72.320±1.743
99	$L_1 - L_6$	std(D_level_4)	72.254±1.900
100	$L_1 - L_8$	range (p2_4)	72.258±2.076
101	$L_1 - L_2$	norm(p2_4)	71.449±1.293
102	$L_1 - R_6$	std(pks2_4)	74.479±2.105
103	$L_1 - R_6$	mean (p2_4)	74.548±1.289
104	$L_1 - R_6$	mean (pks2_4)	73.063±1.324
105	$L_1 - R_7$	sum (pks)	73.067±1.904
106	$L_1 - R_6$	std(D_level_4)	73.470±2.460
107	$L_1 - R_6$	norm(D_level_4)	73.332±2.083
108	$L_1 - R_6$	min(D_level_4)	72.596±1.678
109	$L_1 - R_1$	std(pks2_4)	72.931±2.307
110	$L_1 - R_6$	mad(pks2_4, 0)	72.729±2.465
111	$L_1 - R_2$	mad(pks2_4, 0)	72.932±2.694
112	$L_1 - R_4$	std(D_level_4)	72.398±3.020
113	$L_1 - R_6$	range (p)	71.448±1.281
114	$L_6 - R_1$	mad(pks2_5, 0)	73.536±2.405
115	$L_6 - R_5$	kurtosis (signals)	73.806±1.438
116	$R_1 - R_2$	std(p2_4)	73.538±1.518
117	$R_1 - R_3$	std(pks2_4)	73.532±1.642
118	$R_1 - R_7$	std(signals) / mean (signals)	77.983±1.718
119	$L_1 + R_8$	std(p2_4)	74.007±1.644
120	$L_1 + R_8$	std(p2_3)	73.809±2.368
121	$L_1 + L_5$	std(p2_4)	73.068±1.818
122	$L_1 + R_1$	wentropy (p2_4, 'shannon')	72.729±1.605
123	$L_1 + R_1$	mad(pks2_4, 0)	72.729±1.255
124	$L_1 + R_2$	norm(pks2_4)	72.458±1.689
125	$L_1 + L_8$	std(pks2_4)	72.799±2.527
126	$L_1 + R_2$	std(pks2_5)	73.466±2.331
127	$L_1 + R_2$	range (p2_4)	73.469±2.956

Çizelge B.11: (devam) BiorSplines3.3 dalgacıđı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıřtır)

128	$L_1 + R_1$	std(p2_4)	72.190±1.536
129	$L_1 + R_1$	mean(pks2_4)	73.542±3.121
130	$L_1 + L_6$	std(pks2_4)	71.651±1.079
131	$L_1 + R_8$	wentropy(pks2_6, 'shannon')	71.987±1.538
132	$L_4 + L_5$	kurtosis(signals)	74.141±1.843
133	$L_4 + L_7$	kurtosis(signals)	73.670±0.834
134	$L_4 + L_5$	skewness(signals)	71.920±1.876
135	$L_4 + L_8$	skewness(signals)	73.941±1.728
136	$L_5 + L_8$	std(signals)/mean(signals)	73.130±2.749
137	$L_5 + L_8$	mad(signals, 1)	72.455±1.855
138	$L_5 + L_8$	median(signals)	72.725±1.815
139	$L_5 + R_1$	std(D_level_4)	72.591±2.315
140	$R_1 + R_5$	mad(p2_4, 0)	74.342±3.233
141	$R_1 + R_5$	mad(pks2_4, 0)	72.120±1.64
142	$R_5 + R_8$	std(signals)/mean(signals)	75.356±2.582
143	$R_5 + R_6$	kurtosis(signals)	72.728±2.142
144	R2 - R6	Ed(5)	73.534±1.397
145	R2 - R7	wentropy(Ed, 'shannon')	72.862±1.593
146	L7 + R4	wentropy(Ed, 'shannon')	72.793±2.413

Çizelge B.12: Coiflets2 dalgacıği kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

Öznitelik No	Ayak Sinyali	Sinyale Uygulanan İstatistiksel Fonksiyon	Özniteliğın Tek Başına Sınıflandırma Doğruluđu (%)
1	L1 - L3	std(D_level_4)	77.378±1.956
2	L1 - L3	std(p2_4)	77.310±1.511
3	L1 - L2	wentropy(p2_3, 'shannon')	77.641±1.235
4	L1 - L3	mad(pks2_3, 0)	77.173±2.087
5	L1 - R6	std(p2_2)	77.642±2.434
6	L1 - L2	mean(pks2_3)	76.164±1.508
7	L1 - L2	mad(pks2_3, 0)	76.294±1.307
8	L1 - L3	std(p2_3)	76.232±1.940
9	L1 - L2	wentropy(D_level_3, 'shannon')	76.767±2.521
10	L1 - L2	norm(p2_3)	75.692±1.284
11	L1 - L3	min(D_level_4)	76.030±1.608
12	L1 - L2	std(p2_3)	75.822±1.973
13	L1 - L2	mad(p2_3, 0)	76.161±2.166
14	L1 - R6	range(p2_2)	76.496±2.033
15	L1 - R6	max(p2_2)	76.496±2.033
16	L1 - L3	max(p2_4)	75.154±1.805
17	L1 - L3	mad(p2_4, 0)	75.561±2.590
18	L1 - L2	mean(p2_3)	75.419±1.424
19	L1 - L3	wentropy(p2_3, 'shannon')	75.019±1.510
20	L1 - L3	range(p2_4)	74.952±1.816
21	L1 - L2	std(pks2_3)	74.276±1.161
22	L1 - L3	norm(p2_3)	74.821±2.771
23	L1 - L3	wentropy(D_level_3, 'shannon')	74.750±2.272
24	L1 - L3	mad(p2_3, 0)	75.021±2.611
25	L1 - L2	range(p2_4)	73.872±1.071
26	L1 - L3	sum(p2_3)	73.670±1.602
27	L1 - L2	std(D_level_3)	74.076±2.254
28	L1 - L3	norm(p2_3, 1)	73.67 ±1.602

Çizelge B.12: (devam) Coiflets2 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

29	L1 - L4	norm(pks2_3)	73.943±2.210
30	L1 - L2	max(p2_4)	73.536±1.118
31	L1 - R4	std(p2_3)	74.147±2.781
32	L1 - R6	std(pks2_2)	73.604±1.810
33	L1 - L4	mad(p2_3, 0)	73.533±1.809
34	L1 - R6	std(D_level_2)	73.470±1.738
35	L1 - L3	std(D_level_3)	73.606±1.874
36	L1 - L2	norm(pks2_3)	74.277±3.004
37	L1 - L3	range(D_level_4)	73.674±2.624
38	L1 - L6	std(p2_4)	73.874±2.467
39	L1 - L3	sum(pks2_3)	73.133±1.767
40	L1 - L3	mad(D_level_3, 0)	73.540±2.520
41	L1 - L3	norm(p2_4)	73.066±2.140
42	L1 - L4	wentropy(pks2_3, 'shannon')	73.403±2.587
43	L1 - L6	min(D_level_5)	72.728±0.948
44	L1 - L2	sum(p_3)	72.593±1.041
45	L1 - L3	norm(pks2_3, 1)	72.796±1.961
46	L1 - L2	max(p2_3)	72.792±1.563
47	L1 - L2	range(p2_3)	72.792±1.563
48	L1 - L2	norm(p2_3, 1)	72.593±1.041
49	L1 - L2	norm(p2_4)	72.862±1.733
50	L1 - R4	norm(p2_3)	72.932±2.316
51	L1 - L3	wentropy(p2_4, 'shannon')	72.729±2.083
52	L1 - L3	mean(p2_3)	73.342±3.236
53	L1 - L6	range(p2_4)	72.325±1.866
54	L1 - L2	norm(D_level_3)	72.929±2.849
55	L1 - L2	wentropy(pks2_3, 'shannon')	72.93 ±2.708
56	L1 - L5	range(pks2_3)	72.797±2.397
57	L1 - L3	std(pks2_4)	73.070±2.861
58	L1 - L2	sum(pks2_3)	72.729±2.371
59	L1 - L2	range(D_level_3)	72.120±1.802
60	L1 - R4	max(p2_3)	72.326±1.852
61	L1 - L2	mad(D_level_3, 0)	72.861±1.826

Çizelge B.12: (devam) Coiflets2 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

62	L1 - L3	norm(p)	72.125±2.085
63	L1 - R4	range(p2_3)	71.990±1.883
64	L5 - R1	median(D_level_2)	73.338±2.053
65	L5 - R6	wentropy(w, 'shannon')	73.402±2.178
66	L6 - R1	std(D_level_2)	75.825±2.401
67	L6 - R1	std(p2_2)	75.763±3.585
68	L6 - R1	max(D_level_2)	73.939±2.697
69	L6 - R1	max(pks2_2)	73.939±2.697
70	L6 - R4	skewness(D_level_5)	73.264±2.052
71	L6 - R1	wentropy(pks2_2, 'shannon')	73.063±2.761
72	R1 - RT	std(signals)/mean(signals)	78.789±1.903
73	R1 - R6	median(p2_3)	75.555±1.661
74	R1 - R6	std(p2_4)	74.280±2.653
75	R1 - R6	mad(p2_4, 1)	73.066±1.946
76	R1 - R2	range(p2_4)	72.189±1.688
77	R1 - R2	max(p2_4)	72.189±1.688
78	R1 - R6	std(pks2_4)	72.729±2.128
79	R1 - R3	mad(pks2_4, 0)	71.648±1.603
80	R3 - R6	norm(w)	75.219±0.790
81	R3 - R6	wentropy(w, 'shannon')	72.186±1.850
82	L1 - L6	std(D_level_6)	75.758±1.022
83	L1 - L2	mad(pks2_3, 0)	75.487±1.899
84	L1 - L2	std(p2_3)	74.140±1.158
85	L1 - L4	mad(pks2_3, 0)	73.872±1.928
86	L1 - L6	mean(pks)	74.416±2.425
87	L1 - L4	wentropy(p2_3, 'shannon')	73.470±1.355
88	L1 - L6	std(signals)	73.397±1.564
89	L1 - L4	range(p2_3)	73.469±2.224
90	L1 - L5	max(pks2_3)	73.062±1.282
91	L1 - L5	max(D_level_3)	73.062±1.282
92	L1 - L4	wentropy(pks2_3, 'shannon')	72.864±1.749
93	L1 - L4	max(p2_3)	73.268±2.389
94	L1 - L2	mad(p2_3, 0)	73.063±1.713

Çizelge B.12: (devam) Coiflets2 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

95	L1 - L6	median(pks)	73.200±1.950
96	L1 - L6	range(D_level_4)	72.664±1.744
97	L1 - L6	wentropy(pks2_4, 'shannon')	72.393±1.920
98	L1 - L6	std(p)	72.396±2.390
99	L1 - L6	std(p2_4)	72.792±2.178
100	L1 - L5	range(pks2_3)	72.390±1.631
101	L1 - L6	range(p2_6)	71.785±0.684
102	L1 - L6	std(D_level_4)	72.124±2.051
103	L1 - L6	max(p2_6)	72.052±2.007
104	L1 - L4	norm(D_level_3)	71.923±1.823
105	L1 - R7	sum(pks)	73.067±1.904
106	L1 - R6	min(D_level_4)	72.728±1.835
107	L1 - R7	max(D_level_4)	72.659±2.312
108	L1 - R7	max(pks2_4)	72.659±2.312
109	L1 - R6	mean(p2_4)	72.528±2.218
110	L1 - R5	wentropy(D_level_3, 'shannon')	71.786±1.721
111	L1 - R6	range(p)	71.448±1.281
112	L5 - R7	skewness(D_level_5)	76.096±2.618
113	L5 - R6	skewness(D_level_5)	72.860±2.819
114	L6 - R5	kurtosis(signals)	73.806±1.438
115	R1 - R6	range(D_level_5)	72.594±2.061
116	R1 - R6	median(p2_3)	73.265±2.369
117	R1 - RT	std(signals)/mean(signals)	77.983±1.718
118	L1 + L6	mean(p2_3)	74.748±1.639
119	L1 + R8	min(D_level_6)	73.674±1.545
120	L1 + R1	std(D_level_4)	73.266±2.844
121	L1 + R7	norm(D_level_1)	73.935±3.328
122	L1 + L6	wentropy(pks2_3, 'shannon')	72.329±2.328
123	L4 + L7	kurtosis(signals)	73.670±0.834
124	L4 + L5	kurtosis(signals)	74.141±1.843
125	L4 + L5	skewness(signals)	71.92 ±1.876
126	L4 + R2	median(D_level_4)	75.487±1.828
127	L4 + L8	skewness(signals)	73.941±1.728

Çizelge B.12: (devam) Coiflets2 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

128	L5 + L8	std(signals)/mean(signals)	73.130±2.749
129	L5 + L8	median(signals)	72.725±1.815
130	L5 + L8	mad(signals, 1)	72.455±1.855
131	R1 + R6	mad(pks2_4, 0)	77.039±1.686
132	R5 + R8	std(signals)/mean(signals)	75.356±2.582
133	R5 + R6	kurtosis(signals)	72.728±2.142
134	L4 + R4	median(Ed)	78.586±1.463
135	L4 + R4	Ed(4)	76.369±2.582
136	L4 + R4	mad(Ed, 1)	76.496±2.085
137	L1 - RT	std(Ed)/mean(Ed)	75.757±1.510
138	L6 + R5	Ed(5)	74.614±2.505
139	R3 - R7	kurtosis(Ed)	73.063±1.290
140	L4 + R8	Ed(5)	73.134±2.236
141	L5 + R5	sum(Ed)	72.389±1.274
142	L5 + R5	norm(Ed, 1)	72.389±1.274
143	L5 + R5	mean(Ed)	72.389±1.274
144	L7 - R5	Ed(5)	72.458±2.355
145	R3 - R7	Ed(5)	73.136±3.029
146	L7 + R5	Ed(5)	72.321±1.610
147	L5 + R8	Ed(5)	72.256±2.048

Çizelge B.13: Daubechies2 dalgacıđı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

Öznitelik No	Ayak Sinyali	Sinyale Uygulanan İstatistiksel Fonksiyon	Özniteliđin Tek Bařına Sınıflandırma Doğruluđu (%)
1	L1 - L3	range (p2_4)	78.587±1.512
2	L1 - L3	max (p2_4)	78.250±1.790
3	L1 - L3	std (D_level_3)	76.431±1.200
4	L1 - L2	mean (p2_3)	76.969±1.979
5	L1 - L3	std (p2_3)	75.492±1.947
6	L1 - L2	std (pks2_3)	76.162±2.383
7	L1 - L2	wentropy (p2_3, 'shannon')	75.557±2.242
8	L1 - L2	range (pks2_4)	74.748±1.268
9	L1-R2	wentropy (p2_3, 'shannon')	74.815±1.711
10	L1 - L2	norm (p2_3)	74.817±2.063
11	L1 - L2	norm (D_level_3)	74.479±1.335
12	L1 - L3	wentropy (p2_3, 'shannon')	75.560±2.766
13	L1 - L2	norm (pks2_3)	74.213±1.817
14	L1 - L3	range (D_level_4)	74.478±2.102
15	L1 - L3	min (D_level_3)	73.944±1.981
16	L1-R2	max (p2_3)	74.072±1.962
17	L1-R2	range (p2_3)	74.072±1.962
18	L1-R4	max (p2_3)	74.345±1.828
19	L1-R4	range (p2_3)	74.345±1.828
20	L1 - L2	std (D_level_3)	73.874±2.048
21	L1-L6	range (D_level_4)	73.805±2.454
22	L1 - L2	std (p2_3)	73.741±2.318
23	L1-L6	mean (p2_3)	73.469±1.960
24	L1 - L4	mean (w2_1)	73.739±2.405
25	L1 - L3	norm (p2_3)	73.472±1.745
26	L1-L6	mean (pks2_3)	73.810±3.004
27	L1 - L2	wentropy (pks2_3, 'shannon')	73.132±1.212
28	L1 - L4	min (D_level_3)	73.943±2.329

Çizelge B.13: (devam) Daubechies2 dalgacıği kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

29	L1-L6	min(D_level_5)	73.404±1.790
30	L1 - L3	norm(D_level_3)	73.471±2.571
31	L1 - L2	std(p2_4)	74.345±2.859
32	L1 - L3	mad(D_level_3, 0)	74.010±2.920
33	L1-L6	min(D_level_4)	73.803±2.467
34	L1-R4	std(D_level_3)	73.471±2.267
35	L1 - L2	mean(pks2_3)	73.133±1.961
36	L1 - L3	std(p2_4)	72.865±2.174
37	L1-L6	std(D_level_3)	73.333±2.432
38	L1 - L4	wentropy(D_level_3, 'shannon')	73.202±2.360
39	L1-R4	wentropy(D_level_2, 'shannon')	73.265±1.714
40	L1-R2	std(p2_4)	72.864±1.698
41	L1-L8	min(D_level_3)	72.527±1.828
42	L1 - L3	wentropy(p2_2, 'shannon')	72.928±2.182
43	L1 - L3	wentropy(D_level_3, 'shannon')	72.729±1.970
44	L1 - L2	mad(D_level_3, 0)	72.863±2.134
45	L1 - L3	mad(p2_3, 0)	73.273±2.888
46	L1-R6	mad(pks2_2, 0)	72.260±1.954
47	L1 - L3	norm(p)	72.125±2.085
48	L1 - L2	range(pks2_3)	72.792±2.337
49	L1-R6	std(pks2_2)	72.794±2.162
50	L4-R7	std(w2_4)	75.083±1.686
51	L4-R7	norm(w2_4)	74.682±2.955
52	L5-R5	mad(w2_3, 0)	73.470±1.787
53	L4-R7	mean(w2_4)	73.465±1.926
54	L5-R6	wentropy(w, 'shannon')	73.402±2.178
55	L4-R6	std(w2_4)	73.468±2.260
56	L4-R6	wentropy(w2_4, 'shannon')	72.799±2.323
57	L5-R5	wentropy(w2_3, 'shannon')	72.799±2.219
58	L6-R1	std(D_level_2)	73.532±2.727
59	L6-R1	std(p2_2)	72.929±2.487
60	L8-R5	mad(w2_3, 0)	74.747±2.507
61	R1-RT	std(signals)/mean(signals)	78.789±1.903

Çizelge B.13: (devam) Daubechies2 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

62	R1-R2	min(D_level_4)	74.612±1.179
63	R1-R2	std(p2_4)	72.998±1.826
64	R1-R6	mean(p2_4)	72.260±2.052
65	R3-R6	norm(w)	75.219±0.790
66	R3-R6	sum(locs2_3)	73.270±1.721
67	R3-R6	norm(locs2_3, 1)	73.270±1.721
68	R3-R6	wentropy(w, 'shannon')	72.186±1.850
69	L1-L6	std(p2_4)	75.755±2.086
70	L1-L6	min(D_level_4)	74.750±1.922
71	L1-L6	mean(pks)	74.416±2.425
72	L1-L2	std(p2_3)	74.674±2.454
73	L1-L6	std(signals)	73.397±1.564
74	L1-L5	range(D_level_3)	73.332±1.729
75	L1-L6	mean(pks2_3)	73.132±1.562
76	L1-L6	mad(p2_3, 0)	74.209±2.659
77	L1-L6	median(pks)	73.200±1.950
78	L1-L4	std(p2_3)	73.335±2.417
79	L1-L6	range(D_level_4)	72.324±1.727
80	L1-L6	std(p)	72.396±2.390
81	L1-L5	min(D_level_3)	72.192±1.969
82	L1-L5	mad(p2_4, 0)	72.120±2.062
83	L1-L6	std(D_level_6)	71.717±1.645
84	L1-L6	std(D_level_5)	71.315±0.993
85	L1-R7	sum(pks)	73.067±1.904
86	L1-R2	range(D_level_3)	73.398±2.367
87	L1-R8	min(D_level_3)	72.259±1.556
88	L1-R6	wentropy(D_level_3, 'shannon')	72.256±2.176
89	L1-R6	range(p)	71.448±1.281
90	L6-R1	mad(pks2_4, 0)	72.859±2.621
91	L6-R1	median(D_level_3)	72.391±2.187
92	L6-R5	kurtosis(signals)	73.806±1.438
93	L7-R1	mad(p2_4, 1)	73.874±2.148
94	R1-RT	std(signals)/mean(signals)	77.983±1.718

Çizelge B.13: (devam) Daubechies2 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

95	L1+R8	range (p2_3)	72.524±2.247
96	L1+R8	max (p2_3)	72.658±2.334
97	L4+L7	kurtosis (signals)	73.670±0.834
98	L4+L5	kurtosis (signals)	74.141±1.843
99	L4+L5	skewness (signals)	71.920±1.876
100	L4+L8	skewness (signals)	73.941±1.728
101	L5+L8	std(signals)/mean(signals)	73.130±2.749
102	L5+L8	mad(signals, 1)	72.455±1.855
103	L5+L8	median(signals)	72.725±1.815
104	L4+R6	wentropy(w2_4, 'shannon')	72.122±2.031
105	R5+R8	std(signals)/mean(signals)	75.356±2.582
106	R5+R6	kurtosis (signals)	72.728±2.142
107	L5+R5	Ed (4)	77.650±2.400
108	L7-R5	Ed (5)	76.566±1.335
109	L5+R5	mad(Ed, 1)	76.769±2.334
110	L4+R5	Ed (4)	76.636±2.683
111	L4+R4	mad(Ed, 1)	75.691±2.554
112	L4+R4	median(Ed)	75.556±2.879
113	L4+R4	Ed (4)	74.481±2.678
114	L4+R4	Ed (3)	74.346±2.306
115	L1-RT	std(Ed)/mean(Ed)	74.079±3.345
116	L4+R5	mad(Ed, 1)	72.729±1.813
117	L5+R6	Ed (5)	73.404±2.153
118	L4+R5	median(Ed)	73.338±2.494
119	L4+R5	wentropy(Ed, 'log energy')	73.329±2.340
120	L6+R5	median(Ed)	72.793±1.880
121	L7+R5	Ed (5)	73.002±2.503
122	L5+R5	median(Ed)	72.996±1.698
123	L6+R5	Ed (5)	72.729±2.341
124	R1-LT	skewness(Ed)	72.324±1.905
125	L7-R5	Ea	71.920±1.586

Çizelge B.14: Daubechies3 dalgacıği kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

Öznitelik No	Ayak Sinyali	Sinyale Uygulanan İstatistiksel Fonksiyon	Özniteliğin Tek Başına Sınıflandırma Doğruluğu (%)
1	R1	median(w2_1)	72.524±1.829
2	L1 - L2	std(D_level_3)	77.847±1.388
3	L1 - L2	wentropy(D_level_3, 'shannon')	77.713±2.172
4	L1 - L2	range(pks2_4)	77.173±1.757
5	L1 - L2	mad(p2_3, 0)	77.237±1.561
6	L1 - L3	std(pks2_4)	76.907±2.123
7	L1 - L3	std(D_level_4)	76.701±1.929
8	L1 - L3	range(pks2_4)	76.095±1.811
9	L1 - L2	wentropy(pks2_3, 'shannon')	75.963±1.884
10	L1 - L2	max(D_level_4)	76.297±1.996
11	L1 - L2	range(D_level_4)	75.692±1.493
12	L1 - L2	norm(D_level_3)	75.963±2.090
13	L1 - L2	max(pks2_4)	75.759±1.978
14	L1 - R6	range(pks2_3)	75.627±2.357
15	L1 - L2	std(pks2_3)	75.157±2.278
16	L1 - R6	std(pks2_2)	75.156±2.310
17	L1 - L2	mean(pks2_3)	75.154±2.462
18	L1 - L2	std(p2_3)	74.678±1.714
19	L1 - L3	max(p2_4)	75.154±2.669
20	L1 - L2	sum(p2_3)	75.151±2.234
21	L1 - L2	norm(pks2_3)	75.021±2.590
22	L1 - L2	wentropy(p2_3, 'shannon')	74.411±1.333
23	L1 - L2	norm(p2_3, 1)	75.151±2.234
24	L1 - L3	std(p2_3)	74.009±2.038
25	L1 - L3	range(p2_4)	74.683±2.449
26	L1 - L3	max(pks2_4)	74.480±2.699
27	L1 - L3	max(D_level_4)	74.480±2.699
28	L1 - R2	range(p2_4)	74.685±2.787

Çizelge B.14: (devam) Daubechies3 dalgacıği kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

29	L1 - R6	max(pks2_3)	74.544±1.762
30	L1 - L6	std(pks2_3)	73.805±1.171
31	L1 - L2	norm(p2_3)	74.142±2.245
32	L1 - L3	mad(pks2_4, 0)	73.538±1.734
33	L1 - R2	max(p2_4)	74.011±2.409
34	L1 - L2	mad(D_level2_3, 0)	73.736±1.902
35	L1 - L4	std(pks2_4)	74.073±2.894
36	L1 - R6	norm(pks2_2)	73.606±2.746
37	L1 - R6	max(D_level_3)	73.670±1.572
38	L1 - L2	mean(p2_3)	73.266±1.540
39	L1 - R4	mad(p2_3, 0)	73.338±2.101
40	L1 - L5	min(D_level_3)	73.601±2.036
41	L1 - R2	max(D_level_4)	73.132±1.350
42	L1 - R2	max(pks2_4)	73.132±1.350
43	L1 - L3	wentropy(D_level_4, 'shannon')	73.537±2.464
44	L1 - R6	mad(D_level_2, 0)	72.999±1.749
45	L1 - L6	std(pks2_4)	73.132±2.144
46	L1 - L4	mad(pks2_4, 0)	73.199±2.622
47	L1 - R2	range(D_level_4)	73.873±3.091
48	L1 - L3	std(p2_4)	72.797±1.900
49	L1 - L6	mean(pks2_5)	73.472±2.929
50	L1 - L6	range(pks2_4)	73.265±2.082
51	L1 - R4	max(p2_3)	72.731±2.272
52	L1 - L4	std(p2_3)	72.729±2.235
53	L1 - R4	range(p2_3)	72.664±2.346
54	L1 - L3	wentropy(pks2_3, 'shannon')	72.663±2.112
55	L1 - L6	mad(pks2_4, 0)	72.596±1.800
56	L1 - L4	wentropy(pks2_3, 'shannon')	72.389±1.746
57	L1 - L3	norm(p)	72.125±2.085
58	L1 - R4	std(pks2_3)	72.664±2.623
59	L1 - L6	min(D_level_5)	72.257±1.673
60	L1 - L6	std(p2_3)	72.594±2.489
61	L1 - R4	std(D_level_3)	72.056±1.979

Çizelge B.14: (devam) Daubechies3 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

62	L1 - L2	range (p2_3)	71.720±1.659
63	L1 - L2	max (p2_3)	71.720±1.659
64	L1 - R6	range (p2_2)	71.851±1.842
65	L1 - R6	max (p2_2)	71.851±1.842
66	L5 - R6	wentropy(w, 'shannon')	73.402±2.178
67	L6 - R1	mad(pks2_2, 0)	76.297±1.520
68	L6 - R1	std(p2_2)	74.007±1.808
69	L6 - R1	std(D_level_2)	74.547±3.007
70	L6 - R1	mad(p2_2, 0)	72.931±2.106
71	L6 - R1	wentropy(pks2_2, 'shannon')	72.257±1.883
72	L6 - R1	max(pks2_3)	72.056±1.929
73	R1 - RT	std(signals)/mean(signals)	78.789±1.903
74	R1 - R6	median(p2_4)	74.212±1.831
75	R1 - R6	mad(p2_3, 1)	72.594±1.710
76	R1 - R3	median(w2_1)	73.134±2.403
77	R1 - R3	std(pks2_4)	72.930±2.352
78	R3 - R6	norm(w)	75.219±0.790
79	R3 - R6	wentropy(w, 'shannon')	72.186±1.850
80	L1 - L6	std(D_level_6)	74.009±1.195
81	L1 - L6	mean(pks)	74.416±2.425
82	L1 - L5	min(D_level_3)	73.534±1.967
83	L1 - L3	range(pks2_4)	73.938±2.518
84	L1 - R1	skewness(p2_2)	73.067±1.564
85	L1 - L6	std(signals)	73.397±1.564
86	L1 - L3	std(pks2_4)	73.133±2.556
87	L1 - L6	median(pks)	73.200±1.950
88	L1 - L6	max(p2_4)	72.728±1.673
89	L1 - L4	norm(D_level_3)	72.797±1.863
90	L1 - L6	range(p2_4)	72.392±1.651
91	L1 - L4	std(D_level_3)	72.192±1.826
92	L1 - L4	min(D_level_3)	72.724±2.251
93	L1 - L6	std(p)	72.396±2.390
94	L1 - L2	mad(D_level_3, 0)	72.924±2.756

Çizelge B.14: (devam) Daubechies3 dalgacıđı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıřtır)

95	L1 - L2	norm(pks2_3, 1)	72.257±1.471
96	L1 - R1	kurtosis(p2_2)	73.202±3.196
97	L1 - R6	range(D_level_4)	74.278±2.336
98	L1 - R6	std(p2_4)	74.078±1.883
99	L1 - R6	std(D_level_4)	73.740±2.115
100	L1 - R6	std(D_level_3)	74.009±2.446
101	L1 - R6	min(D_level_4)	73.665±2.679
102	L1 - R7	sum(pks)	73.067±1.904
103	L1 - R6	range(p)	71.448±1.281
104	L6 - R1	range(pks2_5)	73.805±0.925
105	L6 - R1	mad(p2_4, 0)	73.875±2.526
106	L6 - R1	mad(pks2_3, 0)	72.593±1.948
107	L6 - R5	kurtosis(signals)	73.806±1.438
108	R1 - R4	median(w2_1)	73.670±1.796
109	R1 - R3	kurtosis(pks2_2)	72.660±1.709
110	R1 - R2	kurtosis(p2_2)	72.255±2.135
111	R1 - RT	std(signals)/mean(signals)	77.983±1.718
112	L1 + R2	range(p2_4)	76.639±2.187
113	L1 + R2	max(p2_4)	76.435±2.126
114	L1 + R2	range(D_level_4)	73.537±1.827
115	L1 + L2	mad(D_level_5, 1)	72.795±2.123
116	L1 + L7	max(pks2_3)	73.134±2.410
117	L1 + L7	max(D_level_3)	73.134±2.410
118	L1 + R3	mad(pks2_3, 0)	72.658±2.092
119	L1 + R2	std(p2_4)	72.458±2.416
120	L4 + L7	kurtosis(signals)	73.670±0.834
121	L4 + L5	kurtosis(signals)	74.141±1.843
122	L4 + L5	skewness(signals)	71.920±1.876
123	L4 + L8	skewness(signals)	73.941±1.728
124	L5 + L8	std(signals)/mean(signals)	73.130±2.749
125	L5 + L8	median(signals)	72.725±1.815
126	L5 + L8	mad(signals, 1)	72.455±1.855
127	R1 + R5	norm(locs2_1)	73.607±1.782

Çizelge B.14: (devam) Daubechies3 dalgacıđı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz dođrulama kullanılmıřtır)

128	R1 + R6	mad(p2_4, 1)	74.816±3.209
129	R1 + R5	norm(locs2_1, 1)	73.133±1.682
130	R1 + R5	sum(locs2_1)	73.133±1.682
131	R1 + R5	mad(p2_4, 0)	72.862±2.49
132	R5 + R8	std(signals)/mean(signals)	75.356±2.582
133	R5 + R6	kurtosis(signals)	72.728±2.142
134	L4 + R4	mad(Ed, 1)	76.901±2.183
135	L4 + R4	median(Ed)	75.761±2.407
136	L4 + R4	Ed(4)	75.421±2.233
137	L4 + L6	skewness(Ed)	74.545±2.366
138	L4 + R8	Ed(5)	73.064±2.329
139	L4 + R5	median(Ed)	72.392±2.095
140	L7 - R7	std(Ed)/mean(Ed)	71.986±1.684
141	L4 - R4	Ed(4)	72.867±2.622
142	L6 + R5	Ed(5)	73.405±2.750

Çizelge B.15: Daubechies4 dalgacıği kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

Öznitelik No	Ayak Sinyali	Sinyale Uygulanan İstatistiksel Fonksiyon	Özniteliğın Tek Başına Sınıflandırma Doğruluđu (%)
1	L1 - L3	std(D_level_4)	78.453±1.734
2	L1 - L3	mad(pks2_4, 0)	78.049±1.635
3	L1 - L3	std(pks2_4)	77.910±1.373
4	L1 - L3	max(D_level_4)	77.509±2.467
5	L1 - L3	wentropy(p2_3, 'shannon')	77.238±1.132
6	L1 - L3	max(pks2_4)	77.509±2.467
7	L1 - R6	std(p2_2)	77.174±1.486
8	L1 - L3	mean(pks2_4)	76.769±1.436
9	L1 - L3	std(pks2_3)	76.703±1.275
10	L1 - L3	max(p2_4)	76.432±1.633
11	L1 - L3	norm(pks2_4)	75.960±1.747
12	L1 - L3	range(p2_4)	75.961±1.706
13	L1 - L3	range(D_level_4)	75.960±2.597
14	L1 - L3	std(p2_4)	75.624±1.200
15	L1 - L2	mad(pks2_3, 0)	75.761±2.213
16	L1 - L2	norm(p2_3)	75.824±2.197
17	L1 - R4	std(p2_3)	75.892±2.388
18	L1 - L3	mad(p2_4, 0)	75.561±2.337
19	L1 - L2	wentropy(D_level_3, 'shannon')	75.286±1.625
20	L1 - L2	mean(p2_3)	75.219±1.696
21	L1 - L3	wentropy(pks2_3, 'shannon')	75.289±1.884
22	L1 - L3	sum(pks2_3)	75.422±1.583
23	L1 - L2	max(D_level_4)	75.222±2.154
24	L1 - L3	norm(pks2_3)	75.086±2.172
25	L1 - L2	max(pks2_4)	75.222±2.154
26	L1 - L2	mad(p2_3, 0)	74.814±1.798
27	L1 - L3	wentropy(pks2_4, 'shannon')	74.818±2.421
28	L1 - L3	norm(pks2_3, 1)	75.085±1.745

Çizelge B.15: (devam) Daubechies4 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

29	L1 - L3	range(pks2_4)	75.291±3.200
30	L1 - L4	std(pks2_4)	74.615±1.697
31	L1 - L2	std(p2_4)	74.616±2.068
32	L1 - L2	wentropy(p2_3, 'shannon')	74.478±1.632
33	L1 - L2	std(pks2_4)	74.479±2.210
34	L1 - L4	mad(pks2_4, 0)	74.212±1.757
35	L1 - L6	std(pks2_4)	74.814±1.942
36	L1 - L3	mad(D_level_3, 0)	74.014±2.631
37	L1 - L6	range(D_level_3)	74.277±1.846
38	L1 - L4	range(pks2_4)	74.479±2.206
39	L1 - R5	std(D_level_3)	73.940±1.873
40	L1 - L3	std(p2_3)	74.078±2.662
41	L1 - L2	norm(D_level_3)	74.007±2.150
42	L1 - L6	mad(pks2_4, 0)	74.008±1.871
43	L1 - L3	mad(p2_3, 0)	73.943±2.428
44	L1 - L2	std(p2_3)	73.601±1.972
45	L1 - L2	std(D_level_3)	74.206±2.310
46	L1 - L3	norm(p2_3)	73.336±1.767
47	L1 - L6	mean(p2_3)	73.203±2.519
48	L1 - L2	mean(pks2_3)	73.742±2.876
49	L1 - L6	range(pks2_4)	73.406±2.629
50	L1 - L2	max(p2_3)	73.398±1.720
51	L1 - L2	max(p2_4)	73.537±2.357
52	L1 - L2	wentropy(pks2_3, 'shannon')	73.400±2.367
53	L1 - L3	wentropy(D_level_4, 'shannon')	72.866±2.269
54	L1 - R4	std(pks2_3)	73.468±2.635
55	L1 - L2	range(p2_3)	73.398±1.720
56	L1 - R4	wentropy(p2_3, 'shannon')	72.999±2.515
57	L1 - L3	norm(D_level_3, 1)	72.592±1.668
58	L1 - L2	range(D_level_3)	73.268±2.774
59	L1 - R4	std(D_level_3)	73.204±2.800
60	L1 - L2	range(p2_4)	73.402±2.461
61	L1 - L2	mad(p2_4, 0)	73.134±2.606

Çizelge B.15: (devam) Daubechies4 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

62	L1 - L2	mad(D_level_3, 0)	72.792±1.393
63	L1 - L2	norm(D_level_3, 1)	72.864±2.552
64	L1 - L2	norm(p2_4)	72.593±1.225
65	L1 - L5	std(p2_4)	73.338±3.137
66	L1 - L5	mad(pks2_4, 0)	72.595±2.071
67	L1 - R6	std(D_level_2)	72.527±2.248
68	L1 - R4	wentropy(pks2_3, 'shannon')	73.063±2.303
69	L1 - R5	std(p2_3)	72.529±2.813
70	L1 - L3	std(D_level_3)	72.596±2.060
71	L1 - L2	norm(pks2_3)	72.393±2.032
72	L1 - L3	wentropy(D_level_3, 'shannon')	72.189±1.486
73	L1 - R4	wentropy(D_level_3, 'shannon')	72.123±1.927
74	L1 - L4	max(pks2_4)	72.590±2.244
75	L1 - L3	norm(p)	72.125±2.085
76	L1 - L4	max(D_level_4)	72.590±2.244
77	L1 - L2	range(D_level_2)	72.521±2.394
78	L1 - R6	mad(pks2_2, 0)	72.926±2.271
79	L1 - L2	norm(pks2_3, 1)	72.459±1.989
80	L1 - L6	mean(pks2_4)	71.920±1.643
81	L1 - L6	min(D_level_5)	72.256±2.025
82	L1 - R1	mad(p2_4, 1)	72.055±2.027
83	L1 - L2	std(D_level_4)	71.984±1.897
84	L5 - R6	wentropy(w, 'shannon')	73.402±2.178
85	L4 - R7	median(D_level_5)	72.459±1.719
86	L6 - R1	std(D_level_2)	75.220±3.140
87	L6 - R1	range(D_level_3)	73.466±2.466
88	L6 - R1	wentropy(p2_2, 'shannon')	72.866±2.541
89	R1 - RT	std(signals)/mean(signals)	78.789±1.903
90	R1 - R6	mad(D_level_4, 1)	73.805±1.961
91	R1 - R6	std(p2_4)	74.410±1.662
92	R2 - R7	mad(w2_4, 0)	72.862±2.410
93	R1 - R6	max(p2_4)	72.529±2.381
94	R1 - R2	range(p2_4)	71.919±1.566

Çizelge B.15: (devam) Daubechies4 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

95	R1 - R6	mad(p2_4, 0)	73.335±3.098
96	R1 - R8	mad(pks2_4, 0)	72.323±1.985
97	R1 - R2	max(p2_4)	71.718±1.692
98	R3 - R7	wentropy(w2_4, 'log energy')	75.894±1.619
99	R3 - R6	norm(w)	75.219±0.790
100	R3 - R7	mean(w2_4)	73.536±1.270
101	R3 - R6	wentropy(w, 'shannon')	72.186±1.850
102	L1 - L6	mad(p2_4, 0)	75.963±1.575
103	L1 - L3	std(pks2_4)	74.818±2.605
104	L1 - L6	mean(pks)	74.416±2.425
105	L1 - L6	std(p2_4)	74.143±2.194
106	L1 - L5	range(D_level_3)	73.736±1.581
107	L1 - L4	wentropy(pks2_3, 'shannon')	73.876±2.456
108	L1 - L6	std(pks2_4)	73.672±1.710
109	L1 - L3	wentropy(pks2_4, 'shannon')	73.604±1.804
110	L1 - L2	mad(p2_3, 0)	73.602±1.594
111	L1 - L2	std(pks2_4)	73.471±1.890
112	L1 - L6	std(D_level_6)	73.605±2.030
113	L1 - R1	kurtosis(D_level_3)	73.738±2.222
114	L1 - L6	mad(D_level_6, 0)	73.804±1.782
115	L1 - L4	range(pks2_6)	73.332±1.595
116	L1 - L6	std(signals)	73.397±1.564
117	L1 - L4	wentropy(p2_3, 'shannon')	72.930±1.979
118	L1 - L6	mad(pks2_4, 0)	73.200±2.294
119	L1 - L6	median(pks)	73.200±1.950
120	L1 - L4	mad(pks2_3, 0)	73.333±2.740
121	L1 - L4	mad(p2_3, 0)	73.202±3.055
122	L1 - L4	std(pks2_3)	72.529±1.917
123	L1 - L2	mad(D_level_3, 0)	72.724±2.065
124	L1 - L3	max(pks2_4)	72.864±2.683
125	L1 - L3	max(D_level_4)	72.864±2.683
126	L1 - L6	std(p)	72.396±2.390
127	L1 - R6	std(p2_4)	75.755±2.652

Çizelge B.15: (devam) Daubechies4 dalgacıđı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıřtır)

128	L1 - R6	norm(D_level_3)	74.143±1.085
129	L1 - R6	mad(pks2_4, 0)	74.279±2.518
130	L1 - R6	min(D_level_4)	72.863±1.385
131	L1 - R6	range(p2_4)	73.808±2.616
132	L1 - R7	sum(pks)	73.067±1.904
133	L1 - R6	range(pks2_3)	73.404±2.258
134	L1 - R6	max(pks2_4)	73.198±2.153
135	L1 - R6	max(D_level_4)	73.198±2.153
136	L1 - R6	max(p2_4)	73.000±2.459
137	L1 - R6	norm(pks2_3)	72.326±1.558
138	L1 - R6	norm(p2_3)	71.920±1.766
139	L1 - R6	range(p)	71.448±1.281
140	L6 - R5	kurtosis(signals)	73.939±1.418
141	R1 - R4	mad(pks2_4, 0)	74.075±2.747
142	R1 - R6	range(D_level_5)	72.929±1.899
143	R1 - R6	min(D_level_5)	73.195±2.146
144	R1 - RT	std(signals)/mean(signals)	77.983±1.718
145	L1 + R2	mad(pks2_4, 0)	76.162±1.345
146	L1 + R2	min(D_level_4)	74.208±1.888
147	L1 + L5	std(w2_1)	72.932±1.679
148	L1 + R8	norm(p2_6)	73.131±1.103
149	L1 + R1	kurtosis(D_level_3)	72.798±2.172
150	L4 + L7	kurtosis(signals)	73.67 ±0.834
151	L4 + L5	kurtosis(signals)	74.141±1.843
152	L4 + L5	skewness(signals)	71.920±1.876
153	L4 + L8	skewness(signals)	73.941±1.728
154	L5 + L8	std(signals)/mean(signals)	72.996±2.857
155	L5 + L8	median(signals)	72.725±1.815
156	L5 + L8	mad(signals, 1)	72.455±1.855
157	R5 + R8	std(signals)/mean(signals)	75.356±2.582
158	R5 + R6	kurtosis(signals)	72.728±2.142
159	L4 + R4	median(Ed)	76.631±2.271
160	L4 + R4	Ed(4)	77.310±2.985

Çizelge B.15: (devam) Daubechies4 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

161	L4 + R4	mad(E _d , 1)	76.028±1.721
162	R4 + R8	Ed (5)	75.627±2.440
163	L1 - RT	skewness (Ed)	74.949±2.701
164	L7 - R7	std(E _d) /mean (Ed)	73.804±1.975
165	L4 + R4	wentropy (Ed, 'log energy')	73.805±1.762
166	L5 + R4	Ed (4)	73.876±2.781
167	L1 - RT	std(E _d) /mean (Ed)	73.801±2.567
168	L6 - R7	std(E _d) /mean (Ed)	73.202±2.184
169	L4 - R4	Ed (4)	73.402±2.773
170	R2 - R6	skewness (Ed)	72.597±1.952
171	L4 + R5	Ed (4)	72.057±1.717
172	L6 + R5	Ed (5)	72.523±2.090

Çizelge B.16: Daubechies5 dalgacıđı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıřtır)

Öznitelik No	Ayak Sinyali	Sinyale Uygulanan İstatistiksel Fonksiyon	Özniteliđin Tek Bařına Sınıflandırma Doğruluđu (%)
1	L1	wentropy(p2_3, 'shannon')	72.123±1.947
2	L1 - L3	std(D_level_4)	78.452±1.104
3	L1 - L3	std(p2_4)	77.577±0.968
4	L1 - L2	mad(pks2_4, 0)	75.355±1.495
5	L1 - L2	std(p2_4)	75.758±2.013
6	L1 - L3	range(p2_4)	75.286±1.129
7	L1 - L2	mad(p2_3, 0)	75.491±2.125
8	L1 - L2	mean(p2_3)	75.356±2.353
9	L1 - L3	max(p2_4)	75.084±1.004
10	L1 - L3	mean(p2_4)	75.288±2.132
11	L1 - L2	min(D_level_3)	75.491±2.302
12	L1 - L3	range(D_level_4)	75.555±1.431
13	L1 - L3	std(pks2_4)	74.885±1.790
14	L1 - R2	max(D_level_4)	75.218±1.734
15	L1 - R2	max(pks2_4)	75.218±1.734
16	L1 - L3	mad(D_level_4, 0)	75.286±2.315
17	L1 - L3	mean(pks2_3)	75.017±2.221
18	L1 - L3	mean(pks2_4)	74.480±1.547
19	L1 - L2	wentropy(D_level_3, 'shannon')	74.749±1.862
20	L1 - L3	max(pks2_3)	74.211±1.991
21	L1 - L3	max(D_level_3)	74.211±1.991
22	L1 - L3	max(pks2_4)	74.011±1.798
23	L1 - L3	mean(p2_3)	74.078±1.784
24	L1 - R6	std(D_level_2)	73.739±1.245
25	L1 - L6	range(p2_5)	74.479±2.107
26	L1 - R4	mad(p2_3, 0)	73.807±1.645
27	L1 - L3	max(D_level_4)	74.011±1.798
28	L1 - R4	std(p2_3)	74.751±2.644

Çizelge B.16: (devam) Daubechies5 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

29	L1 - L2	std(pks2_4)	74.142±2.322
30	L1 - L3	mad(pks2_4, 0)	73.539±2.130
31	L1 - R2	max(p2_4)	73.738±2.271
32	L1 - L4	std(p2_3)	73.403±2.042
33	L1 - L3	std(D_level_3)	74.481±3.361
34	L1 - R2	range(p2_4)	74.007±2.596
35	L1 - R4	std(D_level_3)	74.014±2.917
36	L1 - L2	mad(pks2_3, 0)	73.401±1.929
37	L1 - L3	mad(p2_4, 0)	73.741±2.730
38	L1 - L6	mean(pks2_4)	73.068±1.739
39	L1 - L2	std(p2_3)	74.070±2.509
40	L1 - L3	range(pks2_4)	73.334±2.267
41	L1 - R5	std(D_level_3)	73.536±2.737
42	L1 - R2	std(p2_4)	73.471±2.215
43	L1 - L2	max(p2_4)	73.470±2.278
44	L1 - L4	std(pks2_4)	73.747±3.466
45	L1 - L2	std(D_level_3)	73.066±1.749
46	L1 - L6	mad(pks2_5, 0)	73.270±2.290
47	L1 - R4	mad(pks2_3, 0)	73.402±2.561
48	L1 - R4	wentropy(p2_3, 'shannon')	73.472±2.717
49	L1 - L2	range(pks2_3)	72.794±0.965
50	L1 - L2	range(p2_4)	72.861±1.657
51	L1 - R6	mad(p2_2, 0)	72.664±2.171
52	L1 - R5	std(p2_3)	72.528±1.815
53	L1 - R4	wentropy(D_level_3, 'shannon')	74.147±4.016
54	L1 - L2	norm(pks2_3)	72.594±2.066
55	L1 - L3	wentropy(pks2_3, 'shannon')	72.328±2.188
56	L1 - R2	mad(pks2_4, 0)	72.663±2.358
57	L1 - L2	range(D_level_4)	72.525±1.948
58	L1 - L3	sum(pks2_3)	72.327±1.932
59	L1 - L6	std(pks2_5)	72.525±1.744
60	L1 - R1	skewness(pks2_3)	72.456±2.077
61	L1 - L3	norm(p)	72.125±2.085

Çizelge B.16: (devam) Daubechies5 dalgacıği kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

62	L1 - L2	max(D_level_3)	71.990±1.826
63	L1 - R5	std(p2_1)	72.257±1.943
64	L1 - L3	range(pks2_3)	71.986±1.542
65	L1 - L2	max(pks2_3)	71.99 ±1.826
66	L1 - L6	max(p2_5)	72.861±2.619
67	L1 - L2	norm(D_level_3, 1)	71.783±1.693
68	L2 - R6	std(pks2_2)	73.537±2.049
69	L5 - R6	wentropy(w, 'shannon')	73.402±2.178
70	L6 - R1	std(D_level_2)	72.592±1.627
71	R1 - RT	std(signals)/mean(signals)	78.789±1.903
72	R1 - R6	mad(pks2_3, 1)	75.958±1.274
73	R1 - R3	mad(pks2_3, 0)	73.201±2.601
74	R1 - R2	std(p2_4)	72.797±2.083
75	R1 - R3	mad(D_level_3, 0)	71.854±1.769
76	R1 - R3	std(pks2_4)	73.265±2.657
77	R3 - R6	norm(w)	75.219±0.790
78	R3 - R6	wentropy(w, 'shannon')	72.186±1.850
79	L1 - L2	range(pks2_3)	74.948±2.362
80	L1 - L6	mean(pks)	74.416±2.425
81	L1 - L4	norm(p2_3)	73.738±2.010
82	L1 - L2	mad(pks2_3, 0)	73.940±1.694
83	L1 - L6	wentropy(p2_3, 'shannon')	74.010±3.005
84	L1 - L6	std(signal)	73.397±1.564
85	L1 - L2	wentropy(pks2_4, 'shannon')	73.134±1.703
86	L1 - L6	median(pks)	73.200±1.950
87	L1 - L6	wentropy(D_level_4, 'shannon')	73.000±2.435
88	L1 - L3	std(pks2_4)	72.798±2.545
89	L1 - L6	std(p)	72.396±2.390
90	L1 - L2	std(p2_4)	72.390±2.033
91	L1 - L4	mean(p2_5)	72.254±2.223
92	L1 - L3	range(pks2_3)	71.985±1.975
93	L1 - L4	min(D_level_3)	71.581±1.516
94	L1 - L6	range(p2_4)	71.715±1.387

Çizelge B.16: (devam) Daubechies5 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

95	L1 - R6	range (pks2_4)	75.761±2.559
96	L1 - R6	mad (pks2_4, 0)	75.020±2.572
97	L1 - R7	sum (pks)	73.067±1.904
98	L1 - R4	std (pks2_4)	73.064±2.279
99	L1 - R6	std (pks2_4)	72.863±2.241
100	L1 - R6	max (D_level_4)	72.123±2.060
101	L1 - R6	max (pks2_4)	72.123±2.060
102	L1 - R6	range (p)	71.448±1.281
103	L6 - R5	kurtosis (signals)	73.939±1.418
104	R1 - RT	std (signals) / mean (signals)	77.983±1.718
105	R2 - R7	skewness (D_level_5)	72.525±1.464
106	L1 + R2	max (D_level_4)	74.008±2.508
107	L1 + R2	max (pks2_4)	74.008±2.508
108	L1 + R2	std (p2_4)	74.008±1.958
109	L1 + R1	kurtosis (D_level_3)	73.606±1.811
110	L1 + L6	std (pks2_3)	72.928±1.546
111	L1 + R1	std (pks2_4)	73.669±2.956
112	L1 + RT	kurtosis (p2_3)	73.336±2.249
113	L4 + L7	kurtosis (signals)	73.670±0.834
114	L4 + L5	kurtosis (signals)	74.141±1.843
115	L4 + L5	skewness (signals)	71.920±1.876
116	L4 + L8	skewness (signals)	73.941±1.728
117	L5 + L8	median (signals)	72.725±1.815
118	L5 + L8	mad (signals, 1)	72.455±1.855
119	L5 + L8	std (signals) / mean (signals)	72.996±2.857
120	R1 + R5	max (pks2_4)	72.860±2.411
121	R1 + R5	max (D_level_4)	72.860±2.411
122	R5 + R8	std (signals) / mean (signals)	75.356±2.582
123	R5 + R6	kurtosis (signals)	72.728±2.142
124	L4 + R4	Ed (4)	77.442±1.467
125	L4 + R4	mad (Ed, 1)	76.226±1.636
126	R4 + R8	Ed (5)	74.278±1.495
127	L4 + R4	median (Ed)	75.016±2.468

Çizelge B.16: (devam) Daubechies5 dalgacıđı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

128	R3 - R7	Ed (5)	73.938±1.924
129	R3 - R6	sum (Ed)	73.604±2.502
130	R3 - R6	norm (Ed, 1)	73.604±2.502
131	R3 - R6	mean (Ed)	73.604±2.502
132	L1 - RT	std (Ed) /mean (Ed)	73.806±2.510



Çizelge B.17: Daubechies6 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

Öznelik No	Ayak Sinyali	Sinyale Uygulanan İstatistiksel Fonksiyon	Özneliğın Tek Başına Sınıflandırma Doğruluğı (%)
1	L1	max(D_level_4)	71.923±1.694
2	L1	max(pks2_4)	71.923±1.694
3	L1 - L2	min(D_level_4)	79.126±1.605
4	L1 - L3	std(p2_4)	77.779±1.175
5	L1 - L3	std(D_level_4)	77.577±1.101
6	L1 - L3	mad(p2_4, 0)	76.972±1.074
7	L1 - L2	std(p2_3)	77.507±1.822
8	L1 - L2	mad(p2_3, 0)	76.702±0.868
9	L1 - L2	std(D_level_3)	76.768±2.271
10	L1 - R5	std(p2_3)	76.299±1.597
11	L1 - L3	mad(D_level_4, 0)	75.826±1.644
12	L1 - R4	mad(pks2_3, 0)	76.500±2.727
13	L1 - L4	wentropy(p2_3, 'shannon')	75.356±2.086
14	L1 - R6	range(p2_2)	75.422±1.992
15	L1 - R4	std(p2_3)	75.966±2.927
16	L1 - R6	max(p2_2)	75.422±1.992
17	L1 - L2	mad(D_level_3, 0)	75.823±2.400
18	L1 - L2	wentropy(p2_3, 'shannon')	75.354±2.409
19	L1 - L2	mean(p2_3)	75.355±2.303
20	L1 - R4	std(pks2_3)	74.886±2.088
21	L1 - L3	range(p2_4)	74.479±1.276
22	L1 - L6	mad(pks2_5, 1)	74.613±1.242
23	L1 - L3	wentropy(p2_3, 'shannon')	74.549±1.561
24	L1 - L2	wentropy(D_level_3, 'shannon')	74.346±1.134
25	L1 - R4	std(D_level_3)	74.412±1.546
26	L1 - L2	std(pks2_3)	75.487±1.995
27	L1 - L3	max(p2_4)	74.412±1.245
28	L1 - L2	mean(pks2_3)	75.089±2.741

Çizelge B.17: (devam) Daubechies6 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

29	L1 - R4	wentropy(p2_3, 'shannon')	74.346±1.171
30	L1 - L3	mad(D_level_3, 0)	74.547±1.786
31	L1 - L2	norm(p2_3)	74.611±2.350
32	L1 - L4	wentropy(D_level_3, 'shannon')	74.478±2.531
33	L1 - L4	max(pks2_4)	74.546±2.537
34	L1 - L2	norm(D_level_3)	73.939±1.596
35	L1 - L4	max(D_level_4)	74.546±2.537
36	L1 - L3	norm(p2_3)	73.944±2.088
37	L1 - L3	mean(pks2_4)	74.213±2.414
38	L1 - L3	mad(pks2_4, 0)	74.145±2.696
39	L1 - L2	mad(pks2_3, 0)	74.074±2.290
40	L1 - L3	range(pks2_4)	73.604±2.090
41	L1 - R4	wentropy(D_level_3, 'shannon')	73.134±1.897
42	L1 - L2	sum(p2_3)	73.604±2.517
43	L1 - L2	norm(p2_3, 1)	73.604±2.517
44	L1 - R1	std(p2_4)	73.536±1.990
45	L1 - L2	std(pks2_4)	73.203±2.430
46	L1 - L6	wentropy(pks2_5, 'shannon')	72.997±1.497
47	L1 - L3	std(pks2_3)	73.272±2.898
48	L1 - R4	mad(p2_3, 0)	73.603±2.459
49	L1 - L2	sum(pks2_3)	73.267±2.343
50	L1 - R1	mad(p2_4, 0)	72.799±2.188
51	L1 - R4	max(p2_3)	73.203±2.621
52	L1 - L3	norm(pks2_3, 1)	73.068±2.751
53	L1 - L3	std(pks2_4)	73.067±2.648
54	L1 - R4	range(p2_3)	73.203±2.621
55	L1 - L3	sum(pks2_3)	73.068±2.751
56	L1 - L2	norm(D_level_3, 1)	72.929±1.783
57	L1 - L3	wentropy(p2_4, 'shannon')	73.198±2.262
58	L1 - L3	std(D_level_3)	73.540±2.678
59	L1 - R4	norm(D_level_3)	72.935±2.827
60	L1 - L3	mad(p2_3, 0)	72.731±2.649
61	L1 - L6	std(pks2_4)	73.267±2.745

Çizelge B.17: (devam) Daubechies6 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

62	L1 - L2	range(D_level_4)	72.592±1.685
63	L1 - R4	norm(D_level_2)	72.528±1.620
64	L1 - L3	min(D_level_4)	72.526±2.450
65	L1 - L3	std(p2_3)	73.068±2.690
66	L1 - L2	std(p2_4)	72.928±2.670
67	L1 - L3	norm(p)	72.125±2.085
68	L1 - L2	std(D_level_4)	71.989±1.822
69	L1 - L3	wentropy(D_level_3, 'shannon')	72.932±2.676
70	L1 - L6	std(pks2_5)	72.729±2.445
71	L5 - R6	wentropy(w, 'shannon')	73.402±2.178
72	L6 - R1	std(D_level_2)	75.826±1.436
73	L6 - R1	norm(p2_2)	74.478±1.694
74	L6 - R1	wentropy(p2_2, 'shannon')	74.209±1.537
75	L6 - R1	std(p2_2)	73.202±2.653
76	L6 - R1	mad(p2_2, 0)	73.668±3.448
77	R1 - RT	std(signals)/mean(signals)	78.789±1.903
78	R1 - R5	mad(p2_4, 0)	74.747±1.759
79	R1 - R6	mad(pks2_5, 1)	72.863±1.737
80	R1 - R3	std(D_level_4)	72.323±1.665
81	R1 - R2	max(p2_4)	72.662±2.404
82	R3 - R6	norm(w)	75.219±0.790
83	R3 - R6	wentropy(w, 'shannon')	72.186±1.850
84	L1 - L4	wentropy(D_level_3, 'shannon')	74.547±1.496
85	L1 - L4	max(D_level_4)	74.951±2.679
86	L1 - L4	max(pks2_4)	74.075±2.055
87	L1 - L2	mad(D_level_3, 0)	74.815±2.236
88	L1 - L6	std(D_level_6)	74.077±1.842
89	L1 - L6	mean(pks)	74.416±2.425
90	L1 - L6	range(D_level_5)	74.944±2.406
91	L1 - L6	std(signals)	73.397±1.564
92	L1 - L6	median(pks)	73.200±1.950
93	L1 - L4	wentropy(p2_3, 'shannon')	72.865±2.349
94	L1 - L4	min(D_level_3)	71.988±1.840

Çizelge B.17: (devam) Daubechies6 dalgacıği kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

95	L1 - L6	std(p)	72.396±2.390
96	L1 - L6	max(D_level_4)	72.931±2.741
97	L1 - R6	range(pks2_4)	74.273±2.187
98	L1 - R6	mad(D_level_4, 0)	73.809±2.483
99	L1 - R7	sum(pks)	73.067±1.904
100	L1 - R2	wentropy(p2_3, 'shannon')	72.190±1.836
101	L1 - R6	max(pks2_4)	71.989±1.582
102	L1 - R6	max(D_level_4)	71.989±1.582
103	L1 - R6	range(p)	71.448±1.281
104	L1 - R6	norm(pks2_3, 1)	71.043±1.017
105	L6 - R5	kurtosis(signals)	73.939±1.418
106	R1 - R6	std(pks2_5)	71.786±1.392
107	R1 - RT	std(signals)/mean(signals)	77.983±1.718
108	R4 - RT	wentropy(w2_1, 'log energy')	72.123±1.655
109	L1 + R2	std(pks2_4)	73.736±1.579
110	L1 + R1	std(p2_4)	74.747±2.871
111	L1 + L8	range(D_level_3)	72.056±1.883
112	L1 + R3	range(pks2_4)	72.057±1.718
113	L4 + L7	kurtosis(signals)	73.670±0.834
114	L4 + L5	kurtosis(signals)	74.141±1.843
115	L4 + L5	skewness(signals)	71.920±1.876
116	L4 + L8	skewness(signals)	73.941±1.728
117	L5 + L8	std(signals)/mean(signals)	72.996±2.857
118	L5 + L8	median(signals)	72.725±1.815
119	L5 + L8	mad(signals, 1)	72.455±1.855
120	R5 + R8	std(signals)/mean(signals)	75.356±2.582
121	R5 + R6	kurtosis(signals)	72.728±2.142
122	L4 + R4	mad(Ed, 1)	77.373±1.915
123	L4 + R4	Ed(4)	76.298±2.278
124	L4 + R4	median(Ed)	74.410±2.121
125	R3 - R7	Ed(5)	74.074±2.325
126	L7 + R5	Ea	74.143±1.999
127	L4 - R4	Ed(4)	73.066±1.223

Çizelge B.17: (devam) Daubechies6 dalgacıđı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

128	L4 - R7	Ed (5)	73.467±1.649
129	R4 + R8	Ed (5)	73.270±2.433
130	L4 + R8	Ed (5)	72.256±1.898



Çizelge B.18: Daubechies7 dalgacıği kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

Öznitelik No	Ayak Sinyali	Sinyale Uygulanan İstatistiksel Fonksiyon	Özniteliğın Tek Başına Sınıflandırma Doğruluđu (%)
1	L1	norm(pks2_3)	72.927±1.763
2	L1	wentropy(p2_3, 'shannon')	72.058±2.032
3	L1 - L3	std(D_level_4)	78.252±1.030
4	L1 - L3	mad(pks2_4, 0)	77.510±0.959
5	L1 - L2	wentropy(pks2_3, 'shannon')	76.698±1.354
6	L1 - L3	std(p2_4)	77.305±2.245
7	L1 - L6	range(D_level_5)	76.768±1.758
8	L1 - L3	std(pks2_4)	76.368±2.139
9	L1 - R4	std(D_level_3)	76.501±2.536
10	L1 - L3	mad(p2_4, 0)	76.229±1.956
11	L1 - R6	std(pks2_2)	75.824±1.994
12	L1 - L3	std(pks2_3)	75.359±1.998
13	L1 - L4	mad(pks2_3, 0)	74.817±1.784
14	L1 - R4	std(pks2_3)	75.623±2.655
15	L1 - L3	min(D_level_4)	75.019±1.891
16	L1 - L2	mad(p2_3, 0)	75.353±2.113
17	L1 - L2	norm(D_level_3)	74.815±2.273
18	L1 - L2	mean(pks2_3)	74.683±2.181
19	L1 - R6	mad(pks2_3, 0)	75.215±2.677
20	L1 - L2	std(p2_3)	74.813±2.022
21	L1 - L3	mad(D_level_4, 0)	75.224±2.882
22	L1 - R6	range(D_level_2)	74.141±1.715
23	L1 - L4	std(pks2_3)	74.277±2.118
24	L1 - R4	wentropy(D_level_3, 'shannon')	74.213±2.269
25	L1 - L2	norm(pks2_3)	74.543±2.114
26	L1 - R6	std(p2_2)	74.343±1.941
27	L1 - L4	std(p2_3)	74.074±2.218
28	L1 - L3	wentropy(pks2_3, 'shannon')	74.213±2.321

Çizelge B.18: (devam) Daubechies7 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

29	L1 - R6	max(pks2_2)	73.874±1.974
30	L1 - R6	max(D_level_2)	73.874±1.974
31	L1 - R6	std(D_level_2)	74.278±2.806
32	L1 - L3	max(p2_4)	74.075±2.118
33	L1 - L2	mad(pks2_3, 0)	73.806±1.931
34	L1 - L3	wentropy(p2_4, 'shannon')	73.608±2.146
35	L1 - L3	range(p2_4)	74.008±2.225
36	L1 - L3	mad(pks2_3, 0)	73.876±2.461
37	L1 - L3	mean(pks2_4)	74.078±2.601
38	L1 - L2	wentropy(D_level_3, 'shannon')	73.540±2.361
39	L1 - L3	norm(pks2_3)	73.470±1.702
40	L1 - R5	std(p2_3)	73.473±2.438
41	L1 - L2	std(D_level_3)	73.602±1.999
42	L1 - L2	mad(D_level_3, 0)	73.604±2.107
43	L1 - L3	wentropy(D_level_3, 'shannon')	73.133±2.260
44	L1 - L2	max(pks2_4)	73.063±1.748
45	L1 - L2	wentropy(p2_3, 'shannon')	73.133±2.324
46	L1 - L2	max(D_level_4)	73.063±1.748
47	L1 - R4	wentropy(pks2_3, 'shannon')	73.673±3.310
48	L1 - L2	std(D_level_4)	73.204±2.384
49	L1 - L3	range(D_level_4)	72.663±1.760
50	L1 - R4	std(p2_3)	73.339±2.899
51	L1 - L2	std(pks2_3)	74.281±3.762
52	L1 - R4	mad(p2_3, 0)	73.405±2.773
53	L1 - L3	wentropy(D_level_4, 'shannon')	73.339±2.867
54	L1 - R6	mad(p2_2, 0)	73.605±3.798
55	L1 - L2	norm(pks2_3, 1)	72.928±2.182
56	L1 - L2	mean(p2_3)	72.728±1.535
57	L1 - L6	range(p2_5)	72.858±2.045
58	L1 - L6	std(pks2_5)	73.129±2.544
59	L1 - L2	std(p2_4)	72.729±2.111
60	L1 - L2	mad(D_level_4, 0)	73.000±2.431
61	L1 - L2	mad(p2_4, 0)	72.661±2.354

Çizelge B.18: (devam) Daubechies7 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

62	L1 - L2	max (pks2_3)	72.526±2.545
63	L1 - L3	mean (p2_4)	72.464±2.457
64	L1 - L3	norm (p)	72.125±2.085
65	L1 - R6	min (D_level_2)	72.730±2.990
66	L1 - L2	max (D_level_3)	72.460±2.428
67	L1 - L2	range (p2_4)	72.327±1.906
68	L1 - L3	wentropy (p, 'shannon')	72.125±2.515
69	L1 - R4	wentropy (p2_3, 'shannon')	72.734±3.557
70	L1 - L6	max (p2_5)	72.320±2.412
71	L1 - L3	norm (p2_4)	72.659±2.502
72	L1 - L2	max (p2_4)	72.125±2.078
73	L1 - L2	range (pks2_4)	71.918±1.824
74	L1 - R5	mad (p2_3, 0)	71.648±1.056
75	L1 - R5	mad (pks2_3, 0)	72.255±1.785
76	L4 - R1	std (D_level_3)	74.342±2.411
77	L5 - R6	wentropy (w, 'shannon')	73.402±2.178
78	L6 - R1	std (pks2_2)	77.304±1.606
79	L6 - R1	norm (p2_2)	74.682±1.688
80	L6 - R1	std (D_level_2)	74.212±1.681
81	L6 - R1	wentropy (D_level_2, 'shannon')	74.341±2.981
82	L6 - R1	norm (pks2_2)	73.872±1.988
83	L6 - R1	wentropy (p2_2, 'shannon')	73.939±2.949
84	L6 - R1	wentropy (pks2_2, 'shannon')	73.399±2.869
85	R1 - RT	std (signals) / mean (signals)	78.789±1.903
86	R1 - R2	max (pks2_4)	74.008±1.519
87	R1 - R2	max (D_level_4)	74.008±1.519
88	R1 - R3	mean (pks2_4)	73.065±1.281
89	R1 - R3	std (pks2_4)	73.740±2.024
90	R1 - R3	mad (pks2_3, 0)	73.131±1.369
91	R1 - R6	range (pks2_5)	73.803±1.954
92	R1 - R2	std (pks2_4)	72.931±1.821
93	R1 - R6	mad (pks2_3, 1)	73.467±2.422
94	R1 - R3	mean (pks2_3)	72.457±1.648

Çizelge B.18: (devam) Daubechies7 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

95	R1 - R5	mad(p2_4, 0)	71.649±0.985
96	R3 - R6	norm(w)	75.219±0.790
97	R3 - R6	wentropy(w, 'shannon')	72.186±1.850
98	L1 - L4	range(pks2_4)	74.948±1.327
99	L1 - L6	mean(pks)	74.416±2.425
100	L1 - L6	range(D_level_6)	73.872±1.528
101	L1 - L6	std(D_level_6)	73.873±2.463
102	L1 - L4	norm(D_level_3)	73.535±2.050
103	L1 - L2	std(p2_4)	74.079±3.071
104	L1 - L6	std(signals)	73.397±1.564
105	L1 - L2	std(pks2_3)	73.134±1.853
106	L1 - L6	mean(pks2_4)	73.264±1.415
107	L1 - L2	std(pks2_4)	73.470±1.672
108	L1 - L6	median(pks)	73.200±1.950
109	L1 - L2	mad(pks2_3, 0)	73.399±2.452
110	L1 - L4	wentropy(p2_3, 'shannon')	73.267±2.412
111	L1 - L4	wentropy(pks2_3, 'shannon')	73.135±2.429
112	L1 - L6	mad(p2_4, 0)	72.592±2.203
113	L1 - L4	std(pks2_3)	72.393±1.955
114	L1 - L5	mad(p2_3, 0)	72.796±2.558
115	L1 - L6	std(p)	72.396±2.390
116	L1 - L6	mad(D_level_5, 0)	72.995±2.422
117	L1 - L6	std(p2_4)	72.588±2.282
118	L1 - R1	mad(p2_3, 1)	72.732±2.300
119	L1 - L4	std(p2_4)	71.984±1.494
120	L1 - R6	range(D_level_4)	74.408±2.329
121	L1 - R6	std(p2_4)	73.944±2.549
122	L1 - R7	sum(pks)	73.067±1.904
123	L1 - R6	mad(p2_4, 0)	72.930±2.225
124	L1 - R6	max(pks2_4)	73.602±3.483
125	L1 - R6	max(D_level_4)	73.602±3.483
126	L1 - R6	range(p)	71.448±1.281
127	L6 - R1	range(p2_4)	72.592±1.898

Çizelge B.18: (devam) Daubechies7 dalgacıđı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıřtır)

128	L6 - R5	kurtosis (signals)	73.939±1.418
129	R1 - R7	median (p2_5)	72.123±1.657
130	R1 - RT	std (signals) /mean (signals)	77.983±1.718
131	R5 - RT	median (w2_1)	72.794±1.959
132	R6 - RT	kurtosis (pks2_5)	72.456±1.341
133	L1 + R1	std (p2_4)	74.280±2.354
134	L1 + R1	kurtosis (p2_3)	75.016±2.595
135	L1 + R4	std (D_level_3)	73.268±2.394
136	L1 + R2	range (pks2_4)	72.997±1.656
137	L1 + R1	mad (p2_4, 0)	72.932±2.132
138	L1 + R5	std (D_level_3)	71.852±1.452
139	L1 + L8	range (pks2_4)	72.862±2.760
140	L4 + L5	kurtosis (signals)	74.141±1.843
141	L4 + L7	kurtosis (signals)	73.670±0.834
142	L4 + L5	skewness (signals)	71.920±1.876
143	L4 + L8	skewness (signals)	73.941±1.728
144	L5 + L8	median (signals)	72.725±1.815
145	L5 + L8	std (signals) /mean (signals)	72.996±2.857
146	L5 + L8	mad (signals, 1)	72.455±1.855
147	R1 + R6	mad (p2_4, 0)	72.527±2.404
148	R5 + R8	std (signals) /mean (signals)	75.356±2.582
149	R5 + R6	kurtosis (signals)	72.728±2.142
150	L4 + R4	Ed (4)	76.703±3.121
151	R4 + R8	Ed (5)	75.223±2.094
152	R3 - R7	kurtosis (Ed)	75.151±1.819
153	L4 + R4	mad (Ed, 1)	75.896±2.756
154	R3 - R7	skewness (Ed)	75.487±2.210
155	L4 + R4	median (Ed)	74.142±1.919
156	L4 - R7	Ed (5)	74.546±2.448
157	R2 - R7	kurtosis (Ed)	75.156±2.977
158	L5 + R4	Ed (4)	74.279±2.711
159	R3 - R6	kurtosis (Ed)	73.400±1.726
160	L5 + R8	Ed (5)	73.535±1.650

Çizelge B.18: (devam) Daubechies7 dalgacıđı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıřtır)

161	L5 + R4	mad(E _d , 1)	73.470±2.400
162	R3 - R6	norm(E _d , 1)	72.933±1.822
163	R2 - R7	Ed(5)	73.267±2.494
164	R3 - R6	mean(E _d)	72.933±1.822
165	R3 - R6	sum(E _d)	72.933±1.822
166	L1 - RT	skewness(E _d)	73.803±3.183
167	L1 - RT	kurtosis(E _d)	72.527±2.036
168	R3 - R6	E _a	72.258±1.983

Çizelge B.19: DMeyer dalgacığı kullanılarak elde edilen özniteliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

Öznitelik No	Ayak Sinyali	Sinyale Uygulanan İstatistiksel Fonksiyon	Özniteliğin Tek Başına Sınıflandırma Doğruluğu (%)
1	L1	mad(pks2_3, 1)	76.233±2.064
2	L1	std(p2_4)	73.401±2.781
3	L1 - L2	std(pks2_3)	77.914±1.418
4	L1 - L2	mean(pks2_3)	78.115±1.35
5	L1 - L2	wentropy(pks2_3, 'shannon')	77.442±1.127
6	L1 - L2	norm(pks2_3)	76.162±1.627
7	L1 - R4	wentropy(D_level_3, 'shannon')	75.894±1.084
8	L1 - R6	std(p2_2)	76.432±2.078
9	L1 - L2	mad(D_level_3, 0)	76.027±1.387
10	L1 - L3	std(p2_4)	75.758±1.377
11	L1 - L2	std(p2_3)	75.760±1.953
12	L1 - L3	mad(pks2_4, 0)	75.624±2.318
13	L1 - L2	sum(pks2_3)	75.289±2.139
14	L1 - L2	norm(pks2_3, 1)	75.557±2.292
15	L1 - R6	std(D_level_2)	75.826±2.968
16	L1 - L2	mad(pks2_3, 0)	74.883±1.108
17	L1 - R4	norm(D_level_3)	75.562±2.958
18	L1 - L2	norm(p2_3)	74.949±1.602
19	L1 - L3	mad(p2_4, 0)	74.819±1.809
20	L1 - R4	mean(p2_3)	75.082±1.930
21	L1 - L3	norm(pks2_3, 1)	74.615±1.915
22	L1 - L2	max(pks2_3)	74.949±2.261
23	L1 - L2	mean(p2_3)	75.489±2.867
24	L1 - L2	wentropy(p2_3, 'shannon')	75.084±2.262
25	L1 - R4	norm(p2_3)	74.820±2.613
26	L1 - L2	mad(D_level_3, 1)	75.353±2.780
27	L1 - L2	max(D_level_3)	74.949±2.261
28	L1 - R4	std(pks2_3)	74.753±2.410

Çizelge B.19: (devam) DMeyer dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

29	L1 - L3	sum(pks2_3)	74.480±1.859
30	L1 - L2	norm(p2_3, 1)	74.544±1.699
31	L1 - L4	wentropy(p2_3, 'shannon')	74.815±2.722
32	L1 - L2	sum(p2_3)	74.544±1.699
33	L1 - R4	mad(p2_3, 0)	74.817±2.247
34	L1 - L2	mad(p2_3, 0)	74.480±1.979
35	L1 - L2	std(pks2_4)	74.548±2.199
36	L1 - L4	std(pks2_3)	74.144±2.352
37	L1 - R4	std(p2_3)	74.281±2.536
38	L1 - L2	mad(pks2_4, 0)	74.078±1.957
39	L1 - L3	wentropy(D_level_4, 'shannon')	74.010±1.839
40	L1 - L3	std(D_level_4)	74.076±1.781
41	L1 - L2	range(D_level_3)	74.010±2.047
42	L1 - L2	mad(p2_3, 1)	73.939±1.392
43	L1 - R2	std(pks2_4)	74.077±2.010
44	L1 - L2	range(pks2_3)	74.343±2.122
45	L1 - L2	wentropy(D_level_3, 'shannon')	74.010±2.281
46	L1 - L2	min(D_level_4)	74.076±2.894
47	L1 - L4	norm(pks2_3)	73.604±1.569
48	L1 - L2	std(D_level_3)	74.006±1.871
49	L1 - L3	std(pks2_3)	73.743±2.138
50	L1 - L2	median(pks2_3)	73.942±2.488
51	L1 - L2	norm(D_level_3)	74.072±2.543
52	L1 - L4	norm(D_level_3)	74.137±2.928
53	L1 - L2	range(p2_4)	74.142±2.333
54	L1 - L2	mad(pks2_3, 1)	73.737±2.069
55	L1 - L4	mad(pks2_3, 0)	73.604±2.645
56	L1 - L3	mad(pks2_3, 1)	72.997±1.728
57	L1 - L3	std(pks2_4)	73.270±2.473
58	L1 - L2	median(p2_3)	73.809±3.193
59	L1 - R4	wentropy(pks2_3, 'shannon')	73.203±2.944
60	L1 - R4	std(D_level_3)	73.137±2.648
61	L1 - L3	wentropy(p2_4, 'shannon')	73.404±2.155

Çizelge B.19: (devam) DMeyer dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

62	L1 - L3	max(p2_4)	73.201±2.377
63	L1 - R4	min(D_level_3)	72.595±1.722
64	L1 - R6	mad(p2_3, 0)	72.728±1.618
65	L1 - L3	median(pks2_3)	73.137±3.009
66	L1 - R6	std(pks2_2)	72.864±2.352
67	L1 - L3	range(p2_4)	72.797±2.073
68	L1 - L4	mad(p2_3, 1)	72.660±1.676
69	L1 - L3	norm(D_level_3)	72.598±2.443
70	L1 - L3	median(p2_3)	72.531±2.050
71	L1 - L2	max(p2_4)	72.860±2.418
72	L1 - R4	mad(pks2_3, 0)	72.392±1.619
73	L1 - L3	norm(p)	72.125±2.085
74	L1 - L2	norm(D_level_4)	72.795±2.271
75	L1 - L4	std(D_level_3)	72.461±2.290
76	L1 - R4	median(p2_3)	72.794±2.448
77	L1 - L3	mad(p2_3, 1)	72.462±2.383
78	L1 - L3	norm(p2_4)	72.734±2.714
79	L1 - L3	mean(pks2_3)	72.460±2.150
80	L1 - L2	norm(pks2_4)	72.525±2.030
81	L1 - L2	wentropy(D_level_4, 'shannon')	72.257±2.234
82	L1 - L3	norm(D_level_3, 1)	71.448±1.270
83	L5 - R6	wentropy(w, 'shannon')	73.402±2.178
84	L6 - R1	mad(pks2_2, 0)	74.143±2.295
85	L6 - R1	norm(p2_2)	73.937±1.844
86	L6 - R1	wentropy(pks2_2, 'shannon')	73.738±1.516
87	L6 - R1	wentropy(p2_2, 'shannon')	73.802±1.591
88	L6 - R1	std(D_level_2)	73.672±2.198
89	L6 - R1	mean(pks2_2)	73.337±2.021
90	L6 - R1	mad(D_level_2, 0)	73.403±2.443
91	L6 - R1	std(pks2_2)	72.868±2.265
92	L6 - R1	std(p2_2)	71.653±1.511
93	R1 - RT	std(signals)/mean(signals)	78.789±1.903
94	R1 - R6	mad(D_level_3, 1)	75.085±2.549

Çizelge B.19: (devam) DMeyer dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

95	R1 - R6	median(p2_3)	74.411±1.994
96	R1 - R3	mad(D_level_3, 0)	72.323±2.163
97	R1 - R3	mean(p2_3)	71.920±1.623
98	R3 - R6	norm(w)	75.219±0.790
99	R3 - R6	wentropy(w, 'shannon')	72.186±1.850
100	L1 - R1	kurtosis(p2_3)	76.632±0.823
101	L1 - L4	wentropy(pks2_3, 'shannon')	74.885±1.766
102	L1 - L6	mean(pks)	74.416±2.425
103	L1 - L4	mad(pks2_3, 0)	73.671±2.229
104	L1 - L4	max(pks2_3)	73.402±2.748
105	L1 - L6	std(signals)	73.397±1.564
106	L1 - L4	max(D_level_3)	73.402±2.748
107	L1 - L4	wentropy(p2_3, 'shannon')	72.998±2.014
108	L1 - L6	median(pks)	73.200±1.950
109	L1 - L6	std(p2_4)	73.398±1.870
110	L1 - L6	std(D_level_4)	73.063±2.456
111	L1 - L6	mad(p2_4, 0)	73.065±2.884
112	L1 - L2	mad(pks2_3, 1)	73.262±2.794
113	L1 - L4	norm(pks2_3)	73.133±2.866
114	L1 - L6	std(p)	72.396±2.390
115	L1 - L4	norm(p2_3)	72.527±1.944
116	L1 - R6	range(D_level_3)	75.960±2.170
117	L1 - RT	kurtosis(p2_3)	73.877±1.594
118	L1 - R6	max(p2_3)	73.742±2.215
119	L1 - R6	range(p2_3)	73.810±2.262
120	L1 - R7	sum(pks)	73.067±1.904
121	L1 - R1	mad(pks2_4, 0)	72.794±2.160
122	L1 - RT	kurtosis(D_level_3)	72.192±1.952
123	L1 - R6	range(p)	71.448±1.281
124	L6 - R5	kurtosis(signals)	73.806±1.438
125	R1 - R6	std(p2_4)	74.479±1.006
126	R1 - R3	wentropy(pks2_2, 'log energy')	73.805±1.844

Çizelge B.19: (devam) DMeyer dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

127	R1 - R6	std(D_level_3)	71.985±1.908
128	R1 - RT	std(signals)/mean(signals)	77.983±1.718
129	L1 + L5	mean(pks2_3)	75.421±1.713
130	L1 + R1	std(pks2_4)	74.341±1.444
131	L1 + R2	mad(p2_4, 0)	74.006±1.767
132	L1 + R2	std(pks2_4)	73.337±2.432
133	L1 + R4	std(pks2_3)	72.662±1.434
134	L1 + R1	std(D_level_4)	72.458±1.482
135	L1 + L6	mad(pks2_3, 0)	72.458±2.439
136	L1 + R1	mad(pks2_4, 0)	72.725±2.379
137	L4 + L5	kurtosis(signals)	74.141±1.843
138	L4 + L7	kurtosis(signals)	73.670±0.834
139	L4 + L5	skewness(signals)	71.92 ±1.876
140	L4 + L8	skewness(signals)	73.941±1.728
141	L5 + L8	std(signals)/mean(signals)	73.130±2.749
142	L5 + L8	median(signals)	72.725±1.815
143	L5 + L8	mad(signals, 1)	72.455±1.855
144	R5 + R8	std(signals)/mean(signals)	75.356±2.582
145	R5 + R6	kurtosis(signals)	72.728±2.142
146	L4 + R4	median(Ed)	72.729±2.332
147	L4 + R4	Ed(4)	73.134±2.869
148	L4 + R5	median(Ed)	72.393±2.163

Çizelge B.20: Haar dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

Öznelik No	Ayak Sinyali	Sinyale Uygulanan İstatistiksel Fonksiyon	Özneliğin Tek Başına Sınıflandırma Doğruluğu (%)
1	L1	max (p2_3)	73.872±1.829
2	L1	range (p2_3)	73.739±1.874
3	L1 - L3	mad (p2_4, 0)	75.892±1.536
4	L1 - L3	std (p2_4)	76.229±2.228
5	L1 - L2	std (D_level_2)	75.487±1.697
6	L1 - L2	wentropy (D_level_3, 'shannon')	75.627±2.183
7	L1 - L2	wentropy (pks2_3, 'shannon')	74.816±1.983
8	L1 - R6	range (D_level_2)	75.086±2.232
9	L1 - L3	wentropy (D_level_2, 'shannon')	74.678±1.847
10	L1 - L2	mad (p2_3, 0)	74.140±1.380
11	L1 - L2	norm (D_level_3)	74.207±1.619
12	L1 - L3	min (D_level_3)	73.268±1.432
13	L1 - L3	norm (D_level_3)	72.998±1.893
14	L1 - L2	max (p2_3)	73.000±1.742
15	L1 - L2	std (p2_4)	72.729±1.314
16	L1 - L2	range (p2_3)	73.000±1.742
17	L1 - L3	min (D_level_4)	73.408±2.642
18	L1 - L3	std (p2_3)	73.069±2.466
19	L1 - L3	std (D_level_2)	73.000±2.323
20	L1 - L2	range (D_level_3)	74.340±2.830
21	L1 - L4	min (D_level_4)	73.271±2.522
22	L1 - L3	wentropy (D_level_3, 'shannon')	72.323±1.103
23	L1 - L3	std (D_level_3)	73.339±2.784
24	L1 - L2	wentropy (D_level_2, 'shannon')	72.326±1.874
25	L1 - L6	range (p2_3)	72.793±2.331
26	L1 - L2	mad (p2_2, 0)	72.190±1.889
27	L1 - L3	norm (p)	72.125±2.085
28	L1 - L6	mean (p2_4)	72.458±2.202

Çizelge B.20: (devam) Haar dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

29	L1 - L6	std(D_level_3)	72.657±2.267
30	L1 - R5	median(pks2_4)	71.922±1.290
31	L1 - L6	mad(D_level_5, 0)	72.055±1.583
32	L1 - L6	max(p2_3)	72.188±2.078
33	L2 - L6	min(pks2_1)	74.141±2.560
34	L2 - L6	mean(pks2_3)	72.122±1.634
35	L2 - L6	median(w2_2)	72.188±1.878
36	L5 - R6	wentropy(w, 'shannon')	73.402±2.178
37	L5 - R1	median(pks2_4)	72.866±2.702
38	R1 - RT	std(signals)/mean(signals)	78.789±1.903
39	R1 - R2	mad(p2_4, 0)	74.480±2.002
40	R1 - R2	std(p2_4)	74.212±2.026
41	R1 - LT	kurtosis(pks2_4)	74.343±2.006
42	R1 - R3	std(D_level_3)	72.996±2.344
43	R3 - R6	norm(w)	75.219±0.790
44	R3 - R6	wentropy(w, 'shannon')	72.186±1.850
45	L1 - L6	mean(pks)	74.416±2.425
46	L1 - L6	mad(D_level_5, 1)	73.739±1.910
47	L1 - L6	range(p2_4)	73.539±2.442
48	L1 - L3	min(D_level_3)	73.402±1.687
49	L1 - L6	std(signals)	73.397±1.564
50	L1 - L6	median(pks)	73.200±1.950
51	L1 - L8	min(D_level_6)	73.264±1.482
52	L1 - L4	min(D_level_4)	73.606±2.849
53	L1 - L6	range(D_level_5)	72.590±1.525
54	L1 - L6	std(p)	72.396±2.390
55	L1 - L6	mean(p2_5)	72.796±2.491
56	L1 - L5	max(p2_3)	72.190±2.064
57	L1 - L5	range(p2_3)	71.987±1.672
58	L1 - L6	range(D_level_6)	71.111±1.071
59	L1 - R6	range(p2_4)	76.366±1.319
60	L1 - R6	max(p2_4)	75.961±1.251
61	L1 - R7	sum(pks)	73.067±1.904

Çizelge B.20: (devam) Haar dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

62	L1 - R6	range (p)	71.448±1.281
63	L4 - R6	std (p2_2) /mean (p2_2)	73.737±3.208
64	L6 - R5	kurtosis (signals)	73.806±1.438
65	R1 - R6	mean (pks2_3)	74.070±2.785
66	R1 - R6	mad (pks2_4, 1)	72.460±1.601
67	R1 - R6	median (pks2_1)	72.120±1.563
68	R1 - RT	std (signals) /mean (signals)	77.983±1.718
69	R2 - R7	mad (w2_4, 1)	73.604±1.409
70	L1 + R5	median (w2_3)	74.949±2.232
71	L1 + R1	mad (p2_4, 0)	73.738±2.683
72	L1 + R2	range (D_level_2)	73.199±2.166
73	L1 + L8	std (pks2_2) /mean (pks2_2)	73.198±2.090
74	L1 + R5	median (pks2_4)	72.456±1.995
75	L1 + R5	mean (w2_4)	72.731±2.276
76	L4 + L7	kurtosis (signals)	73.670±0.834
77	L4 + L5	kurtosis (signals)	74.141±1.843
78	L4 + L5	skewness (signals)	71.920±1.876
79	L4 + L8	skewness (signals)	73.941±1.728
80	L5 + L8	std (signals) /mean (signals)	73.130±2.749
81	L4 + R4	median (w2_2)	72.930±2.439
82	L5 + L8	median (signals)	72.725±1.815
83	L5 + L8	mad (signals, 1)	72.455±1.855
84	L7 + R2	skewness (pks2_4)	71.988±1.626
85	R5 + R8	std (signals) /mean (signals)	75.356±2.582
86	R5 + R6	kurtosis (signals)	72.728±2.142
87	R3 - R6	mean (Ed)	75.892±1.450
88	R3 - R6	sum (Ed)	75.892±1.450
89	R3 - R6	norm (Ed, 1)	75.892±1.450
90	L4 + R4	Ed (2)	75.424±2.143
91	L7 + R5	Ed (5)	76.031±2.147
92	L6 + R5	Ed (4)	76.096±2.935
93	L5 + L7	mad (Ed, 0)	75.222±1.727
94	L4 + R5	median (Ed)	74.954±2.375

Çizelge B.20: (devam) Haar dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

95	L7 - R5	Ed (5)	75.221±2.290
96	L4 + R5	Ed (3)	74.005±1.443
97	L4 + R4	min (Ed)	74.887±3.081
98	R5 + R6	Ed (5)	74.748±2.214
99	L4 + R4	Ed (1)	74.887±3.081
100	L4 + R5	min (Ed)	73.939±1.441
101	L4 + R5	Ed (1)	73.939±1.441
102	L5 + R6	std (Ed)	73.535±1.095
103	L4 + R5	Ed (4)	73.808±2.407
104	L5 + L7	wentropy (Ed, 'shannon')	73.335±1.937
105	R3 - R6	Ea	73.265±2.339
106	L6 - R5	Ed (5)	73.808±2.530
107	L4 + R4	Ed (3)	72.863±1.760
108	L5 + L8	Ea	73.133±2.403
109	L6 - R5	Ea	72.528±2.004
110	L4 - R5	wentropy (Ed, 'log energy')	72.998±2.325
111	L5 + L8	mean (Ed)	72.458±1.795
112	L5 + L8	sum (Ed)	72.458±1.795
113	L5 + L8	norm (Ed, 1)	72.458±1.795
114	L6 + R5	Ed (5)	72.865±2.708

Çizelge B.21: Symlets2 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

Öznelik No	Ayak Sinyali	Sinyale Uygulanan İstatistiksel Fonksiyon	Özneliğin Tek Başına Sınıflandırma Doğruluğu (%)
1	L1 - L3	range (p2_4)	78.587±1.512
2	L1 - L3	max (p2_4)	78.250±1.790
3	L1 - L3	std (D_level_3)	76.431±1.200
4	L1 - L2	mean (p2_3)	76.969±1.979
5	L1 - L3	std (p2_3)	75.492±1.947
6	L1 - L2	std (pks2_3)	76.162±2.383
7	L1 - L2	wentropy (p2_3, 'shannon')	75.557±2.242
8	L1 - L2	range (pks2_4)	74.748±1.268
9	L1 - R2	wentropy (p2_3, 'shannon')	74.815±1.711
10	L1 - L2	norm (p2_3)	74.817±2.063
11	L1 - L2	norm (D_level_3)	74.479±1.335
12	L1 - L3	wentropy (p2_3, 'shannon')	75.560±2.766
13	L1 - L2	norm (pks2_3)	74.213±1.817
14	L1 - L3	range (D_level_4)	74.478±2.102
15	L1 - L3	min (D_level_3)	73.944±1.981
16	L1 - R2	max (p2_3)	74.072±1.962
17	L1 - R2	range (p2_3)	74.072±1.962
18	L1 - R4	max (p2_3)	74.345±1.828
19	L1 - R4	range (p2_3)	74.345±1.828
20	L1 - L2	std (D_level_3)	73.874±2.048
21	L1 - L6	range (D_level_4)	73.805±2.454
22	L1 - L2	std (p2_3)	73.741±2.318
23	L1 - L6	mean (p2_3)	73.469±1.960
24	L1 - L4	mean (w2_1)	73.739±2.405
25	L1 - L3	norm (p2_3)	73.472±1.745
26	L1 - L6	mean (pks2_3)	73.810±3.004
27	L1 - L2	wentropy (pks2_3, 'shannon')	73.132±1.212
28	L1 - L4	min (D_level_3)	73.943±2.329

Çizelge B.21: (devam) Symlets2 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

29	L1 - L6	min(D_level_5)	73.404±1.790
30	L1 - L3	norm(D_level_3)	73.471±2.571
31	L1 - L2	std(p2_4)	74.345±2.859
32	L1 - L3	mad(D_level_3, 0)	74.010±2.920
33	L1 - L6	min(D_level_4)	73.803±2.467
34	L1 - R4	std(D_level_3)	73.471±2.267
35	L1 - L2	mean(pks2_3)	73.133±1.961
36	L1 - L3	std(p2_4)	72.865±2.174
37	L1 - L6	std(D_level_3)	73.333±2.432
38	L1 - L4	wentropy(D_level_3, 'shannon')	73.202±2.360
39	L1 - R4	wentropy(D_level_2, 'shannon')	73.265±1.714
40	L1 - R2	std(p2_4)	72.864±1.698
41	L1 - L8	min(D_level_3)	72.527±1.828
42	L1 - L3	wentropy(p2_2, 'shannon')	72.928±2.182
43	L1 - L3	wentropy(D_level_3, 'shannon')	72.729±1.970
44	L1 - L2	mad(D_level_3, 0)	72.863±2.134
45	L1 - L3	mad(p2_3, 0)	73.273±2.888
46	L1 - R6	mad(pks2_2, 0)	72.260±1.954
47	L1 - L3	norm(p)	72.125±2.085
48	L1 - L2	range(pks2_3)	72.792±2.337
49	L1 - R6	std(pks2_2)	72.794±2.162
50	L1 - L2	range(p2_4)	71.044±0.928
51	L4 - R7	std(w2_4)	75.083±1.686
52	L4 - R7	norm(w2_4)	74.682±2.955
53	L5 - R5	mad(w2_3, 0)	73.470±1.787
54	L4 - R7	mean(w2_4)	73.465±1.926
55	L5 - R6	wentropy(w, 'shannon')	73.402±2.178
56	L4 - R6	std(w2_4)	73.468±2.260
57	L4 - R6	wentropy(w2_4, 'shannon')	72.799±2.323
58	L5 - R5	wentropy(w2_3, 'shannon')	72.799±2.219
59	L6 - R1	std(D_level_2)	73.532±2.727
60	L6 - R1	std(p2_2)	72.929±2.487
61	L8 - R5	mad(w2_3, 0)	74.747±2.507

Çizelge B.21: (devam) Symlets2 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

62	R1 - RT	std(signals)/mean(signals)	78.789±1.903
63	R1 - R2	min(D_level_4)	74.612±1.179
64	R1 - R2	std(p2_4)	72.998±1.826
65	R1 - R6	mean(p2_4)	72.260±2.052
66	R3 - R6	norm(w)	75.219±0.790
67	R3 - R6	sum(locs2_3)	73.270±1.721
68	R3 - R6	norm(locs2_3, 1)	73.270±1.721
69	R3 - R6	wentropy(w, 'shannon')	72.186±1.850
70	L1 - L6	std(p2_4)	75.755±2.086
71	L1 - L6	min(D_level_4)	74.750±1.922
72	L1 - L6	mean(pks)	74.416±2.425
73	L1 - L2	std(p2_3)	74.674±2.454
74	L1 - L6	std(signals)	73.397±1.564
75	L1 - L5	range(D_level_3)	73.332±1.729
76	L1 - L6	mean(pks2_3)	73.132±1.562
77	L1 - L6	mad(p2_3, 0)	74.209±2.659
78	L1 - L6	median(pks)	73.200±1.950
79	L1 - L4	std(p2_3)	73.335±2.417
80	L1 - L6	range(D_level_4)	72.324±1.727
81	L1 - L6	std(p)	72.396±2.39
82	L1 - L5	min(D_level_3)	72.192±1.969
83	L1 - L5	mad(p2_4, 0)	72.120±2.062
84	L1 - L6	std(D_level_6)	71.717±1.645
85	L1 - L6	std(D_level_5)	71.315±0.993
86	L1 - R7	sum(pks)	73.067±1.904
87	L1 - R2	range(D_level_3)	73.398±2.367
88	L1 - R8	min(D_level_3)	72.259±1.556
89	L1 - R6	wentropy(D_level_3, 'shannon')	72.256±2.176
90	L1 - R6	range(p)	71.448±1.281
91	L6 - R1	mad(pks2_4, 0)	72.859±2.621
92	L6 - R1	median(D_level_3)	72.391±2.187
93	L6 - R5	kurtosis(signals)	73.806±1.438
94	L7 - R1	mad(p2_4, 1)	73.874±2.148

Çizelge B.21: (devam) Symlets2 dalgacığı kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

95	R1 - RT	std(signals)/mean(signals)	77.983±1.718
96	L1 + R8	range(p2_3)	72.524±2.247
97	L1 + R8	max(p2_3)	72.658±2.334
98	L4 + L7	kurtosis(signals)	73.670±0.834
99	L4 + L5	kurtosis(signals)	74.141±1.843
100	L4 + L5	skewness(signals)	71.920±1.876
101	L4 + L8	skewness(signals)	73.941±1.728
102	L5 + L8	std(signals)/mean(signals)	73.130±2.749
103	L5 + L8	mad(signals, 1)	72.455±1.855
104	L5 + L8	median(signals)	72.725±1.815
105	L4 + R6	wentropy(w2_4, 'shannon')	72.122±2.031
106	R5 + R8	std(signals)/mean(signals)	75.356±2.582
107	R5 + R6	kurtosis(signals)	72.728±2.142
108	L5 + R5	Ed(4)	77.650±2.400
109	L7 - R5	Ed(5)	76.566±1.335
110	L5 + R5	mad(Ed, 1)	76.769±2.334
111	L4 + R5	Ed(4)	76.636±2.683
112	L4 + R4	mad(Ed, 1)	75.691±2.554
113	L4 + R4	median(Ed)	75.556±2.879
114	L4 + R4	Ed(4)	74.481±2.678
115	L4 + R4	Ed(3)	74.346±2.306
116	L1 - RT	std(Ed)/mean(Ed)	74.079±3.345
117	L4 + R5	mad(Ed, 1)	72.729±1.813
118	L5 + R6	Ed(5)	73.404±2.153
119	L4 + R5	median(Ed)	73.338±2.494
120	L4 + R5	wentropy(Ed, 'log energy')	73.329±2.340
121	L6 + R5	median(Ed)	72.793±1.880
122	L7 + R5	Ed(5)	73.002±2.503
123	L5 + R5	median(Ed)	72.996±1.698
124	L6 + R5	Ed(5)	72.729±2.341
125	R1 - LT	skewness(Ed)	72.324±1.905
126	L7 - R5	Ea	71.920±1.586

Çizelge B.22: HHD kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

Öznelik No	Ayak Sinyali	Sinyale Uygulanan İstatistiksel Fonksiyon	Özneliğin Tek Başına Sınıflandırma Doğruluğu (%)
1	L1	mad(modes_3, 0)	73.737±1.755
2	L1 + L2	std(modes_3)	72.389±1.974
3	L1 + L2	mad(modes_3, 1)	72.256±2.117
4	L1 + R1	mean(p2_2)	72.525±1.312
5	L1 + R1	median(p3_1)	73.462±2.926
6	L1 + R2	mad(p2_2, 0)	76.024±1.821
7	L1 + R2	std(w4_1)	74.477±2.291
8	L1 + R2	range(modes_2)	73.736±2.330
9	L1 + R2	mad(pks2_2, 0)	72.455±2.052
10	L1 + R2	range(modes_1)	72.457±1.615
11	L1 + R2	mean(w4_1)	72.524±2.279
12	L2 + R1	mad(modes_2, 0)	73.271±2.426
13	L2 + R1	mad(pks3_1, 0)	73.200±2.668
14	L2 + R1	std(p2_2)	72.126±1.910
15	L1 - L4	min(modes_1)	73.128±2.590
16	L1 - L4	mean(p3_3)	71.585±1.353
17	L1 - L4	median(instfreq_2)	71.920±1.910
18	L1 - R4	mad(pks3_1, 0)	73.599±2.035
19	L1 - R4	std(pks3_1)	73.264±2.772
20	L1 - R6	mean(w4_1)	72.189±1.659
21	L1 - R7	sum(pks)	72.394±1.856
22	R1 - R7	max(pks)	72.932±2.391
23	R1 - R7	max(signals)	72.932±2.391
24	L1 - L5	mad(p2_3, 0)	76.162±2.272
25	L1 - L5	std(pks2_3)	74.680±2.398
26	L1 - L5	std(p2_3)	74.409±2.918
27	L1 - L5	mad(pks2_3, 0)	74.410±2.582

Çizelge B.22: (devam) HHD kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

28	L1 - R5	mean(p3_1)	73.130±2.333
29	L1 - R5	std(signals)/mean(signals)	76.433±1.950
30	L5	std(signals)/mean(signals)	72.661±2.197
31	L7	kurtosis(w3_2)	73.198±1.283
32	L1 - L6	kurtosis(pks4_2)	73.602±2.627
33	L1 - L6	mean(pks)	74.007±1.970
34	L1 - L6	std(signals)	73.130±1.543
35	L1 - L6	wentropy(p3_2, 'shannon')	72.659±2.343
36	L1 - L6	median(pks3_2)	72.458±1.659
37	L1 - L6	mad(amplitude_2, 1)	73.334±2.852
38	L1 - L6	wentropy(amplitude_2, 'shannon')	72.994±2.933
39	L1 - L6	mean(pxx)	72.997±1.784
40	L1 - L6	median(pks)	73.739±3.438
41	L1 - L7	median(p2_3)	72.592±1.885
42	L1 - R2	mean(modes_1)	74.344±2.336
43	L1 - R2	range(amplitude_2)	72.728±1.310
44	L1 - R2	mad(amplitude_1, 0)	73.064±2.742
45	L1 - R2	max(pks3_2)	72.191±2.033
46	L1 - R2	max(amplitude_2)	72.191±2.033
47	L1 - R8	mad(p3_2, 1)	71.786±1.736
48	L1 - L5	mad(p3_3, 0)	73.333±1.934
49	L1 - L5	mad(amplitude_2, 1)	71.920±1.687
50	R1 - RT	std(signals)/mean(signals)	77.512±2.355
51	R1 - RT	skewness(pks3_2)	72.056±1.775
52	L1 + L6	mad(p3_1, 1)	72.993±2.729
53	L1 + R8	max(amplitude_4)	75.288±2.354
54	L1 + R8	max(pks3_4)	75.288±2.354
55	L1 + R8	range(p3_4)	74.817±2.129
56	L1 + R8	max(p3_4)	74.817±2.129
57	L1 + R8	max(p2_4)	73.942±2.357
58	L1 + R8	range(modes_4)	73.804±2.230
59	L1 + R8	range(pks3_4)	73.739±2.627

Çizelge B.22: (devam) HHD kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

60	L1 + R8	range(amplitude_4)	73.738±2.328
61	L1 + R8	median(w4_1)	73.670±2.330
62	L1 + R8	range(p2_4)	73.065±2.492
63	L1 + R8	std(amplitude_4)	72.457±2.154
64	L1 + R8	mad(amplitude_4, 0)	73.337±2.839
65	L1 + R8	norm(pks2_1)	71.582±1.383
66	L2 - L5	kurtosis(p3_1)	72.321±1.935
67	L3 - L6	wentropy(pks4_4, 'log energy')	72.453±2.215
68	L6 - R5	kurtosis(signals)	74.342±2.264
69	R1 - R3	std(p3_2)	72.321±1.748
70	R1 - R6	mean(w4_2)	74.007±2.551
71	R1 - R6	mean(pks3_2)	72.662±1.533
72	L4 + L5	kurtosis(signals)	74.143±2.524
73	L4 + L7	kurtosis(signals)	73.601±1.060
74	L4 + L8	kurtosis(signals)	72.120±1.904
75	L4 + L8	skewness(signals)	73.739±3.448
76	L4 + L8	std(signals)/mean(signals)	74.608±3.539
77	L4 + L8	median(signals)	72.794±2.051
78	L4 + R6	mad(p4_3, 0)	72.995±2.160
79	L4 + RT	min(signals)	71.988±1.606
80	L6 + R6	mad(w3_3, 1)	75.281±3.173
81	L8 + R1	std(modes_4)	76.161±1.453
82	L8 + R1	mean(pks3_5)	74.480±1.291
83	L8 + R1	mad(amplitude_5, 0)	75.015±1.328
84	L8 + R1	max(modes_4)	73.806±2.228
85	L8 + R1	max(pks2_4)	73.806±2.228
86	L8 + R1	std(p2_4)	73.335±2.750
87	L8 + R1	median(pks3_5)	73.132±2.674
88	L8 + R1	mad(modes_4, 0)	71.849±1.722
89	R4 + R6	skewness(pks3_1)	72.255±1.248
90	R4 + R6	max(pks2_2)	72.192±2.129
91	R4 + R6	max(modes_2)	72.192±2.129
92	R4 + R7	min(pks3_1)	74.614±2.428

Çizelge B.22: (devam) HHD kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

93	R5 + R6	kurtosis(signals)	73.667±2.062
94	R5 + R7	wentropy(w, 'shannon')	72.729±1.687
95	R5 + R7	min(pks3_1)	72.595±1.921
96	R5 + R7	norm(w)	72.322±1.01
97	R5 + R8	std(signals)/mean(signals)	74.003±2.823
98	L1 - L3	norm(p2_2, 1)	73.336±2.630
99	L1 - L3	sum(p2_2)	73.336±2.630
100	L1 - L3	wentropy(p, 'shannon')	72.656±1.907
101	L1 - L3	wentropy(p2_2, 'shannon')	72.933±2.398
102	L1 - L3	wentropy(amplitude_2, 'shannon')	73.340±3.087
103	L1 - L3	mad(p3_1, 0)	72.386±2.358
104	L1 - L6	mad(pks3_1, 1)	73.671±2.725
105	L1 - L6	mean(pxx)	72.997±1.784
106	L1 - L6	median(p3_1)	72.732±2.113
107	L1 - L6	mad(p3_2, 0)	72.598±2.525
108	L1 - L7	range(pks3_3)	72.523±1.127
109	L1 - R2	mad(p3_2, 0)	75.081±1.770
110	L1 - R2	mean(pks3_2)	74.478±2.310
111	L1 - R2	norm(amplitude_2, 1)	72.526±2.149
112	L1 - R2	sum(amplitude_2)	72.526±2.149
113	L1 - R4	std(pks2_1)	72.723±2.253
114	L1 - R4	norm(modes_1)	72.726±1.696
115	L1 - R6	wentropy(pks2_1, 'shannon')	75.889±2.688
116	L1 - R6	wentropy(p2_1, 'shannon')	74.409±1.472
117	L1 - R6	mad(pks2_1, 0)	74.001±2.700
118	L1 - R6	mean(pks2_1)	73.067±2.387
119	L2 - R1	mad(w4_1, 0)	73.265±2.420
120	L2 - R1	mean(w4_1)	72.054±2.004
121	L4 - R4	std(modes_2)/mean(modes_2)	73.938±3.148
122	L5 - R6	wentropy(w, 'shannon')	73.401±2.506
123	L6 - R7	median(p3_3)	73.804±0.902
124	R1 - R2	mad(p2_2, 0)	75.963±2.294
125	R1 - R2	std(pks2_2)	73.199±2.077

Çizelge B.22: (devam) HHD kullanılarak elde edilen özneliklerin listesi (10-katlı çapraz doğrulama kullanılmıştır)

126	R1 - R2	mad(amplitude_2, 1)	73.329±2.643
127	R1 - R2	mad(pks2_2, 0)	72.526±1.958
128	R1 - R2	std(p2_2)	72.862±2.438
129	R1 - R2	mad(pks3_2, 0)	72.324±1.570
130	R1 - R3	mad(amplitude_2, 0)	74.005±2.048
131	R1 - R3	std(amplitude_2)	74.006±2.074
132	R1 - R3	mad(pks3_2, 0)	72.458±1.317
133	R1 - R6	mad(pks2_1, 1)	73.402±1.548
134	R1 - R6	median(p3_2)	72.053±1.950
135	R3 - R6	norm(w)	75.153±1.693

EK C

Çizelge C.1: Baseline algoritmaların benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Veri kümesi	Baseline	
	Rasgele Tahmin Edici	ZeroR
Banknote Authentication	50.35	55.54
Blood Transfusion	49.63	76.20
Boston Housing2	1.34	5.93
Breast Cancer	49.04	65.52
Breast Tissue	17.36	18.87
Cardiotocography-1	33.57	77.85
Cardiotocography-2	10.00	27.23
Climate Model	49.52	91.48
DNA	33.30	51.91
Glass	14.21	35.51
HTRU2	50.32	90.84
Ionosphere	50.46	64.10
Leaf	2.62	4.71
Letter Recognition	3.90	4.07
Libras Movement	7.31	5.56
LSVT Voice Rehabilitation	48.33	66.67
Madelon	49.62	50.00
MAGIC Gamma Telescope	50.14	64.84
Parkinson Speech	50.09	50.00
QSAR biodegradation	49.55	66.26
Satellite	16.54	23.82
Seeds	34.38	33.33
Shuttle	14.25	78.60

Çizelge C.1: (devam) Baseline algoritmaların benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Sonar	50.00	53.37
Vehicle	25.45	25.77
Vertebral Column	49.13	67.74
Vowel	8.56	9.09
Wall Following Robot Navigation-1	24.81	40.41
Wall Following Robot Navigation-2	24.60	40.41
Wall Following Robot Navigation-3	24.85	40.41
Wine Quality-red	16.30	42.59
Wine Quality-white	14.32	44.88
Zoo	14.95	40.59
Ortalama	29.96	45.88
Yerel Rank	3	30

Çizelge C.2: Bayesian öğrenme algoritmalarının benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Veri kümesi	Bayesian öğrenme					
	A1DE	A2DE	HMM	Bayesian Logistic Regression	Bayesian Network	Naive Bayes
Banknote Authentication	95.04	96.21	55.54	96.94	92.13	84.26
Blood Transfusion	74.73	73.93	23.80	76.20	74.73	74.87
Boston Housing2	95.45	–	1.38	–	93.28	89.33
Breast Cancer	96.57	96.57	–	–	97.14	96.14
Breast Tissue	65.09	63.21	19.81	–	66.04	70.75
Cardiotocography-1	91.35	92.43	77.85	–	86.27	82.36
Cardiotocography-2	81.51	84.34	18.06	–	74.04	71.54
Climate Model	92.22	92.78	8.52	91.48	92.22	95.00
DNA	95.29	–	24.07	–	94.92	94.48
Glass	76.17	75.70	32.71	–	74.77	49.53
HTRU2	97.42	97.95	90.84	96.51	96.87	94.51
Ionosphere	93.45	94.30	35.90	86.32	89.46	82.34
Leaf	66.47	64.71	3.53	–	65.00	74.12
Letter Recognition	91.30	94.31	3.95	–	74.36	64.08
Libras Movement	75.28	–	6.67	–	61.39	62.78
LSVT Voice Rehabilitation	82.54	–	33.33	66.67	73.81	54.76
Madelon	62.00	–	50.00	59.95	61.35	58.40
MAGIC Gamma Telescope	82.52	83.94	64.84	75.11	77.77	72.69
Parkinson Speech	59.62	61.25	50.00	58.65	59.90	59.04
QSAR biodegradation	82.75	83.60	33.74	82.84	80.66	75.92
Satellite	89.15	–	23.82	–	82.24	79.60
Seeds	89.05	89.52	33.33	–	88.10	91.43
Shuttle	99.91	99.94	78.60	–	99.44	92.93
Sonar	81.25	80.29	53.37	75.00	80.29	67.79
Vehicle	71.16	71.16	25.77	–	60.99	45.27
Vertebral Column	80.32	80.00	67.74	80.00	76.45	77.74

Çizelge C.2: (devam) Bayesian öğrenme algoritmalarının benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Vowel	59.70	63.94	9.09	–	51.62	56.57
Wall Following Robot Navigation-1	99.84	99.84	40.41	–	99.91	90.58
Wall Following Robot Navigation-2	99.82	99.40	40.41	–	99.89	89.11
Wall Following Robot Navigation-3	95.27	–	40.41	–	88.23	52.46
Wine Quality-red	57.72	57.60	0.63	–	57.79	54.72
Wine Quality-white	52.65	54.94	0.41	–	48.12	44.30
Zoo	97.03	97.03	40.59	–	97.03	97.03
Ortalama	82.72	82.65	34.04	78.81	79.28	74.13
Yerel Rank	12	13	0	2	5	5

Çizelge C.3: Fonksiyon öğrenme algoritmalarının benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Veri kümesi	Fonksiyon öğrenme											
	LibSVM	Logistic	Multilayer Perceptron	RBF Network	SMO	SPegasos	Voted Perceptron	SGD	FLDA	Kernel Logistic Regression	LDA	QDA
Banknote Authentication	98.03	99.13	99.93	95.12	98.03	98.54	97.59	98.47	97.59	98.91	97.67	98.40
Blood Transfusion	76.20	77.14	78.21	77.41	76.20	76.60	76.20	75.94	65.91	77.14	76.74	77.94
Boston Housing2	49.60	83.99	84.39	–	50.99	–	–	–	–	–	96.44	82.02
Breast Cancer	96.71	96.71	94.99	95.99	96.71	96.42	90.84	96.57	–	96.71	–	–
Breast Tissue	60.38	72.64	64.15	66.98	60.38	–	–	–	–	–	65.09	69.81
Cardiotocography-1	89.37	89.28	91.77	85.65	89.37	–	–	–	–	–	87.30	83.44
Cardiotocography-2	78.69	83.07	82.97	72.91	78.79	–	–	–	–	–	74.27	69.38
Climate Model	95.37	95.37	93.89	94.26	95.37	96.30	92.03	96.48	83.15	95.37	95.56	92.96
DNA	93.41	90.49	94.57	94.57	93.47	–	–	–	–	–	94.85	90.43
Glass	57.48	66.36	67.29	65.42	57.48	–	–	–	–	–	65.42	57.94
HTRU2	97.56	97.90	98.07	96.88	97.56	97.52	97.18	97.88	97.80	–	97.46	96.73
Ionosphere	87.46	89.17	90.31	91.45	87.46	85.76	85.47	86.04	85.76	88.89	86.32	89.17
Leaf	54.71	76.47	78.82	69.71	55.29	–	–	–	–	–	81.18	64.71

Çizelge C.3: (devam) Fonksiyon öğrenme algoritmalarının benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Letter Recognition	82.35	77.41	82.50	???	82.22	–	–	–	–	–	70.19	88.65
Libras Movement	74.72	68.33	80.56	76.67	74.72	–	–	–	–	–	67.50	86.11
LSVT Voice Rehabilitation	84.13	68.25	84.92	65.08	84.13	82.54	66.67	83.33	56.35	81.75	–	–
Madelon	55.10	53.90	???	63.95	54.95	53.25	56.95	53.25	53.75	53.95	53.75	52.00
MAGIC Gamma Telescope	79.15	79.11	85.87	78.43	79.15	79.06	74.02	79.15	79.47	–	78.41	78.44
Parkinson Speech	63.27	63.46	66.25	59.04	63.17	63.46	55.67	64.62	62.69	63.37	62.12	56.25
QSAR biodegradation	85.50	85.97	87.11	77.44	85.59	86.54	77.63	86.82	84.83	85.97	85.97	72.23
Satellite	86.64	86.03	89.20	83.84	86.68	–	–	–	–	–	84.09	85.73
Seeds	93.81	95.24	95.24	92.38	93.81	–	–	–	–	–	96.67	95.24
Shuttle	96.94	96.82	99.71	98.03	96.93	–	–	–	–	–	94.39	83.20
Sonar	75.96	73.56	80.77	72.12	75.96	75.00	75.96	79.33	75.00	75.96	74.52	77.40
Vehicle	74.47	80.61	81.91	65.60	74.47	–	–	–	–	–	78.13	85.34
Vertebral Column	78.71	85.48	84.52	80.32	78.71	85.48	79.03	84.19	80.32	85.48	84.19	82.90
Vowel	58.08	58.89	75.35	???	57.98	–	–	–	–	–	54.14	79.29
Wall Following Robot Navigation-1	77.25	99.67	92.63	98.68	77.20	–	–	–	–	–	70.77	90.60

Çizelge C.3: (devam) Fonksiyon öğrenme algoritmalarının benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Wall Following Robot Navigation-2	77.31	99.80	96.24	92.69	77.27	–	–	–	–	–	71.32	88.58
Wall Following Robot Navigation-3	71.46	70.49	87.90	66.81	71.50	–	–	–	–	–	65.74	68.07
Wine Quality-red	58.22	59.79	61.48	54.72	58.22	–	–	–	–	–	59.54	54.66
Wine Quality-white	52.12	53.55	54.61	51.96	52.12	–	–	–	–	–	53.14	48.33
Zoo	96.04	94.06	96.04	95.05	97.03	–	–	–	–	–	96.04	96.04
Ortalama	77.46	80.85	84.44	79.31	77.54	82.81	78.86	83.24	76.89	82.14	78.03	78.77
Yerel Rank	1	6	15	2	2	1	0	1	0	2	4	4

Çizelge C.4: Örnek tabanlı öğrenme algoritmalarının benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Veri kümesi	Örnek tabanlı öğrenme			
	IBk	KStar	LWL	RseslibKnn
Banknote Authentication	99.85	99.85	86.01	100.00
Blood Transfusion	69.92	76.34	76.20	73.93
Boston Housing2	82.81	87.55	19.76	94.47
Breast Cancer	95.42	95.57	91.85	96.42
Breast Tissue	71.70	65.09	43.40	65.09
Cardiotocography-1	91.16	91.67	85.84	92.14
Cardiotocography-2	78.27	80.53	45.86	82.36
Climate Model	89.07	90.00	91.48	93.15
DNA	74.64	76.55	63.59	90.99
Glass	70.56	75.70	44.86	77.57
HTRU2	97.13	???	97.73	97.93
Ionosphere	87.18	84.05	82.91	92.02
Leaf	73.53	70.59	40.00	73.24
Letter Recognition	95.97	95.87	41.33	97.20
Libras Movement	85.83	84.44	44.44	83.61
LSVT Voice Rehabilitation	73.81	33.33	69.05	77.78
Madelon	54.15	50.00	61.85	72.35
MAGIC Gamma Telescope	80.94	82.71	72.53	85.32
Parkinson Speech	63.65	65.96	55.87	69.90
QSAR biodegradation	84.46	82.27	72.13	85.40
Satellite	90.26	90.71	75.63	91.14
Seeds	94.29	92.86	91.90	93.33
Shuttle	99.94	99.90	86.89	99.95
Sonar	86.54	84.62	73.56	85.10
Vehicle	71.28	70.92	47.16	70.80
Vertebral Column	81.61	83.23	77.42	81.94

Çizelge C.4: (devam) Örnek tabanlı öğrenme algoritmalarının benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Vowel	98.38	97.88	37.07	98.08
Wall Following Robot Navigation-1	98.83	96.63	78.85	99.01
Wall Following Robot Navigation-2	97.21	97.75	78.85	97.69
Wall Following Robot Navigation-3	88.18	94.48	69.96	94.06
Wine Quality-red	63.98	65.79	55.22	70.36
Wine Quality-white	65.84	65.97	45.71	70.01
Zoo	96.04	96.04	89.11	95.05
Ortalama	83.41	82.03	66.49	86.28
Yerel Rank	8	5	0	21

Çizelge C.5: Farklı öğrenme yaklaşımları kullanan algoritmaların benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Veri kümesi	HyperPipes	VFI	Input Mapped Classifier	CHIRP	FLR
Banknote Authentication	68.51	78.50	98.54	99.49	74.34
Blood Transfusion	24.47	36.76	76.87	77.01	65.78
Boston Housing2	86.76	92.69	89.53	76.48	88.93
Breast Cancer	88.13	95.57	94.13	97.14	74.39
Breast Tissue	65.09	66.98	66.04	66.98	66.98
Cardiotocography-1	82.69	75.21	93.23	90.40	75.59
Cardiotocography-2	45.01	52.87	83.21	78.36	46.61
Climate Model	64.63	58.70	92.59	93.15	87.04
DNA	32.83	70.59	92.12	93.97	42.31
Glass	50.47	57.01	65.89	65.42	47.20
HTRU2	93.32	93.31	97.91	97.82	86.53
Ionosphere	35.90	94.87	88.89	90.03	75.50
Leaf	63.24	43.24	61.18	71.76	72.65
Letter Recognition	22.27	61.24	88.12	85.20	22.33
Libras Movement	58.61	58.61	69.72	71.39	60.83
LSVT Voice Rehabilitation	79.37	62.70	73.81	84.92	71.43
Madelon	50.00	47.80	70.35	62.90	49.90
MAGIC Gamma Telescope	66.45	66.49	85.06	85.69	57.13
Parkinson Speech	52.02	55.19	62.98	66.92	52.40
QSAR biodegradation	58.20	61.42	82.37	85.78	63.41
Satellite	47.85	70.40	86.62	88.21	55.40
Seeds	80.48	87.62	91.90	90.48	84.29
Shuttle	84.32	79.98	99.97	99.73	87.41
Sonar	60.58	57.69	71.15	75.00	60.10
Vehicle	37.71	53.07	49.41	69.74	35.22

Çizelge C.5: (devam) Farklı öğrenme yaklaşımları kullanan algoritmaların benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Vertebral Column	67.42	72.90	81.61	84.52	68.71
Vowel	28.38	50.81	78.99	85.35	39.70
Wall Following Robot Navigation-1	99.87	99.98	100.00	61.35	99.95
Wall Following Robot Navigation-2	99.73	96.85	100.00	97.91	99.82
Wall Following Robot Navigation-3	93.33	92.19	99.65	89.39	68.15
Wine Quality-red	10.69	35.71	60.73	65.67	34.58
Wine Quality-white	4.96	27.79	58.88	64.29	37.08
Zoo	95.05	90.10	92.08	94.06	92.08
Ortalama	60.56	68.03	81.93	82.02	64.96
Yerel Rank	1	3	12	17	2

Çizelge C.6: Kural öğrenme algoritmalarının benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Veri kümesi	Kural öğrenme												
	Conjunctive Rule	Decision Table	DTNB	JRip	NNge	OneR	PART	Ridor	FURIA	MODLEM	OLM	RoughSet	Vfdr
Banknote Authentication	86.01	95.92	95.85	98.18	98.32	83.60	98.98	98.40	99.20	97.01	54.96	98.83	86.52
Blood Transfusion	76.20	74.87	74.87	78.61	69.65	75.00	77.01	76.60	77.41	74.33	56.28	67.51	66.18
Boston Housing2	10.47	90.51	93.87	71.15	91.70	61.66	88.74	81.03	83.60	–	1.78	93.28	25.10
Breast Cancer	92.56	94.71	95.71	94.56	96.85	91.99	94.85	95.57	95.14	95.99	84.98	–	95.99
Breast Tissue	40.57	59.43	61.32	63.21	67.92	54.72	67.92	66.04	63.21	63.21	32.08	66.04	66.98
Cardiotocography-1	77.85	90.59	87.44	93.46	92.05	82.08	93.13	92.71	93.74	91.02	79.21	89.75	79.35
Cardiotocography-2	42.90	69.24	78.32	82.60	81.28	47.60	83.44	81.00	83.49	82.64	22.06	78.41	49.76
Climate Model	91.48	92.22	93.52	93.70	92.04	89.81	92.96	93.52	94.63	91.85	8.52	91.85	85.93
DNA	62.34	90.93	???	91.81	88.76	63.28	92.72	92.37	94.38	92.91	29.28	+++	77.06
Glass	44.86	70.09	74.30	66.82	66.82	57.48	67.76	68.69	74.77	70.56	32.71	71.50	46.26
HTRU2	97.70	97.79	97.77	97.85	97.78	97.58	97.83	97.68	97.87	97.46	90.90	97.60	94.66
Ionosphere	81.48	85.19	89.17	90.60	90.88	79.49	90.03	87.18	91.45	91.74	51.28	84.62	86.89
Leaf	7.35	47.94	62.94	53.82	70.59	23.82	64.71	63.24	65.00	57.35	5.88	64.12	52.35

Çizelge C.6: (devam) Kural öğrenme algoritmalarının benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Letter Recognition	6.96	65.08	76.20	86.24	91.64	17.22	88.62	???	+++	89.17	40.62	+++	37.87
Libras Movement	10.83	42.50	58.89	52.22	77.78	21.11	70.56	56.94	64.44	60.83	27.50	+++	55.00
LSVT Voice Rehabilitation	69.05	69.84	75.40	79.37	80.95	75.40	73.81	81.75	83.33	76.98	33.33	80.16	54.76
Madelon	58.95	69.25	???	74.90	51.70	53.60	59.50	69.55	72.75	53.00	50.00	+++	54.45
MAGIC Gamma Telescope	73.19	82.43	82.23	84.77	82.19	70.52	85.46	82.73	84.69	+++	70.07	77.41	72.43
Parkinson Speech	54.90	58.46	60.48	63.94	61.54	56.63	59.71	61.83	65.10	63.94	50.00	68.17	55.67
QSAR biodegradation	70.05	80.85	81.52	82.37	82.75	77.16	85.21	82.56	84.36	81.61	36.02	83.32	68.34
Satellite	43.64	82.10	82.42	86.88	87.86	59.98	87.13	85.98	89.00	87.23	39.33	+++	64.43
Seeds	64.76	86.19	88.10	90.95	90.95	82.38	92.86	90.95	93.81	91.43	55.71	92.38	90.95
Shuttle	86.86	99.74	99.74	99.97	99.96	94.69	99.97	99.98	99.98	99.97	98.32	99.97	95.55
Sonar	65.87	75.00	75.96	76.92	72.12	63.46	80.29	73.56	78.85	70.67	53.37	72.60	68.75
Vehicle	41.49	65.84	64.30	66.67	62.41	52.25	73.40	69.86	71.04	72.58	26.12	71.51	46.34
Vertebral Column	77.42	80.65	80.97	82.26	80.32	73.87	80.97	83.87	84.52	82.26	57.42	80.00	78.06
Vowel	17.68	51.72	53.74	66.67	83.94	30.81	73.94	71.21	77.78	66.57	9.90	83.84	39.39
Wall Following Robot Navigation-1	78.83	99.65	99.87	99.95	99.95	81.96	99.98	99.96	100.00	99.98	38.98	100.00	—
Wall Following Robot Navigation-2	78.83	99.65	99.87	99.95	99.85	81.96	99.98	99.96	100.00	99.98	42.96	100.00	83.60

Çizelge C.6: (devam) Kural öğrenme algoritmalarının benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Wall Following Robot Navigation-3	69.89	91.11	96.19	98.79	94.01	75.64	99.78	99.45	99.65	97.86	40.49	97.62	59.20
Wine Quality-red	55.91	57.35	57.66	57.97	64.67	54.91	59.10	56.29	59.41	66.79	27.96	67.23	49.59
Wine Quality-white	49.06	52.29	49.82	54.80	62.66	46.10	59.35	52.98	53.29	65.56	27.11	66.35	41.45
Zoo	59.41	87.13	93.07	90.10	95.05	73.27	93.07	90.10	89.11	94.06	85.15	93.07	96.04
Ortalama	58.95	77.46	80.05	80.97	82.63	65.18	82.81	81.36	83.28	81.50	44.25	82.86	66.40
Yerel Rank	0	0	1	2	6	0	6	2	13	1	0	5	1

Çizelge C.7: Karar ağacı algoritmalarının benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Veri kümesi	Karar ağacı öğrenme														
	ADTree	BFTree	Decision Stump	FT	J48	LADTree	LMT	NBTree	Random Forest	Random Tree	REPTree	Simple Cart	Hoefding Tree	CDT	ExtraTree
Banknote Authentication	97.96	98.40	84.33	99.93	98.54	97.38	99.93	98.40	99.27	98.54	98.03	98.25	94.46	97.81	98.32
Blood Transfusion	76.74	78.61	76.20	78.74	76.87	78.34	77.94	74.06	73.93	70.32	79.28	77.27	74.60	78.21	69.52
Boston Housing2	–	84.98	10.28	79.84	89.53	72.92	90.91	95.45	95.45	90.32	84.78	84.39	89.33	76.28	85.38
Breast Cancer	95.99	94.99	92.42	96.71	94.13	95.71	96.57	96.85	96.57	94.56	94.42	94.99	95.99	94.56	–
Breast Tissue	–	68.87	40.57	70.75	66.04	60.38	68.87	60.38	71.70	62.26	72.64	68.87	66.98	69.81	61.32
Cardiotocography-1	–	92.57	77.85	93.23	93.23	92.00	94.03	92.57	94.78	91.91	92.99	92.85	80.10	92.66	89.65
Cardiotocography-2	–	83.02	43.09	83.30	83.21	79.92	84.62	79.59	88.48	79.02	82.31	82.83	50.71	81.51	74.08
Climate Model	94.44	91.30	91.48	96.11	92.59	93.70	96.11	94.63	93.70	91.11	91.48	91.85	94.81	91.11	90.00
DNA	–	92.40	62.34	95.01	92.12	92.81	96.23	92.53	94.82	76.02	92.81	92.84	94.22	92.75	82.23
Glass	–	71.50	44.86	64.95	65.89	67.29	68.22	69.16	80.37	68.69	62.15	70.56	46.26	63.08	62.15
HTRU2	97.71	97.86	97.67	97.94	97.91	97.88	97.98	96.87	98.08	96.91	97.78	97.92	97.48	97.80	96.49
Ionosphere	91.74	89.17	82.62	89.17	88.89	89.74	93.73	89.17	93.16	86.04	88.89	88.32	88.60	89.17	88.60
Leaf	–	62.06	8.24	78.24	61.18	54.41	81.18	64.12	76.76	57.94	58.24	63.53	70.29	57.94	57.06

Çizelge C.7: (devam) Karar ağacı algoritmalarının benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Letter Recognition	–	86.36	7.09	89.86	88.12	51.33	92.14	86.84	96.42	86.03	84.26	86.97	64.25	83.77	82.29
Libras Movement	–	62.78	11.11	78.33	69.72	53.06	76.94	73.06	82.50	67.22	56.67	66.67	63.06	55.28	63.89
LSVT Voice Rehabilitation	77.78	80.16	70.63	82.54	73.81	78.57	85.71	71.43	83.33	70.63	80.95	82.54	66.67	80.95	75.40
Madelon	71.95	76.20	61.85	57.55	70.35	72.65	71.30	73.80	64.55	54.20	73.40	79.65	58.15	76.00	54.00
MAGIC Gamma Telescope	82.46	84.39	71.60	86.14	85.06	82.97	86.17	85.37	88.04	82.13	85.01	85.03	81.86	85.05	80.04
Parkinson Speech	63.17	63.85	53.56	62.98	62.98	63.17	65.10	64.71	69.62	60.87	61.63	64.81	58.56	62.31	57.21
QSAR biodegradation	81.33	81.71	70.71	85.78	82.37	83.13	85.88	83.32	85.97	82.46	81.90	81.14	72.89	83.13	81.14
Satellite	–	86.68	43.71	87.07	86.62	82.75	87.65	82.18	91.66	84.82	86.25	87.04	79.58	85.78	83.85
Seeds	–	93.33	65.24	95.24	91.90	91.90	95.24	90.95	94.29	90.00	90.00	90.48	90.95	89.52	89.05
Shuttle	–	99.97	86.89	99.92	99.97	99.86	99.94	99.98	99.99	99.97	99.95	99.97	97.53	99.94	99.94
Sonar	77.88	71.15	73.08	80.77	71.15	78.85	77.88	75.48	85.10	73.56	75.00	71.15	68.75	74.04	67.31
Vehicle	–	69.62	39.83	79.91	74.35	67.85	82.27	67.49	76.00	69.86	73.17	70.92	45.86	69.86	65.72
Vertebral Column	82.58	81.61	77.42	85.48	81.61	79.35	85.48	81.29	83.55	79.03	80.32	82.90	77.74	83.23	76.45
Vowel	–	75.15	17.68	79.29	78.99	46.46	87.07	76.87	95.66	77.88	67.37	77.07	56.67	68.08	74.65
Wall Following Robot Navigation-1	–	100.00	78.85	99.98	100.00	100.00	99.98	99.65	100.00	100.00	100.00	100.00	98.92	100.00	97.49

Çizelge C.7: (devam) Karar ağacı algoritmalarının benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Wall Following Robot Navigation-2	–	100.00	78.85	99.96	100.00	100.00	99.96	99.91	100.00	99.96	100.00	100.00	98.37	100.00	97.64	
Wall Following Robot Navigation-3	–	99.29	69.89	98.57	99.65	98.81	98.09	98.50	99.67	97.73	99.58	99.41	52.47	99.58	92.36	
Wine Quality-red	–	59.98	55.03	58.79	60.73	58.79	60.54	59.16	70.86	63.48	57.79	60.79	54.22	58.16	61.16	
Wine Quality-white	–	57.82	44.88	55.84	58.88	51.67	61.15	55.59	70.36	62.84	55.80	58.37	44.88	54.53	61.92	
Zoo	–	92.08	59.41	95.05	92.08	95.05	93.07	95.05	97.03	94.06	89.11	92.08	96.04	89.11	95.05	
Ortalama		83.98	82.66	59.07	84.33	82.68	79.05	86.00	82.56	87.63	80.62	81.64	83.07	74.89	81.24	78.48
Yerel Rank		0	2	0	4	2	2	9	2	20	1	4	3	0	2	0

Çizelge C.8: Farklı öğrenme yaklaşımlarını benimseyen ve benchmark veri kümeleri üzerinde en iyi öğrenme sağlayan temel öğrencilerin global ölçekte karşılaştırma sonuçları (birinci grup)

Veri kümesi	ZeroR	AIDE	Multilayer Perceptron	RseslibKnn	CHIRP	FURIA	Random Forest	REPTree	LDA	Bayesian Network	Logistic	SGD	LMT	VFI	Naive Bayes	A2DE
Banknote Authentication	55.54	95.04	99.93	100.00	99.49	99.20	99.27	98.03	97.67	92.13	99.13	98.47	99.93	78.50	84.26	96.21
Blood Transfusion	76.20	74.73	78.21	73.93	77.01	77.41	73.93	79.28	76.74	74.73	77.14	75.94	77.94	36.76	74.87	73.93
Boston Housing2	5.93	95.45	84.39	94.47	76.48	83.60	95.45	84.78	96.44	93.28	83.99	–	90.91	92.69	89.33	–
Breast Cancer	65.52	96.57	94.99	96.42	97.14	95.14	96.57	94.42	–	97.14	96.71	96.57	96.57	95.57	96.14	96.57
Breast Tissue	18.87	65.09	64.15	65.09	66.98	63.21	71.70	72.64	65.09	66.04	72.64	–	68.87	66.98	70.75	63.21
Cardiotocography-1	77.85	91.35	91.77	92.14	90.40	93.74	94.78	92.99	87.30	86.27	89.28	–	94.03	75.21	82.36	92.43
Cardiotocography-2	27.23	81.51	82.97	82.36	78.36	83.49	88.48	82.31	74.27	74.04	83.07	–	84.62	52.87	71.54	84.34
Climate Model	91.48	92.22	93.89	93.15	93.15	94.63	93.70	91.48	95.56	92.22	95.37	96.48	96.11	58.70	95.00	92.78

Çizelge C.8: (devam) Farklı öğrenme yaklaşımlarını benimseyen ve benchmark veri kümeleri üzerinde en iyi öğrenme sağlayan temel öğrencilerin global ölçekte karşılaştırma sonuçları (birinci grup)

DNA	51.91	95.29	94.57	90.99	93.97	94.38	94.82	92.81	94.85	94.92	90.49	–	96.23	70.59	94.48	–
Glass	35.51	76.17	67.29	77.57	65.42	74.77	80.37	62.15	65.42	74.77	66.36	–	68.22	57.01	49.53	75.70
HTRU2	90.84	97.42	98.07	97.93	97.82	97.87	98.08	97.78	97.46	96.87	97.90	97.88	97.98	93.31	94.51	97.95
Ionosphere	64.10	93.45	90.31	92.02	90.03	91.45	93.16	88.89	86.32	89.46	89.17	86.04	93.73	94.87	82.34	94.30
Leaf	4.71	66.47	78.82	73.24	71.76	65.00	76.76	58.24	81.18	65.00	76.47	–	81.18	43.24	74.12	64.71
Letter Recognition	4.07	91.30	82.50	97.20	85.20	+++	96.42	84.26	70.19	74.36	77.41	–	92.14	61.24	64.08	94.31
Libras Movement	5.56	75.28	80.56	83.61	71.39	64.44	82.50	56.67	67.50	61.39	68.33	–	76.94	58.61	62.78	–
LSVT Voice Rehabilitation	66.67	82.54	84.92	77.78	84.92	83.33	83.33	80.95	–	73.81	68.25	83.33	85.71	62.70	54.76	–
Madelon	50.00	62.00	???	72.35	62.90	72.75	64.55	73.40	53.75	61.35	53.90	53.25	71.30	47.80	58.40	–
MAGIC Gamma Telescope	64.84	82.52	85.87	85.32	85.69	84.69	88.04	85.01	78.41	77.77	79.11	79.15	86.17	66.49	72.69	83.94
Parkinson Speech	50.00	59.62	66.25	69.90	66.92	65.10	69.62	61.63	62.12	59.90	63.46	64.62	65.10	55.19	59.04	61.25
QSAR biodegradation	66.26	82.75	87.11	85.40	85.78	84.36	85.97	81.90	85.97	80.66	85.97	86.82	85.88	61.42	75.92	83.60
Satellite	23.82	89.15	89.20	91.14	88.21	89.00	91.66	86.25	84.09	82.24	86.03	–	87.65	70.40	79.60	–
Seeds	33.33	89.05	95.24	93.33	90.48	93.81	94.29	90.00	96.67	88.10	95.24	–	95.24	87.62	91.43	89.52
Shuttle	78.60	99.91	99.71	99.95	99.73	99.98	99.99	99.95	94.39	99.44	96.82	–	99.94	79.98	92.93	99.94
Sonar	53.37	81.25	80.77	85.10	75.00	78.85	85.10	75.00	74.52	80.29	73.56	79.33	77.88	57.69	67.79	80.29

Çizelge C.8: (devam) Farklı öğrenme yaklaşımlarını benimseyen ve benchmark veri kümeleri üzerinde en iyi öğrenme sağlayan temel öğrencilerin global ölçekte karşılaştırma sonuçları (birinci grup)

Vehicle	25.77	71.16	81.91	70.80	69.74	71.04	76.00	73.17	78.13	60.99	80.61	–	82.27	53.07	45.27	71.16
Vertebral Column	67.74	80.32	84.52	81.94	84.52	84.52	83.55	80.32	84.19	76.45	85.48	84.19	85.48	72.90	77.74	80.00
Vowel	9.09	59.70	75.35	98.08	85.35	77.78	95.66	67.37	54.14	51.62	58.89	–	87.07	50.81	56.57	63.94
Wall Following Robot Navigation-1	40.41	99.84	92.63	99.01	61.35	100.00	100.00	100.00	70.77	99.91	99.67	–	99.98	99.98	90.58	99.84
Wall Following Robot Navigation-2	40.41	99.82	96.24	97.69	97.91	100.00	100.00	100.00	71.32	99.89	99.80	–	99.96	96.85	89.11	99.40
Wall Following Robot Navigation-3	40.41	95.27	87.90	94.06	89.39	99.65	99.67	99.58	65.74	88.23	70.49	–	98.09	92.19	52.46	–
Wine Quality-red	42.59	57.72	61.48	70.36	65.67	59.41	70.86	57.79	59.54	57.79	59.79	–	60.54	35.71	54.72	57.60
Wine Quality-white	44.88	52.65	54.61	70.01	64.29	53.29	70.36	55.80	53.14	48.12	53.55	–	61.15	27.79	44.30	54.94
Zoo	40.59	97.03	96.04	95.05	94.06	89.11	97.03	89.11	96.04	97.03	94.06	–	93.07	90.10	97.03	97.03
Ortalama	45.88	82.72	84.44	86.28	82.02	83.28	87.63	81.64	78.03	79.28	80.85	83.24	86.00	68.03	74.13	82.65
Global Rank	0	1	1	3	1	2	12	4	3	2	2	1	4	1	1	1

Çizelge C.9: Farklı öğrenme yaklaşımlarını benimseyen ve benchmark veri kümeleri üzerinde en iyi öğrenme sağlayan temel öğrenicilerin global ölçekte karşılaştırma sonuçları (ikinci grup)

Veri kümesi	QDA	Simple Cart	IBk	Spegasos	Kernel Logistic Regression	FT	Input Mapped Classifier	RoughSet	BFTree	J48	LADTree	Random Tree	CDT	PART	SMO
Banknote Authentication	98.40	98.25	99.85	98.54	98.91	99.93	98.54	98.83	98.40	98.54	97.38	98.54	97.81	98.98	98.03
Blood Transfusion	77.94	77.27	69.92	76.60	77.14	78.74	76.87	67.51	78.61	76.87	78.34	70.32	78.21	77.01	76.20
Boston Housing2	82.02	84.39	82.81	–	–	79.84	89.53	93.28	84.98	89.53	72.92	90.32	76.28	88.74	50.99
Breast Cancer	–	94.99	95.42	96.42	96.71	96.71	94.13	–	94.99	94.13	95.71	94.56	94.56	94.85	96.71
Breast Tissue	69.81	68.87	71.70	–	–	70.75	66.04	66.04	68.87	66.04	60.38	62.26	69.81	67.92	60.38
Cardiotocography-1	83.44	92.85	91.16	–	–	93.23	93.23	89.75	92.57	93.23	92.00	91.91	92.66	93.13	89.37
Cardiotocography-2	69.38	82.83	78.27	–	–	83.30	83.21	78.41	83.02	83.21	79.92	79.02	81.51	83.44	78.79
Climate Model	92.96	91.85	89.07	96.30	95.37	96.11	92.59	91.85	91.30	92.59	93.70	91.11	91.11	92.96	95.37

Çizelge C.9: (devam) Farklı öğrenme yaklaşımlarını benimseyen ve benchmark veri kümeleri üzerinde en iyi öğrenme sağlayan temel öğrencilerin global ölçekte karşılaştırma sonuçları (ikinci grup)

DNA	90.43	92.84	74.64	–	–	95.01	92.12	+++	92.40	92.12	92.81	76.02	92.75	92.72	93.47
Glass	57.94	70.56	70.56	–	–	64.95	65.89	71.50	71.50	65.89	67.29	68.69	63.08	67.76	57.48
HTRU2	96.73	97.92	97.13	97.52	–	97.94	97.91	97.60	97.86	97.91	97.88	96.91	97.80	97.83	97.56
Ionosphere	89.17	88.32	87.18	85.76	88.89	89.17	88.89	84.62	89.17	88.89	89.74	86.04	89.17	90.03	87.46
Leaf	64.71	63.53	73.53	–	–	78.24	61.18	64.12	62.06	61.18	54.41	57.94	57.94	64.71	55.29
Letter Recognition	88.65	86.97	95.97	–	–	89.86	88.12	+++	86.36	88.12	51.33	86.03	83.77	88.62	82.22
Libras Movement	86.11	66.67	85.83	–	–	78.33	69.72	+++	62.78	69.72	53.06	67.22	55.28	70.56	74.72
LSVT Voice Rehabilitation	–	82.54	73.81	82.54	81.75	82.54	73.81	80.16	80.16	73.81	78.57	70.63	80.95	73.81	84.13
Madelon	52.00	79.65	54.15	53.25	53.95	57.55	70.35	+++	76.20	70.35	72.65	54.20	76.00	59.50	54.95
MAGIC Gamma Telescope	78.44	85.03	80.94	79.06	–	86.14	85.06	77.41	84.39	85.06	82.97	82.13	85.05	85.46	79.15
Parkinson Speech	56.25	64.81	63.65	63.46	63.37	62.98	62.98	68.17	63.85	62.98	63.17	60.87	62.31	59.71	63.17
QSAR biodegradation	72.23	81.14	84.46	86.54	85.97	85.78	82.37	83.32	81.71	82.37	83.13	82.46	83.13	85.21	85.59
Satellite	85.73	87.04	90.26	–	–	87.07	86.62	+++	86.68	86.62	82.75	84.82	85.78	87.13	86.68
Seeds	95.24	90.48	94.29	–	–	95.24	91.90	92.38	93.33	91.90	91.90	90.00	89.52	92.86	93.81
Shuttle	83.20	99.97	99.94	–	–	99.92	99.97	99.97	99.97	99.97	99.86	99.97	99.94	99.97	96.93
Sonar	77.40	71.15	86.54	75.00	75.96	80.77	71.15	72.60	71.15	71.15	78.85	73.56	74.04	80.29	75.96

Çizelge C.9: (devam) Farklı öğrenme yaklaşımlarını benimseyen ve benchmark veri kümeleri üzerinde en iyi öğrenme sağlayan temel öğrencilerin global ölçekte karşılaştırma sonuçları (ikinci grup)

Vehicle	85.34	70.92	71.28	–	–	79.91	49.41	71.51	69.62	74.35	67.85	69.86	69.86	73.40	74.47
Vertebral Column	82.90	82.90	81.61	85.48	85.48	85.48	81.61	80.00	81.61	81.61	79.35	79.03	83.23	80.97	78.71
Vowel	79.29	77.07	98.38	–	–	79.29	78.99	83.84	75.15	78.99	46.46	77.88	68.08	73.94	57.98
Wall Following Robot Navigation-1	90.60	100.00	98.83	–	–	99.98	100.00	100.00	100.00	100.00	100.00	100.00	100.00	99.98	77.20
Wall Following Robot Navigation-2	88.58	100.00	97.21	–	–	99.96	100.00	100.00	100.00	100.00	100.00	99.96	100.00	99.98	77.27
Wall Following Robot Navigation-3	68.07	99.41	88.18	–	–	98.57	99.65	97.62	99.29	99.65	98.81	97.73	99.58	99.78	71.50
Wine Quality-red	54.66	60.79	63.98	–	–	58.79	60.73	67.23	59.98	60.73	58.79	63.48	58.16	59.10	58.22
Wine Quality-white	48.33	58.37	65.84	–	–	55.84	58.88	66.35	57.82	58.88	51.67	62.84	54.53	59.35	52.12
Zoo	96.04	92.08	96.04	–	–	95.05	92.08	93.07	92.08	92.08	95.05	94.06	89.11	93.07	97.03
Ortalama	78.77	83.07	83.41	82.81	82.14	84.33	81.93	82.86	82.66	82.68	79.05	80.62	81.24	82.81	77.54
Global Rank	2	3	2	1	1	1	2	2	2	2	2	1	2	1	1

Çizelge C.10: Topluluk/meta öğrenme algoritmalarının (birinci grup) benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Veri kümesi	Topluluk/meta öğrenme													
	AdaBoostM1 ¹	Attribute Selected Classifier ¹	Bagging ¹	Classification Via Clustering ²	Classification Via Regression ¹	Dagging ¹	Decorate ¹	END ¹	Filtered Classifier ¹	Grading ³	LogitBoost ¹	MultiBoostAB ¹	MultiClass Classifier ¹	Multi Scheme ⁴
Banknote Authentication	99.42	93.73	99.27	57.43	99.27	97.16	99.78	99.34	96.36	99.42	96.50	99.27	99.27	99.34
Blood Transfusion	73.66	76.34	75.94	58.96	73.26	79.01	74.20	74.06	74.87	76.87	78.07	74.20	73.93	73.93
Boston Housing2	95.85	95.85	95.45	9.29	95.45	72.73	+++	94.47	96.05	95.65	92.29	95.85	95.65	95.85
Breast Cancer	96.57	96.42	96.42	95.71	96.85	96.57	97.00	96.42	95.99	96.42	95.85	96.28	96.57	96.57
Breast Tissue	70.75	73.58	71.70	39.62	70.75	67.92	72.64	72.64	59.43	71.70	68.87	72.64	74.53	70.75
Cardiotocography-1	94.68	94.40	94.68	53.43	94.83	91.02	94.78	95.06	93.27	94.50	92.62	94.78	95.16	94.68

¹ Temel Öğrenici olarak Random Forest algoritması seçildi. Diğer parametreler varsayılan değerlere sahiptir.

² Parametrelerin varsayılan değerleri kullanıldı.

³ Meta Öğrenici olarak Random Forest algoritması, Temel Öğrenici olarak: Random Forest, IBk, J48 ve NaiveBayes seçildi. Diğer parametreler varsayılan değerlere sahiptir.

⁴ Temel Öğrenici olarak: Random Forest, IBk, J48 ve NaiveBayes seçildi. Diğer parametreler varsayılan değerlere sahiptir.

Çizelge C.10: (devam) Topluluk/meta öğrenme algoritmalarının (birinci grup) benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Cardiotocography-2	87.49	85.42	87.86	30.29	87.91	80.95	+++	88.57	86.08	87.58	84.85	88.24	88.48	87.96
Climate Model	93.89	93.70	93.33	53.33	93.70	91.48	92.41	93.52	93.52	94.07	93.52	93.89	93.70	93.33
DNA	94.66	92.78	95.42	59.38	95.54	92.03	95.79	95.79	95.61	95.54	93.75	95.32	95.67	95.14
Glass	79.91	78.97	79.44	42.52	78.97	64.49	81.31	82.24	73.36	77.57	71.50	78.97	80.37	80.84
HTRU2	98.01	97.91	98.07	91.87	98.04	97.92	+++	98.07	97.81	97.98	97.77	98.12	98.09	97.85
Ionosphere	93.16	92.88	94.02	70.94	93.16	90.31	93.73	93.45	92.02	94.30	92.59	93.45	94.02	94.02
Leaf	78.53	76.47	78.24	6.76	75.00	63.53	78.53	79.12	61.18	79.71	71.18	76.76	76.18	77.65
Letter Recognition	96.36	95.84	???	5.04	96.34	91.41	+++	???	93.24	+++	73.48	96.36	???	96.44
Libras Movement	83.89	79.72	82.22	11.11	78.89	66.67	86.39	83.89	75.56	85.28	72.50	83.33	82.22	83.06
LSVT Voice Rehabilitation	84.13	82.54	84.92	51.59	84.13	77.78	81.75	84.13	82.54	80.95	76.19	81.75	83.33	83.33
Madelon	64.05	79.10	63.60	57.55	63.85	58.50	63.80	63.50	67.55	66.80	61.75	62.95	65.35	63.60
MAGIC Gamma Telescope	88.08	83.88	87.97	59.02	88.13	86.80	+++	88.11	83.68	87.40	80.53	88.16	88.00	88.03
Parkinson Speech	70.10	68.08	68.37	49.62	70.87	67.50	70.29	69.81	61.63	68.94	61.35	68.46	70.48	69.42
QSAR biodegradation	86.45	85.78	86.73	52.80	86.64	83.98	87.49	86.07	84.55	87.58	83.03	87.96	87.01	87.01
Satellite	91.73	91.78	91.73	39.81	91.55	88.56	+++	91.76	89.49	91.89	85.50	91.66	91.84	91.70
Seeds	93.81	94.29	93.33	65.24	93.33	88.10	92.38	93.33	90.00	94.76	93.33	94.76	93.33	94.29

Çizelge C.10: (devam) Topluluk/meta öğrenme algoritmalarının (birinci grup) benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Shuttle	99.99	99.85	99.99	76.67	99.99	99.89	+++	99.99	99.95	99.99	99.96	99.99	99.99	99.99
Sonar	84.13	77.40	81.73	54.33	84.62	76.44	87.98	84.13	78.85	85.10	79.33	82.21	81.25	84.62
Vehicle	75.06	71.51	75.18	37.12	75.77	71.63	75.89	74.82	70.45	76.24	70.92	75.30	74.40	76.00
Vertebral Column	83.23	78.06	84.84	66.45	83.87	82.90	84.52	85.16	81.29	84.84	80.97	85.48	84.84	83.23
Vowel	95.66	90.51	95.56	9.29	94.44	79.29	97.17	94.95	68.48	96.36	62.73	96.16	94.55	95.35
Wall Following Robot Navigation-1	100.00	100.00	100.00	48.75	100.00	99.89	+++	100.00	99.76	100.00	99.98	100.00	100.00	100.00
Wall Following Robot Navigation-2	100.00	100.00	100.00	47.09	100.00	99.89	+++	100.00	99.78	100.00	99.98	100.00	100.00	100.00
Wall Following Robot Navigation-3	99.52	99.52	99.51	49.23	99.51	98.52	+++	99.58	98.26	99.58	98.92	99.62	99.62	99.58
Wine Quality-red	70.67	69.29	71.04	46.97	70.17	61.54	70.73	71.11	59.04	67.35	57.97	71.36	71.29	69.92
Wine Quality-white	70.29	68.56	69.82	45.24	70.03	59.31	+++	70.52	57.04	68.58	53.65	70.25	70.54	70.25
Zoo	97.03	94.06	95.05	60.40	97.03	87.13	96.04	97.03	96.04	96.04	97.03	97.03	96.04	95.05
Ortalama	87.60	86.61	87.23	48.57	87.33	82.15	85.21	87.52	83.42	87.47	82.38	87.59	87.37	87.54
Global Rank	4	3	3	0	5	1	1	5	1	3	1	7	3	3

Çizelge C.11: Topluluk/meta öğrenme algoritmalarının (ikinci grup) benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Veri kümesi	Topluluk/meta öğrenme													
	Class Balanced ND ¹	Data Near Balanced ND ¹	Furthest Centroid ND ¹	ND ¹	Random Pair ND ¹	Ordinal Class Classifier ¹	Raced Incremental Logit Boost ¹	Random Committee ¹	Random SubSpace ¹	Rotation Forest ¹	Stacking ³	Threshold Selector ¹	Vote ⁴	Randomizable Filtered Classifier ¹
Banknote Authentication	99.34	99.34	99.34	98.54	99.34	99.27	55.54	99.27	97.59	100.00	99.78	99.34	98.91	99.20
Blood Transfusion	73.93	73.93	73.93	76.87	73.93	73.66	76.20	74.06	76.20	76.47	76.74	60.03	75.80	75.00
Boston Housing2	90.51	89.92	93.87	80.63	92.09	94.66	5.93	95.85	95.65	94.47	71.74	–	94.47	95.26
Breast Cancer	96.42	96.42	96.42	94.13	96.42	96.42	65.52	96.71	96.57	97.28	96.42	96.57	95.99	96.42
Breast Tissue	69.81	67.92	77.36	63.21	70.75	69.81	18.87	74.53	72.64	77.36	73.58	–	73.58	69.81
Cardiotocography-1	94.45	95.01	94.73	93.18	94.73	95.20	92.24	95.01	93.79	93.84	94.78	–	93.13	90.78
Cardiotocography-2	87.96	87.72	88.43	83.54	88.29	88.52	82.36	88.85	87.91	86.69	88.66	–	86.17	72.48
Climate Model	94.07	94.07	94.07	92.59	94.07	93.70	91.48	93.52	91.48	92.22	94.44	92.22	93.70	92.22
DNA	95.79	95.17	95.70	94.16	95.29	94.95	95.20	95.64	94.32	95.61	96.05	–	95.04	65.73

Çizelge C.11: (devam) Topluluk/meta öğrenme algoritmalarının (ikinci grup) benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Glass	80.37	79.91	78.50	69.63	78.50	80.37	35.51	80.37	80.84	76.64	77.10	–	74.77	68.22
HTRU2	98.06	98.06	98.06	97.91	98.06	98.09	97.87	98.05	97.78	98.02	97.80	98.02	97.96	97.69
Ionosphere	93.45	93.45	93.45	88.89	93.45	94.02	64.10	93.45	92.59	94.59	92.88	94.02	92.88	90.31
Leaf	74.12	74.41	72.35	55.29	72.94	74.71	4.71	77.94	77.35	82.35	74.71	–	77.65	68.24
Letter Recognition	95.90	95.77	95.52	86.09	95.59	+++	+	+++	+++	+++	96.98	–	95.81	85.56
Libras Movement	81.67	81.11	83.33	61.67	81.11	80.28	5.56	83.06	84.72	88.61	85.56	–	83.89	78.61
LSVT Voice Rehabilitation	84.13	84.13	84.13	73.81	84.13	83.33	66.67	84.13	84.13	82.54	76.19	81.75	73.02	61.90
Madelon	63.50	63.50	63.50	70.35	63.50	65.35	58.50	65.60	64.55	60.35	66.40	61.60	64.05	52.70
MAGIC Gamma Telescope	87.97	87.97	87.97	85.06	87.97	88.00	87.71	88.13	+++	87.56	87.04	88.03	84.90	83.73
Parkinson Speech	68.75	68.75	68.75	62.98	68.75	70.48	50.00	69.81	69.90	70.29	65.58	67.60	66.25	66.83
QSAR biodegradation	86.92	86.92	86.92	82.37	86.92	87.01	66.26	86.54	86.45	87.30	83.98	86.92	85.50	83.41
Satellite	91.50	91.52	91.61	85.81	91.72	91.61	91.42	91.90	91.78	91.36	92.37	–	90.38	88.00
Seeds	93.33	94.29	93.81	94.29	93.81	92.86	33.33	93.81	93.33	95.24	92.38	–	94.76	91.90
Shuttle	99.99	99.99	99.99	99.97	99.99	99.99	99.86	99.99	99.94	99.98	99.99	–	99.96	99.95
Sonar	84.13	84.13	84.13	71.15	84.13	81.25	53.37	85.58	86.06	84.62	86.06	82.69	83.17	80.29
Vehicle	77.07	77.07	75.30	72.46	74.82	74.82	25.77	75.53	74.47	78.84	75.41	–	74.94	67.26

Çizelge C.11: (devam) Topluluk/meta öğrenme algoritmalarının (ikinci grup) benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Vertebral Column	86.45	86.45	86.45	81.61	86.45	84.84	67.74	84.84	79.03	86.45	85.16	84.52	86.13	79.68
Vowel	93.74	92.93	92.53	72.42	93.43	92.22	9.09	96.06	94.14	97.98	98.28	–	94.65	89.60
Wall Following Robot Navigation-1	99.98	99.98	99.98	99.98	100.00	99.98	99.98	100.00	88.82	98.88	100.00	–	100.00	97.82
Wall Following Robot Navigation-2	99.96	99.96	99.98	99.98	100.00	99.98	99.96	100.00	98.75	98.85	100.00	–	99.87	98.99
Wall Following Robot Navigation-3	99.63	99.63	99.60	99.34	99.49	99.58	99.29	99.62	98.53	97.12	99.71	–	97.54	88.49
Wine Quality-red	71.11	70.54	70.98	60.54	70.73	69.54	42.59	70.92	70.86	70.92	69.67	–	67.73	66.35
Wine Quality-white	69.89	69.58	69.72	56.86	69.52	70.03	63.43	70.66	70.21	70.76	70.56	–	67.39	66.78
Zoo	96.04	97.03	97.03	91.09	96.04	96.04	40.59	96.04	97.03	95.05	95.05	–	96.04	92.08
Ortalama	87.27	87.17	87.50	81.71	87.15	86.89	60.83	87.67	86.69	87.76	86.70	84.10	86.55	81.86
Global Rank	2	3	4	0	4	2	0	4	1	10	9	0	1	0

Çizelge C.12: Topluluk/meta öğrenme algoritmalarının (üçüncü grup) benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Veri kümesi	Topluluk/meta öğrenme		
	Weighted Instances Handler Wrapper ¹	Iterative Classifier Optimizer ¹	Real AdaBoost ¹
Banknote Authentication	99.27	99.49	99.56
Blood Transfusion	73.66	72.99	73.93
Boston Housing2	95.65	+++	–
Breast Cancer	96.42	96.42	96.42
Breast Tissue	73.58	72.64	–
Cardiotocography-1	94.83	+++	–
Cardiotocography-2	88.33	+++	–
Climate Model	93.70	93.89	93.70
DNA	95.14	+++	–
Glass	80.37	81.31	–
HTRU2	98.09	+++	98.07
Ionosphere	94.02	92.88	94.02
Leaf	78.24	+++	–
Letter Recognition	96.42	+++	–

Çizelge C.12: (devam) Topluluk/meta öğrenme algoritmalarının (üçüncü grup) benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Libras Movement	83.33	+++	–
LSVT Voice Rehabilitation	83.33	86.51	85.71
Madelon	65.35	+++	67.00
MAGIC Gamma Telescope	88.00	+++	88.38
Parkinson Speech	70.48	70.38	70.87
QSAR biodegradation	87.01	+++	87.30
Satellite	91.93	+++	–
Seeds	93.81	92.86	–
Shuttle	99.99	+++	–
Sonar	81.25	85.58	86.54
Vehicle	74.94	+++	–
Vertebral Column	84.84	84.84	83.23
Vowel	95.35	+++	–
Wall Following Robot Navigation-1	100.00	100.00	–
Wall Following Robot Navigation-2	100.00	100.00	–

Çizelge C.12: (devam) Topluluk/meta öğrenme algoritmalarının (üçüncü grup) benchmark veri kümeleri üzerindeki sınıflandırma sonuçları

Wall Following Robot Navigation-3	99.63	+++	–
Wine Quality-red	69.92	+++	–
Wine Quality-white	70.38	+++	–
Zoo	95.05	96.04	–
Ortalama	87.65	88.39	86.52
Global Rank	3	3	2

Çizelge C.13: Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmalarının global ölçekte karşılaştırma sonuçları (birinci grup)

Veri kümesi	SMO	AIDE	LogitBoost ¹	RseslibKn	Random SubSpace ¹	FURIA	Random Forest	Rotation Forest ¹	REPtree	LDA	Furthest Centroid ND ¹	Ordinal Class Classifier ¹	Random Committee ¹	SGD	LMT	END ¹
Banknote Authentication	98.03	95.04	96.50	100.00	97.59	99.20	99.27	100.00	98.03	97.67	99.34	99.27	99.27	98.47	99.93	99.34
Blood Transfusion	76.20	74.73	78.07	73.93	76.20	77.41	73.93	76.47	79.28	76.74	73.93	73.66	74.06	75.94	77.94	74.06
Boston Housing2	50.99	95.45	92.29	94.47	95.65	83.60	95.45	94.47	84.78	96.44	93.87	94.66	95.85	–	90.91	94.47
Breast Cancer	96.71	96.57	95.85	96.42	96.57	95.14	96.57	97.28	94.42	–	96.42	96.42	96.71	96.57	96.57	96.42
Breast Tissue	60.38	65.09	68.87	65.09	72.64	63.21	71.70	77.36	72.64	65.09	77.36	69.81	74.53	–	68.87	72.64
Cardiotocography-1	89.37	91.35	92.62	92.14	93.79	93.74	94.78	93.84	92.99	87.30	94.73	95.20	95.01	–	94.03	95.06
Cardiotocography-2	78.79	81.51	84.85	82.36	87.91	83.49	88.48	86.69	82.31	74.27	88.43	88.52	88.85	–	84.62	88.57
Climate Model	95.37	92.22	93.52	93.15	91.48	94.63	93.70	92.22	91.48	95.56	94.07	93.70	93.52	96.48	96.11	93.52

Çizelge C.13: (devam) Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmalarının global ölçekte karşılaştırma sonuçları (birinci grup)

DNA	93.47	95.29	93.75	90.99	94.32	94.38	94.82	95.61	92.81	94.85	95.70	94.95	95.64	–	96.23	95.79
Glass	57.48	76.17	71.50	77.57	80.84	74.77	80.37	76.64	62.15	65.42	78.50	80.37	80.37	–	68.22	82.24
HTRU2	97.56	97.42	97.77	97.93	97.78	97.87	98.08	98.02	97.78	97.46	98.06	98.09	98.05	97.88	97.98	98.07
Ionosphere	87.46	93.45	92.59	92.02	92.59	91.45	93.16	94.59	88.89	86.32	93.45	94.02	93.45	86.04	93.73	93.45
Leaf	55.29	66.47	71.18	73.24	77.35	65.00	76.76	82.35	58.24	81.18	72.35	74.71	77.94	–	81.18	79.12
Letter Recognition	82.22	91.30	73.48	97.20	+++	+++	96.42	+++	84.26	70.19	95.52	+++	+++	–	92.14	???
Libras Movement	74.72	75.28	72.50	83.61	84.72	64.44	82.50	88.61	56.67	67.50	83.33	80.28	83.06	–	76.94	83.89
LSVT Voice Rehabilitation	84.13	82.54	76.19	77.78	84.13	83.33	83.33	82.54	80.95	–	84.13	83.33	84.13	83.33	85.71	84.13
Madelon	54.95	62.00	61.75	72.35	64.55	72.75	64.55	60.35	73.40	53.75	63.50	65.35	65.60	53.25	71.30	63.50
MAGIC Gamma Telescope	79.15	82.52	80.53	85.32	+++	84.69	88.04	87.56	85.01	78.41	87.97	88.00	88.13	79.15	86.17	88.11
Parkinson Speech	63.17	59.62	61.35	69.90	69.90	65.10	69.62	70.29	61.63	62.12	68.75	70.48	69.81	64.62	65.10	69.81
QSAR biodegradation	85.59	82.75	83.03	85.40	86.45	84.36	85.97	87.30	81.90	85.97	86.92	87.01	86.54	86.82	85.88	86.07
Satellite	86.68	89.15	85.50	91.14	91.78	89.00	91.66	91.36	86.25	84.09	91.61	91.61	91.90	–	87.65	91.76
Seeds	93.81	89.05	93.33	93.33	93.33	93.81	94.29	95.24	90.00	96.67	93.81	92.86	93.81	–	95.24	93.33
Shuttle	96.93	99.91	99.96	99.95	99.94	99.98	99.99	99.98	99.95	94.39	99.99	99.99	99.99	–	99.94	99.99
Sonar	75.96	81.25	79.33	85.10	86.06	78.85	85.10	84.62	75.00	74.52	84.13	81.25	85.58	79.33	77.88	84.13

Çizelge C.13: (devam) Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmalarının global ölçekte karşılaştırma sonuçları (birinci grup)

Vehicle	74.47	71.16	70.92	70.80	74.47	71.04	76.00	78.84	73.17	78.13	75.30	74.82	75.53	–	82.27	74.82
Vertebral Column	78.71	80.32	80.97	81.94	79.03	84.52	83.55	86.45	80.32	84.19	86.45	84.84	84.84	84.19	85.48	85.16
Vowel	57.98	59.70	62.73	98.08	94.14	77.78	95.66	97.98	67.37	54.14	92.53	92.22	96.06	–	87.07	94.95
Wall Following Robot Navigation-1	77.20	99.84	99.98	99.01	88.82	100.00	100.00	98.88	100.00	70.77	99.98	99.98	100.00	–	99.98	100.00
Wall Following Robot Navigation-2	77.27	99.82	99.98	97.69	98.75	100.00	100.00	98.85	100.00	71.32	99.98	99.98	100.00	–	99.96	100.00
Wall Following Robot Navigation-3	71.50	95.27	98.92	94.06	98.53	99.65	99.67	97.12	99.58	65.74	99.60	99.58	99.62	–	98.09	99.58
Wine Quality-red	58.22	57.72	57.97	70.36	70.86	59.41	70.86	70.92	57.79	59.54	70.98	69.54	70.92	–	60.54	71.11
Wine Quality-white	52.12	52.65	53.65	70.01	70.21	53.29	70.36	70.76	55.80	53.14	69.72	70.03	70.66	–	61.15	70.52
Zoo	97.03	97.03	97.03	95.05	97.03	89.11	97.03	95.05	89.11	96.04	97.03	96.04	96.04	–	93.07	97.03
Ortalama	77.54	82.72	82.38	86.28	86.69	83.28	87.63	87.76	81.64	78.03	87.50	86.89	87.67	83.24	86.00	87.52
Global Rank	1	1	1	2	1	2	4	7	3	2	4	2	4	1	1	5

Çizelge C.14: Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmalarının global ölçekte karşılaştırma sonuçları (ikinci grup)

Veri kümesi	MultiBoostAB ¹	VFI	Iterative Classifier Optimizer ¹	Simple Cart	Real AdaBoost ¹	Classification Via Regression ¹	Stacking ³	AdaBoostM1 ¹	Bagging ¹	Grading ³	MultiClass Classifier ¹	Multi Scheme ⁴	Class Balanced ND ¹	Data Near Balanced ND ¹	Random Pair ND ¹	Weighted Instances Handler Wrapper ¹
Banknote Authentication	99.27	78.50	99.49	98.25	99.56	99.27	99.78	99.42	99.27	99.42	99.27	99.34	99.34	99.34	99.34	99.27
Blood Transfusion	74.20	36.76	72.99	77.27	73.93	73.26	76.74	73.66	75.94	76.87	73.93	73.93	73.93	73.93	73.93	73.66
Boston Housing2	95.85	92.69	+++	84.39	–	95.45	71.74	95.85	95.45	95.65	95.65	95.85	90.51	89.92	92.09	95.65
Breast Cancer	96.28	95.57	96.42	94.99	96.42	96.85	96.42	96.57	96.42	96.42	96.57	96.57	96.42	96.42	96.42	96.42
Breast Tissue	72.64	66.98	72.64	68.87	–	70.75	73.58	70.75	71.70	71.70	74.53	70.75	69.81	67.92	70.75	73.58
Cardiotocography-1	94.78	75.21	+++	92.85	–	94.83	94.78	94.68	94.68	94.50	95.16	94.68	94.45	95.01	94.73	94.83
Cardiotocography-2	88.24	52.87	+++	82.83	–	87.91	88.66	87.49	87.86	87.58	88.48	87.96	87.96	87.72	88.29	88.33
Climate Model	93.89	58.70	93.89	91.85	93.70	93.70	94.44	93.89	93.33	94.07	93.70	93.33	94.07	94.07	94.07	93.70

Çizelge C.14: (devam) Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmalarının global ölçekte karşılaştırma sonuçları (ikinci grup)

DNA	95.32	70.59	+++	92.84	–	95.54	96.05	94.66	95.42	95.54	95.67	95.14	95.79	95.17	95.29	95.14
Glass	78.97	57.01	81.31	70.56	–	78.97	77.10	79.91	79.44	77.57	80.37	80.84	80.37	79.91	78.50	80.37
HTRU2	98.12	93.31	+++	97.92	98.07	98.04	97.80	98.01	98.07	97.98	98.09	97.85	98.06	98.06	98.06	98.09
Ionosphere	93.45	94.87	92.88	88.32	94.02	93.16	92.88	93.16	94.02	94.30	94.02	94.02	93.45	93.45	93.45	94.02
Leaf	76.76	43.24	+++	63.53	–	75.00	74.71	78.53	78.24	79.71	76.18	77.65	74.12	74.41	72.94	78.24
Letter Recognition	96.36	61.24	+++	86.97	–	96.34	96.98	96.36	???	+++	???	96.44	95.90	95.77	95.59	96.42
Libras Movement	83.33	58.61	+++	66.67	–	78.89	85.56	83.89	82.22	85.28	82.22	83.06	81.67	81.11	81.11	83.33
LSVT Voice Rehabilitation	81.75	62.70	86.51	82.54	85.71	84.13	76.19	84.13	84.92	80.95	83.33	83.33	84.13	84.13	84.13	83.33
Madelon	62.95	47.80	+++	79.65	67.00	63.85	66.40	64.05	63.60	66.80	65.35	63.60	63.50	63.50	63.50	65.35
MAGIC Gamma Telescope	88.16	66.49	+++	85.03	88.38	88.13	87.04	88.08	87.97	87.40	88.00	88.03	87.97	87.97	87.97	88.00
Parkinson Speech	68.46	55.19	70.38	64.81	70.87	70.87	65.58	70.10	68.37	68.94	70.48	69.42	68.75	68.75	68.75	70.48
QSAR biodegradation	87.96	61.42	+++	81.14	87.30	86.64	83.98	86.45	86.73	87.58	87.01	87.01	86.92	86.92	86.92	87.01
Satellite	91.66	70.40	+++	87.04	–	91.55	92.37	91.73	91.73	91.89	91.84	91.70	91.50	91.52	91.72	91.93
Seeds	94.76	87.62	92.86	90.48	–	93.33	92.38	93.81	93.33	94.76	93.33	94.29	93.33	94.29	93.81	93.81
Shuttle	99.99	79.98	+++	99.97	–	99.99	99.99	99.99	99.99	99.99	99.99	99.99	99.99	99.99	99.99	99.99
Sonar	82.21	57.69	85.58	71.15	86.54	84.62	86.06	84.13	81.73	85.10	81.25	84.62	84.13	84.13	84.13	81.25

Çizelge C.14: (devam) Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmalarının global ölçekte karşılaştırma sonuçları (ikinci grup)

Vehicle	75.30	53.07	+++	70.92	–	75.77	75.41	75.06	75.18	76.24	74.40	76.00	77.07	77.07	74.82	74.94
Vertebral Column	85.48	72.90	84.84	82.90	83.23	83.87	85.16	83.23	84.84	84.84	84.84	83.23	86.45	86.45	86.45	84.84
Vowel	96.16	50.81	+++	77.07	–	94.44	98.28	95.66	95.56	96.36	94.55	95.35	93.74	92.93	93.43	95.35
Wall Following Robot Navigation-1	100.00	99.98	100.00	100.00	–	100.00	100.00	100.00	100.00	100.00	100.00	100.00	99.98	99.98	100.00	100.00
Wall Following Robot Navigation-2	100.00	96.85	100.00	100.00	–	100.00	100.00	100.00	100.00	100.00	100.00	100.00	99.96	99.96	100.00	100.00
Wall Following Robot Navigation-3	99.62	92.19	+++	99.41	–	99.51	99.71	99.52	99.51	99.58	99.62	99.58	99.63	99.63	99.49	99.63
Wine Quality-red	71.36	35.71	+++	60.79	–	70.17	69.67	70.67	71.04	67.35	71.29	69.92	71.11	70.54	70.73	69.92
Wine Quality-white	70.25	27.79	+++	58.37	–	70.03	70.56	70.29	69.82	68.58	70.54	70.25	69.89	69.58	69.52	70.38
Zoo	97.03	90.10	96.04	92.08	–	97.03	95.05	97.03	95.05	96.04	96.04	95.05	96.04	97.03	96.04	95.05
Ortalama	87.59	68.03	88.39	83.07	86.52	87.33	86.70	87.60	87.23	87.47	87.37	87.54	87.27	87.17	87.15	87.65
Global Rank	7	1	3	3	2	5	3	4	3	3	3	3	2	3	4	3

Çizelge C.15: Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmalarının global ölçekte karşılaştırma sonuçları (üçüncü grup)

Veri kümesi	Decorate ¹	QDA	IBk	Input Mapped Classifier	RoughSet	BFTree	J48	LADTree	Random Tree	CDT	Attribute Selected Classifier ¹	Vote ⁴	PART	Bayesian Network	Naive Bayes	A2DE		
Banknote Authentication	99.78	98.40	99.85	98.54	98.83	98.40	98.54	97.38	98.54	97.81	93.73	98.91	98.98	92.13	84.26	96.21		
Blood Transfusion	74.20	77.94	69.92	76.87	67.51	78.61	76.87	78.34	70.32	78.21	76.34	75.80	77.01	74.73	74.87	73.93		
Boston Housing2	+	+	+	82.02	82.81	89.53	93.28	84.98	89.53	72.92	90.32	76.28	95.85	94.47	88.74	93.28	89.33	–
Breast Cancer	97.00	–	95.42	94.13	–	94.99	94.13	95.71	94.56	94.56	96.42	95.99	94.85	97.14	96.14	96.57		
Breast Tissue	72.64	69.81	71.70	66.04	66.04	68.87	66.04	60.38	62.26	69.81	73.58	73.58	67.92	66.04	70.75	63.21		
Cardiotocography-1	94.78	83.44	91.16	93.23	89.75	92.57	93.23	92.00	91.91	92.66	94.40	93.13	93.13	86.27	82.36	92.43		
Cardiotocography-2	+	+	+	69.38	78.27	83.21	78.41	83.02	83.21	79.92	79.02	81.51	85.42	86.17	83.44	74.04	71.54	84.34
Climate Model	92.41	92.96	89.07	92.59	91.85	91.30	92.59	93.70	91.11	91.11	93.70	93.70	92.96	92.22	95.00	92.78		

Çizelge C.15: (devam) Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmalarının global ölçekte karşılaştırma sonuçları (üçüncü grup)

DNA	95.79	90.43	74.64	92.12	+++	92.40	92.12	92.81	76.02	92.75	92.78	95.04	92.72	94.92	94.48	–
Glass	81.31	57.94	70.56	65.89	71.50	71.50	65.89	67.29	68.69	63.08	78.97	74.77	67.76	74.77	49.53	75.70
HTRU2	+++	96.73	97.13	97.91	97.60	97.86	97.91	97.88	96.91	97.80	97.91	97.96	97.83	96.87	94.51	97.95
Ionosphere	93.73	89.17	87.18	88.89	84.62	89.17	88.89	89.74	86.04	89.17	92.88	92.88	90.03	89.46	82.34	94.30
Leaf	78.53	64.71	73.53	61.18	64.12	62.06	61.18	54.41	57.94	57.94	76.47	77.65	64.71	65.00	74.12	64.71
Letter Recognition	+++	88.65	95.97	88.12	+++	86.36	88.12	51.33	86.03	83.77	95.84	95.81	88.62	74.36	64.08	94.31
Libras Movement	86.39	86.11	85.83	69.72	+++	62.78	69.72	53.06	67.22	55.28	79.72	83.89	70.56	61.39	62.78	–
LSVT Voice Rehabilitation	81.75	–	73.81	73.81	80.16	80.16	73.81	78.57	70.63	80.95	82.54	73.02	73.81	73.81	54.76	–
Madelon	63.80	52.00	54.15	70.35	+++	76.20	70.35	72.65	54.20	76.00	79.10	64.05	59.50	61.35	58.40	–
MAGIC Gamma Telescope	+++	78.44	80.94	85.06	77.41	84.39	85.06	82.97	82.13	85.05	83.88	84.90	85.46	77.77	72.69	83.94
Parkinson Speech	70.29	56.25	63.65	62.98	68.17	63.85	62.98	63.17	60.87	62.31	68.08	66.25	59.71	59.90	59.04	61.25
QSAR biodegradation	87.49	72.23	84.46	82.37	83.32	81.71	82.37	83.13	82.46	83.13	85.78	85.50	85.21	80.66	75.92	83.60
Satellite	+++	85.73	90.26	86.62	+++	86.68	86.62	82.75	84.82	85.78	91.78	90.38	87.13	82.24	79.60	–
Seeds	92.38	95.24	94.29	91.90	92.38	93.33	91.90	91.90	90.00	89.52	94.29	94.76	92.86	88.10	91.43	89.52
Shuttle	+++	83.20	99.94	99.97	99.97	99.97	99.97	99.86	99.97	99.94	99.85	99.96	99.97	99.44	92.93	99.94
Sonar	87.98	77.40	86.54	71.15	72.60	71.15	71.15	78.85	73.56	74.04	77.40	83.17	80.29	80.29	67.79	80.29

Çizelge C.15: (devam) Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmalarının global ölçekte karşılaştırma sonuçları (üçüncü grup)

Vehicle	75.89	85.34	71.28	49.41	71.51	69.62	74.35	67.85	69.86	69.86	71.51	74.94	73.40	60.99	45.27	71.16	
Vertebral Column	84.52	82.90	81.61	81.61	80.00	81.61	81.61	79.35	79.03	83.23	78.06	86.13	80.97	76.45	77.74	80.00	
Vowel	97.17	79.29	98.38	78.99	83.84	75.15	78.99	46.46	77.88	68.08	90.51	94.65	73.94	51.62	56.57	63.94	
Wall Following Robot Navigation-1	+++	90.60	98.83	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	99.98	99.91	90.58	99.84
Wall Following Robot Navigation-2	+++	88.58	97.21	100.00	100.00	100.00	100.00	100.00	99.96	100.00	100.00	99.87	99.98	99.89	89.11	99.40	
Wall Following Robot Navigation-3	+++	68.07	88.18	99.65	97.62	99.29	99.65	98.81	97.73	99.58	99.52	97.54	99.78	88.23	52.46	—	
Wine Quality-red	70.73	54.66	63.98	60.73	67.23	59.98	60.73	58.79	63.48	58.16	69.29	67.73	59.10	57.79	54.72	57.60	
Wine Quality-white	+++	48.33	65.84	58.88	66.35	57.82	58.88	51.67	62.84	54.53	68.56	67.39	59.35	48.12	44.30	54.94	
Zoo	96.04	96.04	96.04	92.08	93.07	92.08	92.08	95.05	94.06	89.11	94.06	96.04	93.07	97.03	97.03	97.03	
Ortalama	85.21	78.77	83.41	81.93	82.86	82.66	82.68	79.05	80.62	81.24	86.61	86.55	82.81	79.28	74.13	82.65	
Global Rank	1	1	1	2	2	2	2	2	1	2	3	1	1	1	1	1	

Çizelge C.16: Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının temel öğrenicilere göre benchmark veri kümeleri üzerindeki doğruluk oranları

Veri kümesi	Temel Öğreniciler					
	k-EYK ⁵	NB ⁶	DYM ⁷	DA ⁸	KA	RF ⁹
Banknote Authentication	99.93±0.00	87.15±0.11	98.52±0.04	97.67±0.00	98.79±0.18	99.49±0.07
Blood Transfusion	77.72±0.15	76.74±0.13	76.20±0.00	77.41±0.27	77.76±0.28	78.30±0.34
Boston Housing2	96.44±0.00	96.44±0.00 ¹⁰	96.44±0.00	96.18±0.23 ¹¹	91.30±0.20	96.38±0.11
Breast Cancer	96.38±0.17	97.14±0.38	96.95±0.08	96.14±0.14	96.33±0.30	97.04±0.30
Breast Tissue	66.98±0.00	71.38±0.54	75.16±0.54	69.50±0.54	72.33±0.30	72.33±0.54
Cardiotocography-1	93.01±0.12	89.29±0.07 ¹⁰	90.73±0.04	88.55±0.21	94.37±0.24	95.48±0.17
Cardiotocography-2	76.84±0.15	73.35±0.24 ¹⁰	83.87±0.24	75.15±0.12	85.45±0.28	88.51±0.26
Climate Model	96.36±0.28	95.49±0.11	96.79±0.11	95.99±0.21	94.81±0.49	95.19±0.19

⁵ En yakın komşu sayısı k = 1

⁶ Dağılım türü değeri aksi belirtilmedikçe DistributionNames = “normal” dir.

⁷ Çekirdek fonksiyonu ; KernelFunction = “linear”

⁸ Diskriminant türü değeri aksi belirtilmedikçe DiscrimType = “linear”

⁹ OOBPrediction = “on”

¹⁰ DistributionNames = “kernel”

¹¹ DiscrimType = “pseudolinear”

Çizelge C.16: (devam) Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının temel öğrenicilere göre benchmark veri kümeleri üzerindeki doğruluk oranları

DNA	90.01±0.13	94.15±0.10 ¹²	96.25±0.10	95.42±0.11	94.95±0.14	96.56±0.05
Glass	76.01±0.71	70.87±1.08 ¹⁰	64.64±0.54	64.95±0.47	75.39±0.71	81.46±0.71
HTRU2	97.28±0.05	97.38±0.00	97.93±0.01	97.47±0.00	97.88±0.01	98.04±0.02
Ionosphere	94.11±0.16	92.69±0.16 ¹⁰	91.83±0.44	91.26±0.33 ¹¹	93.35±0.33	94.97±0.16
Leaf	65.29±0.78	73.14±0.61	80.49±0.34	82.84±0.34	69.71±1.06	79.02±0.17
Letter Recognition	97.15±0.03	66.24±0.05	85.69±0.04	70.28±0.05	92.16±0.21	96.91±0.03
Libras Movement	90.28±0.28	67.04±0.42	86.48±0.32	72.50±0.56	80.74±0.85	86.76±0.32
LSVT Voice Rehabilitation	91.80±0.46	87.83±0.46	93.65±0.79	94.71±2.79¹¹	91.27±1.59	93.39±0.46
Madelon	90.42±0.10	61.88±0.06	???	62.65±0.18	87.28±0.16	89.73±0.08
MAGIC Gamma Telescope	81.97±0.04	76.78±0.00	79.19±0.01	78.42±0.02	86.11±0.04	88.21±0.03
Parkinson Speech	67.82±0.36	59.68±0.31	65.45±0.48	64.33±0.19	72.72±0.39	73.37±0.33
QSAR biodegradation	84.23±0.14	77.98±0.30 ¹⁰	88.50±0.05	86.54±0.09	87.71±0.43	88.28±0.11
Satellite	91.55±0.06	79.77±0.03	87.35±0.11	84.29±0.03	91.09±0.10	92.19±0.08
Seeds	91.75±0.27	91.11±0.73	96.03±0.27	97.14±0.00	94.44±0.27	94.60±0.27

¹² DistributionNames = "mn"

Çizelge C.16: (devam) Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının temel öğrenicilere göre benchmark veri kümeleri üzerindeki doğruluk oranları

Shuttle	99.86±0.00	99.69±0.00 ¹⁰	98.47±0.01	95.03±0.01	99.98±0.00	99.99±0.00
Sonar	90.71±0.56	84.29±0.28 ¹⁰	83.17±0.00	82.53±0.73	89.74±0.73	89.74±0.27
Vehicle	68.40±0.18	53.82±0.84	80.69±0.07	78.61±0.24	74.98±0.25	76.20±0.71
Vertebral Column	85.48±0.85	80.65±0.32	85.91±0.19	84.62±1.04	83.98±0.67	84.95±0.49
Vowel	99.09±0.00	58.92±0.76	73.13±0.44	57.31±0.35	80.61±0.76	95.82±0.21
Wall Following Robot Navigation-1	98.85±0.08	90.70±0.04	98.11±0.04	70.87±0.04	100.00±0.00	100.00±0.00
Wall Following Robot Navigation-2	98.78±0.05	91.65±0.03	98.19±0.00	71.57±0.04	100.00±0.00	100.00±0.00
Wall Following Robot Navigation-3	95.51±0.02	63.39±0.08	73.78±0.10	66.12±0.11	99.49±0.02	99.58±0.02
Wine Quality-red	66.19±0.30	58.75±0.04	58.72±0.00	59.58±0.10	68.61±0.23	71.92±0.39
Wine Quality-white	67.03±0.15	52.66±0.07	52.50±0.03	54.62±0.10	66.01±0.43	71.57±0.10
Zoo	98.02±0.00	93.40±1.14 ¹²	96.37±0.57	93.07±0.00 ¹¹	92.08±0.00	97.36±0.57
Ortalama	87.31	79.13	85.22	80.40	87.32	89.80
Rank	8	0	6	3	2	17

Çizelge C.17: Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının temel öğrenicilere göre benchmark veri kümeleri üzerindeki gerçek ve tahmini hata oranları

Veri kümesi	k-EYK		NB		DYM		DA		KA		RF	
	Gerçek	ϵ_{min}	Gerçek	ϵ_{min}	Gerçek	ϵ_{min}	Gerçek	ϵ_{min}	Gerçek	ϵ_{min}	Gerçek	ϵ_{min}
Banknote Authentication	0.07±0.00	0.07±0.00	12.85±0.11	14.24±0.08	1.48±0.04	1.55±0.04	2.33±0.00	2.33±0.00	1.21±0.18	1.35±0.23	0.51±0.07	1.99±0.56
Blood	22.28±0.15	22.33±0.13	23.26±0.13	23.31±0.08	23.80±0.00	23.80±0.00	22.59±0.27	22.59±0.27	22.24±0.28	23.04±0.40	21.70±0.34	22.06±0.27
Boston Housing2	3.56±0.00	3.56±0.00	3.56±0.00	3.56±0.00	3.56±0.00	3.56±0.00	3.82±0.23	3.95±0.22	8.70±0.20	9.13±0.36	3.62±0.11	3.65±0.11
Breast Cancer	3.62±0.17	4.28±0.26	2.86±0.38	2.91±0.33	3.05±0.08	3.11±0.09	3.86±0.14	4.23±0.31	3.67±0.30	3.88±0.43	2.96±0.30	3.49±0.34
Breast Tissue	33.02±0.00	33.96±0.94	28.62±0.54	29.87±0.54	24.84±0.54	25.30±0.36	30.50±0.54	30.50±0.54	27.67±1.96	28.73±2.28	27.67±0.54	29.25±0.00
Cardiotocography -1	6.99±0.12	7.06±0.06	10.71±0.07	10.89±0.00	9.27±0.05	9.37±0.09	11.45±0.21	11.90±0.27	5.63±0.24	5.72±0.24	4.52±0.17	4.63±0.22
Cardiotocography -2	23.16±0.15	23.41±0.14	26.65±0.24	28.06±0.63	16.13±0.24	17.60±0.71	24.85±0.12	25.93±0.30	14.55±0.28	14.66±0.29	11.49±0.26	11.95±0.22
Climate Model	3.64±0.28	3.81±0.17	4.51±0.11	4.79±0.09	3.21±0.11	4.06±0.51	4.01±0.21	4.01±0.21	5.19±0.49	5.32±0.41	4.81±0.19	5.31±0.21
DNA	9.99±0.13	10.06±0.13	5.85±0.10	5.91±0.09	3.75±0.10	3.76±0.08	4.58±0.11	4.95±0.21	5.05±0.14	5.14±0.07	3.44±0.05	3.62±0.04
Glass	23.99±0.71	24.50±0.74	29.13±1.08	29.51±0.94	35.36±0.54	36.79±1.47	35.05±0.47	37.03±0.20	24.61±0.71	24.89±0.79	18.54±0.71	19.97±0.41
HTRU2	2.72±0.05	2.88±0.11	2.62±0.00	2.66±0.00	2.07±0.01	2.22±0.01	2.53±0.00	2.99±0.00	2.12±0.01	2.14±0.03	1.96±0.02	2.00±0.01
Ionosphere	5.89±0.16	6.27±0.34	7.31±0.16	7.80±0.65	8.17±0.44	9.72±0.43	8.74±0.33	9.61±0.46	6.65±0.33	6.77±0.42	5.03±0.16	5.15±0.02
Leaf	34.71±0.78	34.88±0.73	26.86±0.61	27.17±0.61	19.51±0.34	20.48±0.86	17.16±0.34	18.99±0.82	30.29±1.06	30.48±1.08	20.98±0.17	22.21±1.16
Letter Recognition	2.85±0.03	2.87±0.04	33.76±0.05	33.92±0.15	14.32±0.04	17.76±0.04	29.72±0.05	32.51±0.06	7.84±0.21	7.87±0.22	3.09±0.03	3.13±0.04

Çizelge C.17: (devam) Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının temel öğrencilere göre benchmark veri kümeleri üzerindeki gerçek ve tahmini hata oranları

Libras Movement	9.72±0.28	10.02±0.31	32.96±0.42	34.37±0.33	13.52±0.32	13.94±0.47	27.50±0.56	28.20±0.17	19.26±0.85	19.38±0.81	13.24±0.32	15.04±1.24
LSVT Voice Rehabilitation	8.20±0.46	8.79±0.76	12.17±0.46	12.70±0.79	6.35±0.79	6.99±1.27	5.29±2.79	5.32±2.75	8.73±1.59	10.27±0.38	6.61±0.46	7.10±0.30
Madelon	9.58±0.10	9.65±0.15	38.12±0.06	38.64±0.29	???	???	37.35±0.18	37.50±0.12	12.72±0.16	12.88±0.22	10.27±0.08	10.54±0.23
MAGIC Gamma Telescope	18.03±0.04	18.10±0.05	23.22±0.00	23.22±0.00	20.81±0.01	20.83±0.01	21.58±0.02	21.87±0.03	13.89±0.04	13.94±0.03	11.79±0.03	12.26±0.11
Parkinson Speech	32.18±0.36	32.52±0.29	40.32±0.31	40.85±0.43	34.55±0.48	35.50±0.59	35.67±0.19	36.60±0.15	27.28±0.39	27.50±0.37	26.63±0.33	26.84±0.36
QSAR biodegradation	15.77±0.14	15.86±0.17	22.02±0.30	22.98±0.44	11.50±0.05	12.14±0.39	13.46±0.09	14.91±0.29	12.29±0.43	12.39±0.48	11.72±0.11	12.47±0.32
Satellite	8.45±0.06	8.50±0.06	20.23±0.03	20.69±0.22	12.65±0.11	12.84±0.22	15.71±0.03	16.22±0.03	8.91±0.10	8.97±0.13	7.81±0.08	8.09±0.12
Seeds	8.25±0.27	9.96±0.54	8.89±0.73	10.95±0.48	3.97±0.27	5.40±0.73	2.86±0.00	3.99±1.08	5.56±0.27	6.56±0.51	5.40±0.27	7.30±0.73
Shuttle	0.14±0.00	0.54±0.24	0.31±0.00	0.33±0.00	1.53±0.01	1.65±0.02	4.97±0.01	5.09±0.09	0.13±0.00	0.10±0.07	0.01±0.00	0.14±0.00
Sonar	9.29±0.56	9.86±0.60	15.71±0.28	17.14±1.39	16.83±0.00	17.43±0.22	17.47±0.73	17.71±0.69	10.26±0.73	10.47±0.66	10.26±0.28	11.24±0.42
Vehicle	31.60±0.18	31.81±0.14	46.18±0.84	46.18±0.84	19.31±0.07	24.09±0.29	21.39±0.24	23.74±1.50	25.02±0.25	25.41±0.26	23.80±0.71	25.93±1.26
Vertebral Column	14.52±0.85	14.94±0.69	19.35±0.32	19.35±0.32	14.09±0.19	15.05±0.19	15.38±1.04	15.93±0.75	16.02±0.67	16.67±0.46	15.05±0.49	15.88±0.53
Vowel	0.91±0.00	1.15±0.24	41.08±0.76	42.19±0.79	26.87±0.44	31.41±0.27	42.69±0.35	43.00±0.29	19.39±0.76	19.66±0.70	4.18±0.21	5.43±0.56
Wall Following Robot Navigation-1	1.15±0.08	1.16±0.08	9.30±0.04	9.30±0.04	1.89±0.04	1.89±0.04	29.13±0.04	29.20±0.02	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00

Çizelge C.17: (devam) Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının temel öğrencilere göre benchmark veri kümeleri üzerindeki gerçek ve tahmini hata oranları

Wall Following Robot Navigation-2	1.22±0.05	1.56±0.03	8.35±0.03	8.35±0.03	1.81±0.00	1.83±0.00	28.43±0.04	28.43±0.04	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00
Wall Following Robot Navigation-3	4.49±0.02	4.83±0.01	36.61±0.08	37.09±0.25	26.22±0.10	32.69±0.19	33.88±0.11	35.74±2.08	0.51±0.02	1.15±0.04	0.42±0.02	1.44±0.02
Wine Quality-red	33.81±0.30	34.18±0.10	41.26±0.04	41.34±0.06	41.28±0.00	41.56±0.15	40.42±0.10	40.61±0.14	31.39±0.23	31.61±0.17	28.08±0.39	28.33±0.29
Wine Quality- white	32.97±0.15	33.10±0.06	47.34±0.07	47.60±0.18	47.50±0.03	48.21±0.19	46.38±0.10	46.98±0.21	33.99±0.43	34.04±0.42	28.43±0.10	28.58±0.06
Zoo	1.98±0.00	2.64±0.57	6.60±1.14	6.75±1.10	3.63±0.57	4.29±0.57	6.93±0.00	6.93±0.00	7.92±0.00	8.58±0.57	2.64±0.57	3.96±0.99
Ortalama	12.69	13.00	20.87	21.35	14.78	15.80	19.63	20.29	12.69	12.99	10.20	10.88

Çizelge C.18: Özniteliklerin bağımsızlığı varsayımına göre VIBES algoritmasının temel öğrencilere göre benchmark veri kümeleri üzerindeki doğruluk oranları

Veri kümesi	Temel Öğrenciler					
	k-EYK ⁵	NB ⁶	DYM ⁷	DA ⁸	KA	RF ⁹
Banknote Authentication	99.93±0.00	86.86±0.04	98.40±0.00	97.67±0.00	98.40±0.13	99.22±0.15
Blood Transfusion	76.92±0.08	76.20±0.00	76.25±0.08	77.36±0.28	77.09±0.28	76.83±0.20
Boston Housing2	96.44±0.00	96.31±0.11 ¹⁰	???	96.44±0.00¹¹	89.53±0.71	96.11±0.11
Breast Cancer	95.28±0.14	96.61±0.08	96.71±0.14	95.95±0.17	96.33±0.30	96.90±0.17
Breast Tissue	66.67±0.54	66.67±1.09	70.75±0.00	66.67±0.54	68.24±0.54	69.81±0.94
Cardiotocography-1	90.23±0.27	85.25±0.10 ¹⁰	90.04±0.05	87.69±0.10	92.30±0.23	94.59±0.17
Cardiotocography-2	73.69±0.23	57.01±0.26 ¹⁰	83.15±0.31	74.44±0.17	81.37±0.26	88.05±0.17
Climate Model	91.85±0.19	95.06±0.39	95.74±0.00	95.19±0.19	92.16±0.28	93.21±0.17
DNA	73.55±0.23	92.53±0.05 ¹²	???	95.46±0.13	91.90±0.11	95.06±0.10
Glass	73.83±0.93	64.33±0.27 ¹⁰	63.24±0.27	62.31±1.50	68.69±2.04	78.19±2.40
HTRU2	97.34±0.04	97.49±0.01	???	97.47±0.02	97.49±0.06	98.02±0.02
Ionosphere	91.26±0.33	92.21±0.44 ¹⁰	89.08±0.59	89.08±1.35 ¹¹	92.40±0.44	94.11±1.00
Leaf	60.88±0.51	59.80±0.45	80.10±0.45	80.98±0.74	61.76±1.06	75.98±1.11
Letter Recognition	95.89±0.05	64.33±0.11	???	70.17±0.03	86.35±0.12	96.42±0.05
Libras Movement	87.41±0.42	64.54±0.85	80.56±1.47	64.91±1.58	66.85±0.80	82.59±1.43

Çizelge C.18: (devam) Özniteliklerin bağımsızlığı varsayımına göre VIBES algoritmasının temel öğrenicilere göre benchmark veri kümeleri üzerindeki doğruluk oranları

LSVT Voice Rehabilitation	95.50±0.46	91.53±0.46	87.57±0.46	84.39±1.65 ¹¹	93.39±1.21	93.39±0.92
Madelon	64.53±0.42	68.57±0.20	???	69.67±0.10	74.73±0.49	75.83±0.45
MAGIC Gamma Telescope	78.17±0.04	74.23±0.03	???	78.42±0.01	81.76±0.35	87.93±0.14
Parkinson Speech	63.88±0.49	63.72±0.43	65.99±0.36	65.16±0.29	64.01±0.65	69.26±0.39
QSAR biodegradation	81.39±0.22	82.02±0.05 ¹⁰	87.39±0.28	85.78±0.09	80.63±0.52	87.11±0.58
Satellite	90.68±0.27	79.60±0.08	87.10±0.16	83.98±0.24	85.83±0.79	91.50±0.14
Seeds	89.84±0.55	90.63±0.27	94.44±0.99	96.67±0.00	92.70±0.99	93.65±0.99
Shuttle	99.82±0.01	??? ¹⁰	???	94.39±0.01	99.98±0.00	99.99±0.00
Sonar	83.17±0.48	88.14±0.56¹⁰	86.22±2.47	85.74±0.73	82.05±1.94	84.46±1.21
Vehicle	65.76±0.59	53.51±0.25	79.39±0.38	78.25±0.35	69.98±0.89	75.49±0.61
Vertebral Column	83.98±0.75	78.60±0.19	85.59±0.19	81.40±2.46	79.35±1.94	83.98±0.49
Vowel	98.89±0.36	56.80±0.25	73.00±0.38	54.21±0.56	73.20±1.55	94.95±0.36
Wall Following Robot Navigation-1	98.78±0.04	90.60±0.04	98.00±0.05	70.83±0.06	100.00±0.00	100.00±0.00
Wall Following Robot Navigation-2	97.26±0.07	89.08±0.13	98.04±0.05	71.31±0.06	100.00±0.00	100.00±0.00
Wall Following Robot Navigation-3	88.39±0.09	58.80±0.71	73.81±0.08	66.01±0.24	99.40±0.07	99.47±0.07

Çizelge C.18: (devam) Özniteliklerin bağımsızlığı varsayımına göre VIBES algoritmasının temel öğrencilere göre benchmark veri kümeleri üzerindeki doğruluk oranları

Wine Quality-red	60.04±0.70	57.24±0.26	58.79±0.17	59.08±0.20	59.91±0.92	71.50±0.94
Wine Quality-white	59.73±0.33	50.18±0.14	52.15±0.09	53.12±0.09	57.98±0.41	70.56±0.11
Zoo	98.02±0.00	89.44±0.57 ¹²	96.04±0.00	93.07±0.00 ¹¹	89.77±0.57	97.03±0.99
Ortalama	83.91	76.81	82.60	79.49	83.20	88.22
Rank	6	1	5	5	2	17

Çizelge C.19: Özniteliklerin bağımsızlığı varsayımına göre VIBES algoritmasının temel öğrencilere göre benchmark veri kümeleri üzerindeki gerçek ve tahmini hata oranları

Veri kümesi	k-EYK		NB		DYM		DA		KA		RF	
	Gerçek	ϵ_{min}	Gerçek	ϵ_{min}	Gerçek	ϵ_{min}	Gerçek	ϵ_{min}	Gerçek	ϵ_{min}	Gerçek	ϵ_{min}
Banknote Authentication	0.07±0.00	0.07±0.00	13.14±0.04	15.60±0.15	1.60±0.00	1.60±0.00	2.33±0.00	2.33±0.00	1.60±0.13	1.60±0.13	0.78±0.15	0.78±0.15
Blood	23.08±0.08	24.43±0.10	23.80±0.00	23.80±0.00	23.75±0.08	23.75±0.08	22.64±0.28	22.64±0.28	22.91±0.28	24.15±0.87	23.17±0.20	23.92±0.22
Boston Housing2	3.56±0.00	3.56±0.00	3.69±0.11	3.69±0.11	???	???	3.56±0.00	6.67±0.55	10.47±0.71	13.74±1.38	3.89±0.11	3.89±0.11
Breast Cancer	4.72±0.14	4.72±0.14	3.39±0.08	5.46±0.17	3.29±0.14	3.29±0.14	4.05±0.17	4.05±0.17	3.67±0.30	6.73±0.48	3.10±0.17	3.64±1.11
Breast Tissue	33.33±0.54	43.49±1.79	33.33±1.09	37.68±1.83	29.25±0.00	29.25±0.00	33.33±0.54	33.33±0.54	31.76±0.54	41.91±1.47	30.19±0.94	32.72±5.22
Cardiotocography -1	9.77±0.27	9.77±0.27	14.75±0.10	18.07±0.11	9.96±0.05	9.96±0.05	12.31±0.10	12.31±0.10	7.70±0.23	7.70±0.23	5.41±0.17	5.41±0.17
Cardiotocography -2	26.31±0.23	26.31±0.23	42.99±0.26	54.35±0.19	16.85±0.31	16.85±0.31	25.56±0.17	25.56±0.17	18.63±0.26	18.63±0.26	11.95±0.17	11.95±0.17
Climate Model	8.15±0.19	12.08±0.79	4.94±0.39	4.94±0.39	4.26±0.00	4.26±0.00	4.81±0.19	4.81±0.19	7.84±0.28	10.36±0.49	6.79±0.11	6.79±0.11
DNA	26.45±0.23	26.45±0.23	7.47±0.05	7.47±0.05	???	???	4.54±0.13	6.71±0.15	8.10±0.11	10.82±0.28	4.94±0.10	4.94±0.10
Glass	26.17±0.93	26.17±0.93	35.67±0.27	41.44±2.13	36.76±0.27	36.76±0.27	37.69±1.50	37.69±1.50	31.31±2.04	33.85±5.53	21.81±2.40	21.81±2.40
HTRU2	2.66±0.04	4.12±0.02	2.51±0.01	2.56±0.01	???	???	2.53±0.02	2.53±0.02	2.51±0.06	3.58±0.08	1.98±0.02	1.98±0.02
Ionosphere	8.74±0.33	16.30±0.40	7.79±0.44	10.06±1.90	10.92±0.59	12.06±0.82	10.92±1.35	13.68±1.59	7.60±0.44	12.92±1.55	5.89±1.00	6.46±1.90
Leaf	39.12±0.51	39.12±0.51	40.20±0.45	40.20±0.45	19.90±0.45	19.90±0.45	19.02±0.74	19.02±0.74	38.24±1.06	38.24±1.06	24.02±1.11	24.02±1.11
Letter Recognition	4.11±0.05	4.11±0.05	35.68±0.11	35.68±0.11	???	???	29.83±0.03	29.83±0.03	13.66±0.12	13.66±0.12	3.58±0.05	3.58±0.05

Çizelge C.19: (devam) Özniteliklerin bağımsızlığı varsayımına göre VIBES algoritmasının temel öğrencilere göre benchmark veri kümeleri üzerindeki gerçek ve tahmini hata oranları

Libras Movement	12.59±0.42	12.59±0.42	35.46±0.85	35.46±0.85	19.44±1.47	19.44±1.47	35.09±1.58	35.09±1.58	33.15±0.80	33.15±0.80	17.41±1.43	17.41±1.43
LSVT Voice Rehabilitation	4.50±0.46	21.64±1.02	8.47±0.46	22.86±0.17	12.43±0.46	19.87±2.29	15.61±1.65	23.05±0.36	6.61±1.21	21.75±0.60	6.61±0.92	18.27±0.78
Madelon	35.47±0.42	35.47±0.42	31.43±0.20	38.94±0.11	???	???	30.33±0.10	39.29±0.39	25.27±0.49	25.27±0.49	24.17±0.45	35.76±1.29
MAGIC Gamma Telescope	21.83±0.04	21.83±0.04	25.77±0.03	27.38±0.00	???	???	21.58±0.01	21.58±0.01	18.24±0.35	18.24±0.35	12.08±0.14	12.08±0.14
Parkinson Speech	36.12±0.49	37.04±1.13	36.28±0.43	41.76±0.20	34.01±0.36	36.46±0.32	34.84±0.29	39.39±0.37	35.99±0.65	40.06±0.48	30.74±0.39	30.74±0.39
QSAR biodegradation	18.61±0.22	19.91±0.33	17.98±0.05	26.10±1.08	12.61±0.28	12.61±0.28	14.22±0.09	14.22±0.09	19.37±0.52	20.67±1.84	12.89±0.58	12.89±0.58
Satellite	9.32±0.27	9.32±0.27	20.40±0.08	20.40±0.08	12.90±0.16	12.90±0.16	16.02±0.24	16.02±0.24	14.17±0.79	14.17±0.79	8.50±0.14	8.50±0.14
Seeds	10.16±0.55	10.72±1.33	9.37±0.27	10.29±1.33	5.56±0.99	5.56±0.99	3.33±0.00	3.33±0.00	7.30±0.99	7.30±0.99	6.35±0.99	6.35±0.99
Shuttle	0.18±0.01	0.18±0.01	???	???	???	???	5.61±0.01	5.62±0.00	0.02±0.00	0.02±0.00	0.01±0.00	0.01±0.00
Sonar	16.83±0.48	16.83±0.48	11.86±0.56	25.54±0.85	13.78±2.47	26.56±1.29	14.26±0.73	25.99±0.46	17.95±1.94	29.80±0.82	15.55±1.21	15.55±1.21
Vehicle	34.24±0.59	34.24±0.59	46.49±0.25	56.37±1.17	20.61±0.38	20.61±0.38	21.75±0.35	21.75±0.35	30.02±0.89	30.02±0.89	24.51±0.61	24.51±0.61
Vertebral Column	16.02±0.75	16.02±0.75	21.40±0.19	21.40±0.19	14.41±0.19	15.48±0.32	18.60±2.46	18.60±2.46	20.65±1.94	20.65±1.94	16.02±0.49	16.02±0.49
Vowel	1.11±0.36	1.11±0.36	43.20±0.25	43.20±0.25	27.00±0.38	27.00±0.38	45.79±0.56	45.79±0.56	26.80±1.55	26.80±1.55	5.05±0.36	5.05±0.36
Wall Following Robot Navigation-1	1.22±0.04	1.22±0.04	9.40±0.04	9.40±0.04	2.00±0.05	2.00±0.05	29.17±0.06	29.17±0.06	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00

Çizelge C.19: (devam) Özniteliklerin bağımsızlığı varsayımına göre VIBES algoritmasının temel öğrencilere göre benchmark veri kümeleri üzerindeki gerçek ve tahmini hata oranları

Wall Following Robot Navigation-2	2.74±0.07	2.74±0.07	10.92±0.13	10.92±0.13	1.96±0.05	1.96±0.05	28.69±0.06	32.21±3.11	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00
Wall Following Robot Navigation-3	11.61±0.09	11.61±0.09	41.20±0.71	48.79±3.46	26.19±0.08	26.19±0.08	33.99±0.24	33.99±0.24	0.60±0.07	0.60±0.07	0.53±0.07	0.53±0.07
Wine Quality-red	39.96±0.70	39.96±0.70	42.76±0.26	46.00±0.63	41.21±0.17	44.58±0.27	40.92±0.20	40.92±0.20	40.09±0.92	40.88±2.15	28.50±0.94	28.50±0.94
Wine Quality-white	40.27±0.33	40.27±0.33	49.82±0.14	52.56±0.60	47.85±0.09	47.85±0.09	46.88±0.09	46.88±0.09	42.02±0.41	42.02±0.41	29.44±0.11	29.44±0.11
Zoo	1.98±0.00	1.98±0.00	10.56±0.57	10.56±0.57	3.96±0.00	3.96±0.00	6.93±0.00	6.93±0.00	10.23±0.57	10.23±0.57	2.97±0.99	2.97±0.99
Ortalama	16.09	17.44	23.19	26.53	17.40	18.49	20.51	21.85	16.80	18.77	11.78	12.62

Çizelge C.20: Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmaları ile VIBES algoritmasının karşılaştırma sonuçları (birinci grup)

Veri kümesi	VIBES ¹³	RseslibKnn	Rotation Forest ¹	REPTree	LDA	Furthest Centroid ND ¹	Random Committee ¹	SGD	END ¹	MultiBoostAB ¹	Real Adaboost ¹	Stacking ³	Random Forest	Ordinal Class Classifier ¹	Classification Via Regression ¹	AdaBoostM1 ¹
Banknote Authentication	99.49	100.00	100.00	98.03	97.67	99.34	99.27	98.47	99.34	99.27	99.56	99.78	99.27	99.27	99.27	99.42
Blood Transfusion	78.30	73.93	76.47	79.28	76.74	73.93	74.06	75.94	74.06	74.20	73.93	76.74	73.93	73.66	73.26	73.66
Boston Housing2	96.38	94.47	94.47	84.78	96.44	93.87	95.85	–	94.47	95.85	–	71.74	95.45	94.66	95.45	95.85
Breast Cancer	97.04	96.42	97.28	94.42	–	96.42	96.71	96.57	96.42	96.28	96.42	96.42	96.57	96.42	96.85	96.57
Breast Tissue	72.33	65.09	77.36	72.64	65.09	77.36	74.53	–	72.64	72.64	–	73.58	71.70	69.81	70.75	70.75
Cardiotocography-1	95.48	92.14	93.84	92.99	87.30	94.73	95.01	–	95.06	94.78	–	94.78	94.78	95.20	94.83	94.68

¹³ Temel Öğrenici olarak Random Forest algoritması seçildi.

Çizelge C.20: (devam) Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmaları ile VIBES algoritmasının karşılaştırma sonuçları (birinci grup)

Cardiotocography-2	88.51	82.36	86.69	82.31	74.27	88.43	88.85	–	88.57	88.24	–	88.66	88.48	88.52	87.91	87.49
Climate Model	95.19	93.15	92.22	91.48	95.56	94.07	93.52	96.48	93.52	93.89	93.70	94.44	93.70	93.70	93.70	93.89
DNA	96.56	90.99	95.61	92.81	94.85	95.70	95.64	–	95.79	95.32	–	96.05	94.82	94.95	95.54	94.66
Glass	81.46	77.57	76.64	62.15	65.42	78.50	80.37	–	82.24	78.97	–	77.10	80.37	80.37	78.97	79.91
HTRU2	98.04	97.93	98.02	97.78	97.46	98.06	98.05	97.88	98.07	98.12	98.07	97.80	98.08	98.09	98.04	98.01
Ionosphere	94.97	92.02	94.59	88.89	86.32	93.45	93.45	86.04	93.45	93.45	94.02	92.88	93.16	94.02	93.16	93.16
Leaf	79.02	73.24	82.35	58.24	81.18	72.35	77.94	–	79.12	76.76	–	74.71	76.76	74.71	75.00	78.53
Letter Recognition	96.91	97.20	+++	84.26	70.19	95.52	+++	–	?	96.36	–	96.98	96.42	+++	96.34	96.36
Libras Movement	86.76	83.61	88.61	56.67	67.50	83.33	83.06	–	83.89	83.33	–	85.56	82.50	80.28	78.89	83.89
LSVT Voice Rehabilitation	93.39	77.78	82.54	80.95	–	84.13	84.13	83.33	84.13	81.75	85.71	76.19	83.33	83.33	84.13	84.13
Madelon	89.73	72.35	60.35	73.40	53.75	63.50	65.60	53.25	63.50	62.95	67.00	66.40	64.55	65.35	63.85	64.05
MAGIC Gamma Telescope	88.21	85.32	87.56	85.01	78.41	87.97	88.13	79.15	88.11	88.16	88.38	87.04	88.04	88.00	88.13	88.08
Parkinson Speech	73.37	69.90	70.29	61.63	62.12	68.75	69.81	64.62	69.81	68.46	70.87	65.58	69.62	70.48	70.87	70.10
QSAR biodegradation	88.28	85.40	87.30	81.90	85.97	86.92	86.54	86.82	86.07	87.96	87.30	83.98	85.97	87.01	86.64	86.45
Satellite	92.19	91.14	91.36	86.25	84.09	91.61	91.90	–	91.76	91.66	–	92.37	91.66	91.61	91.55	91.73
Seeds	94.60	93.33	95.24	90.00	96.67	93.81	93.81	–	93.33	94.76	–	92.38	94.29	92.86	93.33	93.81

Çizelge C.20: (devam) Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmaları ile VIBES algoritmasının karşılaştırma sonuçları (birinci grup)

Shuttle	99.99	99.95	99.98	99.95	94.39	99.99	99.99	–	99.99	99.99	–	99.99	99.99	99.99	99.99	99.99
Sonar	89.74	85.10	84.62	75.00	74.52	84.13	85.58	79.33	84.13	82.21	86.54	86.06	85.10	81.25	84.62	84.13
Vehicle	76.20	70.80	78.84	73.17	78.13	75.30	75.53	–	74.82	75.30	–	75.41	76.00	74.82	75.77	75.06
Vertebral Column	84.95	81.94	86.45	80.32	84.19	86.45	84.84	84.19	85.16	85.48	83.23	85.16	83.55	84.84	83.87	83.23
Vowel	95.82	98.08	97.98	67.37	54.14	92.53	96.06	–	94.95	96.16	–	98.28	95.66	92.22	94.44	95.66
Wall Following Robot Navigation-1	100.00	99.01	98.88	100.00	70.77	99.98	100.00	–	100.00	100.00	–	100.00	100.00	99.98	100.00	100.00
Wall Following Robot Navigation-2	100.00	97.69	98.85	100.00	71.32	99.98	100.00	–	100.00	100.00	–	100.00	100.00	99.98	100.00	100.00
Wall Following Robot Navigation-3	99.58	94.06	97.12	99.58	65.74	99.60	99.62	–	99.58	99.62	–	99.71	99.67	99.58	99.51	99.52
Wine Quality-red	71.92	70.36	70.92	57.79	59.54	70.98	70.92	–	71.11	71.36	–	69.67	70.86	69.54	70.17	70.67
Wine Quality-white	71.57	70.01	70.76	55.80	53.14	69.72	70.66	–	70.52	70.25	–	70.56	70.36	70.03	70.03	70.29
Zoo	97.36	95.05	95.05	89.11	96.04	97.03	96.04	–	97.03	97.03	–	95.05	97.03	96.04	97.03	97.03
Ortalama	89.80	86.28	87.76	81.64	78.03	87.50	87.67	83.24	87.52	87.59	86.52	86.70	87.63	86.89	87.33	87.60
Global Rank	14	2	6	3	2	3	4	1	4	4	1	4	3	1	3	3

Çizelge C.21: Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmaları ile VIBES algoritmasının karşılaştırma sonuçları (ikinci grup)

Veri kümesi	Bagging ¹	Grading ³	MultiClass Classifier ¹	Multi Scheme ⁴	Class Balanced ND ¹	Data Near Balanced ND ¹	Random Pair ND ¹	Weighted Instances Handler Wrapper ¹	QDA	IBk	FURIA	Iterative Classifier Optimizer ¹	Simple Cart	Input Mapped Classifier	RoughSet	BFTree
Banknote Authentication	99.27	99.42	99.27	99.34	99.34	99.34	99.34	99.27	98.40	99.85	99.20	99.49	98.25	98.54	98.83	98.40
Blood Transfusion	75.94	76.87	73.93	73.93	73.93	73.93	73.93	73.66	77.94	69.92	77.41	72.99	77.27	76.87	67.51	78.61
Boston Housing2	95.45	95.65	95.65	95.85	90.51	89.92	92.09	95.65	82.02	82.81	83.60	+++	84.39	89.53	93.28	84.98
Breast Cancer	96.42	96.42	96.57	96.57	96.42	96.42	96.42	96.42	–	95.42	95.14	96.42	94.99	94.13	–	94.99
Breast Tissue	71.70	71.70	74.53	70.75	69.81	67.92	70.75	73.58	69.81	71.70	63.21	72.64	68.87	66.04	66.04	68.87
Cardiotocography-1	94.68	94.50	95.16	94.68	94.45	95.01	94.73	94.83	83.44	91.16	93.74	+++	92.85	93.23	89.75	92.57
Cardiotocography-2	87.86	87.58	88.48	87.96	87.96	87.72	88.29	88.33	69.38	78.27	83.49	+++	82.83	83.21	78.41	83.02
Climate Model	93.33	94.07	93.70	93.33	94.07	94.07	94.07	93.70	92.96	89.07	94.63	93.89	91.85	92.59	91.85	91.30

Çizelge C.21: (devam) Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmaları ile VIBES algoritmasının karşılaştırma sonuçları (ikinci grup)

DNA	95.42	95.54	95.67	95.14	95.79	95.17	95.29	95.14	90.43	74.64	94.38	+++	92.84	92.12	+++	92.40
Glass	79.44	77.57	80.37	80.84	80.37	79.91	78.50	80.37	57.94	70.56	74.77	81.31	70.56	65.89	71.50	71.50
HTRU2	98.07	97.98	98.09	97.85	98.06	98.06	98.06	98.09	96.73	97.13	97.87	+++	97.92	97.91	97.60	97.86
Ionosphere	94.02	94.30	94.02	94.02	93.45	93.45	93.45	94.02	89.17	87.18	91.45	92.88	88.32	88.89	84.62	89.17
Leaf	78.24	79.71	76.18	77.65	74.12	74.41	72.94	78.24	64.71	73.53	65.00	+++	63.53	61.18	64.12	62.06
Letter Recognition	???	+++	???	96.44	95.90	95.77	95.59	96.42	88.65	95.97	+	+++	86.97	88.12	+++	86.36
Libras Movement	82.22	85.28	82.22	83.06	81.67	81.11	81.11	83.33	86.11	85.83	64.44	+++	66.67	69.72	+++	62.78
LSVT Voice Rehabilitation	84.92	80.95	83.33	83.33	84.13	84.13	84.13	83.33	-	73.81	83.33	86.51	82.54	73.81	80.16	80.16
Madelon	63.60	66.80	65.35	63.60	63.50	63.50	63.50	65.35	52.00	54.15	72.75	+++	79.65	70.35	+++	76.20
MAGIC Gamma Telescope	87.97	87.40	88.00	88.03	87.97	87.97	87.97	88.00	78.44	80.94	84.69	+++	85.03	85.06	77.41	84.39
Parkinson Speech	68.37	68.94	70.48	69.42	68.75	68.75	68.75	70.48	56.25	63.65	65.10	70.38	64.81	62.98	68.17	63.85
QSAR biodegradation	86.73	87.58	87.01	87.01	86.92	86.92	86.92	87.01	72.23	84.46	84.36	+++	81.14	82.37	83.32	81.71
Satellite	91.73	91.89	91.84	91.70	91.50	91.52	91.72	91.93	85.73	90.26	89.00	+++	87.04	86.62	+++	86.68
Seeds	93.33	94.76	93.33	94.29	93.33	94.29	93.81	93.81	95.24	94.29	93.81	92.86	90.48	91.90	92.38	93.33
Shuttle	99.99	99.99	99.99	99.99	99.99	99.99	99.99	99.99	83.20	99.94	99.98	+++	99.97	99.97	99.97	99.97
Sonar	81.73	85.10	81.25	84.62	84.13	84.13	84.13	81.25	77.40	86.54	78.85	85.58	71.15	71.15	72.60	71.15

Çizelge C.21: (devam) Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmaları ile VIBES algoritmasının karşılaştırma sonuçları (ikinci grup)

Vehicle	75.18	76.24	74.40	76.00	77.07	77.07	74.82	74.94	85.34	71.28	71.04	+++	70.92	49.41	71.51	69.62
Vertebral Column	84.84	84.84	84.84	83.23	86.45	86.45	86.45	84.84	82.90	81.61	84.52	84.84	82.90	81.61	80.00	81.61
Vowel	95.56	96.36	94.55	95.35	93.74	92.93	93.43	95.35	79.29	98.38	77.78	+++	77.07	78.99	83.84	75.15
Wall Following Robot Navigation-1	100.00	100.00	100.00	100.00	99.98	99.98	100.00	100.00	90.60	98.83	100.00	100.00	100.00	100.00	100.00	100.00
Wall Following Robot Navigation-2	100.00	100.00	100.00	100.00	99.96	99.96	100.00	100.00	88.58	97.21	100.00	100.00	100.00	100.00	100.00	100.00
Wall Following Robot Navigation-3	99.51	99.58	99.62	99.58	99.63	99.63	99.49	99.63	68.07	88.18	99.65	+++	99.41	99.65	97.62	99.29
Wine Quality-red	71.04	67.35	71.29	69.92	71.11	70.54	70.73	69.92	54.66	63.98	59.41	+++	60.79	60.73	67.23	59.98
Wine Quality-white	69.82	68.58	70.54	70.25	69.89	69.58	69.52	70.38	48.33	65.84	53.29	+++	58.37	58.88	66.35	57.82
Zoo	95.05	96.04	96.04	95.05	96.04	97.03	96.04	95.05	96.04	96.04	89.11	96.04	92.08	92.08	93.07	92.08
Ortalama	87.23	87.47	87.37	87.54	87.27	87.17	87.15	87.65	78.77	83.41	83.28	88.39	83.07	81.93	82.86	82.66
Global Rank	3	3	3	3	2	2	4	3	1	1	2	2	2	2	2	2

Çizelge C.22: Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmaları ile VIBES algoritmasının karşılaştırma sonuçları (üçüncü grup)

Veri kümesi	J48	LADTree	Random Tree	CDT	Attribute Selected Classifier ¹	Vote ⁴	PART
Banknote Authentication	98.54	97.38	98.54	97.81	93.73	98.91	98.98
Blood Transfusion	76.87	78.34	70.32	78.21	76.34	75.80	77.01
Boston Housing2	89.53	72.92	90.32	76.28	95.85	94.47	88.74
Breast Cancer	94.13	95.71	94.56	94.56	96.42	95.99	94.85
Breast Tissue	66.04	60.38	62.26	69.81	73.58	73.58	67.92
Cardiotocography-1	93.23	92.00	91.91	92.66	94.40	93.13	93.13
Cardiotocography-2	83.21	79.92	79.02	81.51	85.42	86.17	83.44
Climate Model	92.59	93.70	91.11	91.11	93.70	93.70	92.96
DNA	92.12	92.81	76.02	92.75	92.78	95.04	92.72
Glass	65.89	67.29	68.69	63.08	78.97	74.77	67.76
HTRU2	97.91	97.88	96.91	97.80	97.91	97.96	97.83
Ionosphere	88.89	89.74	86.04	89.17	92.88	92.88	90.03
Leaf	61.18	54.41	57.94	57.94	76.47	77.65	64.71

Çizelge C.22: (devam) Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmaları ile VIBES algoritmasının karşılaştırma sonuçları (üçüncü grup)

Letter Recognition	88.12	51.33	86.03	83.77	95.84	95.81	88.62
Libras Movement	69.72	53.06	67.22	55.28	79.72	83.89	70.56
LSVT Voice Rehabilitation	73.81	78.57	70.63	80.95	82.54	73.02	73.81
Madelon	70.35	72.65	54.20	76.00	79.10	64.05	59.50
MAGIC Gamma Telescope	85.06	82.97	82.13	85.05	83.88	84.90	85.46
Parkinson Speech	62.98	63.17	60.87	62.31	68.08	66.25	59.71
QSAR biodegradation	82.37	83.13	82.46	83.13	85.78	85.50	85.21
Satellite	86.62	82.75	84.82	85.78	91.78	90.38	87.13
Seeds	91.90	91.90	90.00	89.52	94.29	94.76	92.86
Shuttle	99.97	99.86	99.97	99.94	99.85	99.96	99.97
Sonar	71.15	78.85	73.56	74.04	77.40	83.17	80.29
Vehicle	74.35	67.85	69.86	69.86	71.51	74.94	73.40
Vertebral Column	81.61	79.35	79.03	83.23	78.06	86.13	80.97
Vowel	78.99	46.46	77.88	68.08	90.51	94.65	73.94
Wall Following Robot Navigation-1	100.00	100.00	100.00	100.00	100.00	100.00	99.98
Wall Following Robot Navigation-2	100.00	100.00	99.96	100.00	100.00	99.87	99.98

Çizelge C.22: (devam) Benchmark veri kümeleri üzerinde en iyi öğrenmeyi gerçekleştiren Makine Öğrenmesi algoritmaları ile VIBES algoritmasının karşılaştırma sonuçları (üçüncü grup)

Wall Following Robot Navigation-3	99.65	98.81	97.73	99.58	99.52	97.54	99.78
Wine Quality-red	60.73	58.79	63.48	58.16	69.29	67.73	59.10
Wine Quality-white	58.88	51.67	62.84	54.53	68.56	67.39	59.35
Zoo	92.08	95.05	94.06	89.11	94.06	96.04	93.07
Ortalama	82.68	79.05	80.62	81.24	86.61	86.55	82.81
Global Rank	2	2	1	2	2	1	1

Çizelge C.23: Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının (Temel Öğrenici olarak RF algoritması seçildi) topluluk oluşturma yöntemi için kullandığı ileri yönlü arama ve bu aramanın optimize edilmiş halinin karşılaştırması

Veri kümesi	Optimize Edilmemiş İleri Yönlü Arama				Optimize Edilmiş İleri Yönlü Arama			
	Doğruluk Oranı	Hata Oranı	ϵ_{\min}	İterasyon Sayısı	Doğruluk Oranı	Hata Oranı	ϵ_{\min}	İterasyon Sayısı
Banknote Authentication	99.4169	0.5831	0.5831	25	99.4169	0.5831	0.5831	25
Blood Transfusion	78.0749	21.9251	23.7968	25	78.0749	21.9251	23.7968	25
Boston Housing2	96.0474	3.9526	4.1502	361	96.0474	3.9526	4.1502	304
Breast Cancer	97.1388	2.8612	3.2631	100	97.1388	2.8612	3.2631	60
Breast Tissue	71.6981	28.3019	30.4997	100	71.6981	28.3019	30.4997	90
Cardiotocography-1	95.2023	4.7977	5.0809	484	95.2023	4.7977	5.0809	462
Cardiotocography-2	88.8053	11.1947	12.8756	484	88.8053	11.1947	12.8756	242
Climate Model	95.1852	4.8148	5.1852	361	95.1852	4.8148	5.1852	95
DNA	96.4846	3.5154	4.5848	32761	96.4846	3.5154	4.5848	4706
Glass	81.3084	18.6916	22.1638	100	81.3084	18.6916	22.1638	80
HTRU2	98.0836	1.9164	2.2225	81	98.0836	1.9164	2.2225	54
Ionosphere	95.4416	4.5584	5.7513	1225	95.4416	4.5584	5.7513	700
Leaf	78.2353	21.7647	24.2474	225	78.2353	21.7647	24.2474	135

Çizelge C.23: (devam) Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının (Temel Öğrenici olarak RF algoritması seçildi) topluluk oluşturma yöntemi için kullandığı ileri yönlü arama ve bu aramanın optimize edilmiş halinin karşılaştırması

Letter Recognition	96.9550	3.0450	3.6025	289	96.9550	3.0450	3.6025	170
Libras Movement	86.6667	13.3333	16.2200	8281	86.6667	13.3333	16.2200	1274
LSVT Voice Rehabilitation	92.0635	7.9365	11.0371	96721	92.0635	7.9365	11.0371	10574
Madelon	89.8500	10.1500	11.4363	251001	89.8500	10.1500	11.4363	6012
MAGIC Gamma Telescope	88.2282	11.7718	12.0643	121	88.2282	11.7718	12.0643	66
Parkinson Speech	74.2308	25.7692	28.9850	729	74.2308	25.7692	28.9850	405
QSAR biodegradation	88.2464	11.7536	11.7536	1764	88.2464	11.7536	11.7536	378
Satellite	92.1834	7.8166	8.3944	1369	92.1834	7.8166	8.3944	333
Seeds	94.2857	5.7143	5.7143	64	94.2857	5.7143	5.7143	32
Shuttle	99.9879	0.0121	0.0121	100	99.9879	0.0121	0.0121	50
Sonar	90.3846	9.6154	12.7949	3721	90.3846	9.6154	12.7949	1098
Vehicle	76.5957	23.4043	23.4043	361	76.5957	23.4043	23.4043	133
Vertebral Column	83.2258	16.7742	16.7742	49	83.2258	16.7742	16.7742	35
Vowel	94.7475	5.2525	5.9580	121	94.7475	5.2525	5.9580	66
Wall Following Robot Navigation-1	100.0000	0.0000	0.0002	9	100.0000	0.0000	0.0002	9
Wall Following Robot Navigation-2	100.0000	0.0000	0.0002	25	100.0000	0.0000	0.0002	25

Çizelge C.23: (devam) Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının (Temel Öğrenici olarak RF algoritması seçildi) topluluk oluşturma yöntemi için kullandığı ileri yönlü arama ve bu aramanın optimize edilmiş halinin karşılaştırması

Wall Following Robot Navigation-3	99.4685	0.5315	0.5315	625	99.4685	0.5315	0.5315	175
Wine Quality-red	71.6698	28.3302	29.2168	144	71.6698	28.3302	29.2168	120
Wine Quality-white	71.4169	28.5831	29.6916	144	71.4169	28.5831	29.6916	144
Zoo	97.0297	2.9703	2.9703	289	97.0297	2.9703	2.9703	119
Ortalama	89.6472	10.3528	11.6995	12189.67	89.6472	10.3528	11.6995	854.42

Çizelge C.24: Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının (Temel Öğrenici olarak RF algoritması seçildi) topluluk oluşturma yöntemi için kullandığı optimize edilmiş ileri yönlü arama ve Genetik Algoritmalar kullanılarak yapılan arama sonuçlarının karşılaştırması

Veri kümesi	Optimize Edilmiş İleri Yönlü Arama					Genetik Algoritmalar					
	Doğruluk Oranı	Hata Oranı	ϵ_{min}	İterasyon Sayısı	Geçen Süre (sn)	Doğruluk Oranı	Hata Oranı	ϵ_{min}	Nesil Sayısı	Fonksiyon Değerlendirme Sayısı	Geçen Süre (sn)
Banknote Authentication	99.4169	0.5831	0.5831	15	0.058685	99.3440	0.5831	0.5831	51	2600	10.191476
Blood Transfusion	78.7433	21.2567	23.2620	20	0.041674	78.7433	21.2567	23.2620	51	2600	6.972463
Boston Housing2	96.2451	3.7549	4.6074	304	0.293662	96.2451	3.7549	4.2001	51	10400	25.550034
Breast Cancer	97.1388	2.8612	3.7744	80	0.172681	97.2818	2.7182	3.3261	51	10400	31.981797
Breast Tissue	71.6981	28.3019	29.9327	60	0.099794	73.5849	26.4151	30.2501	52	10600	21.142529
Cardiotocography-1	95.5786	4.4214	4.8835	286	1.345274	95.4845	4.5155	4.8795	56	11400	73.454431
Cardiotocography-2	88.5230	11.4770	11.9178	154	0.872638	88.5701	11.4299	11.8320	56	11400	73.737469
Climate Model	95.3704	4.6296	5.0663	133	0.333069	95.3704	4.6296	5.0663	55	11200	31.286147
DNA	96.4532	3.5468	4.6822	3077	24.597772	96.1707	3.8293	4.6437	74	15000	196.253851
Glass	80.8411	19.1589	22.8847	80	0.111624	81.3084	18.6916	21.5332	55	11200	20.856240
HTRU2	98.0724	1.9276	2.0031	54	1.578518	98.0724	1.9276	2.0031	51	10400	413.077772
Ionosphere	95.4416	4.5584	6.0113	350	0.285617	94.8718	5.1282	5.7202	55	11200	15.585612
Leaf	78.5294	21.4706	23.5838	135	0.199834	78.5294	21.4706	23.5838	54	11000	29.223677

Çizelge C.24: (devam) Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının (Temel Öğrenici olarak RF algoritması seçildi) topluluk oluşturma yöntemi için kullandığı optimize edilmiş ileri yönlü arama ve Genetik Algoritmalar kullanılarak yapılan arama sonuçlarının karşılaştırması

Letter Recognition	96.9200	3.0800	3.5672	187	6.386031	96.9550	3.0450	3.5999	55	11200	555.831321
Libras Movement	86.3889	13.6111	18.3333	1183	1.163846	84.4444	15.5556	16.1563	80	16200	41.785495
LSVT Voice Rehabilitation	92.8571	7.1429	10.6320	11196	22.502176	85.7143	14.2857	10.5102	107	21600	49.333968
Madelon	89.8500	10.1500	11.4087	8016	35.855803	86.9000	13.1000	11.2286	221	44400	358.011738
MAGIC Gamma Telescope	88.2913	11.7087	12.0107	66	0.767842	88.2913	11.7087	12.0107	51	10400	345.466452
Parkinson Speech	73.4615	26.5385	29.2103	351	0.402598	74.2308	25.7692	29.3133	65	13200	43.024215
QSAR biodegradation	88.2464	11.7536	12.4058	462	0.647448	88.1517	11.8483	12.2354	68	13800	20.551912
Satellite	92.2766	7.7234	8.2387	259	2.594841	92.1834	7.8166	8.3004	54	11000	200.114083
Seeds	94.2857	5.7143	6.6667	40	0.053766	94.2857	5.7143	6.6667	51	10400	13.948776
Shuttle	99.9879	0.0121	0.0138	60	2.067241	99.9879	0.0121	0.0191	51	10400	1113.828880
Sonar	87.9808	12.0192	15.1283	976	2.046921	88.4615	11.5385	15.4470	59	12000	15.694298
Vehicle	75.2955	24.7045	25.3102	133	0.185382	75.4137	24.5863	26.0417	58	11800	16.481717
Vertebral Column	84.8387	15.1613	16.4453	35	0.046664	84.8387	15.1613	16.4453	51	10400	11.541295
Vowel	95.9596	4.0404	4.9275	66	0.100759	95.9596	4.0404	4.9275	51	10400	25.194149
Wall Following Robot Navigation-1	100.0000	0.0000	0.0002	9	0.117467	100.0000	0.0000	0.0002	51	2600	37.719194

Çizelge C.24: (devam) Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının (Temel Öğrenici olarak RF algoritması seçildi) topluluk oluşturma yöntemi için kullandığı optimize edilmiş ileri yönlü arama ve Genetik Algoritmalar kullanılarak yapılan arama sonuçlarının karşılaştırması

Wall Following Robot Navigation-2	100.0000	0.0000	0.0002	25	0.114239	100.0000	0.0000	0.0002	51	2600	18.476143
Wall Following Robot Navigation-3	99.5418	0.4582	0.4582	125	0.829448	99.5418	0.4582	0.4582	62	12600	201.871010
Wine Quality-red	72.0450	27.9550	28.6374	72	0.236844	72.1701	27.8299	28.9138	53	10800	55.684598
Wine Quality-white	71.6823	28.3177	29.2932	96	0.389237	71.6823	28.3177	29.2932	52	10600	121.288167
Zoo	98.0198	1.9802	1.9802	187	0.286016	98.0198	1.9802	2.5171	52	10600	18.799865
Ortalama	89.6963	10.3036	11.4503	857.3333	3.2359	89.4184	10.5793	11.3626	62.2727	11709.09	127.6958
Rank			24						24		

Çizelge C.25: Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının (Temel Öğrenici olarak RF algoritması seçildi) özniteliklerin enformasyon ölçümü yöntemlerine göre karşılaştırma sonuçları (deneyler 3 kere tekrarlandı)

Veri kümesi	Relief Algoritması				Shannon Enformasyon Algoritması			
	Doğruluk Oranı	Hata Oranı	ϵ_{\min}	İterasyon Sayısı	Doğruluk Oranı	Hata Oranı	ϵ_{\min}	İterasyon Sayısı
Banknote Authentication	99.3926±0.08	0.6074±0.08	0.6074±0.08	20.00±5.00	99.3683±0.04	0.6317±0.04	0.6317±0.04	15.00±0.00
Blood Transfusion	78.3868±0.20	21.6132±0.20	23.6631±0.27	20.00±0.00	78.2531±0.08	21.7469±0.08	23.2040±0.26	25.00±0.00
Boston Housing2	96.3109±0.11	3.6891±0.11	4.4478±0.20	316.67±43.88	95.7181±0.11	4.2819±0.11	5.0183±0.21	202.67±54.85
Breast Cancer	96.9003±0.08	3.0997±0.08	3.5459±0.19	73.33±23.09	97.4249±0.14	2.5751±0.14	3.2462±0.23	86.67±11.55
Breast Tissue	73.2704±1.09	26.7296±1.09	28.6135±2.35	56.67±15.28	73.8994±1.96	26.1006±1.96	28.5213±1.02	73.33±20.82
Cardiotocography-1	95.3904±0.05	4.6096±0.05	4.8771±0.19	227.33±127.02	95.2179±0.27	4.7821±0.27	5.1367±0.15	183.33±25.40
Cardiotocography-2	88.7896±0.29	11.2104±0.29	11.7733±0.26	220.00±66.00	88.8366±0.07	11.1634±0.07	11.9406±0.17	212.67±50.81
Climate Model	95.3086±0.11	4.6914±0.11	5.0000±0.00	120.33±21.94	94.4444±0.19	5.5556±0.19	6.3459±0.22	171.00±0.00
DNA	96.3486±0.08	3.6514±0.08	4.6168±0.17	3077.33±905.00	96.6729±0.03	3.3271±0.03	4.6095±0.06	4163.00±652.60
Glass	81.3084±0.93	18.6916±0.93	21.1292±0.90	80.00±20.00	79.7508±0.71	20.2492±0.71	21.5127±0.42	56.67±15.28
HTRU2	98.0221±0.01	1.9779±0.01	2.0618±0.01	60.00±10.39	98.0556±0.01	1.9444±0.01	2.1333±0.17	63.00±15.59
Ionosphere	94.9668±0.16	5.0332±0.16	6.1595±0.09	478.33±145.72	94.4919±0.16	5.5081±0.16	6.5266±0.23	455.00±252.39
Leaf	78.8235±0.51	21.1765±0.51	23.9338±0.50	155.00±17.32	77.1569±1.33	22.8431±1.33	24.2878±1.36	90.00±25.98

Çizelge C.25: (devam) Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının (Temel Öğrenici olarak RF algoritması seçildi) özniteliklerin enformasyon ölçümü yöntemlerine göre karşılaştırma sonuçları (deneyler 3 kere tekrarlandı)

Letter Recognition	96.9700±0.07	3.0300±0.07	3.6112±0.05	158.67±19.63	96.4783±0.05	3.5217±0.05	3.8273±0.31	90.67±19.63
Libras Movement	86.3889±0.56	13.6111±0.56	16.1248±0.42	1365.00±157.62	87.1296±0.58	12.8704±0.58	15.6511±0.25	940.33±52.54
LSVT Voice Rehabilitation	92.5926±0.46	7.4074±0.46	10.2623±0.96	10159.33±2804.75	89.1534±0.46	10.8466±0.46	13.1103±0.72	8708.00±6582.63
Madelon	89.9000±0.22	10.1000±0.22	11.1942±0.20	12525.00±1325.52	74.9000±0.40	25.1000±0.40	30.7598±0.22	8684.00±2085.83
MAGIC Gamma Telescope	88.2089±0.08	11.7911±0.08	12.0363±0.08	66.00±0.00	88.0687±0.02	11.9313±0.02	12.2321±0.02	66.00±0.00
Parkinson Speech	73.4295±0.80	26.5705±0.80	28.2646±0.95	351.00±194.70	71.7308±0.67	28.2692±0.67	30.5444±0.56	342.00±102.22
QSAR biodegradation	88.2148±0.22	11.7852±0.22	12.8633±0.87	462.00±84.00	89.0047±0.09	10.9953±0.09	12.1579±0.04	518.00±174.86
Satellite	92.2041±0.15	7.7959±0.15	8.2955±0.11	431.67±85.45	92.3491±0.05	7.6509±0.05	8.2892±0.05	481.00±133.41
Seeds	94.2857±0.95	5.7143±0.95	5.9498±1.32	48.00±13.86	95.2381±0.00	4.7619±0.00	4.7619±0.00	42.67±18.48
Shuttle	99.9879±0.00	0.0121±0.00	0.0121±0.00	60.00±10.00	99.9925±0.00	0.0075±0.00	0.0100±0.00	83.33±11.55
Sonar	89.9038±0.83	10.0962±0.83	14.1729±1.11	935.33±372.72	89.7436±1.00	10.2564±1.00	14.6168±0.71	772.67±253.96
Vehicle	76.5563±0.36	23.4437±0.36	23.6801±0.14	88.67±10.97	75.7289±0.30	24.2711±0.30	24.6651±0.38	95.00±0.00
Vertebral Column	84.5161±0.00	15.4839±0.00	16.2373±1.30	32.67±14.57	84.5161±0.00	15.4839±0.00	16.6518±2.02	30.33±8.08
Vowel	95.8249±0.12	4.1751±0.12	4.4791±0.26	62.33±16.80	95.2525±0.63	4.7475±0.63	4.7475±0.63	44.00±0.00
Wall Following Robot Navigation-1	100.0±0.00	0.0000±0.00	0.0002±0.00	9.00±0.00	100.0±0.00	0.0000±0.00	0.0002±0.00	9.00±0.00

Çizelge C.25: (devam) Özniteliklerin bağımlılığı varsayımına göre VIBES algoritmasının (Temel Öğrenici olarak RF algoritması seçildi) özniteliklerin enformasyon ölçümü yöntemlerine göre karşılaştırma sonuçları (deneyler 3 kere tekrarlandı)

Wall Following Robot Navigation-2	100.0±0.00	0.0000±0.00	0.0002±0.00	25.00±0.00	100.0±0.00	0.0000±0.00	0.0002±0.00	25.00±0.00
Wall Following Robot Navigation-3	99.5479±0.04	0.4521±0.04	0.4521±0.04	125.00±0.00	99.5551±0.05	0.4949±0.05	0.4949±0.05	125.00±0.00
Wine Quality-red	72.2118±0.34	27.7882±0.34	28.7046±0.46	100.00±13.86	71.9616±0.41	28.0384±0.41	28.9343±0.57	92.00±30.20
Wine Quality-white	71.5938±0.05	28.4062±0.05	29.2121±0.32	84.00±12.00	71.2604±0.22	28.7396±0.22	29.6207±0.35	96.00±12.00
Zoo	98.3498±0.57	1.6502±0.57	1.6502±0.57	107.67±35.39	97.3597±0.57	2.6403±0.57	2.6403±0.57	141.67±76.66
Ortalama	89.8153	10.1846	11.1403	972.77	89.0519	10.9495	12.0251	823.76
Rank			22					14

Çizelge C.26: Baseline algoritmaların Parkinson veri kümesi üzerindeki sınıflandırma sonuçları

Veri kümesine Uygulanan Sinyal Dönüşümü	Baseline	
	Rastgele Tahmin Edici ¹⁴	ZeroR
Fourier Dönüşümü	50.24±8.76	56.36
Gaussian2	48.00±3.58	56.36
BiorSplines3.3	49.15±5.38	56.36
Coiflets2	50.24±3.03	56.36
Daubechies2	50.24±6.52	56.36
Daubechies3	50.42±4.78	56.36
Daubechies4	50.18±6.88	56.36
Daubechies5	50.00±4.14	56.36
Daubechies6	49.33±5.91	56.36
Daubechies7	50.00±4.43	56.36
Dmeyer	50.97±5.65	56.36
Haar	49.70±6.70	56.36
Symlets2	50.42±4.73	56.36
Hilbert-Huang Dönüşümü	50.55±8.30	56.36
Ortalama	49.96	56.36
Yerel Rank	0	14

¹⁴ Deneyler 10 kere gerçekleştirildi.

Çizelge C.27: Bayesian Öğrenme algoritmalarının normalize edilmiş Parkinson veri kümesi üzerindeki sınıflandırma sonuçları

Veri kümesine Uygulanan Sinyal Dönüşümü	Bayesian Öğrenme						
	A1DE	A2DE	BayesNet	Bayesian Logistic Regression	NaiveBayes	NaiveBayes Simple	NaiveBayes Updateable
Fourier Dönüşümü	89.09	88.48	89.09	84.24	86.67	85.45	86.67
Gaussian2	86.67	88.48	89.09	86.06	86.67	86.67	86.67
BiorSplines3.3	87.88	86.06	90.91	86.67	87.88	89.09	87.88
Coiflets2	87.27	90.30	90.91	87.88	87.88	88.48	87.88
Daubechies2	88.48	86.67	88.48	87.88	87.27	86.06	87.27
Daubechies3	93.94	90.91	92.12	87.27	89.70	89.70	89.70
Daubechies4	92.12	90.30	91.52	86.06	86.67	86.67	86.67
Daubechies5	88.48	89.09	87.88	87.27	87.88	87.88	87.88
Daubechies6	90.30	89.09	89.70	83.03	88.48	85.45	88.48
Daubechies7	89.70	90.30	91.52	86.67	89.09	89.09	89.09
DMeyer	85.45	87.27	87.88	87.27	87.88	87.88	87.88
Haar	89.70	89.70	89.70	85.45	87.88	90.30	87.88
Symlets2	88.48	86.67	88.48	87.88	87.27	86.06	87.27
Hilbert-Huang Dönüşümü	89.70	89.70	93.94	86.06	87.88	89.09	87.88
Ortalama	89.09	88.79	90.09	86.41	87.79	87.71	87.79
Yerel Rank	6	1	9	0	1	2	1

Çizelge C.28: Fonksiyon Öğrenme algoritmalarının normalize edilmiş Parkinson veri kümesi üzerindeki sınıflandırma sonuçları

Veri kümesine Uygulanan Sinyal Dönüşümü	Fonksiyon Öğrenme																
	FLDA	Kernel Logistic Regression ¹⁵	LDA	İbLINEAR	İbSVM	Logistic	MLP Classifier ¹⁵	Multilayer Perceptron ¹⁵	QDA	RBF Classifier ¹⁵	RBF Network ¹⁵	SGD	Simple Logistic ¹⁵	SMO	SPegasos ¹⁵	Voted Perceptron	
Fourier Dönüşümü	81.82	86.67	84.85	86.06	86.06	86.67	82.42	81.82	81.82	83.64	83.64	78.18	85.45	84.85	85.45	83.64	83.64
Gaussian2	81.82	87.27	84.85	86.06	86.67	87.27	83.03	80.61	80.61	84.24	84.85	79.39	84.85	85.45	85.45	81.82	83.64
BiorSplines3.3	87.27	86.67	89.70	84.24	87.27	88.48	87.88	87.88	87.88	86.06	85.45	75.15	91.52	90.91	87.88	84.85	86.67
Coiflets2	85.45	85.45	86.06	87.88	87.27	85.45	82.42	82.42	82.42	87.27	84.85	76.97	88.48	84.85	86.67	84.85	85.45
Daubechies2	85.45	86.67	88.48	84.24	87.88	83.03	79.39	80.61	80.61	83.03	86.06	84.85	83.64	86.67	86.67	84.24	85.45
Daubechies3	86.06	90.30	93.33	89.09	87.88	87.27	86.06	87.27	87.27	87.27	84.24	81.82	88.48	87.88	88.48	90.30	86.67
Daubechies4	90.30	88.48	86.67	88.48	84.24	86.67	87.88	87.88	87.88	88.48	89.09	82.42	86.67	87.88	88.48	89.09	86.67
Daubechies5	85.45	84.85	86.67	86.67	86.06	86.67	82.42	84.85	84.85	86.67	86.06	79.39	87.88	86.06	86.67	85.45	86.06
Daubechies6	86.67	87.88	87.88	84.24	83.64	87.27	83.64	83.03	83.03	82.42	85.45	75.76	83.64	83.03	83.64	85.45	84.24

¹⁵ Özellik seçiminde kullanılan temel öğrenici, uygulanan sinyal dönüşümü için en yüksek doğruluğu veren temel öğrenicidir. Örneğin Fonksiyon Öğrenme algoritmaları arasında Fourier dönüşümü için seçilen temel öğrenici Logistic sınıflandırıcıdır.

Çizelge C.28: (devam) Fonksiyon Öğrenme algoritmalarının normalize edilmiş Parkinson veri kümesi üzerindeki sınıflandırma sonuçları

Daubechies7	89.09	89.70	84.85	86.67	88.48	89.70	87.88	83.64	83.64	89.09	87.27	75.76	88.48	86.06	86.06	86.67	86.67
DMeyer	90.30	83.64	84.85	84.85	85.45	87.27	83.64	80.61	80.61	83.64	85.45	79.39	86.06	86.06	86.67	84.24	89.70
Haar	88.48	83.64	88.48	89.09	90.30	86.06	86.67	80.00	80.00	85.45	84.85	78.18	86.06	83.03	87.88	84.24	86.67
Symlets2	85.45	86.67	88.48	84.24	87.88	85.45	79.39	80.61	80.61	83.03	86.06	84.85	83.03	86.67	87.88	84.24	85.45
Hilbert-Huang Dönüşümü	87.27	80.61	87.27	87.27	88.48	87.27	83.03	80.00	80.00	86.06	85.45	80.61	87.88	81.21	89.09	83.64	86.67
Ortalama	86.49	86.32	87.32	86.36	86.97	86.75	83.98	82.95	82.95	85.45	85.63	79.48	86.58	85.76	86.93	85.19	85.98
Yerel Rank	2	4	4	0	1	3	0	0	0	0	0	0	3	0	1	0	0

Çizelge C.29: Örnek Öğrenme algoritmalarının normalize edilmiş Parkinson veri kümesi üzerindeki sınıflandırma sonuçları

Veri kümesine Uygulanan Sinyal Dönüşümü	Örnek Öğrenme					
	IBk	IbklG ₁₅	Kstar ₁₅	LocalKnn ₁₅	LWL	RseslibKnn ₁₅
Fourier Dönüşümü	81.21	81.21	76.97	80.00	79.39	80.61
Gaussian2	83.64	83.64	80.00	82.42	79.39	84.24
BiorSplines3.3	83.64	83.64	83.64	80.61	79.39	83.03
Coiflets2	87.27	87.27	81.82	85.45	82.42	86.06
Daubechies2	84.24	84.24	77.58	80.00	80.00	79.39
Daubechies3	87.27	87.27	80.61	83.03	82.42	83.03
Daubechies4	91.52	91.52	84.85	83.64	82.42	86.06
Daubechies5	90.91	90.91	79.39	86.06	80.61	81.82
Daubechies6	88.48	88.48	86.06	83.03	82.42	80.61
Daubechies7	87.88	87.88	80.00	81.21	79.39	84.24
DMeyer	90.30	90.30	85.45	83.64	80.61	81.21
Haar	84.85	84.85	76.97	80.61	82.42	79.39
Symlets2	84.24	84.24	77.58	80.00	80.00	79.39
Hilbert-Huang Dönüşümü	91.52	91.52	80.00	80.61	83.03	84.85
Ortalama	86.93	86.93	80.78	82.17	80.99	82.42
Yerel Rank	13	13	1	0	0	1

Çizelge C.30: Çeşitli öğrenme yaklaşımlarına sahip algoritmaların normalize edilmiş Parkinson veri kümesi üzerindeki sınıflandırma sonuçları

Veri kümesine Uygulanan Sinyal Dönüşümü	CHIRP	FLR	HyperPipes	VFI
Fourier Dönüşümü	82.42	72.73	77.58	65.45
Gaussian2	81.21	72.73	78.18	65.45
BiorSplines3.3	82.42	80.61	82.42	83.03
Coiflets2	83.64	83.03	86.67	86.06
Daubechies2	83.64	78.79	83.64	84.24
Daubechies3	86.06	77.58	86.67	87.27
Daubechies4	81.82	80.61	89.70	87.27
Daubechies5	83.03	82.42	82.42	84.24
Daubechies6	83.64	80.00	84.24	79.39
Daubechies7	84.24	80.61	86.06	85.45
DMeyer	83.03	83.03	83.64	84.24
Haar	83.03	79.39	81.82	86.06
Symlets2	80.00	78.79	83.64	84.24
Hilbert-Huang Dönüşümü	84.85	81.21	89.09	87.27
Ortalama	83.07	79.40	83.98	82.12
Yerel Rank	2	0	4	8

Çizelge C.31: Kural Tabanlı Öğrenme algoritmalarının normalize edilmiş Parkinson veri kümesi üzerindeki sınıflandırma sonuçları

Veri kümesine Uygulanan Sinyal Dönüşümü	Kural Öğrenme												
	Conjunctive Rule	Decision Table	DTNB ₁₅	FURIA ₁₅	JRip	MODLEM ₁₅	NN _{ge}	OLM	OneR	PART	Ridor	RoughSet ₁₅	Vfdt
Fourier Dönüşümü	80.61	86.67	84.24	79.39	80.61	78.18	83.03	72.73	78.79	81.82	79.39	74.55	86.06
Gaussian2	80.61	84.85	81.21	81.82	83.64	78.79	83.03	75.15	78.79	83.03	83.03	82.42	86.67
BiorSplines3.3	81.21	81.82	82.42	78.18	83.64	78.79	83.03	78.18	78.79	86.67	83.64	80.00	86.67
Coiflets2	80.61	82.42	80.00	79.39	84.85	78.79	82.42	78.79	80.00	82.42	84.85	78.79	87.27
Daubechies2	80.61	83.64	81.82	80.00	83.64	81.21	81.21	83.64	79.39	83.64	83.03	76.97	87.88
Daubechies3	80.61	88.48	87.27	81.82	86.67	78.18	84.24	79.39	80.00	89.70	83.03	78.18	87.88
Daubechies4	81.21	85.45	83.64	84.85	86.67	83.03	82.42	81.82	79.39	89.70	80.00	81.82	86.67
Daubechies5	80.00	86.06	80.00	80.00	83.03	80.61	82.42	80.61	79.39	83.03	83.64	80.00	88.48
Daubechies6	80.00	86.67	86.06	80.00	83.64	80.61	82.42	81.21	79.39	82.42	84.24	84.24	88.48
Daubechies7	80.61	86.67	83.64	80.61	82.42	78.79	84.85	83.64	78.18	84.85	83.03	79.39	89.09
DMeyer	80.61	81.82	79.39	79.39	82.42	78.79	83.03	80.00	78.18	87.88	83.03	75.76	86.67

Çizelge C.31: (devam) Kural Tabanlı Öğrenme algoritmalarının normalize edilmiş Parkinson veri kümesi üzerindeki sınıflandırma sonuçları

Haar	81.82	84.85	79.39	80.61	88.48	77.58	83.03	81.21	78.79	87.27	84.24	84.24	89.09
Symlets2	80.61	83.64	81.82	83.03	83.03	81.21	81.21	83.64	79.39	83.64	83.03	76.97	87.88
Hilbert-Huang Dönüşümü	78.18	90.30	86.67	81.82	81.82	80.00	81.82	78.79	80.00	86.67	84.85	86.06	87.88
Ortalama	80.52	85.24	82.68	80.78	83.90	79.61	82.73	79.91	79.18	85.20	83.07	79.96	87.62
Yerel Rank	0	2	0	0	0	0	0	0	0	4	0	0	9

Çizelge C.32: Karar Ağacı Öğrenme algoritmalarının normalize edilmiş Parkinson veri kümesi üzerindeki sınıflandırma sonuçları

Veri kümesine Uygulanan Sinyal Dönüşümü	Karar Ağacı Öğrenme																
	ADTree	BFTree	CDT	Decision Stump	Extra Tree	FT ¹⁵	Hoeffding Tree	J48	J48 Consolidated	J48graft	LADTree ¹⁵	LMT ¹⁵	NBTree ¹⁵	Random Forest ¹⁵	RandomTree	REPTree	SimpleCart ¹⁵
Fourier Dönüşümü	83.03	79.39	81.21	77.58	76.97	84.24	86.06	81.21	83.64	80.61	79.39	83.03	78.18	81.82	77.58	80.00	78.18
Gaussian2	83.03	79.39	81.82	77.58	81.21	82.42	86.67	83.64	82.42	83.03	81.82	83.64	81.82	84.85	79.39	81.21	80.00
BiorSplines3.3	83.64	84.24	84.24	79.39	78.79	84.24	86.67	83.64	86.67	84.24	76.97	84.24	76.97	80.00	84.24	81.82	72.73
Coiflets2	86.06	83.03	84.24	79.39	74.55	81.21	87.27	83.03	82.42	85.45	75.76	83.64	76.97	81.21	81.21	83.64	76.97
Daubechies2	83.03	85.45	83.03	79.39	79.39	81.21	87.88	83.64	81.82	83.03	78.79	86.06	80.61	83.03	80.00	81.21	75.76
Daubechies3	89.09	84.24	82.42	79.39	81.82	81.21	87.88	90.30	87.88	87.27	84.24	83.03	86.67	84.24	81.82	81.82	81.21
Daubechies4	86.06	86.06	80.61	81.21	78.79	80.00	86.67	86.06	88.48	84.85	85.45	83.03	83.03	84.85	86.06	81.82	86.06
Daubechies5	81.82	84.85	83.64	79.39	79.39	82.42	88.48	79.39	81.21	83.03	72.73	84.24	78.79	83.64	84.24	83.03	75.15
Daubechies6	85.45	83.64	80.00	80.00	80.00	79.39	88.48	85.45	81.82	86.06	80.00	82.42	79.39	83.03	80.00	80.00	72.73
Daubechies7	87.88	83.03	84.24	80.00	80.61	84.85	89.09	80.61	83.64	83.64	78.79	82.42	80.61	84.24	82.42	78.79	75.15
DMeyer	83.64	86.06	81.21	80.00	78.79	80.61	86.67	85.45	84.85	84.85	78.79	81.82	84.85	82.42	84.24	83.03	75.15

Çizelge C.32: (devam) Karar Ağacı Öğrenme algoritmalarının normalize edilmiş Parkinson veri kümesi üzerindeki sınıflandırma sonuçları

Haar	87.88	85.45	83.03	78.79	81.82	77.58	89.09	89.70	86.06	86.06	82.42	80.61	83.03	82.42	84.85	80.61	85.45
Symlets2	83.03	85.45	83.03	79.39	76.97	81.21	87.88	84.85	81.82	83.03	78.79	86.06	81.21	85.45	80.00	81.21	75.76
Hilbert-Huang Dönüşümü	89.70	82.42	82.42	79.39	76.97	80.61	87.88	85.45	89.70	88.48	89.09	81.21	84.85	86.67	84.85	84.24	80.61
Ortalama	85.24	83.76	82.51	79.35	79.01	81.51	87.62	84.46	84.46	84.55	80.22	83.25	81.21	83.42	82.21	81.60	77.92
Yerel Rank	1	0	0	0	0	0	10	2	3	0	0	0	0	0	0	0	0

Çizelge C.33: Temel Öğrencilerin Parkinson veri tabanı üzerindeki en iyi oldukları veri kümeleri ve sonuçları

Veri kümesine Uygulanan Sinyal Dönüşümü	AIDE	BayesNet	SGD	LDA	IBk	IBkLG	FLDA	NaiveBayes Simple	libSVM
Fourier Dönüşümü	89.09	89.09	85.45	84.85	81.21	81.21	81.82	85.45	86.06
Gaussian2	86.67	89.09	84.85	84.85	83.64	83.64	81.82	86.67	86.67
BiorSplines3.3	87.88	90.91	91.52	89.70	83.64	83.64	87.27	89.09	87.27
Coiflets2	87.27	90.91	88.48	86.06	87.27	87.27	85.45	88.48	87.27
Daubechies2	88.48	88.48	83.64	88.48	84.24	84.24	85.45	86.06	87.88
Daubechies3	93.94	92.12	88.48	93.33	87.27	87.27	86.06	89.70	87.88
Daubechies4	92.12	91.52	86.67	86.67	91.52	91.52	90.30	86.67	84.24
Daubechies5	88.48	87.88	87.88	86.67	90.91	90.91	85.45	87.88	86.06
Daubechies6	90.30	89.70	83.64	87.88	88.48	88.48	86.67	85.45	83.64
Daubechies7	89.70	91.52	88.48	84.85	87.88	87.88	89.09	89.09	88.48
DMeyer	85.45	87.88	86.06	84.85	90.30	90.30	90.30	87.88	85.45

Çizelge C.33: (devam) Temel Öğrencilerin Parkinson veri tabanı üzerindeki en iyi oldukları veri kümeleri ve sonuçları

Haar	89.70	89.70	86.06	88.48	84.85	84.85	88.48	90.30	90.30
Symlets2	88.48	88.48	83.03	88.48	84.24	84.24	85.45	86.06	87.88
Hilbert-Huang Dönüşümü	89.70	93.94	87.88	87.27	91.52	91.52	87.27	89.09	88.48
Ortalama	89.09	90.09	86.58	87.32	86.93	86.93	86.49	87.71	86.97
Global Rank	6	7	1	2	2	2	1	1	1

Çizelge C.34: Topluluk/Meta Öğrenme algoritmalarının Parkinson veri kümesi üzerindeki sonuçları (Birinci grup)

Veri kümesine Uygulanan Sinyal Dönüşümü	Topluluk/Meta Öğrenme												
	AdaBoostM1 ¹⁶	Bagging ¹⁶	Classification via Cluctering ¹⁷	Classification via Regression ¹⁷	Dagging ¹⁶	Decorate ¹⁶	END ¹⁶	Filtered Classifier ¹⁶	Iterative Classifier Optimizer ¹⁷	LogitBoost ¹⁷	MultiBoostAB ¹⁶	MultiClass Classifier ¹⁶	MultiClass Classifier Updateable ¹⁷
Fourier Dönüşümü	85.45	86.67	77.58	77.58	82.42	88.48	89.09	89.09	79.39	78.79	84.24	89.09	83.64
Gaussian2	86.67	87.88	81.82	81.82	85.45	85.45	88.48	88.48	78.79	78.79	86.06	88.48	81.21
BiorSplines3.3	84.85	86.06	81.82	82.42	80.61	91.52	91.52	84.85	77.58	84.85	87.27	91.52	91.52
Coiflets2	84.24	86.67	84.24	82.42	85.45	87.88	90.91	90.91	82.42	80.61	86.06	90.91	84.24
Daubechies2	88.48	87.27	79.39	80.61	80.00	83.64	88.48	88.48	80.61	81.21	86.06	88.48	81.21
Daubechies3	87.88	89.09	80.61	82.42	83.64	89.70	93.94	93.94	83.64	83.64	89.70	93.94	83.64
Daubechies4	89.09	89.70	78.79	82.42	80.61	89.70	92.12	92.12	85.45	86.06	90.30	92.12	80.61

¹⁶ Verilen sinyal dönüşümünü en yüksek doğruluk oranında sınıflandıran temel öğrenici kullanıldı. temel öğreniciler Çizelge C.33'da gösterilmektedir (Aynı doğruluk oranına sahip temel öğreniciler için ortalama doğruluk oranına bakıldı). Öznitelik seçiminde ise wrapper yöntemi kullanıldı. Wrapper yöntemi için temel öğrenici olarak varsayılan temel öğrenici kullanıldı.

¹⁷ Varsayılan parametreler kullanıldı. Öznitelik seçiminde ise wrapper yöntemi kullanıldı. Wrapper yöntemi için temel öğrenici olarak varsayılan temel öğrenici kullanıldı.

Çizelge C.34: (devam) Topluluk/Meta Öğrenme algoritmalarının Parkinson veri kümesi üzerindeki sonuçları (Birinci grup)

Daubechies5	90.30	89.09	80.61	79.39	84.24	90.30	90.91	76.97	78.79	80.00	90.91	90.91	82.42
Daubechies6	89.09	83.64	80.00	81.21	81.21	88.48	90.30	90.30	83.03	81.82	87.88	90.30	82.42
Daubechies7	87.27	89.09	81.82	78.79	86.67	89.70	91.52	91.52	83.03	84.24	90.91	91.52	80.00
DMeyer	89.09	87.27	79.39	78.79	81.21	90.30	90.30	81.21	81.21	81.21	89.09	90.30	80.61
Haar	84.85	89.09	86.06	81.82	87.27	89.70	90.30	87.88	83.64	81.82	87.88	90.30	83.03
Symlets2	88.48	87.27	79.39	80.61	80.00	83.64	88.48	88.48	80.61	81.21	86.06	88.48	81.21
Hilbert-Huang Dönüşümü	84.85	89.70	82.42	77.58	84.24	89.09	93.94	93.94	82.42	83.03	89.09	93.94	81.82
Ortalama	87.19	87.75	81.00	80.56	83.07	88.40	90.74	88.44	81.47	81.95	87.97	90.74	82.68
Yerel Rank	0	0	0	0	0	2	9	6	0	0	0	9	1

Çizelge C.35: Topluluk/Meta Öğrenme algoritmalarının Parkinson veri kümesi üzerindeki sonuçları (İkinci grup)

Veri kümesine Uygulanan Sinyal Dönüşümü	Topluluk/Meta Öğrenme													
	OrdinalClass Classifier ¹⁶	RacedIncremental LogitBoost ¹⁷	Random Committee ¹⁷	Randomizable Filtered Classifier ¹⁶	Random SubSpace ¹⁶	RealAdaBoost ¹⁶	Rotation Forest ¹⁶	Threshold Selector ¹⁶	Weighted InstancesHandler Wrapper ¹⁶	Grading ¹⁸	MultiScheme ¹⁸	Stacking ¹⁸	StackingC ¹⁸	Vote ¹⁸
Fourier Dönüşümü	89.09	56.36	76.36	77.58	87.27	88.48	84.85	87.88	89.09	89.70	89.09	88.48	87.88	89.09
Gaussian2	88.48	56.36	81.21	78.18	87.88	88.48	83.03	86.06	88.48	88.48	89.09	88.48	87.88	89.09
BiorSplines3.3	91.52	56.36	84.24	85.45	83.03	83.64	90.30	91.52	91.52	89.09	90.30	91.52	91.52	91.52
Coiflets2	90.91	56.36	81.21	78.18	86.06	90.30	83.03	84.24	90.91	88.48	90.91	87.27	90.91	86.67
Daubechies2	88.48	56.36	78.79	81.82	83.64	87.88	80.00	81.82	88.48	89.09	87.88	88.48	88.48	88.48
Daubechies3	93.94	56.36	83.64	82.42	89.70	84.85	86.06	89.70	93.94	93.33	93.94	92.12	93.94	89.09
Daubechies4	92.12	56.36	83.64	81.21	90.30	87.27	82.42	87.88	92.12	90.91	92.12	89.70	90.91	89.70

¹⁸ Verilen sinyal dönüşümünü en yüksek doğruluk oranında sınıflandıran iki temel öğrenici seçildi. Öznitelik seçiminde ise wrapper yöntemi kullanıldı. Wrapper yöntemi için temel öğrenici olarak en yüksek doğruluk oranına sahip temel öğrenici kullanıldı.

Çizelge C.35: (devam) Topluluk/Meta Öğrenme algoritmalarının Parkinson veri kümesi üzerindeki sonuçları (İkinci grup)

Daubechies5	90.91	56.36	82.42	76.97	85.45	90.91	84.24	90.30	90.91	86.67	90.91	84.24	91.52	91.52
Daubechies6	90.30	56.36	82.42	85.45	84.85	89.70	81.21	86.06	90.30	90.30	90.30	89.70	89.09	83.64
Daubechies7	91.52	56.36	85.45	80.61	89.09	88.48	85.45	87.27	91.52	90.91	91.52	90.30	91.52	91.52
DMeyer	90.30	56.36	78.79	77.58	83.03	89.09	81.21	89.09	90.30	84.85	90.30	84.24	90.30	90.30
Haar	90.30	56.36	81.21	81.21	89.70	84.24	88.48	85.45	90.30	89.09	90.30	87.88	89.70	84.85
Symlets2	88.48	56.36	78.79	81.82	83.64	87.88	80.00	81.82	88.48	89.09	87.88	88.48	88.48	88.48
Hilbert-Huang Dönüşümü	93.94	56.36	87.27	79.39	91.52	92.12	83.64	92.12	93.94	93.94	80.00	92.73	93.33	84.85
Ortalama	90.74	56.36	81.82	80.56	86.80	88.09	83.85	87.23	90.74	89.57	89.61	88.83	90.39	88.49
Yerel Rank	9	0	0	0	0	0	0	1	9	5	8	1	6	5

Çizelge C.36: Makine Öğrenmesi algoritmalarının (VIBES hariç) Parkinson veri kümesi üzerindeki karşılaştırma sonuçları (Sadece temel öğrencilerden daha iyi bir öğrenme gerçekleştiren Topluluk Öğrenciler listede yer almaktadır)

Veri kümesine Uygulanan Sinyal Dönüşümü	AIDE	BayesNet	SGD	IBk	IBkLG	FLDA	NaiveBayes Simple	libSVM	Grading	StackingC	Vote	Ortalama
Fourier Dönüşümü	89.09	89.09	85.45	81.21	81.21	81.82	85.45	86.06	89.70	87.88	89.09	89.09
Gaussian2	86.67	89.09	84.85	83.64	83.64	81.82	86.67	86.67	88.48	87.88	89.09	89.09
BiorSplines3.3	87.88	90.91	91.52	83.64	83.64	87.27	89.09	87.27	89.09	91.52	91.52	89.29
Coiflets2	87.27	90.91	88.48	87.27	87.27	85.45	88.48	87.27	88.48	90.91	86.67	89.42
Daubechies2	88.48	88.48	83.64	84.24	84.24	85.45	86.06	87.88	89.09	88.48	88.48	89.64
Daubechies3	93.94	92.12	88.48	87.27	87.27	86.06	89.70	87.88	93.33	93.94	89.09	89.76
Daubechies4	92.12	91.52	86.67	91.52	91.52	90.30	86.67	84.24	90.91	90.91	89.70	89.24
Daubechies5	88.48	87.88	87.88	90.91	90.91	85.45	87.88	86.06	86.67	91.52	91.52	88.83
Daubechies6	90.30	89.70	83.64	88.48	88.48	86.67	85.45	83.64	90.30	89.09	83.64	88.89
Daubechies7	89.70	91.52	88.48	87.88	87.88	89.09	89.09	88.48	90.91	91.52	91.52	88.61
DMeyer	85.45	87.88	86.06	90.30	90.30	90.30	87.88	85.45	84.85	90.30	90.30	88.33

Çizelge C.36: (devam) Makine Öğrenmesi algoritmalarının (VIBES hariç) Parkinson veri kümesi üzerindeki karşılaştırma sonuçları (Sadece temel öğrencilerden daha iyi bir öğrenme gerçekleştiren Topluluk Öğrenciler listede yer almaktadır)

Haar	89.70	89.70	86.06	84.85	84.85	88.48	90.30	90.30	89.09	89.70	84.85	89.29
Symlets2	88.48	88.48	83.03	84.24	84.24	85.45	86.06	87.88	89.09	88.48	88.48	89.09
Hilbert-Huang Dönüşümü	89.70	93.94	87.88	91.52	91.52	87.27	89.09	88.48	93.94	93.33	84.85	89.70
Ortalama	89.09	90.09	86.58	86.93	86.93	86.49	87.71	86.97	89.57	90.39	88.49	
Global Rank	3	4	1	1	1	1	1	1	5	2	2	

Çizelge C.37: VIBES algoritmasının (temel öğrenici olarak k-EYK kullanıldı) farklı parametrelerine göre Parkinson veri kümesi üzerindeki karşılaştırma sonuçları (Herbir deney 3 kere tekrarlandı)

Veri kümesine Uygulanan Sinyal Dönüşümü	k-EYK					
	Dependent = <i>True</i> Measure = <i>relief</i> generateEnsemble = <i>Optimize</i>	Dependent = <i>True</i> Measure = <i>relief</i> generateEnsemble = <i>GA</i>	Dependent = <i>True</i> Measure = <i>shannon</i> generateEnsemble = <i>Optimize</i>	Dependent = <i>True</i> Measure = <i>shannon</i> generateEnsemble = <i>GA</i>	Dependent = <i>False</i> generateEnsemble = <i>Optimize</i>	Dependent = <i>False</i> generateEnsemble = <i>GA</i>
Fourier Dönüşümü	82.42±1.05	83.43±0.93	76.36±0.61	76.57±0.93	85.25±1.95	86.67±0.61
Gaussian2	84.65±0.35	85.05±0.35	71.72±0.35	71.92±0.35	84.85±1.05	85.66±0.70
BiorSplines3.3	82.83±0.35	78.99±1.40	81.21±0.00	71.92±0.93	92.12±0.61	87.07±2.52
Coiflets2	89.09±1.05	83.84±0.70	77.98±0.70	72.32±0.35	93.33±0.61	87.47±0.93
Daubechies2	86.87±0.35	84.85±0.61	80.00±0.61	71.92±0.70	91.92±1.52	89.90±0.70
Daubechies3	85.05±0.93	80.40±2.52	79.60±0.93	65.25±0.35	92.32±0.70	90.30±0.61

Çizelge C.37: (devam) VIBES algoritmasının (temel öğrenici olarak k-EYK kullanıldı) farklı parametrelerine göre Parkinson veri kümesi üzerindeki karşılaştırma sonuçları (Herbir deney 3 kere tekrarlandı)

Daubechies4	83.84±0.93	82.63±0.35	76.16±0.35	74.14±0.35	93.74±0.35	88.69±0.70
Daubechies5	80.40±0.35	75.76±1.60	79.19±1.26	73.94±1.82	91.52±1.21	89.29±1.95
Daubechies6	85.45±1.60	79.19±0.70	78.59±0.35	74.34±0.35	91.31±1.40	87.88±1.05
Daubechies7	88.89±0.70	84.44±0.93	75.56±0.35	70.30±0.00	92.93±1.52	89.90±0.35
DMeyer	87.47±0.35	81.62±0.93	81.62±0.93	78.99±1.40	92.73±1.05	88.89±0.70
Haar	86.46±0.70	84.65±0.70	82.83±1.26	74.95±1.26	92.73±1.05	90.91±0.61
Symlets2	87.07±0.70	85.25±0.70	79.19±0.93	72.12±0.61	91.52±0.61	89.70±0.61
Hilbert-Huang Dönüşümü	87.88±0.61	83.64±3.21	82.22±1.26	73.94±1.05	97.37±0.35	93.94±1.05
Ortalama	85.60	82.41	78.73	73.04	91.69	89.02
Yerel Rank	0	0	0	0	12	2

Çizelge C.38: VIBES algoritmasının (temel öğrenici olarak NB kullanıldı) farklı parametrelerine göre Parkinson veri kümesi üzerindeki karşılaştırma sonuçları (Herbir deney 3 kere tekrarlandı)

Veri kümesine Uygulanan Sinyal Dönüşümü	NB					
	Dependent = <i>True</i> Measure = <i>relief</i> generateEnsemble = <i>Optimize</i>	Dependent = <i>True</i> Measure = <i>relief</i> generateEnsemble = <i>GA</i>	Dependent = <i>True</i> Measure = <i>shannon</i> generateEnsemble = <i>Optimize</i>	Dependent = <i>True</i> Measure = <i>shannon</i> generateEnsemble = <i>GA</i>	Dependent = <i>False</i> generateEnsemble = <i>Optimize</i>	Dependent = <i>False</i> generateEnsemble = <i>GA</i>
Fourier Dönüşümü	86.26±0.70	86.46±0.35	85.45±0.61	85.86±0.70	88.08±0.35	88.48±0.61
Gaussian2	84.65±0.35	85.45±0.00	84.24±0.00	84.65±0.35	87.47±0.93	88.28±0.35
BiorSplines3.3	84.65±0.35	83.03±0.00	83.64±0.00	81.82±0.00	87.27±1.05	82.22±1.26
Coiflets2	87.68±0.35	82.42±0.00	84.24±0.61	81.82±0.00	89.49±0.70	83.23±0.93
Daubechies2	84.44±0.93	81.41±0.35	83.64±0.00	81.82±0.00	88.89±0.93	86.06±0.61
Daubechies3	84.85±0.00	81.21±0.00	81.82±0.00	80.00±0.00	90.51±0.70	82.22±1.75

Çizelge C.38: (devam) VIBES algoritmasının (temel öğrenici olarak NB kullanıldı) farklı parametrelerine göre Parkinson veri kümesi üzerindeki karşılaştırma sonuçları (Herbir deney 3 kere tekrarlandı)

Daubechies4	84.85±0.61	81.21±0.00	83.03±0.00	81.62±0.35	89.29±0.93	83.64±0.61
Daubechies5	83.64±0.00	82.42±0.00	83.43±0.35	82.83±0.35	88.48±0.61	83.84±0.35
Daubechies6	86.26±0.35	83.03±0.00	83.43±0.70	80.61±0.00	88.08±0.93	85.05±1.75
Daubechies7	86.87±0.93	82.83±0.35	83.64±0.00	80.61±0.00	89.49±0.35	83.43±1.26
DMeyer	86.87±0.35	82.42±0.00	82.02±0.35	80.61±0.00	88.28±0.35	88.81±0.35
Haar	83.64±0.00	80.00±0.00	83.64±0.00	81.41±0.35	89.49±0.35	87.07±0.35
Symlets2	84.24±0.61	81.62±0.35	83.84±0.70	81.62±0.35	88.69±0.35	86.06±0.61
Hilbert-Huang Dönüşümü	85.86±0.35	81.82±0.00	83.03±0.00	81.21±0.00	91.31±0.35	85.05±1.26
Ortalama	85.34	82.52	83.51	81.89	88.92	85.25
Yerel Rank	0	0	0	0	11	3

Çizelge C.39: VIBES algoritmasının (temel öğrenici olarak DYM kullanıldı) farklı parametrelerine göre Parkinson veri kümesi üzerindeki karşılaştırma sonuçları (Herbir deney 3 kere tekrarlandı)

Veri kümesine Uygulanan Sinyal Dönüşümü	DYM					
	Dependent = <i>True</i> Measure = <i>relief</i> generateEnsemble = <i>Optimize</i>	Dependent = <i>True</i> Measure = <i>relief</i> generateEnsemble = <i>GA</i>	Dependent = <i>True</i> Measure = <i>shannon</i> generateEnsemble = <i>Optimize</i>	Dependent = <i>True</i> Measure = <i>shannon</i> generateEnsemble = <i>GA</i>	Dependent = <i>False</i> generateEnsemble = <i>Optimize</i>	Dependent = <i>False</i> generateEnsemble = <i>GA</i>
Fourier Dönüşümü	86.46±0.35	86.26±0.35	85.86±0.70	86.06±0.61	86.87±0.70	87.88±0.61
Gaussian2	84.44±0.35	85.05±0.35	83.84±0.35	84.65±0.35	87.27±0.00	88.08±0.70
BiorSplines3.3	89.09±0.61	86.87±0.35	87.47±0.35	85.25±0.35	89.70±0.61	83.84±1.26
Coiflets2	88.28±0.93	83.84±0.35	87.88±1.05	84.04±0.35	89.09±0.00	83.03±0.00
Daubechies2	89.29±0.93	86.26±0.35	90.30±0.61	88.08±1.26	89.49±0.70	86.87±1.75
Daubechies3	90.71±0.35	86.06±0.00	90.91±0.00	88.69±0.35	91.11±0.70	83.84±2.29

Çizelge C.39: (devam) VIBES algoritmasının (temel öğrenici olarak DYM kullanıldı) farklı parametrelerine göre Parkinson veri kümesi üzerindeki karşılaştırma sonuçları (Herbir deney 3 kere tekrarlandı)

Daubechies4	89.70±0.61	86.06±0.00	87.27±1.05	83.64±0.61	90.71±0.70	85.25±1.75
Daubechies5	89.90±0.35	85.66±0.93	87.88±1.60	85.25±0.35	89.90±0.35	86.67±0.61
Daubechies6	90.30±0.00	88.89±0.35	89.29±0.93	88.08±0.70	89.70±0.61	84.44±0.35
Daubechies7	89.70±0.00	86.87±0.35	89.09±1.05	85.66±0.35	90.10±0.93	85.66±1.40
DMeyer	91.31±0.35	87.68±0.35	89.49±0.35	87.88±0.00	90.10±0.93	82.83±0.35
Haar	86.46±1.26	84.24±1.05	88.28±0.93	85.66±0.35	91.31±0.35	86.67±1.05
Symlets2	88.69±0.93	86.46±0.93	90.30±0.61	87.68±0.35	88.89±0.35	87.07±0.35
Hilbert-Huang Dönüşümü	89.09±1.21	87.07±0.35	89.09±0.61	85.86±0.35	91.72±1.52	85.45±0.61
Ortalama	88.82	86.23	88.35	86.18	89.71	85.54
Yerel Rank	3	0	2	0	8	2

Çizelge C.40: VIBES algoritmasının (temel öğrenici olarak DDA kullanıldı) farklı parametrelerine göre Parkinson veri kümesi üzerindeki karşılaştırma sonuçları (Herbir deney 3 kere tekrarlandı)

Veri kümesine Uygulanan Sinyal Dönüşümü	DDA					
	Dependent = <i>False</i> generateEnsemble = GA	Dependent = <i>False</i> generateEnsemble = <i>Optimize</i>	Dependent = <i>True</i> Measure = <i>shannon</i> generateEnsemble = GA	Dependent = <i>True</i> Measure = <i>shannon</i> generateEnsemble = <i>Optimize</i>	Dependent = <i>True</i> Measure = <i>relief</i> generateEnsemble = GA	Dependent = <i>True</i> Measure = <i>relief</i> generateEnsemble = <i>Optimize</i>
Fourier Dönüşümü	86.06±0.61	86.67±0.61	86.26±0.93	86.87±0.93	87.68±0.35	88.28±0.35
Gaussian2	85.66±0.70	86.26±0.35	84.44±0.35	85.05±0.70	88.28±0.35	88.89±0.35
BiorSplines3.3	88.08±1.26	85.45±0.61	87.07±1.26	81.41±0.70	89.90±0.70	80.00±0.61
Coiflets2	88.89±0.93	85.05±0.70	86.26±0.93	83.43±0.35	88.89±0.35	81.62±0.35
Daubechies2	85.05±0.35	84.24±0.00	86.87±0.35	84.04±0.70	88.28±0.35	84.85±0.61
Daubechies3	91.52±0.61	87.47±0.35	89.70±0.00	85.86±0.35	89.70±0.61	83.03±2.10

Çizelge C.40: (devam) VIBES algoritmasının (temel öğrenici olarak DDA kullanıldı) farklı parametrelerine göre Parkinson veri kümesi üzerindeki karşılaştırma sonuçları (Herbir deney 3 kere tekrarlandı)

Daubechies4	91.31±0.35	87.27±0.61	87.07±0.93	84.44±0.70	89.70±0.00	82.63±0.35
Daubechies5	88.08±0.35	85.45±0.00	86.46±1.26	83.43±0.93	89.49±1.26	86.06±1.60
Daubechies6	88.89±1.26	87.88±0.00	88.48±1.21	84.65±0.35	88.89±0.35	82.83±1.85
Daubechies7	89.09±0.61	87.88±0.00	88.69±0.35	84.65±0.70	89.29±0.70	83.84±1.26
DMeyer	89.90±0.93	86.26±1.95	84.24±0.61	79.80±0.35	89.29±1.85	80.40±0.93
Haar	85.25±0.35	83.43±0.35	87.47±0.35	84.04±0.35	91.11±0.35	86.26±0.35
Symlets2	86.26±0.70	85.05±0.35	86.67±0.61	83.23±0.35	88.89±0.70	85.25±1.40
Hilbert-Huang Dönüşümü	89.09±1.60	87.07±0.70	87.27±0.61	84.44±0.35	91.72±1.40	85.45±0.61
Ortalama	88.08	86.10	86.93	83.95	89.37	84.24
Yerel Rank	5	0	0	0	9	2

Çizelge C.41: VIBES algoritmasının (temel öğrenici olarak KA kullanıldı) farklı parametrelerine göre Parkinson veri kümesi üzerindeki karşılaştırma sonuçları (Herbir deney 3 kere tekrarlandı)

Veri kümesine Uygulanan Sinyal Dönüşümü	KA					
	Dependent = <i>True</i> Measure = <i>relief</i> generateEnsemble = <i>Optimize</i>	Dependent = <i>True</i> Measure = <i>relief</i> generateEnsemble = <i>GA</i>	Dependent = <i>True</i> Measure = <i>shannon</i> generateEnsemble = <i>Optimize</i>	Dependent = <i>True</i> Measure = <i>shannon</i> generateEnsemble = <i>GA</i>	Dependent = <i>False</i> generateEnsemble = <i>Optimize</i>	Dependent = <i>False</i> generateEnsemble = <i>GA</i>
Fourier Dönüşümü	86.87±0.35	87.68±0.35	85.86±1.75	85.66±1.26	87.88±0.61	88.48±1.60
Gaussian2	87.07±0.35	87.27±1.05	84.24±1.21	84.65±0.35	86.26±0.35	88.08±0.70
BiorSplines3.3	89.09±0.61	86.46±0.93	88.69±0.35	83.84±0.35	92.73±1.05	87.47±1.26
Coiflets2	88.69±0.35	84.04±0.35	88.48±1.05	85.66±0.93	92.93±1.40	87.68±1.52
Daubechies2	87.27±0.61	84.85±0.61	88.48±1.05	83.43±0.35	92.93±2.13	89.90±1.26
Daubechies3	89.70±0.61	87.27±0.61	90.51±0.93	88.08±0.35	92.93±1.26	89.29±1.52

Çizelge C.41: (devam) VIBES algoritmasının (temel öğrenici olarak KA kullanıldı) farklı parametrelerine göre Parkinson veri kümesi üzerindeki karşılaştırma sonuçları (Herbir deney 3 kere tekrarlandı)

Daubechies4	91.11±1.40	86.87±0.35	88.69±0.70	84.65±0.35	93.33±0.61	88.48±1.05
Daubechies5	88.69±0.35	85.25±0.70	88.28±0.70	86.06±0.61	92.12±1.21	88.89±0.93
Daubechies6	88.69±0.35	85.66±0.70	87.47±0.93	83.03±0.61	93.13±0.93	87.68±1.75
Daubechies7	89.49±1.40	84.85±0.00	88.28±0.35	83.43±0.35	94.34±1.26	89.29±0.70
DMeyer	89.29±0.70	84.85±0.61	86.87±0.70	83.23±0.93	93.33±2.42	87.47±0.70
Haar	89.49±1.40	86.46±0.93	89.09±1.05	86.67±0.61	92.93±1.26	91.52±1.21
Symlets2	87.47±0.35	85.25±0.35	89.29±0.70	84.85±1.21	94.14±0.35	90.51±1.85
Hilbert-Huang Dönüşümü	90.71±0.70	85.66±0.35	89.09±0.61	85.86±0.70	95.76±0.00	94.34±0.70
Ortalama	88.83	85.89	88.09	84.94	92.48	89.22
Yerel Rank	0	0	0	0	12	2

Çizelge C.42: VIBES algoritmasının (temel öğrenici olarak RF kullanıldı) farklı parametrelerine göre Parkinson veri kümesi üzerindeki karşılaştırma sonuçları (Herbir deney 3 kere tekrarlandı)

Veri kümesine Uygulanan Sinyal Dönüşümü	RF					
	Dependent = <i>True</i> Measure = <i>relief</i> generateEnsemble = <i>Optimize</i>	Dependent = <i>True</i> Measure = <i>relief</i> generateEnsemble = <i>GA</i>	Dependent = <i>True</i> Measure = <i>shannon</i> generateEnsemble = <i>Optimize</i>	Dependent = <i>True</i> Measure = <i>shannon</i> generateEnsemble = <i>GA</i>	Dependent = <i>False</i> generateEnsemble = <i>Optimize</i>	Dependent = <i>False</i> generateEnsemble = <i>GA</i>
Fourier Dönüşümü	86.87±0.70	87.07±0.35	87.27±0.61	87.07±0.35	85.66±0.93	86.46±0.35
Gaussian2	87.07±0.93	87.27±0.00	85.05±0.35	84.85±0.00	85.45±1.82	86.67±0.61
BiorSplines3.3	87.47±0.93	83.64±0.00	87.27±0.00	83.84±0.35	90.91±1.05	89.29±1.52
Coiflets2	87.27±0.00	85.45±0.00	86.06±0.61	83.64±0.00	91.92±0.93	88.48±1.05
Daubechies2	85.86±0.35	82.42±0.61	85.66±0.70	83.43±0.35	90.51±0.93	88.08±1.26
Daubechies3	87.68±0.35	85.05±0.35	86.87±0.93	84.65±0.35	92.12±0.61	90.30±0.61

Çizelge C.42: (devam) VIBES algoritmasının (temel öğrenici olarak RF kullanıldı) farklı parametrelerine göre Parkinson veri kümesi üzerindeki karşılaştırma sonuçları (Herbir deney 3 kere tekrarlandı)

Daubechies4	86.87±0.35	84.04±0.70	86.67±0.00	84.24±0.00	92.73±1.05	88.48±0.61
Daubechies5	86.46±0.35	84.04±0.35	87.27±0.61	84.65±0.70	90.10±0.93	88.08±0.93
Daubechies6	87.88±0.61	85.05±0.35	85.45±0.61	82.83±0.35	90.71±0.70	87.47±0.35
Daubechies7	87.68±0.35	85.45±0.00	86.87±0.35	84.44±0.35	92.73±0.61	90.30±1.05
DMeyer	89.09±0.61	86.46±0.35	85.66±0.35	81.01±0.35	91.92±0.70	88.08±1.26
Haar	86.87±0.70	84.24±0.00	86.46±0.35	84.04±0.35	92.93±1.26	90.91±0.61
Symlets2	86.67±0.61	82.63±0.35	85.66±0.35	83.23±0.35	90.91±0.00	89.09±1.05
Hilbert-Huang Dönüşümü	90.91±0.00	88.08±0.35	86.46±0.70	84.85±0.00	95.35±1.26	93.33±1.60
Ortalama	87.48	85.06	86.33	84.06	91.00	88.93
Yerel Rank	0	1	1	0	12	0

Çizelge C.43: VIBES algoritmasının (temel öğrenici olarak k-EYK kullanıldı) HHD uygulanmış Parkinson veri kümesi üzerinde k-değerine göre sınıflandırma sonuçları (Herbir deney 10 kere gerçekleştirildi)

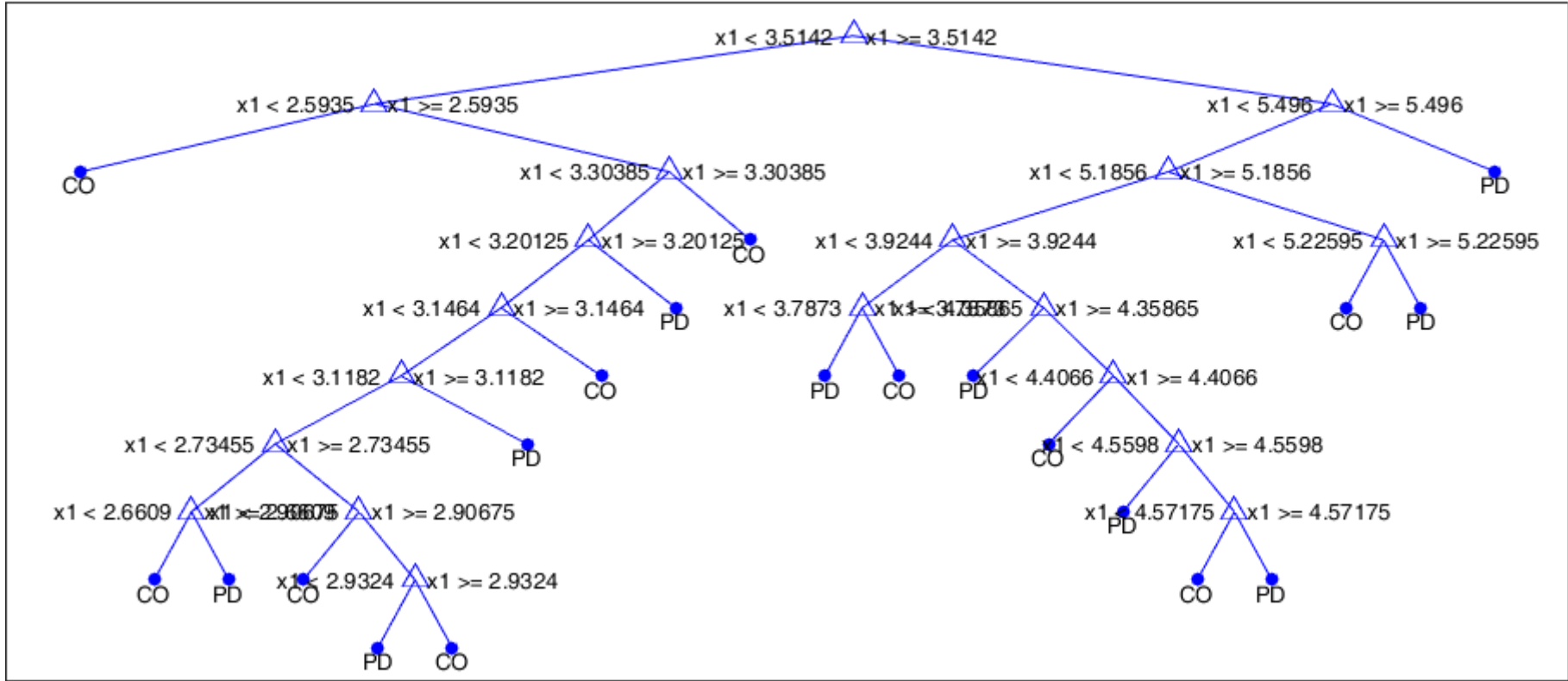
k-değeri	Dependent = <i>False</i> generateEnsemble = <i>Optimize</i>
1	96.3636±1.0690
2	94.0000±1.2600
3	95.0303±0.8944
4	95.1515±1.3092
5	95.0303±1.5064
6	95.6364±1.0613
7	96.0606±1.0400
8	95.9394±0.8595
9	95.9394±0.9918
10	95.5152±0.9561

Çizelge C.44: VIBES algoritmasının (temel öğrenici olarak k-EYK kullanıldı) HHD uygulanmış Parkinson veri kümesi üzerinde k-KÇD'ye göre sınıflandırma sonuçları (Herbir deney 10 kere gerçekleştirildi)

k-kat	Dependent = <i>False</i> generateEnsemble = <i>Optimize</i> k = 1
2	93.8788±1.2920
5	96.1818±0.9057
10	95.4545±1.2856
LOOÇD	95.7576±0.0000
Ortalama	95.3182

Çizelge C.45: VIBES algoritmasının (temel öğrenici olarak KA kullanıldı) HHD uygulanmış Parkinson veri kümesi üzerinde k-KÇD'ye göre sınıflandırma sonuçları (Herbir deney 10 kere gerçekleştirildi)

k-kat	Dependent = <i>False</i> generateEnsemble = <i>Optimize</i>
2	95.4545±1.0785
5	95.9394±1.3717
10	95.4545±0.7693
LOOÇD	97.5758±0.0000
Ortalama	96.1061



Şekil C.1: HHD'li Parkinson veri kümesi için 43. özneliğe göre oluşturulan karar ağacı modeli (Öznelikler Shannon entropiye göre sıralandı)

```
1 if x1<3.5142 then node 2 elseif x1>=3.5142 then node 3 else PD
2 if x1<2.5935 then node 4 elseif x1>=2.5935 then node 5 else CO
3 if x1<5.496 then node 6 elseif x1>=5.496 then node 7 else PD
4 class = CO
5 if x1<3.30385 then node 8 elseif x1>=3.30385 then node 9 else CO
6 if x1<5.1856 then node 10 elseif x1>=5.1856 then node 11 else PD
7 class = PD
8 if x1<3.20125 then node 12 elseif x1>=3.20125 then node 13 else CO
9 class = CO
10 if x1<3.9244 then node 14 elseif x1>=3.9244 then node 15 else PD
11 if x1<5.22595 then node 16 elseif x1>=5.22595 then node 17 else CO
12 if x1<3.1464 then node 18 elseif x1>=3.1464 then node 19 else CO
13 class = PD
14 if x1<3.7873 then node 20 elseif x1>=3.7873 then node 21 else PD
15 if x1<4.35865 then node 22 elseif x1>=4.35865 then node 23 else PD
16 class = CO
17 class = PD
18 if x1<3.1182 then node 24 elseif x1>=3.1182 then node 25 else CO
19 class = CO
20 class = PD
21 class = CO
22 class = PD
23 if x1<4.4066 then node 26 elseif x1>=4.4066 then node 27 else PD
24 if x1<2.73455 then node 28 elseif x1>=2.73455 then node 29 else CO
25 class = PD
26 class = CO
27 if x1<4.5598 then node 30 elseif x1>=4.5598 then node 31 else PD
28 if x1<2.6609 then node 32 elseif x1>=2.6609 then node 33 else CO
29 if x1<2.90675 then node 34 elseif x1>=2.90675 then node 35 else CO
30 class = PD
31 if x1<4.57175 then node 36 elseif x1>=4.57175 then node 37 else PD
32 class = CO
33 class = PD
34 class = CO
35 if x1<2.9324 then node 38 elseif x1>=2.9324 then node 39 else CO
36 class = CO
37 class = PD
38 class = PD
39 class = CO
```

Şekil C.2: HHD'li Parkinson veri kümesi için 43. özneliğe göre oluşturulan karar ağacı kural modeli (Öznelikler Shannon entropiye göre sıralandı)


```

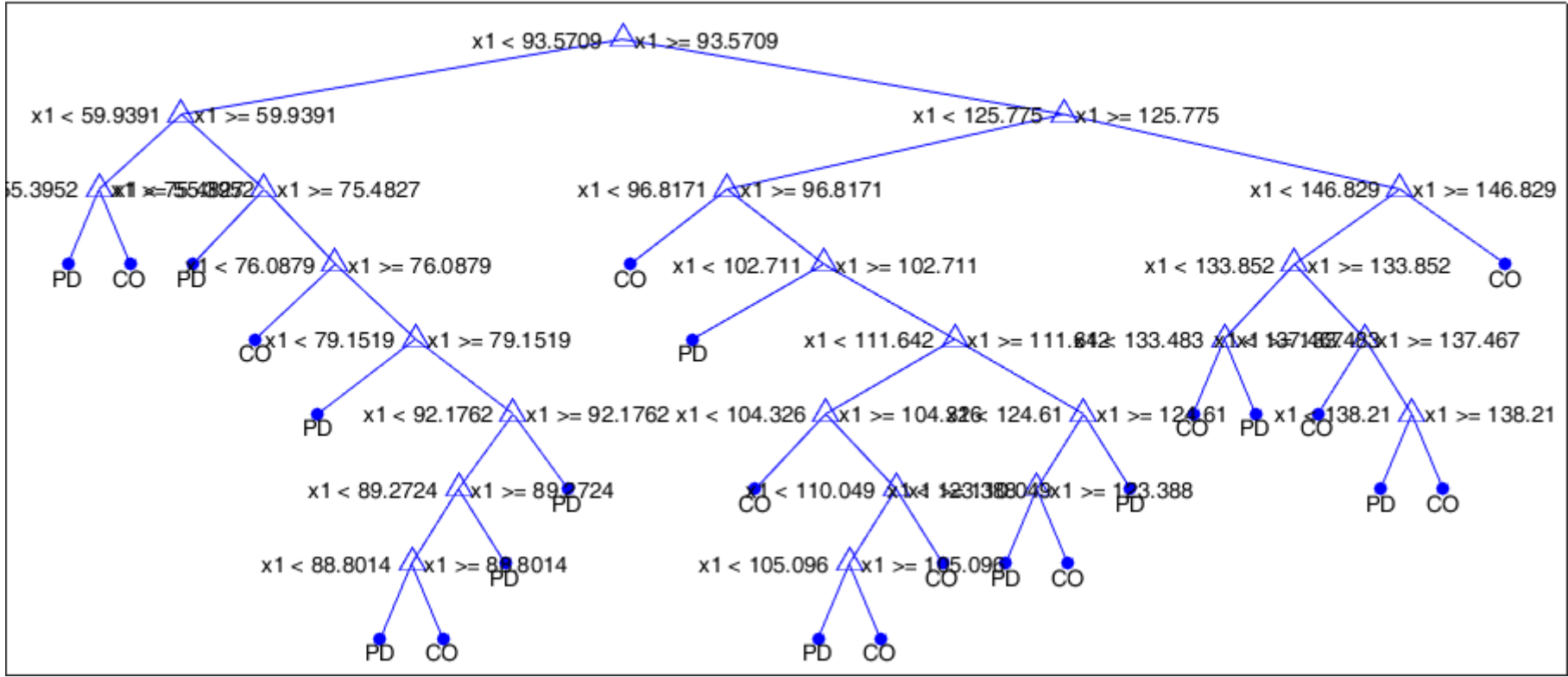
1 if x1<9.8255 then node 2 elseif x1>=9.8255 then node 3 else PD
2 if x1<7.74645 then node 4 elseif x1>=7.74645 then node 5 else PD
3 if x1<9.98575 then node 6 elseif x1>=9.98575 then node 7 else CO
4 if x1<5.9471 then node 8 elseif x1>=5.9471 then node 9 else PD
5 if x1<8.11315 then node 10 elseif x1>=8.11315 then node 11 else PD
6 class = CO
7 if x1<10.0046 then node 12 elseif x1>=10.0046 then node 13 else CO
8 class = PD
9 if x1<6.00255 then node 14 elseif x1>=6.00255 then node 15 else PD
10 class = CO
11 if x1<8.88975 then node 16 elseif x1>=8.88975 then node 17 else PD
12 class = PD
13 if x1<10.7694 then node 18 elseif x1>=10.7694 then node 19 else CO
14 class = CO
15 if x1<6.6902 then node 20 elseif x1>=6.6902 then node 21 else PD
16 if x1<8.51215 then node 22 elseif x1>=8.51215 then node 23 else PD
17 if x1<9.4012 then node 24 elseif x1>=9.4012 then node 25 else PD
18 if x1<10.0819 then node 26 elseif x1>=10.0819 then node 27 else CO
19 if x1<14.9896 then node 28 elseif x1>=14.9896 then node 29 else CO
20 class = PD
21 if x1<6.81695 then node 30 elseif x1>=6.81695 then node 31 else PD
22 if x1<8.2105 then node 32 elseif x1>=8.2105 then node 33 else PD
23 class = CO
24 class = PD
25 if x1<9.47255 then node 34 elseif x1>=9.47255 then node 35 else PD
26 class = CO
27 if x1<10.1108 then node 36 elseif x1>=10.1108 then node 37 else CO
28 if x1<14.3486 then node 38 elseif x1>=14.3486 then node 39 else CO
29 class = CO
30 class = CO
31 if x1<7.5201 then node 40 elseif x1>=7.5201 then node 41 else PD
32 class = PD
33 if x1<8.26175 then node 42 elseif x1>=8.26175 then node 43 else PD
34 class = CO
35 class = PD
36 class = PD
37 if x1<10.5926 then node 44 elseif x1>=10.5926 then node 45 else CO
38 if x1<10.9209 then node 46 elseif x1>=10.9209 then node 47 else CO
39 class = PD
40 if x1<7.4154 then node 48 elseif x1>=7.4154 then node 49 else PD
41 class = PD
42 class = CO
43 class = PD
44 if x1<10.4599 then node 50 elseif x1>=10.4599 then node 51 else CO
45 class = PD
46 class = CO
47 if x1<10.9483 then node 52 elseif x1>=10.9483 then node 53 else CO
48 class = PD
49 class = CO

```

Şekil C.4: HHD’li Parkinson veri kümesi için 68. özniteliğe göre oluşturulan karar ağacı kural modeli (Öz nitelikler Shannon entropiye göre sıralandı)

```
50 if x1<10.4534 then node 54 elseif x1>=10.4534 then node 55 else CO
51 class = CO
52 class = PD
53 if x1<14.1316 then node 56 elseif x1>=14.1316 then node 57 else CO
54 class = CO
55 class = PD
56 if x1<13.929 then node 58 elseif x1>=13.929 then node 59 else CO
57 class = CO
58 if x1<13.0927 then node 60 elseif x1>=13.0927 then node 61 else CO
59 class = PD
60 if x1<12.8543 then node 62 elseif x1>=12.8543 then node 63 else CO
61 class = CO
62 if x1<11.9583 then node 64 elseif x1>=11.9583 then node 65 else CO
63 class = PD
64 if x1<11.8289 then node 66 elseif x1>=11.8289 then node 67 else CO
65 class = CO
66 class = CO
67 class = PD
```

Şekil C.4: HHD'li Parkinson veri kümesi için 68. özniteliğe göre oluşturulan karar ağacı kural modeli (Öznitelikler Shannon entropiye göre sıralandı)



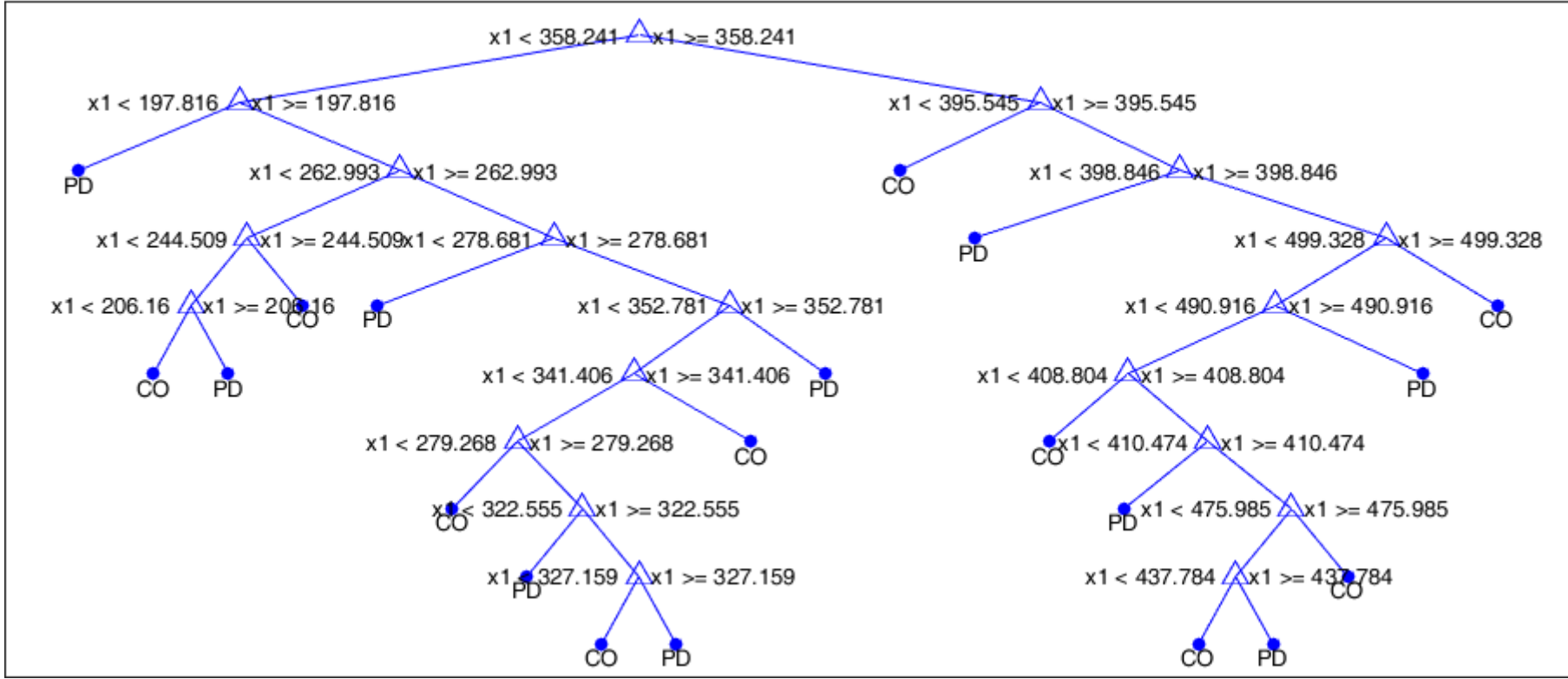
Şekil C.5: HHD'li Parkinson veri kümesi için 127. özneliğe göre oluşturulan karar ağacı modeli (Öznelikler Shannon entropiye göre sıralandı)

```

1 if x1<93.5709 then node 2 elseif x1>=93.5709 then node 3 else PD
2 if x1<59.9391 then node 4 elseif x1>=59.9391 then node 5 else PD
3 if x1<125.775 then node 6 elseif x1>=125.775 then node 7 else CO
4 if x1<55.3952 then node 8 elseif x1>=55.3952 then node 9 else PD
5 if x1<75.4827 then node 10 elseif x1>=75.4827 then node 11 else PD
6 if x1<96.8171 then node 12 elseif x1>=96.8171 then node 13 else CO
7 if x1<146.829 then node 14 elseif x1>=146.829 then node 15 else CO
8 class = PD
9 class = CO
10 class = PD
11 if x1<76.0879 then node 16 elseif x1>=76.0879 then node 17 else PD
12 class = CO
13 if x1<102.711 then node 18 elseif x1>=102.711 then node 19 else PD
14 if x1<133.852 then node 20 elseif x1>=133.852 then node 21 else CO
15 class = CO
16 class = CO
17 if x1<79.1519 then node 22 elseif x1>=79.1519 then node 23 else PD
18 class = PD
19 if x1<111.642 then node 24 elseif x1>=111.642 then node 25 else CO
20 if x1<133.483 then node 26 elseif x1>=133.483 then node 27 else CO
21 if x1<137.467 then node 28 elseif x1>=137.467 then node 29 else CO
22 class = PD
23 if x1<92.1762 then node 30 elseif x1>=92.1762 then node 31 else PD
24 if x1<104.326 then node 32 elseif x1>=104.326 then node 33 else CO
25 if x1<124.61 then node 34 elseif x1>=124.61 then node 35 else PD
26 class = CO
27 class = PD
28 class = CO
29 if x1<138.21 then node 36 elseif x1>=138.21 then node 37 else CO
30 if x1<89.2724 then node 38 elseif x1>=89.2724 then node 39 else PD
31 class = PD
32 class = CO
33 if x1<110.049 then node 40 elseif x1>=110.049 then node 41 else CO
34 if x1<123.388 then node 42 elseif x1>=123.388 then node 43 else PD
35 class = PD
36 class = PD
37 class = CO
38 if x1<88.8014 then node 44 elseif x1>=88.8014 then node 45 else PD
39 class = PD
40 if x1<105.096 then node 46 elseif x1>=105.096 then node 47 else PD
41 class = CO
42 class = PD
43 class = CO
44 class = PD
45 class = CO
46 class = PD
47 class = CO

```

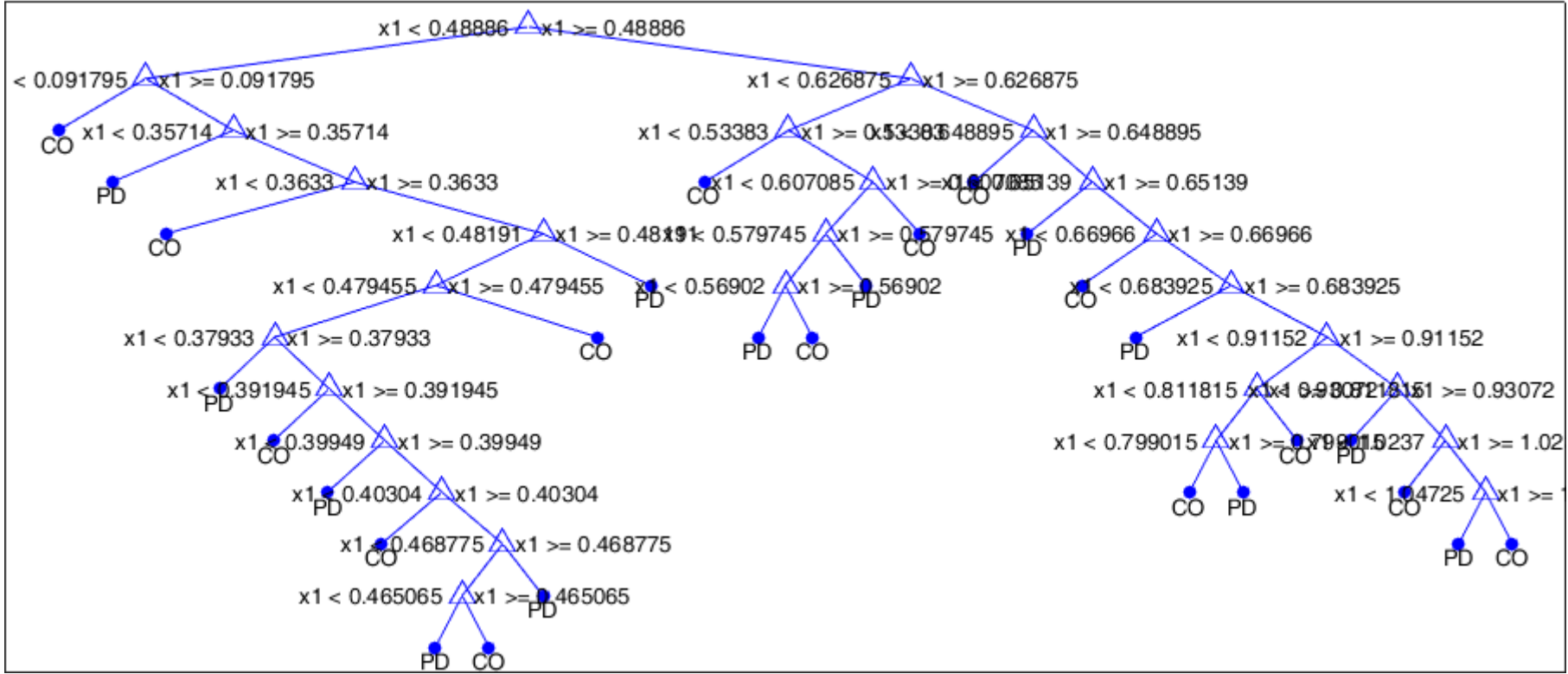
Şekil C.6: HHD'li Parkinson veri kümesi için 127. özneliğe göre oluşturulan karar ağacı kural modeli (Öznelikler Shannon entropiye göre sıralandı)



Şekil C.7: HHD'li Parkinson veri kümesi için 3. özneliğe göre oluşturulan karar ağacı modeli (Öznitelikler Shannon entropiye göre sıralandı)


```
1 if x1<358.241 then node 2 elseif x1>=358.241 then node 3 else PD
2 if x1<197.816 then node 4 elseif x1>=197.816 then node 5 else PD
3 if x1<395.545 then node 6 elseif x1>=395.545 then node 7 else CO
4 class = PD
5 if x1<262.993 then node 8 elseif x1>=262.993 then node 9 else PD
6 class = CO
7 if x1<398.846 then node 10 elseif x1>=398.846 then node 11 else CO
8 if x1<244.509 then node 12 elseif x1>=244.509 then node 13 else CO
9 if x1<278.681 then node 14 elseif x1>=278.681 then node 15 else PD
10 class = PD
11 if x1<499.328 then node 16 elseif x1>=499.328 then node 17 else CO
12 if x1<206.16 then node 18 elseif x1>=206.16 then node 19 else PD
13 class = CO
14 class = PD
15 if x1<352.781 then node 20 elseif x1>=352.781 then node 21 else PD
16 if x1<490.916 then node 22 elseif x1>=490.916 then node 23 else CO
17 class = CO
18 class = CO
19 class = PD
20 if x1<341.406 then node 24 elseif x1>=341.406 then node 25 else PD
21 class = PD
22 if x1<408.804 then node 26 elseif x1>=408.804 then node 27 else CO
23 class = PD
24 if x1<279.268 then node 28 elseif x1>=279.268 then node 29 else PD
25 class = CO
26 class = CO
27 if x1<410.474 then node 30 elseif x1>=410.474 then node 31 else CO
28 class = CO
29 if x1<322.555 then node 32 elseif x1>=322.555 then node 33 else PD
30 class = PD
31 if x1<475.985 then node 34 elseif x1>=475.985 then node 35 else CO
32 class = PD
33 if x1<327.159 then node 36 elseif x1>=327.159 then node 37 else PD
34 if x1<437.784 then node 38 elseif x1>=437.784 then node 39 else CO
35 class = CO
36 class = CO
37 class = PD
38 class = CO
39 class = PD
```

Şekil C.8: HHD'li Parkinson veri kümesi için 3. özniteliğe göre oluşturulan karar ağacı kural modeli (Öznitelikler Shannon entropiye göre sıralandı)



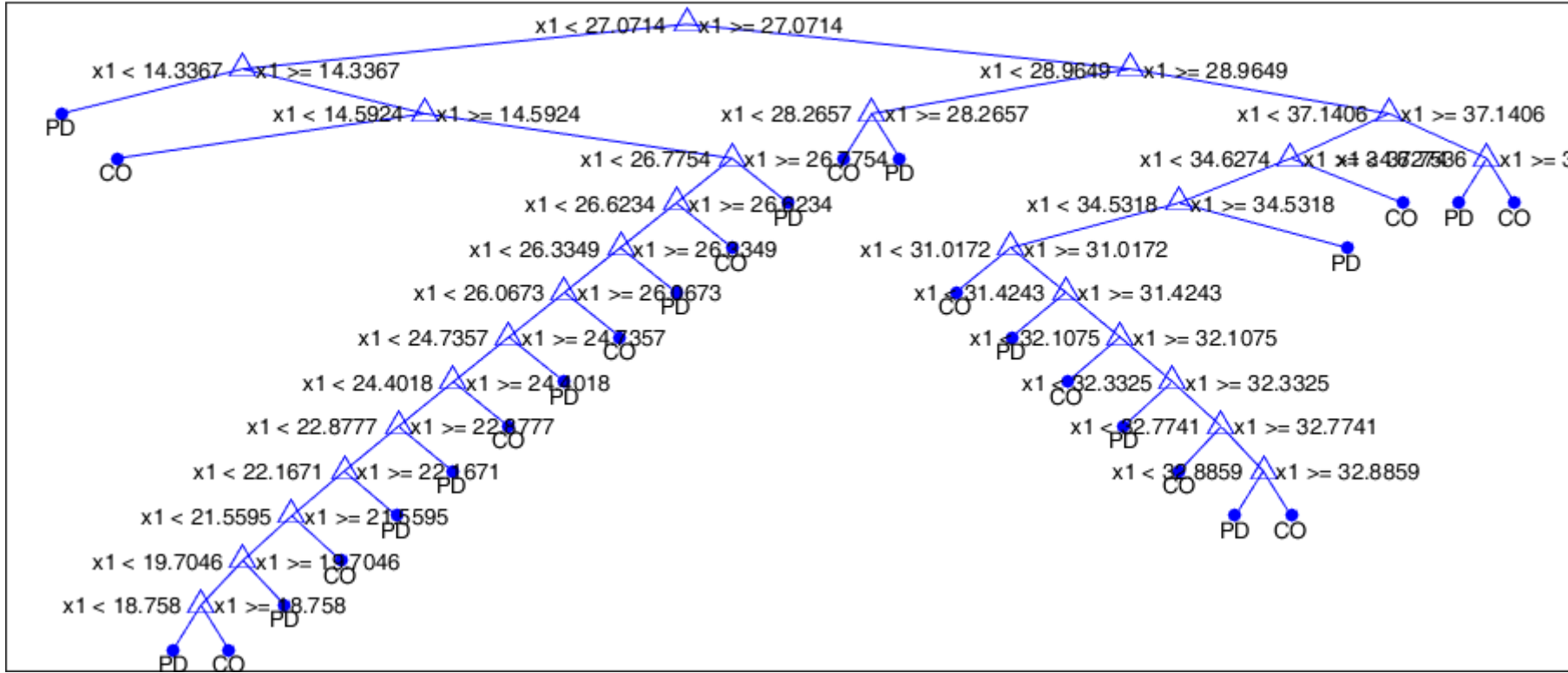
Şekil C.9: HHD'li Parkinson veri kümesi için 111. özneliğe göre oluşturulan karar ağacı modeli (Öznelikler Shannon entropiye göre sıralandı)

```

1 if x1<0.48886 then node 2 elseif x1>=0.48886 then node 3 else PD
2 if x1<0.091795 then node 4 elseif x1>=0.091795 then node 5 else PD
3 if x1<0.626875 then node 6 elseif x1>=0.626875 then node 7 else CO
4 class = CO
5 if x1<0.35714 then node 8 elseif x1>=0.35714 then node 9 else PD
6 if x1<0.53383 then node 10 elseif x1>=0.53383 then node 11 else PD
7 if x1<0.648895 then node 12 elseif x1>=0.648895 then node 13 else CO
8 class = PD
9 if x1<0.3633 then node 14 elseif x1>=0.3633 then node 15 else PD
10 class = CO
11 if x1<0.607085 then node 16 elseif x1>=0.607085 then node 17 else PD
12 class = CO
13 if x1<0.65139 then node 18 elseif x1>=0.65139 then node 19 else CO
14 class = CO
15 if x1<0.48191 then node 20 elseif x1>=0.48191 then node 21 else PD
16 if x1<0.579745 then node 22 elseif x1>=0.579745 then node 23 else PD
17 class = CO
18 class = PD
19 if x1<0.66966 then node 24 elseif x1>=0.66966 then node 25 else CO
20 if x1<0.479455 then node 26 elseif x1>=0.479455 then node 27 else PD
21 class = PD
22 if x1<0.56902 then node 28 elseif x1>=0.56902 then node 29 else PD
23 class = PD
24 class = CO
25 if x1<0.683925 then node 30 elseif x1>=0.683925 then node 31 else CO
26 if x1<0.37933 then node 32 elseif x1>=0.37933 then node 33 else PD
27 class = CO
28 class = PD
29 class = CO
30 class = PD
31 if x1<0.91152 then node 34 elseif x1>=0.91152 then node 35 else CO
32 class = PD
33 if x1<0.391945 then node 36 elseif x1>=0.391945 then node 37 else PD
34 if x1<0.811815 then node 38 elseif x1>=0.811815 then node 39 else CO
35 if x1<0.93072 then node 40 elseif x1>=0.93072 then node 41 else CO
36 class = CO
37 if x1<0.39949 then node 42 elseif x1>=0.39949 then node 43 else PD
38 if x1<0.799015 then node 44 elseif x1>=0.799015 then node 45 else CO
39 class = CO
40 class = PD
41 if x1<1.0237 then node 46 elseif x1>=1.0237 then node 47 else CO
42 class = PD
43 if x1<0.40304 then node 48 elseif x1>=0.40304 then node 49 else PD
44 class = CO
45 class = PD
46 class = CO
47 if x1<1.04725 then node 50 elseif x1>=1.04725 then node 51 else CO
48 class = CO
49 if x1<0.468775 then node 52 elseif x1>=0.468775 then node 53 else PD
50 class = PD
51 class = CO
52 if x1<0.465065 then node 54 elseif x1>=0.465065 then node 55 else PD
53 class = PD
54 class = PD
55 class = CO

```

Şekil C.10: HHD'li Parkinson veri kümesi için 111. öz niteliğe göre oluşturulan karar ağacı kural modeli (Öznitelikler Shannon entropiye göre sıralandı)



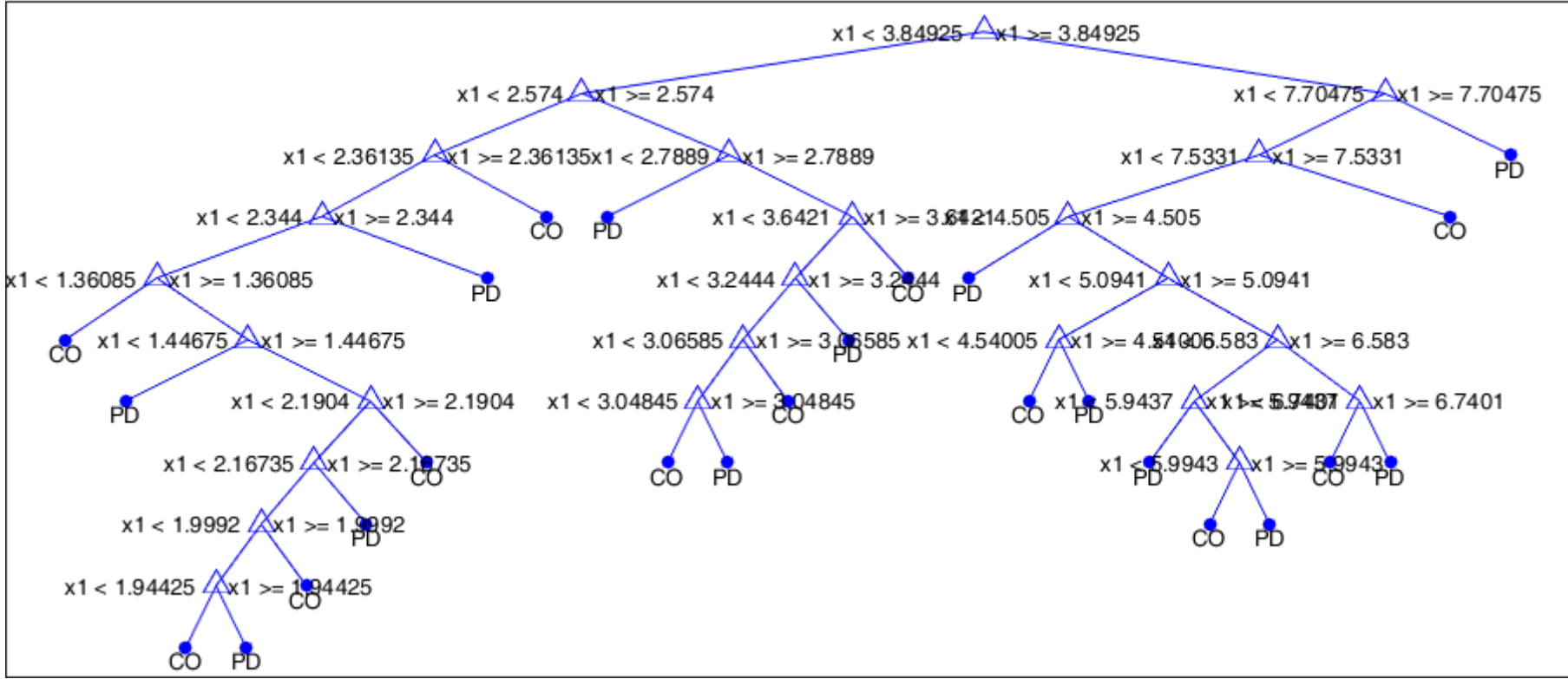
Şekil C.11: HHD'li Parkinson veri kümesi için 99. özneliğe göre oluşturulan karar ağacı modeli (Öznelikler Shannon entropiye göre sıralandı)

```

1 if x1<27.0714 then node 2 elseif x1>=27.0714 then node 3 else PD
2 if x1<14.3367 then node 4 elseif x1>=14.3367 then node 5 else PD
3 if x1<28.9649 then node 6 elseif x1>=28.9649 then node 7 else CO
4 class = PD
5 if x1<14.5924 then node 8 elseif x1>=14.5924 then node 9 else PD
6 if x1<28.2657 then node 10 elseif x1>=28.2657 then node 11 else CO
7 if x1<37.1406 then node 12 elseif x1>=37.1406 then node 13 else CO
8 class = CO
9 if x1<26.7754 then node 14 elseif x1>=26.7754 then node 15 else PD
10 class = CO
11 class = PD
12 if x1<34.6274 then node 16 elseif x1>=34.6274 then node 17 else CO
13 if x1<37.7536 then node 18 elseif x1>=37.7536 then node 19 else CO
14 if x1<26.6234 then node 20 elseif x1>=26.6234 then node 21 else PD
15 class = PD
16 if x1<34.5318 then node 22 elseif x1>=34.5318 then node 23 else CO
17 class = CO
18 class = PD
19 class = CO
20 if x1<26.3349 then node 24 elseif x1>=26.3349 then node 25 else PD
21 class = CO
22 if x1<31.0172 then node 26 elseif x1>=31.0172 then node 27 else CO
23 class = PD
24 if x1<26.0673 then node 28 elseif x1>=26.0673 then node 29 else PD
25 class = PD
26 class = CO
27 if x1<31.4243 then node 30 elseif x1>=31.4243 then node 31 else CO
28 if x1<24.7357 then node 32 elseif x1>=24.7357 then node 33 else PD
29 class = CO
30 class = PD
31 if x1<32.1075 then node 34 elseif x1>=32.1075 then node 35 else CO
32 if x1<24.4018 then node 36 elseif x1>=24.4018 then node 37 else PD
33 class = PD
34 class = CO
35 if x1<32.3325 then node 38 elseif x1>=32.3325 then node 39 else CO
36 if x1<22.8777 then node 40 elseif x1>=22.8777 then node 41 else PD
37 class = CO
38 class = PD
39 if x1<32.7741 then node 42 elseif x1>=32.7741 then node 43 else CO
40 if x1<22.1671 then node 44 elseif x1>=22.1671 then node 45 else PD
41 class = PD
42 class = CO
43 if x1<32.8859 then node 46 elseif x1>=32.8859 then node 47 else CO
44 if x1<21.5595 then node 48 elseif x1>=21.5595 then node 49 else PD
45 class = PD
46 class = PD
47 class = CO
48 if x1<19.7046 then node 50 elseif x1>=19.7046 then node 51 else PD
49 class = CO
50 if x1<18.758 then node 52 elseif x1>=18.758 then node 53 else PD
51 class = PD
52 class = PD
53 class = CO

```

Şekil C.12: HHD'li Parkinson veri kümesi için 99. özneteliğe göre oluşturulan karar ağacı kural modeli (Öznetelikler Shannon entropiye göre sıralandı)



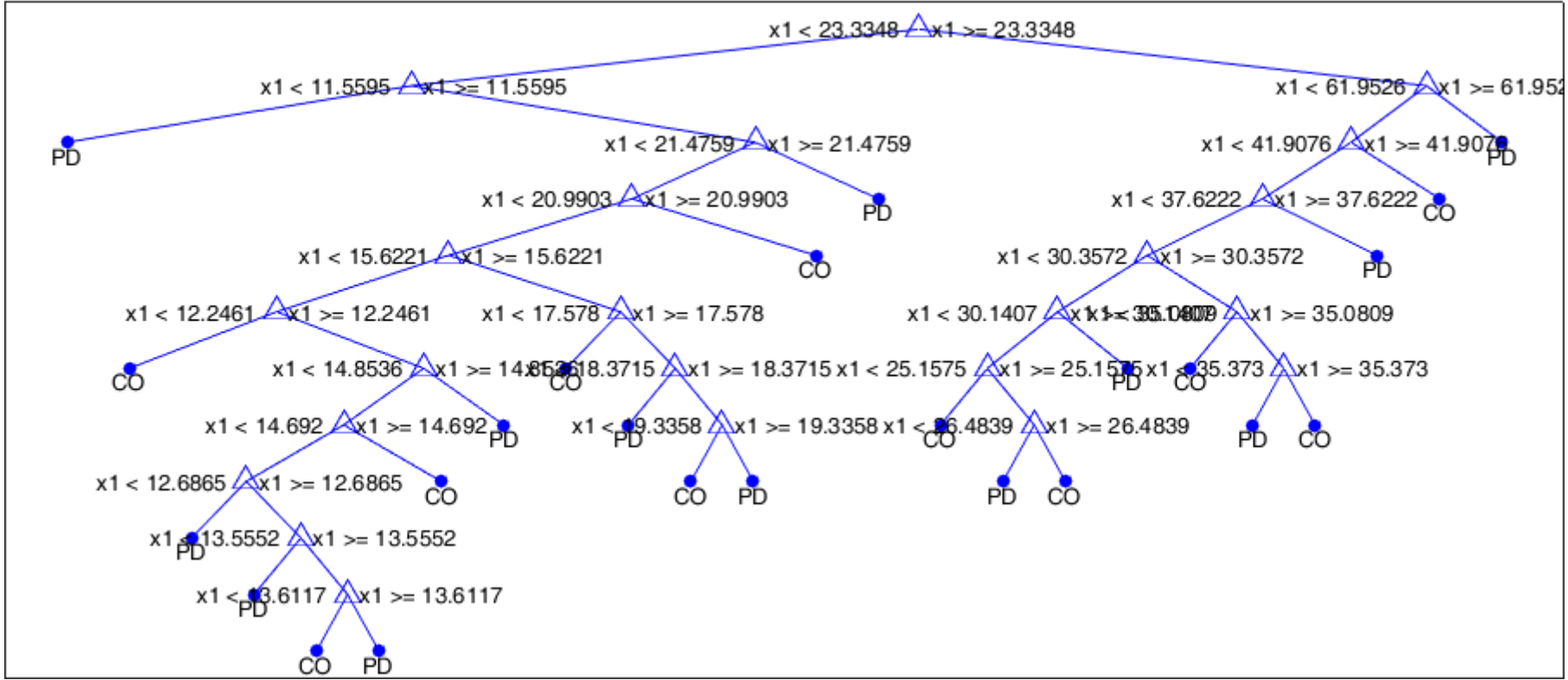
Şekil C.13: HHD'li Parkinson veri kümesi için 126. özneliğe göre oluşturulan karar ağacı modeli (Öznitelikler Shannon entropiye göre sıralandı)

```

1 if x1<3.84925 then node 2 elseif x1>=3.84925 then node 3 else PD
2 if x1<2.574 then node 4 elseif x1>=2.574 then node 5 else CO
3 if x1<7.70475 then node 6 elseif x1>=7.70475 then node 7 else PD
4 if x1<2.36135 then node 8 elseif x1>=2.36135 then node 9 else CO
5 if x1<2.7889 then node 10 elseif x1>=2.7889 then node 11 else CO
6 if x1<7.5331 then node 12 elseif x1>=7.5331 then node 13 else PD
7 class = PD
8 if x1<2.344 then node 14 elseif x1>=2.344 then node 15 else CO
9 class = CO
10 class = PD
11 if x1<3.6421 then node 16 elseif x1>=3.6421 then node 17 else CO
12 if x1<4.505 then node 18 elseif x1>=4.505 then node 19 else PD
13 class = CO
14 if x1<1.36085 then node 20 elseif x1>=1.36085 then node 21 else CO
15 class = PD
16 if x1<3.2444 then node 22 elseif x1>=3.2444 then node 23 else CO
17 class = CO
18 class = PD
19 if x1<5.0941 then node 24 elseif x1>=5.0941 then node 25 else PD
20 class = CO
21 if x1<1.44675 then node 26 elseif x1>=1.44675 then node 27 else CO
22 if x1<3.06585 then node 28 elseif x1>=3.06585 then node 29 else CO
23 class = PD
24 if x1<4.54005 then node 30 elseif x1>=4.54005 then node 31 else CO
25 if x1<6.583 then node 32 elseif x1>=6.583 then node 33 else PD
26 class = PD
27 if x1<2.1904 then node 34 elseif x1>=2.1904 then node 35 else CO
28 if x1<3.04845 then node 36 elseif x1>=3.04845 then node 37 else CO
29 class = CO
30 class = CO
31 class = PD
32 if x1<5.9437 then node 38 elseif x1>=5.9437 then node 39 else PD
33 if x1<6.7401 then node 40 elseif x1>=6.7401 then node 41 else PD
34 if x1<2.16735 then node 42 elseif x1>=2.16735 then node 43 else CO
35 class = CO
36 class = CO
37 class = PD
38 class = PD
39 if x1<5.9943 then node 44 elseif x1>=5.9943 then node 45 else PD
40 class = CO
41 class = PD
42 if x1<1.9992 then node 46 elseif x1>=1.9992 then node 47 else CO
43 class = PD
44 class = CO
45 class = PD
46 if x1<1.94425 then node 48 elseif x1>=1.94425 then node 49 else CO
47 class = CO
48 class = CO
49 class = PD

```

Şekil C.14: HHD’li Parkinson veri kümesi için 126. özneliğe göre oluşturulan karar ağacı kural modeli (Öznelikler Shannon entropiye göre sıralandı)



Şekil C.15: HHD'li Parkinson veri kümesi için 13. özneliğe göre oluşturulan karar ağacı modeli (Öznelikler Shannon entropiye göre sıralandı)


```

1 if x1<23.3348 then node 2 elseif x1>=23.3348 then node 3 else PD
2 if x1<11.5595 then node 4 elseif x1>=11.5595 then node 5 else PD
3 if x1<61.9526 then node 6 elseif x1>=61.9526 then node 7 else CO
4 class = PD
5 if x1<21.4759 then node 8 elseif x1>=21.4759 then node 9 else PD
6 if x1<41.9076 then node 10 elseif x1>=41.9076 then node 11 else CO
7 class = PD
8 if x1<20.9903 then node 12 elseif x1>=20.9903 then node 13 else PD
9 class = PD
10 if x1<37.6222 then node 14 elseif x1>=37.6222 then node 15 else CO
11 class = CO
12 if x1<15.6221 then node 16 elseif x1>=15.6221 then node 17 else PD
13 class = CO
14 if x1<30.3572 then node 18 elseif x1>=30.3572 then node 19 else CO
15 class = PD
16 if x1<12.2461 then node 20 elseif x1>=12.2461 then node 21 else PD
17 if x1<17.578 then node 22 elseif x1>=17.578 then node 23 else PD
18 if x1<30.1407 then node 24 elseif x1>=30.1407 then node 25 else CO
19 if x1<35.0809 then node 26 elseif x1>=35.0809 then node 27 else CO
20 class = CO
21 if x1<14.8536 then node 28 elseif x1>=14.8536 then node 29 else PD
22 class = CO
23 if x1<18.3715 then node 30 elseif x1>=18.3715 then node 31 else PD
24 if x1<25.1575 then node 32 elseif x1>=25.1575 then node 33 else CO
25 class = PD
26 class = CO
27 if x1<35.373 then node 34 elseif x1>=35.373 then node 35 else CO
28 if x1<14.692 then node 36 elseif x1>=14.692 then node 37 else PD
29 class = PD
30 class = PD
31 if x1<19.3358 then node 38 elseif x1>=19.3358 then node 39 else PD
32 class = CO
33 if x1<26.4839 then node 40 elseif x1>=26.4839 then node 41 else CO
34 class = PD
35 class = CO
36 if x1<12.6865 then node 42 elseif x1>=12.6865 then node 43 else PD
37 class = CO
38 class = CO
39 class = PD
40 class = PD
41 class = CO
42 class = PD
43 if x1<13.5552 then node 44 elseif x1>=13.5552 then node 45 else PD
44 class = PD
45 if x1<13.6117 then node 46 elseif x1>=13.6117 then node 47 else PD
46 class = CO
47 class = PD

```

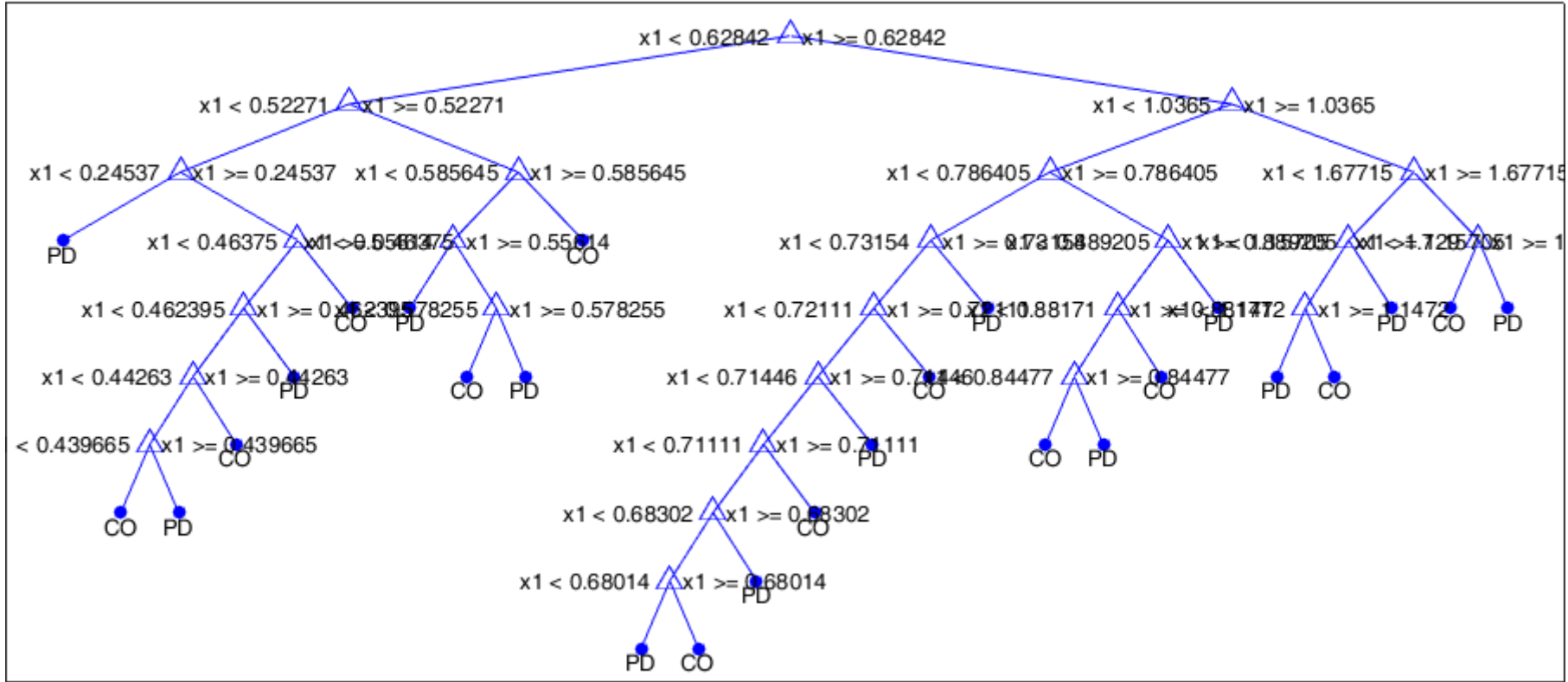
Şekil C.16: HHD'li Parkinson veri kümesi için 13. özneliğe göre oluşturulan karar ağacı kural modeli (Öznelikler Shannon entropiye göre sıralandı)


```

1 if x1<19.117 then node 2 elseif x1>=19.117 then node 3 else PD
2 if x1<11.7213 then node 4 elseif x1>=11.7213 then node 5 else PD
3 if x1<23.056 then node 6 elseif x1>=23.056 then node 7 else CO
4 class = PD
5 if x1<11.8075 then node 8 elseif x1>=11.8075 then node 9 else PD
6 if x1<22.4419 then node 10 elseif x1>=22.4419 then node 11 else PD
7 if x1<32.1073 then node 12 elseif x1>=32.1073 then node 13 else CO
8 class = CO
9 if x1<17.1105 then node 14 elseif x1>=17.1105 then node 15 else PD
10 if x1<22.1754 then node 16 elseif x1>=22.1754 then node 17 else CO
11 class = PD
12 if x1<28.9924 then node 18 elseif x1>=28.9924 then node 19 else CO
13 if x1<38.4681 then node 20 elseif x1>=38.4681 then node 21 else CO
14 if x1<16.7505 then node 22 elseif x1>=16.7505 then node 23 else PD
15 class = PD
16 if x1<21.7704 then node 24 elseif x1>=21.7704 then node 25 else CO
17 class = CO
18 if x1<23.5427 then node 26 elseif x1>=23.5427 then node 27 else CO
19 class = CO
20 if x1<33.269 then node 28 elseif x1>=33.269 then node 29 else PD
21 class = CO
22 class = PD
23 class = CO
24 if x1<21.1111 then node 30 elseif x1>=21.1111 then node 31 else CO
25 class = PD
26 class = CO
27 if x1<23.6448 then node 32 elseif x1>=23.6448 then node 33 else CO
28 class = PD
29 if x1<34.0443 then node 34 elseif x1>=34.0443 then node 35 else PD
30 if x1<20.3565 then node 36 elseif x1>=20.3565 then node 37 else CO
31 class = CO
32 class = PD
33 if x1<28.9536 then node 38 elseif x1>=28.9536 then node 39 else CO
34 class = CO
35 if x1<34.6604 then node 40 elseif x1>=34.6604 then node 41 else PD
36 class = CO
37 class = PD
38 if x1<25.1119 then node 42 elseif x1>=25.1119 then node 43 else CO
39 class = PD
40 class = PD
41 if x1<35.2781 then node 44 elseif x1>=35.2781 then node 45 else PD
42 class = CO
43 if x1<25.4767 then node 46 elseif x1>=25.4767 then node 47 else CO
44 class = CO
45 class = PD
46 class = PD
47 if x1<28.8075 then node 48 elseif x1>=28.8075 then node 49 else CO
48 if x1<28.6307 then node 50 elseif x1>=28.6307 then node 51 else CO
49 class = CO
50 if x1<27.8363 then node 52 elseif x1>=27.8363 then node 53 else CO
51 class = PD
52 if x1<26.8375 then node 54 elseif x1>=26.8375 then node 55 else CO
53 class = CO
54 class = CO
55 class = PD

```

Şekil C.18: HHD'li Parkinson veri kümesi için 35. özniteliğe göre oluşturulan karar ağacı kural modeli (Öznitelikler Shannon entropiye göre sıralandı)



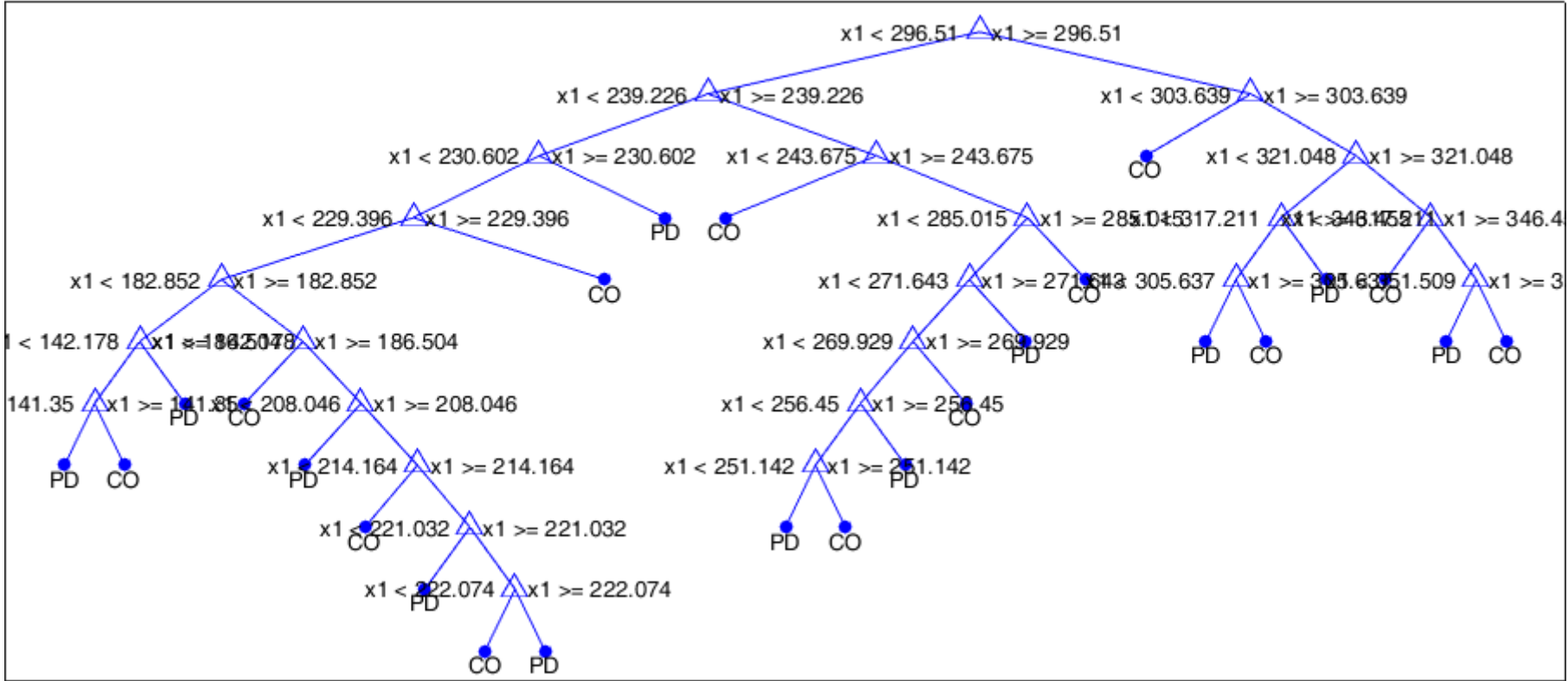
Şekil C.19: HHD'li Parkinson veri kümesi için 131. özneliğe göre oluşturulan karar ağacı modeli (Öznelikler Shannon entropiye göre sıralandı)

```

1 if x1<0.62842 then node 2 elseif x1>=0.62842 then node 3 else PD
2 if x1<0.52271 then node 4 elseif x1>=0.52271 then node 5 else CO
3 if x1<1.0365 then node 6 elseif x1>=1.0365 then node 7 else PD
4 if x1<0.24537 then node 8 elseif x1>=0.24537 then node 9 else CO
5 if x1<0.585645 then node 10 elseif x1>=0.585645 then node 11 else CO
6 if x1<0.786405 then node 12 elseif x1>=0.786405 then node 13 else PD
7 if x1<1.67715 then node 14 elseif x1>=1.67715 then node 15 else PD
8 class = PD
9 if x1<0.46375 then node 16 elseif x1>=0.46375 then node 17 else CO
10 if x1<0.55614 then node 18 elseif x1>=0.55614 then node 19 else PD
11 class = CO
12 if x1<0.73154 then node 20 elseif x1>=0.73154 then node 21 else PD
13 if x1<0.889205 then node 22 elseif x1>=0.889205 then node 23 else PD
14 if x1<1.15705 then node 24 elseif x1>=1.15705 then node 25 else PD
15 if x1<1.729 then node 26 elseif x1>=1.729 then node 27 else PD
16 if x1<0.462395 then node 28 elseif x1>=0.462395 then node 29 else CO
17 class = CO
18 class = PD
19 if x1<0.578255 then node 30 elseif x1>=0.578255 then node 31 else PD
20 if x1<0.72111 then node 32 elseif x1>=0.72111 then node 33 else PD
21 class = PD
22 if x1<0.88171 then node 34 elseif x1>=0.88171 then node 35 else CO
23 class = PD
24 if x1<1.1472 then node 36 elseif x1>=1.1472 then node 37 else PD
25 class = PD
26 class = CO
27 class = PD
28 if x1<0.44263 then node 38 elseif x1>=0.44263 then node 39 else CO
29 class = PD
30 class = CO
31 class = PD
32 if x1<0.71446 then node 40 elseif x1>=0.71446 then node 41 else PD
33 class = CO
34 if x1<0.84477 then node 42 elseif x1>=0.84477 then node 43 else CO
35 class = CO
36 class = PD
37 class = CO
38 if x1<0.439665 then node 44 elseif x1>=0.439665 then node 45 else CO
39 class = CO
40 if x1<0.71111 then node 46 elseif x1>=0.71111 then node 47 else PD
41 class = PD
42 class = CO
43 class = PD
44 class = CO
45 class = PD
46 if x1<0.68302 then node 48 elseif x1>=0.68302 then node 49 else PD
47 class = CO
48 if x1<0.68014 then node 50 elseif x1>=0.68014 then node 51 else PD
49 class = PD
50 class = PD
51 class = CO

```

Şekil C.20: HHD’li Parkinson veri kümesi için 131. öz niteliğe göre oluşturulan karar ağacı kural modeli (Öznitelikler Shannon entropiye göre sıralandı)



Şekil C.21: HHD'li Parkinson veri kümesi için 45. özneliğe göre oluşturulan karar ağacı modeli (Öznelikler Shannon entropiye göre sıralandı)

```
1 if x1<296.51 then node 2 elseif x1>=296.51 then node 3 else PD
2 if x1<239.226 then node 4 elseif x1>=239.226 then node 5 else PD
3 if x1<303.639 then node 6 elseif x1>=303.639 then node 7 else CO
4 if x1<230.602 then node 8 elseif x1>=230.602 then node 9 else PD
5 if x1<243.675 then node 10 elseif x1>=243.675 then node 11 else PD
6 class = CO
7 if x1<321.048 then node 12 elseif x1>=321.048 then node 13 else CO
8 if x1<229.396 then node 14 elseif x1>=229.396 then node 15 else PD
9 class = PD
10 class = CO
11 if x1<285.015 then node 16 elseif x1>=285.015 then node 17 else PD
12 if x1<317.211 then node 18 elseif x1>=317.211 then node 19 else CO
13 if x1<346.455 then node 20 elseif x1>=346.455 then node 21 else CO
14 if x1<182.852 then node 22 elseif x1>=182.852 then node 23 else PD
15 class = CO
16 if x1<271.643 then node 24 elseif x1>=271.643 then node 25 else PD
17 class = CO
18 if x1<305.637 then node 26 elseif x1>=305.637 then node 27 else CO
19 class = PD
20 class = CO
21 if x1<351.509 then node 28 elseif x1>=351.509 then node 29 else CO
22 if x1<142.178 then node 30 elseif x1>=142.178 then node 31 else PD
23 if x1<186.504 then node 32 elseif x1>=186.504 then node 33 else PD
24 if x1<269.929 then node 34 elseif x1>=269.929 then node 35 else PD
25 class = PD
26 class = PD
27 class = CO
28 class = PD
29 class = CO
30 if x1<141.35 then node 36 elseif x1>=141.35 then node 37 else PD
31 class = PD
32 class = CO
33 if x1<208.046 then node 38 elseif x1>=208.046 then node 39 else PD
34 if x1<256.45 then node 40 elseif x1>=256.45 then node 41 else PD
35 class = CO
36 class = PD
37 class = CO
38 class = PD
39 if x1<214.164 then node 42 elseif x1>=214.164 then node 43 else PD
40 if x1<251.142 then node 44 elseif x1>=251.142 then node 45 else PD
41 class = PD
42 class = CO
43 if x1<221.032 then node 46 elseif x1>=221.032 then node 47 else PD
44 class = PD
45 class = CO
46 class = PD
47 if x1<222.074 then node 48 elseif x1>=222.074 then node 49 else PD
48 class = CO
49 class = PD
```

Şekil C.22: HHD'li Parkinson veri kümesi için 45. özniteliğe göre oluşturulan karar ağacı kural modeli (Öznitelikler Shannon entropiye göre sıralandı)


```
1 if x1<-0.52695 then node 2 elseif x1>=-0.52695 then node 3 else PD
2 if x1<-1.09325 then node 4 elseif x1>=-1.09325 then node 5 else CO
3 if x1<-0.110075 then node 6 elseif x1>=-0.110075 then node 7 else PD
4 if x1<-1.2891 then node 8 elseif x1>=-1.2891 then node 9 else CO
5 if x1<-0.622645 then node 10 elseif x1>=-0.622645 then node 11 else CO
6 if x1<-0.140345 then node 12 elseif x1>=-0.140345 then node 13 else PD
7 class = PD
8 if x1<-1.29745 then node 14 elseif x1>=-1.29745 then node 15 else CO
9 class = CO
10 if x1<-0.80511 then node 16 elseif x1>=-0.80511 then node 17 else PD
11 class = CO
12 if x1<-0.457915 then node 18 elseif x1>=-0.457915 then node 19 else PD
13 class = CO
14 if x1<-1.3905 then node 20 elseif x1>=-1.3905 then node 21 else CO
15 class = PD
16 if x1<-0.866325 then node 22 elseif x1>=-0.866325 then node 23 else CO
17 class = PD
18 class = PD
19 if x1<-0.44477 then node 24 elseif x1>=-0.44477 then node 25 else PD
20 if x1<-1.439 then node 26 elseif x1>=-1.439 then node 27 else CO
21 class = CO
22 if x1<-0.918175 then node 28 elseif x1>=-0.918175 then node 29 else PD
23 class = CO
24 class = CO
25 if x1<-0.19895 then node 30 elseif x1>=-0.19895 then node 31 else PD
26 if x1<-1.96575 then node 32 elseif x1>=-1.96575 then node 33 else CO
27 class = PD
28 if x1<-1.00282 then node 34 elseif x1>=-1.00282 then node 35 else PD
29 class = PD
30 if x1<-0.40633 then node 36 elseif x1>=-0.40633 then node 37 else PD
31 class = PD
32 class = CO
33 if x1<-1.6882 then node 38 elseif x1>=-1.6882 then node 39 else CO
34 class = PD
35 class = CO
36 class = PD
37 if x1<-0.3572 then node 40 elseif x1>=-0.3572 then node 41 else PD
38 class = PD
39 class = CO
40 class = PD
41 if x1<-0.347765 then node 42 elseif x1>=-0.347765 then node 43 else PD
42 class = CO
43 class = PD
```

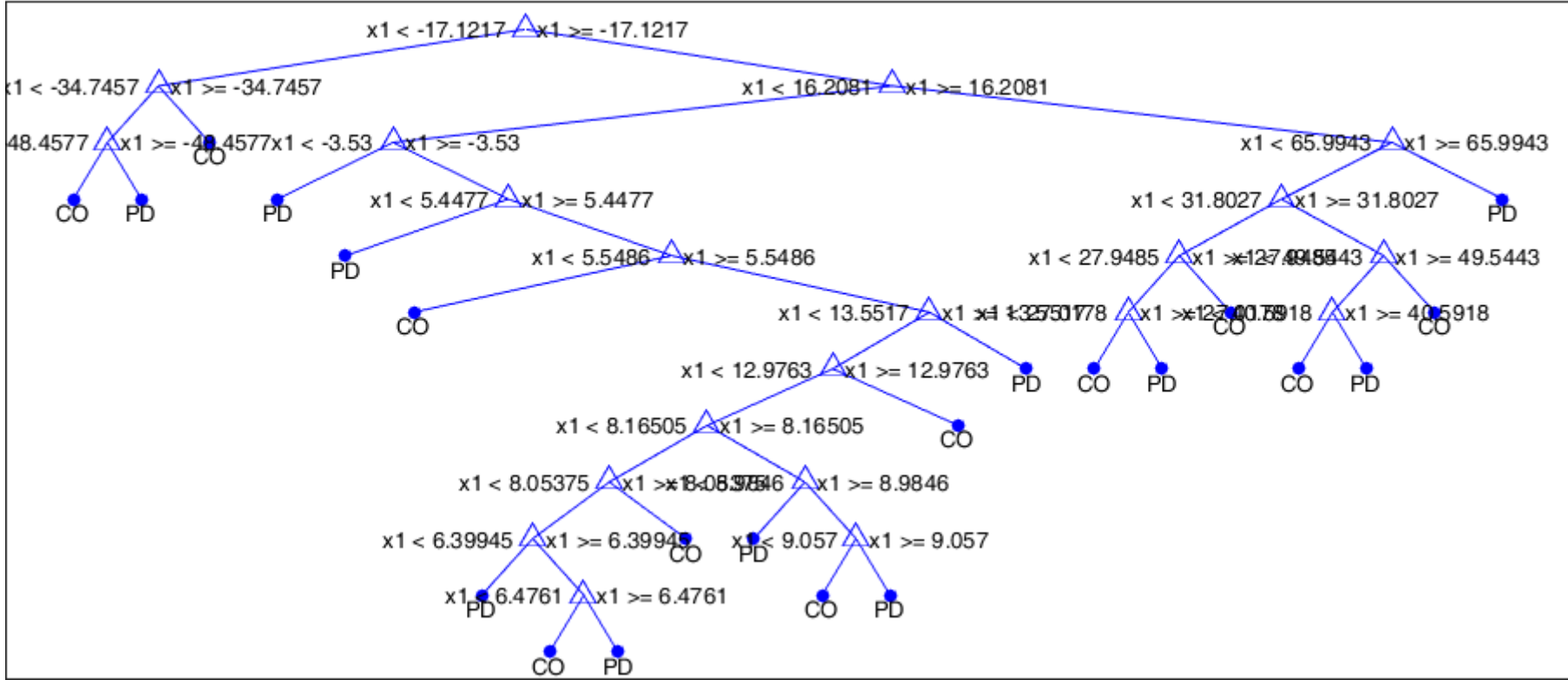
Şekil C.24: HHD'li Parkinson veri kümesi için 129. özneliğe göre oluşturulan karar ağacı kural modeli (Öznelikler Shannon entropiye göre sıralandı)


```

1 if x1<31926.3 then node 2 elseif x1>=31926.3 then node 3 else PD
2 if x1<4254.19 then node 4 elseif x1>=4254.19 then node 5 else PD
3 if x1<83436.7 then node 6 elseif x1>=83436.7 then node 7 else CO
4 if x1<-974.71 then node 8 elseif x1>=-974.71 then node 9 else PD
5 if x1<7132.01 then node 10 elseif x1>=7132.01 then node 11 else PD
6 if x1<38142.6 then node 12 elseif x1>=38142.6 then node 13 else CO
7 class = PD
8 if x1<-1649.67 then node 14 elseif x1>=-1649.67 then node 15 else PD
9 class = PD
10 class = CO
11 if x1<26635.3 then node 16 elseif x1>=26635.3 then node 17 else PD
12 class = CO
13 if x1<38396.5 then node 18 elseif x1>=38396.5 then node 19 else CO
14 class = PD
15 class = CO
16 if x1<19821.6 then node 20 elseif x1>=19821.6 then node 21 else PD
17 if x1<27839.8 then node 22 elseif x1>=27839.8 then node 23 else PD
18 class = PD
19 if x1<68171.5 then node 24 elseif x1>=68171.5 then node 25 else CO
20 if x1<19199.4 then node 26 elseif x1>=19199.4 then node 27 else PD
21 if x1<22614.3 then node 28 elseif x1>=22614.3 then node 29 else PD
22 class = CO
23 class = PD
24 if x1<65176.1 then node 30 elseif x1>=65176.1 then node 31 else CO
25 class = CO
26 if x1<8261.17 then node 32 elseif x1>=8261.17 then node 33 else PD
27 class = CO
28 class = PD
29 if x1<22986 then node 34 elseif x1>=22986 then node 35 else PD
30 if x1<40075.8 then node 36 elseif x1>=40075.8 then node 37 else CO
31 class = PD
32 class = PD
33 if x1<9843.9 then node 38 elseif x1>=9843.9 then node 39 else PD
34 class = CO
35 if x1<23251 then node 40 elseif x1>=23251 then node 41 else PD
36 class = CO
37 if x1<45232.3 then node 42 elseif x1>=45232.3 then node 43 else CO
38 class = CO
39 if x1<12527.4 then node 44 elseif x1>=12527.4 then node 45 else PD
40 class = PD
41 if x1<23350.5 then node 46 elseif x1>=23350.5 then node 47 else PD
42 if x1<44081 then node 48 elseif x1>=44081 then node 49 else CO
43 if x1<55053.1 then node 50 elseif x1>=55053.1 then node 51 else CO
44 class = PD
45 if x1<12986.1 then node 52 elseif x1>=12986.1 then node 53 else PD
46 class = CO
47 class = PD
48 if x1<40522.3 then node 54 elseif x1>=40522.3 then node 55 else CO
49 class = PD
50 class = CO
51 if x1<56668 then node 56 elseif x1>=56668 then node 57 else CO
52 class = CO
53 class = PD
54 class = PD
55 class = CO
56 class = PD
57 class = CO

```

Şekil C.26: HHD'li Parkinson veri kümesi için 81. özniteliğe göre oluşturulan karar ağacı kural modeli (Öznitelikler Shannon entropiye göre sıralandı)



Şekil C.27: HHD'li Parkinson veri kümesi için 87. özneliğe göre oluşturulan karar ağacı modeli (Öznelikler Shannon entropiye göre sıralandı)

```

1 if x1<-17.1217 then node 2 elseif x1>=-17.1217 then node 3 else PD
2 if x1<-34.7457 then node 4 elseif x1>=-34.7457 then node 5 else CO
3 if x1<16.2081 then node 6 elseif x1>=16.2081 then node 7 else PD
4 if x1<-48.4577 then node 8 elseif x1>=-48.4577 then node 9 else CO
5 class = CO
6 if x1<-3.53 then node 10 elseif x1>=-3.53 then node 11 else PD
7 if x1<65.9943 then node 12 elseif x1>=65.9943 then node 13 else CO
8 class = CO
9 class = PD
10 class = PD
11 if x1<5.4477 then node 14 elseif x1>=5.4477 then node 15 else PD
12 if x1<31.8027 then node 16 elseif x1>=31.8027 then node 17 else CO
13 class = PD
14 class = PD
15 if x1<5.5486 then node 18 elseif x1>=5.5486 then node 19 else PD
16 if x1<27.9485 then node 20 elseif x1>=27.9485 then node 21 else CO
17 if x1<49.5443 then node 22 elseif x1>=49.5443 then node 23 else CO
18 class = CO
19 if x1<13.5517 then node 24 elseif x1>=13.5517 then node 25 else PD
20 if x1<27.0178 then node 26 elseif x1>=27.0178 then node 27 else CO
21 class = CO
22 if x1<40.5918 then node 28 elseif x1>=40.5918 then node 29 else PD
23 class = CO
24 if x1<12.9763 then node 30 elseif x1>=12.9763 then node 31 else PD
25 class = PD
26 class = CO
27 class = PD
28 class = CO
29 class = PD
30 if x1<8.16505 then node 32 elseif x1>=8.16505 then node 33 else PD
31 class = CO
32 if x1<8.05375 then node 34 elseif x1>=8.05375 then node 35 else PD
33 if x1<8.9846 then node 36 elseif x1>=8.9846 then node 37 else PD
34 if x1<6.39945 then node 38 elseif x1>=6.39945 then node 39 else PD
35 class = CO
36 class = PD
37 if x1<9.057 then node 40 elseif x1>=9.057 then node 41 else PD
38 class = PD
39 if x1<6.4761 then node 42 elseif x1>=6.4761 then node 43 else PD
40 class = CO
41 class = PD
42 class = CO
43 class = PD

```

Şekil C.28: HHD'li Parkinson veri kümesi için 87. özneliğe göre oluşturulan karar ağacı kural modeli (Öznelikler Shannon entropiye göre sıralandı)

Çizelge C.46: FFD uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)

Gizli Katman Sayısı	Eğitim (%)	Doğrulama (%)	Test (%)	Tümü (%)
10	83.10±3.75	84.00±7.20	81.87±7.84	83.05±2.34
50	83.54±3.38	83.33±7.28	82.13±9.73	83.29±2.11
100	83.39±4.05	81.07±6.88	82.53±7.89	82.91±3.39
150	83.36±2.47	84.27±6.88	82.13±5.73	83.31±1.81
200	82.20±2.50	82.93±9.15	<u>83.73±6.96</u>	82.55±2.34
250	83.65±2.31	86.00±6.10	81.60±7.47	83.70±1.47
300	83.91±3.15	83.33±6.81	78.80±6.14	83.05±2.65
350	82.43±3.96	85.20±8.21	80.27±7.50	82.53±3.04
400	82.78±3.66	84.00±8.53	79.20±8.50	82.42±3.00
Ortalama	83.15	83.79	81.36	82.98

Çizelge C.47: Gaussian2 uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)

Gizli Katman Sayısı	Eğitim (%)	Doğrulama (%)	Test (%)	Tümü (%)
10	84.23±3.60	83.07±7.91	79.73±6.64	83.37±2.89
50	83.59±4.16	82.53±7.75	77.87±7.77	82.57±3.65
100	83.68±5.54	84.67±6.48	78.67±9.70	83.07±4.91
150	83.48±2.87	82.93±6.12	80.80±6.16	82.99±2.66
200	84.81±2.95	82.67±9.59	77.73±9.38	83.41±2.30
250	83.19±5.73	80.53±6.10	<u>82.13±9.26</u>	82.63±4.85
300	82.41±4.98	80.13±8.17	79.60±7.60	81.64±4.36
350	83.86±2.11	82.40±6.61	80.67±8.86	83.15±2.02
400	83.71±4.75	80.53±8.65	81.60±6.53	82.91±3.76
Ortalama	83.66	82.16	79.87	82.86

Çizelge C.48: BiorSplines3.3 uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)

Gizli Katman Sayısı	Eğitim (%)	Doğrulama (%)	Test (%)	Tümü (%)
10	84.75±3.44	86.53±7.08	82.80±6.74	84.73±2.64
50	85.65±3.75	82.80±7.14	80.40±8.29	84.42±2.59
100	86.00±4.36	85.20±6.74	80.80±7.08	85.09±3.50
150	85.54±3.08	82.67±5.59	81.60±9.37	84.51±1.89
200	85.19±4.22	85.20±7.51	<u>83.73±5.65</u>	84.97±2.85
250	86.29±3.69	84.93±6.86	82.27±6.27	85.47±2.83
300	86.09±2.66	81.60±5.72	80.13±6.34	84.51±2.29
350	84.61±2.80	84.27±6.12	83.07±6.45	84.32±2.37
400	86.32±4.07	82.13±7.98	83.20±5.79	85.21±2.92
Ortalama	85.60	83.93	82.00	84.80

Çizelge C.49: Coiflets2 uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)

Gizli Katman Sayısı	Eğitim (%)	Doğrulama (%)	Test (%)	Tümü (%)
10	85.10±3.18	84.13±6.34	82.67±7.81	84.59±1.80
50	84.55±3.34	83.20±7.31	83.60±6.92	84.20±2.91
100	84.58±3.42	83.87±8.63	83.20±5.40	84.26±2.83
150	85.36±4.60	86.40±5.81	80.13±8.69	84.73±3.14
200	84.23±4.97	83.60±6.92	82.13±7.91	83.82±4.40
250	84.55±3.25	82.93±6.96	82.00±8.39	83.92±2.60
300	84.55±3.38	83.60±6.75	81.73±7.25	83.98±2.52
350	82.90±4.65	82.80±6.32	82.93±5.94	82.89±3.30
400	84.78±3.46	83.73±6.96	82.67±8.23	84.30±2.78
Ortalama	84.51	83.81	82.34	84.08

Çizelge C.50: Daubechies2 uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)

Gizli Katman Sayısı	Eğitim (%)	Doğrulama (%)	Test (%)	Tümü (%)
10	85.88±3.93	84.67±7.13	80.93±7.91	84.95±2.94
50	85.07±5.08	84.40±7.15	80.27±6.88	84.24±4.15
100	84.96±3.82	82.53±6.52	82.00±6.45	84.14±2.80
150	84.32±4.64	82.80±8.80	80.40±7.00	83.49±3.79
200	84.93±4.87	82.80±7.51	80.13±7.08	83.88±3.93
250	85.57±3.12	84.00±6.48	79.87±9.37	84.46±3.02
300	84.23±4.97	86.67±8.49	81.87±6.70	84.24±3.88
350	84.75±3.39	82.93±8.40	<u>82.53±8.17</u>	84.14±1.99
400	85.80±6.12	86.13±7.91	80.80±5.50	85.09±5.04
Ortalama	85.06	84.10	80.98	84.29

Çizelge C.51: Daubechies3 uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)

Gizli Katman Sayısı	Eğitim (%)	Doğrulama (%)	Test (%)	Tümü (%)
10	86.17±2.84	84.40±6.84	<u>84.67±6.97</u>	85.68±2.38
50	86.12±5.05	85.33±6.83	83.33±7.94	85.58±4.17
100	87.71±3.95	87.20±7.53	83.07±6.45	86.93±3.62
150	84.99±5.75	84.13±8.76	82.53±7.54	84.48±5.23
200	86.35±3.09	85.87±6.45	81.07±9.15	85.47±2.80
250	86.87±2.82	83.47±7.91	84.53±6.95	86.00±2.33
300	85.86±3.60	85.33±6.67	83.47±9.14	85.41±3.36
350	85.91±4.52	84.67±8.41	84.27±7.27	85.47±3.07
400	86.75±2.09	85.87±7.18	83.60±5.79	86.14±1.72
Ortalama	86.30	85.14	83.39	85.68

Çizelge C.52: Daubechies4 uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)

Gizli Katman Sayısı	Eğitim (%)	Doğrulama (%)	Test (%)	Tümü (%)
10	84.84±2.97	81.73±7.40	80.67±5.05	83.74±2.33
50	84.09±4.72	82.67±7.74	80.27±7.35	83.29±3.86
100	83.80±4.36	82.53±7.75	80.67±7.13	83.13±3.19
150	83.04±4.78	83.73±6.21	80.67±7.94	82.79±3.84
200	84.12±4.27	82.27±7.33	<u>82.40±6.94</u>	83.58±3.33
250	84.14±2.90	80.53±7.33	80.00±6.64	82.97±2.03
300	83.19±5.83	81.73±7.33	80.80±7.68	82.61±5.09
350	83.51±3.02	81.87±6.87	81.73±7.02	82.99±2.52
400	84.84±3.45	83.60±6.42	80.93±6.62	84.06±2.04
Ortalama	83.95	82.30	80.90	83.24

Çizelge C.53: Daubechies5 uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)

Gizli Katman Sayısı	Eğitim (%)	Doğrulama (%)	Test (%)	Tümü (%)
10	84.67±3.92	85.07±6.21	81.73±8.05	84.28±2.67
50	84.75±4.52	83.87±6.77	82.27±6.01	84.24±3.01
100	83.97±4.62	83.20±8.36	81.60±6.78	83.49±4.18
150	85.57±3.57	86.00±5.83	<u>83.33±7.21</u>	85.29±2.34
200	84.55±3.38	84.27±6.64	82.00±5.92	84.12±2.73
250	84.70±3.60	85.60±7.55	80.80±8.69	84.24±2.98
300	84.55±3.35	83.60±9.24	81.87±8.71	84.00±2.53
350	83.57±2.90	83.47±7.26	82.27±6.10	83.35±2.35
400	84.38±2.99	82.00±9.32	80.13±6.68	83.37±2.46
Ortalama	84.52	84.12	81.78	84.04

Çizelge C.54: Daubechies6 uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)

Gizli Katman Sayısı	Eğitim (%)	Doğrulama (%)	Test (%)	Tümü (%)
10	84.09±1.99	84.27±5.84	82.67±6.33	83.90±1.64
50	84.52±3.28	83.73±6.03	84.27±8.27	84.36±2.13
100	85.30±3.24	83.73±7.12	80.80±6.92	84.38±2.22
150	84.67±4.77	84.40±6.84	81.33±7.67	84.12±3.60
200	84.72±2.05	83.47±6.87	82.13±5.43	84.14±1.49
250	85.13±5.31	84.80±7.68	78.80±9.29	84.12±4.55
300	84.41±2.71	82.53±6.77	84.13±7.46	84.08±1.52
350	84.43±2.60	82.13±6.79	80.93±7.10	83.56±1.76
400	84.29±2.88	82.53±7.46	83.60±8.49	83.92±2.35
Ortalama	84.62	83.51	82.07	84.06

Çizelge C.55: Daubechies7 uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)

Gizli Katman Sayısı	Eğitim (%)	Doğrulama (%)	Test (%)	Tümü (%)
10	87.19±4.34	86.67±6.59	80.80±6.76	86.14±3.01
50	85.88±4.09	83.60±6.51	82.53±6.93	85.03±3.59
100	86.06±3.09	84.67±7.21	84.00±7.20	85.54±2.14
150	86.99±6.50	84.93±8.25	81.87±6.95	85.90±5.15
200	85.91±3.50	83.87±6.77	83.60±6.33	85.25±2.37
250	85.71±3.01	84.27±7.86	81.73±8.12	84.89±2.02
300	86.32±3.70	82.80±7.36	82.40±7.97	85.19±2.71
350	86.84±2.66	84.53±7.18	82.53±5.99	85.84±1.85
400	86.81±3.21	85.33±5.59	82.53±6.60	85.94±2.21
Ortalama	86.41	84.52	82.44	85.52

Çizelge C.56: DMeyer uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)

Gizli Katman Sayısı	Eğitim (%)	Doğrulama (%)	Test (%)	Tümü (%)
10	88.52±4.49	86.80±7.94	82.67±8.29	87.37±3.74
50	89.19±5.01	85.20±7.44	82.13±7.41	87.52±4.19
100	89.22±4.26	87.87±7.54	83.60±6.51	88.16±3.54
150	88.43±4.80	84.93±8.83	81.60±6.61	86.87±4.41
200	88.06±6.01	84.93±6.27	81.07±9.27	86.53±4.99
250	87.68±5.39	85.73±7.98	83.33±6.31	86.73±4.51
300	86.81±5.02	82.80±7.73	<u>85.07±7.27</u>	85.94±3.77
350	88.64±4.19	84.80±8.03	83.87±7.82	87.33±3.46
400	87.01±3.72	84.67±6.04	81.73±7.40	85.86±2.62
Ortalama	88.17	85.30	82.79	86.92

Çizelge C.57: Haar uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)

Gizli Katman Sayısı	Eğitim (%)	Doğrulama (%)	Test (%)	Tümü (%)
10	87.07±2.95	86.13±8.90	<u>84.13±6.85</u>	86.48±2.43
50	89.10±3.18	86.93±7.27	84.00±5.25	88.00±2.40
100	88.26±2.55	86.53±6.60	81.73±8.71	87.01±1.83
150	86.43±4.38	86.67±6.91	83.87±7.16	86.08±3.57
200	86.32±3.60	86.93±5.55	83.33±7.51	85.96±3.57
250	87.07±3.64	84.00±9.16	82.53±5.89	85.92±2.94
300	86.78±4.11	85.07±7.04	83.07±6.62	85.96±3.45
350	87.86±2.55	83.73±6.72	83.87±9.42	86.63±2.00
400	87.68±2.93	86.13±5.82	82.13±8.45	86.61±2.24
Ortalama	87.40	85.79	83.18	86.52

Çizelge C.58: Symlets2 uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)

Gizli Katman Sayısı	Eğitim (%)	Doğrulama (%)	Test (%)	Tümü (%)
10	84.75±5.39	83.33±7.94	79.60±6.75	83.76±4.79
50	85.42±3.74	84.93±7.33	79.47±6.54	84.44±2.77
100	85.25±5.79	81.73±7.25	<u>82.80±7.21</u>	84.34±4.14
150	85.54±3.56	84.80±7.89	81.20±6.90	84.77±2.45
200	85.04±4.14	83.87±7.61	82.00±7.91	84.40±3.07
250	85.45±3.52	83.33±7.36	81.33±8.81	84.51±2.81
300	84.87±3.44	85.87±6.28	80.53±7.56	84.36±2.43
350	84.61±2.59	82.93±7.42	81.60±8.64	83.90±2.09
400	84.78±4.63	83.87±8.37	81.73±5.43	84.18±3.79
Ortalama	85.08	83.85	81.14	84.30

Çizelge C.59: HHD uygulanmış, CfsSubsetEval ile özellik seçimi yapılmış ve normalize edilmiş Parkinson veri kümesi üzerinde Yapay Sinir Ağı'nın sınıflandırma sonuçları (Verinin %70'i eğitim, %15'i doğrulama ve %15'i test için kullanılmıştır. Herbir deney 30 kere gerçekleştirildi)

Gizli Katman Sayısı	Eğitim (%)	Doğrulama (%)	Test (%)	Tümü (%)
10	84.70±3.53	81.60±7.09	<u>81.07±8.59</u>	83.68±2.81
50	84.75±4.47	81.60±7.02	78.80±6.49	83.37±3.31
100	85.13±4.93	81.07±6.96	76.67±7.87	83.23±3.84
150	83.94±4.30	82.93±8.85	79.20±8.88	83.07±3.13
200	83.91±6.04	81.33±9.06	80.67±8.28	83.03±4.84
250	83.74±3.90	80.00±8.85	78.80±6.23	82.42±2.96
300	83.57±4.38	84.13±7.08	76.93±10.22	82.65±3.47
350	83.88±4.46	80.00±7.72	80.93±8.05	82.85±3.65
400	83.13±6.14	82.00±7.41	76.53±7.70	81.96±4.64
Ortalama	84.08	81.63	78.84	82.92



ÖZGEÇMİŞ

Ad-Soyad : Fatih AYDIN

Doğum Tarihi ve Yeri : 1982 - Malatya

E-posta : fatih.aydin@klu.edu.tr

ÖĞRENİM DURUMU:

- **Lisans** : 2004, Fırat Üniversitesi, Mühendislik Fakültesi, Bilgisayar Müh.
- **Yüksek Lisans** : 2011, Trakya Üniversitesi, Bilgisayar Müh. Anabilim Dalı

MESLEKİ DENEYİM VE ÖDÜLLER:

- **Mesleki Deneyim:** 2006-2008, STS Emniyet ve Bilişim Sistemleri A.Ş.
2008-devam, Kırklareli Üniversitesi

TEZDEN TÜRETİLEN YAYINLAR, SUNUMLAR VE PATENTLER:

- **Aydin, F.** & Aslan, Z., 2017. Diagnosis of Neuro-degenerative Diseases Using Machine Learning Methods and Wavelet Transform. *Journal of the Faculty of Engineering and Architecture of Gazi University*, 32(3), pp.749-766.
- **Aydin, F.** & Aslan, Z., 2017. Classification of Neurodegenerative Diseases using Machine Learning Methods. *International Journal of Intelligent Systems and Applications in Engineering*, 1(5), pp. 1-9. Available at: <http://dx.doi.org/10.18201/ijisae.2017526689>.

DİĞER YAYINLAR, SUNUMLAR VE PATENTLER:

- Demir C., **Aydın, F.**, 2017. Yapay Öğrenme ile Türkiye'nin Kurulu Gücünün 2023 Yılına Kadar Tahmini. *UMAS'17 International Engineering Research Symposium*. Düzce, Turkey, pp. 52-60.
- Ucar, E., Kocamaz, A.F., **Aydın, F.**, 2015. Kalp rahatsızlıklarına yönelik teletıp uzman sistemi. TR Patent No. 2012 04926, Ankara: TPE.
- Cihan, F., **Aydın, F.**, Kocamaz, A.F., 2013. A New Method for Fast Computation of Factorials of Numbers. *Balkan Journal of Mathematics*, 1, pp. 16-27.
- Zontul, M., **Aydın, F.**, Dogan, G., Sener, S., Kaynar, O., 2013. Wind speed forecasting using REPTree and Bagging methods in Kırklareli-Turkey. *Journal of Theoretical and Applied Information Technology*, 56(1), pp. 17-29.

- **Aydin, F.**, Dogan, G., 2013. Development of a new integer hash function with variable length using prime number set. *Balkan Journal of Electrical & Computer Engineering*, 1, pp. 10-14.
- **Kocamaz, A.F.**, **Aydin, F.**, Ucar, E., 2011. Developing of a learning-based system to assist treatment process of arrhythmia patients. *Scientific Research and Essays*, 6(32), pp. 6654-6665.

