**T.C.**
**BAHÇEŞEHİR ÜNİVERSİTESİ**

# YENİDEN SIRALAMALI YÜKSEK BOYUTLU MODEL GÖSTERİLİM İLE VERİ MODELLEMESİ

**Yüksek Lisans Tezi**

**Çağrı AKSU**

**Istanbul, 2011**

**T.C.**
**BAHÇEŞEHİR ÜNİVERSİTESİ**

**Fen Bilimleri Enstitüsü**

**Bilgi Teknolojileri Programı**

# YENİDEN SIRALAMALI YÜKSEK BOYUTLU MODEL GÖSTERİLİM İLE VERİ MODELLEMESİ

**Yüksek Lisans Tezi**

**Çağrı AKSU**

**Danışman: Yrd. Doç. Dr. M. Alper TUNGA**

**Istanbul, 2011**

**T.C.**
**BAHÇEŞEHİR ÜNİVERSİTERSİ**
**Fen Bilimleri Enstitüsü**
**Bilgi Teknolojileri**

Tezin Başlığı : Yeniden Sıralamalı Yüksek Boyutlu Model Gösterilim ile Veri Modellemesi
Öğrencinin Adı Soyadı : Çağrı AKSU
Tez Savunma Tarihi :

Bu yüksek lisans tezi Fen Bilimleri Enstitüsü tarafından onaylanmıştır.

Doç. Dr. F. Tunç BOZBURA
Enstitü Müdür V.

Bu tez tarafımızca okunmuş, nitelik ve içerik açısından bir Yüksek Lisans tezi olarak yeterli görülmüş ve kabul edilmiştir.

Tez Sınav Jürisi Üyeleri :

Yrd. Doç. Dr. M. Alper TUNGA (Tez Danışmanı) :

Yrd. Doç. Dr. Ergün ŞİMŞEK :

Yrd. Doç. Dr. Tevfik AYTEKİN :

# TEŞEKKÜR

Bu tez çalışmasının her aşamasında bana yardımlarını bıkmadan sunan değerli hocam Yrd. Doç. Dr. M. Alper Tunga' ya ve bana olan inançları ve esirgemedikleri destekleri için aileme teşekkürlerimi sunarım.

Mayıs 2011 Çağrı AKSU

# ÖZET

YENİDEN SIRALAMALI YÜKSEK BOYUTLU MODEL GÖSTERİLİM İLE VERİ MODELLEMESİ

Aksu, Çağrı

Bilgi Teknolojileri Programı

M. Alper Tunga

Mayıs 2011, 103 sayfa.

Bu tez çalışmasında, çok değişkenli fonksiyonların yaklaştırımı ve sınıflandırma problemleri ile ilgilenilmektedir. Bu amaçla bilimsel yazında geçmekte olan Yüksek Boyutlu Model Gösterilim Yöntemi (YBMG) ve Yeniden Sıralamalı Yüksek Boyutlu Model Gösterilim Yöntemi (YSYBMG) incelenmektedir. Bu tezin amacı, YSYBMG yöntemi ile gerçek veri kümeleri üzerinde sınıflandırma çözümleri üretmektir. Bu amaçla bu tez içerisinde YSYBMG yöntemi farklı yaklaşımlar ile yeniden yapılandırılmıştır. Elde edilen farklı modellerin sınıflandırma başarıları ölçülmüştür. Bu tez içerisinde yapılan analiz çalışmaları IHDMR yazılımı ile yapılmıştır. IHDMR yazılımı bu analiz çalışmaları için geliştirilmiştir. Elde edilen sonuçlar bilinen sınıflandırma algoritmalarının sonuçları ile karşılaştırılmıştır. Analiz çalışmaları sonucunda, 16 farklı model oluşturulmuştur. Bu modeller, farklı 7 veri kümesi üzerinde denenmiştir. Oluşturulan yeni YSYBMG modellerinin sınıflandırma problemlerinde başarılı sonuçlar verdiği gözlenmiştir.

**Anahtar kelimeler :** YBMG, YSYBMG, Veri modelleme, Sınıflandırma, Veri madenciliği

# ABSTRACT

DATA MODELLING WITH INDEXING HIGH DIMENTIONAL MODEL REPRESENTATION METHOD

Aksu, Çağrı

Bilgi Teknolojileri Programı

M. Alper Tunga

May 2011, 103 pages.

In this thesis, we are dealing with multivariate interpolation and classification problems. For this purpose, the fundamental properties of High Dimensional Model Representation Method (HDMR) and Indexing High Dimensional Model Representation (IHDMR) are analyzed. The aim of this thesis is to produce solutions for classification on real data sets with IHDMR method. For this purpose, in this thesis IHDMR was restructured through a number of different approaches The classification performance of different IHDMR models are measured. The analysis in this thesis was done with IHDMR software. IHDMR software was developed for this analysis work. The results obtained are compared with the results of well-known classification algorithms. As the result of this analysis, 16 different models were built. These models were tested on 7 different data sets. It is observed that our new IHDMR models work successfully in the classification problems.

**Key words :** HDMR, IHDMR, Data modeling, Clasification, Data mining

# İÇİNDEKLER

# TABLOLAR

# ŞEKİLLER

# 1. GİRİŞ

## 1.1 PROBLEM

Günümüzde, insanoğlunun serüveni içerisinde aldığı yol itibarı ile geldiği noktadan daha ileri gidebilmesini mümkün kılacak yegane olgu, insanoğlunun giderek daha da karmaşıklaşan problemleri çözebilme yeteneğini geliştirmesidir. İnsanoğlunun giderek karmaşıklaşan ve bu karmaşıklık itibarı ile çözümü de oldukça fazla değişkene bağlı olan yeni problemleri çözme yeteneği, kuşkusuz yeni edineceği donanımlar ile mümkündür. İnsanoğlu geliştirdiği yöntemler ile hesaplanamayacak zorluktaki sonuçlara ulaşmaya çalışmakta, bazı varsayımlar altında bu sonuçlara yaklaşımlarda bulunacak yeni araçlar geliştirmektedir.

Günümüzün çözülmesi zor problemleri arasında önemli bir yer alan sınıflandırma problemleri içinde benzer bir durum söz konusudur. Sınıflandırma en yaygın veri madenciliği görevidir ve insani bir zorunluluk olarak görünmektedir. İnsanlar dünyayı anlamak ve iletişim kurmak için sürekli olarak sınıflandırma problemleri ile karşılaşmaktadırlar .

Bir sınıflandırıcı, sınıf etiketleri bilinen örneklerden oluşan bir veri kümesi verildiğinde, sınıf etiketi bilinmeyen örneklerin dahil oldukları sınıfları belirleyecek kısa ve anlamlı açıklamalar üretir. Bu açıklamaların üretilmesi birçok değişkene bağlı olarak zorlaşmaktatır. Herhangi bir örneğin dahil olduğu sınıfın belirlenmesi için yapılan hesaplamalar  da bu değişkenlere bağlı olarak zorlaşmaktadır.

Sınıflandırma problemleri, bilinen sınıfların içerdiği örneklerin genel özelliklerinden yola çıkılarak ele alınan herhangi bir nesnenin veya olayın hangi sınıf altında gösterileceğini inceleyen problemlerdir. Örneğin, kan örneği alınan deneğin şeker hastası olup olmadığının anlaşılması denekten alınan kanın özelliklerinin irdelenmesi ile ortaya çıkan bir sonuçtur. Eğer kan değerleri şeker hastalığının sebep olduğu kan

değerleri ile eşleşiyor ise denek şeker hastaları grubunun altında gösterilebilecek bir örnek olarak nitelendirilir.

Bu şeker hastalığının belirlenmesine yönelik sınıflandırma problemi kan değerlerinin belli aralıklara düşmelerinin gözlenmesi ile modellenebilir ve bu model yolu ile çözülebilir, deneklerin hangi sınıfa dahil oldukları belirlenebilir. Fakat her sınıflandırma problemi için bu şekilde oluşturulacak modeller ile doğru sonuçlara ulaşılamıyabilir. Bu durumda regresyon ve bayesyen yöntemler gibi istatistiki yöntemler ile modelleme yapılabilir ya da yapay sinir ağları ile bir model oratya çıkabilir. Bu farklı model kurma yöntemleri her farklı veri kümesi için farklı bir çözüm ortaya koyana kadar devam edebilir.

Sınıfandırma probleminin üç temel ayırıcı niteliği bulunur :

- Öğrenme gözetimlidir.
- Bağımlı değişken kesiklidir.
- Ele alınan örnekleri önceden tanımlı sınıflara atayan modeller oluşturmaktadır.

Kullanılması olası yöntemlerin büyük bir bölümü sınıflandırma problemleri için analitik bir yapı ortaya çıkaramaz. Bir çok yöntem ile analitik yapı oluşturmak bir çok durumda imkansızdır.

## 1.2 AMAÇ

Bu tez çalışması içerisinde, sınıflandırma problemleri ile uğraşılmaktadır. Sınıflandırma problemlerinin YBMG temelinde modellenmesi ve oluşturulacak modellerin gerçek veri kümeleri üzerinde denenmesi amaçlanmaktadır. Gerçek veri setleri üzerinde uygulanan modelleme yöntemleri ile sınıflandırma problemlerinin çözümlerinin analitik bir yapıya kavuşturulması ve sınıf tahminlerini bu analitik yapı üzerinden yapılmasının sınıflandırma problemlerinin çözüm yollarına yeni bir bakış açısı

getireceği ve elde edilen bu analitik yapı ile başarılı sınıf tahminlerinin yapılacağı düşünülmektedir.

Elde edileceği düşünülen sonuçların, bilinen sınıflandırma algoritmaları ile karşılaştırılarak ortaya çıkan farklar üzerinden oluşturulan modellerin başarılarının farklı veri kümeleri için ölçülmesi amaçlanmaktadır.

Bu tez çalışması içerisinde, Yeniden Sıralamalı YBMG metodu içerisinde kullanılan eğitim düğümleri ve endeks uzayı düğümlerinin eşleştirilmesine yönelik yöntemlerin, test düğümü için bulunacak olan yeni endeks düğümünün tespitine yönelik yöntemlerin farklı veri kümeleri için yöntemin daha başarılı bir performans sağlayacağı düşünülmektedir.

Test düğümüne karşılık gelecek olan endeks düğümünün belirlenmesi için yeni bir yaklaşım olan Çok Değişkenli Regresyona Dayalı Yöntem ile endeks düğümü belirleyecek yapı analitik bir yapı olarak seçilebileceği ve sınıf tahminleri için belirlenen endeks düğümü yerine eğitim düğümünün doğrudan kullanılabileceği düşünülmektedir.

Ayrıca bu çalışmaların yapılabilmesini mümkün kılacak bir yazılımın oluşturulması da bu tezin amaçlarından birisidir. Oluşturulacak yazılım sayesinde YBMG tabanlı modelleme yöntemleri oluşturulabilmesinin ve denenmesinin mümkün kılınması istenmektedir. Bu tez içerisinde denenen yöntemlerin diğer araştırmacılar ile paylaşılması ve kullanılmasının mümkün kılınması bu yazılım sayesinde sağlanacaktır. Bu yazılım çalışmasının taşınabilir olması, kurulum gerektirmeden her işletim sisteminde çalıştırılabilmesi yani platformdan bağımsız olması ve geliştirilebilir olması amaçlanmıştır.

Oluşturulacak yazılımın bu yöntemi denemek isteyen kullanıcılara kolaylık sağlayacağı ve bu modelleme yöntemi üzerinde geliştirme çalışmaları yapacak kullanıcılara bir temel olacağı düşünülmektedir.

## 1.3 YÖNTEM

Bu tez çalışmasında üzerinde durulmuş olan ve M. Alper Tunga (2003) tarafından geliştirilmiş olan Yeniden Sıralamalı Yüksek Boyutlu Model Gösterilim yöntemidir (EYBMG).

YSYBMG metodu, fonksiyon değeri bilinmeyen test düğümlerinin fonksiyon değerlerinin bulunması için veri modellemesi yapmaktadır. Yöntem sayesinde, eğitim veri kümesi ile oluşturulan model ile birlikte test veri kümesi içerisindeki test düğümleri için sınıf tahmini yapılması amaçlamaktadır. Amaç dorultusunda oluşturulan modeller ile test düğümünün dahil olduğu sınıf bir fonksiyon ile tahmin edilmektedir.

Yöntem eğitim düğümlerini, oluşturulan endeks uzay üzerindeki düğümlerle eşleştirmekte sonrasında test düğümleri için eğitim düğümlerinden yararlanarak uygun bir endeks düğümü tespit etmekte ve tespit edilen düğümü kullanarak YBMG ile oluşturulmuş analitik yapı ile test düğümünün sınıfını belirlemeye çalışmaktadır.

Bu tez çalışması içerisinde YSYBMG yönteminin sahip olduğu esneklikten yararlanılarak, yöntem üzerinde farklı metrikler ve yöntemler deneyerek yöntemin farklı gerçek veri kümeleri üzerindeki etkinliği gözlenecektir.

Yazılımın oluşturulması süreci taşınabilirlik, her işletim sisteminde kurulum gerekmeden çalışabilirlik gibi amaçlar doğrultusunda şekillendirilecektir. Bu anlamda bu tez çalışması içerisinde Java programlama dili kullanılacaktır. Java programlama dili açık kaynaklıdır ve çeşitli java grupları sayesinde matematik, istatistik, veri madenciliği, veri modelleme gibi konularda diğer dillere göre çok daha üstün bir destek sağlamaktadır. Yazılımın içerisinde sağlanan bu destekten faydalanılmıştır.

Yazılım eclipse yazılım aracı üzerinde java swing kütüphanesi görsel tasarım aracı ile yapılandırılmıştır.

## 1.4 VARSAYIMLAR

Bu tez çalışması içerisinde denenen modelleme yöntemleri, en az üç öznitelik içeren veri kümeleri için geliştirilmiştir. Veri kümeleri sayısal ve kategorik öznitelikler dışında tarih ya da benzer öznitelikler içeremezler. Daha az sayıda öznitelik ve farklı tanımlı öznitelikler içeren veri kümeleri için bu modelleme yöntemlerinin uygulanması anlamlı olmaz.

Bu tez çalışması içerisinde oluşturulan IHDMR yazılımı da en az üç öznitelik içeren ve içerisinde eksik veri bulunmayan veri kümeleri ile çalışmaktadır. Bu veri kümeleri sadece sayısal ve kategorik öznitelikler içermelidirler. Yöntemlerin üzerinde deneneceği veri kümelerinin sınıf belirteçleri sınıflandırma problemleri şartlarına uymalı yani, kategorik olmalıdırlar. Bu şartlara uymayan veri kümeleri yazılıma yüklenemez ve dolayısı ile üzerinde herhangi bir işlem yapılamaz.

Üzerinde analiz çalışmasının yapılacağı veri kümeleri arff formatında olmadır. Bu veri kümelerinin iki farklı senaryo ile yazılıma yüklenmesi mümkündür. Birinci senaryoda tek bir veri kümesi yüklenir ve eğitim veri kümesi rastgele olarak seçilir. İkinci senaryoda eğitim veri kümesi ve test veri kümesi ayrı ayrı seçilir. Bu senaryo içerisinde eğitim veri kümesi içerisinden oluşturulacak olan endeks uzayının içereceği düğüm sayısına göre düğüm eksiltmesi yapılacağı göz önününde bulundurulmalıdır.

# 2. İLGİLİ BİLİMSEL YAZIN

20.yüzyıl içerisinde Sobol (1993) isimli bir Rus istatistikçi Kolmogorov'un (1963) bir çalışmasına dayanarak Yüksek Boyutlu Model Gösterilim (YBMG) yöntemini geliştirmiştir. Sobol, bu yöntemi ilk kez duyarlılık indislerinin hesaplanmasında kullandığından, istatistikte "Sobol Yöntemi" olarak da bilinir. Sobol'un açılımı, Herschel Rabitz tarafından çeşitli problemlere uygulanarak yeniden ele alınarak çarpımsal yapıdaki bir ağırlık ve herbir bağımsız değişken için verilen [a,b] tanım aralığı altında genelleştirilmiştir (Rabitz ve Aliş, 1999). 2003 yılında ise Metin Demiralp (2003), YBMG'nin özelliklerini bir adım daha geliştirerek, yönteme diklik koşulu tanımını kazandırmıştır .

Birçok bilim adamıda bu konudaki çalışmalarını sürdürmektedirler, bu bölüm içerisinde bu çalışmalardan kısaca bahsedilecektir.

Yüksek Boyutlu Model Gösteriliminin kısmi türevli diferansiyel denklemlerin çözümünde kullanılmaktadır. A. Kurşunlu ile M. Demiralp'in "Additive And Factorized HDMR Applications to the Multivariate Diffusion Equation Under Vanishing Derivative Boundary Conditions"(2003) isimli makalesinde difüzyon denkleminin çözümünde Toplamsal ve Çarpımsal YBMG'den faydalanılmaktadır. Ayrıca, T. Civelek ile M. Demiralp'in (2003) "An HDMR Application to the Scrödinger's Equation for Free Particles Under an External Field with Dipole Polarization and Vanishing Flux Boundary Conditions" isimli makalesinde Schrödinger denkleminin çözümünde ÇYBMG ve Evrimsel YBMG kullanılmıştır .

"Efficient Input-Output Model Representation" isimli makalede Herschel Rabitz , Ömer F. Aliş, Jeffrey Shorter ve Kyurhee Shim, 0-D Stratosferik Kimyasal Kinetik modelle çalışılmışlardır (1999). Denemeler 45 derece kuzey enleminde 20 km yükseklikte uçak içerisinde gerçekleştirilmiştir. Uyarlanan model 39 kimyasal çeşit ve 108 farklı

reaksiyon içermektedir. Model 46 girdi ve 39 çıktı ile çalışmaktadır. Ölçümler günde yaklasık 2000 sefer ve bir yıl boyunca devam etmiştir. Ölçümler kimyasal maddelerin miktarları üzerinden yapılmış ve elde edilen veriler Kesme YBMG verileri ile karşılaştırılmıştır. Değişik zaman aralılarında yapılan ölçümler, Kesme YBMG ile elde edilenler rakamların yaklaşık %2 hata ile yaklaşımda bulunduğunu göstermiştir.

"Correlation Method for Variance Reduction of Monte Carlo Integration in RS-HDMR" isimli makalede Li G, Rabitz H, Wang SW ve Georgopoulos PS. fotokimyasal kutu modeli (photochemical box model) incelemiştir (2003). 28 çeşit kimyasal ve 63 reaksiyon ile şekillenen problemde ozon üretimi (P), ozon bozunması oranı (D) ve P-D eğilimi hesaplanmıştır. Sayıları 300, 500, 1000, 3000 ve 5000 olan 5 örneklem seçilerek doğrudan Monte-Carlo kullanılarak birinci, ikinci ve üçüncü mertebeden yaklaştırımlar hesaplanmıştır. Aynı örneklem uzaylarında ortogonal polinomlar kullanıldığında ise ikinci mertebeden Seçkisiz Örneklemeli YBMG yaklaştırımı oldukça iyi sonuçlar verdiği gözlenmiştir.

H.Rabitz ve F.A.Alış'ın "Efficient Implementation of HDMR" (2003) isimli makalesinde Seçkisiz Örneklemeli YBMG kullanılarak, duyarlılık indisleri yaklaşık olarak hesaplanmıştır. "A recursive Algorithm for Finding HDMR Terms for Sensitivity Analysis" isimli makalelerinde H. Kaya, M. Kaplan ve H. Saygın'ın (2003) bu konuda çalışmaları bulunmaktadır. "Multicut HDMR with an Application to an Ionospheric Model" isimli makalede G. Li, J. Schoendorf, T. Ho, H. Rabitz (2004) iyonosferic modeli, hem 3 alt tanım kümesi kullanılarak Kesme YBMG ile, hem de aynı 3 referans noktası için Çoklu Kesme YBMG ile çözülmüştür . M. Demiralp ile M. A. Tunga (2003) "Data Partitioning via GHDMR and Multivariate Interpolative Applications" isimli makalede, fonksiyonun ağ yapı üzerindeki düğümlerde aldığı değerler kullanılarak fonksiyonun analitik yapısı belirlenmeye çalışılmaktadırlar.

"Practical Approaches to Construct RS-HDMR Component Functions" isimli makalede Rabitz, H. Li, G. ve Wang, S. (2002) makalelerinde RS-HDMR yöntemi ile atmosferik modelleme konusunda kolay ulaşılır ve yüksek kesinlik düzeyine sahip

yaklaştırmalarda bulunmuşlardır.

“An HDMR Application to the Optimal Control of Harmonic Oscillator” (2003) makalede Demiralp, M. ve Kaman, T. ve “HDMR Approximation of an Evolution Operator with a First Order Partial Differential Operator Argument” (2003) isimli makalelerde Demiralp, M. ve Yaman, İ. dalga denklemlerini incelemişlerdir. Bu çalışmalarda birinci veya ikinci mertebeden EYBMG terimlerinin belirlenmesi tatmin edici bir sonuç elde edilmesi için yeterli olmaktadır.

Duyarlılık indislerinin hesaplanması ile ilgili olarak Sobol’un “Theorems and Examples on High Dimensional Model Representation” (2003) isimli makalesinde, HDMR açılımının her fonksiyon tipinde verimli sonuçlar vermediği, daha iyi sonuçlar elde edebilmek için daha yüksek mertebeden yaklaştırımlar yapılması gerektiği belirtilmiştir. M. Demiralp ile B. Tunga’nın “Hybrid YBMG Approximants and Their Utilization in Applications ” (2003) isimli makalesinde Melez YBMG kullanılarak yapısı içerisinde hem toplamasal hemde çarpımsal bileşenler bulunan fonksiyonları incelemişlerdir. Makalede farklı m, N ve α değerleri için Melez YBMG açılımları hesaplanmış ve Melez YBMG'nin hem toplamsal hem de çarpımsal özellikler gösteren fonksiyonlar için daha iyi sonuçlar verdiği gözlenmiştir.

Esen Akkemik ve Metin Demiralp’ in “Algebraic eigenvalue problem modeling via high dimensional model representation” (2003) isimli makalesinde matris özdeğer problemlerinin çözümü için yeni bir metod geliştirilmiştir. Bu metod öncelikle Yüksek Boyutlu Model Gösterilim Yöntemlerine dayanmaktadır. Makalede cebrik özdeğer problemlerinin çözümü için Yüksek Boyutlu Model Gösterilim kullanımında A matrisi bağımsız HDMR değişkeni, A matrisinin özvektörü olan u vektörü ise bağımlı HDMR değişkeni olarak alınmıştır. Özdeğer problemlerinin çözümü için kesme HDMR’ın yorumlanmasından sonra λ özdeğer parametresinin yaklaşık değerinin bulunabilmesi için HDMR yaklaştırımı sıfıra eşitlenmiştir. N. A. Baykara ile M. Demiralp’in “Hyperspherical or Hyperellipsoidal Coordinates in the Evaluation of High Dimensional Model Representation Approximants” (2003) isimli makalesinde

hiperküre, hiperelipsoid gibi farklı bölgelerde incelenen fonksiyonların YBMG açılımı gerekli dönüşümler yapılarak hesaplanmıştır.

M. A. Tunga ve M. Demiralp' in, "A Hybrid Programming for Projective Displaying of High Dimensional Model Representation Approximants" (2003), "A Factorized High Dimensional Model Representation on the Partitioned Random Discrete Data" (2004), "A Factorized High Dimensional Model Representation on the Nodes of a Finite Hyperprismatic Regular Grid" (2005), "Hybrid High Dimensional Model Representation (HHDMR) on the Partitioned Data" (2006), "A New Approach for Data Partitioning Through High Dimensional Model Representation" (2008), "Bound analysis in univariately truncated Generalized High Dimensional Model Representation for random-data partitioning: Interval GHDMR" (2009), "A Reverse Technique for Lumping High Dimensional Model Representation Method" (2009), isimli makalelerinde, farklı nitelikteki problemler üzerinde farklı bakış açıları ile yeniden şekillendirilmiş farklı YBMG yöntemleri denenmiş ve başarılı sonuçlar alınmıştır.

"A tool to improve the execution time of air quality models" (2008) isimli makalede M.C. Gomez, V. Tchijov, F. Leon ve A. Aguilar, hava değerlendirme modellerinin uygulama sürelerini kısaltmak için kullanmışlardır.

T. Ziehn, A.S. Tomlin, "A global sensitivity study of sulfur chemistry in a premixed methane flame model using HDMR" (2008) isimli makalede T. Ziehn ve A.S. Tomlin karmaşık kimyasal mekanizmalarda duyarlılık analizi için kullanmışlardır. YBMG ile gerçekleştirilen yaklaşımın bu tip kimyasal olaylar için çok kuvvetli bir araç olduğu sonucuna varmışlardır. T. Ziehn, A.S. Tomlin, yaptıkları analizleri "GUI-HDMR a software tool for global sensitivity analysis of complex models" (2008) isimli makalede verilen yazılım ile gerçekleştirmişlerdir.

"Probabilistic analysis using high dimensional model representation and fast Fourier transform." (2008) isimli makalede B.N. Rao ve R. Chowdhury, rastgele yüklere matuz kalan mekanik sistemlerin güvenilirliğini, materyal ve yapısal özelliklerini

incelemişlerdir. YBMG bu analiz içerisinde limit durumundaki fonksiyonlara yaklaşım için kullanılmıştır. Burada yaklaşımlar birinci dereceden YBMG bileşenleri ile gerçeklenmiştir. B.N. Rao ve R. Chowdhury, “Hybrid high dimensional model representation for reliability analysis” (2008) isimli makalelerinde ise, Melez YBMG ile aynı probleme farklı bir yaklaşım ile yaklaşmışlar ve başarılı sonuçlar almışlardır.

“High dimensional model representation for piece-wise continuous function approximation” (2009)isimli makalede Rajib Chowdhury, B. N. Rao ve A. Meher Prasad, parçalı sürekli fonksiyonlar için YBMG yönteminin başarısını incelemişlerdir. Sürekli fonksiyonlar için her zaman parçalı sürekli bir fonksiyon ile yaklaştırım yapılamadığı ve bu gibi durumlar YBMG yönteminin sağlıklı bir yaklaştırım yaptığı sonucuna varmışlardır.

M. A. Tunga "A Matrix Based Indexing HDMR Method for Multivariate Data Modelling" (2011) isimli makalesinde, YBMG yöntemini çok değişkenli vei bölümleme süreçlerine uygulamaktadır. Yöntem ile veri kümeleri daha az sayıda değişken içeren veri kümeleri haline getirilerek elde edilen analitik yapı ile başarılı yaklaştırımlar yapılabilmektedir.

# 3. YÜKSEK BOYUTLU MODEL GÖSTERİLİM (YBMG)

Çok değişkene bağlı problemlerin çözümünün bilgisayar üzerinde hesaplanması genellikle zaman ve bellek kullanımındaki yüksek değerler nedeni ile günümüz teknolojisinin sunduğu işlem yapabilme kapasitesi ile mümkün olmamaktadır. Bu tip problemlerin hesaplanması sırasında ortaya çıkan bellek gereksiniminin üstel olarak genişlediği ve sonunda içinden çıkılmaz bir duruma dönüştüğü gözlenmektedir. Bu sebeple çok değişkenli problemlerin çözümüne yönelik çalışmaların matematiksel anlamda etkili temsili modeller ile yapılması zorunludur. Bu modeller böl ve yönet mantığı ile oluşturulabilir. YBMG de böl ve yönet mantığını kullanmaktadır. YBMG yapısı sabit terim ile başlamakta sonra bir değişkenli terimler ile devam etmektedir. Bir değişkenli terimler sonrası iki değişkenli terimler ve onlardan sonra da üç değişkenli terimler gelmektedir. Bu yapı değişken sayıları artan terimler ile genişlemektedir.

Verilen bir $f(x_1,\ldots,x_N)$ çok değişkenli fonksiyonu için YBMG açılımı aşağıdaki eşitlikle verilebilir (Sobol, 1993).

$$f(x_1,\ldots,x_N)= f_0+\sum_{i_1=1}^{N} f_{i_1}(x_{i_1})+\sum_{\substack{i_1,i_2=1\\ i_1<i_2}}^{N} f_{i_1 i_2}(x_{i_2},x_{i_1}) + \sum_{\substack{i_1,i_2,i_3=1\\ i_1<i_2<i_3}}^{N} f_{i_1 i_2 i_3}(x_{i_1},x_{i_2},x_{i_3})+\ldots+f_{1\ldots N}(x_1,\ldots,x_N) \quad \textbf{(3.1)}$$

Denklem (3.1) 'in sol tarafında bulunan ve $f(x_1,\ldots,x_N)$ şeklinde gösterilen ve $x_1,\ldots,x_N$ ile simgelenen N bağımsız değişkene bağlı çok değişkenli bir fonksiyon, denklemin sağ tarafında sonlu sayıda bileşenin toplamı şeklinde yazılmıştır. İlk bileşen $f_0$ ile simgelenen bir sabittir. Daha sonraki ilk N bileşen $f_1(x_1), f_2(x_2),\ldots f_N(x_N)$ şeklinde simgelenen ve her biri tek bir bağımsız

değişkene bağlı fonksiyonlardır. Bunların ardından da her biri iki bağımsız değişkene bağlı $N(N-1)/2$ adet fonksiyon gelmektedir. Diğer bileşenler de bu şekilde, gittikçe artan sayıda bağımsız değişken içeren fonksiyonlardan oluşmaktadır. Sağ taraftaki terimlerin toplam adedi $2^N$ dir.

Denklem (3.1) ile verilen açılımın sağ yanında bulunan terimler aşağıda verilen koşulu sağlayacak şekilde bulunmaktadır (Demiralp, 2003).

$$\int_{a_1}^{b_1} dx_1 \cdots \int_{a_N}^{b_N} dx_N W(x_1,\ldots x_N) f_i(x_i) = 0, \qquad 1 \le i \le N \tag{3.2}$$

Yukarıda verilen koşula ait bağıntıda bulunan $W(x_1,\ldots x_N)$ ağırlık fonksiyonu tek değişkenli fonksiyonların çarpımından oluşturulmuş bir fonksiyondur(Demiralp, 2003).

$$W(x_1,\ldots x_N) = \prod_{j=1}^{N} W_j(x_j), \quad x_j \in \left[a_j, b_j\right], \qquad 1 \le j \le N \tag{3.3}$$

Denklem (3.2) 'de verilen koşul aşağıda verilen diklik koşuluna karşılık gelmektedir (Demiralp, 2003).

$$(f_{i_1 i_2 \ldots i_k},\ f_{i_1 i_2 \ldots i_l}) = 0, \quad \{i_1,\ldots,i_k\} \neq \{i_1,\ldots,i_l\}, \qquad 1 \le k,l \le N \tag{3.4}$$

Bu koşul (3.1)' de verilen denklemin sağ tarafında kalan bileşenlerin birbirine dik olduğunu göstermektedir. Burada diklik koşulu bir iç çarpım üzerinden tanımlanmakta ve Denklem (3.1) 'de verilen gerek $f(x_1,\ldots,x_N)$ fonksiyonunun gerekse sağ yandaki bileşenlerin karesi integre edilebilen fonksiyonlar olduğu varsayılmaktadır.

Kare integrallar ve iç çarpım, bağımsız değişkenler için baştan belirlenen belli bir aralık üzerinde tanımlanmakta ve her bir değişken için o değişkene bağlı olarak verilen bir ağırlık fonksiyonu kullanılmaktadır. Bu bağlamda, iççarpımın tanımı genel olarak aşağıdaki gibi yazılabilir.

$$(u,v)\equiv\int_{a_1}^{b_1} dx_1 W(x_1)\cdots\int_{a_N}^{b_N} dx_N W(x_N)\,u(x_1,\ldots,x_N)\,v(x_1,\ldots,x_N) \qquad \textbf{(3.5)}$$

Ayrıca yukarıda sözü edilen bileşenlerin kolayca saptanabilmesi için, ağırlık fonksiyonun her bileşeninin ilgili aralık üzerindeki integralının 1 olduğu yani,

$$\int_{a_j}^{b_j} dx_j W_j(x_j) \;=\; 1, \qquad 1\le j\le N \qquad \textbf{(3.6)}$$

eşitliğinin geçerli olduğu da varsayılmaktadır. (3.1) eşitliğinin sağ tarafındaki bileşenlerin herbirinin belirlenmesi için bir takım izdüşüm operatörlerinden yararlanılabilir. Bu bağlamda, YBMG' nin sabit terimi için tanımlanan $I_0$ operatörü karesi integrallenebilen herhangi bir $F(x_1,\ldots,x_N)$ fonksiyonu için aşağıdaki gibidir.

$$I_0F(x_1,\ldots,x_N)\equiv\int_{a_1}^{b_1} dx_1 W_1(x_1)\cdots\int_{a_N}^{b_N} dx_N W_N(x_N)F(x_1,\ldots,x_N) \qquad \textbf{(3.7)}$$

$I_0$ operatörü (2.1) eşitliğinin her iki tarafına da uygulandığında diklik koşulu gereği sabit terim dışındaki terimler yok olacaktır. Bu uygulama sonrasında,

$$f_0=I_0f(x_1,\ldots,x_N) \qquad \textbf{(3.8)}$$

eşitliği elde edilmiş olur. YBMG' nin tek bağımsız değişkenli bileşenlerini belirleyebilmek için uygulanacak operatör, $1 \le k_1 \le N$ koşulu altında aşağıdaki gibidir.

$$I_{k_1} F(x_1,\ldots,x_N) \equiv \int_{a_1}^{b_1} dx_1 W(x_1) \cdots \int_{a_{k_1-1}}^{b_{k_1-1}} dx_{k_1-1} W_{k_1-1}(x_{k_1-1}) \times \int_{a_{k_1+1}}^{b_{k_1+1}} dx_{k_1+1} W_{k_1+1}(x_{k_1+1}) \cdots \int_{a_N}^{b_N} dx_N W_N(x_N) F(x_1,\ldots,x_N) \quad \textbf{(3.9)}$$

$I_{k_1}$ operatörü (2.1) eşitliğinin her iki tarafına da uygulandığında diklik koşulu gereği

$$f_{k_1}(x_{k_1}) = I_{k_1} f(x_1,\ldots,x_N) - f_{0,} \qquad 1 \le k_1 \le N \quad \textbf{(3.10)}$$

eşitliği elde edilmiş olur. YBMG' nin iki bağımsız değişkenli bileşenlerini belirleyebilmek için uygulanacak operatör, $1 \le k_1 < k_2 \le N$ koşulu altında aşağıdaki gibidir.

$$I_{k_1 k_2} F(x_1,\ldots,x_N) \equiv \int_{a_1}^{b_1} dx_1 W(x_1) \cdots \int_{a_{k_1-1}}^{b_{k_1-1}} dx_{k_1-1} W_{k_1-1}(x_{k_1-1}) \times \int_{a_{k_1+1}}^{b_{k_1+1}} dx_{k_1+1} W_{k_1+1}(x_{k_1+1}) \cdots \int_{a_{k_2-1}}^{b_{k_2-1}} dx_{k_2-1} W_{k_2-1}(x_{k_2-1}) \times \int_{a_{k_2+1}}^{b_{k_2+1}} dx_{k_2+1} W_{k_2+1}(x_{k_2+1}) \cdots \int_{a_N}^{b_N} dx_N W_N(x_N) F(x_1,\ldots,x_N) \quad \textbf{(3.11)}$$

$I_{k_1 k_2}$ operatörü (3.1) eşitliğinin her iki tarafına da uygulanırsa diklik koşulu gereği

$$f_{k_1k_2}(x_{k_1},x_{k_2})=I_{k_1k_2}f(x_1,\ldots,x_N)-f_{k_2}(x_{k_2})-f_{k_1}(x_{k_1})-f_0 \tag{3.12}$$

eşitliği elde edilmiş olur. YBMG' nin üç bağımsız değişkenli bileşenlerini belirleyebilmek için uygulanacak operatör, $1\le k_1<k_2<k_3\le N$ koşulu altında aşağıdaki gibidir.

$$\begin{aligned} I_{k_1k_2k_3}F(x_1,x_2,\ldots,x_N)\equiv & \int_{a_1}^{b_1}dx_1\,W(x_1)\cdots\int_{a_{k_1-1}}^{b_{k_1-1}}dx_{k_1-1}\,W_{k_1-1}(x_{k_1-1}) \\ & \times\int_{a_{k_1+1}}^{b_{k_1+1}}dx_{k_1+1}\,W_{k_1+1}(x_{k_1+1})\cdots\int_{a_{k_2-1}}^{b_{k_2-1}}dx_{k_2-1}\,W_{k_2-1}(x_{k_2-1}) \\ & \times\int_{a_{k_2+1}}^{b_{k_2+1}}dx_{k_2+1}\,W_{k_2+1}(x_{k_2+1})\cdots\int_{a_{k_3-1}}^{b_{k_3-1}}dx_{k_3-1}\,W_{k_3-1}(x_{k_3-1}) \\ & \times\int_{a_{k_3+1}}^{b_{k_3+1}}dx_{k_3+1}\,W_{k_3+1}(x_{k_3+1})\cdots\int_{a_N}^{b_N}dx_N\,W_N(x_N)F(x_1,x_2,\ldots,x_N) \end{aligned} \tag{3.13}$$

$I_{k_1k_2k_3}$ operatörü 1 eşitliğinin her iki tarafına da uygulanırsa diklik koşulu gereği

$$\begin{aligned} f_{k_1k_2k_3}(x_{k_1},x_{k_2},x_{k_3})=I_{k_1k_2k_3}f(x_1,\ldots,x_N)-f_{k_3k_2}(x_{k_3},x_{k_2})-f_{k_3k_1}(x_{k_3},x_{k_1}) \\ -f_{k_2k_1}(x_{k_2},x_{k_1})-f_{k_3}(x_{k_3})-f_{k_2}(x_{k_2})-f_{k_1}(x_{k_1})-f_0 \end{aligned} \tag{3.14}$$

eşitliği elde edilmiş olur. Diğer YBMG bileşenleri de aynı yolla bulunabilmektedir. Bu tez çalışması içerisinde, sabit terim, birli terimler, ikili terimler ve üçlü terimler kullanılarak elde edilen analitik yapı kullanılacaktır. YBMG nin sabit teriminden sonrası yaklaştırım içerisine dahil edilmez ise bu yaklaştırıma "sabit yaklaştırım" denilmektedir. Eğer sabit terim sonrasında bir değişkenli terimler alınıyorsa "bir

değişkenli yaklaştırım", iki değişkenli terimler alınıyor ise "iki değişkenli yaklaştırım" denilmekte ve bu isimlendirme bu şekilde devam etmektedir. Bu kesme işlemleri aşağıdaki gibi gösterilmektedir.

$$
\begin{aligned}
s_0(x_1,\ldots,x_N) &= f_0 \\
s_1(x_1,\ldots,x_N) &= s_0(x_1,\ldots,x_N)+\sum_{i_1=1}^{N} f_{i_1}(x_{i_1}) \\
s_2(x_1,\ldots,x_N) &= s_1(x_1,\ldots,x_N)+\sum_{\substack{i_1,i_2=1 \\ i_1<i_2}}^{N} f_{i_1 i_2}(x_{i_1},x_{i_2}) \\
s_3(x_1,\ldots,x_N) &= s_2(x_1,\ldots,x_N)+\sum_{\substack{i_1,i_2,i_3=1 \\ i_1<i_2<i_3}}^{N} f_{i_1 i_2 i_3}(x_{i_1},x_{i_2},x_{i_3})
\end{aligned}
\qquad \textbf{(3.15)}
$$

# 4. YENİDEN SIRALAMALI YÜKSEK BOYUTLU MODEL GÖSTERİLİM (YSYBMG)

Yeniden Sıralamalı Yüksek Boyutlu Model Gösterilim yöntemi, böl yönet mantığı ile çok boyutlu bir veri kümesini az boyutlu veri kümelerine bölümleyerek interpolasyon ve sınıflandırma problemlerini modellemektedir. Yöntem, eğitim veri kümesi ile oluşturulan model ile birlikte test veri kümesi içerisindeki test düğümleri için sınıf tahmini yapılmasını amaçlamaktadır. Oluşturulan bu model matematiksel anlamda, test düğümünün dahil olduğu sınıfı bir fonksiyon ile tahmin etmektedir. Bu fonksiyona dahil olan bağımsız değişkenler test düğümünün öz nitelikleridir.

Eğitim veri kümesinin genel yapısı, ilgili sınıf değeri ile birlikte aşağıdaki gibi tanımlanmaktadır.

$$T \equiv \left\{ t_s | t_s = (v_1^{(s)}, \ldots, v_N^{(s)}, \varphi_s), \quad \varphi_s \equiv f(v_1^{(s)}, \ldots, v_N^{(s)}), \quad 1 \leq s \leq m \right\} \qquad \textbf{(4.1)}$$

Burada N öznitelik sayısı, m eğitim düğümü sayısı ve $\varphi$ eğitim düğümünün dahil olduğu sınıfı belirtir simgedir. Her test düğümünün tanımı aşağıdaki gibi verilebilir.

$$\Theta \equiv \left\{ \theta_s | \theta_s = (\mu_1^{(s)}, \ldots, \mu_N^{(s)}), \quad 1 \leq s \leq q \right\} \qquad \textbf{(4 .2)}$$

Burada q test düğümlerinin sayısını simgelemektedir.

Eğitim düğümlerini ve eğitim düğümlerinin sınıf değerlerini kullanarak bir model geliştirmek için, öncelikle Sobol tarafından önerilen ve (3.1) ile verilen Yüksek Boyutlu Model Gösterilim açılımı kullanılmaktadır.

Yüksek Boyutlu Model Gösterilim yöntemi ile dik geometriye sahip olan bir veri kümesi için model geliştirilebilmektedir (Sobol, 1993) Eğer parametrelerin alabilecekleri farklı değerlerin sayısı $n_1, n_{2,}\ldots, n_n$ ise, o zaman bu dik yapı parametre kümelerinin kartezyen çarpımı üzerinden inşa edilebilir. Bu belirtilen veri kümesi aşağıdaki gibi tanımlanabilir.

$$D \equiv D_1 \times D_2 \times \cdots \times D_N, \quad D_j = \left\{ \xi_j^{(1)}, \ldots, \xi_j^{(n_j)} \right\}, \quad 1 \le j \le N \qquad \textbf{(4 .3)}$$

Ancak, gerçek hayat problemlerinde belirtilen kümenin tüm düğümlerindeki sınıf değerlerini bilmek mümkün değildir. Bu tanım ile birlikte elimizde $n_1 \times n_2 \times \cdots \times n_n$ tane düğümümüzün olduğunu biliyoruz. Bununla birlikte bu hesaplama gerçek uygulamalarda mümkün olmamaktadır. Bu yüzden Yeniden Sıralamalı YBMG dik bir endeks uzayı yaratır ve gerçek değer uzayı ile uygun şekilde bire bir eşleştirir. Bu yolla test düğümünün sınıfını hesaplayacak YBMG yapısının, gerçek model yerine kurduğumuz bu yeni model ile uygulanması mümkün kılınmış olmaktadır. Burada her bir $\xi$ ana veri kümesi içerisindeki parametrelerin endeks uzayı içerisindeki karşılığı olmaktadır.

Endeks uzayını oluşturmak için ilk adım, eğitim düğümlerinin sayısını, yani m sayısını asal çarpanlarına ayırmak olacaktır. Örnek vermek gerekirse, üzerinde çalışılacak eğitim veri kümesi 4 öznitelik ve 100 örnek içeriyor ise değerlendirmeye alınacak olan asal çarpanlar 2, 2, 5 ve 5 dir. Bu durumda endeks uzayını oluşturan her bir endeks kümesi $x_1 \in \{1,2\}$, $x_2 \in \{1,2\}$, $x_3 \in \{1,2,3,4,5\}$ $ve\, x_4 \in \{1,2,3,4,5\}$ şeklinde yazılmaktadır. Bu seçim eşsiz değildir. İstenilir ise, üzerinde çalışılacak veri kümesinin özelliklerine göre şekillendirilebilmektedir. Yöntemin uygulanmasındaki bir sonraki adım, endeks kümelerinin kartezyen çarpılmasıdır. Bu işlem sonrası elde edilecek sonuç kümesi dik bir endeks uzayıdır. Elde edilen bu uzay üzerindeki düğümler ile eğitim veri kümesini oluşturan düğümlerin eşleştirilmesi durumu üzerinde çalışılan veri kümesinin özelliklerine göre şekillendirilebilmektedir.

Bu aşamada eğitim verisini modellemek için, YBMG bileşenleri uygulanacak yöntem için yeniden tanımlanmalıdır. Bu bağlamda sadece veri kümesi ve o veri kümesinin düğümlerindeki sayı değerleri modellemede kullanılacağından YBMG bünyesindeki ağırlık fonksiyonu Dirac delta fonksiyonun doğrusal kombinasyonları olarak seçilmelidir.

$$W_j(x_j) \equiv \sum_{k_j=1}^{n_j} \alpha_{k_j}^{(j)} \delta(x_j - \xi_j^{(k_j)}), \quad \sum_{k_j=1}^{n_j} \alpha_{k_j}^{(j)} = 1, \quad x_j \in [a_j, b_j], \quad 1 \le j \le N \qquad \textbf{(4.4)}$$

Burada $\alpha$ her eğitim düğümü için farklı bir önem derecesi belirten katsayıdır. Bu ağırlık fonksiyonuna göre YBMG sabiti tanımı aşağıdaki gibidir.

$$f_0 = \sum_{k_1=1}^{n_1} \sum_{k_2=1}^{n_2} \cdots \sum_{k_N=1}^{n_N} \left( \prod_{j=1}^{N} \alpha_{k_j}^{(j)} \right) f(\xi_1^{(k_1)}, \ldots, \xi_N^{(k_N)}) \qquad \textbf{(4.5)}$$

Bir değişkenli YBMG bileşenleri $1 \le i_1 \le N$ koşulu altında aşağıdaki gibidir.

$$f_{i_1}(\xi_{i_1}^{(k_{i_1})}) = \sum_{k_1=1}^{n_1} \cdots \sum_{k_{i_1-1}=1}^{n_{i_1-1}} \sum_{k_{i_1+1}=1}^{n_{i_1+1}} \cdots \sum_{k_N=1}^{n_N} \left( \prod_{\substack{j=1 \\ j \ne i_1}}^{N} \alpha_{k_j}^{(j)} \right) f(\xi_1^{(k_1)}, \ldots, \xi_{i_1}^{(k_{i_1})}, \ldots, \xi_N^{(k_N)}) - f_0 \qquad \textbf{(4.6)}$$

$1 \le i_1 < i_2 \le N$ koşulu altında iki değişkenli YBMG bileşenleri aşağıdaki gibidir.

$$f_{i_1 i_2}(\xi_{i_1}^{(k_{i_1})}, \xi_{i_2}^{(k_{i_2})}) = \sum_{k_1=1}^{n_1} \cdots \sum_{k_{i_1-1}=1}^{n_{i_1-1}} \sum_{k_{i_1+1}=1}^{n_{i_1+1}} \cdots \sum_{k_{i_2-1}=1}^{n_{i_2-1}} \sum_{k_{i_2+1}=1}^{n_{i_2+1}} \cdots \sum_{k_N=1}^{n_N} \left( \prod_{\substack{j=1 \\ j \ne i_1 \wedge j \ne i_2}}^{N} \alpha_{k_j}^{(j)} \right) \qquad \textbf{(4.7)}$$

$$\times f(\xi_1^{(k_1)}, \ldots, \xi_{i_1}^{(k_{i_1})}, \ldots, \xi_{i_2}^{(k_{i_2})}, \ldots, \xi_N^{(k_N)}) - f_{i_1}(\xi_{i_1}^{(k_{i_1})}) - f_{i_1}(\xi_{i_2}^{(k_{i_2})}) - f_0$$

YBMG bileşenleri içerisinden üç değişkenli olanları $1 \le i_1 < i_2 < i_3 \le N$ koşulu altında aşağıdaki gibidir.

$$
\begin{aligned}
f_{i_1 i_2 i_3}(\xi_{i_1}^{(k_{i_1})}, \xi_{i_2}^{(k_{i_2})}, \xi_{i_3}^{(k_{i_3})}) \;=\; & \sum_{k_1=1}^{n_1} \cdots \sum_{k_{i_1-1}=1}^{n_{i_1-1}} \sum_{k_{i_1+1}=1}^{n_{i_1+1}} \cdots \sum_{k_{i_2-1}=1}^{n_{i_2-1}} \sum_{k_{i_2+1}=1}^{n_{i_2+1}} \cdots \sum_{k_{i_3-1}=1}^{n_{i_3-1}} \sum_{k_{i_3+1}=1}^{n_{i_3+1}} \\
& \times \cdots \sum_{k_N=1}^{n_N} \left( \prod_{\substack{j=1 \\ j \neq i_1 \wedge j \neq i_2 \wedge j \neq i_3}}^{N} \alpha_{k_j}^{(j)} \right) f(\xi_1^{(k_1)}, \ldots, \xi_{i_1}^{(k_{i_1})}, \ldots, \xi_{i_2}^{(k_{i_2})}, \ldots, \xi_{i_3}^{(k_{i_3})}, \ldots, \xi_N^{(k_N)}) \\
& - f_{i_1 i_2}(\xi_{i_1}^{(k_{i_1})}, \xi_{i_2}^{(k_{i_2})}) - f_{i_1 i_3}(\xi_{i_1}^{(k_{i_1})}, \xi_{i_3}^{(k_{i_3})}) - f_{i_2 i_3}(\xi_{i_2}^{(k_{i_2})}, \xi_{i_3}^{(k_{i_3})}) - f_{i_1}(\xi_{i_1}^{(k_{i_1})}) \\
& - f_{i_2}(\xi_{i_2}^{(k_{i_2})}) - f_{i_3}(\xi_{i_3}^{(k_{i_3})}) - f_0
\end{aligned}
\qquad \textbf{(4.8)}
$$

Bu bileşen yapılarının kullanılması sonrasında bir sabit değer, N tane bir parametreli veri kümesi, $N(N-1)/2$ tane iki parametreli veri kümesi ve $N(N-1)(N-2)/6$ tane üç parametreli veri kümesi elde edilmektedir. Problemi modellemek için gerekli olan analitik yapıya ulaşmak için aşağıdaki Lagrange katsayıları kullanılır (Tunga, 2007),

$$
L_{k_{i_1}}(x_{i_1}) \;\equiv\; \prod_{\substack{j=1 \\ j \neq k_{i_1}}}^{n_{i_1}} \frac{(x_{i_1} - \xi_{i_1}^{(j)})}{(\xi_{i_1}^{(k_{i_1})} - \xi_{i_1}^{(j)})} \qquad \xi_{i_1}^{(k_{i_1})} \in D_{i_1}, \quad 1 \le k_{i_1} \le n_{i_1}, \quad 1 \le i_1 \le N \qquad \textbf{(4.9)}
$$

ve aşağıdaki çok terimliler tanımlanır[4.10].

$$
p_{i_1}(x_{i_1}) \equiv \sum_{k_{i_1}=1}^{n_{i_1}} L_{k_{i_1}}(x_{i_1}) f_{i_1}(\xi_{i_1}^{(k_{i_1})}) \qquad \textbf{(4.10.1)}
$$

$$p_{i_1 i_2}(x_{i_1}, x_{i_2}) \equiv \sum_{k_{i_1}=1}^{n_{i_1}} \sum_{k_{i_2}=1}^{n_{i_2}} L_{k_{i_1}}(x_{i_1}) L_{k_{i_2}}(x_{i_2}) f_{i_1 i_2}(\xi_{i_1}^{(k_{i_1})}, \xi_{i_2}^{(k_{i_2})}) \qquad \textbf{(4.10.2)}$$

$$p_{i_1 i_2 i_3}(x_{i_1}, x_{i_2}, x_{i_3}) \equiv \sum_{k_{i_1}=1}^{n_{i_1}} \sum_{k_{i_2}=1}^{n_{i_2}} \sum_{k_{i_3}=1}^{n_{i_3}} L_{k_{i_1}}(x_{i_1}) L_{k_{i_2}}(x_{i_2}) L_{k_{i_3}}(x_{i_3}) f_{i_1 i_2 i_3}(\xi_{i_1}^{(k_{i_1})}, \xi_{i_2}^{(k_{i_2})}, \xi_{i_3}^{(k_{i_3})},)$$

$$\xi_{i_1}^{(k_{i_1})} \in D_{i_1}, \quad \xi_{i_1}^{(k_{i_2})} \in D_{i_2}, \quad \xi_{i_1}^{(k_{i_3})} \in D_{i_3}, \quad 1 \le i_1 < i_2 < i_3 \le N \qquad \textbf{(4.10.3)}$$

Yukarıda yapılan tanımlamalar sonrası Yeniden Sıralamalı YBMG üç terimli olarak aşağıdaki gibi yazılır.

$$f(x_1, \ldots, x_N) \approx f_0 + \sum_{i_1=1}^{N} p_{i_1}(x_{i_1}) + \sum_{\substack{i_1, i_2=1 \\ i_1 < i_2}}^{N} p_{i_1 i_2}(x_{i_1}, x_{i_2}) + \sum_{\substack{i_1, i_2, i_2=1 \\ i_1 < i_2 < i_3}}^{N} p_{i_1 i_2 i_3}(x_{i_1}, x_{i_2}, x_{i_3}) \qquad \textbf{(4.11)}$$

Bu yaklaştırma endeks uzayı içerisindeki düğümler için geçerlidir. Fakat test düğümleri endeks uzayı içerisinde değildir. Bu yüzden her test düğümü için endeks uzayı içerisinde karşılık gelen düğüm yaratılmalıdır. Bu süreç içerisinde ilk yapılan, test düğümüne benzer bir eğitim düğümü bulunmasıdır. Bu benzer düğüm çeşitli benzerlik ya da uzaklık ölçüleri ile belirlenebilmektedir. Bilimsel yazında belirtilen işlem öklit uzaklığı kullanılarak yapılmıştır (Tunga, 2007). Test düğümü ve eğitim düğümü arasındaki öklit uzaklıkları içerisinden en kısa olanını sağlayan eğitim düğümü referans olarak alınmaktadır. Bu tez çalışmasının amaçlarından biri de belirtilen bu sürecin değişik benzerlik ve uzaklık yöntemleri aracılığı ile daha etkin hale getirilmesidir. Bu amaçla geliştirilen adımlar bir sonraki bölümde verilmektedir.

# 5. VERİ MODELLEME

Bu bölümde, oluşturulacak olan veri modelleme yöntemini eğitmek için kullanılacak olan eğitim veri kümesinin sıralanması, oluşturulmuş bulunan endeks uzay üzerindeki düğümler ile eşleştirilmesi ve sonrasında modelin test edileceği veri kümeleri için daha önceden oluşturulmuş olan endeks uzay içersinde uygun düğümlerin üretiminin yapılmasına yönelik yöntemler önerilmiş ve irdelenmiştir. Bu bölümde gerçekleştirilecek çalışmalar için kullanılacak olan eğitim veri kümesi için, aykırı değer analizi ya da öznitelik önem kontrolü gibi veri kümesinin sayısal niteliklerini değiştirecek yöntemlerin bu aşamadan önce uygulanması gerekmektedir. Bu bölümde kullanılacak veri kümesinin sayısal niteliklerinin uygulanacak veri modelleme yönteminin başarısı için değişmemesi gerekmektedir.

Uygulanacak olan aşamalar sırası ile;

- eğitim için kullanılacak olan veri kümesinin sayısının belirlenmesi ve bu sayıya göre endeks uzayın her bir parametresinin boyutunun belirlenmesi,

- eğitim veri kümesinin sıralanması ve endeks uzaydaki düğümlerle eşleştirilmesi,

- test veri kümesi için endeks uzayında uygun düğümlerin belirlenmesi, olarak verilebilir

## 5.1 BOYUT BELİRLEME ALGORİTMALARI

Bu bölümün temel amacı, YSYBMG yönteminin uygulanması için gereken endeks uzayının boyutlarını hesaplamak ve bu boyutları belirlemek için harcanan gereksiz zaman kaybının önüne geçmektir. Bu çalışma içerisinde, bu amaca yönelik iki algoritma yazılmıştır.

### 5.1.1 proCent Algoritması

Bu algoritma, eğitim ve test veri kümesinin, girilen tek bir veri kümesi içerisinden rastgele seçilecekleri durumlarda kullanılmak üzere hazırlanmıştır. proCent algoritması, oluşturulacak endeks uzayının boyutlarını belirlerken dikkate aldığı ana unsur oluşturulacak uzay içerisindeki düğümlerin sayısının gerçek veri kümesinin içerdiği örnek sayısının $\frac{2}{3}$ 'ü kadar ya da bu durum endeks uzayı koşullarına uymuyor ise bu sayıya en yakın olacak şekilde belirlenmesini sağlamaktır.

$$k_i = \left(k_{i_1}, k_{i_2}, \cdots, k_{i_s}\right), \quad kt_i = k_{i_1} \times k_{i_2} \times \cdots \times k_{i_s}, \quad ub = \frac{2N}{3}$$
$$hu = kt_i, \quad kt_i : \min(|kt_i - ub|) \tag{5.1}$$

Burada $k_i$ endeks uzayı için denenecek boyutlar, N gerçek uzayın içerdiği örnek sayısı, s örneklerin içerdiği özniteliklerinin sayısı, *ub* ulaşılması beklenen endeks uzayı düğüm sayısı, *hu* hesaplanan endeks uzayının içereceği düğüm sayısıdır. Algoritma temel olarak gerçek veri kümesinin içerdiği örnek sayısının asal çarpanlarını belirlemekte, bu çarpanları endeks uzayı boyutları olarak atayarak ortaya çıkan uzayın içerdiği düğüm sayısının amaçlanan rakama ulaşıp ulaşmadığını kontrol etmekte, eğer hedefe ulaşılmamışsa boyutları küçülterek ya da büyüterek uygun boyutları yakalamaya çalışmaktadır.

Bu hesaplamalar sırasında göz önünde tutulan bir diğer koşul ise, elde edilecek boyutların arasındaki farkların en küçük olacak şekilde seçilmesidir. Örneğin 315 düğüm içeren bir veri kümesi bölünürken $\{1, 1, 2, 3, 5, 7\}$ gibi oluşturacağı uzayın içerdiği düğüm sayısı olan 210 tam olarak $\frac{2N}{3}$ oranına uysa da, bu boyutlandırma yerine $\{2, 2, 2, 3, 3, 3\}$ gibi 216 düğüm içeren bir boyutlandırma tercih edilmektedir. Bu tercihin ana sebebi uygulanacak modelleme yöntemleri için

endeks uzayı bileşenlerinin aldıkları farklı değer sayılarının büyük bir öneme sahip olmalarıdır. $\{1, 1, 2, 3, 5, 7\}$ boyutlarında bir endeks uzayın, 1. ve 2. bileşenlerinin hiçbir farklı değer içermiyor olduklarından dolayı çözüme katkısı $\{2, 2, 2, 3, 3, 3\}$ boyutlarındaki bir endeks uzaya göre daha düşük olacaktır. Bu durumun sebebi, eşleştirme algoritmalarının farklı değer içermeyen endeks uzay bileşenleri ile sağlıklı bir eşleştirme yapamayacak olmalarıdır. Bu anlamda aşağıdaki koşula uyumluluk aranmaktadır.

$$hu = kt_i, \quad kt_i: \min\left(\sum_{j=1}^{s}\sum_{l=1}^{s}\frac{(kt_{i_j}-kt_{i_l})^2}{N}\right) \tag{5.2}$$

Burada s örneklerin içerdikleri özniteliklerin sayısıdır. Bu eşitlik endeks uzayı bileşenlerinin içerdikleri farklı değer sayılarının diğer bileşenlerin içerdiği farklı değer sayıları ile aralarındaki farkların minumum olması gerektini göstermektedir.

### 5.1.2 proCentOne Algoritması

proCentOne algoritması eğitim ve test veri kümelerinin, sisteme ayrı ayrı girilmesi durumunda kullanılmak üzere hazırlanmıştır. proCentOne algoritması oluşturulacak endeks uzayının boyutlarını belirlerken dikkate aldığı ana unsur, oluşturulacak uzay içerisindeki düğümlerin sayısının gerçek veri kümesinin içerdiği örnek sayısına eşit olması ya da bu durum endeks uzayı koşullarına uymuyor ise bu sayıya en yakın olacak şekilde belirlenmesini sağlamaktadır.

$$\begin{gathered} k_i = \left\{k_{i_1}, k_{i_2}, \cdots, k_{i_s}\right\}, \quad kt_i = k_{i_1} \times k_{i_2} \times \cdots \times k_{i_s}, \quad ub = N \\ hu = kt_i, \quad kt_i: \min(|kt_i - ub|) \end{gathered} \tag{5.3}$$

Burada değişkenler ve yaklaşım yöntemi proCent algoritması içerisinde anlatıldığı gibidir. ProCentOne algoritmasında da proCent algoritması gibi hesaplamalar sırasında

göz önünde tutulan bir diğer koşul, elde edilecek boyutların arasındaki farkların en küçük olacak şekilde seçilmesidir.

Örneğin 390 düğüm içeren bir veri kümesi bölünürken $\{1, 1, 2, 3, 5, 13\}$ gibi oluşturacağı uzayın içerdiği düğüm sayısı tam olarak 390 olsada, bu boyutlandırma yerine $\{2, 2, 2, 3, 4, 4\}$ gibi 384 düğüm içeren bir boyutlandırma tercih edilmektedir. Dolayısı ile burada da (3.2) eşitliliğine uyumluluk aranmaktadır.

## 5.2 SIRALAMA ALGORİTMALARI

Bu bölümün temel amacı, farklı eğitim veri kümesi dizilimleri ve bunlara uygun olarak oluşturulmuş bulunan endeks uzay içersindeki noktalar ile eşleşmelerin oluşturulan model üzerindeki etkilerinin gözlemlenebilmesi için farklı dizilim algoritmaları oluşturmaktır.

Oluşturulan algoritmalar ile ortaya konulacak olan farklı eşleşmeler, oluşturulacak olan modeli doğrudan etkilemektedir. Bu etkileme olayı test veri kümesi için oluşturulacak yeni endeks uzay düğümlerinin belirlenmesinde kilit rol oynamaktadır. Çünkü yeni endeks düğümleri belirleme çalışmaları, daha evvel oluşturulmuş olan endeks uzayı üzerindeki düğümlerle eşleşmiş olan eğitim veri kümesi içerisindeki düğümler üzerinden yürütülmektedir.

Dolayısı ile referans olarak kabul edilen eğitim düğümünün eşleştiği endeks düğümünün farklı seçilimleri, test edilecek veri kümesi için oluşturulacak endeks düğümlerinin farklılaşması durumunu ortaya çıkarmaktadır.

Kullanılacak olan algoritmalar için uygulama amaçlı oluşturulan eşitlikler aşağıdaki gibi verilmiştir.

$$A_{mxn}=\begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mn} \end{bmatrix},\quad I_{mxn}=\begin{bmatrix} I_{11} & I_{12} & \cdots & I_{1n} \\ I_{21} & I_{22} & \cdots & I_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ I_{m1} & I_{m2} & .... & I_{mn} \end{bmatrix},$$

$$R=\begin{bmatrix} R_1 \\ R_2 \\ \vdots \\ R_m \end{bmatrix},\quad FS=\begin{bmatrix} FS_1=0 \\ FS_2=1 \\ \vdots \\ FS_k=k \end{bmatrix} \qquad \textbf{(5.4)}$$

Burada $A_{mxn}$ eğitim veri kümesini temsil eden matris, $I_{mxn}$ endeks uzayını temsil eden matris $R_{1xm}$ A eğitim veri kümesi içersindeki düğümlerin sınıflarını belirten matris $FS_{1xm}$ A eğitim veri kümesi içersindeki sınıfları belirten matris olarak tanımlanmıştır.

Kullanılacak olan dizilim algoritmalarının düğümler üzerindeki etkilerinin gösterilmesi için Tablo 5.1 deki örnek veri kümesi kullanılacaktır.

**Tablo 5.1 : Örnek ham veri kümesi**

| | **1 2 3 4** | **sınıf** | | **1 2 3 4** | **sınıf** |
|---|---|---|---|---|---|
| **1** | 1, 6, 1, 3 | 11 | **6** | 1, 3, 4, 2 | 10 |
| **2** | 3, 5, 1, 3 | 12 | **7** | 0, 6, 2, 2 | 10 |
| **3** | 1, 4, 4, 1 | 10 | 8 | 4, 6, 3, 2 | 16 |
| **4** | 1, 6, 5, 0 | 8 | **9** | 4, 6, 0, 4 | 16 |
| **5** | 3, 6, 1, 1 | 11 | 10 | 3, 6, 4, 3 | 16 |

### 5.2.1 Class Sıralama Algoritması

Bu yöntem eğitim veri kümesinin R sınıf matrisi elemanlarına göre dizilmesi ve bu dizilimin FS içerisindeki sıraya göre yapılmasını amaçlamaktadır. Oluşturulan gruplar içinde herhangi bir sıralama olmaz. Endeks uzayı için bir değişiklik söz konusu

değildir. Eşleştirme sıraya göre yapılır. Belirtilen bu sonuçlar Tablo 5.2 ile verilmektedir.

Bu sürecin örnek veri kümesi kullanılarak yapılacak olan tarifi şu şekildedir:

Örnek veri kümesi içerisindeki düğümler sınıflarına göre dizilmelidirler. Bu dizilime göre örnek veri kümesi içerisinde dizilimden önce 4. sırada bulunan "1, 6, 5, 0 sınıfı : 8" düğümü dizilim sonrası en küçük sınıf sayısına sahip olduğu için 1. sıraya yerleşmiştir. Dizilim öncesi 2. sırada bulunan "3, 5, 1, 3 sınıfı : 12 " düğümü dizilim sorası 7. sıraya yerleştirilmiştir.

**Tablo 5.2 : Class ile dizilim sonrası eşleştirme sonucu**

| | eğitim düğümleri | | indeks uzayı |
|---|---|---|---|
| ↑ | 1, 6, 5, 0 | sınıfı : 8 | 1, 1, 1, 1 |
| ↑ | 1, 4, 4, 1 | sınıfı : 10 | 1, 1, 1, 2 |
| ↑ | 1, 3, 4, 2 | sınıfı:10 | 1, 1, 1, 3 |
| ↑ | 0, 6, 2, 2 | sınıfı:10 | 1, 1, 1, 4 |
| ↓ | 1, 6, 1, 3 | sınıfı:11 | 1, 1, 1, 5 |
| ↓ | 3, 6, 1, 1 | sınıfı : 11 | 1, 1, 2, 1 |
| ↓ | 3, 5, 1, 3 | sınıfı : 12 | 1, 1, 2, 2 |
| | 4, 6, 3, 2 | sınıfı : 16 | 1, 1, 2, 3 |
| | 4, 6, 0, 4 | sınıfı : 16 | 1, 1, 2, 4 |
| | 3, 6, 4, 3 | sınıfı : 16 | 1, 1, 2, 5 |

Bu dizilim algoritması ile hedeflenen eğitim düğümlerinin endeks uzay içerisinde sınıflarına göre ayrıklaştırılmasıdır. Elde edilen bu ayrıklık durumu endeks uzayında, farklı sınıftan düğümlerin bölünmez alanlar oluşturması ile sonuçlanmaktadır.

### 5.2.2 Order Sıralama Algoritması

Bu yöntem, class yöntemi ile elde edilmiş dizilim sorasında eşleştirilme yapılmadan önce sınıflarına göre dizilmiş olan düğümleri son özniteliklerinden başlayarak her

öznitelik için küçükten büyüğe doğru olacak şekilde yeni bir dizilim algoritmasına tabi tutulması ile gerçekleştirilir. Endeks uzayı için bir değişiklik söz konusu değildir. Eşleştirme sıraya göre yapılır. Eşleştirme sonuçları Tablo 5.2 ile verilmektedir.

Bu sürecin örnek veri kümesi kullanılarak yapılacak olan tarifi şu şekildedir:

Örnek veri kümesi içerisindeki sınıflarına göre sıralanmış düğümler, sınıfları içerisinde tekrar dizilmektedirler. Bu dizilime göre, örnek veri kümesi içerisinde dizilimden önce sınıfları "10" olan "0, 6, 2, 2 sınıfı : 10" ve "1, 4, 4, 1 sınıfı:10" düğümleri özniteliklerinin büyüklük ve küçüklüklerine göre değerlendirilerek yer değiştirilmişlerdir. Aynı sınıfa sahip "1, 3, 4, 2 sınıfı:10" düğümü veri kümesi içerisindeki yerini korumuştur.

**Tablo 5.3 : Order ile dizilim sonrası eşleştirme sonucu**

| | eğitim düğümleri | | indeks uzayı |
|---|---|---|---|
| | 1, 6, 5, 0 | sınıfı : 8 | 1, 1, 1, 1 |
| ↑ | 0, 6, 2, 2 | sınıfı : 10 | 1, 1, 1, 2 |
| | 1, 3, 4, 2 | sınıfı:10 | 1, 1, 1, 3 |
| ↓ | 1, 4, 4, 1 | sınıfı:10 | 1, 1, 1, 4 |
| | 1, 6, 1, 3 | sınıfı:11 | 1, 1, 1, 5 |
| | 3, 6, 1, 1 | sınıfı : 11 | 1, 1, 2, 1 |
| | 3, 5, 1, 3 | sınıfı : 12 | 1, 1, 2, 2 |
| ↑ | 3, 6, 4, 3 | sınıfı : 16 | 1, 1, 2, 3 |
| | 4, 6, 0, 4 | sınıfı : 16 | 1, 1, 2, 4 |
| ↓ | 4, 6, 3, 2 | sınıfı : 16 | 1, 1, 2, 5 |

Bu dizilim algoritması ile hedeflenen, sınıflarına göre dizilmiş eğitim düğümlerinin, endeks uzay içerisinde sınıflarına göre ayrık olarak oluşturacakları alanlar içerisinde eşleştikleri noktaların, rastgele seçilimini bir kurala bağlamak ve alanlar içerisinde düzenli bir yayılımı mümkün kılmaktır. Elde edilen bu kendi içerisinde düzenli ayrıklık durumu, elde edilen bölünmez alanlar içerisinde oluşturulacak yeni düğümler için referans bir kurala izin vermektedir.

### 5.2.3 use rank Sıralama Algoritması

Bu yöntem, order yöntemi ile eğitim düğümleri dizildikten sonra endeks uzay boyutları ile eğitim veri kümesinin özniteliklerinin içerdikleri farklı değer sayılarına göre bir eşleştirme yapılmasını ve bu eşleştirme sonrası endeks uzayı boyutlarının bu eşleştirmeye göre yeniden belirlenmesini amaçlamaktadır. Amaçlanan yeni dizilimi sağlayacak eşleştirme, endeks uzayının en büyük boyutu ile eğitim veri kümesi içerisindeki öznitelikler arasından en çok farklı değer içeren özniteliğin eşleştirilmesi ve sonraki boyutlar ve öznitelikler arasındaki eşleşmelerinde aynı kuralla devam etmesi şeklinde gerçeklenmektedir. Gerekli dizilim sonrası eğitim düğümleri ve endeks uzayı noktaları arasındaki eşleştirme sıraya göre yapılmaktadır.

Bu sürecin örnek veri kümesi kullanılarak yapılacak olan tarifi şu şekildedir:

Örnek veri kümesi içerisinde en fazla farklı değer içeren 3. özniteliktir. Sonraki, 4 ve son ikisi de 1. ve 2. dir. Endeks uzayında ise en fazla farklı değer içeren sütunlar sırasıyla 4, 3, 2, 1 numaralı sütunlardır. Bu durumda indeks uzayında 3. ve 4. sütunlar yer değiştirmelidir. Bu sıralama algoritması sonucunda örnek ham veri kümesi için elde edilen yeni sıralama Tablo 5.4 ' de verilmektedir.

**Tablo 5.4 : Use rank ile dizilim sonrası eşleştirme**

| eğitim düğümleri | | indeks uzayı ← → |
|---|---|---|
| 1, 6, 5, 0 | sınıfı : 8 | 1, 1, 1, 1 |
| 0, 6, 2, 2 | sınıfı : 10 | 1, 1, 2, 1 |
| 1, 3, 4, 2 | sınıfı:10 | 1, 1, 3, 1 |
| 1, 4, 4, 1 | sınıfı:10 | 1, 1, 4, 1 |
| 1, 6, 1, 3 | sınıfı:11 | 1, 1, 5, 1 |
| 3, 6, 1, 1 | sınıfı : 11 | 1, 1, 1, 2 |
| 3, 5, 1, 3 | sınıfı : 12 | 1, 1, 2, 2 |
| 3, 6, 4, 3 | sınıfı : 16 | 1, 1, 3, 2 |
| 4, 6, 0, 4 | sınıfı : 16 | 1, 1, 4, 2 |
| 4, 6, 3, 2 | sınıfı : 16 | 1, 1, 5, 2 |

Bu dizilim yöntemi ile hedeflenen; endeks uzayının, eğitim veri kümesi içerisindeki özniteliklerin içerdikleri değerler anlamıyla genişlikleri ile uygun olacak şekilde oluşturulmasıdır.

Bu işlem sonrası endeks uzayın, eğitim veri kümesinin farklı değerler içeren öznitelikleri bakımından daha iyi bir temsil yeteneğine sahip olacağı düşünülmektedir. Bu daha iyi temsil yeteneği, endeks uzayı içerisinde eğitim düğümleri ile oluşturulan ayrık kümeler içerisindeki düzenli yapının kalitesinin arttırılmasıdır. Bu durum, yeni endeks düğümler oluşturulurken gözönüne alınabilecek yeni bir kural ortaya çıkarmış olmaktadır.

### 5.2.4 use rate Sıralama Algoritması

Bu yöntem, eğitim veri kümesi içerisindeki düğümlerin içerdikleri değerlerin veri kümesi öznitelikleri için infogainattributeeval algoritması ile elde edilen değerler ( $IG_i$ ) ile çarpılıp toplanması ve elde edilen bu toplamlara göre sıralanması mantığına dayanmaktadır. Bu yöntem use rank yönteminden sonra uygulanmaktadır. Endeks uzayı içerisindeki düğümler içinde aynı işlem gerçeklenmektedir.

#### 5.2.4.1 Infogainattributeeval algoritmasının kullanılması

Bu algoritma bir düzensizlik ve öngörülemezlik ölçüsü olarak nitelendirilen entropi' yi temel olarak almaktadır. Yöntem eğitim veri kümesi içerisindeki düzensizliği ölçer ve özniteliklerin bu düzensizlik üzerindeki etkisini sorgular. Eğitim veri kümesi Y için hesaplanan entropi $H(Y)$ ile eğitim veri kümesi içerisindeki sınıfların kümesi X in hesaplanan entropi üzerindeki etkisi $H(Y/X)$ ve $IG$ elde edilen faydadır.

Eğitim veri kümesi Y için hesaplanan entropi aşağıdaki bağıntı kullanılarak bulunur

$$H(Y)=-\sum_{y\in Y} p(y).\log_2(p(y)) \qquad \textbf{(5.5)}$$

Eğitim veri kümesi içerisindeki sınıfların kümesi X in hesaplanan entropi üzerindeki etkisi ise aşağıdaki gibidir.

$$H(Y/X)=-\sum_{x\in X} p(x)\sum_{y\in Y} p(y/x).\log_2(p(y/x)) \quad \textbf{(5.6)}$$

Her öznitelik için elde edilen faydanın, $IG_i$ hesaplanması

$$IG=H(Y)-H(Y/X)=H(Y)-H(X/Y),\ IG = \begin{bmatrix} IG_1 \\ IG_2 \\ \vdots \\ IG_n \end{bmatrix} \quad (5.7)$$

bağıntısı ile gerçekleştirilir.

### 5.2.4.2 Sıralama ölçüsünün hesaplanması

Sıralama ölçüsü eğitim düğümü içersindeki noktaların özniteliklerin sağladıkları fayda ile çarpılıp toplanması ile elde edilen bir sayıdır. Düğümler bu sayıya göre küçükten büyüğe sıralanırlar. Belittilen yeni sıralanma Tablo V.5' de verilmektedir.

$$ÇT = \begin{bmatrix} ÇT_1 \\ ÇT_2 \\ \vdots \\ ÇT_n \end{bmatrix} \quad ÇT_i = \sum_{j=1}^{n} A_{ij} IG_j \quad \textbf{(5.8)}$$

Bu sıralama algoritması ile hedeflenen özniteliklerin entropi üzerindeki etkilerinin kullanılarak ayrık alanlar içerisinde oluşturulmuş düzeni, eğitim veri kümesi içerisindeki düzene uydurabilmektir. Bu sayede kaliteli bir eşleşme yapılabileceği düşünülmektedir.

**Tablo 5.5 : Use rate ile dizilim sonrası eşleştirme sonuçları**

| eğitim düğümleri | | | indeks uzayı | |
|---|---|---|---|---|
| | | | ← | → |
| ↑ | 3, 6, 1, 1 | sınıfı : 11 | | 1, 1, 1, 1 |
| ↑ | 1, 4, 4, 1 | sınıfı:10 | ↑ | 1, 1, 1, 2 |
| ↓ | 0, 6, 2, 2 | sınıfı : 10 | ↓ | 1, 1, 2, 1 |
| ↓ | 1, 3, 4, 2 | sınıfı:10 | ↑ | 1, 1, 2, 2 |
| ↓ | 1, 6, 5, 0 | sınıfı : 8 | ↓ | 1, 1, 3, 1 |
| ↓ | 1, 6, 1, 3 | sınıfı:11 | ↑ | 1, 1, 3, 2 |
| | 3, 5, 1, 3 | sınıfı : 12 | ↓ | 1, 1, 4, 1 |
| ↑ | 4, 6, 0, 4 | sınıfı : 16 | ↑ | 1, 1, 4, 2 |
| ↑ | 4, 6, 3, 2 | sınıfı : 16 | ↓ | 1, 1, 5, 1 |
| ↓ | 3, 6, 4, 3 | sınıfı : 16 | | 1, 1, 5, 2 |

**NOT: Bu örnekte** $IG_i = i$ **eşitliği kullanılmıştır.**

## 5.3 YENİ ENDEKS DÜĞÜMÜ BELİRLEME ALGORİTMALARI

Yeni endeks düğümü bulmak için oluşturulan algoritmalar, daha önce yapılmış olan eğitim veri kümesi, $A_{mxn}$ ve endeks uzayı, $I_{mxn}$ eşleştirmeleri ile ortaya çıkarılan endeks uzayı üzerinde, test veri kümesinin, $T_{kxn}$ içerdiği test düğümleri için uygun bir endeks düğümü yaratmayı amaçlamaktadırlar.

$$T_{kxn} = \begin{bmatrix} T_{11} & T_{12} & \cdots & T_{1n} \\ T_{21} & T_{22} & \cdots & T_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ T_{k1} & T_{k2} & .... & T_{kn} \end{bmatrix} \quad (5.9)$$

Bu algoritmalar ile yapılan tahminin doğruluğu doğal olarak, test düğümünün dahil olduğu sınıfın tahmin edildiği, YSYBMG ile elde edilmiş analitik yapıdan alınacak sonucu doğrudan etkilemektedir. Yapılan tahminin doğruluğu ile düğümün sınıfının belirlenmesi doğru orantılı iki tahmin sürecidir. Bu tahmin süreci aynı sınıf içerisinde bir noktanın referans alınmasının sağlanması ya da belirli bir kurala göre referans alınan herhangi bir noktadan veya noktalardan yola çıkarak doğru endeks düğümünü

yaratacak fonksiyonun bulunması şeklinde benzerlik ve/veya yakınlık ölçüleri kullanılarak gerçeklenebileceği gibi tüm eğitim veri kümesi kullanılarak ortak kullanıma açık bir $f(T_{11}, T_{12,}\ldots, T_{1n})$ fonksiyonu üreterek de gerçeklenebilmektedir.

Üretilmiş olan $f(T_{11}, T_{12,}\ldots, T_{1n})$ fonksiyonu test düğümü için yeni endeks uzayı düğümü belirleme işini bir fonksiyona yüklediğinden dolayı, doğal olarak test düğümünün dahil olduğu sınıfın belirlenmesi için belirlenen tahmin süreci içerisindeki bir aşamayı atlamakta ve süreci kısaltmaktadır.

Yeni endeks düğümü bulma algoritmaları bu düşünceler ışığında iki farklı yaklaşım kullanılarak ortaya konulmuştur. Bu iki farklı yaklaşım, fonksiyonel ve benzerlik yaklaşımlarıdır.

### 5.3.1 Fonksiyonel Yaklaşımlar

Fonksiyonel yaklaşım ile ulaşılması düşünülen temel hedef, eğitim veri kümesinin özelliklerinden ya da eğitim veri kümesinden elde edilecek bilgiler kullanılarak yeni endeks düğümler için bir $f(T_{11}, T_{12,}\ldots, T_{1n})$ kestiricisi elde etmektir. Bu kestirici fonksiyon sayesinde her test düğümü için doğru endeks düğümü oluşturulacak olması nedeniyle bir kere oluşturulduktan sonra istenilen kadar test düğümü için herhangi bir artı tahmin işlemine gerek olmadan gerekli endeks düğümünü belirleyecektir.

Elde edilen bu endeks düğüm kestiricisi fonksiyon YSYBMG yöntemi ile oluşturulacak olan analitik yapı içerisine gömüldüğünde ise, elde edeceğimiz denklem ile birlikte test düğümlerinin mensup oldukları sınıfları, hiçbir işlemden geçirmeden elde edilen denklemde yerine koyma yöntemini uygulayarak belirlenebilecektir.

Bu yöntem ile bir kere oluşturulacak olan analitik yapı sayesinde yüksek işlem maliyeti olan sınıflandırma işlemlerine gerek duyulmadan hesaplamalar yapılabilecek ya da örneğin bir çiçeğin özelliklerini bir fonksiyonda yerine koyarak artı bir işlem yapılmadan çiçeğin türünü belirten sınıf belirteci belirlenebilecektir.

#### 5.3.1.1 Çok değişkenli regresyona dayalı yöntem

Bu yöntem, test için kullanılacak test düğümlerinin her özniteliği için ayrı bir çok değişkenli regresyon denklemi oluşturulması ve bu denklemler ile test düğümleri için endeks düğümleri oluşturmayı amaçlamaktadır.

Regresyon denklemleri, eğitim veri seti kullanılarak herbir öz nitelik için ayrı olarak üretilir. Regresyon denklemleri oluşturulurken, hangi özniteliğin tahmin edicisi olacak ise o öznitelik $A_i$ veri kümesinden çıkarılır, geri kalan öznitelikler bağımsız değişkenler olarak atanır ve çıkarılan özniteliğe karşılık gelen yani aynı sırada bulunan endeks uzayı boyutu, $I_i$ ulaşılmaya çalışılacak olan bağımlı değişken olan $Y$ olarak tanımlanır.

Herbir öznitelik için oluşturulan bu eşitlikler sistemine $K_i$ eşitlikler sistemi denilmektedir. Örneğin birinci öznitelik için oluşturulmuş eşitlikler sistemi $K_1$ olarak adlandırılmaktadır. Oluşturulan bu eşitlik sistemi aşağıda gösterilmiştir.

$$Y = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix} = \begin{bmatrix} I_{11} \\ I_{21} \\ \vdots \\ I_{n1} \end{bmatrix},$$

$$X_{mx(n-1)} = \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1(n-1)} \\ X_{21} & X_{22} & \cdots & X_{2(n-1)} \\ \vdots & \vdots & \ddots & \vdots \\ X_{m1} & X_{m2} & .... & X_{m(n-1)} \end{bmatrix} = \begin{bmatrix} A_{12} & A_{13} & \cdots & A_{1n} \\ A_{22} & A_{23} & \cdots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{m2} & A_{m3} & \cdots & A_{mn} \end{bmatrix} \quad \textbf{(5.10)}$$

Elde edilen $Y$ ve $X_{mx(n-1)}$ kullanılarak regresyon denklemi elde edilir. Regresyon denklemi içerisinde yer alan bağımsız değişkenlerin $X_i$ alacakları değerler $T_{1i}$ olarak belirlenmektedir.

$$X = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_{(n-1)} \end{bmatrix}, \quad M = T^t, \quad M_i = \begin{bmatrix} T_{12} \\ T_{13} \\ \vdots \\ T_{1n} \end{bmatrix}, \quad X = M_i \tag{5.11}$$

Regresyon denklemi $\hat{\beta}$ katsayıları vektörü (5.12) eşitliği ile belirlenmektedir. Elde edilen $\hat{\beta}$ katsayılar vektörü kullanılarak elde edilen regresyon denklemi (5.13) eşitliğinde verilmiştir

$$\hat{\beta} = (X^T X)^{-1} X^T Y \tag{5.12}$$

$$\hat{Y}_1 = \beta_0 + \beta_1 X_1 + \ldots + \beta_{(n-1)} X_{(n-1)} + e_1 \tag{5.13}$$

$K_i$ eşitlikler sistemi her öznitelik için tekrar oluşturulur. Bu eşitlik sisteminden yola çıkılarak her öznitelik için yeni bir çok değişkenli regresyon denklemi oluşturulur. Bu süreç sonrasında öznitelik sayısı kadar regresyon denklemi elde edilmiş olmalıdır. Bu regresyon denklemleri öznitelik sayısının bir eksiği kadar bağımsız değişken içermektedir.

Elde edilen regresyon denklemleri ile birlikte yeni oluşturulacak endeks düğümünün her noktası için bir tahmin edici denklem ve tahmini denklem sonucu, $\hat{Y}_i$ elde edilmiş olacaktır.

$$\begin{aligned} \hat{Y}_1 &= \beta_{00} + \beta_{11} T_{12} + \beta_{21} T_{13} + \ldots + \beta_{1(n-1)} T_{1n} + e_1 \\ \hat{Y}_2 &= \beta_{01} + \beta_{12} T_{11} + \beta_{22} T_{13} + \ldots + \beta_{2(n-1)} T_{1n} + e_2 \\ &\vdots \\ \hat{Y}_n &= \beta_{0n} + \beta_{1n} T_{11} + \beta_{2n} T_{12} + \ldots + \beta_{n(n-1)} T_{1(n-1)} + e_n \end{aligned} \tag{5.14}$$

$\hat{Y}_i$ denklemleri kullanılarak test düğümleri için endeks düğümleri oluşturulmaktadır. Bu hesaplama için tahmin edilecek öznitelik düğümden çıkarılmakta ve geri kalan

öznitelikler regresyon denkleminde kullanılarak tahmin yapılmaktadır. Bu süreç herbir öznitelik için tekrar edildikten sonra yeni indeks düğümü $YI_z$ oluşturulmaktadır.

$$YI_z = \{\hat{Y}_{z1}, \hat{Y}_{z2}, \ldots, \hat{Y}_{zn}\} \quad \textbf{(5.15)}$$

### 5.3.2 Benzerliğe Dayalı Yaklaşımlar

Benzerliğe dayalı yaklaşımlar, değişik benzerlik ya da uzaklık ölçülerini kullanarak test düğümü için yakın ve/veya benzer eğitim düğümleri ve/veya kümeleri bulunması ve bu elde edilen bilgilerden yola çıkılarak yeni bir endeks düğüm oluşturulması amacını taşımaktadır.

Benzerliğe dayalı tahmin süreçleri için iki farklı yönelim olabilmektedir. Bunlardan birincisi, referans alınacak eğitim düğümünün aynı sınıfa mensup olması dolayısı ile bu eğitim düğümünün eşleştiği endeks düğümüne yakın yeni bir endeks düğümü oluşturulmasıdır. Bu yönelim, eğitim veri kümesi ile eşleşen endeks uzayının eğitim düğümlerinin sınıflarına göre ayrık alanlar içermesinden yola çıkmakta ve üretilecek endeks düğümünün ayrık alanlar içerisine düşeceği öngörülmektedir. İkinci yönelim, belirlenmiş bir kurala göre belirlenmiş referans noktası üzerinden yine belli bir kurala uyarak ve/veya bir fonksiyon yardımı ile yeni bir endeks düğümü oluşturulması fikrine dayanmaktadır.

#### 5.3.2.1 Değişen varyans oranları yöntemi

Değişen varyans oranları yöntemi, temel olarak test düğümlerinin, eğitim veri kümesi içerisindeki farklı sınıflara dahil olan eğitim düğümlerinin öznitelikleri için hesaplanan varyans değerlerinin üzerinde yaratacakları etkinin ayrışma bakımından önemli bir ölçüt olduğunu kabul eder ve en az etki yaratılan grubu test düğümü için yeni eğitim veri kümesi olarak kabul eder. Bu etkinin gözlenebilmesi için, $A_{mxn}$ yani eğitim için kullanılacak veri kümesi, eğitim düğümlerinin sınıflarına göre farklı gruplara, $G_i$

bölünmektedir. $R_{1xm}$ Matrisi, $A_{mxn}$ eğitim veri kümesinin içerdiği sınıfları içeren matris olmak üzere $R_{1xm}$ matrisinin rankı farklı grup sayısını vermektedir.

$$R = \begin{bmatrix} R_1 \\ R_2 \\ \vdots \\ R_m \end{bmatrix}, \quad G = \{G_1, G_2, \ldots, G_{rank(R)}\} \tag{5.16}$$

Burada G tüm grupları kapsayan küme olarak belirlenmiştir. G kümesinin alt kümeleri $G_i$ ler aşağıdaki gibi gösterilmişlerdir.

$$G_i = GA_{m_i n} = \begin{bmatrix} A_{k1} & A_{k2} & \cdots & A_{kn} \\ A_{(k+1)1} & A_{(k+1)2} & \cdots & A_{(k+1)n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{m_i 1} & A_{m_i 2} & \cdots & A_{m_i n} \end{bmatrix} \tag{5.17}$$

$G_i$ veri kümelerinin içerdikleri öznitelikler, $G_{ij}$ ler aşağıdaki gibi gösterilmektedirler.

$$G_{ij} = GA_i = \begin{bmatrix} A_{kj} \\ A_{(k+1)j} \\ \vdots \\ A_{m_i j} \end{bmatrix} \tag{5.18}$$

Hesaplamalar sırasında kullanılacak eşitlikler ortaya konulduktan sonra eğitim veri kümelerinin sınıflara göre bölünmesi ile oluşan $G_i$ 'nin özniteliklerini temsil eden $G_{ij}$ 'ler için varyans hesabı yapılmalıdır.

$$var(x) = s^2 = \frac{1}{N-1} \sum_{1}^{N} (x_i - \bar{x})^2 \tag{5.19}$$

Her $G_{ij}$ özniteliği için hesaplanan $var(G_{ij})$ varyansı $V_i$ genel varyans matrisi içerisindeki yerini almaktadır.

$$V_i = \begin{bmatrix} var(G_{11}) & var(G_{12}) & \cdots & var(G_{1n}) \\ var(G_{21}) & var(G_{22}) & \cdots & var(G_{2n}) \\ \vdots & \vdots & \ddots & \vdots \\ var(G_{rank(R)1}) & var(G_{rank(R)2}) & \cdots & var(G_{rank(R)n}) \end{bmatrix} \quad \textbf{(5.20)}$$

Bu süreç sonrası $V_i$ varyans matrisi elde edilmiştir. Elde edilen V varyans matrisi ile değişimi gözlemlemek amacıyla karşılaştırma yapılması gereken YV varyans matrisi oluşturulmalıdır. Bu matris için test düğümü, $G_i$ kümelerine eklenir ve yeniden varyans hesaplanır. Test düğümünün $G_i$ kümesine eklenmesi ile elde edilen $YG_i$ matrisi üzerinden yeniden varyans hesaplamaları yapılır.

$$YG_i = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{m_i1} & A_{m_i2} & \cdots & A_{m_in} \\ T_{g1} & T_{g1} & \cdots & T_{gn} \end{bmatrix} \quad \textbf{(5.21)}$$

$YG_i$ matrisi içersindeki özniteliklerin varyanslarının $var(YG_{ij})$ hesaplanması ile elde edilen YV matrisi aşağıdaki gibidir.

$$YV_i = \begin{bmatrix} var(YG_{11}) & var(YG_{12}) & \cdots & var(YG_{1n}) \\ var(YG_{21}) & var(YG_{22}) & \cdots & var(YG_{2n}) \\ \vdots & \vdots & \ddots & \vdots \\ var(YG_{rank(R)1}) & var(YG_{rank(R)2}) & \cdots & var(YG_{rank(R)n}) \end{bmatrix} \quad \textbf{(5.22)}$$

Bir sonraki adımda, $T_g$ test düğümünün $G_i$ 'lere eklenerek hesaplanan $YV_i$ ve $V_i$ varyans matrisleri arasındaki farkların oranları tespit edilmektedir. Bu oranlar

toplanarak oranlar toplamı $OT_{ij}$ elde edilir.

$$OT_{ij} = \sum_{j=1}^{n} \left| (var(G_{ij}) - var(YG_{ij})) / var(G_{ij}) \right| \quad \textbf{(5.23)}$$

Elde edilen rank(R) adet $OT_{ij}$ arasından en küçük olanınını hesaplarken kullandığımız $G_i$ bu aşamadan sonra üzerinde çalışılacak eğitim veri kümesi olarak kabul edilir, $YA_{cxn}$ ve diğer $G_i$ 'ler dışlanır.

$$YA_{cxn} = \left\{ G_i \ : \ G_i \Rightarrow min(OT_{ij}) \right\} \quad \textbf{(5.24)}$$

Bu aşamadan sonra, yeni eğitim veri kümesi içerisinden her test düğümü için öklit uzaklığı ile en yakın eğitim düğümü, $EYED_i$ tespit edilecektir . $EYED_i$ Öklit uzaklığı $d(q, p)_i$ temel alınarak tesbit edilmektedir.

$$d(q, p)_i = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2} \quad \textbf{(5.25)}$$

$$EYED_i = \left\{ YA_i \ : \ YA_i \Rightarrow min(\sqrt{\sum_{j=1}^{n} (T_{ij} - YA_{ij})^2}) \right\} \quad \textbf{(5.26)}$$

$YA_{cxn}$ içerisinden $T_i$ için bulunan $EYED_i$ noktasının eşleştiği endeks düğümü, bu yöntem içerisinde yeni endeks düğüm arayışındaki başlangıç noktası olacaktır. Bu düğümü oluşturan noktalar üzerinden HF fonksiyonu ile hareket ederek uygun noktalar tespit edilecektir.

$$\hat{Y}_{zi} = HF(EYED_{ij}) = \begin{cases} I_{ij} + I_{ij} \dfrac{(EYED_{ij} - T_{iz})}{EYED_{ij}}, & EYED_{ij} \neq 0 \\ I_{ij}, & EYED_{ij} = 0 \end{cases} \quad \textbf{(5.27)}$$

Noktaların tespiti sonrası $YI_z = \{\hat{Y}_{z1}, \hat{Y}_{z2}, \ldots, \hat{Y}_{zn}\}$ eşitliği bize yeni endeks düğümünü vermektedir.

### 5.3.2.2 Mahalanobis uzaklığına dayalı tahminsel yaklaşım

Bu yöntem, eğitim veri kümesi içerisinde yer alan farklı sınıfların $G_i$ merkezlerinin ve eğitim veri kümesi içindeki düğümlerinin, test düğümlerine olan uzaklıklarının Mahalonobis metriğiyle ölçülmesi ve yeni endeks düğümü, $YI_z$ tespit etmek için iki nokta tespit edilmesi mantığına dayanmaktadır.

Bu yöntem içerisinde tespit edilecek ilk nokta, mahalonobis metriğine göre küme merkezi, test düğümüne en yakın olan $G_i$ içerisindeki yine mahalonobis metriğine göre test düğümüne en yakın olan eğitim düğümüdür. İkinci tespit edilecek nokta ise, eğitim veri kümesi içerisinde mahalonobis metriğine göre test düğümüne en yakın düğümdür.

Seçilen bu iki noktanın, test düğümüne olan uzaklıklarının oranı ile bu düğümlere karşılık gelen endeks düğümleri ağırlıklandırılmakta ve bu ağırlıklandırılmış iki endeks düğümü öznitelikleri toplanmaktadır. Bu toplama sonrası elde edilen nokta test düğümü için üretilen yeni endeks düğümü olarak kabul edilmektedir.

Test edilecek düğümün $G_i$ küme merkezine olan mahalanobis uzaklığının formülü aşağıdaki gibidir.

$$DM((T^T)_i, \mu_{G_i}) = \sqrt{((T^T)_i - \mu_{G_j})^T S_{G_j}^{-1} ((T^T)_i - \mu_{G_j})} \quad \textbf{(5.28)}$$

En yakın kümenin $G_i$ nin elemanları ile test edilen düğüm arasıdaki uzaklığı aşağıdaki formül vermektedir.

$$DM((T^T)_i, G_{jp}) = \sqrt{((T^T)_i - G_{jp})^T S_{G_j}^{-1} ((T^T)_i - G_{jp})} \quad \textbf{(5.29)}$$

Tespit edilecek ikinci nokta olan $A_{mxn}$ içerisindeki mahalanobis uzaklığına göre en yakın düğümün hesaplanması için kullanılan formül (V.30) de verilmiştir.

$$DM((T^T)_i,(A^T)_i) = \sqrt{((T^T)_i-(A^T)_i)^T S^{-1}((T^T)_i-(A^T)_i)} \quad \textbf{(5.30)}$$

Birinci noktanın tespiti, en yakın sınıf merkezinin bulunması,

$$EYKM_i = \left\{ G_j \ : \ G_j \Rightarrow min(DM_j((T^t)_i, \mu_{G_i})) \right\} \quad \textbf{(5.31)}$$

ve bu sınıf içerisindeki en yakın noktanın tespit edilmesi ile elde edilir.

$$EYKE_i = \left\{ G_{jp} \ : \ G_{jp} \Rightarrow min(DM_j(T_i, G_{jp})) \right\} \quad \textbf{(5.32)}$$

İkinci nokta ise tüm eğitim veri kümesi içerisinden en yakın olanının tespit edilmesi ile elde edilir.

$$EYED_i = \left\{ (A^t)_j \ : \ (A^t)_j \Rightarrow min(DM(T_i,(A^t)_j)) \right\} \quad \textbf{(5.33)}$$

Elde edilen $EYKE_i$ ve $EYED_i$ düğümleri $(T^T)_i$ düğümüne olan uzaklıklarına orantılı olarak $YI_z$ oluşturulurken kullanılırlar. Bu orantı (5.34) de verildiği gibi hesaplanmaktadır ve $HO_i$ bu orantıyı simgelemektedir.

$$HO_i = \frac{DM(T_i, EYKE_{ij})}{DM(T_i, EYKE_{ij}) + DM(T_i, EYED_{ij})} \quad \textbf{(5.34)}$$

Elde edilen $HO_i$ orantısı aşağıdaki eşitlikteki yerine konarak düğümü oluşturan noktalar tesbit edilir.

$$\hat{Y}_{zi} = HF(EYKE_{ij}, EYED_{ij}) = EYKE_{ij} HO_i + EYED_{ij}(1 - HO_i) \quad \textbf{(5.35)}$$

Noktaların belirlenmesi sonrası $YI_z = \{\hat{Y}_{z1}, \hat{Y}_{z2}, \ldots, \hat{Y}_{zn}\}$ eşitliği yeni endeks düğümünü vermektedir.

### 5.3.2.3 Öklit uzaklığına dayalı yöntem

Bu yöntem test edilecek düğüme öklit uzaklığıyla en yakın olan eğitim düğümünün bulunması ve bu düğümün eşleştiği endeks düğümünden hareketle yeni bir endeks düğümü yaratılması mantığına dayanmaktadır.

Bu yöntemde önemli olan HF fonksiyonunun düzenlenmesidir. HF fonksiyonu referans noktasından, test düğümünün öznitelikleri ve eğitim düğümünün özniteliklerinin karşılaştırılması ya da oranlanması ile elde edilen rakamları kullanarak hareket ederek, yaratılacak yeni endeks düğümünün yerini bulur.

Bu çalışmada HF fonksiyonu oransal yaklaşımla elde edilmiştir. Bu yaklaşım ile öklit metriğine göre eğitim veri kümesi içerisinden en yakın eğitim düğümü seçilmektedir. Seçilen eğitim düğümü ve test edilecek test düğümü öznitelikleri arasındaki farkın oranı ile eğitim düğümüne karşılık gelen endeks düğümü öznitelikleri ağırlıklandırılır.

Bu ağırlıklandırma sonrası elde edilen yeni düğüm test düğümü için yeni oluşturulan endeks düğümü olarak kabul edilir. Oluşturulan endeks düğümünün yeri ile ilgili herhangi bir kısıt bulunmadığından istenilir ise. bu endeks düğümü eğitim düğümüne karşılık gelen endeks düğümünün asimptotik komşuluğunda olacak şekilde seçilebilir.

$$EYED_i = \left\{ (A^T)_i :\ (A^T)_i \Rightarrow min\left(\sqrt{\sum_{j=1}^{n} (T_{ij} - A_{ij})^2}\right) \right\} \quad \textbf{(5.36)}$$

$$\hat{Y}_{zi} = HF(EYED_{ij}) = \left\{ \begin{array}{ll} I_{ij} + I_{ij} \dfrac{\dfrac{(EYED_{ij} - T_{iz})}{EYED_{ij}}}{n^2}, & EYED_{ij} \neq 0 \\ I_{ij}, & EYED_{ij} = 0 \end{array} \right\} \quad \textbf{(5.37)}$$

Buradaki n, test düğümü sayısıdır. Noktaların tespiti sonrası $YI_z = \{\hat{Y}_{z1}, \hat{Y}_{z2}, \ldots, \hat{Y}_{zn}\}$ eşitliği bize yeni indeks düğümümüzü vermektedir.

# 6. YSYBMG MASAÜSTÜ UYGULAMASI

YSYBMG masaüstü uygulaması, üzerinde YSYBMG yöntemi ve bu tez çalışması içerisinde önerilen modelleme yöntemlerinin kullanılması ve sınanması amacını taşımaktadır. Kullanıcıların veri kümesini ayrıntıları ile inceleyebilmesi ve gerekli görülür ise veri kümesi üzerinde değişiklikler yapabilmesini mümkün kılmaktadır.

Bu durum kuşkusuz üzerinde çalışılan veri kümesinin içeriğinin ve veri kümesinin sayısal nicelikleri üzerinde yapılan değişikliklere yöntemlerin verecekleri tepkiyi ölçmek bakımından yararlı olacaktır. Bu gözlemler sonrası araştırmacılar yöntemleri daha yakından tanıma fırsatını yakalayacaklardır.

Bu uygulama içerisinde istenilir ise eğitim veri kümesi ve test edilecek veri kümesi ayrı ayrı girilebilmekte, eğer tek bir veri kümesi kullanılacak ise uygulama uygun endeks uzayı oluşturarak veri kümesini endeks uzayı boyutlarına göre bölebilmektedir. Bu bölünmeye ilişkin bilgi selectAttribute arayüzü içerisinde görülebilmektedir. Eğer farklı bir bölünme sayısı istenilir ise, endeks uzayı sınırları mainGui arayüzü içerisinden girilerek istenilen bölünme sağlanabilmektedir.

Ön inceleme safhası sonrası veri kümesi üzerinde sınamaların yapılabileceği bir çalışma ortamı yaratmak iddiasında olan bu kullanıcı arayüzü bu anlamıyla yeni çalışmalar için yararlı olmak gibi bir misyonu taşırken aynı zamanda da, değişik veri kümeleri ile önerilen yöntemlerin farklı kombinasyonları ile sınanması gibi kendini denetime açmak gibi bir amaç taşır. Bu amaçla, uygulama içerisinde yöntemler arasında çapraz eşleşmeler yapılabilmekte ve bu yöntem kombinasyonlarının istenilen sayıda tekrar edilebileceği bir arayüz bulunmaktadır.

Bu arayüz sayesinde, denemenin tekrar edilmesi, sonuçların dosyalanması gibi zaman alıcı işler uygulama tarafından standartlaştırılarak çözülmekte ve kullanıcı için hızlı bir

deneme süreci yaratılmaktadır. Bu çoklu deneme süreci sonrası, her bir deneme için ".pdf" uzantılı bir rapor dosyası oluşturulmaktadır. Bu dosya veri kümesine ilişkin yapısal bilgileri, denemeye ilişkin bilgileri ve sonuçlara ilişkin istatistikleri tutmaktadır. Bu raporlar istenilen deney sayısında uygulanan yöntemlerin kullanılması ile adlandırılmış bir klasör içerisinde tutulmakta ve yöntemlerin her biri için ayrı klasör oluşturulmaktadır.

Ayrıca her bir yöntem kombinasyonu için denemelerin tamamından yola çıkarak ".pdf" uzantılı bir rapor hazırlanmaktadır. Bu rapor her bir yöntem kombinasyonu için ayrıca hazırlanmakta ve totalReport klasörü altında toplanmaktadır.

Oluşturulan son rapor ise resultreport.pdf ismi ile oluşturulan denemelerin karşılaştırılması için hesaplanmış değerlerini bir tablo halinde sunan pdf dosyasıdır. Bu dosya sayesinde deney sahibi algoritmaları karşılaştırabilme imkanını elde etmiş olur.

Oluşturulan bu klasörler de veri kümesinin ismi ile oluşturulmuş bir klasör içerisinde tutulmaktadır. Bunların haricinde üretilen rapor dosyaları içerisinde her algoritma eşleşmesi için denemeler sonrası en iyi ve en kötü sonuçların elde edildiği veri kümeleri eğitim ve test kümeleri olacak şekilde dosyalanmaktadır. Her algoritma eşleşmesi için toplam 4 adet veri kümesi dosyalanmaktadır.

Örnek dosyalama gösterimi;

- **iris/**
  - **class-multiregresion/**
    - iris-21.03.2011 10.18.20.pdf
    - iris-21.03.2011 10.18.22.pdf
  - **rank-matchMahalanobis/**
    - iris-21.03.2011 10.18.36.pdf
    - iris-21.03.2011 10.19.20.pdf

- **totalReport/**
  - iris-class-multiregresion 21.03.2011 10.18.30.pdf
  - iris-rank-matchMahalanobis 21.03.2011 10.18.30.pdf
- resultreport.pdf
- **dataset/**
  - **training/**
    - best-training-iris-class-multiRegresion.arff
    - worst-training-iris-class-multiRegresion.arff
  - **testing/**
    - best-testing-iris-class-multiRegresion.arff
    - worst-testing-iris-class-multiRegresion.arff

## 6.1 YSYBMG MASAÜSTÜ UYGULAMASI İÇİN YAPISAL BİLGİLER

YSYBMG masaüstü uygulaması Java programlama dili ile Swing kütüphaneleri kullanılarak yazılmıştır. Bu sayede Java sanal makinası kurulu olan her bilgisayarda işletim sistemi farkı gözetmeden, kurulum gerektirmeden çalışmaktadır. Bu uygulama içerisinde kullanılan tüm java kütüphaneleri ve araçları açık kaynaklı olarak kullanıma sunulmuş kütüphaneler ve araçlardır.

Bu durum sonraki zamanlar içinde geliştirme ortamının korunması ve geliştirme faaliyetlerinin sürdürülebilir olması bakımından önemlidir. Gelecek zamanlarda uygulamanın çatısı altında geliştirme çalışmaları yapılabilmesini mümkün kılmak bakımından kullanılması uygun görülen bu yönelim ile birlikte uygulama açık kaynak kodlu olarak dağıtılabilirdir. Bu uygulama içerisinde kullanılacak veri dosyaları .arff uzantılı olan dosyalardır. Bu dosya formatının kullanılmasının amacı, takip edilen ve içerisinde bulunulduğu düşünülen araştırma çevreleri tarafından tanınması ve kullanılmasıdır.

### 6.1.1 Yararlanılan ve İncelenen Harici Java Kütüphaneleri

**Apache commons-math-2.1 :** Açık kaynak kodlu java matematik kütüphanesidir. Bazı matematiksel işlemler için kullanılmıştır.

**Michael Thomas Flanagan's Java Scientific Library :** Ticari olmayan kullanıma açık bilimsel java kütüphanesidir, üzerinde incelemelerde bulunulmuştur.

**jfreechart-1.0.13 :** Açık kaynak kodlu grafik üretim aracıdır ve veri grafiklerinin üretilmesi amacı ile kullanılmıştır.

**Visual Swing :** Açık kaynak kodlu görsel Swing tasarım aracıdır ve kullanıcı arayüzünün tasarımında kullanılmıştır.

**javastat :** Açık kaynak kodlu Java istatistik kütüphanesidir ve üzerinde incelemelerde bulunulmuştur.

**javaml-0.1.6 :** Açık kaynak kodlu Java özdevinimli öğrenme kütüphanesidir ve bazı matematiksel işlemler için kullanılmıştır.

**Weka :** Açık kaynak kodlu veri madenciliği platformudur ve kaynak kodları .arff uzantılı veri dosyalarının uygulamaya yüklenmesi amacı ile kullanılmıştır.

**iText-2.1.5 :** Açık kaynak kodlu Java pdf formatlı dosya oluşturma aracıdır ve .pdf uzantılı raporların üretimi için kullanılmıştır.

**metaquant :** Açık kaynak kodlu java istatistiksel analiz kütüphanesidir ve üzerinde incelemelerde bulunulmuştur.

**Junit 3.0 :** Açık kaynak kodlu yazılım test aracıdır ve birim testler için kullanılmıştır.

**Maven 2 :** Proje yönetim aracıdır ve projenin üzerinde yapılandırıldığı araçtır.

**Log4j :** Gerçek zamanlı yazılım hatalarını kayıt etmeye yarayan araçtır .

## 6.2 UYGULAMANIN TANITILMASI

YSYBMG masaüstü uygulaması temel olarak beş kullanıcı arayüzünden oluşmaktadır.

- indexingHDMR
- experiment
- ihdmr
- selectAttribute
- visualData

### 6.2.1 indexingHDMR Arayüzü

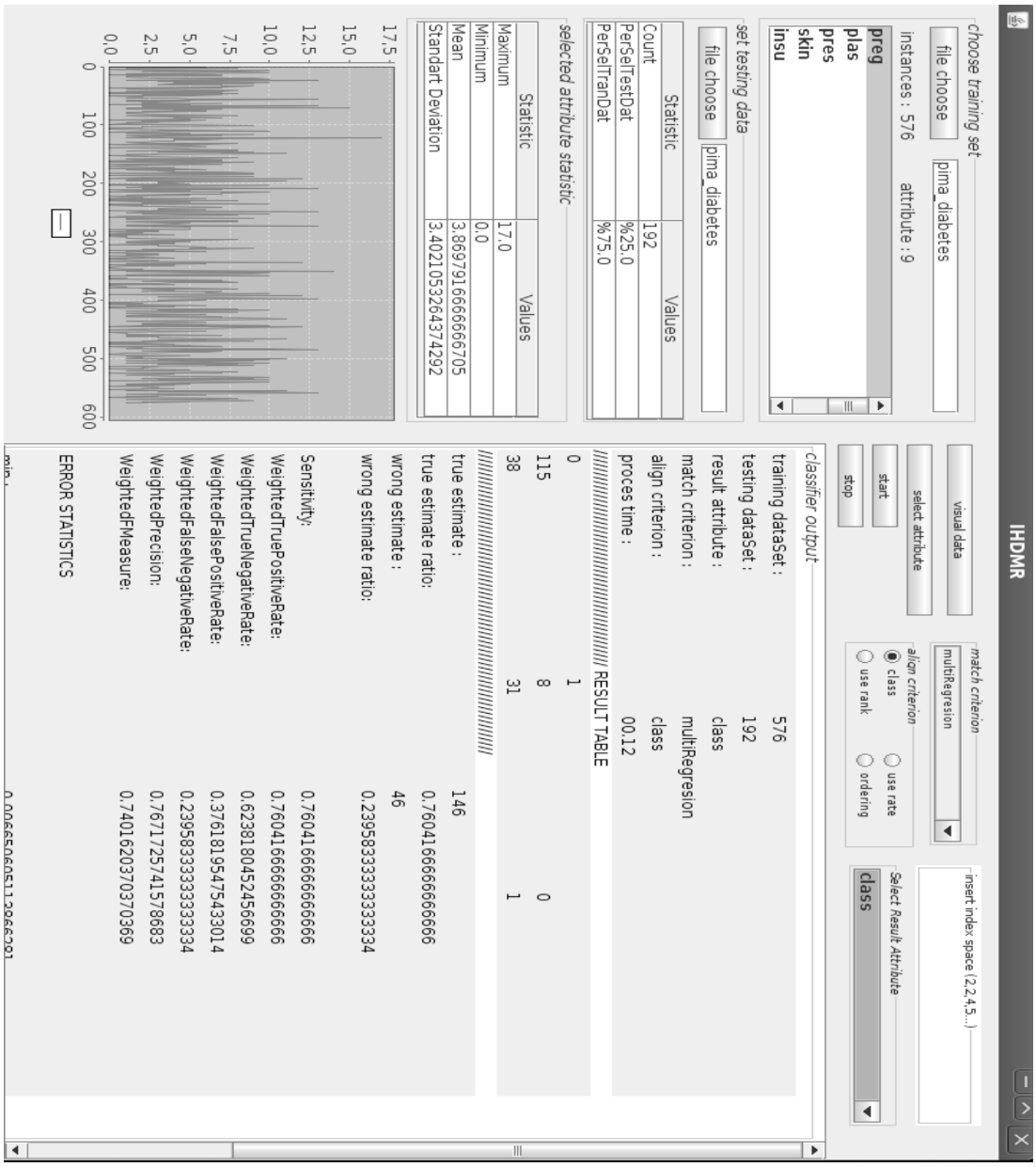
Bu kullanıcı arayüzü, ana arayüzdür ve uygulama Şekil VI.4 deki arayüz ile başlamaktadır. Diğer iki kullanıcı arayüzüne bu kullanıcı arayüzünden ulaşılmaktadır. Bu arayüz üzerinde iki tane düğme bulunmaktadır.


**Şekil 6.1 : indexingHDMR Arayüzü**

"ihdmr" Düğmesi analize yönelik arayüzünün açılmasını sağlar. "experiment" Düğmesi, çoklu denemeye yönelik arayüzün açılmasını sağlar.

### 6.2.2 ihdmr Arayüzü

Bu Şekil 6.5 de görünen arayüz, veri kümesi üzerinde detaylı inceleme yapmak isteyen ve/veya veri kümesinin sayısal niteliklerini değiştirecek kullanıcılara yönelik hazırlanmıştır. Bu amaçla bu arayüz üzerinden selectAttribute ve visualData arayüzlerine ulaşılmakta ve bu arayüzlerde veri kümesi üzerinde yapılan değişikliklerle birlikte analize tabi tutulacak kümesinin son hali belirlenebilmektedir.




**Şekil 6.2 : IHDMR Arayüzü**

Bu arayüz üzerinden eğitim veri kümesi ve test edilecek veri kümesi ayrı ayrı girilebilmektedir. Eğer tek bir veri kümesi üzerinden analizin tamamlanması isteniyorsa, uygulama girilen eğitim kümesini %66,6 eğitim veri kümesi, %33,3 test veri kümesi oranlarına mümkün olduğunca yakın bir şekilde bölecektir.

Bu bölünmenin temeli oluşturulabilecek endeks uzayı olduğundan uygulama veri kümesini, veri kümesinden elde ettiği bilgilerle kendi içerisinde oluşturduğu endeks uzayının boyutlarına göre ikiye bölmektedir. İlk bölüm endeks uzayı boyutunda olan eğitim veri kümesi diğer bölüm ise test için kullanılacak veri kümesi olarak atanmaktadır.

Uygulama endeks uzayı için iki farklı algoritma kullanmaktadır. Birinci algoritma proCent.java sınıfı içerisinde oluşturulmuş olan algoritmadır. Bu algoritma tek bir veri seti ile analiz yapıldığında uygun indeks uzayı sınırlarını üretir. Bu sınırlar yukarıda belirtilen şekliyle %66,6 eğitim veri kümesi %33,3 test veri kümesi oranlarına mümkün olduğunca yakın olacak şekilde belirlenmeye çalışılır. Veri kümesi bölünmesi bu algoritmadan dönen sonuçlarla oluşturulan endeks uzayının boyutlarına göre yapılır.

İkinci algoritma proCentOne.java sınıfı içerisinde oluşturulmuş olan algoritmadır. Bu algoritma, eğitim veri kümesi ve test için kullanılacak veri kümesi ayrı ayrı uygulamaya yüklendiği analiz süreçlerinde endeks uzayı boyutlarını üretir.

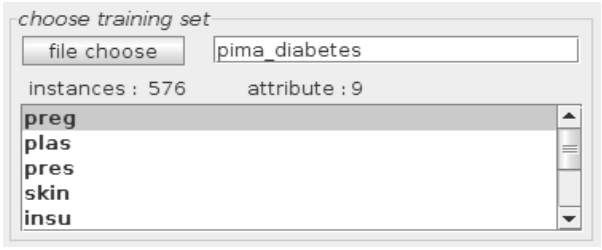
Bu algoritmanın amacı, eğitim veri kümesi ile eşleşecek bir endeks uzay için sınır noktaları belirlemektir. Bu sebeple, algoritma eğitim veri kümesine en yakın endeks uzayı için sınırlar üretmeye çalışır, bu endeks uzayı boyutu eğitim veri kümesinden daha büyük olamaz ve üretim bu kurala göre yapılır. Eğer üretilen endeks uzayı boyutu eğitim veri kümesinden küçük ise eğitim veri kümesi içerisinden eşitliği sağlayacak sayıda düğüm rastgele olarak seçilerek silinir.

Bu uygulama içerisinde endeks uzayı için kullanıcı girdisi de kullanılabilmektedir. Eğer kullanıcı endeks uzayı sınırlarını kendi belirlemek ister ise, uygulama belirtilen

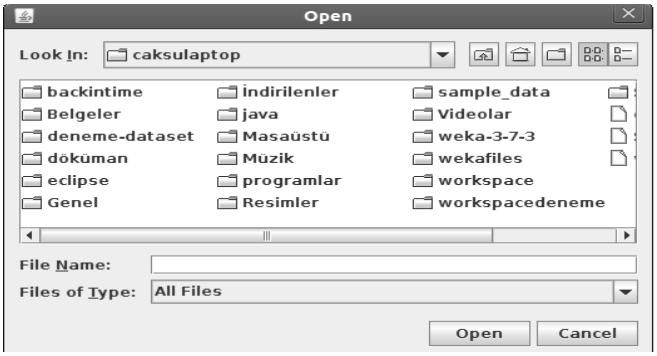
sınırlara göre endeks uzayı üretilmekte ve analiz bu endeks uzayının sayısal niteliklerine bağlı kalınarak yapılmaktadır. Bu arayüz üzerinde veri kümesinin istatistik bilgileri ve grafikleri görülebilmektedir. Veri kümesi içindeki düğümlerin sınıflarını belirten öznitelik bu arayüz içerisinden seçilmektedir. Uygulanacak algoritma kombinasyonu belirlenebilmektedir. Analiz sonrası sonuçlar bu arayüz üzerinden okunabilmektedir.

### 6.2.2.1 Arayüz içeriği

Şekil 6.6' da görünen "choose training set" birleşeni, eğitim kümesinin uygulamaya yüklenmesi ve tanıtılması işinin yürütüldüğü ana birleşendir. Bu birleşen üzerindeki "file choose" düğmesi Şekil 6.7 görünen dosya seçiciyi açarak kullanılacak veri dosyasının seçilebilmesini sağlamaktadır. Seçilen veri dosyasının ismi yanındaki metin boşluğuna yazılmaktadır.

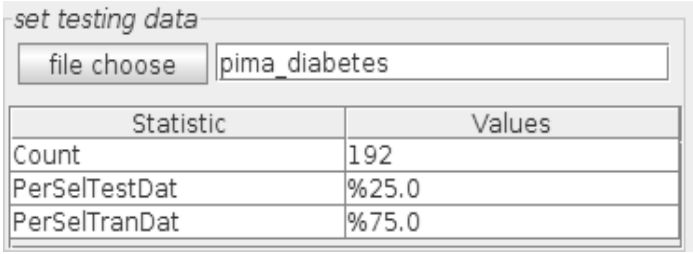



**Şekil 6.3 : "choose training set" birleşeni.**




**Şekil 6.4 : Dosya seçici**

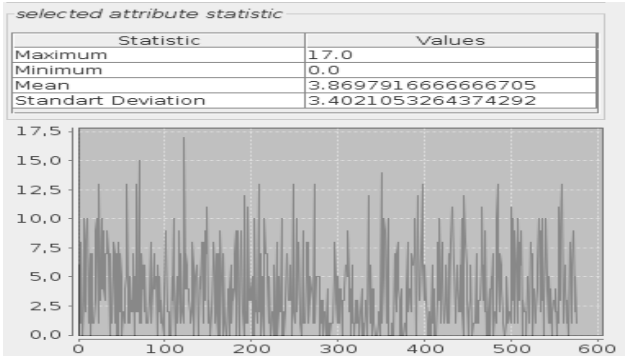
Test veri kümesinin uygulamaya yüklenmesi ve tanıtılması işinin yürütüldüğü ana bileşen Şekil 6.8' de görünen "set testing data" bileşenidir. Test veri kümesi girişi isteğe bağlıdır. Eğer test için bir veri girişi yok ise uygulama eğitim veri kümesini endeks uzay ile eşleşecek şekilde bölecek ve geri kalan veriyi test verisi olarak kullanacaktır.




**Şekil 6.5 : "set testing data" bileşeni.**

"set testing data" bileşeni içerisinde test veri kümesinin, eğitim veri kümesine oranı ve düğüm sayısı gibi istatistikler verilmektedir.

Şekil 6.9' da görünen "selected attribute statistic" bileşeni seçili olan özniteliği görselleştirmektedir.

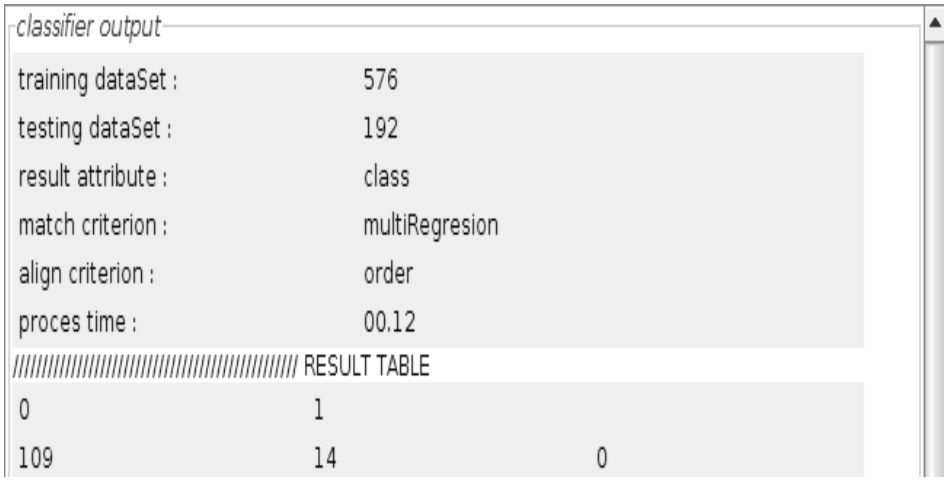



**Şekil 6.6 : "selected attribute statistic" birleşeni.**

Arayüz içerisinde eşleştirme algoritmasının seçilebileceği bölüm, "match criterion" bölümüdür. Arayüz içerisinde sıralama algoritmasının seçilebileceği bölüm, "align criterion" bölümüdür. Arayüz içerisinde analizler sırasında kullanılacak olan endeks uzayının sınırlarının belirlenebileceği alan, "insert indeks space" bölümüdür.

Buradaki sınır belirleme olayı isteğe bağlıdır. Herhangi bir sınır belirlenmemiş ise uygulama kendiliğinden uygun endeks uzay sınırları atayacaktır. “select result attribute” açılabilir kutusu, veri kümesi içerisindeki örneklerin sınıflarını belirten öz niteliğin belirlenmesini sağlar.

Analiz sonuçlarının kullanıcıya gösterildiği alan, Şekil 6.10' da gösterilen “classifier output” alanıdır. Bu alan içerisinde yapılan deneme ve denemenin sonuçları ile bilgiler kullanıcıya gösterilmektedir.

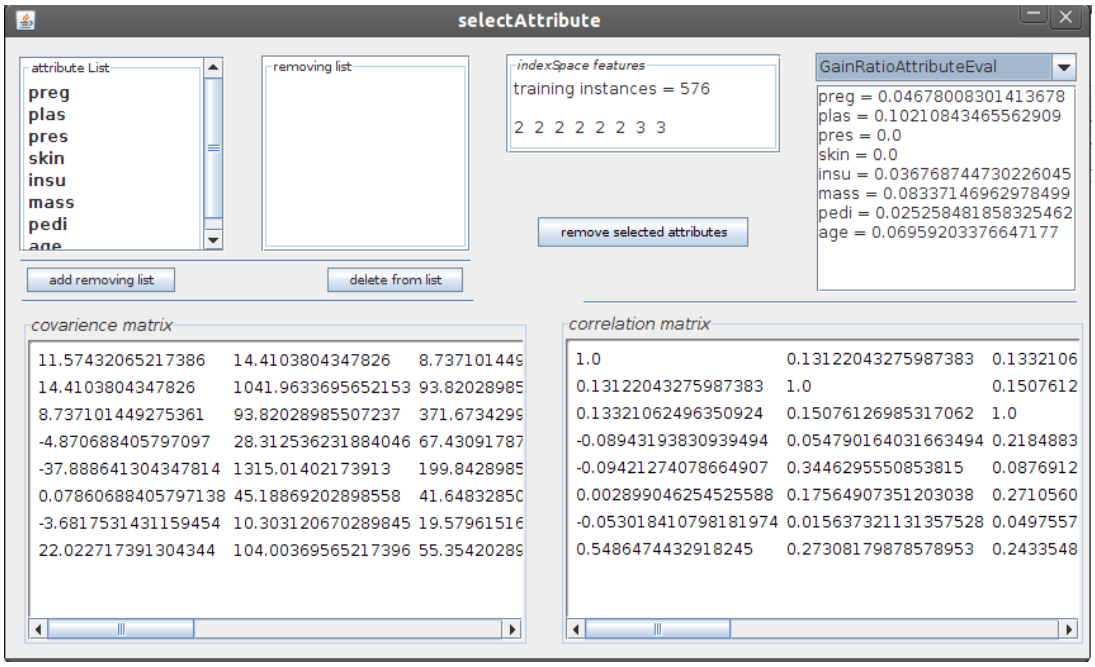



**Şekil 6.10 : Sonuç gösterim alanı.**

### 6.2.3 Select Attribute

Şekil 6.11' de görünen bu arayüz, kullanıcının kullanacağı veri kümesi içerisindeki öznitelikleri sayısal olarak tanımasını ve özniteliklerin analize dahil edilip edilmeyeceğine karar vermesini sağlamaktır. Kullanıcı bu arayüz içerisinde analiz içerisine dahil edilmesini istemediği öznitelikleri veri kümesinden dışlayabilmektedir. Bu karar analiz sürecinin sonucunu doğrudan etkileyeceği için bu kararı alacak olan kullanıcıya karar verme sürecinde etkili olacak sayısal veriler bu arayüz içerisinde sağlanmaktadır.

Bu arayüz içerisinde öncelikle dikkat edilmesi gereken veri endeks uzayı sınırlarıdır. Endeks uzayı sınırları veri kümesinin boyutuna göre düzenlendiği için bazı endeks uzayı sınırları istenilenden küçük olabilir. Eğer eğitim kümesinin içerdiği düğüm sayısı öznitelikleri temsil edebilecek sayıda endeks düğümü oluşturmaya yeterli değil ise bazı endeks uzayı sınırlar 1 olarak sabitlenir ve analiz içerisindeki etkinliklerini kaybederler. Bu sebeple istenirse veri kümesinden 1 olarak sabitlenmiş her endeks uzayı sınırı için bir öznitelik silinebilir. İstenirse düşük seviyeli görülen boyutları yükseltmek için de bu işlem yapılabilir.




**Şekil 6.11 : "select attribute" Arayüzü.**

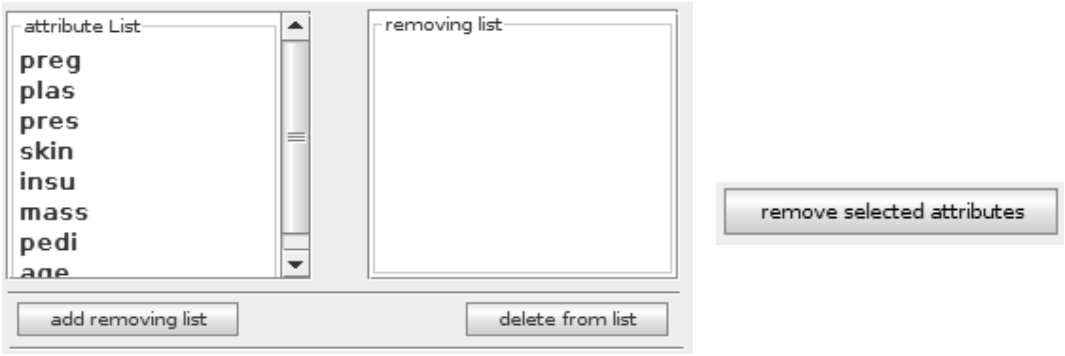
Arayüz içerisinde hangi özniteliğin silineceği konusunda karar verebilmesi için kullanıcıya bazı veriler verilmektedir. Bunlardan korelasyon ve kovaryans matrisleri öznitelikler arasındaki bağlantıyı görmek için kullanılabilmektedirler. Bu ilişki korelasyon matrisi $R_{mxn}$ içerisinden daha net görülebilmektedir.

Korelasyon katsayısı $R_{ij}$ , -1 ve 1 arasında tanımlıdır. Değer -1 'e yakın ise ilişki ters yönlü, 1 'e yakın ise ilişki doğrusaldır. Eğer $R_{ij}$ sıfır ise değişkenler arasında ilişki yoktur. Bu değerler içerisinden rahatlıkla ile birbiri ile ilişkili öznitelikler

seçilebilecektir. Bu matrisler dışında öznitelik elemesinde kullanılacak diğer ölçütler bir açılabilir kutu içerisinde kullanıcı ilgisine sunulmuştur.

#### 6.2.3.1 Arayüz içeriği

Şekil 6.12' de görünen "attribute add-remo" bölümü içerisinde öznitelikleri gösterilir ve silinecek özniteliklerle ilgili işlemler yapılabilir. "attribute List", veri kümesi içerisindeki özniteliklerin "removing List" ise silinecek özniteliklerin listesidir.




**şekil 6.12:"attribute add-remo" birleşeni.**

"remove selected attributes" düğmesi "removinglist" içerisindeki öznitelikleri veri kümesinden siler. Bu işlemin geri dönüşü yoktur.

"indexSpace features" alanı içerisinde uygulamanın önerdiği endeks uzayı sınırları ve buna karşılık gelen eğitim veri kümesinin içereceği örnek sayısı gösterilmektedir. Bu sayıların önemi, veri kümesi içerisindeki örnek sayısının yetersiz olduğu analiz süreçlerinde gereksiz özniteliklerin atılması ile daha iyi bir sonuç alınabilecek bir veri kümesi yaratılabilmesi için kullanılabilmeleridir.

"attribute ratio" bileşeni içerisinde özniteliklerin önemleri değişik algoritmalarla ölçeklenmektedir. Bu algoritmalar aşağıda verilmiştir:

- GainRatioAttributeEval
- ChiSquaredAttributeEval
- InfoGainAttributeEval
- PrincipalComponents
- SVMAttributeEval
- ReliefFAttributeEval
- OneRAttributeEval

Bu algoritmalar Weka veri madenciliği aracı kütüphanelerinden kullanılmıştır. Seçilen algoritma için liste içerisinde özniteliklerin algoritmalara göre rakamsal karşılıkları görünmektedir.

Arayüz üzerinde bulunan “covarience matrix” alanı, veri kümesi içerisindeki öznitelikler kovaryans matrisini göstermektedir. “correlation matrix” alanı ise, veri kümesi içerisindeki özniteliklerin korelasyon matrisini göstermektedir.

### 6.2.4 visualData Arayüzü

Şekil 6.13' de görünen bu kullanıcı arayüzünün temel amacı kullanıcının üzerinde çalıştığı veri kümesini görebilmesini, görsel olarak inceleyebilmesini sağlamak ve istenmeyen örneklerin veri kümesinden çıkarılmasını mümkün kılmaktır.

Arayüz üzerindeki veri tablosu, veri kümesinin sayısal olarak görselleşmesini amaçlamaktadır. Tablonun ilk sütunu, sütunun işaretlenmesini sağlayan bir işaret kutusudur. Bir sonraki sütun satır numarası ve sonraki sütunlar veri kümesi içerisindeki özniteliklerdir. Tablonun ilk satırı özniteliklerin isimlerinden oluşmaktadır.

**Şekil 6.13:"visual Data" Arayüzü.**

Öznitelikler listesi içerisindeki öznitelikler seçilebilirdir. Bu arayüz içerisindeki işlemler öznitelik referansını bu listeden alırlar. Ayrıca seçili öznitelik görselleştirilir. Arayüz üzerinde seçili özniteliklerin istatistiklerinin gösterildiği bir tablo bileşeni mevcuttur. Arayüz üzerinde seçili özniteliklerin dağılımını gösteren bir grafik mevcuttur. “diz-seç-sil” bileşeni, öznitelik listesi içerisinde seçili olan öznitelik ile ilgili işlemler yapmaktadır. Öznitelikler sıralanabilir, seçilebilir ve silinebilirlerdir.

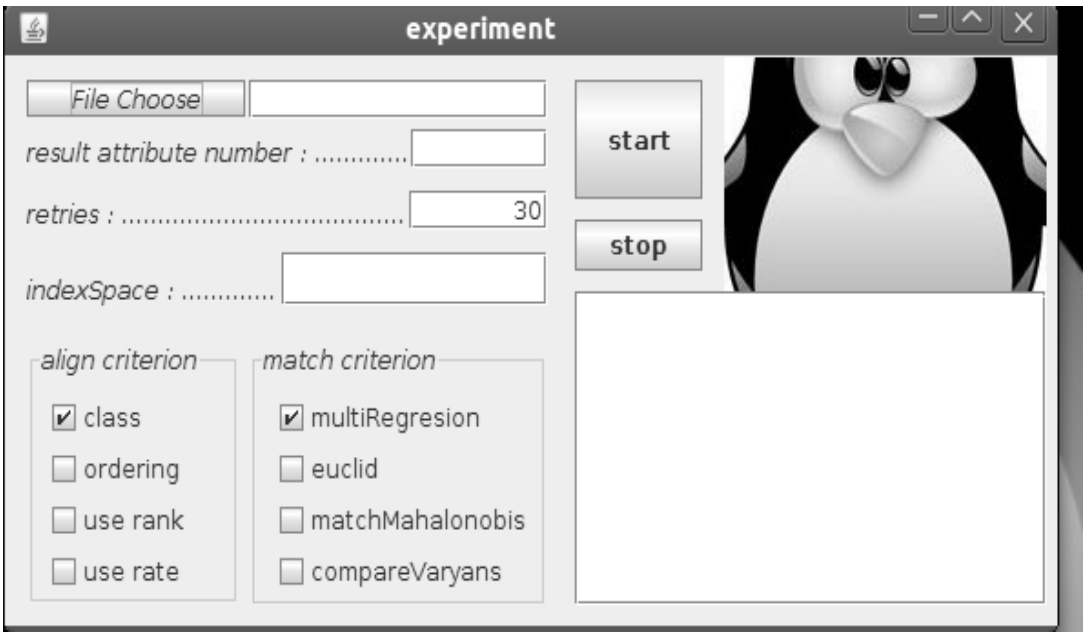

Bu bileşen içerisindeki <=, >=, = düğmeleri, sağ taraflarındaki karşılaştırılacak değer girilmesi için bırakılmış alan içerisindeki değeri seçili öznitelik değerleri ile karşılaştırır. Bu karşılaştırma işlemi “find” düğmesi ile başlar. Sonuçlar veri tablosuna yazılır. “Show data set” düğmesi elde kalan tüm veri kümesini gösterir. “Sort” düğmesi, seçili özniteliğe göre veri kümesini küçükten büyüğe doğru sıralar. “reverse” düğmesi, sort düğmesinin tam aksi yönde çalışır ve veri kümesini seçili özniteliğe göre büyükten küçüğe doğru sıralar. “discard” düğmesi , seçili olan satırların seçilmişliklerini ortadan kaldırır. “select” düğmesi, görünen satırları seçer. “delete” düğmesi, seçili olan satırları siler. Bu işlemin geri dönüşü yoktur.

### 6.2.5 experiment Arayüzü

Şekil 6.15' de görünen bu arayüzün amacı çoklu deneme yapabilmektir. Bu amaçla sadece çok gerekli olan bilgiler kullanıcıdan alınır ve sonuçlar dosyalanır. Bu felsefenin dayandığı nokta, kuşkusuz kullanıcı ile daha az iletişime geçen ve bu sayede zaman kaybını asgari düzeye çeken bir anlayışın içinde olmaktır.

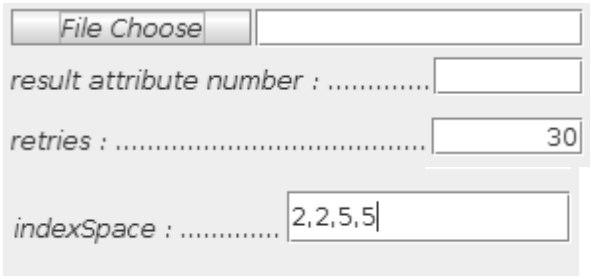
Kullanıcıdan hangi eşleştirmelerle çalışmak istediği ve bu eşleşmeleri kaçar defa denemek istediği bilgileri alınır ve uygulama duraksamadan rapor dosyalarını oluşturmayı bitirinceye kadar devam eder.

Bu arayüz kullanılırken uygulama sona erdirilirse rapor dosyaları eksik kalır ve algoritma kombinasyonları için oluşturulacak genel raporlar eğer rapor dosyalarının oluşturulması tam olarak bitmediyse oluşturulamaz.

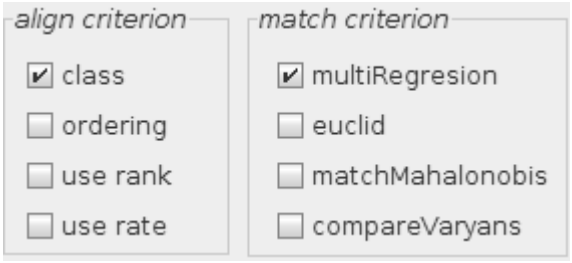



### 6.2.5.1 experiment arayüzünün içeriği

Şekil 6.16' da görünen kullanıcı bilgilerinin girileceği alan içerisinde kullanıcıdan alınması gereken bilgiler kullanıcıdan istenmektedir. Bu bilgilerin eksikliği durumunda uygulama analiz sürecini başlatmayacaktır. Kullanıcı tarafından, analiz sırasında kullanılacak olan endeks uzayı sınırlarının belirtilmesi için, aşağıdaki kullanıcı arayüzü bileşeni kullanılmaktadır.




Şekil 6.17' de görünen seçim menüsünden kullanıcı denemek istediği kombinasyonları uygulamaya tanıtır. Tüm kombinasyonlar olasıdır. Fakat eşleştirme yapılmaya müsait olmayan bir seçimsizlik durumu olursa algoritma analiz aşamasını başlatmayacaktır.



# 7. UYGULAMALAR

Bu bölümde önerilen algoritmalar değişik veri kümeleri üzerinde denenmektedir. Yapılan deneyler ile önerilen algoritmalar hem kendi aralarında değerlendirmeye tabi tutulmakta hem de diğer bilinen algoritmalar ile karşılaştırmaktadır. Bu karşılaştırmaların ana amacı algoritma kombinasyonlarının farklı veri kümelerine verdikleri tepkileri sınamak ve bu tepkileri bilinen sınıflandırma algoritmalarıyla karşılaştırarak algoritmaların belirgin özelliklerini ve performanslarını ortaya çıkarmaktır.

Bu denemeler süreci içerisinde 7 farklı veri kümesi üzerinde önerilen 16 farklı algoritma kombinasyonu ve bilinen 4 farklı sınıflandırma algoritması kullanılarak gerçekleştirilmiştir.

Denenen algoritmalar içerisinden bu tezin önerdiği algoritmalar ile ilgili denemeler bu çalışma için yazılmış olan YSYBMG veri madenciliği aracı ile, Weka içersinden alınmış 4 farklı profesyonel algoritma için yapılan denemeler ise Weka veri madenciliği aracı ile yapılmıştır.

Her bir veri kümesi ile kullanılan her bir algoritma kombinasyonu için test süreci 30 kere tekrarlanmıştır. Her tekrar için eğitim ve test veri kümesi seçimleri rastgele yapılmıştır. Sonuçlar, bu 30 rastgele tekrarın aritmetik ortalaması olarak alınmış ve gerekli hesaplamalar bu ortalamalar üzerinden yapılmıştır.

Bu deney süreci IHDMR yazılımı tarafından yönetilmekte ve sürecin başlangıcı ile sonu arasında bir kesinti olmadan sonuçlandırılmaktadır. Bu deney süreci sonrasında yazılım sonuçları dosyalanmaktadır. Dosyalama her bir deneme için bir pdf dosyası, her bir algoritma için denemeler toplamını içeren bir pdf dosyası ve sonuçlara ilişkin bir pdf dosyası olarak gerçeklenmektedir. Sonuç için oluşturulan pdf dosyası içerisinde kullanılan her bir algoritma için araştırılan değerleri içeren bir tablo bulunmaktadır.

## 7.1 KULLANILAN VERİ KÜMELERİ

Bu tez çalışması içerisinde 7 farklı veri kümesi kullanılmıştır. Bu veri kümeleri UC Irvine Machine Learning Repository içerisinden alınmıştır. Bu veri kümelerinin seçilmesi süreci içerisinde veri setlerinin içerdikleri örnek sayıları, öznitelik sayıları, sınıf sayıları gibi özellikleri dikkate alınmıştır.

Sözü geçen bu özellikler bakımından farklı veri kümelerinin seçilmesi bu tez çalışmasının sonuçlarının anlamlı olmasından önemlidir. Bu sebeple seçilen 7 veri kümesi söyledir; Balance Scale Weight & Distance Database -"balance-scale", Blood Transfusion Service Center Data Set -"blood", BUPA Liver Disorders -"bupa", Pima Indians Diabetes Database -"diabetes", Iris Plants Database -"iris", Teaching Assistant Evaluation -"tae", Deterding Vowel Recognition Data -"vovel". Seçilen veri kümelerinin yapısal özellikleri Tablo 7.1 den takip edilebilir.

**Tablo 7.1 : kullanılan veri kümeleri**

| | **Öznitelik sayısı** | **Örnek sayısı** | **Sayısal öznitelik** | **Kategorik öznitelik** | **Sınıf sayısı** |
|---|---|---|---|---|---|
| **balance-scale** | 4+sınıf | 625 | 4 | 0 | 3 |
| **blood** | 4+sınıf | 748 | 4 | 0 | 2 |
| **bupa** | 6+sınıf | 345 | 6 | 0 | 2 |
| **diabetes** | 8+sınıf | 768 | 8 | 0 | 2 |
| **iris** | 4+sınıf | 150 | 4 | 0 | 3 |
| **tae** | 5+sınıf | 151 | 1 | 4 | 3 |
| **vovel** | 13+sınıf | 990 | 10 | 3 | 11 |

Seçilen veri kümelerinin içerdikleri örnek sayıları için 990 üst ve 150 alt sınır olarak alınmıştır. Daha yüksek veya daha düşük örnek sayıları üzerinde denemeler yapmak mümkündür, fakat oluşturulan modellerin çalışırlığını göstermek bakımından bu sınırlar yeterlidir. Seçilen veri kümeleri öznitelik sayıları bakımından 5 farklı grup

oluşturmaktadırlar. Benzer öznitelik sayılarına sahip veri kümeleri 5 öznitelik içeren veri kümeleridir.

Bu veri kümelerinden “iris”, içerdiği örnek sayısı itibarı ile diğer ikisinden ayrılmaktadır. “balance-scale” ve “blood” veri kümeleri ise içerdikleri sınıf sayıları bakımından birbirinden ayrılmaktadırlar. “tae” veri kümesi içerdiği 4 kategorik öznitelik ile diğer veri kümelerinden ayrılmaktadır. “vovel” veri kümesi içerdiği sınıf sayısı bakımından diğerlerinden ayrılmaktadır.

## 7.2 KULLANILAN ALGORİTMALAR

### 7.2.1 Kullanılan Modeller

Bu tez çalışması içerisinde seçilen gerçek veri kümeleri üzerinde sınıflandırma analizinin yapılacağı veri modelleri, önerilen sıralama ve eşleştirme algoritmalarının çapraz eşleşmeleri ile elde edilen Tablo 7.2' de görünen 16 farklı algoritma kombinasyonu ile oluşturulmuştur. Bu anlamda 16 farklı veri modelleme yöntemi kullanılmıştır.

**Tablo 7.2 : Kullanılan algoritma kombinasyonları.**

| **Model kısaltması** | **Sıralama algoritması** | **Eşleştirme algoritması** |
|---|---|---|
| mr1 | class | multiRegresyon |
| mr2 | use rank | multiRegresyon |
| mr3 | use rate | multiRegresyon |
| mr4 | ordering | multiRegresyon |
| e1 | class | euclid |
| e2 | use rank | euclid |
| e3 | use rate | euclid |
| e4 | ordering | euclid |
| mm1 | class | matchMahalanobis |
| mm2 | use rank | matchMahalanobis |

| | | |
|---|---|---|
| mm3 | use rate | matchMahalanobis |
| mm4 | ordering | matchMahalanobis |
| cv1 | class | compareVaryans |
| cv2 | use rank | compareVaryans |
| cv3 | use rate | compareVaryans |
| cv4 | ordering | compareVaryans |

### 7.2.2 Karşılaştırma Yapılan Algoritmalar

Önerilen algoritmaların çapraz eşleşmeleri ile elde edilen veri modelleme yöntemlerinin karşılaştırılacağı bilinen profesyonel sınıflandırma algoritmaları Weka veri madenciliği aracı içerisinden seçilmiştir. Bu seçim, sınıflandırma algoritmaları sınıflandırma için uyguladıkları farklı yöntemler göz önüne alınarak yapılmıştır. Şekil 7.3 de görünen bu algoritmaların nitelikleri farklıdır. NaiveBayes algoritması bayes teoremi üzerinde çalışan bir algoritmadır. Kstar algoritması entropi temelinde çalışan bir sınıflandırma algoritmasıdır. OneR algoritması sınıflandırma işlemini kural tabanlı olarak yapmaktadır. J48 algoritması karar ağacı algoritmasıdır.

**Tablo 7.3 : Karışılaştırma için kullanılan sınıflandırma algoritmaları**

| Kısaltma | Algoritma | Nitelik |
|---|---|---|
| NB | NaiveBayes | Bayes teoremini temel almaktadır. |
| KS | KStar | Entropiyi temel almaktadır. |
| OR | OneR | Kural tabanlı çalışmaktadır. |
| J | J48 | Karar ağacı algoritması. |

### 7.2.3 Karşılaştırma Tablosunun İçeriği

Denenen algoritmaların sonuçlarının karşılaştırılması için oluşturulacak tablolar içerisinde 7 farlı karşılaştırma ölçütü bulunmaktadır. Bu ölçütler Weka içerisinde denenen algoritmalar ile karşılaştırılacağından Weka algoritma çıktıları ile uyumlu olarak seçilmiştir. Tablo 7.4 içerisinde verilen değerler “En iyi tahmin” değeri hariç

olmak üzere önerilen algoritma kombinasyonları için yapılan 30 denemenin ortalamaları üzerinden hesaplanmaktadır. "En iyi tahmin" değeri, yapılan 30 deneme içerisinden en çok sayıda doğru tahmin yapan denemenin doğru tahmin sayısıdır. En iyi tahmin denenen algoritmaların denemeler içerisinde elde ettikleri en yüksek başarı oranıdır.

**Tablo 7.4 : Karışılaştırma kriterleri**

| **kısaltma** | **kriter** | **tanım** |
|---|---|---|
| ei | En iyi tahmin | 30 deneme içerisindeki en iyi doğru tahmin oranı |
| du | Duyarlılık | $\frac{TP}{TP+FN}$ |
| tp | TP oranı | Ağırlıklı Doğru kabul oranı |
| tn | TN oranı | Ağırlıklı Doğru red oranı |
| ke | Kesinlik | $\frac{TP}{TP+FP}$ |
| f | F ölçüsü | $\frac{2\times Duyarlılık\times Kesinlik}{Duyarlılık+Kesinlik}$ |
| sh | Standart hata | $\sqrt{\frac{\sum_{i=1}^{n}(\hat{Y}_i-Y_i)^2}{N}}$, *Y sınıf değeri , $\hat{Y}$ ise tahmindir.* |

IHDMR yazılımı, denmeler içerisinden yüksek başarım sağlanan veri kümelerini dosyalamaktadır. Bu veri kümeleri Weka içerisinden karşılaştırma amaçlı kullanılan algoritmalar için yine Weka içerisinde denenmekte ve bu algoritmaların "ei" bölümlerine bu veri setleri ile elde edilen başarım oranı konulmaktadır.

Karşılaştırma yapılan algoritmalar için denemeler Weka veri madenciliği aracı üzerinde yapılmıştır. Bu algoritmalar için Weka içerisinde 30 deneme yapılmış ve sonuçlar bu 30 deneme ile hesaplanmıştır. Standart hatanın karekökü (RMSE) hesabı doğrudan elde edilen hata ile hesaplanmış ve herhangi bir standartlaştırma işlemi yapılmamıştır.

## 7.3 VERİ KÜMELERİNE GÖRE SONUÇLAR VE YORUMLAR

### 7.3.1 balance-scale Veri Kümesi

"Balance Scale Weight & Distance Database" veri kümesi sayısal tanımlı 4 adet öznitelik, bir sınıf belirten kategorik tanımlı öznitelik ve 625 örnekten oluşmaktadır. IHDMR yazılımının bu veri kümesi için önerdiği endeks uzayı sınırları {4, 4 , 5 , 5 } ve endeks uzayı boyutu 400' dür. Bu sınırlar dolayısı ile veri kümesi içerisinden 400 örnek eğitim veri kümesi olarak geri kalan 225 örnek de test veri kümesi olarak rastgele seçilerek atanmışlardır.

#### 7.3.1.1 Öznitelikler

"balance-scale" veri kümesinin öznitelikleri Tablo 7.5 da sınıf belirteci ise Tablo 7.6 de gösterilmiştir.

**Tablo 7.5 : balance-scale öznitelik istatistikleri.**

| isim | Nitelik | Maksimum | Minimum | Ortalama | St. Sapma |
|---|---|---|---|---|---|
| left-weight | sayısal | 5 | 1 | 3 | 1.14 |
| left-distance | sayısal | 5 | 1 | 3 | 1.41 |
| right-weight | sayısal | 5 | 1 | 3 | 1.42 |
| right-distance | sayısal | 5 | 1 | 3 | 1.42 |

Bu veri kümesinin sınıf belirteci içerisindeki örneklerin 3 sınıf altında toplandığını göstermektedir. Bu dağılım sınıflar arasında eşit bir paylaşımla olmamıştır. "B" sınıfı içerisindeki örneklerin toplam örnek sayısına oranı oldukça düşük kalmıştır. Bu durum eğitim sürecini etkilemesi olası bir etmendir. Test veri kümesi içerisine seçilecek "B" sınıfı içerisinde olan örnekler için tahmin oranının diğer sınıflara göre düşük olması durumu söz konusudur.

**Tablo 7.6 : balance-scale sınıf belirteci**

| değer | sayı | ağırlık |
|---|---|---|
| L | 288 | 0.46 |
| B | 49 | 0.80 |
| R | 288 | 0.46 |

### 7.3.1.2 Test sonuçları

**Tablo 7.7 : balance-scale deneme sonuçları.**

| | ei | du | tp | tn | ke | f | sh |
|---|---|---|---|---|---|---|---|
| **mr1** | 0.685 | 0.760 | 0.647 | 0.855 | 0.751 | 0.690 | 0.894 |
| **mr2** | 0.689 | 0.731 | 0.609 | 0.866 | 0.758 | 0.667 | 0.794 |
| **mr3** | 0.405 | 0.386 | 0.222 | 0.702 | 0.363 | 0.251 | 1.263 |
| **mr4** | 0.703 | 0.761 | 0.648 | 0.859 | 0.755 | 0.693 | 0.863 |
| **e1** | 0.805 | 0.844 | 0.763 | 0.853 | 0.793 | 0.756 | 0.708 |
| **e2** | 0.818 | 0.845 | 0.766 | 0.854 | 0.797 | 0.759 | 0.692 |
| **e3** | 0.641 | 0.711 | 0.583 | 0.878 | 0.794 | 0.653 | 0.800 |
| **e4** | 0.801 | 0.845 | 0.766 | 0.849 | 0.796 | 0.758 | 0.705 |
| **mm1** | 0.898 | 0.907 | 0.856 | 0.943 | 0.889 | 0.868 | 0.505 |
| **mm2** | 0.907 | 0.910 | 0.861 | 0.942 | 0.886 | 0.871 | 0.463 |
| **mm3** | 0.738 | 0.740 | 0.620 | 0.915 | 0.833 | 0.692 | 0.739 |
| **mm4** | 0.903 | 0.912 | 0.864 | 0.943 | 0.890 | 0.874 | 0.506 |
| **cv1** | 0.778 | 0.818 | 0.728 | 0.769 | 0.750 | 0.697 | 0.921 |
| **cv2** | 0.778 | 0.82 0 | 0.730 | 0.769 | 0.675 | 0.701 | 0.922 |
| **cv3** | 0.605 | 0.678 | 0.539 | 0.817 | 0.685 | 0.595 | 0.960 |
| **cv4** | 0.778 | 0.821 | 0.731 | 0.771 | 0.675 | 0.702 | 0.917 |
| **NB** | 0.875 | 0.900 | 0.900 | 0.910 | 0.830 | 0.860 | 0.435 |
| **K** | 0.844 | 0.880 | 0.880 | 0.900 | 0.810 | 0.850 | 0.486 |
| **OR** | 0.586 | 0.610 | 0.610 | 0.660 | 0.560 | 0.580 | 1.127 |
| **J** | 0.777 | 0.780 | 0.780 | 0.850 | 0.760 | 0.770 | 0.731 |

Denemeler sonrasında elde edilen sonuçlara göre Mahalanobis uzaklığına dayalı tahminsel yaklaşım yöntemi ve farklı sıralama algoritmalarının eşleştirilmesi ile oluşturulan modeller (mm1, mm2, mm3, mm4) genel anlamda kendilerini öne çıkarmış bulunmaktadırlar. Ayrıca sıralama algoritmaları içerisinden "use rate" algoritması dışındaki algoritmaların da bu veri kümesi için iyi sonuçlar verdikleri Tablo 7.7 den gözlemlenebilmektedir.

Bu veri kümesi için yapılan en iyi tahmin oranı 0.907 olarak "mm2" modeli (Mahalanobis uzaklığına dayalı tahminsel yaklaşım yöntemi ve "use rank" dizilim algoritması eşleşmesi) ile elde edilmiştir. Yapılan 30 denemenin ortalamaları ile hesaplanan diğer tablo değerleri içinde genel olarak bu model diğer modellerden ve Weka içerisinden alınan algoritmalardan daha iyi sonuç vermiştir.

Önerilen algoritma eşleşmeleri ile karşılaştırma yapmak üzere Weka'dan seçilen algoritmalar içerisinde NaiveBayes algoritmasının bu veri kümesi için Weka içerisinden alınan diğer sınıflandırma algoritmalarına göre daha iyi sonuç verdiği görülmektedir. Bu algoritma aynı zamanda 0.435 değeri ile en düşük "sh" değerini veren algoritmadır. Bu veri kümesi üzerinde yaptığımız denemeler ile elde ettiğimiz veriler, Mahalonobis uzaklığı ve özniteliklerin içerdiği farklı değer sayıları yardımı ile oluşturduğumuz matematiksel modellemenin, Bayes tabanlı bir algoritma ve örnek tabanlı bir algoritma olan Kstar algoritması ile birbirlerine çok yakın sonuçlar verdiğini göstermektedir.

#### 7.3.1.3 Farklı algoritmaların için farklı eğitim ve test veri kümesi sayıları ile karşılaştırması

Veri kümesinin 4 farklı bölünmesi sonrasında elde edilen 4 farklı eğitim ve test kümeleri üzerinde yapılan denemeler sonrasında, Tablo 7.8' de görünen sonuçlar arasında önemli bir fark görülmemektedir. Bu durumda kaynakların kullanılmasından tasarruf amacı ile 96 eğitim düğümünün kullanılması sonuçlar bakımından bir fark yaratmadığı için tercih edilebilirdir. Ayrıca, kullanılan eğitim düğümlerinin sayısının, veri kümesinin tamamının sayısına oranı veya test veri kümesinin sayısına oranına

doğrudan bir bağ görülmektedir. Bu durumda 96 eğitim düğümü ile eğitilmiş olan matematiksel modelimiz, özniteliklerin değer aralıkları değişmediği sürece kullanılabilir değerdedir. Burada tercihin yüksek eğitim düğümünden yana kullanılması durumunda ise, model eğitilirken harcanacak süre dışında test düğümlerinin değerlendirilmesi süreleri arasında önemli bir fark ortaya çıkmayacaktır.

**Tablo 7.8 : balance-scale karşılaştırma tablosu.**

| | ei | du | tp | tn | ke | f | sh |
|---|---|---|---|---|---|---|---|
| **500 eğitim düğümü ve 125 test düğümü** | | | | | | | |
| **mm1** | 0.889 | 0.907 | 0.856 | 0.940 | 0.888 | 0.868 | 0.504 |
| **mm2** | 0.905 | 0.910 | 0.860 | 0.937 | 0.883 | 0.868 | 0.482 |
| **mm4** | 0.897 | 0.909 | 0.859 | 0.943 | 0.887 | 0.869 | 0.500 |
| **400 eğitim düğümü ve 225 test düğümü** | | | | | | | |
| **mm1** | 0.898 | 0.907 | 0.856 | 0.943 | 0.889 | 0.868 | 0.505 |
| **mm2** | 0.907 | 0.910 | 0.861 | 0.942 | 0.886 | 0.871 | 0.463 |
| **mm4** | 0.903 | 0.912 | 0.864 | 0.943 | 0.890 | 0.874 | 0.506 |
| **200 eğitim düğümü ve 425 test düğümü** | | | | | | | |
| **mm1** | 0.866 | 0.900 | 0.846 | 0.961 | 0.908 | 0.869 | 0.428 |
| **mm2** | 0.888 | 0.908 | 0.859 | 0.941 | 0.885 | 0.869 | 0.465 |
| **mm4** | 0.876 | 0.900 | 0.846 | 0.957 | 0.905 | 0.868 | 0.438 |
| **96 eğitim düğümü ve 529 test düğümü** | | | | | | | |
| **mm1** | 0.878 | 0.887 | 0.827 | 0.941 | 0.879 | 0.849 | 0.483 |
| **mm2** | 0.880 | 0.893 | 0.836 | 0.912 | 0.844 | 0.839 | 0.558 |
| **mm4** | 0.885 | 0.894 | 0.837 | 0.938 | 0.875 | 0.853 | 0.460 |

### 7.3.2 blood Veri Kümesi

Blood Transfusion Service Center veri kümesi sayısal tanımlı 4 adet öznitelik, bir sınıf belirten kategorik tanımlı öznitelik ve 748 örnekten oluşmaktadır. IHDMR yazılımının

bu veri kümesi için önerdiği endeks uzayı sınırları {4, 5 , 5 , 5 } ve endeks uzayı boyutu 500 dür. Bu sınırlar nedeniyle veri kümesi içerisinden 500 örnek eğitim veri kümesi olarak, geri kalan 248 örnek de test veri kümesi olarak rasgele seçilerek atanmışlardır.

#### 7.3.2.1 Öznitelikler

"blood" veri kümesinin öznitelikleri Tablo 7.9 da, sınıf belirteci ise Tablo 7.10 de gösterilmiştir.

**Tablo 7.9 : blood öznitelikleri**

| isim | Nitelik | Maksimum | Minimum | Ortalama | St. Sapma |
|---|---|---|---|---|---|
| R | sayısal | 74 | 0 | 9.51 | 8.09 |
| F | sayısal | 50 | 1 | 5.51 | 5.84 |
| M | sayısal | 12500 | 250 | 1378.68 | 1459.8 |
| T | sayısal | 98 | 2 | 34.28 | 24.377 |

Bu veri kümesinin sınıf belirteci içerisindeki örneklerin 2 sınıf altında toplandığını göstermektedir. Bu dağılım sınıflar arasında "1" sınıfı içerisindeki örneklerin toplam örnek sayısına oranı oldukça düşük kalmıştır. Test veri kümesi içerisine seçilecek "1" sınıfı içerisinde olan örnekler için tahmin oranının diğer sınıflara göre düşük olması durumu söz konusudur.

**Tablo 7.10 : blood sınıf belirteci**

| değer | sayı | ağırlık |
|---|---|---|
| 0 | 570 | 0.76 |
| 1 | 178 | 0.24 |

#### 7.3.3.2 Test sonuçları

Denemeler sonrasında elde edilen sonuçlara göre Değişen Varyans Oranları yöntemi eşleştirilmesi ile oluşturulan algoritmalar diğerlerine göre nispeten daha iyi sonuçlar

vermişlerdir. Dizilim algoritmaları içerisinde göze çarpar bir fark ortaya çıkmamıştır. Bu veri kümesi için oluşan en iyi tahmin oranı 0.843 ile "cv1" modeli (Değişen varyans oranları yöntemi ve class sıralama algoritması eşleşmesi) için elde edilmiştir.

**Tablo 7.11 : blood deneme sonuçları tablosu.**

| | ei | du | tp | tn | ke | f | sh |
|---|---|---|---|---|---|---|---|
| **mr1** | 0.799 | 0.766 | 0.766 | 0.245 | 0.731 | 0.668 | 0.486 |
| **mr2** | 0.791 | 0.748 | 0.748 | 0.263 | 0.652 | 0.671 | 1.181 |
| **mr3** | 0.823 | 0.703 | 0.703 | 0.336 | 0.657 | 0.675 | 0.480 |
| **mr4** | 0.799 | 0.764 | 0.764 | 0.250 | 0.745 | 0.666 | 0.490 |
| **e1** | 0.771 | 0.729 | 0.729 | 0.425 | 0.701 | 0.711 | 0.522 |
| **e2** | 0.763 | 0.725 | 0.725 | 0.414 | 0.700 | 0.710 | 0.525 |
| **e3** | 0.746 | 0.717 | 0.717 | 0.429 | 0.692 | 0.702 | 0.533 |
| **e4** | 0.775 | 0.732 | 0.732 | 0.446 | 0.712 | 0.720 | 0.518 |
| **mm1** | 0.811 | 0.764 | 0.764 | 0.406 | 0.728 | 0.731 | 0.487 |
| **mm2** | 0.803 | 0.757 | 0.757 | 0.408 | 0.721 | 0.725 | 0.493 |
| **mm3** | 0.771 | 0.725 | 0.725 | 0.469 | 0.712 | 0.718 | 0.531 |
| **mm4** | 0.807 | 0.755 | 0.755 | 0.425 | 0.722 | 0.729 | 0.495 |
| **cv1** | 0.843 | 0.760 | 0.760 | 0.295 | 0.708 | 0.686 | 0.490 |
| **cv2** | 0.811 | 0.770 | 0.770 | 0.300 | 0.725 | 0.701 | 0.480 |
| **cv3** | 0.787 | 0.743 | 0.743 | 0.305 | 0.674 | 0.686 | 0.507 |
| **cv4** | 0.807 | 0.770 | 0.770 | 0.287 | 0.721 | 0.696 | 0.481 |
| **NB** | 0.818 | 0.750 | 0.750 | 0.360 | 0.700 | 0.710 | 0.444 |
| **K** | 0.758 | 0.750 | 0.750 | 0.440 | 0.720 | 0.730 | 0.474 |
| **OR** | 0.822 | 0.760 | 0.760 | 0.270 | 0.700 | 0.680 | 0.491 |
| **J** | 0.802 | 0.770 | 0.770 | 0.480 | 0.740 | 0.750 | 0.453 |

Yapılan 30 denemenin ortalamaları ile hesaplanan diğer tablo değerleri içinde ise "mm1" modeli (Mahalanobis uzaklığına dayalı tahminsel yaklaşım yöntemi ve use rank dizilim algoritması eşleşmesi) diğerlerine göre daha iyi bir performans göstermiştir.

Önerilen modeller ile karşılaştırma yapmak üzere Weka'dan seçilen algoritmalar içerisinde J48 algoritmasının bu veri kümesi için diğerlerine göre daha iyi sonuç verdiği görülmektedir. Bu veri kümesi üzerinde yaptığımız denemeler ile elde ettiğimiz veriler, Varyans değişimleri, Euclid uzaklığı ve eğitim veri kümesi içerisindeki örneklerin dahil oldukları sınıfları kullanarak oluşturduğumuz matematiksel modellemenin, Weka'dan seçilen farklı özellikteki algoritmalar ile benzer sonuçlar verdiği gözlenmiştir.

#### 7.3.1.3 Farklı algoritmaların için farklı eğitim ve test veri kümesi sayıları ile karşılaştırması

Veri kümesinin 4 farklı bölünmesi sonrasında elde edilen 4 farklı eğitim ve test kümeleri üzerinde yapılan denemeler sonrasında 625, 500 ve 300 eğitim düğümü ile eğitilen modeller arasında Tablo 7.12' deki sonuçlara göre, önemli bir performans farkı gözlenmemiştir. 96 eğitim düğümü ve cv1 modeli ile şekillendirilen snıflandırma probleminin önemli bir fark göstermediği gözlenmiştir.

**Tablo 7.12 : blood karşılaştırma tablosu.**

| | ei | du | tp | tn | ke | f | sh |
|---|---|---|---|---|---|---|---|
| **625 eğitim düğümü ve 123 test düğümü** | | | | | | | |
| **mr4** | 0.797 | 0.770 | 0.770 | 0.232 | 0.652 | 0.672 | 0.486 |
| **mm1** | 0.814 | 0.758 | 0.758 | 0.243 | 0.575 | 0.654 | 0.490 |
| **cv1** | 0.797 | 0.761 | 0.761 | 0.306 | 0.717 | 0.690 | 0.489 |
| **500 eğitim düğümü ve 248 test düğümü** | | | | | | | |
| **mr4** | 0.799 | 0.764 | 0.764 | 0.250 | 0.745 | 0.666 | 0.490 |
| **mm1** | 0.811 | 0.764 | 0.764 | 0.406 | 0.728 | 0.731 | 0.487 |
| **cv1** | 0.843 | 0.760 | 0.760 | 0.295 | 0.708 | 0.686 | 0.490 |
| **300 eğitim düğümü ve 448 test düğümü** | | | | | | | |
| **mr4** | 0.782 | 0.762 | 0.762 | 0.252 | 0.715 | 0.666 | 0.490 |
| **mm1** | 0.764 | 0.746 | 0.746 | 0.431 | 0.714 | 0.723 | 0.505 |
| **cv1** | 0.777 | 0.758 | 0.758 | 0.277 | 0.695 | 0.677 | 0.493 |
| **96 eğitim düğümü ve 652 test düğümü** | | | | | | | |
| **mr4** | 0.769 | 0.675 | 0.675 | 0.325 | 0.637 | 0.653 | 0.557 |
| **mm1** | 0.750 | 0.717 | 0.717 | 0.404 | 0.688 | 0.699 | 0.532 |
| **cv1** | 0.777 | 0.752 | 0.752 | 0.296 | 0.683 | 0.689 | 0.498 |

96 eğitim düğümü ve mm1 modeli ile şekillendirilen sınıflandırma probleminin diğer eğitim düğümü sayıları ile oluşturulan problemlere göre düşük bir fark gösterdiği gözlenmektedir. 96 eğitim düğümü ve mr4 algoritması ile oluşturulan sınıflandırma problemi ise en iyi tahmin değeri (ei) ile diğer eğitim düğümü sayıları ile elde edilen sınıflandırma problemlerinin en iyi tahmin değerleri (ei) arasında önemli bir fark yoktur. Fakat 30 deneme ortalamaları ile elde edilen diğer tablo değerleri bakımından göze çarpar bir performans düşüşü görülmüştür. Bu düşüş yaklaşık %10 civarlarında olmuştur. Bu durumda kaynakların kullanılmasından tasarruf amacı ile 96 eğitim düğümünün kullanılması cv1 ve mm1 modelleri için olumlu bir tasarruf tedbiri olacaktır. mr4 modeli için ise 300 eğitim düğümünün kullanılması ile elde edilecek

model tercih edilebilirdir. “mr4” Modeli çok değişkenli regresyona dayanan bir yöntemdir. Bu bakımdan, seçilen 96 eğitim düğümünün belirlenmesi için rastgele seçilen 30 farklı örneklemin uygulanan “ordering” dizilim algoritması ile uyum sağlamadığı durumu göz önüne alınmalıdır.

**Tablo 7.13 : blood- mr. karşılaştırma tablosu.**

| **96 eğitim düğümü ve 652 test düğümü için** | | | | | | | |
|---|---|---|---|---|---|---|---|
| | **ei** | **du** | **tp** | **tn** | **ke** | **f** | **sh** |
| **mr1** | 0.764 | 0.717 | 0.717 | 0.283 | 0.634 | 0.660 | 0.529 |
| **mr2** | 0.743 | 0.716 | 0.716 | 0.325 | 0.658 | 0.677 | 0.790 |
| **mr3** | 0.770 | 0.713 | 0.713 | 0.280 | 0.635 | 0.662 | 0.929 |
| **mr4** | 0.769 | 0.675 | 0.675 | 0.325 | 0.637 | 0.653 | 0.557 |

Bu durumda Tablo 7.13 değerleri göz önüne alındığında mr1 modeli için uygulanan “class” dizilim algoritmasının daha iyi sonuçlar verdiği gözlenmektedir.

### 7.3.3 bupa Veri Kümesi

“BUPA Liver Disorders” veri kümesi sayısal tanımlı 6 adet öznitelik, bir sınıf belirten kategorik tanımlı öznitelik ve 345 örnekten oluşmaktadır. IHDMR yazılımının bu veri kümesi için önerdiği endeks uzayı sınırları {2, 2 , 2 , 3 , 3 , 3 } ve endeks uzayı boyutu 216 dır. Bu sınırlar dolayısı ile veri seti içerisinden 216 örnek eğitim veri kümesi olarak, geri kalan 129 örnek de test veri kümesi olarak rastgele seçilerek atanmışlardır.

#### 7.3.3.1 Öznitelikler

“bupa” Veri kümesinin öznitelikleri Tablo 7.14 de, sınıf belirteci ise Tablo 7.15 de gösterilmiştir.

**Tablo 7.14 : bupa öznitelik istatistikleri tablosu**

| isim | Nitelik | Maksimum | Minimum | Ortalama | St. Sapma |
|---|---|---|---|---|---|
| mvc | sayısal | 103 | 65 | 90.16 | 4.448 |
| alkphos | sayısal | 138 | 23 | 69.87 | 18.348 |
| sqpt | sayısal | 155 | 4 | 30.4 | 19.51 |
| sgot | sayısal | 82 | 5 | 24.64 | 10.065 |
| gammagt | sayısal | 297 | 5 | 38.29 | 39.254 |
| drink | sayısal | 20 | 0 | 3.455 | 3.338 |

Bu veri kümesinin sınıf belirteci içerisindeki örneklerin 2 sınıf altında toplandığını göstermektedir. Bu dağılım sınıflar arasındaki dağılım eşit olmamakla birlikte birbirine yakın değerlerdir.

**Tablo 7.15 : bupa sınıf belirteci**

| değer | sayı | ağırlık |
|---|---|---|
| 1 | 145 | 0.42 |
| 2 | 200 | 0.58 |

#### 7.3.3.2 Test sonuçları

Denemeler sonrasında elde edilen sonuçlara göre “class” dizilim algoritması eşleştirilmesi ile oluşturulan algoritmalar diğerlerine göre daha iyi sonuçlar vermişlerdir. Yeni endeks uzayı oluşturan algoritmalar arasında belirgin bir fark ortaya çıkmamıştır. Bu veri kümesi için yapılan en iyi tahmin oranı 0.752 oranı ile “mm1” modeli (Mahalanobis uzaklığına dayalı tahminsel yaklaşım yöntemi ve “class” dizilim algoritması eşleşmesi) ile elde edilmiştir.

Yapılan 30 denemenin ortalamaları ile hesaplanan diğer tablo değerleri içinde ise, “mr1” modeli (Çoklu regresyona dayalı tahminsel yaklaşım yöntemi ve “class” dizilim algoritması eşleşmesi) diğerlerine göre daha iyi bir performans göstermiştir.

**Tablo 7.16 : bupa deneme sonuçları tablosu.**

| | ei | du | tp | tn | ke | f | sh |
|---|---|---|---|---|---|---|---|
| **mr1** | 0.721 | 0.661 | 0.661 | 0.594 | 0.662 | 0.641 | 0.584 |
| **mr2** | 0.690 | 0.615 | 0.615 | 0.656 | 0.656 | 0.615 | 0.679 |
| **mr3** | 0.613 | 0.515 | 0.515 | 0.483 | 0.509 | 0.511 | 0.726 |
| **mr4** | 0.706 | 0.620 | 0.620 | 0.542 | 0.615 | 0.589 | 0.617 |
| **e1** | 0.675 | 0.618 | 0.618 | 0.595 | 0.617 | 0.617 | 0.619 |
| **e2** | 0.698 | 0.614 | 0.614 | 0.601 | 0.616 | 0.615 | 0.622 |
| **e3** | 0.675 | 0.596 | 0.596 | 0.571 | 0.595 | 0.596 | 0.637 |
| **e4** | 0.690 | 0.624 | 0.624 | 0.601 | 0.626 | 0.625 | 0.613 |
| **mm1** | 0.752 | 0.639 | 0.639 | 0.606 | 0.635 | 0.635 | 0.601 |
| **mm2** | 0.714 | 0.638 | 0.638 | 0.625 | 0.638 | 0.638 | 0.602 |
| **mm3** | 0.690 | 0.616 | 0.616 | 0.582 | 0.612 | 0.613 | 0.620 |
| **mm4** | 0.698 | 0.626 | 0.626 | 0.588 | 0.622 | 0.624 | 0.612 |
| **cv1** | 0.659 | 0.567 | 0.567 | 0.418 | 0.487 | 0.458 | 0.658 |
| **cv2** | 0.636 | 0.566 | 0.566 | 0.427 | 0.499 | 0.456 | 0.659 |
| **cv3** | 0.613 | 0.547 | 0.547 | 0.455 | 0.510 | 0.484 | 0.675 |
| **cv4** | 0.644 | 0.570 | 0.570 | 0.432 | 0.513 | 0.453 | 0.656 |
| **NB** | 0.643 | 0.550 | 0.550 | 0.590 | 0.590 | 0.540 | 0.636 |
| **K** | 0.596 | 0.650 | 0.640 | 0.630 | 0.650 | 0.640 | 0.622 |
| **OR** | 0.527 | 0.560 | 0.560 | 0.510 | 0.550 | 0.550 | 0.649 |
| **J** | 0.666 | 0.630 | 0.630 | 0.590 | 0.630 | 0.620 | 0.541 |

Yapılan 30 denemenin ortalamaları ile hesaplanan diğer tablo değerleri içinde ise “mr1” modeli (Çoklu regresyona dayalı tahminsel yaklaşım yöntemi ve “class” dizilim algoritması eşleşmesi) diğerlerine göre daha iyi bir performans göstermiştir.

Önerilen modeller ile karşılaştırma yapmak üzere Weka'dan seçilen algoritmalar içerisinde “K”(Kstar) algoritmasının bu veri kümesi için Weka içerisinden seçilen diğer algoritmalara göre daha iyi sonuç verdiği görülmektedir. Bu veri kümesi üzerinde yaptığımız denemeler ile elde ettiğimiz veriler, çoklu regresyon ve eğitim veri kümesi

içerisindeki örneklerin dahil oldukları sınıfları kullanarak oluşturduğumuz matematiksel modellemenin, Weka'dan seçilen farklı özellikteki algoritmalar ile benzer sonuçlar verdiği gözlenmiştir.

#### 7.3.1.3 Farklı algoritmaların için farklı eğitim ve test veri kümesi sayıları ile karşılaştırması

Veri kümesinin 4 farklı şekilde bölünmesi sonrasında elde edilen 4 farklı eğitim ve test kümeleri üzerinde yapılan denemeler sonrasında, Tablo 7.16' da görünen, elde edilen sonuçların eğitim için kullanılan düğüm sayısının artmasına orantılı olarak iyileştiği gözlenmiştir. 32 ve 288 düğüm sayıları arasındaki performans farkı %10 civarında olmuştur.

Bu bağlamda, eğitim için kullanılacak düğüm sayısının yüksek tutulması sonuçların üzerinde olumlu bir etki yaratmaktadır.

**Tablo 7.17 : bupa karşılaştırma tablosu**

| | ei | du | tp | tn | ke | f | sh |
|---|---|---|---|---|---|---|---|
| **288 eğitim düğümü ve 57 test düğümü** | | | | | | | |
| **mm2** | 0.755 | 0.674 | 0.674 | 0.655 | 0.675 | 0.675 | 0.570 |
| **mm1** | 0.72 | 0.646 | 0.646 | 0.615 | 0.642 | 0.642 | 0.594 |
| **mr1** | 0.808 | 0.695 | 0.695 | 0.627 | 0.695 | 0.681 | 0.557 |
| **216 eğitim düğümü ve 129 test düğümü** | | | | | | | |
| **mm2** | 0.714 | 0.638 | 0.638 | 0.625 | 0.638 | 0.638 | 0.602 |
| **mm1** | 0.752 | 0.639 | 0.639 | 0.606 | 0.635 | 0.635 | 0.601 |
| **mr1** | 0.721 | 0.661 | 0.661 | 0.594 | 0.662 | 0.641 | 0.584 |
| **96 eğitim düğümü ve 229 test düğümü** | | | | | | | |
| **mm2** | 0.659 | 0.612 | 0.612 | 0.589 | 0.61 | 0.611 | 0.623 |
| **mm1** | 0.679 | 0.602 | 0.602 | 0.570 | 0.598 | 0.599 | 0.631 |
| **mr1** | 0.723 | 0.650 | 0.650 | 0.595 | 0.644 | 0.638 | 0.594 |
| **32 eğitim düğümü ve 313 test düğümü** | | | | | | | |
| **mm2** | 0.604 | 0.561 | 0.561 | 0.515 | 0.551 | 0.553 | 0.663 |
| **mm1** | 0.604 | 0.562 | 0.562 | 0.522 | 0.554 | 0.556 | 0.662 |
| **mr1** | 0.710 | 0.608 | 0.608 | 0.579 | 0.605 | 0.606 | 0.650 |

### 7.3.4 diabetes Veri Kümesi

Pima Indians Diabetes Database veri kümesi sayısal tanımlı 8 adet öznitelik, bir sınıf belirten kategorik tanımlı öznitelik ve 768 örnekten oluşmaktadır. IHDMR yazılımının bu veri kümesi için önerdiği endeks uzayı sınırları {2, 2, 2, 2, 2, 2 , 3 , 3 } ve endeks uzayı boyutu 576 dır. Bu sınırlar dolayısı ile veri kümesi içerisinden 576 örnek eğitim veri kümesi olarak geri kalan 192 örnek de test veri kümesi olarak rastgele seçilerek atanmışlardır.

#### 7.3.4.1 Öznitelikler

“diabetes” veri kümesinin öznitelikleri Tablo 7.18 de, sınıf belirteci ise Tablo 7.19 de gösterilmiştir.

**Tablo 7.18 : diabetes öznitelik istatistikleri.**

| isim | Nitelik | Maksimum | Minimum | Ortalama | St. Sapma |
|---|---|---|---|---|---|
| preg | sayısal | 17 | 0 | 3.846 | 3.37 |
| plas | sayısal | 199 | 0 | 120.89 | 31.973 |
| pres | sayısal | 122 | 0 | 69.11 | 19.356 |
| skin | sayısal | 99 | 0 | 20.536 | 15.952 |
| insu | sayısal | 846 | 0 | 79.799 | 115.244 |
| mass | sayısal | 67.1 | 0 | 31.9926 | 7.88416 |
| pedi | sayısal | 2.42 | 0.078 | 0.472 | 0.331 |
| age | sayısal | 81 | 21 | 33.241 | 11.76 |

Bu veri kümesinin sınıf belirteci içerisindeki örneklerin 2 sınıf altında toplandığını göstermektedir. Bu dağılım sınıflar arasında “tested-negatif” sınıfı içerisindeki örneklerin toplam örnek sayısına oranı oldukça düşük kalmıştır. Test veri kümesi içerisine seçilecek “tested-negatif” sınıfı içerisinde olan örnekler için tahmin oranının diğer sınıflara göre düşük olması durumu söz konusudur.

**Tablo 7.19 : diabetes sınıf belirteci.**

| değer | sayı | ağırlık |
|---|---|---|
| tested-pozitif | 500 | 0.65 |
| tested-negatif | 268 | 0.35 |

#### 7.3.4.1 Test sonuçları

Denemeler sonrasında elde edilen ve Tablo 7.20 içerisinde görünen sonuçlara göre herhangi bir dizilim algoritması eşleştirilmesi ile oluşturulan algoritmalar diğerlerine göre daha iyi sonuçlar vermemişlerdir. Yeni endeks uzayı oluşturan algoritmalar arasında belirgin bir fark ortaya çıkmamıştır.

**Tablo 7.20 : diabetes deneme sonuçları tablosu.**

| | ei | du | tp | tn | ke | f | sh |
|---|---|---|---|---|---|---|---|
| **mr1** | 0.813 | 0.761 | 0.761 | 0.617 | 0.758 | 0.744 | 0.489 |
| **mr2** | 0.797 | 0.714 | 0.714 | 0.591 | 0.702 | 0.700 | 0.556 |
| **mr3** | 0.730 | 0.670 | 0.670 | 0.481 | 0.646 | 0.631 | 0.579 |
| **mr4** | 0.766 | 0.724 | 0.724 | 0.618 | 0.714 | 0.714 | 0.526 |
| **e1** | 0.719 | 0.663 | 0.663 | 0.597 | 0.663 | 0.663 | 0.581 |
| **e2** | 0.724 | 0.666 | 0.666 | 0.601 | 0.667 | 0.667 | 0.578 |
| **e3** | 0.714 | 0.659 | 0.659 | 0.536 | 0.644 | 0.649 | 0.584 |
| **e4** | 0.750 | 0.677 | 0.677 | 0.609 | 0.677 | 0.677 | 0.569 |
| **mm1** | 0.745 | 0.685 | 0.685 | 0.628 | 0.687 | 0.686 | 0.562 |
| **mm2** | 0.750 | 0.692 | 0.692 | 0.642 | 0.696 | 0.694 | 0.556 |
| **mm3** | 0.693 | 0.654 | 0.654 | 0.540 | 0.641 | 0.645 | 0.588 |
| **mm4** | 0.787 | 0.707 | 0.707 | 0.681 | 0.718 | 0.711 | 0.541 |
| **cv1** | 0.714 | 0.714 | 0.714 | 0.524 | 0.71 0 | 0.677 | 0.535 |
| **cv2** | 0.792 | 0.722 | 0.722 | 0.531 | 0.717 | 0.689 | 0.527 |
| **cv3** | 0.709 | 0.660 | 0.660 | 0.490 | 0.633 | 0.629 | 0.584 |
| **cv4** | 0.792 | 0.707 | 0.707 | 0.508 | 0.702 | 0.666 | 0.541 |
| **NB** | 0.770 | 0.760 | 0.760 | 0.680 | 0.750 | 0.750 | 0.464 |
| **K** | 0.755 | 0.700 | 0.700 | 0.590 | 0.690 | 0.690 | 0.540 |
| **OR** | 0.765 | 0.730 | 0.730 | 0.610 | 0.730 | 0.720 | 0.500 |
| **J** | 0.739 | 0.730 | 0.730 | 0.670 | 0.730 | 0.730 | 0.492 |

Bu veri kümesi için yapılan en iyi tahmin oranı 0.813 oranı ile “mr1” modeli (Çoklu regresyona dayalı tahminsel yaklaşım yöntemi ve “class” dizilim algoritması eşleşmesi)

için elde edilmiştir. Yapılan 30 denemenin ortalamaları ile hesaplanan diğer tablo değerleri içinde ise “mr1” modeli (Çoklu regresyona dayalı tahminsel yaklaşım yöntemi ve “class” dizilim algoritması eşleşmesi) diğerlerine göre daha iyi bir performans göstermiştir.

Önerilen modeller ile karşılaştırma yapmak üzere Weka'dan seçilen algoritmalar içerisinde NaiveBayes algoritmasının bu veri kümesi için diğerlerine göre daha iyi sonuç verdiği görülmektedir.

Elde edilen en küçük “sh” değeri olan 0.464, yine bu algoritmanın çıktısıdır. Bu veri kümesi üzerinde yaptığımız denemeler ile elde ettiğimiz veriler, çoklu regresyon ve eğitim veri kümesi içerisindeki örneklerin dahil oldukları sınıfları kullanarak oluşturduğumuz matematiksel modellemenin, Weka'dan seçilen farklı özellikteki algoritmalardan NaiveBayes dışındakilerden daha iyi sonuçlar verdiği ve NaiveBayes ile benzer sonuçlar verdiği gözlenmiştir.

#### 7.3.1.3 Farklı algoritmaların için farklı eğitim ve test veri kümesi sayıları ile karşılaştırması

Veri kümesinin 4 farklı şekilde bölünmesi sonrasında elde edilen 4 farklı eğitim ve test kümeleri üzerinde yapılan denemeler sonrasında, elde edilen sonuçların eğitim düğümü sayısına bağlı olduğu fakat bu bağlılığın sonuçlar üzerindeki etkilerinin maliyetler bakımından göz ardı edilebilecek düzeyde olduğu gözlenmiştir.

Bu denemeler sonrasında cv2 modelinin eğitim düğümü sayısından en çok etkilenen model olduğu gözlenmiştir.

**Tablo 7.21 : diabetes karşılaştırma tablosu.**

| | ei | du | tp | tn | ke | f | sh |
|---|---|---|---|---|---|---|---|
| **640 eğitim düğümü ve 128 test düğümü** | | | | | | | |
| **mr1** | 0.813 | 0.757 | 0.757 | 0.611 | 0.753 | 0.739 | 0.493 |
| **cv2** | 0.758 | 0.706 | 0.706 | 0.536 | 0.706 | 0.669 | 0.541 |
| **mr4** | 0.797 | 0.715 | 0.715 | 0.603 | 0.704 | 0.706 | 0.534 |
| **576 eğitim düğümü ve 192 test düğümü** | | | | | | | |
| **mr1** | 0.813 | 0.761 | 0.761 | 0.617 | 0.758 | 0.744 | 0.489 |
| **cv2** | 0.792 | 0.722 | 0.722 | 0.531 | 0.717 | 0.689 | 0.527 |
| **mr4** | 0.766 | 0.724 | 0.724 | 0.618 | 0.714 | 0.714 | 0.526 |
| **256 eğitim düğümü ve 512 test düğümü** | | | | | | | |
| **mr1** | 0.797 | 0.742 | 0.742 | 0.596 | 0.739 | 0.721 | 0.508 |
| **cv2** | 0.752 | 0.713 | 0.713 | 0.542 | 0.704 | 0.684 | 0.536 |
| **mr4** | 0.756 | 0.728 | 0.728 | 0.625 | 0.719 | 0.72 | 0.522 |
| **96 eğitim düğümü ve 672 test düğümü** | | | | | | | |
| **mr1** | 0.764 | 0.736 | 0.736 | 0.594 | 0.729 | 0.717 | 0.514 |
| **cv2** | 0.716 | 0.687 | 0.687 | 0.513 | 0.668 | 0.656 | 0.560 |
| **mr4** | 0.745 | 0.715 | 0.715 | 0.639 | 0.71 | 0.712 | 0.544 |

Tablo 7.22 de gözlendiği üzere Değişen Varyans Oranları yöntemi (cv) ile birlikte kullanılan sıralama algoritmaları 96 eğitim düğümü için oluşturulan model üzerinde farklı etkiler oluşturmamaktadır.

**Tablo 7.22 : diabetes-cv. karşılaştırma tablosu.**

| **96 eğitim düğümü ve 672 test düğümü için** | | | | | | | |
|---|---|---|---|---|---|---|---|
| | **ei** | **du** | **tp** | **tn** | **ke** | **f** | **sh** |
| **cv1** | 0.724 | 0.684 | 0.684 | 0.495 | 0.665 | 0.644 | 0.563 |
| **cv2** | 0.716 | 0.687 | 0.687 | 0.513 | 0.668 | 0.656 | 0.560 |
| **cv3** | 0.716 | 0.664 | 0.664 | 0.551 | 0.651 | 0.655 | 0.580 |
| **cv4** | 0.722 | 0.685 | 0.685 | 0.500 | 0.666 | 0.648 | 0.562 |

### 7.3.5 iris Veri Kümesi

Iris Plants Database veri kümesi sayısal tanımlı 4 adet öznitelik, bir sınıf belirten kategorik tanımlı öznitelik ve 150 örnekten oluşmaktadır. IHDMR yazılımının bu veri kümesi için önerdiği indeks uzayı sınırları {3 , 3, 3, 4} ve endeks uzayı boyutu 108 dır. Bu sınırlar dolayısı ile veri kümesi içerisinden 108 örnek eğitim veri kümesi olarak geri kalan 42 örnek de test veri kümesi olarak rastgele seçilerek atanmışlardır.

#### 7.3.5.1 Öznitelikler

"iris" veri kümesinin öznitelikleri Tablo 7.23 de, sınıf belirteci ise Tablo 7.24 de gösterilmiştir.

**Tablo 7.23 : iris öznitelik istatistikleri.**

| **isim** | **Nitelik** | **Maksimum** | **Minimum** | **Ortalama** | **St. Sapma** |
|---|---|---|---|---|---|
| sepallength | sayısal | 7.9 | 4.3 | 5.843 | 0.8430 |
| sepalwidth | sayısal | 4.4 | 2 | 3.054 | 0.4336 |
| pedallength | sayısal | 6.9 | 1 | 3.7587 | 1.7644 |
| pedalwidth | sayısal | 2.5 | 0.1 | 1.1987 | 0.7631 |

Bu veri kümesinin sınıf belirteci içerisindeki örneklerin 3 sınıf altında toplandığını göstermektedir. Bu dağılım sınıflar arasında eşit olarak gerçekleşmiştir. Bu sebeple eğitim süreçleri içerisinde eğitim düğümünden kaynalı bir fark olmayacaktır.

**Tablo 7.24 : iris sınıf belirteci.**

| değer | sayı | ağırlık |
|---|---|---|
| iris-setosa | 50 | 0.3333 |
| iris-versicolor | 50 | 0.3333 |
| iris-virginica | 50 | 0.3333 |

#### 7.3.5.2 Test sonuçları

Denemeler sonrasında elde edilen sonuçlara göre herhangi bir sıralama algoritması eşleştirilmesi ile oluşturulan model, diğer sıralama algoritmaları eşleştirmesi ile oluşturulan modele göre daha iyi sonuçlar vermemiştir. Yeni endeks uzayı oluşturan algoritmalar arasında belirgin bir fark ortaya çıkmamıştır.

Bu veri kümesi için yapılan en iyi tahmin oranı olan %100 oranını “mr3” dışındaki bütün modeller yakalamıştır. Bunların arasından en düşük “sh” değerine sahip “mm2” modeli (Mahalanobis uzaklığına dayalı tahminsel yaklaşım yöntemi ve “use rank” dizilim algoritması eşleşmesi) kendini diğer modellere göre öne çıkarmıştır.

Yapılan 30 denemenin ortalamaları ile hesaplanan diğer tablo değerleri içinde bu algoritma diğerlerine göre daha iyi bir performans göstermiştir. Elde edilen en küçük “sh” değeri olan 0.15, yine bu algoritmanın çıktısıdır.

**Tablo 7.25 : iris deneme sonuçları tablosu.**

| | ei | du | tp | tn | ke | f | sh |
|---|---|---|---|---|---|---|---|
| **mr1** | 1 | 0.974 | 0.958 | 0.979 | 0.959 | 0.958 | 0.182 |
| **mr2** | 1 | 0.975 | 0.961 | 0.981 | 0.961 | 0.961 | 0.183 |
| **mr3** | 0.905 | 0.869 | 0.800 | 0.901 | 0.799 | 0.798 | 0.444 |
| **mr4** | 1 | 0.971 | 0.954 | 0.977 | 0.954 | 0.954 | 0.196 |
| **e1** | 1 | 0.967 | 0.948 | 0.975 | 0.948 | 0.948 | 0.209 |
| **e2** | 1 | 0.971 | 0.954 | 0.978 | 0.955 | 0.954 | 0.193 |
| **e3** | 1 | 0.971 | 0.954 | 0.977 | 0.955 | 0.954 | 0.200 |
| **e4** | 1 | 0.973 | 0.957 | 0.979 | 0.957 | 0.957 | 0.189 |
| **mm1** | 1 | 0.977 | 0.963 | 0.981 | 0.963 | 0.963 | 0.172 |
| **mm2** | 1 | 0.981 | 0.970 | 0.985 | 0.970 | 0.970 | 0.150 |
| **mm3** | 1 | 0.981 | 0.970 | 0.985 | 0.970 | 0.969 | 0.155 |
| **mm4** | 1 | 0.979 | 0.967 | 0.985 | 0.967 | 0.967 | 0.170 |
| **cv1** | 1 | 0.967 | 0.947 | 0.974 | 0.948 | 0.947 | 0.212 |
| **cv2** | 1 | 0.966 | 0.947 | 0.972 | 0.947 | 0.947 | 0.221 |
| **cv3** | 1 | 0.963 | 0.942 | 0.971 | 0.942 | 0.942 | 0.227 |
| **cv4** | 1 | 0.968 | 0.949 | 0.975 | 0.949 | 0.949 | 0.207 |
| **NB** | 0.952 | 0.960 | 0.960 | 0.980 | 0.960 | 0.960 | 0.200 |
| **K** | 0.952 | 0.940 | 0.940 | 0.970 | 0.940 | 0.940 | 0.244 |
| **OR** | 0.928 | 0.940 | 0.940 | 0.970 | 0.940 | 0.940 | 0.282 |
| **J** | 0.952 | 0.940 | 0.940 | 0.970 | 0.94 | 0.94 | 0.282 |

Önerilen modeller ile karşılaştırma yapmak üzere Weka'dan seçilen algoritmalar içerisinde NaiveBayes algoritmasının bu veri kümesi için Weka içerisinden seçilen diğer sınıflandırma algoritmalarına göre daha iyi sonuç verdiği görülmektedir. Bu veri kümesi üzerinde yaptığımız denemeler ile elde ettiğimiz veriler, oluşturduğumuz modellerin genellikle başarılı sonuçlar verdiğini göstermektedir. Oluşturduğumuz matematiksel modellemelerin birçoğunun, Weka'dan seçilen farklı özellikteki algoritmalardan daha iyi sonuçlar verdiği gözlenmiştir.

#### 7.3.1.3 Farklı algoritmaların için farklı eğitim ve test veri kümesi sayıları ile karşılaştırması

**Tablo 7.26 : iris karşılaştırma tablosu.**

| | ei | du | tp | tn | ke | f | sh |
|---|---|---|---|---|---|---|---|
| **120 eğitim düğümü ve 30 test düğümü** | | | | | | | |
| **mm2** | 1 | 0.974 | 0.959 | 0.981 | 0.96 | 0.959 | 0.185 |
| **mm3** | 1 | 0.981 | 0.971 | 0.986 | 0.971 | 0.970 | 0.131 |
| **mr2** | 0.934 | 0.842 | 0.761 | 0.878 | 0.806 | 0.760 | 0.503 |
| **108 eğitim düğümü ve 42 test düğümü** | | | | | | | |
| **mm2** | 1 | 0.981 | 0.970 | 0.985 | 0.970 | 0.97 0 | 0.150 |
| **mm3** | 1 | 0.981 | 0.970 | 0.985 | 0.970 | 0.969 | 0.155 |
| **mr2** | 1 | 0.975 | 0.961 | 0.981 | 0.961 | 0.961 | 0.183 |
| **54 eğitim düğümü ve 96 test düğümü** | | | | | | | |
| **mm2** | 0.98 | 0.972 | 0.956 | 0.979 | 0.957 | 0.956 | 0.207 |
| **mm3** | 1 | 0.971 | 0.954 | 0.977 | 0.955 | 0.954 | 0.206 |
| **mr2** | 0.99 | 0.969 | 0.951 | 0.975 | 0.951 | 0.951 | 0.214 |
| **24 eğitim düğümü ve 126 test düğümü** | | | | | | | |
| **mm2** | 0.969 | 0.890 | 0.831 | 0.916 | 0.831 | 0.830 | 0.426 |
| **mm3** | 0.985 | 0.914 | 0.867 | 0.934 | 0.867 | 0.865 | 0.354 |
| **mr2** | 0.969 | 0.963 | 0.941 | 0.971 | 0.941 | 0.941 | 0.241 |

Veri kümesinin 4 farklı şekilde bölünmesi sonrasında elde edilen 4 farklı eğitim ve test kümeleri üzerinde yapılan denemeler sonrasında, elde edilen sonuçların eğitim için kullanılan düğüm sayısının artmasına orantılı olarak 120 eğitim düğümü sayısına kadar iyileştiği gözlenmiştir. 120 eğitim düğümü sayısı ile yapılan modellemelerde, mm2 modeli için 0.035 standart hata artışı görülmektedir. mr2 modeli için ise belirgin bir performans kaybı görülmektedir.

Çok değişkenli regresyona dayalı “mr” algoritmasının 120 eğitim düğümü sayısı için diğer dizilim algoritmaları ile birlikte ulaştığı performanslara bakılacak olursa;

**Tablo 7.27 : iris-mr. karşılaştırma tablosu.**

| **120 eğitim düğümü ve 30 test düğümü** | | | | | | | |
|---|---|---|---|---|---|---|---|
| | **ei** | **du** | **tp** | **tn** | **ke** | **f** | **sh** |
| **mr1** | 1 | 0.968 | 0.949 | 0.974 | 0.951 | 0.949 | 0.193 |
| **mr2** | 0.934 | 0.842 | 0.761 | 0.878 | 0.806 | 0.760 | 0.503 |
| **mr3** | 0.901 | 0.862 | 0.790 | 0.900 | 0.793 | 0.786 | 0.507 |
| **mr4** | 1 | 0.951 | 0.924 | 0.959 | 0.928 | 0.923 | 0.278 |

aynı sorunun “rate” dizilim algoritması ie birlikte oluşturulan model (mr3) için de ortaya çıktığı görülmektedir.

Elde edilen sonuçlar doğrultusunda mr2 modeli için 24 eğitim düğümü, mm2 ve mm3 modelleri için de 54 eğitim düğümünün eğitim için yeterli olduğu görülmektedir.

### 7.3.6 tae Veri Kümesi

Teaching Assistant Evaluation veri kümesi sayısal tanımlı 1 adet öznitelik, 4 adet kategorik öznitelik, bir sınıf belirten kategorik tanımlı öznitelik ve 151 örnekten oluşmaktadır. IHDMR yazılımının bu veri kümesi için önerdiği endeks uzayı sınırları {2, 2 ,3, 3, 3} ve endeks uzayı boyutu 108 dır. Bu sınırlar dolayısı ile veri kümesi içerisinden 108 örnek eğitim veri kümesi olarak geri kalan 42 örnek de test veri kümesi olarak rastgele seçilerek atanmışlardır.

#### 7.3.6.1 Öznitelikler

“tae” veri kümesinin öznitelikleri Tablo 7.28 de, sınıf belirteci ise Tablo 7.29 de gösterilmiştir.

**Tablo 7.28 : tae öznitelik istatistikleri**

| isim | Nitelik | Maksimum | Minimum | Ortalama | St. Sapma | farklı |
|---|---|---|---|---|---|---|
| nativeOrNot | kategorik | 2 | 1 | - | - | 2 |
| Course_instructor | kategorik | 25 | 1 | - | - | 25 |
| Course | kategorik | 26 | 1 | - | - | 26 |
| semester | kategorik | 2 | 1 | - | - | 2 |
| class-size | sayısal | 66 | 3 | 27.8676 | 12.8928 | - |

Bu veri kümesinin sınıf belirteci içerisindeki örneklerin 3 sınıf altında toplandığını göstermektedir. Bu dağılım sınıflar arasında eşit olarak gerçekleşmiştir. Bu sebeple eğitim süreçleri içersinde eğitim düğümünden kaynalı bir fark olmayacaktır.

**Tablo 7.29 : tae sınıf belirteci.**

| değer | sayı | ağırlık |
|---|---|---|
| 1 | 49 | 0.325 |
| 2 | 50 | 0.331 |
| 3 | 52 | 0.344 |

#### 7.3.6.2 Test sonuçları

Denemeler sonrasında elde edilen sonuçlara göre herhangi bir sıralama algoritması eşleştirilmesi ile oluşturulan modeller diğer sıralama algoritmaları ile oluşturulan modellere göre daha iyi sonuçlar vermemiştir. Yeni endeks uzayı oluşturan algoritmalar arasında Mahalanobis uzaklığına dayalı tahminsel yaklaşım yönteminin diğerlerine göre daha iyi bir performans ortaya koyduğu gözlenmektedir. Bu veri kümesi için yapılan en iyi tahmin oranı 0.721 oranı ile “mm4” modeli (Mahalanobis uzaklığına dayalı tahminsel yaklaşım yöntemi ve “ordering” dizilim algoritması eşleşmesi) ile elde edilmiştir. Yapılan 30 denemenin ortalamaları ile hesaplanan diğer tablo değerlere göre birkaç algoritma diğerlerine göre daha iyi sonuçlar vermişlerdir.

Bunlar Kstar, “mm2” (Mahalanobis uzaklığına dayalı tahminsel yaklaşım yöntemi ve “use rank” dizilim algoritması eşleşmesidir.) ve “e1” (Euckid uzaklığına dayalı tahminsel yaklaşım yöntemi ve “class” dizilim algoritması eşleşmesi) modelleridir. Elde edilen en küçük “sh” değeri olan 0.835 ise, “mm2” modelinin çıktısıdır.

**Tablo 7.30 : tae deneme sonuçları tablosu.**

| | ei | du | tp | tn | ke | f | sh |
|---|---|---|---|---|---|---|---|
| **mr1** | 0.396 | 0.471 | 0.303 | 0.690 | 0.386 | 0.170 | 0.850 |
| **mr2** | 0.442 | 0.488 | 0.321 | 0.664 | 0.365 | 0.294 | 0.904 |
| **mr3** | 0.419 | 0.487 | 0.320 | 0.67 | 0.309 | 0.236 | 0.952 |
| **mr4** | 0.466 | 0.528 | 0.363 | 0.664 | 0.393 | 0.261 | 0.838 |
| **e1** | 0.675 | 0.684 | 0.548 | 0.774 | 0.548 | 0.548 | 0.988 |
| **e2** | 0.442 | 0.488 | 0.321 | 0.664 | 0.365 | 0.294 | 0.904 |
| **e3** | 0.628 | 0.589 | 0.432 | 0.716 | 0.439 | 0.425 | 1.008 |
| **e4** | 0.652 | 0.669 | 0.529 | 0.766 | 0.533 | 0.529 | 1.012 |
| **mm1** | 0.628 | 0.646 | 0.500 | 0.756 | 0.527 | 0.501 | 0.880 |
| **mm2** | 0.698 | 0.676 | 0.538 | 0.773 | 0.557 | 0.539 | 0.835 |
| **mm3** | 0.559 | 0.583 | 0.425 | 0.718 | 0.484 | 0.413 | 0.887 |
| **mm4** | 0.721 | 0.644 | 0.497 | 0.748 | 0.511 | 0.499 | 0.903 |
| **cv1** | 0.442 | 0.501 | 0.335 | 0.656 | 0.353 | 0.248 | 1.163 |
| **cv2** | 0.489 | 0.515 | 0.349 | 0.658 | 0.322 | 0.257 | 1.167 |
| **cv3** | 0.466 | 0.511 | 0.345 | 0.662 | 0.341 | 0.302 | 1.086 |
| **cv4** | 0.489 | 0.513 | 0.347 | 0.655 | 0.336 | 0.259 | 1.184 |
| **NB** | 0.674 | 0.480 | 0.480 | 0.740 | 0.480 | 0.480 | 0.948 |
| **K** | 0.720 | 0.570 | 0.570 | 0.780 | 0.580 | 0.560 | 0.938 |
| **OR** | 0.628 | 0.430 | 0.430 | 0.720 | 0.430 | 0.420 | 1.067 |
| **J** | 0.604 | 0.460 | 0.460 | 0.730 | 0.460 | 0.440 | 1.019 |

Önerilen algoritma eşleşmeleri ile karşılaştırma yapmak üzere Weka'dan seçilen algoritmalar içerisinde Kstar algoritmasının bu veri kümesi için Weka içerisinden

seçilen diğer sınıflandırma algoritmalarına göre daha iyi sonuç verdiği görülmektedir. Bu veri kümesi üzerinde yaptığımız denemeler ile elde ettiğimiz veriler, oluşturduğumuz modeller içerisinden, Mahalonobis uzaklığı ve özniteliklerin içerdiği farklı değerlerden yararlanarak oluşturduğumuz model ve Euclid uzaklığı ve eğitim veri seti içerisindeki örneklerin sınıflarından yararlanarak oluşturduğumuz modelin Weka'dan seçilen farklı özellikteki algoritmalar ile benzer sonuçlar verdiği gözlenmiştir.

#### 7.3.1.3 Farklı algoritmaların için farklı eğitim ve test veri kümesi sayıları ile karşılaştırması

Veri kümesinin 4 farklı şekilde bölünmesi sonrasında elde edilen 4 farklı eğitim ve test kümeleri üzerinde yapılan denemeler sonrasında, 120 eğitim düğümü sayısına kadar olan eğitim düğümü sayısı artışlarının sonuçlar üzerinde olumlu etkisi olduğu gözlenmiştir. 120 eğitim düğümü sayısı için oluşturulan modellemelerde elde edilen 30 deneme ortalamaları olan sonuçlarda bir fark görülmemiştir. Bu durumda bu veri kümesi için 108 eğitim düğümü ile eğitilmiş modellerin yeterli bir performans gösterdikleri söylenebilirdir.

**Tablo 7.31 : tae karşılaştıma tablosu.**

| | ei | du | tp | tn | ke | f | sh |
|---|---|---|---|---|---|---|---|
| **120 eğitim düğümü ve 31 test düğümü** | | | | | | | |
| **e1** | 0.678 | 0.680 | 0.542 | 0.772 | 0.543 | 0.542 | 0.966 |
| **mm4** | 0.613 | 0.638 | 0.491 | 0.753 | 0.523 | 0.491 | 0.932 |
| **mm2** | 0.646 | 0.687 | 0.551 | 0.776 | 0.569 | 0.554 | 0.825 |
| **108 eğitim düğümü ve 43 test düğümü** | | | | | | | |
| **e1** | 0.675 | 0.684 | 0.548 | 0.774 | 0.548 | 0.548 | 0.988 |
| **mm4** | 0.721 | 0.644 | 0.497 | 0.748 | 0.511 | 0.499 | 0.903 |
| **mm2** | 0.698 | 0.676 | 0.538 | 0.773 | 0.557 | 0.539 | 0.835 |
| **54 eğitim düğümü ve 97 test düğümü** | | | | | | | |
| **e1** | 0.476 | 0.587 | 0.430 | 0.715 | 0.436 | 0.431 | 1.130 |
| **mm4** | 0.505 | 0.552 | 0.389 | 0.699 | 0.405 | 0.377 | 1.105 |
| **mm2** | 0.544 | 0.534 | 0.369 | 0.687 | 0.416 | 0.297 | 1.204 |
| **24 eğitim düğümü ve 127 test düğümü** | | | | | | | |
| **e1** | 0.473 | 0.537 | 0.373 | 0.687 | 0.374 | 0.372 | 1.141 |
| **mm4** | 0.426 | 0.513 | 0.347 | 0.681 | 0.358 | 0.284 | 1.220 |
| **mm2** | 0.481 | 0.514 | 0.349 | 0.688 | 0.444 | 0.243 | 1.253 |

### 7.3.7 vovel Veri Kümesi

Deterding Vowel Recognition Data veri kümesi sayısal tanımlı 10 adet öznitelik, 3 adet kategorik tanımlı öznitelik, bir sınıf belirten kategorik tanımlı öznitelik ve 990 örnekten oluşmaktadır. IHDMR yazılımının bu veri seti için önerdiği endeks uzayı sınırları {1, 1, 1, 1, 2 , 2, 2, 2, 2, 2, 2, 2, 2} ve endeks uzayı boyutu 512 dır. Bu sınırlar dolayısı ile veri kümesi içerisinden 512 örnek eğitim veri kümesi olarak geri kalan 478 örnek de test veri kümesi olarak rastgele seçilerek atanmışlardır.

#### 7.3.7.1 Öznitelikler

"vovel" veri kümesinin öznitelikleri Tablo 7.32 de, sınıf belirteci ise Tablo 7.33 de gösterilmiştir.

**Tablo 7.32 : vovel öznitelik istatistikleri.**

| isim | Nitelik | Maksimum | Minimum | Ortalama | St. Sapma | farklı |
|---|---|---|---|---|---|---|
| Train or Test | kategorik | 2 | 1 | - | - | 2 |
| Speaker Number | kategorik | 15 | 1 | - | - | 15 |
| Sex | kategorik | 2 | 1 | - | - | 2 |
| Feature 0 | kategorik | -0.941 | -5.211 | -3.2037 | 0.86899 | - |
| Feature 1 | sayısal | 5.074 | -1.274 | 1.88176 | 1.175272 | - |
| Feature 2 | sayısal | 1.431 | -2.487 | -0.5078 | 0.71194 | - |
| Feature 3 | sayısal | 2.377 | -1.409 | 0.51548 | 0.75926 | - |
| Feature 4 | sayısal | 1.831 | -2.127 | -0.30566 | 0.664602 | - |
| Feature 5 | sayısal | 2.327 | -0.836 | 0.63024 | 0.60387 | - |
| Feature 6 | sayısal | 1.403 | -1.537 | -0.00436 | 0.46193 | - |
| Feature 7 | sayısal | 2.039 | -1.293 | 0.33656 | 0.5733 | - |
| Feature 8 | sayısal | 1.309 | -1.613 | -0.30298 | 0.57016 | - |
| Feature 9 | sayısal | 1.396 | -1.68 | -0.07134 | 0.60398 | - |

Bu veri kümesinin sınıf belirteci içerisindeki örneklerin 11 sınıf altında toplandığını göstermektedir. Bu dağılım sınıflar arasında eşit olarak gerçekleşmiştir. Bu sebeple eğitim süreçleri içerisinde eğitim düğümünden kaynalı bir fark olmayacaktır.

**Tablo 7.33 : vovel sınıf belirteci.**

| değer | sayı | ağırlık |
|---|---|---|
| hid | 90 | 0.0909 |
| hId | 90 | 0.0909 |
| hEd | 90 | 0.0909 |
| hAd | 90 | 0.0909 |
| hYd | 90 | 0.0909 |
| had | 90 | 0.0909 |
| hOd | 90 | 0.0909 |
| hod | 90 | 0.0909 |
| hUd | 90 | 0.0909 |
| hud | 90 | 0.0909 |
| hed | 90 | 0.0909 |

#### 7.3.7.2 Test sonuçları

Denemeler sonrasında elde edilen ve Tablo 7.34' de görünen sonuçlara göre “use rank” sıralama algoritması eşleştirilmesi ile oluşturulan modeller diğer sıralama algoritmaları ile oluşturulan modellerlere göre daha iyi sonuçlar vermişlerdir. Yeni endeks uzayı oluşturan algoritmalar arasında Mahalanobis uzaklığına dayalı tahminsel yaklaşım yönteminin ve Euclid uzaklığına dayalı yöntemin diğerlerine göre daha iyi bir performans ortaya koyduğu gözlenmektedir.

Bu veri kümesi için yapılan en iyi tahmin oranı 0.962 oranı ile Kstar algoritması ile elde edilmiştir. Yapılan 30 denemenin ortalamaları ile hesaplanan diğer tablo değerlere göre Kstar algoritması diğerlerine göre daha iyi bir performans göstermiştir. Elde edilen en küçük “sh” değeri olan 0.383 değeri de bu algoritmasının çıktısıdır.

Bu veri kümesi üzerinde yaptığımız denemeler ile elde ettiğimiz veriler, oluşturduğumuz modeller içerisinden, Euclid uzaklığı ve eğitim veri seti içerisindeki örneklerin özniteliklerinin içerdiği farklı değerlerden yararlanarak oluşturduğumuz

modelin (e2), Weka'dan seçilen Kstar algoritmasına yakın bir performans gösterdiği gözlenmiştir. E2 modelinin yaptığı en iyi tahmin 0.923' tür. Önerilen modeller ile karşılaştırma yapmak üzere Weka'dan seçilen algoritmalar içerisinde Kstar algoritmasının bu veri kümesi için Weka içerisinden seçilen diğer sınıflandırma algoritmalarına göre daha iyi sonuç verdiği görülmektedir.

**Tablo 7.34 : vovel deneme sonuçları.**

| | **ei** | **du** | **tp** | **tn** | **ke** | **f** | **sh** |
|---|---|---|---|---|---|---|---|
| **mr1** | 0.218 | 0.601 | 0.171 | 0.918 | 0.226 | 0.168 | 2.621 |
| **mr2** | 0.191 | 0.562 | 0.136 | 0.915 | 0.226 | 0.137 | 2.729 |
| **mr3** | 0.149 | 0.522 | 0.106 | 0.911 | 0.117 | 0.097 | 3.607 |
| **mr4** | 0.220 | 0.608 | 0.179 | 0.919 | 0.237 | 0.173 | 2.585 |
| **e1** | 0.888 | 0.944 | 0.819 | 0.982 | 0.821 | 0.818 | 0.703 |
| **e2** | 0.923 | 0.964 | 0.879 | 0.988 | 0.880 | 0.879 | 0.659 |
| **e3** | 0.176 | 0.554 | 0.130 | 0.914 | 0.123 | 0.12 0 | 3.170 |
| **e4** | 0.894 | 0.947 | 0.828 | 0.983 | 0.830 | 0.828 | 0.671 |
| **mm1** | 0.804 | 0.920 | 0.749 | 0.975 | 0.758 | 0.750 | 0.824 |
| **mm2** | 0.860 | 0.941 | 0.808 | 0.981 | 0.816 | 0.809 | 0.824 |
| **mm3** | 0.178 | 0.553 | 0.129 | 0.913 | 0.145 | 0.120 | 3.082 |
| **mm4** | 0.812 | 0.922 | 0.754 | 0.976 | 0.761 | 0.753 | 0.822 |
| **cv1** | 0.507 | 0.773 | 0.409 | 0.942 | 0.452 | 0.399 | 2.672 |
| **cv2** | 0.475 | 0.777 | 0.418 | 0.943 | 0.468 | 0.406 | 2.593 |
| **cv3** | 0.164 | 0.545 | 0.123 | 0.913 | 0.118 | 0.112 | 3.435 |
| **cv4** | 0.505 | 0.772 | 0.407 | 0.941 | 0.443 | 0.399 | 2.607 |
| **NB** | 0.631 | 0.600 | 0.600 | 0.960 | 0.610 | 0.600 | 2.504 |
| **K** | 0.962 | 0.940 | 0.940 | 0.990 | 0.940 | 0.940 | 0.383 |
| **OR** | 0.309 | 0.330 | 0.330 | 0.930 | 0.320 | 0.310 | 3.125 |
| **J** | 0.686 | 0.700 | 0.700 | 0.970 | 0.710 | 0.700 | 2.331 |

#### 7.3.1.3 Farklı algoritmaların için farklı eğitim ve test veri kümesi sayıları ile karşılaştırması

Veri kümesinin 4 farklı bölünmesi sonrasında elde edilen 4 farklı eğitim ve test kümeleri üzerinde yapılan denemeler sonrasında, eğitim düğümü sayısının artmasının elde edilen sonuçlar üzerinde olumlu bir etkisinin olduğu gözlenmektedir.

**Tablo 7.35 : vovel karşılaştırma tablosu.**

| | ei | du | tp | tn | ke | f | sh |
|---|---|---|---|---|---|---|---|
| **768 eğitim düğümü ve 222 test düğümü** | | | | | | | |
| **e2** | 0.969 | 0.986 | 0.950 | 0.995 | 0.95 | 0.95 | 0.355 |
| **e4** | 0.901 | 0.953 | 0.845 | 0.985 | 0.85 | 0.845 | 0.464 |
| **mm2** | 0.924 | 0.968 | 0.893 | 0.990 | 0.898 | 0.894 | 0.505 |
| **512 eğitim düğümü ve 478 test düğümü** | | | | | | | |
| **e2** | 0.923 | 0.964 | 0.879 | 0.988 | 0.880 | 0.879 | 0.659 |
| **e4** | 0.894 | 0.947 | 0.828 | 0.983 | 0.830 | 0.828 | 0.671 |
| **mm2** | 0.860 | 0.941 | 0.808 | 0.981 | 0.816 | 0.809 | 0.824 |
| **348 eğitim düğümü ve 942 test düğümü** | | | | | | | |
| **e2** | 0.897 | 0.958 | 0.861 | 0.987 | 0.863 | 0.862 | 0.928 |
| **e4** | 0.755 | 0.874 | 0.626 | 0.963 | 0.636 | 0.622 | 1.245 |
| **mm2** | 0.821 | 0.931 | 0.779 | 0.978 | 0.787 | 0.781 | 0.993 |
| **256 eğitim düğümü ve 734 test düğümü** | | | | | | | |
| **e2** | 0.750 | 0.902 | 0.700 | 0.970 | 0.707 | 0.701 | 1.393 |
| **e4** | 0.631 | 0.846 | 0.561 | 0.957 | 0.563 | 0.559 | 1.561 |
| **mm2** | 0.693 | 0.874 | 0.627 | 0.963 | 0.645 | 0.629 | 1.668 |

# 8. SONUÇ

Bu tez çalışması içerisinde YSYBMG yöntemi, farklı bakış açıları ile yorumlanmış ve farklı özelliklere sahip gerçek veri kümeleri üzerinde başarılı sınıflandırma süreçleri oluşturulmuştur. Sınıflandırma probleminin çözümüne yönelik 16 farklı model kurulmuş ve bu kurulan modeller 7 farklı veri kümesi üzerinde denenmiştir. Bu denemeler, kurulan modellerin kullanılması amacı ile yazılmış olan IHDMR yazılımı ile yapılmıştır.

Önerilen sıralama algoritmaları ile veri kümeleri ve endeks uzayları farklı şekiller sıralanmış ve bu farklı sıralanmalar yolu ile faklı şekillerde eşleştirilmişlerdir. Önerilen yeni endeks düğümü algoritmaları ile test düğümlerinin oluşturulan endeks uzayı üzerinde eşleşecekleri endeks düğümlerinin belirlenmesi amaçlanmıştır. Bu yeni endeks düğümü belirlenmesi süreci, farklı metrikler ve farklı yaklaşımlar ile genişletilmiş, farklı veri setleri için farklı başarı düzeyleri yakalanmıştır.

Elde edilen sonuçların doğruluğu, üzerinde çalışılan veri kümelerinin 4 farklı parçalanması ile kurulan modellerin tekrar denenmesi yolu ile ve Weka veri madenciliği aracı içerisinden farklı özellikleri sebebiyle seçilmiş 4 farklı sınıflandırma algoritması ile karşılaştırılarak sınanmıştır. Yapılan çalışmalar sonrasın da, YSYBMG yöntemi gerçek veri kümeleri üzerinde denenmiş ve farklı özelliklere sahip veri kümeleri üzerinde çalışabilmesi için farklı algoritmalar ile yeniden şekillendirilmiştir. Yapılan denemeler sonrasında görülmüştür ki, YSYBMG nin esnek yapısı kullanılarak elde edilen yeni modeller ile farklı veri kümeleri üzerinde başarılı sınıflandırma süreçleri inşa edilebilmiştir.

Önerilen yöntemlerin gerçek veri kümeleri üzerine uygulanmasına yönelik hazırlanan IHDMR yazılımı sayesinde denemeler, kısa bir süre içerisinde tamamlanmıştır. Yazılım sayesinde kullanıcının deneme sürecine katkısı veri dosyasının seçilmesi seviyesine kadar çekilebilmektedir. Kullanıcı, seçeceği veri kümesi üzerinde herhangi bir

değişikliğe gitmek istediğinde bunu oldukça kolay bir şekilde yapabilmekte ve yaptığı değişiklikleri aynı an içersinde grafikler ve tablolar sayesinde gözlemleyebilmektedir.

Kullanıcı yaptığı değişiklikler sonrasında, yeniden oluşturulan veri bilgileri sayesinde yaptığı değişikliğin yeterli olup olmadığına ve eğer yeterli görülmedi ise yeniden değişiklikler yapması gerekip gerekmediğine karar verebilmektedir. Deneme sonrasında kullanıcı sonuçları oldukça ayrıntılı bir şekilde kullanıcı arabiriminden takip edebilmektedir. Aynı zamanda yapılan denemelerin pdf formatlı raporlarıda kayıt altına alındığından dolayı kullanıcı yaptığı denemenin sonuçlarını diğer ortamlara aktarabilmektedir.

Çoklu denemeler için oluşturulan raporlar sayesinde kullanıcı farklı yöntemlerin etkilerini aynı tablo içersinde gözlemleyebilmekte ve karşılaştırabilmektedir. Kullanıcı her bir farklı model için en iyi sonucu veren eğitim ve test kümelerini arff formatlı olarak elde etmektedir. Bu sayede iyi sonuçlar elde edilen yöntemleri diğer sınıflandırma algoritmaları ile eşit şartlarda karşılaştırabilme imkanı elde edilmiş olmaktatır.

Bu tez çalışması içerisinde yapılan denemelerde 4 farklı dizilim algoritması ve 4 farklı eşleştirme algoritması ile oluşturulan 16 farklı model 7 farklı veri seti üzerinde uygulanmıştır. Her veri seti için 4 farklı eğitim düğümü sayısı ile yapılan denmelerde, her deneme 30 defa rastgele seçilim uygulanarak tekrarlanmıştır. Bu tekrarların toplam sayısı 13440' tır. Yapılan denemeler sonrasında 15344 adet pdf ve arff formatlı rapor dosyası elde edilmiştir.

Deneme sonuçlarının değerlendirilmesi için Weka veri madenciliği aracı içersinden kullanılan profesyonel sınıflandırma algoritmalarına göre bu tez içersinde önerilen yöntemlerin sonuçları birçok denemede daha iyi olmuştur. Diğer durumlarda ise sonuçlar üzerindeki farklar oldukça düşük seviye olmuştur.

Yeni endeks düğümü algoritmalarını değerlendirmek gerekirse:

Çok değişkenli regresyona dayalı yöntem ile elde edilen sınıflandırma denklemleri başarılı sonuçlar vermiştir. Bu denklemler değişken olarak eğitim düğümü içerisindeki verileri aldıkları için hiçbir işlem yapılmadan sadece denklemde yerine koyma yöntemi ile sonuca varmaktadırlar. Bu yöntem "diabetes" veri kümesi için en iyi sonucu veren yöntemdir.

YSYBMG yöntemi içerisinde yer alan ve bu çalışma içerisinde "euclid" olarak anılan yeni endeks düğümü bulma algoritması, kullanılan gerçek veri kümeleri üzerinde genel olarak ortalama bir performans sağlamıştır. "vovel" veri kümesi için en iyi sonucu veren model içerisinde bu algoritma kullanılmıştır.

Mahalanobis uzaklığına dayalı tahminsel yaklaşım yöntemi genel olarak başarılı sınıflandırma modelleri içerisinde yer almışlardır. Bu yöntemin kullanıldığı modeller ya en iyi ya da en iyi ikinci en iyi sonucu vermişlerdir.

Değişen varyans oranları yönteminin içerisinde kullanıldığı modellerin, genellikle ortalama bir başarı seviyesine sahip olduğu gözlenmiştir. Bu yöntem değer aralıkları küçük ve içerdiği değerler arasındaki farkların oransal olarak küçük olduğu özniteliklere sahip veri kümeleri için iyi sonuçlar vermemektedir.

Yeni sıralama algoritmalarını değerlendirmek gerekirse:

"use rank" sıralama algoritmasını içeren modellerin öznitelikleri farklı sayıda değerler içeren veri kümeleri için iyi sonuç verdiği gözlenmiştir. Bu sıralama algoritması balance-scale, vovel ve blood iyi sonuçlar veren modeller içerisinde kullanılmıştır."use rate" sıralama algoritmasını içeren modeller genellikle düşük seviyede bir başarı düzeyi sağlamışlardır.

YSYBMG yöntemi içerisinde yer alan ve bu çalışma içerisinde “class” olarak anılan sıralama algoritması, kullanılan gerçek veri kümeleri üzerinde genel olarak iyi bir performans sağlamıştır. Bupa veri kümesi için en iyi başarı düzeyini yakalayan model içerisinde bu sıralama algoritması kullanılmıştır.

Genel olarak oluşturulan modellerin hangi veri kümelerinde etkili sonuçlar verdikleri Tablo 8.1 içerisinden takip edilebilir. Bu tablo içersinde modellerin başarılı sonuçlar verdikleri veri kümeleri verilmiştir. Tablo 8.1 içerisinde göze çarpan ilk detay kuşkusuz iris veri kümesi için elde edilen başarı durumudur. Bu veri kümesi için bütün modeller başarılı sonuçlar almıştırlar.

Bu veri kümesi için elde edilen başarının sebebi, veri kümesinin genel olarak sınıflandırma analizine uygun yapısıdır. Bu veri kümesi içerisindeki sınıflar ayrık olduğundan test düğümleri kolaylıkla sınıfladırılabilmektedir. Elde edilen sonuçlarda asıl dikkat çeken durum birçok deneme de bu veri kümesi için %100 başarı eldilmiş olmasıdır.

Çok değişkenli regresyona dayalı yöntem (mr) ile oluşturulan modellerin diabetes veri kümesi üzerinde iyi sonuçlar verdiği görülmektedir. Bu veri kümesi üzerinde Değişen Varyans Oranları yöntemi kullanılarak elde edilen iki modelinde ( cv2, cv4) iyi sonuçlar verdiği gözlenmektedir.

Elde edilen bu sonuçların bu veri kümesinin öznitelikleri içerisindeki değerlerin belli bir düzen ile dağılmış olmalarıdır. Bu düzeni tanımlamak gerekir ise özniteliklerin içerdikleri değer aralıkları ile standart sapmaları arasında güçlü bir pozitif korelasyon olduğu gözlenmiştir.

Mahalanobis uzaklığına dayalı tahminsel yaklaşım (mm) ile oluşturulan modellerin balance-scale veri kümesi üzerinde başarılı bir performans ortaya koyduğu gözlenmektedir. Blood veri kümesi üzerinde “class” sıralama algoritması kullanılarak hazırlanan 3 modelin (cv1, mm1, mr1) iyi sonuçlar verdiği gözlenmiştir.

**Tablo 8.1 : başarılı tahminler.**

| Model | Başarılı olduğu veri kümeleri |
|---|---|
| mr1 | İris, blood, bupa, diabetes |
| mr2 | İris, diabetes |
| mr3 | İris, diabetes |
| mr4 | İris, diabetes |
| e1 | İris, tae, vovel |
| e2 | İris, balance-scale, tae, vovel |
| e3 | İris |
| e4 | İris, vovel |
| mm1 | İris, balance-scale, blood, bupa |
| mm2 | İris, balance-scale, bupa, tae, vovel |
| mm3 | İris, balance-scale |
| mm4 | İris, balance-scale, blood |
| cv1 | İris, blood |
| cv2 | İris, blood, diabetes |
| cv3 | İris |
| cv4 | İris, diabetes |

Tae veri kümesinde ise “use rank” sıralama algoritması kullanılarak elde edilen iki modelin (mm2, e2) iyi sonuçlar verdiği gözlenmiştir. Aynı zamanda bu veri kümesi için euclid uzaklığına dayalı yeni endeks uzayı düğümü belirleme algoritması kullanılarak elde edilen iki model (e1, e2) iyi bir performans göstermiştir. Vovel veri kümesi içinde tae veri kümesi için geçerli bulunan başarı durumuna benzer bir durum söz konusudur. Aynı modeller bu veri kümesi içinde başarılı sonuçlar vermiştirler.

Bu veri kümesinin başarılı sonuçları arasındaki fark, bu veri kümesi için euclid uzaklığına dayalı yeni endeks uzayı düğümü belirleme algoritması kullanılarak elde edilen üç modelin (e1, e2, e4) iyi bir performans göstermesidir. Burada e4 modeli tae

veri kümesinden farklı olarak iyi bir performans göstermiştir.

Bupa veri kümesi için Mahalanobis uzaklığına dayalı tahminsel yaklaşım (mm) ile oluşturulan iki modelin (mm1, mm2) başarılı bir performans gösterdiği gözlenmiştir. Bu veri kümesi için "class" sıralama algoritması kullanılarak hazırlanan 2 modelin (mm1, mr1) iyi sonuçlar verdiği gözlenmiştir.

Elde edilen deneyimler sonrasında, yöntem üzerinde yapılacak çalışmalar ile veri kümelerin yapılarına yönelik analiz çalışmalarının yürütülebilineceği düşünülmektedir. Ayrıca kümeleme analizi için yöntemin geliştirilebileceği düşünülmektedir. Bu geliştirmelerin IHDMR yazılımına küçük eklemeler ile kolayca yapılabileceği düşünülmektedir. Bu tez sonrasında bu çalışmaların bu yönde sürdürülmesinin gerekliliği açıkca ortaya çıkmış bulunmaktadır.

# KAYNAKÇA

***Süreli Yayınlar***


Baykara, N. A.; Demiralp, M.: "*Hyperspherical or Hyperellipsoidal Coordinates in the Evaluation of HDMR Approximants*", The Fourth International Conference on Tools For Mathematical Modeling, St. Petersburg, Russia, June 23-28 (2003)

B.N. Rao, R. Chowdhury, "*Probabilistic analysis using high dimensional model representation and fast Fourier transform"*, International Journal for Computational Methods in Engineering Science & Mechanics 9 (2008) 342–357.

Demiralp, M.: "*High Dimensional Model Representation and Its Application Varieties*", The Fourth International Conference on Tools for Mathematical Modelling, St. Petersburg, Russia, June 23-28 (2003)

Demiralp, M.; Kurşunlu, A.: "*Additive and Factorized HDMR Applications to the Multivariate Diffusion Equation Under Vanishing Derivative Boundary Conditions*", Mathematical Research, Volume 9, St. Petersburg, Russia, June 23- 28 (2003), 315-327.

Demiralp, M.; Civelekoğlu, T.: "*An HDMR Application to the Schrödinger's Equation for Free Particles Under An External Field with Dipole Polarization and Vanishing Flux Boundary Conditions*", The Fourth International Conference on Tools For Mathematical Modelling, St. Petersburg, Russia, June 23-28 (2003)

Demiralp, M.; Tunga, M. A.: "*Data partitioning Via Generalized HDMR and Multivariate Interpolative Applications*", The Fourth International Conference on Tools For Mathematical Modelling, St. Petersburg, Russia, June 23-28 (2003)

Demiralp, M.; Kaman, T.: "*A HDMR Application to the Optimal Control of Harmonic Oscillator*", The Fourth International Conference on Tools For Mathematical Modeling, St. Petersburg, Russia, June 23-28 (2003)

Demiralp, M.; Yaman, İ.: "*HDMR Approximation of an Evolution Operator with a First Order Partial Differential Operator Argument*", App. Num. Anal. And Comp. Math., Wiley CHV, I, (2003) 287-296

Demiralp, M.; Tunga, B.; "*An Hybrid High Dimensional Model Representation Approximants And Their Utilization in Applications, Mathematical Research*", Volume 9, St. Petersburg, Russia, June 23-28 (2003), 438-446.

Demiralp, M ; Akkemik E, '*Algebraic Eigenvalue Problem Modeling via High Dimensional Model Representation*'' , The Fourth International Conference on Tools For Mathematical Modelling, St . Petersburg , Russia, June 23-28 2003

E. Demiralp and M. A. Tunga, "*A Hybrid Programming for Projective Displaying of High Dimensional Model Representation Approximants*", Mathematical Research, 9, 132-145, 2003.

I. Banerjee, M.G. Ierapetritou, "*Design optimization under parameter uncertainty for general black-box models*", Industrial & Engineering Chemistry Research 41 (2002) 6687-6697.

Kaya H.; Kaplan M.; Saygın H.: "*A recursive Algorithm for Finding HDMR Terms for Sensitivity Analysis*", Computational Methods in Sciences and Engineering, VOL 01, (2003) 302-305.

Kolmogorov, A. N.:"*On the Representation of Continuous Functions of Many Variables by Superposition of One Variable and Addition*", English Translation: American Math. Soc., 2, 28 (1963), pp.55-59

Li, G.; Schoendorf, J.; Ho, T.; Rabitz, H.; "*Multicut-HDMR with an Application to an Ionospheric Model*", Journal of Computational Chemistry, 25-9 (2004) 1149-1156

M. A. Tunga, "*A Matrix Based Indexing HDMR Method for Multivariate Data Modeling*", Journal of Mathematical Chemistry, 49(5), 1092-1114, 2011.

M. A. Tunga and M. Demiralp, "*A Factorized High Dimensional Model Representation on the Partitioned Random Discrete Data*", Appl. Num. Anal. Comp. Math., 1, No. 1, 231-241, 2004.

M. A. Tunga and M. Demiralp, "*A Reverse Technique for Lumping High Dimensional Model Representation Method*", WSEAS Transactions on Mathematics, 8, (5), 213-218, 2009.

M. A. Tunga and M. Demiralp, "*A Factorized High Dimensional Model Representation on the Nodes of a Finite Hyperprismatic Regular Grid*", Applied Mathematics and Computation, 164(3), 865-883, 2005. (Times Cited:21)

M. A. Tunga and M. Demiralp, "*Hybrid High Dimensional Model Representation (HHDMR) on the Partitioned Data*", Journal of Computational and Applied Mathematics, 185(1), 107-132, 2006. (Times Cited:16)

M. A. Tunga and M. Demiralp, "*A New Approach for Data Partitioning Through High Dimensional Model Representation*", Int. Journal of Computer Mathematics, 85(12), 1779-1792, 2008. (Times Cited:3)

M. A. Tunga and M. Demiralp, "*Bound Analysis in Univariately Truncated Generalized High Dimensional Model Representation for Random-Data Partitioning: Interval GHDMR*", Applied Numerical Mathematics, 59(6), 1431-1448, 2009. (Times Cited:3)

M. A. Tunga, "*An approximation method to model multivariate interpolation problems : Indexing HDMR*". Mathematical and Computer Modeling (2003),

M.C. Gomez, V. Tchijov, F. Leon, A. Aguilar, "*A tool to improve the execution time of air quality models*", Environmental Modeling & Software 23 (2008) 27–34

Rabitz H. And Alis, O.F.: "*Additive and Multiplicate High Dimensional Representation General Foundations of High Dimensional Model Representational Representations*", J. Math. Chem., 25, (1999) 197-233.

Rabitz, H.; Alıs, Ö.F.; Shorter, J.; Shim K.: "*Efficient Input-Output Model Representation*", Computer Physics Communications, 117 (1999) 11-20.

Rabitz, H.; Li, G.; Wang, S.; Georgopoulos, P. G.: "*Correlation Method for Variance Reduction of Monte Carlo Integration in RS-HDMR*", Journal of Computational Chemistry, 24 (2003) 277-283.

Rabitz, H.; Alış, Ö.F.: "*Efficient implementation of High Dimensional Model Representitaons*", J. Math. Chem., 29 (2001), 127-142.

Rabitz, H.; Li, G.; Wang, S.: "*Practical Approaches to Construct RS-HDMR Component Functions*", Journal of Physical Chemistry A, 106 (2002) 8721 8733

R. Chowdhury, B.N. Rao, "*Hybrid high dimensional model representation for reliability analysis*", Computer Methods in Applied Mechanics & Engineering 198 (2009) 753–765.

R. Chowdhury, B.N. Rao, A.M. Prasad, "*High dimensional model representation for piece-wise continuous function approximation*", Communications in Numerical Methods in Engineering 24 (2008) 1587–1609.

Sobol, I.M.: "*Theorems and Examples on High Dimensional Model Representation*" Reliability Engineering and Safety,79 (2003) 187-193

Sobol, I.M.: "*Sensitivity estimates for nonlinear mathematical models*", Mathematical Modeling and Computational Experiments 1, 407-414, 1993.

T. Ziehn, A.S. Tomlin, "*A global sensitivity study of sulfur chemistry in a premixed methane flame model using HDMR*", International Journal of Chemical Kinetics 40 (2008) 742–753.

T. Ziehn, A.S. Tomlin, "*GUI-HDMR—a software tool for global sensitivity analysis of complex models*", Environmental Modeling & Software 24 (2009) 775-785.

## ÖZGEÇMİŞ

**Adı Soyadı** : Çağrı AKSU

**Sürekli Adresi** : İstanbul/Türkiye

**Doğum Yeri ve Yılı** : Erzincan 1981

**Yabancı Dili** : İngilizce

**İlk Öğretim** : Ambarlı ilkokulu.

**Orta Öğretim** : Avcılar Ticaret Lisesi, Muhasebe. (1999)

**Lisans** : Kırıkkale üniversitesi, İstatistik. (2007)

**Yüksek Lisans** : Bahçeşehir üniversitesi

**Enstitü Adı** : Fen Bilimleri Enstitüsü

**Program Adı** : Bilgi Teknolojileri

**Yayımları** :

**Çalışma Hayatı :** Enerji sektöründe güncel sorunların modellenmesi ve çözüm algoritmaları ile ilgilenmektedir.