

**DOKUZ EYLÜL ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**

**BEYİN-BİLGİSAYAR ETKİLEŞİMİ**  
**VERİLERİN ANALİZİ VE UYGULAMALARI**



**Alican DOĞAN**

**Ocak, 2018**

**İZMİR**

# **BEYİN-BİLGİSAYAR ETKİLEŞİMİ VERİLERİN ANALİZİ VE UYGULAMALARI**

**Dokuz Eylül Üniversitesi Fen Bilimleri Enstitüsü**

**Yüksek Lisans Tezi**

**Bilgisayar Bilimleri Anabilim Dalı, Bilgisayar Bilimleri Programı**



**Alican DOĞAN**

**Ocak, 2018**

**İZMİR**

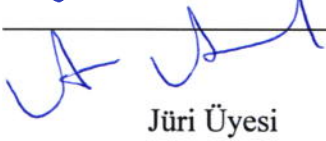
## YÜKSEK LİSANS TEZİ SINAV SONUÇ FORMU

ALİCAN DOĞAN, tarafından PROF. DR. EFENDİ NASİBOĞLU yönetiminde hazırlanan “BEYİN-BİLGİSAYAR ETKİLEŞİMİ VERİLERİN ANALİZİ VE UYGULAMALARI” başlıklı tez tarafımızdan okunmuş, kapsamı ve niteliği açısından bir Yüksek Lisans tezi olarak kabul edilmiştir.


Prof. Dr. Efendi NASİBOĞLU

  
Yönetici

Doç. Dr. Ayşesül Aylay Seyoğlu

  
Jüri Üyesi

Doç. Dr. Derya BİRANCI

  
Jüri Üyesi

Prof. Dr. Kadriye ERTEKİN

Müdür

Fen Bilimleri Enstitüsü

## TEŐEKKÜR

Her zaman bilimsel ilerlemeye yönelik hevesini koruyarak bu esin kaynađını bana aŐılayan, tecrübeleri ve öngörülerıyla tüm çalıŐmalarımnda kolay ve emin bir şekilde yol almama yardımcı olan danışmanım Prof. Dr. Efendi NASİBOĐLU'na, çalıŐtığımız konuya ilgi gösterip birçok konuda bana yardımcı olan ve benimle fikirlerini paylaşan Yrd. Doç. Dr. Mete EMİNAĐAOĐLU'na, araŐtırmalarımnda donanımsal destek veren, veriye ulaşmamı sađlayan ve literatür taramalarımnda yol gösterici bir rol üstlenen Sezin TUNABOYLU'ya, 2210-A Yurtiçi Yüksek Lisans Burs Programı kapsamında burs desteđi sađlayan TÜBİTAK'a ve son olarak tüm eğitim yaşamım süresince yanımda olduđunu hissettiren AİLEME içtenlikle teşekkürlerimi sunarım.

Alican DOĐAN

# BEYİN-BİLGİSAYAR ETKİLEŞİMİ

## VERİLERİN ANALİZİ VE UYGULAMALARI

### ÖZ

Vücudun ana kontrol mekanizması olan beyin, anlaşılması zor bir yapıya sahiptir. İnsan beyni üzerinde incelemeler yapılmaya başladığından beri günümüze dek farklı metotlarla, beynin hareketli yapısını anlaşılır hale getirmek için birçok çalışma yapılmıştır. *Elektroensefalografi (EEG)*, bu yöntemler arasında beyin aktivitesini görüntülemeye yarayanlardan bir tanesidir. Beyin – Bilgisayar Etkileşimi, gelişmekte olan ve üzerinde çok fazla sayıda çalışma yapılmamış bir alandır. Yalnızca düşünceler aracılığıyla teknolojik cihazlarla konuşabilme fikri pek çok yeni ufuklar doğurur.

Çalışmada kullandığımız Emotiv Epoc+ adlı cihaz, EEG sinyallerini okunabilir hale getirmek için faydalanılan araçlardan bir tanesidir. Kafatasında farklı loblara göre özelleşmiş olarak kendine has bölgelere yerleştirilen 14 farklı elektrot sayesinde bu cihaz aracılığıyla beyinsel aktivite gözlemlenir. Bu tez çalışmasında kullanılan EEG verileri, bahsedilen araç ve yazılım kullanılarak elde edilmiştir. Ayrıca çalışmayı desteklemek amaçlı kullanılan sınıflandırma algoritmalarının testi de UCI veritabanından alınan onaylanmış EEG verileri üzerinde gerçekleştirilmiştir.

Beyinde üretilen dalgaların üst üste çakışmasıyla EEG sinyalleri meydana gelir. Bu çalışmada, kullanılan cihaz ve yazılım aracılığıyla bilgisayara aktarılan EEG verileri dosyaya yazdırılmıştır. Verilerin tutulduğu dosyalara gerekli ön işleme operasyonları uygulanarak, kullandığımız programlama ortamı olan Visual C# 2015 de veriler ayrıştırılarak okunabilir hale getirilmiştir. Sınıflandırma aşamasında ise K-En Yakın Komşuluk, C x K – En Yakın Komşuluk, Naive Bayesian kullanılmıştır. Sınıflandırma algoritmalarının uzaklık ölçüm yöntemleri içinde ise Öklid, Bray- Curtis, Hellinger ve Cosine benzerlik ölçümleri bulunmaktadır.

Gerçekleştirdiğim tez çalışmasında, aynı denekten temin edilen anlık EEG sinyalleri kullanılmıştır. Deneğe deney sırasında dört temel ana yön (sağ, sol, yukarı, aşağı) gösterilen bir bilgisayar arayüzü sunulmuştur. Bu gösterilen dört yönden fare imlecinin yalnızca birine gitmesinin deneğin zihninde imgelemesi istenmiştir. Deneğin bu düşünsel süreci EEG ölçüm cihazı yardımıyla kaydedilmiştir. Her bir yön düşüncesi ilgili yön ismiyle etiketlenip dört farklı sınıfa ayrılmıştır. Bu şekilde gruplanan EEG sinyalleri farklı sınıflandırma algoritmaları kullanarak sınıflandırma doğruluk oranlarını karşılaştırmıştır. Nispeten başarılı oran veren algoritma, hangi yönün düşünüldüğünün bilinmediği EEG verileri üzerinde uygulanarak incelenmiştir.

**Anahtar Kelimeler:** Beyin bilgisayar arayüzü (BBA), elektroensefalogram (EEG), EEG göz durumu, sınıflandırma, K – en yakın komşuluk (K-NN), C x K – en yakın komşuluk (C x K-NN)

# **ANALYSIS OF BRAIN-COMPUTER INTERACTION DATA AND RELATED APPLICATIONS**

## **ABSTRACT**

The brain which is the central control mechanism of body, has a complex structure. Many researches have been made to make dynamic structure of human brain more understandable with many different methods Electroencephalography (EEG), is one of these methods able to visualize cerebral activity. Brain – Computer Interface is a developing field about which there are not many researches. The idea of being able to communicate with technological devices via thoughts causes new scientific horizons.

The device named Emotiv Epoc+, used in this study, is one of the tools benefited for making EEG signals readable. Cerebral activities are observed via this device thanks to 14 distinct electrodes placed upon proper specialized locations according to different lobes on brain. EEG data used in this thesis study, has been acquired by utilizing mentioned tool and software. Moreover, test of classification algorithms used with purpose of supporting study has been applied to verified EEG data obtained by UCI data warehouse.

EEG signals are emerged of collusion of the waves generated in brain. EEG data transmitted to a computer via utilized device and software are written to a file. In classification phase, K-Nearest Neighborhood, C x k- Nearest Neighborhood and Naive Bayesian are utilized. As distance measuring methods of these classification algorithms, Euclidean, Bray-Curtis, Hellinger and Cosine Similarity Metrics take place.

In this study, instantaneous EEG signals supplied by the same subject are used. A computer interface in which four main directions are shown is presented to the subject during an experiment. Imagination of movement of the mouse cursor to only one of these four directions is asked from subject. This thinking process of subject is recorded with the help of an EEG measurement device. Each thought of a direction is labeled with respective direction name and all signals are divided into four different classes

with respect to their labeled direction names. EEG signals grouped in this manner are classified using distinct classification algorithms and classification accuracy rates of those algorithms are compared. Quite more successful algorithms are analyzed by being applied to EEG data in which unknown directions are thought among these four directions.

**Keywords:** Brain computer interface (BCI), elektroensefalogram (EEG), EEG eye state, classification C x K – nearest neighborhood (C x K – NN), K – nearest neighborhood (K-NN)





## İÇİNDEKİLER

	Sayfa
YÜKSEK LİSANS TEZİ SINAV SONUÇ FORMU .....	ii
TEŞEKKÜR.....	iii
ÖZ .....	iv
ABSTRACT.....	vi
ŞEKİLLER LİSTESİ .....	x
TABLolar LİSTESİ.....	xi
<b>BÖLÜM BİR - GİRİŞ.....</b>	<b>1</b>
1.1 Beynin Yapısı.....	1
1.2 EEG (Elektroensefalogram).....	2
1.2.1 EEG Osilasyonları .....	4
1.2.2 Artefakt .....	6
<b>BÖLÜM İKİ - BEYİN BİLGİSAYAR ARAYÜZÜ (BBA) .....</b>	<b>8</b>
2.1 Tarihçe.....	9
2.2 BBA'nın Bileşenleri.....	9
2.2.1 İşlevsel Bileşenler.....	10
2.2.2 Deri Altına Müdahalesiz Teknikler.....	11
<b>BÖLÜM ÜÇ - SİNYAL İŞLEME.....</b>	<b>14</b>
3.1 Zaman Alanı.....	14
3.2 Frekans Alanı .....	16
3.3 Beyin Bilgisayar Arayüzü Aşamaları.....	19
<b>BÖLÜM DÖRT - KULLANILAN EKİPMANLAR VE YÖNTEMLER.....</b>	<b>23</b>
4.1 Yararlanılan Malzeme, Donanım ve Yazılımlar .....	23
4.1.1 Malzemeler.....	23

4.1.2 Donanımlar .....	25
4.1.3 Yazılımlar .....	25
4.2 Veriseti .....	26
4.3 Veri Önışleme .....	27
4.4 k – En Yakın Komşuluk Algoritması .....	30
4.5 C x k – En Yakın Komşuluk Algoritması .....	32
4.5.1 10 Parça Çapraz Doğrulama.....	36
4.5.2 Test Aşaması .....	37
<b>BÖLÜM BEŞ - DENEYSEL ÇALIŞMA .....</b>	<b>38</b>
5.1 C x k – En Yakın Komşuluk Algoritmasıyla EEG Göz Durumu Verilerini Sınıflandırma.....	40
5.2 C x k – En Yakın Komşuluk Algoritmasıyla Temel Yönleri (Yukarı, Sol, Sağ, Aşağı) Verilerini Sınıflandırma .....	40
5.3 Program Arayüzü .....	49
<b>BÖLÜM ALTI - DEĞERLENDİRME VE SONUÇLAR .....</b>	<b>55</b>
<b>KAYNAKLAR .....</b>	<b>59</b>
<b>EKLER.....</b>	<b>63</b>

## ŞEKİLLER LİSTESİ

	Sayfa
Şekil 1.1 Beynin ana lobları .....	2
Şekil 1.2 Hans Berger tarafından yayınlanan ilk EEG kaydı.....	3
Şekil 1.3 Delta dalgası.....	4
Şekil 1.4 Teta dalgası .....	5
Şekil 1.5 Alfa dalgası .....	5
Şekil 1.6 Göz açıldığında alfa dalgasında oluşan değişim.....	6
Şekil 1.7 Beta dalgası.....	6
Şekil 1.8 Gama dalgası.....	6
Şekil 1.9 Artefakt çeşitleri.....	7
Şekil 2.1 BBA sistemindeki işlevsel bileşenler ve geribildirim döngüleri .....	11
Şekil 2.2 EEG ile istila etmeyen sinyal alımı için elektrotların yerleşimi .....	13
Şekil 3.1 Zaman alanında yer değiştirme kaydı .....	16
Şekil 3.2 Yer değiştirmenin dolaylı kaydı .....	16
Şekil 3.3 Basitleştirilmiş osilograf işlemi .....	17
Şekil 3.4 Basitleştirilmiş osiloskop işlemi .....	17
Şekil 3.5 Herhangi bir dalga sinüs dalgalarının birleşimiyle üretilebilir .....	18
Şekil 3.6 Zaman ve frekans alanları arasındaki ilişki .....	19
Şekil 3.7 Frekans spektrum örnekleri .....	17
Şekil 3.8 BBA'nın temel işlevsel elemanları .....	22
Şekil 4.1 Beyin Bilgisayar Arabirimi.....	24
Şekil 4.2 Elektrot uçlarındaki sensör sıvısı(a) ve sensör pedleri (b).....	25
Şekil 4.3 Emotiv Epoc+ 14 Channel EEG kayıt cihazı.....	26
Şekil 4.4 Emotiv Pure EEG yazılımının uygulama arayüzüne ait görüntü.....	27
Şekil 4.5 İkili sınıflandırma sırasında $k = 1, 2$ ve $3$ değerleri için $k - NN$ ile sınıf belirleme örneğinin görselleştirilmesi .....	32
Şekil 4.6 $k$ seçiminin çok yüksek bir değer ile yapılmasının yarattığı etkiyi gösteren basit bir görsel .....	33
Şekil 4.7 $C \times k - NN$ algoritmasının taslak kodu .....	24
Şekil 5.1 10 Parça Çapraz Doğrulama aşaması verisetlerinin çeşitli algoritmalara göre karşılaştırmalı sonuç grafiği .....	44

Şekil 5.2 10 Parça Çapraz Doğrulama aşaması Yukarı, Sol, Sağ, Aşağı verisetinin farklı k değerlerine göre karşılaştırmalı sonuç grafiği.....	47
Şekil 5.3 EEG Göz Durumu verileri için C x k – NN ve k –NN yöntemlerinin başarı oranlarını 1-10 arasındaki k değerleri için karşılaştıran grafik.....	51
Şekil 5.4 k - NN ve C x k - NN sınıflandırması yapan programımızın arayüz ana ekranı .....	52
Şekil 5.5 Program arayüzü detaylı sınıflandırma sonuçları ekranı .....	53
Şekil 5.6 Program arayüzü sınıflandırma başarı oranları ekranı .....	55
Şekil 6.1 EEG Göz Durumu verileriyle yapılan bazı çalışmaların başarı oranlarının karşılaştırmalı grafiği.....	51



## TABLolar LİSTESİ

	<b>Sayfa</b>
Tablo 4.1 Dosyaya kaydedilmiş EEG veri sinyallerinin görünümü .....	28
Tablo 4.2 Düzgün veri örnekleri .....	30
Tablo 4.3 Bozuk veri örnekleri .....	29
Tablo 4.4 Ayıklama sonrası elde kalan tüm verilerin sınıflara göre dağılımları .....	29
Tablo 4.5 10 Parça Çapraz Doğrulama öncesi her sınıftaki verilerin parçalara göre dağılımı .....	29
Tablo 4.6 Test aşamasında test verilerinin sınıflara göre dağılımı .....	39
Tablo 5.1 EEG sinyallerinin excel dosyasına kayıtlı halinin görünümü.....	29
Tablo 5.2 Öğrenim kümesindeki verilerin 10 parçalı çapraz doğrulama sırasında her bir sınıf için satır sayıları .....	42
Tablo 5.3 Test kümesindeki verilerin test sırasında her bir sınıf için satır sayıları ...	42
Tablo 5.4 10 Parça Çapraz Doğrulama aşaması verisetlerinin çeşitli algoritmalara göre karşılaştırmalı sonuç tablosu .....	43
Tablo 5.5 C x k - NN algoritmasının her bir parçadaki farklı k değerlerine göre aldığı sınıflandırma sonuçları .....	45
Tablo 5.6 k - NN algoritmasının her bir parçadaki farklı k değerlerine göre aldığı sınıflandırma sonuçları .....	46
Tablo 5.7 10 Parça Çapraz Doğrulama aşaması Yukarı, Sol, Sağ, Aşağı verisetinin farklı k değerlerine göre karşılaştırmalı sonuç tablosu.....	46
Tablo 5.8 Test aşaması Yukarı, Sol, Sağ, Aşağı verisetinin farklı dağılım oranlarına her sınıftaki sinyal miktarları.....	47
Tablo 5.9 Test aşaması Yukarı, Sol, Sağ, Aşağı verisetinin C x k - NN algoritmasıyla farklı k değerlerine göre karşılaştırmalı sonuç tablosu.....	48
Tablo 5.10 Test aşaması Yukarı, Sol, Sağ, Aşağı verisetinin k - NN algoritmasıyla farklı k değerlerine göre karşılaştırmalı sonuç tablosu.....	48
Tablo 5.11 Test aşaması Yukarı, Sol, Sağ, Aşağı verisetinin %75-%25 dağılımına göre C x k - NN algoritmasıyla farklı uzaklık ölçümleri için başarı oranları....	49
Tablo 5.12 Test aşaması Yukarı, Sol, Sağ, Aşağı verisetinin %67-%33 dağılımına göre C x k - NN algoritmasıyla farklı uzaklık ölçümleri için başarı oranları....	49

Tablo 5.13 Test aşaması Yukarı, Sol, Sağ, Aşağı verisetinin %75-%25 dağılımına göre k - NN algoritmasıyla farklı uzaklık ölçümleri için başarı oranları .....	49
Tablo 5.14 Test aşaması Yukarı, Sol, Sağ, Aşağı verisetinin %67-%33 dağılımına göre k - NN algoritmasıyla farklı uzaklık ölçümleri için başarı oranları .....	50
Tablo 5.15 EEG Göz Durumu verileri için C x k - NN ve temel k - NN yöntemlerinin başarı oranlarını 1-10 arasındaki k değerleri için karşılaştıran tablo.....	51
Tablo 6.1 EEG Göz Durumu verileriyle yapılan bazı çalışmaların başarı oranları ...	58



# BÖLÜM BİR

## GİRİŞ

Tüm potansiyel işlevsel mekanizmasının gerçekleşme süreçlerinin günümüzde hala tam olarak saptanamadığı beyin, diğer vücut organlarına kıyasla oldukça anlaşılması zor bir yapıdadır. Beyin, insanın günlük yaşamda hayata geçirdiği tüm eylemleri gerçekleştirir. Hareket etme, hafızadan bilgi çağırma, soluk alış-verişi, yabancı dil konuşma ve daha birçok aktiviteyi yürütür.

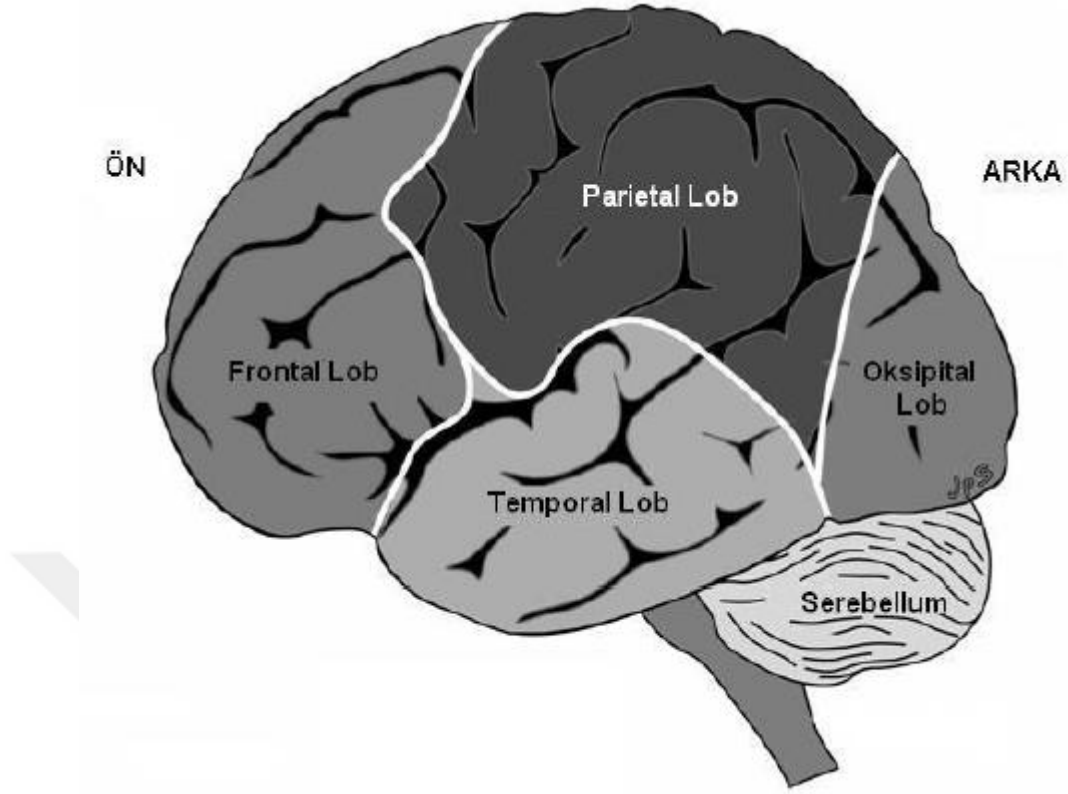
Beynin sahip olduğu nöron miktarı 100 milyardan fazladır. Aşağı yukarı 1400 gr ağırlığında olup, kafatası iskeleti sayesinde hasar verebilecek dış etmenlerden muhafaza edilir. Vücut dengesini sağlar ve kişinin aldığı kararları yönetir. Her bir nöronu iletişim ağındaki bir parça olarak düşünürsek ve ağdaki elemanların birbirleriyle kuracağı iletişim bağlantılarını hayal edersek beynin karmaşık yapısını idrak etmek kolaylaşacaktır.

Beyin mekanizması da, diğer mekanizmalar gibi harici bir girdi alıp onu işleyip bir çıktı oluşturur. Beyin söz konusu olduğunda da sistem teori prensipleri ve analiz yöntemleri uygulanabilir (Tülay, 2009).

Beynin kendi içinde nasıl çalıştığından bahsetmek için beynin yapısı hakkında bilgi verdikten sonra beynin elektriksel aktivitesini gözlemlemeye yarayan *EEG* (*Elektroensefalografi*) kullanımına değinilecektir.

### 1.1 Beynin Yapısı

Yarıküre anlamına gelen hemisfer ismiyle adlandırılan 2 kısım beyin kabuğu anlamına gelen *serebral korteksi* (*cerebral cortex*) meydana getirir. Serebral kortekste dış dünyaya dair duyular aracılığıyla algılamalar gerçekleşir (işitme, görme, koku alma vb.). Yarıkürelerin her bir tanesi *frontal (F)*, *parietal (P)*, *temporal (T)* ve *oksipital (O)* olmak üzere 4 ayrı lobdan oluşur.



Şekil 1.1 Beynin ana lobları (Tülay, 2009)

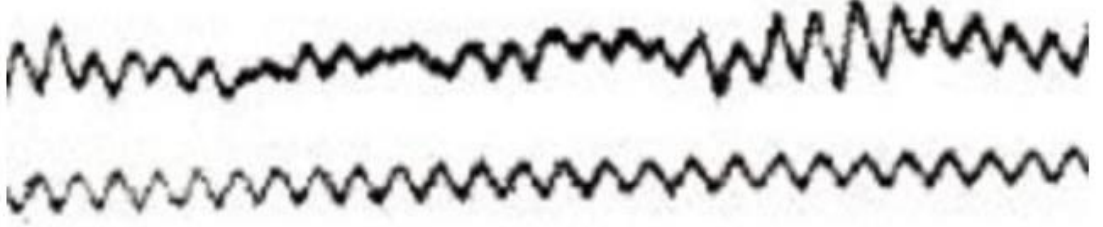
Her bir lob ayrı işlevler için özelleşmiştir. Şekil 1 de tüm bu bahsedilen lobların beyindeki yerleşimi temsil edilmiştir. *Frontal lob*, karar almada, hislerin hâkimiyetinde ve sorunlara çözüm bulmada, *parietal lob*, görme ve temas yoluyla idrak etmede, *temporal lob*, duyma, uzun süreli bellek ve konuşmada, son olarak *okspital lob* ise görsel algılamada etkindir (Aydemir, 2009).

## 1.2 EEG (Elektroensefalogram)

*EEG* beyinde yer alan nöronların birbirleriyle etkileşimini daha anlaşılır hale getirmek için sinir uçlarındaki elektriksel potansiyel farklarını sayısal veriye çeviren sinyallerdir. İsmi Yunan kökenli sözcük grubundan almış olup, “Beyin Elektriksel Resmi” anlamını taşır. İlk defa Richard Caton 1875 yılında beyin elektriksel değişimini incelemiştir. Ancak, burada yapılan gözlem hayvanlar üzerinde gerçekleşmiştir. İlk insan beyni üzerinde yapılan analiz ise 1929 yılında Hans Berger



tarafından gerçekleştirilmiştir. Bu analiz için kaydedilen *EEG* örneği Şekil 1.2’de gösterilmiştir.



Şekil 1.2 Hans Berger tarafından yayınlanan ilk *EEG* kaydı (Tülay, 2009)

1970 öncesi itibariyle *EEG* kayıtları bu alanda yapılan çalışmalara dâhil edilmemiştir. Daha sonraki süreçte ise *EEG* verilerinin beynin işlev mekanizmasını yansıtılabilmek için kayda değer olduğu anlaşılmış ve üzerinde durulmaya başlanmıştır. *EEG* sinyallerini etkileyen birçok faktör bulunur. Bunlardan bazıları *EEG* sinyali alınan denneğin yaşı, cinsiyeti, beyin dalgası frekans aralığı, göz açıp kapama hali ve beyinsel hasarların varlığıdır. Bunlara ek olarak, dış uyaranların kişi üzerinde yarattığı duygu değişimleri de *EEG* sinyalleri üzerinde etki bıraktığı kanıtlanmıştır.

*EEG* verilerini elde etme yöntemleri istila eden yöntemler ve istila etmeyen yöntemler olarak ikiye ayrılır. İstila eden yöntemler kafatasındaki derinin içine nüfuz ederek beyinle temas eden elektrotlar sayesinde sinyale ulaşır. Bu tarz yöntemler cerrahi operasyon ile gerçekleştirilebilir. Öte yandan, istila etmeyen yöntemlerde ise beyin ile direkt bir temas olmaksızın kafatası derisinin üstüne yerleştirilen elektrotlarla gerekli ölçümler yapılır. Denek üzerinde fiziksel bir zedelenme bırakmaz. Bu tez çalışmamızda kullandığımız veriler istila etmeyen bir yöntem kullanılarak elde edilmiştir.

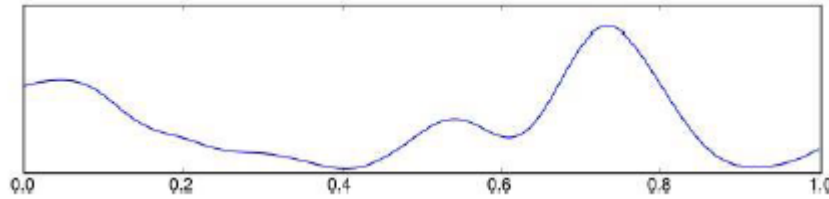
Güncel yaşamda, *EEG*’nin kullanım alanları arasında, klinik tedaviler ve sinir bilimi üzerine yapılan çalışmalar yer alır diyebiliriz. Sinir bilimi alanında, beyinden bilinçli olarak düşünülen bir düşüncenin yöneltilebilir sorulara göre kısmi olarak tespiti

ve klinik alanda sara hastalığı çeken bireylerin iyileşme sürecinde yardım amaçlı kullanılabilir.

### 1.2.1 EEG Osilasyonları

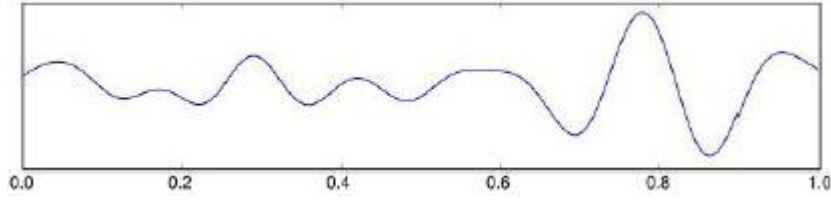
Frekans aralıkları birbirinden farklı olan birden fazla dalga üst üste çakıştığında EEG osilasyonlarını oluşturur. Bu dalgaların çakışması olayı süper pozisyon (süperposition) olarak adlandırılır. Dalgaların frekansı en az 0,5 Hz olup, 100 Hz'i de geçebilir. Sahip olabileceği genlik ölçüleri ise en düşük 10  $\mu V$  ve en yüksek 200  $\mu V$  olabilir. Üzerinde hiçbir önileme yapılmamış EEG kaydında bu dalgalar iç içe olduğundan birer birer saptanamaz. Ancak, filtreleme işlemleri gerçekleştirildikten sonra her bir dalga kendi başına gözlenebilir.

- **Delta( $\delta$ ) Dalgaları:** Frekansları 0,6 ile 4 Hz arasındadır ve genlik miktarları 20 ile 400  $\mu V$  arasında değişebilir. Genel anestezi ya da derin bir uyku gibi çok düşük seviyedeki beyinsel aktivite durumlarında tespit edilmesi çok daha yaygındır. Delta dalgalarının oluşturduğu şekle örnek Şekil 1.3'te görülebilir.



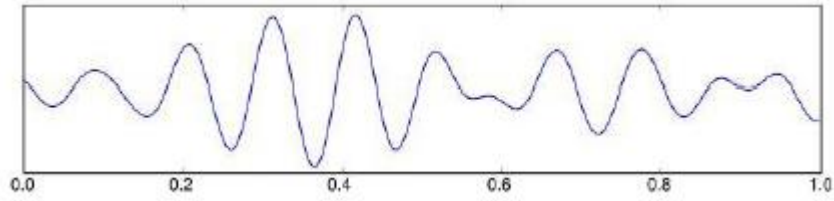
Şekil 1.3 Delta dalgası (Tülay, 2009)

- **Teta( $\theta$ ) Dalgaları:** Frekans aralıkları 4 Hz ile 8 Hz arasında ve genlikleri ise 5 ile 100  $\mu V$  arasında değişiklik gösterir. Uykunun rüya görüldüğü evresi, stres veya normal bireylerde normal derin anestezi gibi beyin düşük seviyede çalıştığı durumlarda oluşur. Teta dalgalarının oluşturduğu şekle örnek Şekil 1.4'te görülebilir.



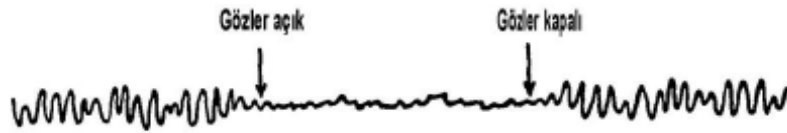
Şekil 1.4 Teta dalgası (Tülay, 2009)

- **Alfa( $\alpha$ ) Dalgaları:** Frekans aralığı  $8-13\text{ Hz}$  ve genlik aralığı  $2-10\ \mu V$  dur. Bilinci açık birey gözlerini kapattığında, herhangi bir dış uyarana maruz kalmadığında ve zihinsel ve fiziksel olarak tamamen dinlenme halinde olduğunda alfa dalgaları oluşur. Alfa dalgalarının oluşturduğu şekle örnek Şekil 1.5'te görülebilir.



Şekil 1.5 Alfa dalgası (Tülay, 2009)

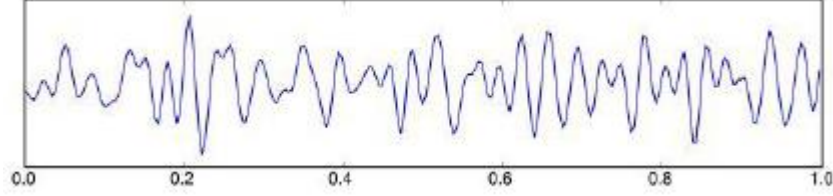
Bireyin beyninin alfa durumundan çıkıp geçici olarak beta durumuna geçmesi tespit edilebilir. Kişi alfa durumundayken göz açtığında ya da bir uyarana tepki verdiğinde dalgalarda betaya yönelik bir değişim gerçekleşir. Gözler açıldığında gerçekleşen alfa dalgasındaki farklılık Şekil 1.6'da gösterilir.



Şekil 1.6 Göz açıldığında alfa dalgasında oluşan değişim (Tülay, 2009)

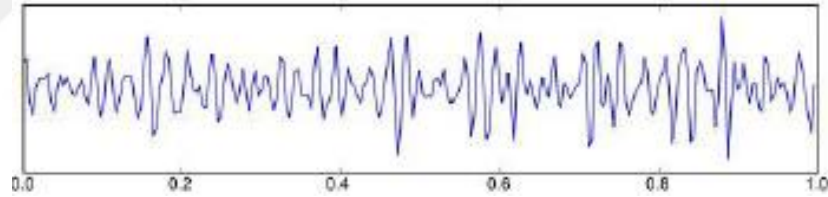
- **Beta( $\beta$ ) Dalgaları:** Frekansları  $13\text{ Hz}$ 'i geçmektedir ve genlikleri  $1-6\ \mu V$  arasındadır. Beta dalgaları normal koşullar altında en yüksek seviye

aktivitelerle eşleşir. Beta dalgalarının oluşturduğu şekle örnek Şekil 1.7’de görülebilir.



Şekil 1.7 Beta dalgası (Tülay, 2009)

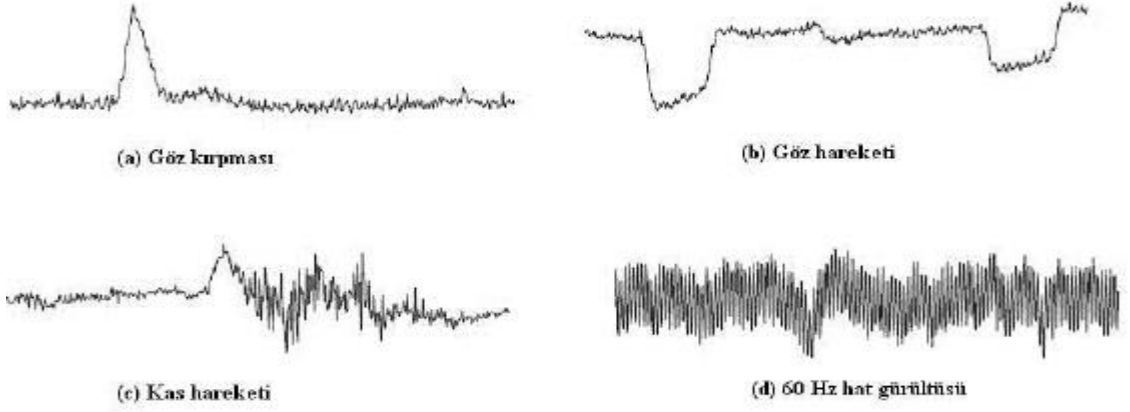
- **Gama( $\gamma$ ) Dalgaları:** Frekansları  $30\text{ Hz}$ 'in üstündedir. Hatta bazı durumlarda  $100\text{ Hz}$  değerine ulaşabilirler. Olağanüstü koşullarda beynin çok kısa süre içinde çok hızlı düşünme eylemi gösterdiği durumlarda gerçekleşir. Günlük yaşamda sık rastlanmaz. Gama dalgalarının oluşturduğu şekle örnek Şekil 1.8 de görülebilir.



Şekil 1.8 Gama dalgası (Tülay, 2009)

### 1.2.2 Artefakt

Artefakt, beyinden elektriksel aktiviteye dair sinyal alınma işlemi sırasında, istenilen veri dışında sinyale karışan gürültülere verilen isimdir. Beyin tarafından üretilmeyen sinyaller bu gruba girer. Artefaktlar hem deneyin sebep olduğu durumlardan hem de teknik aksaklıklardan doğabilir. Çeşitli etkenlerin sebep olduğu artefaktların sinyallerde oluşturduğu değişiklikler Şekil 1.9’da yansıtılır.



Şekil 1.9 Artefakt çeşitleri (Tülay, 2009)

**a-) Göz Kırpması:** *EEG* sinyallerini kayıt etme esnasında en sık karşılaşılan artefaktlardan biridir. Genliği, saf *EEG* sinyaline göre daha fazladır.

**b-) Göz Hareketi:** Göz kırpmasına kıyasla, gözün bir yöne doğru oynaması kafatası üzerinde daha geniş alana yayılım yapmasına sebep olur. İki artefakt çeşidi de birbirine yakın aralıklarla gerçekleşir.

**c-) Kas Hareketi:** Frekansı oldukça geniş bir artefakt türüdür. Denek ilgili kas üzerinde bir kasılma gerçekleştirdiğinde o kasın bulunduğu bölge etrafındaki kanallarda gözlemlenir.

**d-) Hat Gürültüsü:** *EEG* sinyalleri yalnızca birey tarafından değil, çevredeki elektriksel etmenlerden de etkilenebilir. Frekansı  $50\text{ Hz}$  ve üzeri olan elektrik kaynakları *EEG* sinyallerine etki edebilir. Notch filtresi bu çeşit harici artefaktların gürültüsünü gidermek için kullanılır.

## BÖLÜM İKİ

### BEYİN BİLGİSAYAR ARAYÜZÜ (BBA)

İnsanların herhangi bir uzuvlarını kullanmadan bilgisayarlara ya da bilgisayar tabanlı dijital mekanizmalara komut vermesini sağlayan sistemler insan beyni ve komut verilen aygıt arasında doğrudan iletişimi amaçlar.

Literatürde genellikle Beyin Bilgisayar Arayüzü (Brain Computer Interface), kısaca BBA olarak karşımıza çıkan bu çalışma alanının temel girdi (*input*) elemanı insan düşünceleridir. Bu düşünceleri girdi olarak okuyabilme fikri, insanların akıllarından bir şey geçirdiklerinde beyinlerinin çeşitli bölgelerinde elektriksel sinyaller oluşmasına dayanır. Nitekim bu sinyallerin elektronik ortamda algılanması gerekir.

BBA'nde kullanılan sinirsel aktivitenin kayıt edilmesi istila eden ve istila etmeyen teknikler sayesinde gerçekleştirilir. İstila eden tekniklerde elektrotlar kafatasının 1-2 mm içine sızarak deneğe zarar verme riski taşır. Bu nedenle bu tür teknikler genellikle kobaylar ya da maymunlar üzerinde denenir. Bazı durumlarda istila etmeyen tekniklere göre daha verimli işlevsellik gösterdiğinden performansından ötürü kullanılabilir. Kaydedilen sinyaller daha fazla bilgi taşır ve daha çabuk yanıt verir. Bu nedenle eğitim (training) gerekliliğini azaltmış olur.

Öte yandan, istila etmeyen teknikler ise deneğin kafatası üzerine yerleştirilir ve deri altından içeri sızmadığı için deneğe zarar verme durumu söz konusu değildir. EEG bu tekniklerden biridir. *EEG* yani *Elektroensefalogram*, kafatasının belli bölgelerine yerleştirilen elektrotlar sayesinde elektrotların bağlandığı kanallardaki elektriksel değişimi algılar. Elektrotların iki ucundaki potansiyel fark *mV* (*milivolt*) cinsinden ölçülür ve bu değişim zamana bağlı olarak kaydedilir.

Beyinsel aktiviteyi kaydetmeye yarayan bir diğer teknik ise Dikotik Dinleme (*Dichotic Listening – DL*)'dir. Davranışsal seviyede beyin asimetrisini değerlendirmek için kullanılır. Her bir kulağa eş zamanlı olarak birer duyuşsal uyarıcı

yerleřtirerek yapılır. Beyin, aynı anda her iki kulaktan eřit yoęunlukta alınan iki duyuşal uyarıcıyı tanımlayamaz.

## 2.1 Tarihçe

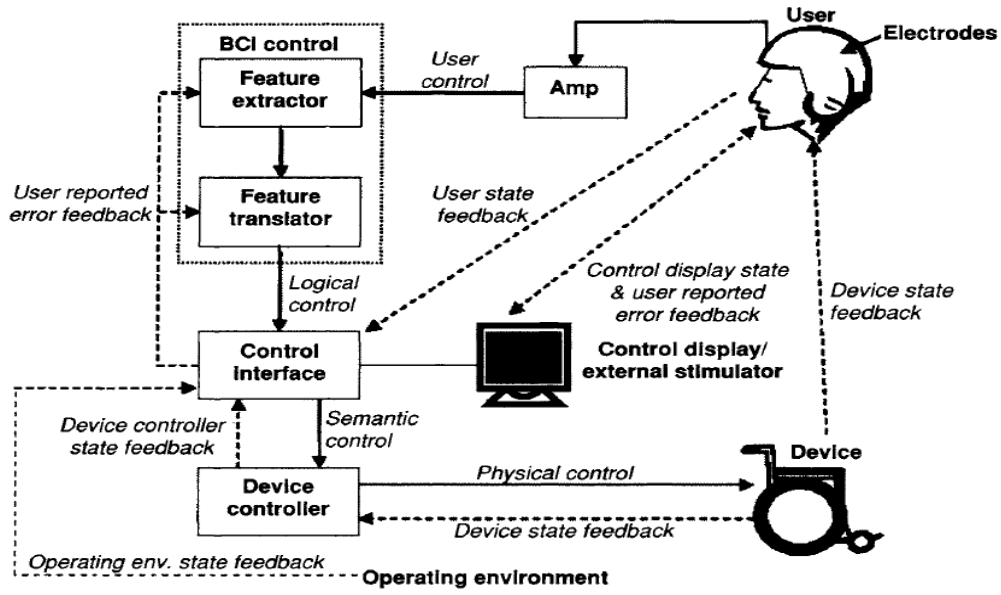
1929'da Hans Berger, beyin aktivitesi tarafından üretilen elektriksel potansiyelleri kaydeden cihaz ortaya çıkardı. Daha sonradan *EEG* yani *Elektroensefalogram* adını aldı.

1970'lerde, arařtırmacılar kafatasından kayıt edilen elektriksel aktiviteye dayanan ilkel kontrol sistemleri geliřtirdiler. Pentagon'un İleri Arařtırma Proje Ajansı(*DARPA*), askerlere yardım etmek için geliřtirilen biyonik cihazlar üzerinde çalışan projelere maddi katkı vermeye bařladı.

1977'de Jacques Vidal tarafından yapılan çalışma, beyinsel aktivitedeki sinyallerin, kullanıcın amacıyla iletiřim kurabilmek için verimli bir řekilde kullanılabileceęini kanıtladı.

## 2.2 BBA'nün Bileřenleri

2003'de Mason ve Birch tarafından yapılan çalışma BBA sisteminin bir modelini ortaya koydu. *EEG*'de *EEG* sinyalleri arasındaki řekil ve davranıř benzerlięinin, büyüklük benzerlięinden çok daha deęerli olduęu öne sürüldü.



Şekil 2.1 Beyin Bilgisayar Arayüzü sistemindeki işlevsel bileşenler ve geribildirim döngüleri (He, 2005)

Şekil 2.1’de, gelişmiş bir BBA sistemindeki bileşenler ve birbirleriyle ilişkileri gösterilmiştir. *Amp*, *Amplifier*’ı temsil eder. Özellik çıkarıcı (*Feature Extractor*) işlenmemiş sinyalleri ilgili özellik vektörlerine dönüştürür. Bu vektörler özellik dönüştürücüsü (*feature translator*) tarafından mantıksal kontrol sınıflarına ayrılır. Kontrol arayüzü ise mantıksal kontrolleri, cihaz kontrolcüsünün (*device controller*) algılayabileceği semantik kontrollere dönüştürür. Cihaz kontrolcüsü bu anlamsal kontrolleri, cihazın uygulayabileceği fiziksel komutlara dönüştürür. Böylece BBA sistemi, kullanıcının niyetini cihazın gerçekleştirdiği bir eyleme uygulamış olur.

### 2.2.1 İşlevsel Bileşenler

BBA sisteminin çalışma ortamındaki değişiklikleri saptayan bileşenlere denir. Kullanıcı ise BBA sistemini için girdi olarak kullanılan kontrol sinyalini üretmek için kasıtlı olarak beyinsel aktivitesinde değişiklik yapan bireylerdir.

İstila eden teknikler genellikle maymunlar ve fareler üzerinde uygulanır ve genellikle Beyin-Makine Arayüzü (BMA) olarak adlandırılır. Bu tekniklerin avantajı, çok yüksek örnekleme oranlarındaki bireysel nöronlardan kayıtlar alınabilir. Bu



kaydedilen sinyaller daha fazla bilgi içerdiğinden ve daha çabuk yanıtlara izin verdiğinden, daha az antrenmana ihtiyaç duyar.

Bu tekniklerin dezavantajı ise, sinyalin uzun-dönem sürekliliğini elde etmek günlerce ve yıllarca sürebilir ve oldukça zor olur.

### **2.2.2 Deri Altına Müdahalesiz Teknikler**

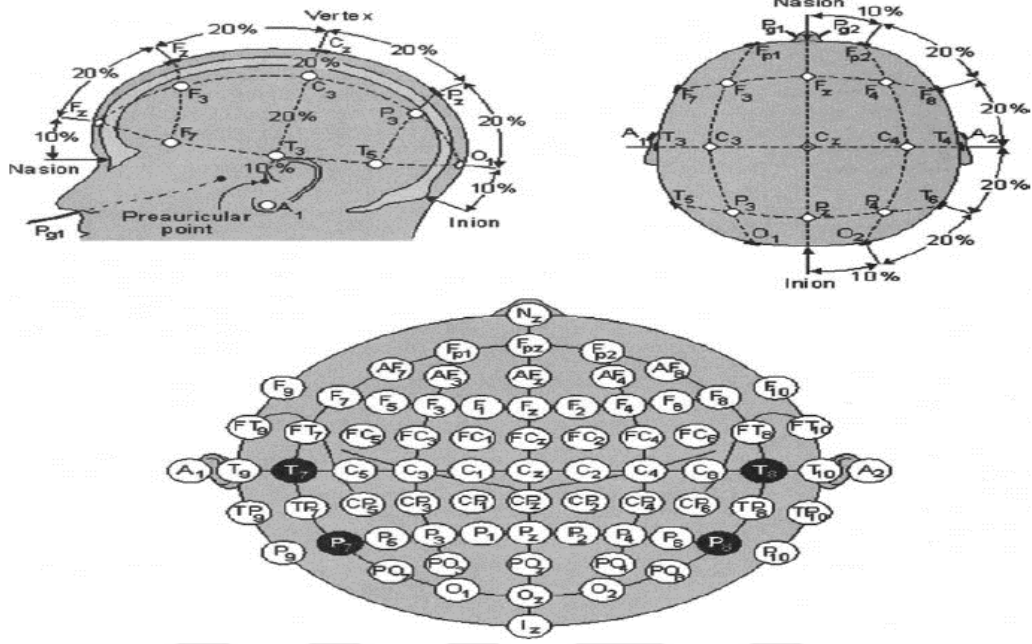
Beyinsel aktivite istila etmeyen araçlar yoluyla birçok yöntemle ölçülebilir. İstila etmeyen yöntemlerin avantajı herhangi bir medikal operasyon ya da cihaza kalıcı bir bağlılık gerektirmediği için kullanıcılar açısından sağlık yönünden bir rahatsızlık riskini azaltır. Beyine sızıntı yapmadan ölçüm alabilme metotları arasında bilgisayarlı tomografi (*CT*), pozitron elektron tomografi (*PET*), tekil foton yayılım hesaplamalı tomografi (*SPECT*), manyetik rezonans görüntüleme (*MRI*), işlevsel manyetik rezonans görüntüleme (*fMRI*), magnetoensefalografi (*MEG*) ve elektroensefalografi (*EEG*) yer alır.

BBA için günümüzde en geçerli sinyal alım tekniği *EEG*'dir. Saniyenin binde birinde yani her milisaniyede bir ölçüm yapabilme kabiliyeti sayesinde yüksek geçici çözünürlüğe sahiptir. Güncel *EEG* araçlarında aynı anda ölçüm yapabilen 256 elektrot bile bulunabilir.

*EEG*'nin laboratuvarındaki ve gerçek hayat koşullarındaki kullanılabilirliği önüne geçilemez bir haldedir. Cihaz taşınabilir ve elektrotlar deneğin kafatası üzerine kolayca yerleştirilebilir. Ayrıca, *EEG* sistemleri var olmaya başladığından beri çok fazla sayıda alanda yaygın bir kullanıma sahiptir. Son olarak, en önemlisi ise yöntemin istila etmemesidir.

Birçok *EEG* tabanlı BBA sistemleri Şekil 11'de ayrıntılı olarak gösterildiği gibi Uluslararası 10/20 Sistemi tarafından önerilen elektrot yerleştirme stratejisi kullanır. Daha iyi boyutsal çözünürlük için, 10/20 sisteminin önerdiği elektrotların arasındaki

boşluklara ilave elektrotlar yerleştiren *10/20* sisteminin bir başka türevinin de kullanılması da ayrıca mevcuttur.



Şekil 2.2 EEG ile istila etmeyen sinyal alımı için elektrotların yerleşimi (He, 2005)

Şekil 2.2’de elektrotları yerleştirme işleminde Uluslararası *10/20* Sisteminin kullanımı verilmiştir. Her bir elektrotu yazılan harfler özel bir alt lobu temsil eder (He, 2005, s. 85-95). Bu harfler ve anlamları aşağıda listelenmiştir.

- FP: Prefrontal Lobe
- F: Frontal Lobe
- T: Temporal Lobe
- C: Central Lobe
- P: Parietal Lobe
- O: Occipital Lobe.

Alt indiste gösterilen sayı, elektrotun beyin yarımküresi içindeki yerine işaret eder. Z, beynin tam anlamıyla orta çizgisini, tek sayılar sol yarımküre üzerinde bir yeri ve

ift sayılar ise sađ yarımküre üzerinde bir yeri sembolize ederler. Elektrotun orta çizgiye olan mesafesi yakınlaştıkça, sayısı azalır ve gitgide  $0$ 'a yaklaşır.



## BÖLÜM ÜÇ

### SİNYAL İŞLEME

Sinyalin fonksiyona dönüştürülebilir bir bilgi olması gerekmektedir. Örnek olarak, ses, sıcaklık, elektromanyetik dalga, gerilim, akım veya görüntüyü düşünebiliriz. İnsan vücudundan alınanlar dâhil, hangi tür sinyal olursa olsun, sinyaller geldiği kaynakla ilgili birçok önemli bilgi taşır. Sinyalleri doğru analiz etmek bize, kaynağın şu anki güncel durumu hakkında fikir verir ve bu sinyaller üzerine yapılan çalışmalar gelecek durumlar hakkında doğru tahminde bulunulmasına yardımcı olur.

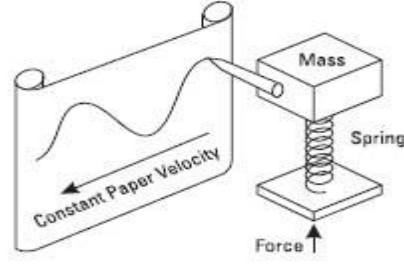
Sinyaller sabit ya da hareketli olabilir. Sabit sinyaller üzerinde çalışmak daha kolaydır. Çünkü yalnızca frekansları ve büyüklükleri vardır. Fakat hareketli sinyaller çoğunlukla beklendiği gibi gözlenmez. Bu tür sinyallerden alınan bilgiler daha değerli olur.

Sinyaller, zaman ve frekans olmak üzere iki ayrı alanda incelenip, ihtiyaç halinde birbirine dönüştürülebilir. Bu şekilde çalışmanın avantajı ise zaman perspektifinde anlaşılması güç olan güncel durum, frekans alanına çevrildiğinde çok net bir şekilde anlaşılabilir. Tam tersi durum da aynı şekilde kolaylık sağlayabilir.

#### 3.1 Zaman Alanı

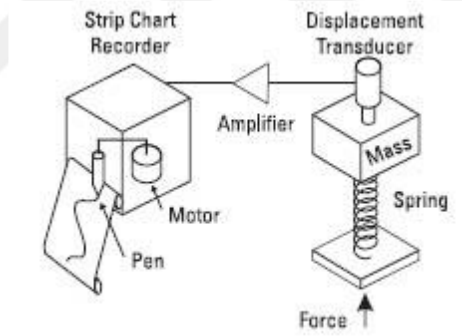
Sinyalleri gözlemlemenin geleneksel yoludur. Zamana bağlı olarak sistemdeki değişen parametreleri kaydetme stildir. Çevredeki değişken parametreleri elektriksel sinyallere dönüştürücüler (*transducer*) aracılığıyla çevrilir. Mikrofonlar, ivmeölçerler, basınçölçerler vs. bu dönüştürücülere örnektir.

Şekil 3.1’de, yaylı bir sistem kullanılarak sabit zamanda kütlenin, uygulanan kuvvete göre yer değiştirmesi kâğıt üzerinde sinyale aktarılmıştır. Yay aşağı hareket ettiğinde, kalem birim zamanda kâğıt üzerinde aşağıya doğru çizim yapar. Yay yukarı hareket ettiğinde ise bunun tam tersi durum gerçekleşir ve sinyal oluşmuş olur.



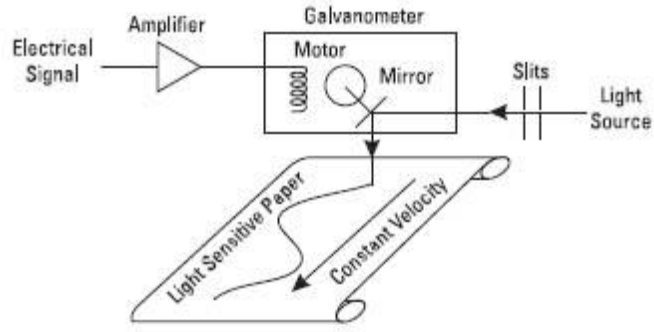
Şekil 3.1 Zaman alanında yer değıştirme kaydı (Agilent, 2000)

Elektriksel bir sinyal ise dolaylı olarak Strip Chart Recorder kullanılarak kâğıt üzerine aktarılabilir. Bu biçimdeki sinyal temsili Şekil 3.1’de gösterilmiştir. Dolaylı dönüştürücülerin kullanıldığı sistemlerin amacı, dış çevre faktörlerinin sinyalleri etkilemesine engel olmaktır. Bu faktörler arasında sürtünme, rüzgâr ve cismin kütlesi sayılabilir.



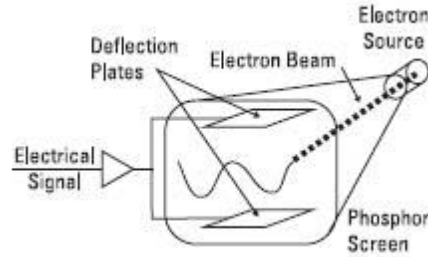
Şekil 3.2 Yer değıştiriminin dolaylı kaydı (Agilent, 2000)

Bu tür sistemlerin kullanışsız olduğu durum ise, cismin yer değıştirmesi kalemin yazım hızından yüksek olduğunda sinyalin hatalı üretilmiş olmasıdır. Çünkü kalemin yazım hızı sonludur. Bu sorunu gidermek için sistemden kalem çıkarılır ve ışığa duyarlı kâğıt yerleştirilir. Bu tür cihazlara osilograf denir. Şekil 3.2’de görüldüğü gibi galvanometre içindeki aynayı çok küçük açıyla döndürmek yeterli olacağından, *strip chart recorder*’a göre çok daha hızlı çizim yapılır.



Şekil 3.3 Basitleştirilmiş osilograf işlemi (Agilent, 2000)

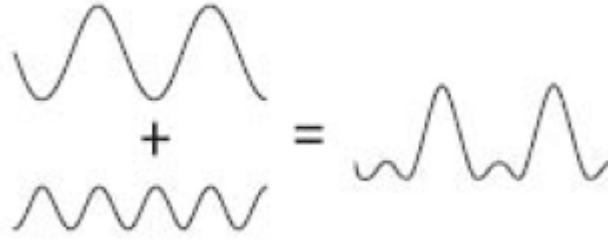
Zaman bölümlü sinyalleri göstermek için kullanılan diğer bir cihaz ise osiloskoptur. Burada, elektrik alanı kullanılarak bir elektron hareket ettirilir. Elektron fosforşıl maddeden yapılmış ekran üzerinde görünür hale gelir. Bu sistem sinyalleri, osilografıtan bile çok daha hızlı şekilde üretebilir. Bunun sebebiyle aynayı değil de yalnızca elektronu hareket ettirmenin gerekli olmasıdır. Elektronun hareketiyle oluşan sinyal Şekil 3.4’te temsil edilmiştir.



Şekil 3.4 Basitleştirilmiş osiloskop işlemi (Agilent, 2000)

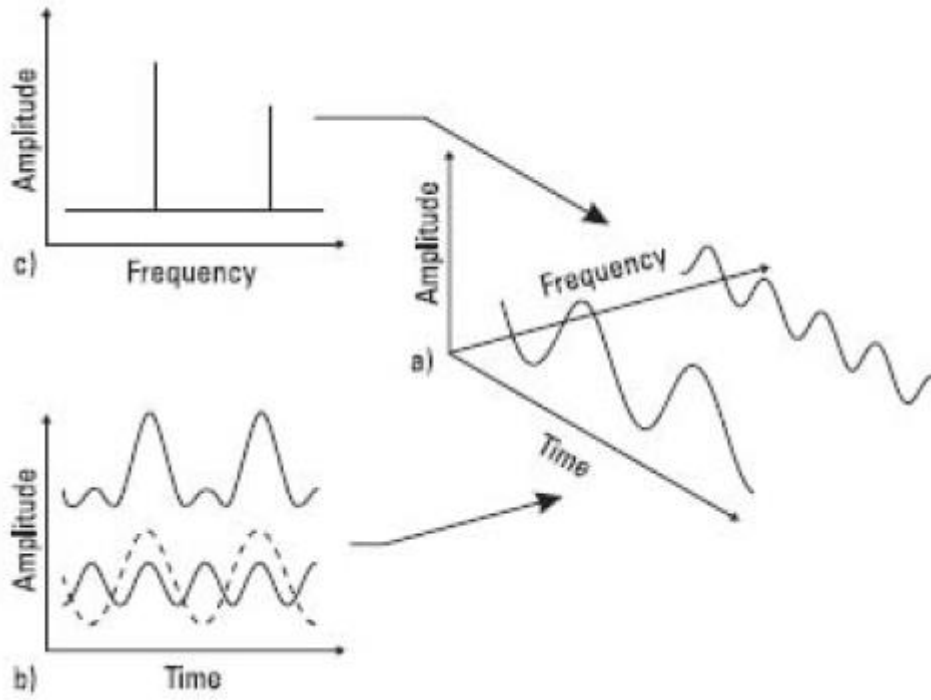
### 3.2 Frekans Alanı

100 yıl önce Baron Jean Baptiste Fourier tarafından gerçek dünyada var olan tüm sinyal dalgaları, sinüs dalgalarının birbirine eklenmiş hali olarak üretilebileceği ileri sürüldü. Şekil 3.5’te 2 sinüs dalgasının birbirine eklenmiş haliyle oluşan bir sinyal gösterilmiştir. Bu sinüs dalgalarının genliği, fazı ve frekansı doğru bir şekilde ayarlanarak istenilen sinyal üretilebilir. Aynı şekilde, dünya üzerindeki her sinyalde sinüs dalgalarına ayrıştırılabilir.



Şekil 3.5 Herhangi bir dalga sinüs dalgalarının birleşimiyle üretilebilir (Agilent, 2000)

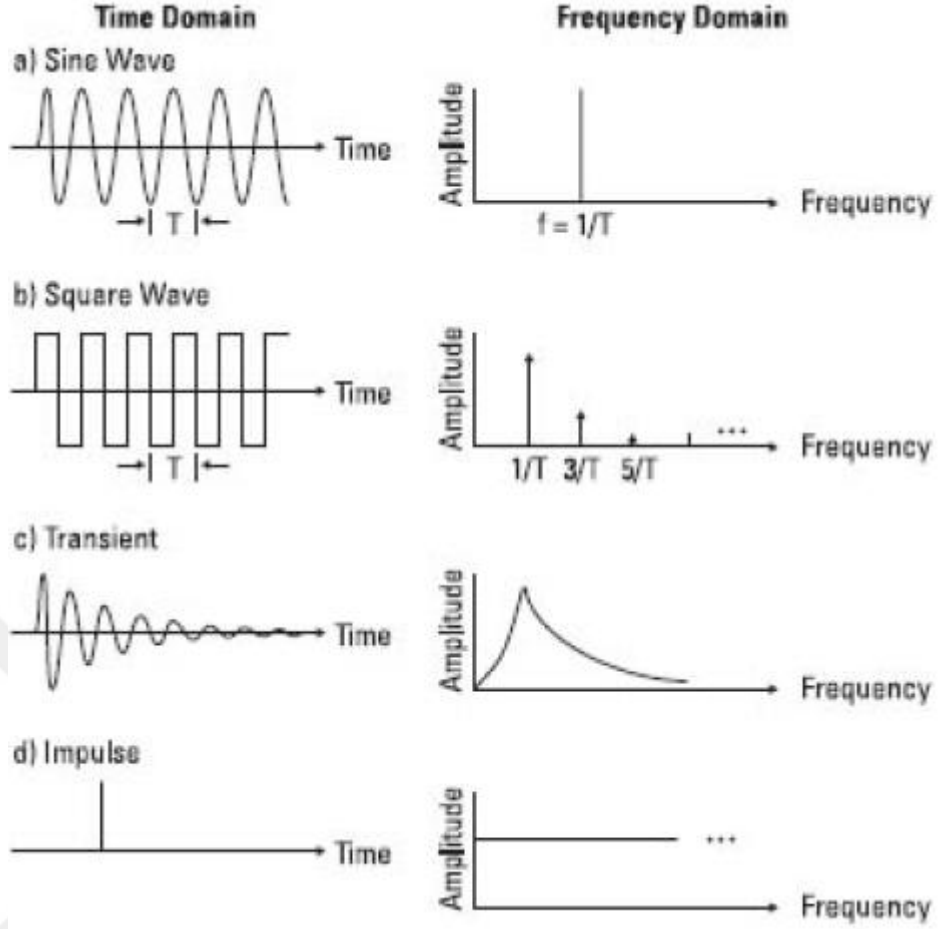
Şekil 3.6.a, bu sinüs dalgalarının birbirine eklenmiş halinin 3 boyutlu grafikte temsilini gösterir. Zaman alanından da tanıdık geleceği üzere, eksenlerin ikisi zaman ve genliktir. Üçüncü eksen ise karmaşık dalga boyunu üretmek için birbirine eklenen sinüs dalgalarını görsel olarak ayrılmasına olanak sağlayan frekanstır. Bu üç boyutlu grafiğe frekans ekseninden bakılarak Şekil 3.6.b görünümü elde edilir. Bu sinüs dalgalarının zaman alanı görünümüdür. Onların her birim zamanda bir birbirine eklenmesi bize asıl dalgayı verir. Fakat grafik Şekil 3.6.c de olduğu gibi zaman ekseninden bakıldığında, bambaşka bir görüntü ortaya çıkar. Burada zamana karşı frekansın yer aldığı grafik oluşur ve frekans alanı olarak adlandırılır. Girdiden ayrılmış her bir sinüs dalgası dikey çizgi biçiminde görülür. Boyu genliğini ve pozisyonu frekansını temsil eder. Her bir çizgi sinüs dalgasını temsil ettiğinden, girdi sinyali frekans alanında eşsiz bir şekilde tanımlanmıştır. Sinyalin temsilinin frekans alanı sinyalin spektrumu olarak adlandırılır. Spektrumun her bir sinüs dalga çizgisi toplam sinyalin bileşeni olarak adlandırılır.



Şekil 3.6 Zaman ve frekans alanları arasındaki ilişki (Agilent, 2000)

a) Zamanı, genliği ve frekansı gösteren 3 boyutlu koordinatlar. b) Zaman alanı görünümü c) Frekans alanı görünümü (Agilent,2000). Aynı 3 boyutlu grafiğe farklı açılardan bakılır. Bu farklı bakış açısı çok faydalı olabilir. Başlangıçta frekans alanı tuhaf ve farklı görünebilir fakat günlük hayatın önemli bir parçasıdır. Kulak-beyin kombinasyonu harika bir frekans alanı analizcisidir. Kulak-beyin ses spektrumunu dar gruplara ayırır ve her gruptaki güç varlığını belirler. Yüksek şiddette arka plan gürültülerinden küçük sesleri kısmen frekans alan kapasitesi sayesinde ayıklayabilir. Doktor anormallikleri yakalamak için hastanın kalbini ve nefes alışverişini dinler. Stetoskop kullanarak, üretilen frekans sayesinde ritimde bir tutarsızlık olup olmadığını duyabilir.





Şekil 3.7 Frekans spektrum örnekleri (Agilent, 2000)

Şekil 3.7’de sinüs dalgasının spektrumunun yalnızca bir çizgi olduğu görülür. Şekil 3.7.b’deki kare dalga sonsuz sayıda sinüs dalgasından oluşur. Şekil 3.7.c’deki dalganın sürekli bir spektrumu vardır. Şekil 3.7.d’deki impuls sinyalindeyse her frekansta enerji vardır.

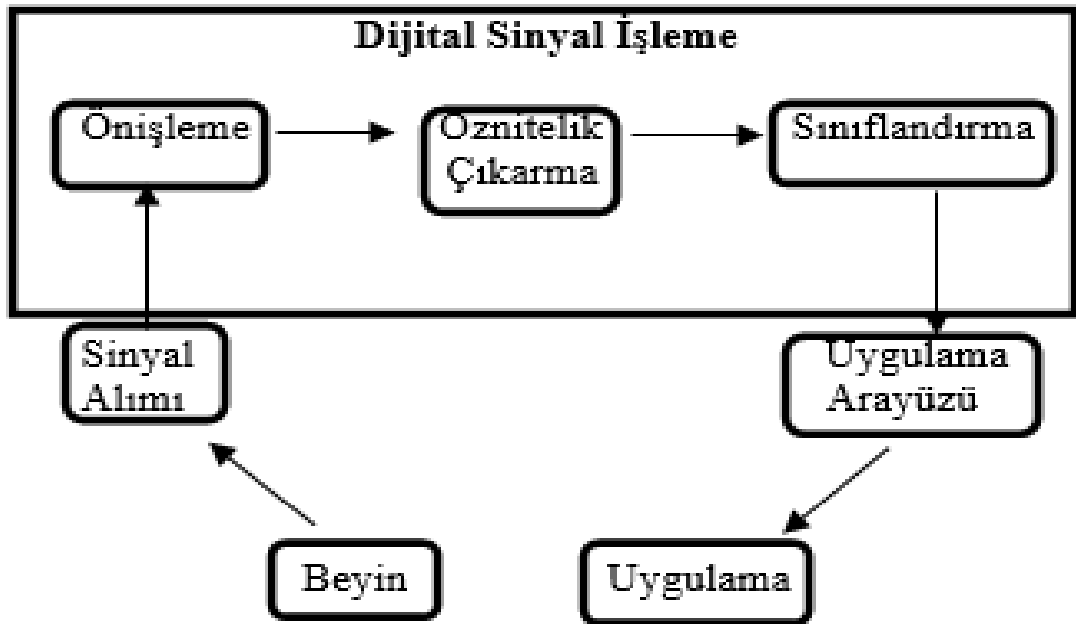
### 3.3 Beyin Bilgisayar Arayüzü Aşamaları

Beyin Bilgisayar Arayüzü (BBA) beyin ile fare imlecinin ya da protez kolun kontrolü gibi bazı harici aktiviteleri yönlendirmek için beyinden sinyal alımını sağlayan cihaz arasında gerçekleşen bir işbirliğidir. Arayüz, beyin ile kontrol edilecek nesne arasında doğrudan iletişim kurar. Örneğin, fare imleci kontrolü durumunda

beyinden fareyi üzerindeki parmağa vücudun sinir-kas (*neuromuscular*) sistemi üzerinden normal yolu takip etmek yerine sinyal beyin tarafından doğrudan iletilir.

Önişleme, BBA'nün temel süreçlerinden biridir. (Şekil 3.8) Beyinden alınan sinyaller, öznelilik çıkarım aşaması için hazır olmaya yeterli düzeyde değildir ve bu nedenle önişleme yapmak kaçınılmazdır. *EEG* sinyalleri yalnızca düşünme sürecinde üretilen nörolojik bilgi taşımakla kalmaz, aynı zamanda ilgilenmediğimiz kaynaklardan da veri taşır ve bu veriler gürültü olarak adlandırılır. Birçok gürültü kaynağından oluşan veri üzerinde işlem yapmak oldukça zordur. Bu sebepten ötürü, *EEG* sinyallerine karışmış gürültüler bir önişleme mekanizmasından geçirilerek yok edilmelidir.

*EEG* sinyallerinin genlikleri çok küçük olduğundan, gürültüler bu sinyaller üzerinde varlığını kolayca gösterir. Kolu soldan sağa hareket ettirmek, kalp atışları ve göz kırpmak gibi fiziksel hareketlerden ve lambadan ışık yayılan ışık ya da çevredeki elektromanyetik dalgalar yayan cihazların varlığı gibi dış etmenlerden etkilenmeye maruz kalırlar.



Şekil 3.8 BBA'nün temel işlevsel elemanları

Literatürde, kullanılan etkili birçok gürültü giderme yöntemleri mevcuttur. Onların arasında en yaygın olanları ise Bağımsız Bileşenler Analizi, Doğrusal ve Doğrusal Olmayan Filtreleme, Dalgacık Dönüşümü, Temel Bileşenler Analizi, Fourier Dönüşümleri ve Kaynak Dipol Analizi'dir. BBA uygulamalarındaki asıl gürültü kaynağı göz kırpmasından kaynaklanandır. Bu çeşit bir gürültü, özellikle gözlere yakın bulunan beyin arka loblarına yerleştirilmiş elektrotlarla tespit edilir (Alexandridis ve Chondrodima, 2014).

Gürültü giderme tekniklerinden farklı olan diğer önışleme operasyonlarıysa ortalama almak, eşik değeri belirlemek, normalizasyon yapmak ve sinyal iyileştirme yapmaktır.

Bir sonraki aşama olan öznitelik çıkarma, düşünceler farklılaştığında kaydedilen *EEG* sinyallerinin önemli özelliklerini belirleme işleminin ardından öznitelik elde etme operasyonudur. Örneğin, ellerini normal bir şekilde kullanamayan engelli bir birey bilgisayardaki imlecin pozisyonunu değiştirmeyi düşündüğünde, *EEG* sinyallerinin nasıl şekil aldığı bir öznitelik çıkarma işlemi haline gelir.

Oluşturulmuş her bir öznitelik vektörü bir sinyali tanımlayabilir fakat sınıflandırma aşamasında hiçbiri %100 doğru yanıt vermeyebilir. Bu nedenle, yüksek başarı oranı yakalamak için farklı özellikler bir arada kullanılmalıdır.

Bir diğer önemli nokta ise, çoklu elektrotlar aracılığıyla kaydedilen *EEG* verisi üzerinde analiz yapılırken öznitelik çıkarmak için hangi elektrotun daha uygun olacağına karar vermektir. Beynin farklı bölümleri farklı tipte görevler için daha hassastır. Örneğin, duyma duyuları beyin arka lobunda işlenir. Sonuç olarak, tüm beyinsel parçaları kaplayan çoklu elektrotlardan gelen bütün veriden faydalanmak yerine, beyinsel aktivitenin çok daha yoğun olduğu beyin kısımlarından alınan veriyi kullanmak daha iyi bir fikir olur. Çünkü bu durum sınıflandırma aşamasında karar vermek için harcanan zamanı kısaltır ve sınıflandırma için çıkarılacak en iyi özniteliği bulmaya olanak sağlar (He, 2005).

Farklı görsel ve zihinsel durumlarda kaydedilen *EEG* sinyallerini, öznitelik vektörlerine göre sınıflara ayırmak BBA'nde önemli bir konudur. Hangi sinyalin hangi sınıfa ait olması gerektiği kararı hızlı ve doğru olmalıdır. Sinyalin hangi sınıftan olduğuna karar verme işlemi sınıflandırma olarak adlandırılır. Literatürde birçok farklı sınıflandırma yöntemleri vardır. Onların en yaygın olarak kullanılanları arasında Destek Vektör Makineleri, k-En Yakın Komşuluk, Doğrusal Diskriminant Analizi, Yapay Sinir Ağları ve Bayes Sınıflandırma gibi algoritmalar bulunur.

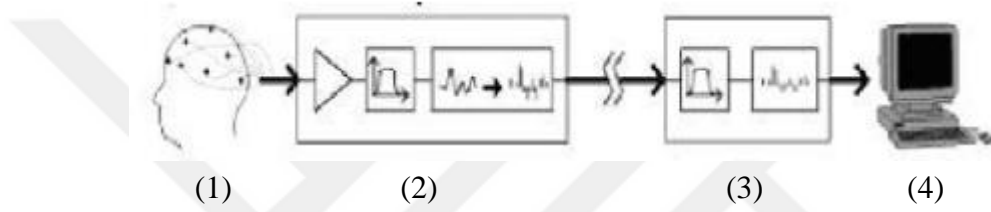


## BÖLÜM DÖRT

### KULLANILAN EKİPMANLAR VE YÖNTEMLER

#### 4.1 Yararlanılan Malzeme, Donanım ve Yazılımlar

Günümüz teknolojisinden faydalanılarak önceleri zahmetli olan *EEG* sinyali alım ve analiz işlemleri artık daha kolaylıkla ve daha kısa sürelerde gerçekleştirilebilir hale geldi. Şekil 4.1’de de gösterildiği gibi, beyin ve bilgisayarın iletişimini sağlayacak köprü bir mekanizma vasıtasıyla *EEG* kaydı alınıp istenilen formata dönüştürülebilir.



Şekil 4.1 Beyin Bilgisayar Arabirimi

Yapılacak ilk işlem beyin ilgilenilen kısımlarında gerçekleşen elektriksel potansiyel farklarını ölçmek için kafatasına elektrotlar yerleştirmektir (Şekil 4.1.1). Daha sonra henüz anlamsal değerler taşımayan *EEG* verilerinin birtakım filtreleme işlemlerinden geçirilerek yorumlanabilir hale getirilmesi gerekir (Şekil 4.1.2). Ardından üzerine işlem uygulanmış verilerin bilgisayarın giriş birimlerinden(portlarından) algılanabilmesi için uygun hale dönüştürülmesi gerekir (Şekil 4.1.3). Son olarak, toplanan bu sinyaller kayıt işlemi için bilgisayara gönderilir ve yazılımın arayüzünden okunur.

#### 4.1.1 Malzemeler

**Elektrotlar:** Kafatasına yerleştirilen 14 kanallı, yani 14 farklı noktada elektrot içeren, cihazın her bir elektrot ucunda tuzlumsu nemli sensörler yer alır. Bu sensörlerin yer aldığı kap ve uçlarına takılan pedler Şekil 4.2’de gösterilmektedir. Elektrot uçlarındaki bu sensörler elektriksel aktiviteyi tespit edebilir. Elektrot uçları kurduğunda cihazın kendine özgü sıvısıyla tekrar nemlendirilir.



(a)



(b)

Şekil 4.2 (a) Elektrot uçlarındaki sensör sıvısı ve (b) sensör pedleri (Emotiv, 2016)

Beyinden sinyal alımı yapmak amacıyla *Emotiv Epoc+ 14 Channel* isimli baş üzerine yerleştirilebilen portatif bir cihazdan yararlanırız. Kendisine ait *USB* bellek aracılığıyla alınan sinyalleri bilgisayar ortamına taşıyoruz. Cihaz herhangi bir kablo içermemektedir. Şekil 4.3, cihazın fotoğrafını yansıtmaktadır.



Şekil 4.3 Emotiv Epoc+ 14 Channel EEG kayıt cihazı (Emotiv, 2016)

#### 4.1.2 Donanımlar

*EEG* sinyallerini analiz etmek ve sınıflandırmak için 1 adet masaüstü bilgisayar kullanılmaktadır. Tüm ön işleme ve sınıflandırma işlemleriyle görsel arayüz oluşturma bu bilgisayar ile yapılmıştır.

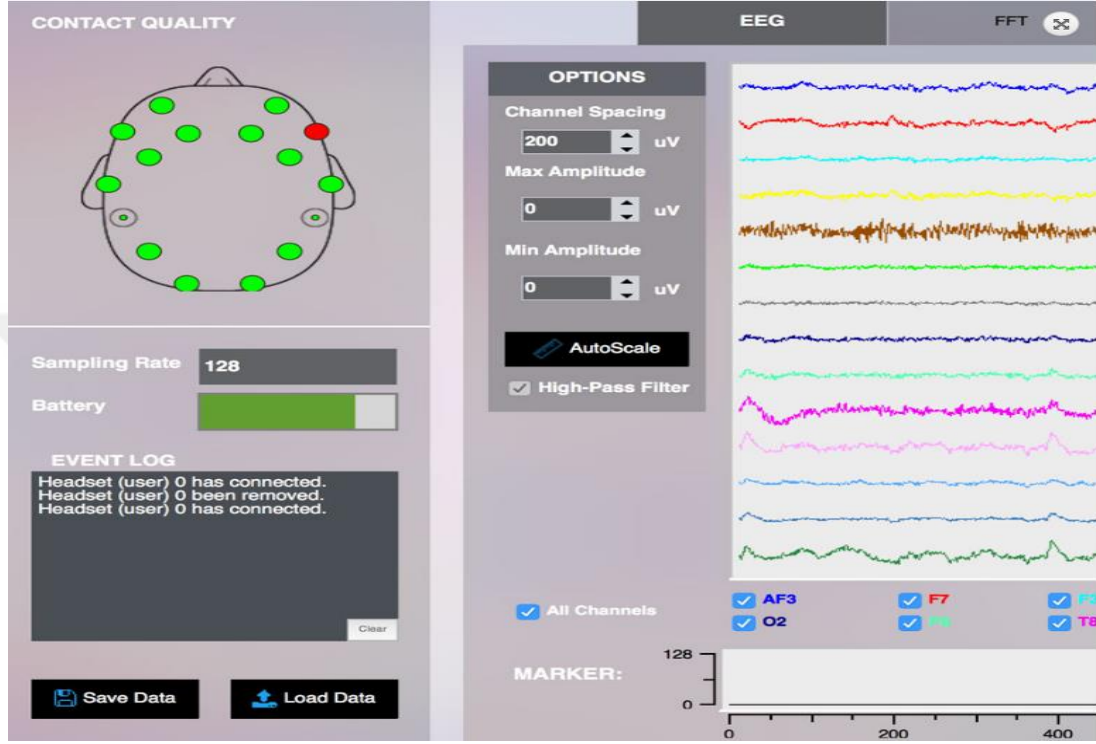
Kullanılan masaüstü bilgisayarın sistem özellikleri:

- *Windows 8.1 Pro* İşletim Sistemi
- *Intel® Core™ i5-4590 CPU @ 3,30GHz* İşlemci
- *8,00 GB RAM* ve *500 GB* Sabit Disk

#### 4.1.3 Yazılımlar

Emotiv cihazıyla alınan *EEG* sinyallerini kullanabilme hakkı elde etmek ve bilgisayara taşımak amacıyla Emotiv Pure *EEG* Raw *EEG* Data yazılımı kullanılmaktadır. Bu yazılımın dinamik arayüzüne ait ekan görüntüsü Şekil 23 de gösterilmektedir. Ayrıca verilerin istenilen formata dönüştürüldükten sonra yapılacak

tüm işlemlerde *Visual Studio 2015 Community* programından ve *C#* programlama dilinden faydalanılmaktadır. Ayrıca kodladığımız sınıflandırma algoritmasının elde ettiği sonuçları diğer popüler sınıflandırma algoritmalarının sonuçlarıyla karşılaştırmak için *Weka* veri madenciliği yazılımını kullanılmıştır.



Şekil 4.4 Emotiv Pure EEG yazılımının uygulama arayüzüne ait görüntü (Emotiv, 2016)

## 4.2 Veriseti

Verisetimiz, farklı deneylere tabi tutulmuş aynı denek tarafından alınan *EEG* sinyallerinden oluşur. Doktora öğrencisi Sezin Tunaboşlu, sınıflandırma algoritmalarımızı üzerinde çalıştırıp test edebilmemiz için bu verilerini bize tahsis etmiştir. Onun deneylerinde, aynı denek 4 farklı oturumdaki deneylere maruz kaldı. Bu oturumlarda bilgisayar arayüzünde çıkan sağ, sol, yukarı ve aşağı olmak üzere 4 farklı yöne sahip okun gösterdiği yönü hayal etmesi istendi. Bir başka deyişle, zihninde canlanan herhangi bir şeyin okun yönüne doğru hareket ettiğini imgelemesi istendi. Ok ekranda her çıktığında rastgele bir yönü göstermiş olup 3 saniye boyunca ekranda



kalmıştır. Her bir oturumda ok 4 yönden yalnızca birini göstermiştir ve aynı yön tekrar etmemiştir.

Bir oturumun tamamlanmasının ardından, *EEG* sinyalleri bir metin dosyasına kaydedilmiştir. Bu kayıta her bir elektrotta kendisinde gözlemlenen o anki elektriksel potansiyel farkı cinsinden değerler yer alır. Bu veri aktarımını deneğin kafasına yerleştirilen *EEG* sinyal alım cihazı mümkün kılmıştır. Emotiv cihazının donanımsal parçalarından biri olan “*Dungle*” adı verilen harici bir *USB* bellek vasıtasıyla sinyal alım cihazı ve bilgisayar arasında köprü oluşturulmuştur. Kablo kullanılmamıştır.

*EEG* verisini alan 14 farklı elektrot vardır. Elektrotlardaki veriler aşağıdaki sırayla kaydedilmiştir:

*AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, AF4*. Bu elektrot kanallarının adlandırılması ve kafatası üzerindeki yerleşimi cihaz tarafından daha önce bahsedilen *10/20* sistemine uygun olarak yapılmıştır.

Bunlardan başka cihaz tarafından denekle ilgili harici olarak kaydedilen başka veriler de bulunur. Sınıflandırma yaparken göz önüne almadığımız bu özellikler ise şunlardır:

*COUNTER, INTERPOLATED, RAW\_CQ, GYROX, GYROY, TIMESTAMP, ES\_TIMESTAMP, FUNC\_ID, FUNC\_VALUE, MARKER, SYNC\_SIGNAL*.

### 4.3 Veri Önışleme

Her şeyden önce, verisetini öğrenim kümesi ve test kümesi olmak üzere 2 bölüme ayırmak gerekti. Arayüzdeki okun ekranda gösterdiği yön ile isimlendirilen 4 farklı sınıf oluşturduk. Bu sınıflar “yukarı”, “aşağı”, “sağ” ve “sol” du. Aynı denekten alınan her bir sınıf için 4 dosya içinde veriler elimizde bulunuyordu. Her bir dosyadaki veri miktarı 1280 satırdı ve biz bunların elektrot kanallarındaki değerlerin tutulduğu 14 sütunıyla ilgilendik. Bir başka deyişle,  $4 \times 1280 \times 14 = 71680$  veri üzerinde çalıştık. Bu verileri %75 öğrenim seti, %25 test kümesi olmak üzere ayırdık. Daha sonra sınıflandırma işlemini zorlaştırmak adına bu oranı %67 - %33 olarak değiştirip

yeniden uygulama yaptık. Her bir sınıf için 4 dosyadaki, toplamda 16 dosyadaki bütün verileri birleştirdik. Bu işlemi yaparken her bir sinyalin hangi sınıfa ait olduğu bilgisini kaybetmemek adına sinyallere 15. özellik olarak sınıf etiketi (class label) özelliğini ekledik. Böylece bütün  $4 \times 4 \times 1280 = 20480$  veri aynı dosyada toplanmış oldu. Ardından bu veriler içinde yanlış kaydedilmiş olan ve ilgili özellik değeriyle ilgisi olmayan değer içeren bozuk verileri ayıklamak gerekti. Bu verileri ayıkladıktan sonra elimizde tüm sınıflardan toplam 19600 veri kaldı. Yani  $20480 - 19600 = 880$  adet üzerinde analiz yapılamayacak sinyali elemiş olduk. Verilerin dosyaya düzenli formatta yazılmış hali Tablo 4.1’de, bozuk veri temsili örneği düzgün ve üzerinde işlem yapılabilir temiz verilerin sınıflara göre dağılımı Tablo 4.2’de verilmiştir.

Tablo 4.1 Dosyaya kaydedilmiş EEG veri sinyallerinin görünümü

COUNTER	INTERPOLATED	AF3	F7	F3	FC5	T7	P7	O1	O2	P8	T8	FC6	F4	F8	AF4
13	0	4329	76,27368421	3925	-97,06315789	4229	236,2736842	4379	-324,7578947	3956	-201,6842105				
14	0	4322	69,08421053	3924	-103,7263158	4226	236,2736842	4377	-321,1684211	3959	-197,0631579				
15	0	4322	71,65263158	3919	-103,2210526	4226	231,1368421	4371	-324,2421053	3966	-188,3473684				
16	0	4325	82,93684211	3924	-92,95789474	4233	231,1368421	4374	-322,7052632	3970	-185,7789474				
17	0	4321	81,90526316	3926	-91,93684211	4232	234,2210526	4375	-320,6526316	3966	-193,9894737				
18	0	4319	74,22105263	3923	-91,42105263	4230	235,7578947	4375	-317,0631579	3965	-200,6526316				
19	0	4324	83,45263158	3929	-86,29473684	4242	238,3157895	4381	-307,8315789	3969	-196,5473684				
20	0	4329	91,13684211	3939	-85,77894737	4248	241,4 4384	-308,3473684	3972	-183,7263158	4347				
21	0	4330	81,90526316	3936	-84,75789474	4239	243,4526316	4381	-318,6 3973	-173,4736842	4346				
22	0	4327	81,90526316	3927	-86,29473684	4239	243,9578947	4383	-322,7052632	3966	-175,5263158				
23	0	4325	86,01052632	3924	-93,98947368	4238	241,4 4383	-323,2210526	3958	-185,2736842	4348				
24	0	4323	71,65263158	3922	-97,06315789	4225	236,2736842	4375	-328,3473684	3957	-190,9157895				
25	0	4317	63,45263158	3921	-95,01052632	4222	235,2421053	4373	-328,3473684	3957	-190,4 4348				

İlk satır her bir sütundaki özellikleri gösterir. Veriler birbirinden *tab* karakteriyle ayrılır

Tablo 4.2 Düzgün veri örnekleri

COUNTER	INTERPOLAF3	F7	F3	FC5	T7	P7	O1	O2	P8	T8	FC6	F4	F8	AF4	RAW_CQ	CQ_AF3	CQ_F7	
13	0	4329	76,27368	3925	-97,0632	4229	236,2737	4379	-324,758	3956	-201,684	4347	253,7053	4267	-87,8316	448	4	4
14	0	4322	69,08421	3924	-103,726	4226	236,2737	4377	-321,168	3959	-197,063	4349	248,5789	4271	-90,4	467	4	4
15	0	4322	71,65263	3919	-103,221	4226	231,1368	4371	-324,242	3966	-188,347	4350	241,9053	4273	-95,5263	463	4	4
16	0	4325	82,93684	3924	-92,9579	4233	231,1368	4374	-322,705	3970	-185,779	4349	246,5263	4274	-93,4737	0	4	4
17	0	4321	81,90526	3926	-91,9368	4232	234,2211	4375	-320,653	3966	-193,989	4348	247,0421	4269	-92,9579	0	4	4
18	0	4319	74,22105	3923	-91,4211	4230	235,7579	4375	-317,063	3965	-200,653	4349	240,8842	4265	-93,6316	0	4	4
19	0	4324	83,45263	3929	-86,2947	4242	238,3158	4381	-307,832	3969	-196,547	4349	248,0632	4271	-93,4737	0	4	4
20	0	4329	91,13684	3939	-85,7789	4248	241,4	4384	-308,347	3972	-183,726	4347	260,8842	4266	-76,5474	0	4	4

Tablo 4.2 Bozuk veri örnekleri

COUNTER	INTERPOLAF3	F7	F3	FC5	T7	P7	O1	O2	P8	T8	FC6	F4	F8	AF4	RAW_CQ	CQ_AF3	CQ_F7
52	0	4313	116,2737	-82,7053	4259	218,3158	4382	3962	-205,274	4345	269,0842	-63,2211	429	4	4	4	4
53	0	4325	118,3158	-82,7053	4256	221,4	4377	3962	-206,295	4344	273,1895	-59,6316	446	4	4	4	4
54	0	4336	111,1368	-80,6526	4244	221,9053	4374	3966	-205,779	4345	267,5474	-52,9579	470	4	4	4	4
55	0	4335	109,0842	-74,5053	4253	220,3684	4376	3970	-193,474	4346	264,9895	-45,7789	463	4	4	4	4
56	0	4339	106,0105	-70,4	4270	227,0421	4375	3971	-176,547	4347	263,4526	-46,2947	0	4	4	4	4
57	0	4347	97,29474	-72,4526	4266	231,6526	4371	3971	-178,6	4347	254,2211	-58,6	448	4	4	4	4
58	0	4345	96,27368	-76,0421	4255	225,4947	4371	3973	-195,011	4347	245,4947	-67,8316	470	4	4	4	4

Tablo 4.3 Ayıklama sonrası elde kalan tüm verilerin sınıflara göre dağılımları

	Sinyal Sayıları
Yukarı	4800
Sol	5200
Sağ	4800
Aşağı	4800

Tablo 4.3 ve Tablo 4.4'e bakarak bozuk verilerin düzgün verilerden nasıl farklılaştığını anlayabiliriz. İlgili tablolarda verilen örnekler incelendiğinde görülüyor ki *F3* özelliğinden itibaren *AF4* özelliğine kadar olan ki tüm sütunlarda değer kayması yaşanmış. Yani, sol sütunla ilgili özellik değeri sağ sütuna kaydırılarak yazılmış ve bazı sütunlardaki *EEG* değerleri kayıp olmuştur. Örneğin, *F8* özelliğine ait değerlerin genelde 4 basamaklı sayılar olduğu görülürken, bu haneye *CQ\_AF3* özelliğine ait olan

4 değeri veri kaydı yapılırken kaydırılmıştır. Bu tablo dışında soldan sağa doğru kayan veriler ya da tüm özelliklerine değeri atılamayan 0 yazılan veriler de ayıklanan sinyallerden bazılarıdır. Bu tür veriler bize analizde katkı sağlayamaz, aksine veriseti hakkında yanlış tahminlerde bulunmamıza sebep olabilir. Bu yüzden onları eledik. Tablo 4.4'te gösterildiği gibi veriler sınıf etiketlerine ve ait oldukları kümelere göre gruplara ayrılmıştır. Sol sınıfına ait verilerin kısmen daha fazla diğer sınıflara ait verilerin eşit olduğu fakat miktarların birbirine çok yakın olduğu görülür. Bu işlemden sonra verilerin normalizasyonunu yaptık. Normalizasyon yöntemi olarak min-max normalizasyonunu tercih ettik.

Tüm verileri sınıflandırmak için Nasibov tarafından önerilen  $C \times k$  – En Yakın Komşuluk Algoritması'ndan yararlandık. Bu algoritmanın işleyiş işleyişi hakkında bilgi vermekte fayda var.

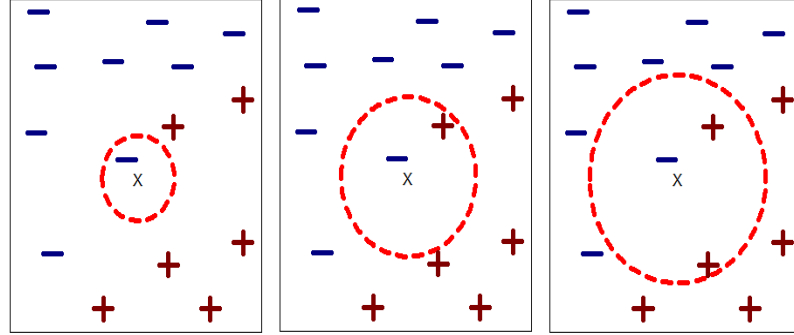
#### 4.4 $k$ – En Yakın Komşuluk Algoritması

$k$  – En Yakın Komşuluk algoritması kayıt tabanlı öğrenim tarafından desteklenen sınıflandırma algoritmaları içerisinde en yaygın kullanılanlardan bir tanesidir. Eğer sınıflandırılacak veriseti içindeki tüm veriler kayıt tabanlı öğrenim için uygunsa, bu durumda  $k$  -  $NN$  algoritmasından faydalanmak akıllıca olur. Henüz sınıfı bilinmeyen bir sinyal ya da herhangi bir değeri,  $k$  adet en yakın komşusuna bakılarak bir sınıfa dahil edilir. Eğer  $k$ ,  $I$ 'e eşitse, o halde bu tekil komşunun sınıfı otomatik olarak sinyalin ait olduğu sınıf haline gelecektir.

Gereklilikler:

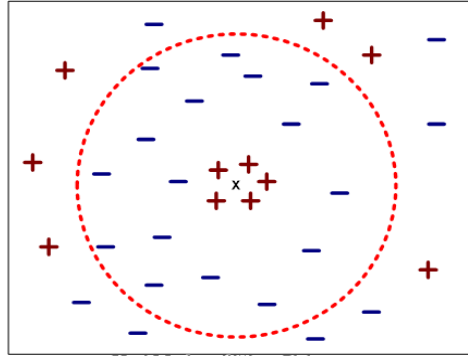
- Antrenman (Öğrenim) Kümesi
- Kayıtlar arasındaki uzaklığı ölçmeye yarayan Uzaklık Ölçü Birimi
- En yakın komşu sayısı ( $k$ )
- Test kayıt nesnesi ile öğrenim kümesindeki tüm elemanlar arasındaki uzaklıkların hesaplanması
- $k$  – en yakın komşu sayısının belirlenmesi

- $k$  adet en yakın komşu sinyaller içindeki çoğunluğun ait olduğu sınıfa atama işleminin gerçekleştirilmesi



Şekil 4.5 İkili sınıflandırma sırasında  $k = 1, 2$  ve  $3$  değerleri için  $k - NV$  ile sınıf belirleme örneğinin görselleştirilmesi

Bu sınıflandırma yaklaşımında, yapılacak en kritik işlem  $k$  değerinin seçilmesidir.  $k$ 'yı seçmenin mantıklı ve aşamalı bir yöntemi yoktur. Çünkü, farklı  $k$  değerleri farklı çeşitte verisetleri üzerinde değişik etkiler oluşturabilir. Bu yüzden, üzerinde işlem yağılan verisetine bağlı olarak değişir. Farklı  $k$  değerlerine göre en yakın  $k$  adet komşu belirleme işlemi Şekil 4.5'te gösterilir. Sonuç olarak, sezgisel olarak denenen farklı  $k$  değerleri test edilir ve bu karşılaştırmanın sonuçlarına göre elde edilen başarı oranları kıyaslanır. En iyi başarı oranını yakalayan  $k$  değeri  $k$  seçimi için en ideal aday haline gelir. Eğer  $k$  değeri çok küçük bir sayı seçilirse, o halde sınıflandırma gürültülerden ve sapan değerlerden çok fazla etkilenir. Aksi durumda ise, yani  $k$  değeri çok büyük bir sayı seçildiğinde, bu durumda bölgesel özellikler kaybedilebilir. Bir başka deyişle, veri sınıfı, çok büyük bir aralık içinden çok fazla sayıda komşu elemanı gözeterek hesaplanır ve analiz edilen test elemanına çok yakın komşular bir sınıfa aitken, kısmi olarak uzak olan komşular bir başka sınıfı temsil edebilir. Böylece, incelenen test elemanının sınıfı çevresindeki uzak komşu nokta ların sınıfı tarafından belirlenmiş olur ve bölgesel olarak en yakında bulunan komşu nokta sınıflandırma işlemine yeterince etki edememiş olur. Şekil 4.6'da noda uzak olanların sınıfı belirlediği açıkça görülür.



Şekil 4.6  $k$  seçiminin çok yüksek bir değer ile yapılmasının yarattığı etkiyi gösteren basit bir görsel

$k$  -  $NN$  sınıflandırma yaklaşımının avantajları ve dezavantajları vardır. Bir yandan, modeli oluşturmak kolay ve yüksek hesaplama maliyeti gerektirmez.  $k$  -  $NN$ . Hiçbir karmaşıklık yaratmayan uygulaması basit bir algoritmadır. Son olarak, çoklu sınıflandırma verileri için oldukça uygundur. Diğer bir taraftan ise  $k$  değerinin seçiminin önceden belirlenmiş bir yöntemi yoktur. Sezgisel olarak ya da testlere tabi tutularak belirlenir. Buna ek olarak, veriseti statik değilse, yani bir başka deyişle verisetine yeni kayıtlar ekleniyorsa, bu durumda onların herbirinin test noduna uzaklığı da hesaplanmalıdır.

#### 4.5 $C \times k$ – En Yakın Komşuluk Algoritması

$C \times k$  – En Yakın Komşuluk yaklaşımı iyi bilinen  $k$  – En Yakın Komşuluk ( $k$ - $NN$ ) algoritmasının geliştirilmiş bir versiyonudur. Bu yaklaşımda, sınıfı henüz belli olmayan değer her bir sınıftan  $k$  adet en yakın komşusuna olan uzaklıkları göz önüne alınarak bir sınıfa yerleştirilir. İlk olarak, tüm elemanlar arasındaki tüm uzaklıklar matematiksel olarak hesaplanır. Böylelikle, bir elemanın diğer bir elemana ne kadar uzak ya da ne kadar yakın olduğu açık hale gelir. Belirli bir sınıfı olmayan vektöre her bir sınıftan  $k$  adet en yakın komşu, toplamda  $C \times k$  adet eleman, analiz edilir. (Ulutagay ve Nasibov, 2016)  $C$  harfi burada sınıfların sayısını temsil eder (Ulutagay ve Nasibov, 2012). Algoritma ismini bu ifadeden alır. Sınıflandırılacak bir vektör ile bu vektöre en yakın  $k$  adet komşu vektörler arasındaki uzaklıkların ortalaması hesaplanır (Ulutagay ve Nasibov, 2009). Bunun sonucunda, henüz bariz bir sınıfı olmayan eleman ile her

bir sınıf arasındaki ortalama uzaklıklar elde edilmiş olur. Bu ortalama uzaklıklar içinden, en yakın uzaklığa sahip sınıf sınıflandırılacak elemanın yeni sınıfı olarak etiketlenir (Nasibov, Özgören, Ulutagay, Öniz ve Kocaaslan, 2010).

Varsayalım ki,  $X = \{x_1, x_2, \dots, x_n\}$   $n$  adet etiketlenmiş örneklerin kümesi,  $C_j = \{x_1^j, x_2^j, \dots, x_{n_j}^j\}$   $j = 1, 2, \dots, C$  önceden bilinen sınıflar,  $n_j$  ise  $n_1 + n_2 + \dots + n_c = n$  olduğu  $C_j$  sınıfındaki elemanların sayısı olsun.  $C \times k$  – En Yakın Komşuluk algoritmasının taslak kodu Şekil 4.7’de verilir:

### **BAŞLA**

Sınıflandırılmamış verinin girdisi  $x$ ,

$k$  ‘yı,  $1 \leq k \leq n$  aralığında seç

### **HER BİR $C_j$ SINIFI İÇİN,**

**OLANA KADAR YAP** ( $C_j$  sınıfında bulunan  $k$  en yakın komşu )

$i = 1$  yap.

$x$  ve komşu arasındaki uzaklığı hesapla

### **EĞER ( $i \leq k$ ), O ZAMAN**

$x_i^{c_j}$  ‘i  $C_j$ ’deki  $k$  en yakın komşuların kümesine ata.

### **EĞER ( $x_i^{c_j}, y$ ’ye önce bulunan komşudan yakınsa) O ZAMAN**

$k$  en yakın komşular kümesindeki en uzak nodu sil.

$x_i^{c_j}$  ‘i  $C_j$ ’deki  $k$  en yakın komşular kümesine ata.

### **KOŞULU BİTİR.**

$i = i + 1$

### **DÖNGÜYÜ BİTİR.**

$C_j$ 'deki  $k$  – en yakın komşular daki  $x$ 'in  $d_j$  ortalama uzaklığını hesapla.

### **DÖNGÜYÜ BİTİR.**

En yakın uzaklığa sahip sınıfı  $d_x = \min_j d_j$ ;

$x_i$  en son bulunan minimumun sınıfı  $r$  ' ye ata.

### **BİTİR.**

Şekil 2.7 C x k -NN algoritmasının taslak kodu (Nasibov ve Ulutagay, 2010)

**Aşama 1:** Çapraz doğrulama ve test işlemleri sırasında her bir sinyalin birbirine olan uzaklığını hesapladık. Uzaklık hesabında *Öklid*, *Cosine Benzerliği*, *Bray-Curtis* ve *Hellinger* uzaklık ölçümünden yararlandık. Bu uzaklık ölçüm yöntemlerinin formülleri verilmiştir ancak yine de kısaca bilgi vermek gerekirse Öklid uzaklık ölçümü için kullanıldığından ve *Bray-Curtis* ve *Hellinger* ise aksi benzerlik ölçümü yaptığından ötürü bu metodları kullandığımız kısımlarda sinyallerin birbirine yakınlığına baktık ve uzaklık değeri küçük olan sinyalleri ele aldık. Ancak, *Cosine Benzerliği* bir benzerlik ölçüm yöntemi olduğundan dolayı bu işlem sırasında benzerlik değeri elde ettik ve bu değerleri incelenmiş sinyallerden küçük değere sahip olanları yeni sınıf bulmak üzere inceledik.

#### **a-) Öklid Uzaklığı Formülü**

$$d_{Euc}(x, y) = \sqrt{\sum_{i=1}^n (X_i^2 + Y_i^2)} \quad (4.1)$$

#### **b-) Cosine Benzerliği Formülü**

$$\text{Benzerlik} = \cos(\theta) = \frac{x \cdot y}{\|x\| \|y\|} = \frac{\sum_{i=1}^n X_i Y_i}{\sqrt{\sum_{i=1}^n X_i^2 \sum_{i=1}^n Y_i^2}} \quad (4.2)$$

#### **c-) Bray-Curtis Uzaklığı Formülü**



$$d_{BC}(x, y) = \frac{\sum_{i=1}^n |X_i - Y_i|}{\sum_{i=1}^n X_i + Y_i} \quad (4.3)$$

#### d-) Hellinger Uzaklığı Formülü

$$d_H(x, y) = \frac{\sqrt{\sum_{i=1}^n (\sqrt{X_i} - \sqrt{Y_i})^2}}{\sqrt{2}} \quad (4.4)$$

**Aşama 2:** Çapraz doğrulama sırasında öğrenim kümesinin test parçasındaki incelenecek sinyale uzaklığına göre öğrenim kümesinin test parçası dışındaki tüm parçalarındaki sinyalleri kendi sınıfı içinde, test sırasında test kümesindeki incelenecek sinyale uzaklığına göre öğrenim kümesindeki tüm sinyalleri kendi sınıfı içinde sıraladık. Bir başka deyişle, bu sıralama sonucunda test edilecek sinyale en yakın sinyaller kendi sınıfı içinde en ön sıralara, en uzak sinyallerse en son sıralara yerleşti.

**Aşama 3:** Öğrenim kümesindeki her sınıfa ait verileri sıraladıktan sonra bu sıralanmış sinyaller arasından belirlenen  $k$  sayısı tane en yakın sinyallere eriştik. Toplamda 4 sınıfımız olduğundan  $4 \times k$  tane sinyale erişmiş olduk. Algoritmadaki  $C$  ifadesi sınıf sayısını temsil eder.  $C \times k$  ifadesi de ismini bu sayılardan alır.

**Aşama 4:**  $k$  tane en yakın komşular ve test kümesindeki sınıflandırılacak sinyal arasındaki uzaklıkların ortalamasını aldık. Örneğin,  $k = 3$  olduğu durumda, yukarı, aşağı, sol ve sağ sınıflarındaki en yakın 3'er komşunun uzaklıklarını toplayıp her sınıf için toplam değerleri 3'e böldük.

**Aşama 5:** Ardından, bu ortalama değerleri karşılaştırdık ve bu ortalamalardan en küçüğünün yer aldığı sınıfı test kümesinde incelenen sinyalin tahmin edilen sınıfı olarak atadık.

**Aşama 6:** Algoritma tarafından tahmin edilen sınıf ile test kümesinde incelenen sinyalin gerçek sınıfını karşılaştırdık. Bu iki sınıf birbirinin aynı olduğunda denememizi başarılı, farklı olduğunda ise hatalı olarak kaydettik.

**Aşama 7:** Bu algoritmayı hem öğrenim verilerinde 10 parça çapraz doğrulama esnasında, hem de test verileriyle test esnasında uyguladık. Çapraz doğrulama sonucunda en iyi oranı veren  $k$  değerini belirleyip bu değeri test işleminde kullandık. Test işlemi sırasında böylelikle birçok  $k$  değeri için deneme yapmamış olduk.

**Aşama 8:** Çapraz doğrulama işlemi sırasında 1'den 10'a kadar olmak üzere  $k$  için farklı değerler denedik. Bu değerlere karşılık gelen başarı oranlarını her bir parça için kaydedip çapraz doğrulama sonunda her  $k$  değeri için parçaların ortalamalarını karşılaştırdık. En yüksek oran veren  $k$  değerini test kümesindeki sinyallerin doğruluğunu tespit etmek üzere kullandık. Örneğin, toplamda 10000 sinyal test ettiyse ve bunların 6457'sinin sınıfı doğru sınıf olarak tahmin edildiyse başarı oranını  $6457 / 10000 = \%64.57$  olarak ifade ettik.

#### **4.5.1 10 Parça Çapraz Doğrulama**

10 Parça Çapraz Doğrulama süreci tüm sınıflandırma algoritmaları için istenildiği takdirde gerçekleştirilebilir ve elde edilen sonuçların güvenilirliğini artırmaya yarar. Bu süreç şu adımlar takip edilerek işleme konur:

- Tüm veriseti bir araya getirilir.
- Homojen dağılım gözetilerek veriseti 10 eşit parçaya ayrılır.
- Her parçada mümkün olduğu kadar her sınıfa ait eleman sayılarının birbirine eşit ya da oldukça yakın olması sağlanmaya çalışılır.
- Her bir parça bir kez test kümesi görevi yaparken geri kalan 9 parça birleşik halde öğrenim kümesi görevi üstlenir.
- Her bir parça için sınıflandırma başarı oranı tespit edilir.
- Tüm başarı oranları elde edildikten sonra tüm parçaların ortalama başarısı hesaplanır.

Kendi programımızda homojen dağılımı ve rastgele dağılımı tamamen sağladık. Tüm parçalardaki veriler tüm verisetindeki sinyaller arasından rastgele olarak seçildi

ve hiçbir sinyal başka bir parçada kendini tekrar etmedi. Parçalardaki sinyallerin sınıflara göre dağılımı Tablo 4.5'te verilmiştir.

Tablo 4.5 10 Parça Çapraz Doğrulama öncesi her sınıftaki verilerin parçalara göre dağılımı

	Yukarı	Sol	Sağ	Aşağı	Toplam
<b>Parça 1</b>	480	520	480	480	1960
<b>Parça 2</b>	480	520	480	480	1960
<b>Parça 3</b>	480	520	480	480	1960
<b>Parça 4</b>	480	520	480	480	1960
<b>Parça 5</b>	480	520	480	480	1960
<b>Parça 6</b>	480	520	480	480	1960
<b>Parça 7</b>	480	520	480	480	1960
<b>Parça 8</b>	480	520	480	480	1960
<b>Parça 9</b>	480	520	480	480	1960
<b>Parça 10</b>	480	520	480	480	1960
<b>Toplam</b>	4800	5200	4800	4800	19600

#### 4.5.2 Test Aşaması

10 Parça Çapraz Doğrulama sonrası tüm parçalardaki ortalama başarı oranlarının 1-10 arası seçilen  $k$  değerlerine bakarak en yüksek sonuç veren  $k$  değerini test aşamasında kullanmak amacıyla kaydettik. Yine rastgele seçilmek ve tekrar etmemek koşuluyla verisetini homojen dağılacak biçimde %75-%25 ve %67-%33 oranlarında test ve öğrenim kümesi olmak üzere parçaladık. Böylece test verilerinin öğrenimini güçlendirerek güvenilirliği artırmayı amaçladık. Her iki durumdaki veri kümelerinin sınıflara göre dağılımı Tablo 4.6'da verilmiştir.

Tablo 4.6 Test aşamasında test verilerinin sınıflara göre dağılımı

	Yukarı	Sol	Sağ	Aşağı	Toplam
<b>Dağılım 1</b> ( %75-%25)	1200	1300	1200	1200	4900
<b>Dağılım 2</b> (%67-%33)	1600	1734	1600	1600	6534

## BÖLÜM BEŞ

### DENEYSEL ÇALIŞMA

#### 5.1 C x k – En Yakın Komşuluk Algoritmasıyla EEG Göz Durumu Verilerini Sınıflandırma

Tez çalışmasında kullanılan verilerin dışında *EEG* verilerinde ikili sınıflandırmadaki başarı oranına dair sonuç almak için *UCI* Makine Öğrenimi Veritabanı tarafından paylaşılan *EEG* Göz Durumu Verileri için bir çalışma daha yaptık. “*Açık Göz*” ve “*Kapalı Göz*” olmak üzere *EEG* Sinyalleri alınan deneğin iki durumunu temsil eden sınıfları, *C x k – En Yakın Komşuluk* algoritması kullanarak tahmin ettik ve bu tahminlerin başarı oranlarını değerlendirdik (Doğan ve Nasibov, 2016).

Verisetimiz *UCI* Makine Öğrenimi Veritabanından alınan *EEG Göz Durumu Veriseti* (Roesler, 2013) olup tek bir deneğin göz açıp kapama durumlarında belli bir süre boyunca (117 saniye) kaydedilmiş *EEG* sinyallerinden oluşmuştur (Aydemir ve Kayıkçıoğlu, 2011). Bu *EEG* verilerinin son sütununa “*Göz Durumu*” özelliği eklenmiş olup, bu özellik değerine deney sırasında sinyalin alındığı anda kişinin gözü açıksa 0, kapalıysa 1 verilmiştir (Tablo 5.1) (Bing, Yao-hua ve Xiao-hong).

*EEG* sinyalleri satır bazlı olarak metin doyasına kayıtlı olarak alınmıştır. Toplam 14980 veri sıralı olarak zamana göre ardışık kayıt edilmiştir. Biz kodlama esnasında her bir sinyalin farklı elektrotlardaki değerlerinin doğru bir şekilde kullanılabilmesi için bu metin dosyası verilerini uygun formata dönüştürdük. *UCI Makine Öğrenimi Veritabanı*'nda da açıklandığı gibi bu sinyaller *Emotiv EEG Sinyal Kayıt Cihazı* kullanılarak elde edilmiştir.

*EEG* verilerini almak için farklı beyin loblarına bağlanmış toplamda 14 farklı elektrot bulunur. Bu elektrottaki değerler şu sırayla kaydedilmiştir:

*AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, AF4.*

Veriler hazır olarak güvenilir kaynaktan temin edilginde pek fazla bir önifleme operasyonu uygulanmamıştır, yalnızca veriler üzerinde sınıflandırma aşamasından önce “*Min-max Normalizasyonu*” uygulanmış, sınıflandırma algoritmasının başarısı test edilmiş ve aynı veriler üzerinde bilinen *k – En Yakın Komşuluk Algoritması*’nın uygulanmasıyla elde edilen sonuçlar ile karşılaştırma yapılmıştır.

Verilerin birimi  $\mu V$  (mikrovolt) cinsindedir. Örneğin, *P8* elektrotunda *4148,21* değeri olan bir veri ölçüm yapıldığı anda deneğin *P8* elektrotunun yerleştiği beyin kısmındaki potansiyel farkının *4148,21  $\mu V$*  yani *4,14821 mV* değerinde olduğu anlamını taşır. Tablo 5.1’e bakıldığında elektrot kanallarındaki anlık ölçülen değerlerin kaydedilmiş hali görülür.

Tablo 5.1 EEG sinyallerinin excel dosyasına kayıtlı halinin görünümü

<b>T8</b>	<b>FC6</b>	<b>F4</b>	<b>F8</b>	<b>AF4</b>	<b>eyeDetection</b>
4238,46	4211,28	4280,51	4635,9	4393,85	0
4226,67	4207,69	4279,49	4632,82	4384,1	0
4222,05	4206,67	4282,05	4628,72	4389,23	0
4235,38	4210,77	4287,69	4632,31	4396,41	0

*14980* adet veri içeren tüm veri setini rastgele biçimde homojen sınıf dağılımını gözetererek *10000* adet veri içeren bir öğrenim kümesi ve *4980* adet veri içeren bir test kümesi olmak üzere iki parçaya ayırdık. Tablo 5.2’de öğrenim kümesindeki verilerin *10* parçalı çapraz doğrulama (*10 Folds Cross Validation*) sırasında oluşan sınıflara göre sayısal dağılımları verilmiştir. Tablo 5.3’de ise test kümesindeki elemanların sınıflara göre sayısal dağılımları gösterilmiştir. Çapraz doğrulama esnasında veriler *1000*’er farklı veri içeren rastgele seçilmiş homojen dağılım gösteren *10* parçaya ayrılmıştır.

Tablo 5.2 Öğrenim kümesindeki verilerin 10 parçalı çapraz doğrulama sırasında her bir sınıf için satır sayıları

Parça	Parçadaki “Açık Göz” Veri Sayısı	Parçadaki “Kapalı Göz” Veri Sayısı	Kalan Kısımdaki “Açık Göz” Veri Sayısı	Kalan Kısımdaki “Açık Göz” Veri Sayısı
Parça 1	540	460	4988	4012
Parça 2	545	455	4983	4017
Parça 3	569	431	4959	4041
Parça 4	563	437	4965	4035
Parça 5	509	491	5019	3981
Parça 6	555	445	4973	4027
Parça 7	562	438	4966	4034
Parça 8	557	443	4971	4029
Parça 9	555	445	4973	4027
Parça 10	573	427	4955	4045
<b>Toplam</b>	<b>5528</b>	<b>4472</b>		

Tablo 5.3 Test kümesindeki verilerin test sırasında her bir sınıf için satır sayıları

	Testteki “Açık Göz” Veri Sayısı	Testteki “Kapalı Göz” Veri Sayısı	Öğrenimdeki “Açık Göz” Veri Sayısı	Öğrenimdeki “Açık Göz” Veri Sayısı
<b>Test Kümesi</b>	2714	2266	5543	4457

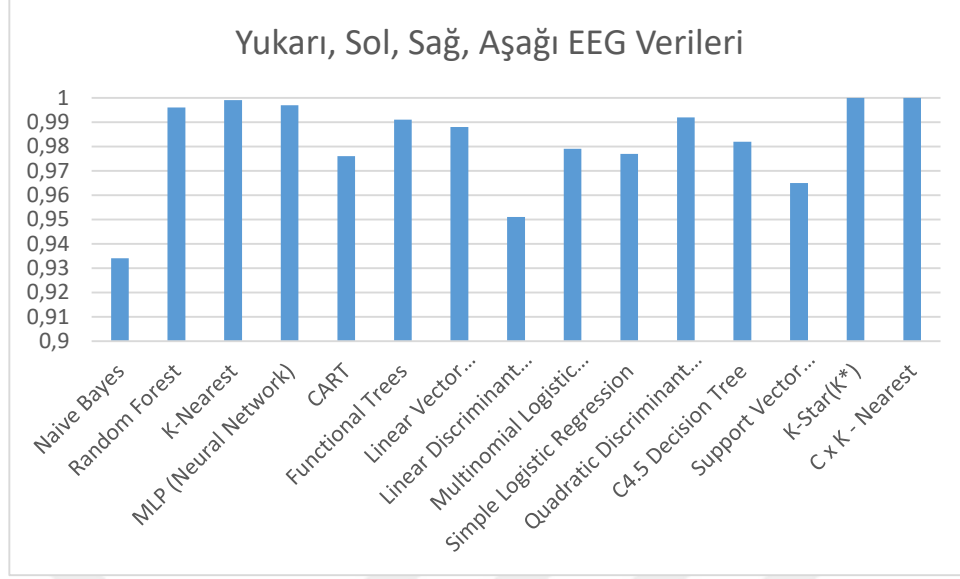
## 5.2 C x k – En Yakın Komşuluk Algoritmasıyla Temel Yönleri (Yukarı, Sol, Sağ, Aşağı) Verilerini Sınıflandırma

Düzenlediğimiz tüm verisetlerini 10 parça çapraz doğrulama aşaması için hazırladıktan sonra çeşitli algoritmalar kullanarak sınıflandırdık. Danışman hocamın önerdiği *C x k- En Yakın Komşuluk Algoritması*'nin önerdiği yöntemi *Visual Studio* ortamında *C#* programlama dili kullanarak kendimiz kodladık. Farklı verisetlerine göre kodu düzenledik. *EEG* sinyal analizinde tercih edilen diğer yaygın sınıflandırma algoritmalarının başarı sonuçlarını da analizimize katmak için *Weka* yazılımını kullanarak verisetlerimizi bu yazılım vasıtasıyla farklı sınıflandırma yöntemlerine tabi tuttuk. 10 parça çapraz doğrulama sonuçlarının farklı verisetleri ve çeşitli

sınıflandırma algoritmaları için karşılaştırmalı sonuçları Tablo 5.4 ve Şekil 5.1’de verilmiştir.

Tablo 5.4 10 Parça Çapraz Doğrulama aşaması verisetlerinin çeşitli algoritmalara göre karşılaştırmalı sonuç tablosu

	<b>Yukarı, Sol, Sağ, Aşağı EEG Verileri</b>
<b>Naive Bayes</b>	0,934
<b>Random Forest</b>	0,996
<b>K-Nearest</b>	0,999
<b>MLP (Neural Network)</b>	0,997
<b>CART</b>	0,976
<b>Functional Trees</b>	0,991
<b>Linear Vector Quantization(LVQ)</b>	0,988
<b>Linear Discriminant Analysis(LDA)</b>	0,951
<b>Multinomial Logistic Regression</b>	0,979
<b>Simple Logistic Regression</b>	0,977
<b>Quadratic Discriminant Analysis(QDA)</b>	0,992
<b>C4.5 Decision Tree</b>	0,982
<b>Support Vector Machine(SVM)</b>	0,965
<b>K-Star(K*)</b>	1
<b><u>C x K - Nearest</u></b>	1



Şekil 5.1 10 Parça Çapraz Doğrulama aşaması verisetlerinin çeşitli algoritmalara göre karşılaştırmalı sonuç grafiği

Tablo 5.4'te farklı kategorilere ait sınıflandırma algoritmaları kullanılmıştır. *Random Forest* algoritması birçok sınıflandırma metodlarının iyi yanlarından yararlanabilen öğrenme kapasitesi yüksek birleşik yapıda bir algoritmadır. Fazla sayıda sınıflandırma yöntemi içerdiği için çalışma zamanı diğerlerine göre daha yavaştır ancak farklı algoritmaların güçlü yanlarından yararlandığı için tekil algoritmalara göre daha iyi sonuç alması beklenir. Başarı oranı verilen *K- En Yakın Komşuluk* algoritmasındaki  $k$  değeri 1 dir ve uzaklık ölçüm yöntemi olarak *Öklid* uzaklığı kullanılmıştır. Bunlardan başka, *MLP* olarak gösterilen, yapay sinir ağları yöntemlerinden biri olan çok katmanlı perceptrondur. Öğrenme oranı 0,3 ve momentum değeri 0,2'dir. 1 gizli katmanı vardır. Bu gizli katmandaki nod sayısı 25 iterasyon sayısı ise 500'dür. *CART* algoritması, Sınıflandırma ve Regresyon Ağacı olarak isimlendirilen yöntemin kısaca belirtilmiş ismidir. İstatistiksel yöntemlerden regresyon ve karar ağacı barındırır. *Functional Trees* metodunda, ağacın iç nodlarında ve yapraklarda lojistik regresyon fonksiyonları kullanılmıştır. Diğer taraftan, *LVQ* da ise 1600 rassal codebook vektörleri her sınıf için yer alır. Bu sayı veri dağılımına yönelik doğru orantılı olarak veriyi eğitmek için seçilmiştir. İterasyon sayısı 10000'dir ve gerekli durumlarda çoğunluk oylaması kullanır. Pencere boyutu 0,7, başlangıç öğrenme oranı 0,8 epsilon değeri ise 0,5 dir. Verisetimiz için uyguladığımız Destek



Vektör Makinesi'nde (*SVM*), makineyi eğitmek için Ardışık Minimal Optimizasyon (*SMO*) algoritmasından yararlanılmıştır. *SVM* ikili sınıflandırmalara daha uygun bir algoritma olsada, ilk sınıf ayırımından sonra geride kalan sınıfları tüm sınıflar ayrışana dek 2 sınıf olarak ele alır ve *EEG* sinyal analizinde tercih edilen yöntemlerden biridir. Son olarak kullanılan  $K^*$  algoritması içinse dağıntı (*entropy*) temelli bir uzaklık fonksiyonu kullanan örnek tabanlı öğrenme sınıflandırıcısı diyebiliriz. Tablo 10'da görüldüğü üzere veriseti tüm sınıflandırıcılar için iyi sonuç vermekle beraber özellikle örnek tabanlı (*instance-based*) algoritmalar üzerinde daha başarılı oran verdiği dikkat çeker. Bu durumda bize verisetinin örnek tabanlı yöntemlere daha uygun olduğu izlenimi verir. Ayrıca  $K^*$  ve  $C \times k - En\ Yakın\ Komşuluk$  metodlarının kayıpsız başarı yakaladığı ve kullanılan veriseti için diğer algoritmalara göre nispeten daha başarılı olduğu tespit edilmiştir.

10 Parça Çapraz Doğrulama esnasında uygulamaya koyduğumuz  $C \times k - NN$  algoritmasındaki her bir parçanın 1-10 arası  $k$  değerleri için verdiği sonuçları gösteren bilgi Tablo 5.5'te verilmiştir. Ayrıca Tablo 5.6'da aynı aşamadaki tüm parçalar için  $k - NN$  algoritmasının aldığı sonuçları yansıtır.

Tablo 5.5  $C \times k - NN$  algoritmasının her bir parçadaki farklı  $k$  değerlerine göre aldığı sınıflandırma sonuçları

	1	2	3	4	5	6	7	8	9	10
<b>Parça 1</b>	0,999	0,999	0,998	0,998	0,999	0,999	0,999	0,999	0,999	0,999
<b>Parça 2</b>	1	1	1	1	1	1	1	1	1	0,999
<b>Parça 3</b>	0,999	0,999	0,999	0,998	0,998	0,998	0,998	0,998	0,997	0,997
<b>Parça 4</b>	1	1	1	1	1	0,999	0,999	0,999	0,999	0,999
<b>Parça 5</b>	0,999	0,998	0,999	0,999	0,999	1	1	0,999	1	0,999
<b>Parça 6</b>	1	0,999	0,999	0,999	0,999	0,998	0,999	0,999	1	1
<b>Parça 7</b>	0,998	0,999	0,999	0,999	0,999	0,998	0,998	0,999	0,999	0,999
<b>Parça 8</b>	1	0,999	0,998	0,998	0,998	0,999	0,999	0,999	0,999	0,998
<b>Parça 9</b>	0,999	1	1	1	0,999	0,998	0,998	0,999	0,997	0,997
<b>Parça 10</b>	0,998	1	1	0,998	0,998	1	0,999	0,997	0,997	0,997
<b>Ort.</b>	0,999	0,999	0,999	0,998	0,998	0,998	0,998	0,998	0,998	0,998

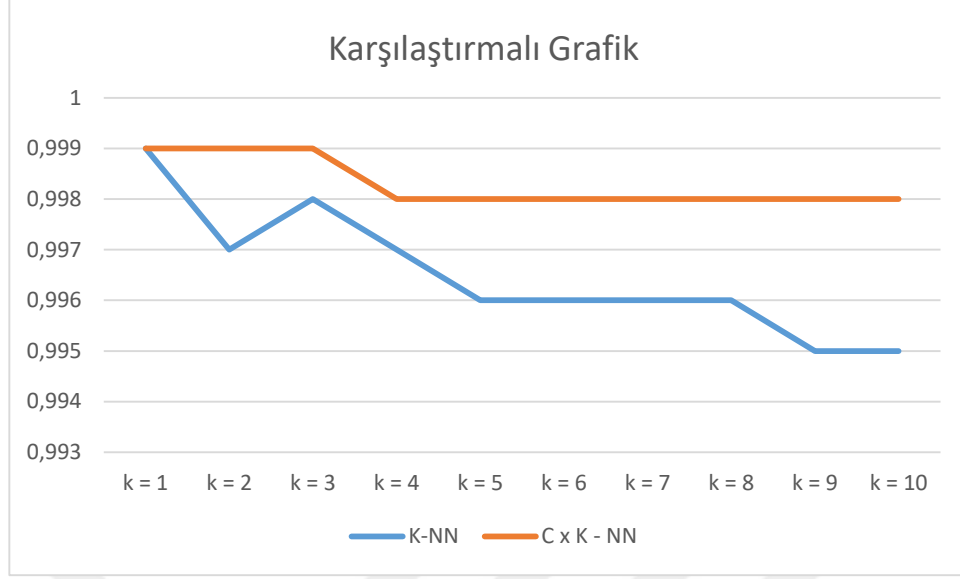
Tablo 5.6 k -NN algoritmasının her bir parçadaki farklı k değerlerine göre aldığı sınıflandırma sonuçları

	1	2	3	4	5	6	7	8	9	10
<b>Parça 1</b>	0,999	0,997	0,998	0,998	0,998	0,997	0,997	0,997	0,997	0,997
<b>Parça 2</b>	1	0,996	0,997	0,998	0,995	0,995	0,996	0,996	0,996	0,995
<b>Parça 3</b>	0,999	0,995	0,999	0,996	0,996	0,995	0,996	0,995	0,994	0,996
<b>Parça 4</b>	1	0,999	0,999	0,997	0,996	0,997	0,996	0,996	0,994	0,996
<b>Parça 5</b>	0,999	0,997	0,997	0,996	0,996	0,996	0,997	0,995	0,995	0,995
<b>Parça 6</b>	1	0,996	0,998	0,996	0,994	0,996	0,997	0,997	0,994	0,994
<b>Parça 7</b>	0,998	0,997	0,997	0,998	0,994	0,996	0,995	0,997	0,996	0,995
<b>Parça 8</b>	1	0,998	0,998	0,997	0,998	0,998	0,997	0,996	0,995	0,994
<b>Parça 9</b>	0,999	0,997	0,997	0,997	0,998	0,994	0,995	0,996	0,996	0,995
<b>Parça 10</b>	0,998	0,998	0,998	0,997	0,997	0,996	0,995	0,997	0,996	0,996
<b>Ort.</b>	0,999	0,996	0,997	0,996	0,995	0,996	0,995	0,996	0,995	0,993

Ek olarak,  $k$  - NN algoritmasını temel alarak geliştirilen  $C \times k$  -NN yönteminin esinlendiği algoritmayla karşılaştırmak adına farklı  $k$  değerleri için aynı veriseti üzerinde 2 yöntemin verdiği sonuçlar Tablo 5.7'de yansıtılmıştır. Bu sonuçlar *10 Parça Çapraz Doğrulama* için geçerlidir.

Tablo 5.7 *10 Parça Çapraz Doğrulama* aşaması *Yukarı, Sol, Sağ, Aşağı* verisetinin farklı  $k$  değerlerine göre karşılaştırmalı sonuç tablosu

	<b>K-NN</b>	<b>C x K - NN</b>
<b>k = 1</b>	0,999	0,999
<b>k = 2</b>	0,997	0,999
<b>k = 3</b>	0,998	0,999
<b>k = 4</b>	0,997	0,998
<b>k = 5</b>	0,996	0,998
<b>k = 6</b>	0,996	0,998
<b>k = 7</b>	0,996	0,998
<b>k = 8</b>	0,996	0,998
<b>k = 9</b>	0,995	0,998
<b>k = 10</b>	0,995	0,998



Şekil 5.2 10 Parça Çapraz Doğrulama aşaması Yukarı, Sol, Sağ, Aşağı verisetinin farklı  $k$  değerlerine göre karşılaştırmalı sonuç grafiği

Tablo 5.7’de görüldüğü üzere tüm  $k$  değerleri için  $C \times k - NN$  yöntemiyle kısmen daha başarılı sonuçlar alındığı açıktır. Buna ek olarak her iki yöntemde de  $k$  değerini 1 olarak seçmenin akıllıca olduğu fark edilir. Şekil 31’e bakıldığında  $C \times k - NN$  yönteminde  $k$  değeri 3’ten büyük olduğunda başarı oranının düştüğü ve sabit seyrettiği görülür.

Aynı veriseti için bahsedilen farklı ayırma oranlarına yönelik test aşamasında elde edilen sonuçları tablo da verilmiştir. Bu sonuçlar kendi kodladığımız  $C \times k - NN$  yöntemi kullanılarak elde edilmiştir. Buna ek olarak, veriler test aşamasında da homojen olarak dağıtılmış test verilerinin farklı dağılım oranlarına göre test kümesindeki sınıflara ait miktarları Tablo 5.8’de gösterilmiştir.

Tablo 5.8 Test aşaması Yukarı, Sol, Sağ, Aşağı verisetinin farklı dağılım oranlarına göre her sınıftaki sinyal miktarları

Test Verileri Dağılım Oranı	Yukarı	Sol	Sağ	Aşağı
75% - 25%	1200	1300	1200	1200
67% - 33%	1600	1733	1600	1600

Tablo 5.8'i incelediğimizde 10 parça çapraz doğrulama aşamasında gözlemlendiği gibi *Yukarı*, *Aşağı* ve *Sağ* sınıflarına ait sinyal miktarlarının aynı olduğu, *Sol* sınıfına ait veri sayısının biraz daha fazla olduğu aşikârdır. Alınan veri miktarında değişim yapmadığımız için tüm sınıflandırma işlemlerinde bu fark gözlenir.

Tablo 5.9 Test aşaması *Yukarı*, *Sol*, *Sağ*, *Aşağı* verisetinin  $C \times k$ -NN algoritmasıyla farklı  $k$  değerlerine göre karşılaştırmalı sonuç tablosu

Test Verileri Dağılım Oranı	1	2	3	4	5	6	7	8	9	10
75% - 25%	0,999	0,999	0,999	0,998	0,998	0,998	0,998	0,998	0,998	0,998
67% - 33%	1	0,999	0,999	0,999	0,999	0,999	0,999	0,999	0,998	0,998

Tablo 5.10 Test aşaması *Yukarı*, *Sol*, *Sağ*, *Aşağı* verisetinin  $k$  - NN algoritmasıyla farklı  $k$  değerlerine göre karşılaştırmalı sonuç tablosu

Test Verileri Dağılım Oranı	1	2	3	4	5	6	7	8	9	10
75% - 25%	0,999	0,997	0,998	0,996	0,996	0,994	0,995	0,992	0,993	0,994
67% - 33%	1	0,998	0,999	0,997	0,996	0,995	0,995	0,994	0,994	0,992

Tablo 5.9 ve Tablo 5.10'da gösterilen sonuçlar iki sinyalin benzerliğini ölçerken *Öklid Uzaklık Yöntemi*'nin kullanılmasıyla elde edilmiştir. Algoritma Uygulaması bölümünde *Aşama 1*'de bahsedilen diğer 3 farklı uzaklık metrikleri için alınan sonuçlar Tablo 5.11 ve Tablo 5.12' de gösterilir. Tablo 5.11'e göre  $k = 1$  değeri için test verileri miktarı daha yoğun, öğrenim sinyalleri sayısı daha az olduğunda tüm test sinyallerinin sınıflarının doğru tespit edildiği gözlenir. Diğer  $k$  değerleri için de kıyaslama da bir değişiklik olmaz. Bindelik düzeyde test verileri sayısı tüm verisetinin  $1/3$ 'ü olduğunda daha iyi oranla sınıflandırma yapıldığı görülür.

Tablo 5.11 Test aşaması *Yukarı, Sol, Sağ, Aşağı* verisetinin %75-%25 dağılımına göre  $C \times k - NN$  algoritmasıyla farklı uzaklık ölçümleri için başarı oranları.

Uzaklık Ölçüm Yöntemi	1	2	3	4	5	6	7	8	9	10
Öklid	0,999	0,999	0,999	0,998	0,998	0,998	0,998	0,998	0,998	0,998
Cosine	0,998	0,999	0,998	0,998	0,998	0,998	0,997	0,996	0,996	0,995
Hellinger	0,998	0,999	0,998	0,998	0,998	0,998	0,998	0,997	0,997	0,996
Bray-Curtis	0,998	0,998	0,998	0,998	0,998	0,998	0,998	0,998	0,997	0,996

Tablo 5.12 Test aşaması *Yukarı, Sol, Sağ, Aşağı* verisetinin %67-%33 dağılımına göre  $C \times k - NN$  algoritmasıyla farklı uzaklık ölçümleri için başarı oranları

Uzaklık Ölçüm Yöntemi	1	2	3	4	5	6	7	8	9	10
Öklid	1	0,999	0,999	0,999	0,999	0,999	0,999	0,999	0,998	0,998
Cosine	0,999	0,999	0,999	0,999	0,999	0,999	0,999	0,999	0,999	0,998
Hellinger	0,997	0,998	0,998	0,998	0,997	0,996	0,996	0,996	0,996	0,995
Bray-Curtis	0,999	0,999	0,999	0,999	0,999	0,999	0,998	0,998	0,998	0,998

Tablo 5.13 Test aşaması *Yukarı, Sol, Sağ, Aşağı* verisetinin %75-%25 dağılımına göre  $k - NN$  algoritmasıyla farklı uzaklık ölçümleri için başarı oranları.

Uzaklık Ölçüm Yöntemi	1	2	3	4	5	6	7	8	9	10
Cosine	0,998	0,996	0,996	0,996	0,995	0,994	0,994	0,994	0,995	0,993
Hellinger	0,998	0,997	0,996	0,996	0,996	0,996	0,995	0,994	0,994	0,994
Bray-Curtis	0,998	0,998	0,997	0,998	0,997	0,998	0,996	0,997	0,997	0,996

Tablo 5.14 Test aşaması *Yukarı, Sol, Sağ, Aşağı* verisetinin %67-%33 dağılımına göre  $k - NN$  algoritmasıyla farklı uzaklık ölçümleri için başarı oranları

Uzaklık Ölçüm Yöntemi	1	2	3	4	5	6	7	8	9	10
<b>Cosine</b>	0,999	0,997	0,997	0,997	0,996	0,994	0,995	0,995	0,995	0,994
<b>Hellinger</b>	0,997	0,998	0,997	0,997	0,979	0,996	0,995	0,995	0,995	0,994
<b>Bray-Curtis</b>	0,999	0,998	0,998	0,998	0,997	0,997	0,996	0,996	0,995	0,996

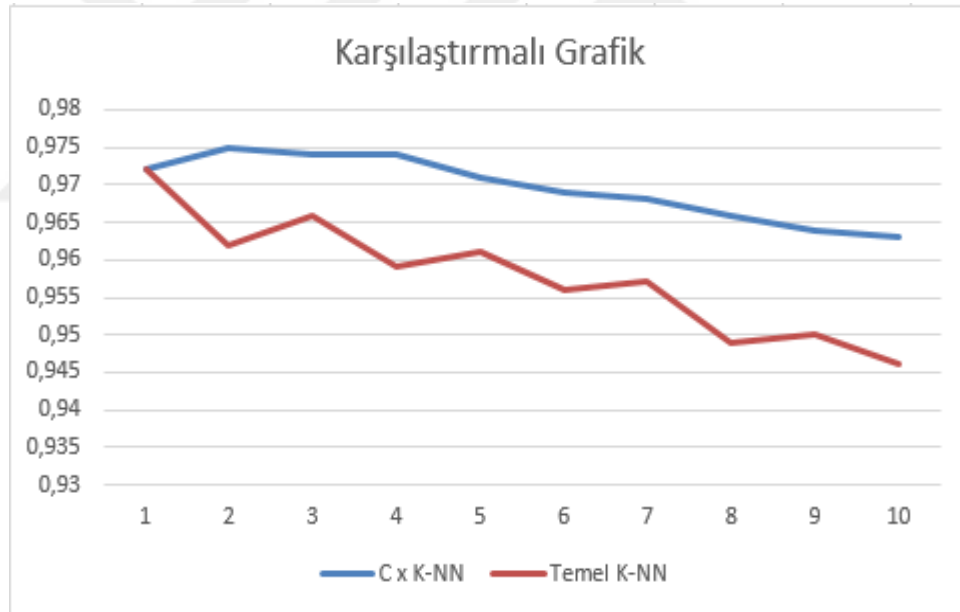
Farklı türden uzaklık metriklerini farklı test-öğrenim küme dağılımları için karşılaştıran Tablo 5.11 ve Tablo 5.12'ye bakıldığında, tüm sonuçlar arasında yüzdeler düzeyde ciddi bir fark olmadığı görülür. Bindelik düzeyden itibaren başlayan küçük iyileşme veya kötüleşmelere rastlanır. Tüm bu karşılaştırmalar göz önüne alındığında *Öklid* uzaklık yönteminin daha iyi başarı elde etmesi dolayısıyla kullandığımız veriseti için tercih edilmesinin akıllıca olduğu saptanır. Tablo 5.13 ve Tablo 5.14'de görüldüğü üzere %75-%25 oranında test kümesi ayırımı yapan çalışma için *Bray-Curtis* uzaklık yönteminin çok az farkla da olsa diğer metriklerle oranla daha geriden geldiği görülür. Fakat diğer dağılım türü olan %67-%33 oranında test kümesinin daha yoğun olduğu dağılım içinse *Hellinger* uzaklık ölçüm (benzeşmezlik) yönteminin biraz daha az başarı verdiği gözlenir.

Bir diğer verisetimiz olan *EEG Göz Durumu* verileriyle elde ettiğimiz başarı oranları Tablo 5.15'te verilmiştir. Bu tablo incelendiğinde görülür ki  $k = 1$  değeri için her iki algoritmadaki sonuçlar birbirinin aynıdır. Bu beklenen bir durumdur. Çünkü,  $k = 1$  değeri için temel  $k - NN$  de test sinyaline yakınlığı bakımından en benzer olan sinyal ele alınmasına karşın,  $C \times k - NN$ 'de tüm sınıflardan en yakın sinyaller arasında en yakın olanı baz alınarak işlem yapılır. Her iki durumda da aynı sinyalden bahsetmek mümkündür. Hangi sınıfa ait olursa olsun en yakın sinyal tektir ve iki algoritma da böylelikle aynı sinyal üzerinde işlem yaparak aynı sonuçlara ulaşır. Öte yandan, Şekil 5.3'te görüldüğü üzere  $k$  değeri arttıkça 2-10 arası tüm değerler için her iki algoritma da başarı oranında ufak düşüşler yaşamasına rağmen  $C \times k - NN$  metodunun kısmi

olarak  $k$  -  $NN$  metodundan daha iyi sonuç verdiği gözlenir. Birden fazla benzer veri tipleri için aynı türden sonuçlar almak  $C x k - NN$  yöntemi için güven sağlar.

Tablo 5.15 EEG Göz Durumu verileri için  $C x k - NN$  ve temel  $k - NN$  yöntemlerinin başarı oranlarını 1-10 arasındaki  $k$  değerleri için karşılaştıran tablo

k Değerleri	C x k-NN	k-NN
1	0,972	0,972
2	0,975	0,962
3	0,974	0,966
4	0,974	0,959
5	0,971	0,961
6	0,969	0,956
7	0,968	0,957
8	0,966	0,949
9	0,964	0,950
10	0,963	0,946



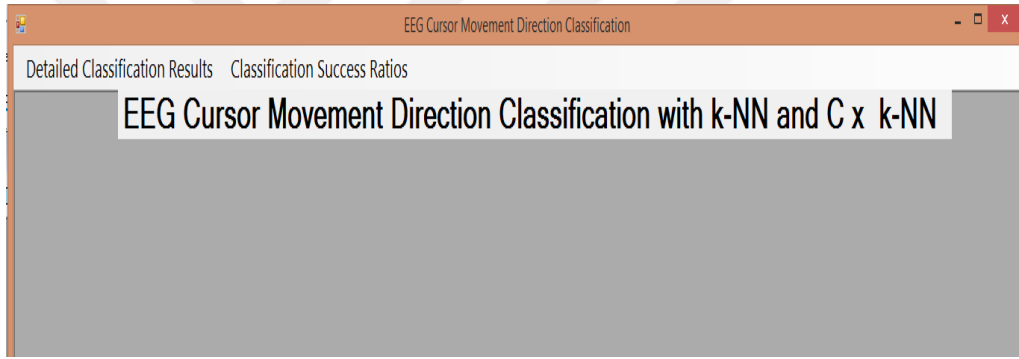
Şekil 5.3 EEG Göz Durumu verileri için  $C x k - NN$  ve temel  $k - NN$  yöntemlerinin başarı oranlarını 1-10 arasındaki  $k$  değerleri için karşılaştıran grafik

### 5.3 Program Arayüzü

Önceki bölümlerde bahsedilen tüm işlemleri *Visual C#* ortamında kodlayarak gerçekleştiren bir program oluşturduk. Bu program sırasıyla kaydedilen verileri

okuma, verileri kullanılabilir hale getirmek için gerekli ön işleme, kullanıcıya anlaşılır ve sade bir arayüz sunarak istediği verilen bazı kıstas seçimine uygun sınıflandırma yapma ve seçilen sinyallerin nasıl ve hangi sırayla ayrıldığına dair bilgiler yansıtma gibi özellikler ortaya koyar.

Program ana ekranında üst menü sekmesi iki farklı işlem çeşidi içerir. Bunlar sırasıyla detaylı sınıflandırma sonuçları ve sınıflandırma başarı oranlarıdır. Kendi kodladığımız programda yalnızca *k - En Yakın Komşuluk* ve *C x k - En Yakın Komşuluk* algoritmaları yer aldığından ötürü arayüz bilgilendirmesinde sadece bu iki algoritmadan bahsedilmiştir.



Şekil 5.4 k - NN ve C x k - NN sınıflandırması yapan programımızın arayüz ana ekranı

Şekil 5.4'te gösterildiği gibi Ana Menü ekranı açıldığında arka planda yapılan işlemler sırasıyla şu şekilde sıralanır:

- Tüm sinyaller metin dosyasından sınıf etiketleriyle birlikte alınır
- Üzerinde işlem yapılabilmesi için gerekli metin-sayı dönüşümleri yapılır.
- Her bir sinyalin özellik bazlı değerleri matriste ilgili hücrelere yerleştirilir
- Alınan tüm sinyaller gözetilerek, her sinyalin her özellik değeri için *min-max normalizasyonu* yapılır.



- *10 Parçalı Çapraz Doğrulama* işlemi sırasında kullanılmak üzere *10* parçanın test ve öğrenim sinyalleri rastgele seçilip ayrılarak ayrı dizi elemanlarına yerleştirilip saklanır.
- Test esnasında kullanılmak üzere tüm veriler rastgele seçim yapılarak %33 test sinyali ve %67 öğrenim sinyali olmak üzere ayrıştırılarak farklı dizi elemanlarında saklanır.

Bu menü sekmelerinden detaylı sınıflandırma sonuçları tıklandığında *10 Parçalı Çapraz Doğrulama* esnasında her bir parçada seçilen sinyallerin hangi sinyaller olduğu ve bu sinyallerin gerçek sınıfları ile tahmin edilen sınıfları karşılaştırılarak sinyal bazlı tahmin doğruluğu bilgisi yansıtılır.

Signal Number	Signal ID	Real Class	Predicted Class
340	12390	Right	Right
341	4404	Up	Up
342	15514	Down	Down
343	2829	Up	Up
344	9310	Left	Left
345	3620	Up	Down
346	15869	Down	Down
347	1133	Up	Up
348	1050	Up	Up
349	6231	Left	Left
350	17582	Down	Down

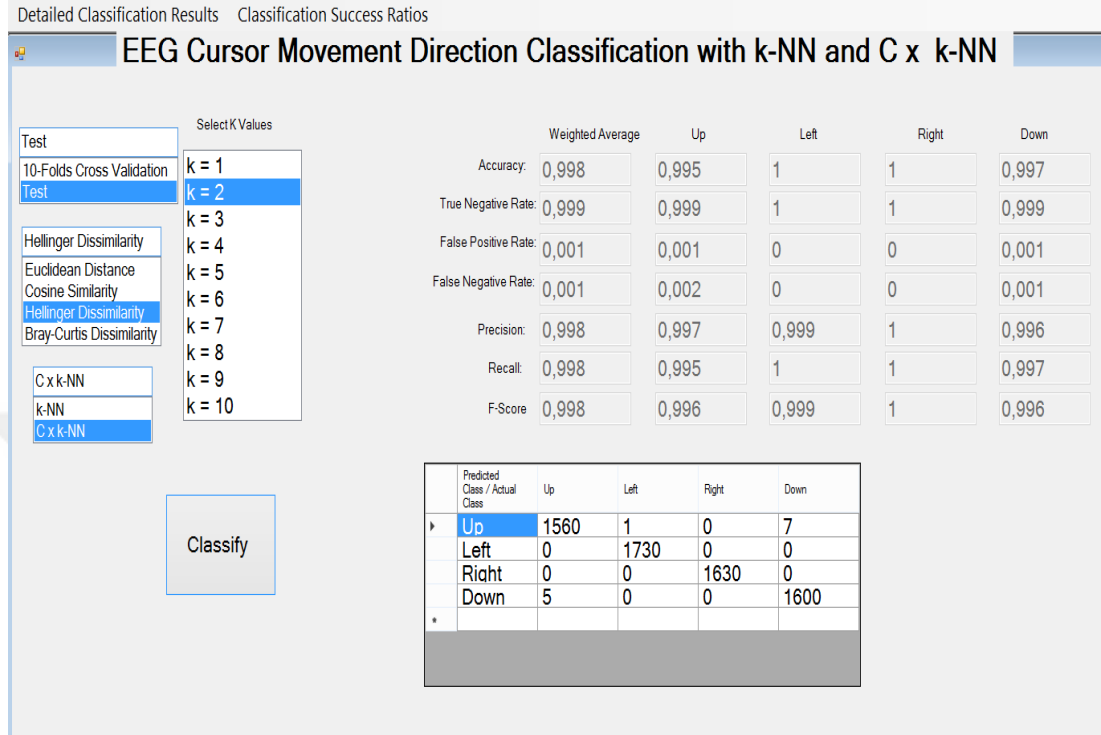
Şekil 5.5 Program arayüzü detaylı sınıflandırma sonuçları ekranı

Şekil 5.5'te görüldüğü üzere Detaylı Sınıflandırma Sonuçları ekranının sol kısmında sınıflandırma algoritması ( $C \times k - NN$  ve  $k - NN$  algoritmalarından yalnızca biri) ve bu algoritmalarda sinyallerin birbirine uzaklığını ölçmek için kullanılacak uzaklık ölçüm yöntemi (Şekil 5.5'te görülen 4 uzaklık metriklerinden yalnızca biri) seçimi yapılır. Görülen ekranda uzaklık ölçümü olarak *Bray-Curtis Uzaklığı* ve sınıflandırma tekniği olarak  $C \times k - NN$  yöntemi seçilmiştir. Daha sonra yeni sinyalin sınıfını belirleyecek en yakın komşu sayısı  $[1-10]$  aralığında belirlenir. Son olarak *10*

Parçalı Çapraz Doğrulama işleminde hangi parçanın detaylı sonuçları yansıtılmak istenirse bu parça numarası [1-10] aralığında seçilir. Şekil 5.5'teki ekranda en yakın komşu sayısı  $k = 3$  ve 10 Parça içinden 9. Parça seçilmiştir. Tüm bu işlemler tamamlandıktan sonra "sınıflandır" butonuna basıldığında sağ taraftaki tablo ekrana gelir. Bu tabloda ilgili parçadaki tüm sinyallerin o parçaya konulma sırası (*Signal Number*, ilk sinyal ilk sırada en son yerleşen sinyal son sırada olmak üzere), tüm veri seti içindeki satır numarası (*Signal ID*), sinyalin gerçek sınıfı (*Real Class*) ve sinyalin sınıflandırma işlemi sonrası tahmin edilen sınıfı (*Predicted Class*) yer alır. Tüm verisetinde 19600 sinyal bulunduğu ve 10 parçaya bu sinyaller eş sayıda ve homojen dağıtıldığı için bir parçada 1960 sinyal bulunur. Bu nedenle sağ tarafta görülen tabloda *Sinyal Numarası* sütunu 1 ile başlayıp 1960 ile son bularak artan sırayla ilerler. İlgili parçaya yerleşen sinyaller tüm veriseti içerisinde rastgele seçilerek dağıtıldığı için *Sinyal ID* sütunundaki değerler [1-19600] aralığında olup sırasız şekilde görünür. Böylelikle ilgili parçaya yerleşen her sinyalin tüm veriseti içinden hangi sinyal olarak seçildiği tespit edilir ve rastgele dağılım işlemi ile aynı sinyalin farklı parçalarda tekrar etmeme durumu görsel olarak kontrol edilebilir. Bu ekranın oluşturulma amacı kodda bir hata olmamasını sağlamak ve istenilen işlemin kesinlikle yapıldığını kontrol etmektir. Örneğin, Şekil 5.5'teki tablo incelendiğinde 9. Parça'da seçilen 347. sıradaki sinyalin tüm verisetindeki 1133. sinyal olduğu görülür. Böylelikle istenildiği halde tüm yerleştirilen sinyallere bakılarak aynı sinyalin tekrar etme durumu veya bazı sinyallerin sıralı olmama durumu incelenebilir. Sinyalin gerçek sınıfı (*Real Class*) o sinyalin hangi yöne ait düşüncüyü temsil ettiğini belirtir. Tahmin edilen sınıfı ise sınıflandırma algoritmalarından biri çalıştırıldıktan sonra ilgili test sinyalinin hangi sinyal sınıfına en yakın olduğunu gösterir. Her iki sınıf değeri birbiriyle eşleşir yani tahmin edilen sınıf doğru olursa, bu durumda her iki sınıf değerinin arka planı yeşil renkte ifade edilir. Aksi takdirde yani sinyalin tahmin edilen sınıfı gerçek sınıftan farklı olduğu durumda da her iki hücredeki değer Şekil 5.5'te 3620. Sinyal de olduğu gibi kırmızı arka plan rengiyle sergilenir. Görsel takip açısından bu renk farkı izlenimi kolaylaştırır.

*10 Parçalı Sınıflandırma Aşaması* detaylı sonuçlar ekranından başka bu aşamanın ve test aşamasının farklı  $k$  değerlerine göre her iki algoritma için çeşitli uzaklık

ölçüleriyle başarı oranlarını birçok sınıflandırma parametresine yönelik görüntüleyen sınıflandırma başarı oranları ekranımız mevcuttur.



Şekil 5.6 Program arayüzü sınıflandırma başarı oranları ekranı

Şekil 5.6'da Sınıflandırma Başarı Oranları ekranı programın arayüzü içinden yansıtılmıştır. Burada görüldüğü üzere test verilerinin sınıflandırma başarısı ölçülmek istenmiştir. Test seçeneği seçildiğinde tüm verisetinin %33,3'lük bir kısmı rastgele olarak test verisi için ayrılır. Bu oran kullandığımız veriseti için 6533 adet sinyal sayısına eşdeğerdir. Bir diğer seçenekte bir önceki ekranda olduğu gibi uzaklık ölçüm yöntemi tercihi yapılır. Şekil 5.6'daki çalışmada Hellinger uzaklık yöntemi seçilmiştir. Test verileri seçimi yapıldığı için 10 Parçalı Çapraz Doğrulama'da olduğu gibi herhangi bir parça seçimi olmayacağından parça seçimine ait liste kutusu görüntülenmemiştir. Yalnızca  $k$  değerine ait seçim [1-10] arası yapılır. Şekildeki çalışmada  $k$  değeri 2 olarak seçilmiştir. Ardından sınıflandırma butonu tıklandığında sınıflandırma işlemi başlar ve bu işlem tamamlandığında ekrana daha önceki bölümlerde bahsedilen istatistiksel sınıflandırma ölçütleri yansıtılır. Bu ölçütler sırasıyla  $TPR$  (Accuracy),  $TNR$ ,  $FPR$ ,  $FNR$ ,  $Precision$ ,  $Recall$  ve  $F-Score$  değerleridir.

Her bir sınıflandırma ölçütü tüm sınıflar için (*yukarı, sol, sağ ve aşağı*) ayrı ayrı ve tüm sınıfların ağırlıklı ortalamasıyla birlikte verilmiştir. Böylelikle sınıflara yönelik başarı oranları ve genel başarı durumu tespit edilebilir. Bunlara ek olarak, tüm sinyallerin sınıflara dağılımına göre miktarları üzerinden sinyalin gerçek sınıfı ve tahmin edilen sınıfına yönelik *confusion matrix* bilgisi görüntülenmiştir. Bu ekranda diyagonal hücrelere yerleşen sayı miktarları sınıfı doğru tahmin edilmiş sinyallerin sayısını gösterir. Örneğin, Şekil 5.6'ya bakıldığında hem gerçek sınıfı hem de tahmin edilen sınıfı yukarı olan 1560 sinyal mevcuttur. Benzer şekilde, hem gerçek sınıfı sağ hem de tahmin edilen sınıfı sağ olan 1630 sinyal vardır. Bu diyagonal hücrelerin dışındaki hücrelerde yer alan sayılar sınıfı yanlış olarak tahmin edilen sinyallerin bilgisini verir. Örneğin, yine Şekil 5.6 incelendiğinde görülür ki gerçek sınıfı sol olduğu halde tahmin edilen sınıfı yukarı olarak etiketlenen 1 sinyal, gerçek sınıfı aşağı olmasına karşın, yukarı sınıfına ait öngörülen 7 sinyal ve son olarak gerçek sınıfı yukarı olsa dahi aşağı sınıfının elemanı olarak etiketlenen 5 sinyal vardır. Bu örnek çalışma üzerinde toplamda 13 sinyalin sınıfı yanlış tahmin edilmiştir.

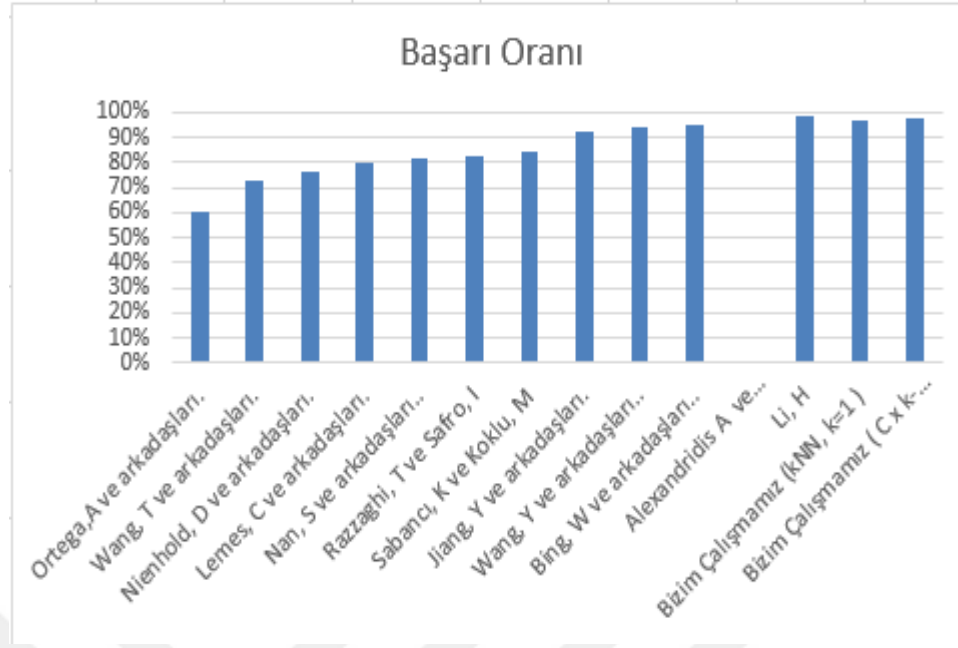
## BÖLÜM ALTI

### DEĞERLENDİRME VE SONUÇLAR

Tablo 5.9’da görüldüğü gibi test verilerinin yoğunluğu artırılıp öğrenme kapasitesinin düşürülerek sınıflandırma işleminin zorlaştırılmasına rağmen alınan sonuçlarının başarı yönünden umut verici olduğu tespit edilir. Bu düzeyde elde ettiğimiz başarı oranı oldukça tatmin edicidir. Ancak, bu düzeyde iyileştirme sağlamadan önce seminer sunumu sırasında paylaştığımız sınıflandırma başarısı bugünkü seviyemize göre bir hayli aşağıdaydı. Seminer sunumu öncesi kendi hazırladığımız kodlarla çalıştırdığımız sınıflandırma algoritmaları *K – En Yakın Komşuluk*, *C x k – En Yakın Komşuluk* ve *Naive Bayesian* yöntemleriydi. İlk deneme süreçlerinde, *10 Parça Çapraz Doğrulama*, verilerin homojen ayrımı, normalizasyon, bozuk verilerin ihracı gibi ön işlemleri gerçekleştirmemiş ve %28’i geçemeyen bir başarı oranı yakalamıştık. Bu düşük başarı oranına sebep olarak bizim ön işleme sürecinde kullanılmaktan eksik kaldığımız düzeltmeler dışında bir de veri alımı sürecinde sinyallere karışan gürültülerin ve elektrot uçlarındaki sensörlerin fazlasıyla hatalı ölçümlere sebep olduğunu öğrendik. Tüm bu bilgiler ışığında algoritmalarımıza ve işlemlerine olan güveni test etmek adına onaylanmış bir benchmark veri seti olan *EEG Göz Durumu* verileriyle çalıştık ve danışmanımız tarafından önerilen *C \* K – En Yakın Komşuluk* Algoritmasının diğer yöntemlere göre fazlasıyla başarılı olduğunu gördük. Bu başarı ikili sınıflandırma için elverişliydi. Aldığımız sonuçları *Weka* veri madenciliği yazılımındaki algoritmalarla karşılaştırdık ve *C x K - NN* metodunun üstünlüğü gördük. Bu aşamadan sonra aynı veri setiyle literatürde yer alan ve yüksek başarı elde etmiş çalışmalarını inceledik. Bu çalışmalarda farklı sınıflandırma algoritma kullanımlarının yanı sıra sonuç iyileştirme adına kullanılan farklı özellik azaltım (*feature reduction*) uygulamaları da yer almıştı. Kendi çalışmamızda bunlara benzer farklı bir istatistiksel özellik azaltım yöntemi ya da filtreleme işlemleri uygulamamamıza karşın aldığımız sonuçlar bu karşılaştırmalar arasında dahi en iyiler arasındaydı. Yapılan bu literatür karşılaştırmasıyla ilgili veriler Tablo 6.1’de yansıtılmıştır.

Tablo 6.1 EEG Göz Durumu verileriyle yapılan bazı çalışmaların başarı oranları

Yazarlar	Başarı Oranı	Yayın Yılı	Algoritma Türü
Ortega, A ve arkadaşları. (Ortega, Crammer ve Lee, 2015)	%60	2015	Makine Öğrenimi
Wang, T ve arkadaşları. (Wang, Guan, Man ve Ting, 2014)	%73	2014	Yapay Sinir Ağları
Nienhold, D ve arkadaşları. (Nienhold, Schwab, Dornberger ve Hanne, 2015)	%76	2015	Yapay Sinir Ağları
Lemes, C ve arkadaşları. (Lemes, Silva ve Batista, 2014)	%80	2014	En Yakın Komşuluk
Nan, S ve arkadaşları. (Nun, Sun, Chen, Li ve Toh, 2015)	%82	2015	DVM
Razzaghi, T ve Safro, I (Razzaghi ve Safro, 2014)	%83	2014	DVM
Sabancı, K ve Koklu, M (Sabancı ve Koklu, 2015)	%84	2015	En Yakın Komşuluk
Jiang, Y ve arkadaşları. (Jiang, Zac ve Lu, 2015)	%92	2015	MDSM
Wang, Y ve arkadaşları. (Wang ve diğerleri, 2015)	%94	2015	Regresyon
Bing, W ve arkadaşları. (Bing, Yao-hua ve Xiao-Hong, 2013)	%95	2013	Yapay Sinir Ağları
Alexandridis A ve Chondrodima, E	%96.8	2014	Benzetimli Tavlama
Li, H (Li, 2015)	%99	2015	En Yakın Komşuluk
<b>Bizim Çalışmamız (kNN, k=1)</b>	<b>%97.2</b>	<b>2016</b>	En Yakın Komşuluk
<b>Bizim Çalışmamız (C x k-NN, k=2)</b>	<b>%97.5</b>	<b>2016</b>	En Yakın Komşuluk



Şekil 6.1 EEG Göz Durumu verileriyle yapılan bazı çalışmaların başarı oranlarının karşılaştırmalı grafiği

Tablo 6.1 ve Şekil 6.1’de görüldüğü üzere 14 kanallı EEG verilerinin göz durumuna göre ikili sınıflandırma işlemi için verisetinin örnek tabanlı sınıflandırmaya uygun olduğu ve ayrıca  $C \times k - NN$  algoritmasının klasik k-NN metoduna göre kısmi iyileşmiş sonuç verdiği görülmüştür. Bunun yanısıra yine 14 kanallı ölçüm yaparak ulaştığımız EEG sinyallerinin çoklu sınıflandırması için de kendi çalışmamızda benzer sonuçlar elde etmemiz kendi verisetimiz için algoritmanın çoklu sınıflandırmaya uygunluğunu göstermiştir.

Gerçekleştirmiş olduğumuz veri alım süreci, bize veri kaydındaki hassasiyetin önemini göstermiştir. İleri ki çalışmalarımızda farklı düşünce ya da beyin-el-ayak koordinasyonuna yönelik çalışmalarımızda veri alım aşamasında sinyalin sınıflandırılmasını etkileyecek gürültülerin giderilmesi işlemleri üzerine veri setimizin hassasiyetine göre daha etkili işlemler yapacağız. Farklı sayıda kanaldan değer alan EEG sinyalleri için sınıflandırma algoritmalarının gösterdiği başarı değişimini inceleyeceğiz. Sinyallerin tam anlamıyla kullanılabilmesi adına gerekli olması olası verisetleri için filtreleme ve sinyal dönüşüm yöntemleri üzerine çalışacağız. Bu aşamaya kadar ki yaptığımız tüm çalışmalar sonucu vardığımız kanı uyguladığımız

sınıflandırma algoritması  $C \times K - NN$ 'nin *EEG* sinyalleri için umut verici olduğu yönündedir. Bu durum bizi gelecekte hakkında hiç fikir sahibi olmadığımız sinyallerin sınıfını doğru tespit eden uygulamalar geliştirme yönünde teşvik eder. Daha hızlı ve daha kesin doğrulukla işlem yapan *BBA* uygulamaları ortaya koymak üzere motive etmektedir.





## KAYNAKLAR

Alexandridis, A. ve Chondrodima, E., (2014). A medical diagnostic tool based on Radial Basis Function Classifiers and Evolutionary Simulated Annealing. *Journal of Biomedical Informatics*, 61-72.

Aydemir, Ö. (2009). EEG tabanlı beyin bilgisayar arayüzleri. *Akademik Bilişim '09 - XI. Akademik Bilişim Konferansı Bildirileri*, 8-10.

Aydemir, Ö., ve Kayıkçıoğlu, T. (2011). Classification of EEG signals recorded during right/left eye movement imagery using Fourier Transform based features. Signal Processing ve Communications Applications (SIU), *IEEE 19th Conference on IEEE*, 415-418.

Bing, W., Yao-hua, M. ve Xiao-hong, Y. (2013). Radial Basis Function Process Neural Network Training based on Generalized Frechet Distance ve GA-SA Hybrid Strategy. *Computer Science ve Engineering : An International Journal (CSEIJ)*, 3(6), December doi : 10.5121/cseij.2013.3601.

Doğan, A, ve Nasibov, E. (2016). Classification of EEG eye state signals using C x K – Nearest Neighborhood Algorithm. *In Proceedings of International Conference on Computer Science ve Engineering*, 20-23 October 2016, Tekirdağ, Turkey, 696-701.

Emotiv. (2016). Retrieved February, 13, 2016, from <https://www.emotiv.com/product/emotiv-epoc-14-channel-mobile-eeeg/>.

He, B. (2005). *Neural engineering* (2. Baskı) (87-97). Minnesota: Kluwer Academic / Plenum Publishers.

- Jiang, Y., Zhao, Q., ve Lu. Y. (2015). Adaptive ensemble with human memorizing characteristics for Data Stream Mining,. *Mathematical Problems in Engineering*, Article ID 874032,18 May (2015), doi:10.1155/2015/874032.
- Lemez C, Silva D, ve Batista G. (2014). Adding diversity to rank examples in anytime nearest neighbor classification. *13th International Conference on Machine Learning ve Applications*, 3-6 December (2014), IEEE Computer Society, 129-134.
- Li H. (2015). Accurate and efficient classification based on Common Principal Component Analysis for Multivariate Time Series. *Elsevier, Neurocomputing*, 171, 744-753.
- Nan S, Sun L, Chen B, Lin Z, ve Toh K. (2015). Density-Dependent Quantized Least Squares Support Vector Machine for large data sets. *IEEE Transactions ve Neural Networks ve Learning Systems*, 28, 94-106.
- Nasibov E., Özgören M., Ulutagay G., Öñiz A. ve Kocaaslan S. (2010), On the analysis of BIS stage epochs via fuzzy clustering. *Biomedizinische Technik/Biomedical Engineering*, 55(3), 147–153.
- Nasibov E. ve Ulutagay G. (2010). Comparative clustering analysis of bispectral index series of brain activity. *Expert Systems with Applications*, 37, 2495-2504.
- Nienhold D, Schwab K, Dornberger R, ve Hanne T. (2015). Effects of weight initialization in a Feedforward Neural Network for classification using a modified generic algorithm. *3rd International Symposium on Computational ve Business Intelligence (ISCBI)*, 6-12.
- Ortega P, Crammer K, ve Lee D. (2015). Belief flows for robust online learning. *Information Theory ve Applications(ITA)*, 1-6 February (2015), 70-77.

- Razzaghi T ve Safro I. (2014). Fast multilevel support vector machines. *arXiv, abs/1410.3348*, October (2014).
- Roesler, O. (2013). Retrieved June, 20, 2016, from <https://archive.ics.uci.edu/ml/datasets/EEG+Eye+State#>.
- Sabancı K ve Koklu M. (2015). The classification of eye state by Using kNN ve MLP classification models according to the eeg signals. *International Journal of Intelligent Systems ve Applications in Engineering*, 3, December (2015), 127-130.
- Tülay, E. (2009). *Beyin elektriksel aktivitesinin ölçümü ve sinyal analizi*. Yüksek Lisans Tezi, İstanbul Kültür Üniversitesi, İstanbul.
- Ulutağay G ve Nasibov, E. (2016). C x K Nearest Neighbor Classification with ordered weighted average distance, *Novel Applications of Intelligent Systems*, 586, March (2016), 105-122.
- Ulutağay G ve Nasibov E. (2012). A New C x K Nearest Neighbor linkage approach to the classification problem. *In Proceedings of the 10th International FLINS Conference*, Istanbul, Turkey, 1, 27-29 August (2012), 471-476.
- Ulutağay G. ve Nasibov E. (2009). Detection of BIS stage levels via fuzzy clustering approach. *Biomedical Engineering Meeting (BIYOMUT), IEEE Xplore Digital Library*, doi: 10.1109/BIYOMUT.2009.5130356, 1-4.
- Wang T, Guan S, Man K, ve Ting T. (2014). EEG eye state identification using incremental attribute learning with time-series classification. *Mathematical Problems in Engineering*, Article ID 365101, doi:10.1155/2014/365101
- Wang T, Guan S, Man K, ve Ting T. (2014). Time series classification for EEG Eye State identification based on incremental attribute learning. *2014 International*

*Symposium on Computer, Consumer ve Control*, Taichung, Taiwan, 10-12 June (2014), 158-161.

Wang Y, Li Y, Xiong M, ve Jin L. (2015). Random Bits Regression: a strong general predictor for big data. in eprint arXiv:1501.02990.



## **EKLER**

### **EK-1: Tez sonuçlarından üretilen çıktılar**

1. Dođan, A, and Nasibov, E. (2016), Classification of EEG Eye State Signals Using C x k – Nearest Neighborhood Algorithm. In Proceedings of International Conference on Computer Science and Engineering, 20-23 October 2016, Tekirdađ, Turkey, 696-701.
2. Dođan, A, and Nasibov, E. (2017), An Efficient Algorithm for Classification of EEG Eye State Data. In Proceedings of 7th World Conference On Innovation And Computer Science (Insode-2017), 20-22 April 2017, Pristine, Kosovo, 26.
3. Dođan, A, and Nasibov, E. (2017), An Efficient Algorithm for Classification of EEG Eye State Data. Global Journal of Information Technology: Emerging Technologies, 6(3), 158-165.