

**KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**





KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsünde

Unvanı Verilmesi İçin Kabul Edilen Tezdir.

Tezin Enstitüye Verildiği Tarih : / /

Tezin Savunma Tarihi : / /

Tez Danışmanı :

Trabzon

ÖNSÖZ

Bu tez çalışmasında desteğini benden esirgemeyen, önerileriyle ufkumu açan danışmanım ve saygıdeğer hocam sayın Prof. Dr. Temel KAYIKÇIOĞLU'na en içten teşekkürlerimi sunarım. Son olarak hayatım boyunca beni sabır ve şefkatle destekleyen aileme sonsuz teşekkürlerimi belirtir, çalışmanın faydalı olmasını dilerim.

Taner YURDUSEVER

Trabzon 2021



TEZ ETİK BEYANNAMESİ

Yüksek Lisans Tezi olarak sunduğum “Çeşitli Aktiviteler Sırasında Kaydedilmiş EEG ve EMG İşaretlerinin Sınıflandırılması” başlıklı bu çalışmayı baştan sona kadar danışmanım Prof. Dr. Temel KAYIKÇIOĞLU ‘un sorumluluğunda tamamladığımı, verileri/örnekleri kendim topladığımı, deneyleri/analizleri ilgili laboratuvarlarda yaptığımı/yaptırdığımı, başka kaynaklardan aldığım bilgileri metinde ve kaynakçada eksiksiz olarak gösterdiğimi, çalışma sürecinde bilimsel araştırma ve etik kurallara uygun olarak davrandığımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul ettiğimi beyan ederim. 28/07/2021

Taner YURDUSEVER

İÇİNDEKİLER

	<u>Sayfa No</u>
ÖNSÖZ	III
TEZ ETİK BEYANNAMESİ.....	IV
İÇİNDEKİLER.....	V
ÖZET	VIII
SUMMARY	IX
ŞEKİLLER DİZİNİ	X
TABLolar DİZİNİ.....	XI
KISALTMALAR DİZİNİ	XII
1. GENEL BİLGİLER.....	1
1.1. Giriş.....	1
1.2. Elektroensefalografi (EEG) Nedir?	5
1.2.1. EEG Yöntemiyle Kaydedilen Dalga Çeşitleri.....	6
1.2.2. EEG Sinyallerinin Elde Edilmesi	7
1.2.3. Bozucu Etkiler.....	8
1.2.4. EEG Yönteminin Avantajları	8
1.3. Elektromiyografi (EMG) Nedir?	9
1.3.1. Motor Üniteleri ve Aksiyon Potansiyeli.....	9
1.3.2. EMG Gürültü Kaynakları	10
1.4. Makine Öğrenmesi Nedir?	11
1.5. Sınıflandırma Algoritmaları	12
1.5.1. K-En Yakın Komşular Yöntemi.....	12
1.5.2. Destek Vektör Makineleri (DVM)	13
1.5.2.1. Doğrusal DVM.....	13
1.5.2.2. Çekirdek Fonksiyonları ile Sınıflandırma	18
1.5.2.3. Çok Sınıflı Verilerde Sınıflandırma	19
1.5.3. Karar Ağaçları	20
1.5.3.1. Bilgi Kazancı.....	20
1.5.3.2. Bilgi Kazancı Oranı.....	21
1.5.3.3. Gini İndeksi	21

1.5.3.4.	Karar Ağacının Oluşturulması.....	22
1.5.4.	Naif Bayes	22
1.5.5.	Yapay Sinir Ağları.....	25
1.6.	Sınıflandırmada Doğrulama Metodları	27
1.6.1.	Hold-Out Metodu	27
1.6.2.	K Katlamalı Çapraz Doğrulama Metodu.....	27
1.1.	Temel Bileşenler Analizi.....	28
1.1.1.	Ortalama Vektörü.....	28
1.1.2.	Standart Sapma ve Varyans.....	29
1.1.3.	Kovaryans Matrisi	29
1.1.4.	Özdeğerler ve Özvektörler	30
1.1.5.	Temel Bileşenler Analizi Uygulaması	30
2.	YAPILAN ÇALIŞMALAR	32
2.1.	Kullanılan Veri Seti.....	32
2.2.	Pencereleme Yöntemleri	36
2.2.1.	Bitişik (Adjacent) Pencereleme.....	36
2.2.2.	Üst Üste (Overlap) Pencereleme	36
2.3.	Dalgacık Dönüşümü.....	38
2.3.1.	Ayrık Dalgacık Dönüşümü.....	38
2.4.	Sınıflandırmada Kullanılan Öznitelik Fonksiyonları	40
2.4.1.	Ortalama Mutlak Değer.....	40
2.4.2.	Hjort Parametreleri	40
2.4.3.	Basıklık.....	41
2.4.4.	Fraktal Boyutlar.....	41
2.4.5.	Sıfır Geçiş Sayısı	41
2.5.	Eğitim Verisinin Normalizasyonu.....	42
2.6.	Başarı Değerlendirme Ölçütleri.....	42
2.6.1.	Doğruluk.....	43
2.6.2.	Doğru Pozitif Oranı (DP Oranı)	43
2.6.3.	Yanlış Pozitif Oranı (YP Oranı).....	44
2.6.4.	Kesinlik	44
2.6.5.	F-Skoru (F-Score).....	44
2.7.	Tez Çalışması Akış Şeması	44

3.	BULGULAR	47
3.1.	Dengelenmemiş Verilerin Sınıflandırma Sonuçları	47
3.1.1.	Uygulama 1	47
3.1.1.	Uygulama 2	53
3.2.	Dengelenmiş Verilerin Sınıflandırma Sonuçları	54
3.3.	Boyutu Azaltılmış Verilerin Sınıflandırma Sonuçları.....	57
3.4.	Bireysel Bazda Yapılan Sınıflandırma Sonuçları.....	59
4.	TARTIŞMA.....	61
4.1.	EEG ve EMG Verilerinin Karşılaştırılması.....	61
4.2.	Kullanılan Özniteliklerin Değerlendirilmesi	62
4.3.	Sınıflandırıcı Performanslarının Karşılaştırılması.....	62
4.4.	Dengeli ve Dengesiz Verilerin Kıyaslanması	62
4.5.	Boyut Azaltmanın Sınıflandırmaya Etkisi	63
4.6.	Bireysel Bazda Yapılan Sınıflandırmanın Değerlendirilmesi	64
5.	SONUÇLAR	65
6.	ÖNERİLER	67
7.	KAYNAKLAR.....	68
8.	EKLER	72
	ÖZGEÇMİŞ	

Yüksek Lisans Tezi

ÖZET

ÇEŞİTLİ AKTİVİTELER SIRASINDA KAYDEDİLMİŞ
EEG VE EMG İŞARETLERİNİN SINIFLANDIRILMASI

Taner YURDUSEVER

Karadeniz Teknik Üniversitesi

Fen Bilimleri Enstitüsü

Elektrik-Elektronik Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. Temel KAYIKÇIOĞLU

2021, 71 Sayfa, 14 Sayfa Ek

Bu tez çalışmasında 11 gündelik aktivitenin yapıldığı esnada eşzamanlı olarak kaydedilmiş EEG ve EMG veri setinin sınıflandırılması hedeflenmektedir. 20 kişiye ait ölçümlerin bulunduğu bu veri seti 1 kanal EEG ile 1 kanal EMG verisinden oluşmaktadır. EEG verilerinin alt bileşenlerini elde etmek için ayrık dalgacık dönüşümü kullanılmıştır. Sonraki aşamada EEG, EMG ve alt bileşenlere ait öznitelikler çıkarılmıştır. Çalışmada kullanılan öznitelikler ortalama mutlak değer, Hjorth değişkenlik parametresi, basıklık, Hurst fraktal indisi ve sıfır geçişleridir. Sınıflandırma aşamasında literatürde yaygın olarak kullanılan; K-En Yakın Komşular, J48, Destek Vektör Makinesi, Yapay Sinir Ağları ve Naif Bayes yöntemleri kullanılmıştır. Sınıflandırma EEG ve EMG verileri ile bu iki veri tipinin birleştirilmesiyle elde edilen veri kümesi üzerinden gerçekleştirilmiştir. En yüksek doğruluk %94.7 ile K-EYK yönteminden, EEG ve EMG öznitelikleri kullanılarak elde edilmiştir. Sınıflandırma performansını iyileştirmek ve gürültü oranını azaltmak amacıyla Temel Bileşen Analizi kullanılmıştır. Böylelikle DVM yönteminden elde edilen başarı %91.3 seviyesine çıkmıştır. Diğer çalışmalara kıyasla, sadece iki kanal elektrot ile kaydedilmiş çok sınıflı bir veri kümesinin yüksek doğrulukla sınıflandırılması ve EEG ve EMG sinyalleri için aynı özniteliklerin kullanılması bu çalışmanın farkını ortaya koymaktadır.

Anahtar Kelimeler: EEG, EMG, KNN, J48, DVM, YSA, Naif Bayes, PCA

Master Thesis

SUMMARY

CLASSIFICATION OF RECORDED EEG AND EMG SIGNALS
DURING VARIOUS ACTIVITIES

Taner YURDUSEVER

Karadeniz Technical University

The Graduate School of Natural and Applied Sciences

Electrical and Electronics Engineering Graduate Program

Supervisor: Prof. Dr. Temel KAYIKÇIOĞLU

2021, 71 Pages, 14 Pages Appendix

In this thesis, it is aimed to classify the EEG and EMG dataset recorded simultaneously during 11 daily activities. This dataset, which includes measurements of 20 people, consists of one channel EEG data and one channel EMG data. Discrete wavelet transform was used to obtain the subcomponents of the EEG signals. In the next step, features of EEG, EMG, and sub-components of EEG signals were extracted. The features used in the study are mean absolute value, Hjorth mobility parameter, kurtosis, Hurst fractal index, and zero crossings. K-Nearest Neighbors (KNN), J48, Support Vector Machine (SVM), Artificial Neural Network (ANN), and Naive Bayes classifiers, which are widely used in the literature, were used in the classification step. EEG and EMG dataset and the dataset was obtained by combining these two data types were used for the classification. The highest accuracy with 94.7% was obtained using the KNN method from the dataset using EEG and EMG features. In addition, Principal Component Analysis was used to improve classification performance and reduce the noise ratio. Thanks to this method, the success obtained from the SVM method increased to 91.3%. Compared to other studies, the high accuracy classification of a multi-class dataset recorded with only two-channel electrodes and using the same features for EEG and EMG signals show the difference of this study.

Key Words: EEG, EMG, KNN, J48, SVM, ANN, Naive Bayes, PCA

ŞEKİLLER DİZİNİ

	<u>Sayfa No</u>
Şekil 1. EEG dalga türleri [17].	6
Şekil 2. Elektrotların yerleşim düzeni (10-20 sistemi) [18].	7
Şekil 3. Aksiyon potansiyeli [28].	10
Şekil 4. Makine öğrenmesinde izlenecek adımlar	11
Şekil 5. Uzaklık fonksiyonlarının geometrik ifadeleri	12
Şekil 6. Destek vektör makineleri yöntemi	14
Şekil 7. YSA katmanları [33].	26
Şekil 8. Perceptronun yapısı	26
Şekil 9. Çapraz doğrulama metodu.....	28
Şekil 10. Deney düzeneği	32
Şekil 11. İlk 5 aktivitenin EEG kaydı (Katılımcı 8 solda, Katılımcı 9 sağda)	33
Şekil 12. Son 6 aktivitenin EEG kaydı (Katılımcı 8 solda, Katılımcı 9 sağda)	34
Şekil 13. İki katılımcıya ait EMG kayıtları (Katılımcı 8 solda, Katılımcı 9 sağda).....	35
Şekil 14. Bitişik pencereleme yöntemi	36
Şekil 15. Üst üste pencereleme yöntemi.....	37
Şekil 16. Hamming fonksiyonu zaman ve frekans domeni görüntüsü [36].	37
Şekil 17. 1000 Hz'lik bir sinyalin alt bantlarına ayrışması	38
Şekil 18. Daubechies dalgacık ailesi [38]......	39
Şekil 19. Tez çalışması akış şeması.....	46
Şekil 20. Sınıfların dağılımı	48
Şekil 21. EEG veri seti özneliklerine ait istatistikler.....	49
Şekil 22. EMG veri seti özneliklerine ait istatistikler	51
Şekil 23. Uygulama 1 EEG+EMG özelliklerinin Pearson korelasyonu	52
Şekil 24. Uygulama 1 EEG+EMG verisine ait KNN karmaşıklık matrisi	53
Şekil 25. Dengeli sınıfların dağılımı.....	55
Şekil 26. Boyutu azaltılmış EEG+EMG verisine ait DVM karmaşıklık matrisi.....	59

TABLolar DİZİNİ

	<u>Sayfa No</u>
Tablo 1. Deney aktiviteleri ve sınıf numaraları.....	32
Tablo 2. EEG verisine ait istatistiksel özellikler	33
Tablo 3. EMG verisine ait istatistiksel özellikler	34
Tablo 4. EEG sinyalinin dalgacık dönüşümü ile elde edilen frekans bantları.....	39
Tablo 5. Karmaşıklık matrisi (Confusion matrix)	43
Tablo 6. Uygulama 1 için kullanılan öznitelikler	48
Tablo 7. Uygulama 1 EEG verisine ait sınıflandırma sonuçları.....	50
Tablo 8. EEG alt bantlarına ait sonuçlar (J48)	50
Tablo 9. Uygulama 1 EMG verisine ait sınıflandırma sonuçları.....	51
Tablo 10. Uygulama 1 EEG+EMG verisine ait sınıflandırma sonuçları.....	52
Tablo 11. Uygulama 2 için kullanılan öznitelikler	54
Tablo 12. Uygulama 2 EEG+EMG verisine ait sınıflandırma sonuçları.....	54
Tablo 13. Dengeli EEG verisine ait sınıflandırma sonuçları.....	55
Tablo 14. Dengeli EMG verisine ait sınıflandırma sonuçları.....	56
Tablo 15. Dengeli EEG+EMG verisine ait sınıflandırma sonuçları.....	56
Tablo 16. Boyutu azaltılmış EEG verisine ait sınıflandırma sonuçları	57
Tablo 17. Boyutu azaltılmış EMG verisine ait sınıflandırma sonuçları	58
Tablo 18. Boyutu azaltılmış EEG+EMG verisine ait sınıflandırma sonuçları	58
Tablo 19. Birey bazlı sınıflandırma sonuçları (9 aktivite).....	59
Tablo 20. Birey bazlı sınıflandırma sonuçları (10 aktivite).....	60
Tablo 21. Birey bazlı sınıflandırma sonuçları (11 aktivite).....	60

KISALTMALAR DİZİNİ

ADD	Ayrık Dalgacık Dönüşümü
BBA	Beyin Bilgisayar Arayüzü
DVM	Destek Vektör Makineleri
EEG	Elektroensefalografi
EKG	Elektrokardiyografi
EMG	Elektromiyografi
KNN	K-Nearest Neighbours
MÜAP	Motor Ünite Aksiyon Potansiyelleri
PCA	Principle Component Analysis
TBA	Temel Bileşen Analizi
WEKA	Waikato Environment for Knowledge Analysis
YSA	Yapay Sinir Ağı

1. GENEL BİLGİLER

1.1. Giriş

Merkezi sinir sistemi zihinsel ve fiziksel fonksiyonları düzenlemekten ve kontrol etmekten sorumlu karmaşık bir yapıdır. Beynimiz dışarıdan veya içeriden aldığı uyarıları işler ve bir cevap iletisi oluşturur. Daha sonra bu ileti omurilik aracılığıyla ilgili organa veya kas grubuna iletilir. Gündelik yaşantımızda bilinçli veya bilinçsiz bir şekilde yaptığımız davranışların kaynağı beyin tarafından üretilen ve sinir hücreleri tarafından iletilen elektriksel işaretlerdir. Dolayısıyla bu sinyaller elektrotlar vasıtasıyla ölçülerek sayısal verilere dönüştürülebilmektedir. Elektrotların bağlandıkları bölgelere göre bu sinyaller farklı isimler almaktadır. Eğer ölçüm kafa derisi üzerinden yapılıyorsa bu sinyallere “Elektroensefalografi (EEG)”, kas üzerinden yapılıyorsa “Elektromiyografi (EMG)” sinyali olarak adlandırılır. Biyolojik sistemlerin ve ölçüm tekniklerinin farklı olması sebebiyle bu sinyaller farklı özelliklere sahiptir. Elde edilen veriler karmaşık ve hassas yapıda olmasına rağmen içerdiği karakteristik özellikler sayesinde çeşitli davranış örüntüleri belirlenebilir.

EEG sinyallerinin hassas yapıda ve küçük genlikte olması, elde edilen ölçümlerden anlamlı verilerin çıkarılmasını zorlaştırır [1]. Ancak bu konuda yapılmış çalışmalar giderek gelişme göstermektedir. Örneğin Akbari vd. yaptıkları bir deneyde sayıların okunuşları dinletilen katılımcılara ait EEG verilerinden, dinledikleri sesleri anlaşılır bir şekilde yeniden oluşturmayı başarmışlardır [2]. Diğer bir çalışmada Koçyiğit vd. [3] 29 gönüllü katılımcıya ait otuz kanallı EEG verisi üzerinde zihinden yapılan aritmetik işlemler ile nötr durumu sınıflandırmayı amaçlamışlardır. EEG verisi 6 seviyeli Ayrık Dalgacık Dönüşümü (ADD) ile alfa, beta, delta, teta ve gama alt frekans bantlarına ayrıştırılmıştır. Bu ön işlem adımından sonra her bir EEG alt bandına ait Ortalama Enerji, Standart Sapma, Basıklık ve Varyans öznitelikleri çıkarılmıştır. Sınıflandırma işlemi Weka (Waikato Environment for Knowledge Analysis) programı kullanılarak yapılmıştır. Sınıflandırma işlemi J48 algoritması kullanılarak 10 kat çapraz doğrulama ile yapılmıştır. Sınıflandırma sonucunda %98.2 oranında sınıflandırma doğruluğu hesaplanmıştır.

Onay doktora tezinde [4], engelli insanların zihinsel ve motor hareket hayal etme görevlerinin EEG verileri üzerinden sınıflandırılması ve bu aktiviteleri insanların en çok ihtiyaç duyduğu kelimelerle eşleştirmeye yönelik bir çalışma yürütmüştür. Çalışmada 8 katılımcının EEG ölçümleri 8 kanal üzerinden alınmıştır. Aktiviteler seçilirken beyin farklı

bölgelerini harekete geçirecek olmasına dikkat edilmiştir. Bu sayede sınıflandırmanın kolaylaştırılması amaçlanmıştır. Bu aktivitelerin çoğu fiziksel kaynaklıdır. Örneğin alkış, dil yuvarlama ve sağ-sol bilek çevirme gibi. Sınıflandırmada farklı öznitelikler K-En Yakın Komşuluk (K-Nearest Neighbours veya KNN), Destek Vektör Makineleri (DVM) ve Yapay Sinir Ağı (YSA) algoritmaları ile sınıflandırılmıştır. Tez çalışmasında bireysel bazda yapılan sınıflandırmalarda yüksek başarı elde eden katılımcılar olsa da tüm katılımcıları kapsayan genel sınıflandırma sonuçları çok daha düşüktür. Bu sebeple sistemin kişi bazlı olması önerilmiştir. Ayrıca aktivite sayısı giderek arttırıldığında elde edilen sınıflandırma sonuçları da düşmektedir. Örneğin bir adımda aktivitelerin ikili sınıflandırılmasında KNN yönteminde %63-93 arası başarı elde edilirken, aktivite sayısı 5'e çıkarıldığında başarının %19-44 aralığına kadar düştüğü gözlemlenmiştir. Ancak çalışma çok sınıflı bir problem üzerinde durduğu için yenilikçidir.

Yapılan çalışmalar fiziksel aktivitelerin EEG üzerinde çeşitli yapaylıklar oluşturduğunu ve bunların yüksek doğrulukla tespit edilebileceğini göstermektedir. Böylece EEG verileri Beyin Bilgisayar Arayüzü (BBA) uygulamaları sayesinde robotik sistemlerin düşünce ile kontrol edilebilir olmasını sağlayabilir ve engelli insanların hayatlarını kolaylaştırabilir. Tosun vd. [5] katılımcıların ellerini fiziksel ve hayali olarak hareket ettirdiği esnada kaydedilmiş, 500 Hz örnekleme frekanslı 19 kanallı EEG verisi üzerinde bu iki tür hareketi sınıflandırmaya çalışmışlardır. Sınıflandırma öncesinde EEG verilerini şebeke gürültüsünden arındırmak için 50 Hz'lik çentik (notch) filtre kullanılmıştır. EEG verilerinin her kanala ait 1-48 Hz arasındaki frekanslarının güç yoğunlukları hesaplanmış ve ortalaması alınmıştır. Elde edilen veri seti tasarlanan Yapay Sinir Ağında sınıflandırılmış, %99.9 doğruluk elde edilmiştir. Başçıl'ın çalışmasında [6] 10 katılımcının EEG ile kayıt altına alınmış sağ-sol çene hareketleri arasındaki ilişki incelenmiştir. YSA ile yapılan sınıflandırmada bu hareketlerin beyin üzerinde oluşturduğu iki farklı EEG sinyalinin %90,14 oranında birbirinden ayrılabilir olduğu bulunmuştur.

Biyolojik sinyallerin sınıflandırılmasında kullanılan öznitelikler sınıflandırma performansını doğrudan etkiler. Literatürde en çok kullanılan öznitelikler verinin en büyük, en küçük değerini bulma veya ortalamasını alma gibi zaman tabanlı veya standart sapma, çarpıklık, basıklık hesaplama gibi istatistik tabanlı yaklaşımlardır. Öznitelik çıkarımında farklı yöntemlerin kullanıldığı çalışmalar da mevcuttur. Örneğin Aydın'ın çalışmasında [7] çoklu ve eşzamanlı görevlerin yerine getirilmesi sırasında kaydedilmiş EEG sinyallerinden zihinsel iş yükü seviyelerinin sınıflandırılması için, fraktal boyut algoritmalarına dayalı

öznitelik çıkarma yöntemleri önerilmiştir. Bu yöntemler 48 katılımcı tarafından puanlanan zihinsel iş yükünün düşük, orta ve yüksek seviye olarak sınıflandırılması için kullanılmıştır. Fraktal boyutlar verinin karmaşıklığını ölçen parametreler olarak tanımlanabilir. Öznitelik çıkarma yöntemi olarak Katz ve Higuchi fraktal boyut algoritmaları kullanılmıştır. Sınıflandırma için DVM, KNN ve Kuadratik Ayırtaç Analizi yöntemleri kullanılmıştır. DVM yöntemi bire karşı diğerleri yaklaşımı ile radyal tabanlı çekirdek fonksiyonu kullanılarak oluşturulmuş ve %95.3 e varan doğrulukla en yüksek başarı Higuchi fraktal boyutu yönteminden elde edilmiştir.

Fiziksel aktivitelerin sınıflandırılmasında EMG temelli çalışmalar da bulunmaktadır. Çalışkan'ın [8] çalışması, 14 farklı parmak hareketine ait EMG verilerini sınıflandırmada kullanılabilecek daha hızlı ve daha yüksek doğruluk oranına sahip yeni bir öznitelik çıkarma yöntemi sunmasıyla ilgilidir. Sınıflandırma aşamasından önce ham veriler Butterworth bant geçiren filtre ile 20-450 Hz aralığına filtrelenmiş ve sinyal üst üste pencereleme işlemiyle daha küçük parçalara ayrılmıştır. Sınıflandırma aşaması için ortalama mutlak değer ile dalga formu uzunluğu özniteliklerinden elde edilen vektörler normalize edildikten sonra belli bir katsayı oranında toplanarak karma bir öznitelik oluşturulmuştur. Daha sonra zaman ve istatistik tabanlı diğer özniteliklerden elde ettiği sonuçlar ile karşılaştırılması yapılmıştır. Sınıflandırma işleminde KNN algoritması 10 kat çapraz doğrulama ile kullanılmıştır. En yüksek başarı oranı %97,48 ile karma yöntemden sağlanmıştır. Elde edilen sonuçlar aynı zamanda çarpıklık, basıklık, standart sapma gibi istatistik tabanlı özniteliklerin ortalama mutlak değer, sıfır geçiş sayısı, mutlak fark toplamı gibi zaman tabanlı yöntemlere göre başarı ve işlem süresi açısından geride kaldığını göstermiştir.

Literatürde EEG ve EMG işaretlerinin birlikte kullanıldığı sınıflandırma çalışmalarının pek yaygın olmadığı görülmektedir. Larraz vd. [9] felçli insanların hareket niyetlerini tespit etmek amacıyla 20 hastaya ait EEG (16 kanal) ve EMG (4 kanal) verisini incelemiştir. Hastalardan ilk görevde bir elini hareket ettirmeleri istenirken sonraki görevde hareketsiz kalmaları istenmiştir. Daha sonra üç veri grubu (EEG, EMG ve EEG ile EMG) için öznitelik vektörleri hazırlanmıştır. Kullanılan öznitelikler 3 kanal EEG verisinin alfa ve beta bantlarının ortalama gücü ve 2 kanal EMG verisinin dalga formu uzunluklarıdır. Üç veri grubunda hareket niyetinin olup olmadığı Doğrusal Ayrımcılık Analizi ile sınıflandırılmıştır. Sınıflandırma sonuçlarına göre en yüksek doğruluk oranı %69.5 ile hem EEG hem de EMG özniteliklerinin kullanıldığı veri setinden elde edilmiştir.

Ayrıca EMG özniteliklerinin EEG özniteliklerine göre daha yüksek doğruluk oranı sağladığı gözlemlenmiştir.

Xie ve arkadaşları [10] da benzer bir çalışma yürütmüşlerdir. 10 felçli, 10 sinir zedelenmesi olan ve 10 sağlıklı insanın katıldığı deneyde, bu kişilerden dizlerini bükmeye çalışmaları daha sonra kaldırmaya çalışmaları istenmiştir. Bu esnada eşzamanlı olarak EEG (14 kanal) ve EMG (2 kanal) kayıtları alınmıştır. Çalışmada bu iki aktivitenin sınıflandırılması amaçlanmıştır. Bir önceki çalışmaya benzer olarak üç veri grubu oluşturulmuştur. Bu gruplar sırasıyla EEG öznitelikleri, EMG öznitelikleri ve EEG ile EMG öznitelikleri birlikte kullanılarak hazırlanmıştır. Daha sonra bu üç grubun frekans temelli öznitelikleri çıkartılmıştır. Sınıflandırma bir tür YSA uygulaması olan Aşırı Öğrenme Makineleri ile yapılmıştır. Üç katılımcı grubunda da en yüksek doğruluk %98.9-85.3 aralığında olup EEG ile EMG özniteliklerinin kullanıldığı veri setinden elde edilmiştir. Bu başarıyı EMG özniteliklerinin kullanıldığı veri seti takip etmiştir. Sonuç olarak katılımcıdan bağımsız olarak EEG ve EMG verilerinin birlikte kullanımı sayesinde daha başarılı BBA sistemleri geliştirilebileceği ifade edilmiştir.

Makine öğrenmesinde bir diğer konu sınıflandırma verimini artırmak için verilerden gereksiz kısımların atılmasıdır. Bu amaçla boyut azaltma yöntemleri geliştirilmiştir. Literatürde en yaygın olan yöntem Temel Bileşenler Analizi veya Principle Component Analysis (PCA) dır. Erin ve Boru [11] EMG ölçümü yapabilen bir bileklik yardımıyla endüstriyel robotun kontrolünü sağlayacak bir model üzerinde çalışmışlardır. EMG sinyalleri ilk önce ön işlemden geçirilmiş ve PCA yöntemi ile boyutu azaltılmıştır. El hareketleriyle yapılan üç komuta ait EMG verileri Random Forest algoritmasıyla sınıflandırılmış ve %95.9 başarı oranı sağlanmıştır.

Bu tez çalışmasında 11 gündelik aktivitelerin yapıldığı esnada eşzamanlı olarak kaydedilmiş EEG ve EMG veri setinin sınıflandırılması hedeflenmektedir. 20 kişiye ait ölçümlerin bulunduğu bu veri seti 1 kanal EEG verisi ile çeneye bağlı 1 kanal EMG verisinden oluşmaktadır. Bu çalışmayı diğer çalışmalardan ayıran en önemli özelliği ise 11 aktiviteyi sınıflandırmayı hedeflemesidir. Genellikle EEG verilerinin hastalık tahmini (var ya da yok) veya duygu durum tahmini (düşük ya da yüksek derecede) çalışmalarında olduğu gibi ikili sınıflandırma problemlerinde kullanıldığı görülmektedir. Nispeten fiziksel ve zihinsel aktivite tanıma çalışmaları çok daha az sayıdadır. Ayrıca bu çalışmalarda çok kanallı kayıtlar kullanılmakta ve bu sayede ölçümlerde yüksek çözünürlük sağlanırken aynı zamanda sınıflandırma başarısına göre kanallar arasında tercih yapma imkânı da

bulunmaktadır. Benzer bir şekilde EMG üzerinde yapılan çalışmalarda da ölçüm alınan kanal sayısının fazla olması tercih edilmektedir. Ancak bu çalışmada kullanılan EEG ve EMG kayıtları tek kanallı verilerdir. Fiziksel hareketlerin iki kanal üzerinden sınıflandırılması çözünürlük ve gürültü faktörleri açısından dezavantaj gibi görünse de ölçümlerin pratik bir şekilde yapılması açısından bir avantajdır. Dolayısıyla bu çalışmadan elde edilen veriler pratik Beyin Bilgisayar Arayüzü (BBA) uygulamalarında kullanılabilir. Örneğin engelli veya hasta insanların çeşitli cihazları kontrol edebilmesi sağlanabilir.

Tez çalışmasında özniteliklerin hazırlanmasında MATLAB, sınıflandırılmasında ise Weka programlarının kullanılmıştır. Ham veriler sınıflandırma aşamasından önce hat gürültüsünden arındırıldıktan sonra pencereleme işlemi ile küçük parçalara ayrılmıştır. EEG işaretlerinden alfa, beta, gama, delta ve teta bileşenlerini elde etmek için ayrık dalgacık dönüşümü kullanılmıştır. Sonraki aşamada EEG, EMG ve alt bileşenlere ait öznitelikler çıkarılmıştır. Çalışmada kullanılan öznitelikler ortalama mutlak değer, Hjorth değişkenlik parametresi, basıklık, Hurst fraktal indisi ve sıfır geçişleridir. Bu öznitelikler hem EEG hem de EMG verileri üzerinde yüksek sınıflandırma doğruluğu verecek şekilde seçilmişlerdir. Elde edilen veri setleri normalize edilerek sınıflandırılmaya uygun hale getirilmiştir. Sınıflandırma aşamasında literatürde yaygın olarak kullanılan KNN, J48, DVM, YSA ve Naif Bayes sınıflandırıcıları kullanılmıştır. En yüksek doğruluk %94.7 ile KNN yönteminden, EEG ve EMG özniteliklerinin kullanıldığı veri setinden elde edilmiştir. En düşük doğruluk oranı Naif Bayes yönteminden elde edilmiştir. Genel olarak J48 algoritmasının KNN yöntemine yakın sonuçlar verdiği gözlemlenmiştir. EEG alt bileşenlerine ait sınıflandırma sonuçlarının dönüştürülmemiş EEG verisine kıyasla daha düşük olduğu gözlemlenmiştir. Ayrıca sınıflandırma performansını iyileştirmek ve gürültü oranını azaltmak amacıyla literatürde de sıkça tercih edilen PCA yöntemi kullanılmıştır. PCA yöntemi sayesinde DVM yönteminden elde edilen başarı %22 artarak %91.3 seviyesine çıkmıştır.

1.2. Elektroensefalografi (EEG) Nedir?

Elektroensefalografi (EEG), beyin aktivitesi sonucu ortaya çıkan elektriksel işaretlerin elektrotlar vasıtasıyla ölçülmesi yöntemidir. EEG işaretlerinin genliği birkaç yüz mikro volt arasında değişmektedir [12]. EEG sinyallerinin analizi açısından örnekleme frekansının 250 Hz-1 kHz arasında seçilmesi tavsiye edilir.

1.2.1. EEG Yöntemiyle Kaydedilen Dalga Çeşitleri

Frekansına göre EEG salınımları 5 kategoriye ayrılabilir:

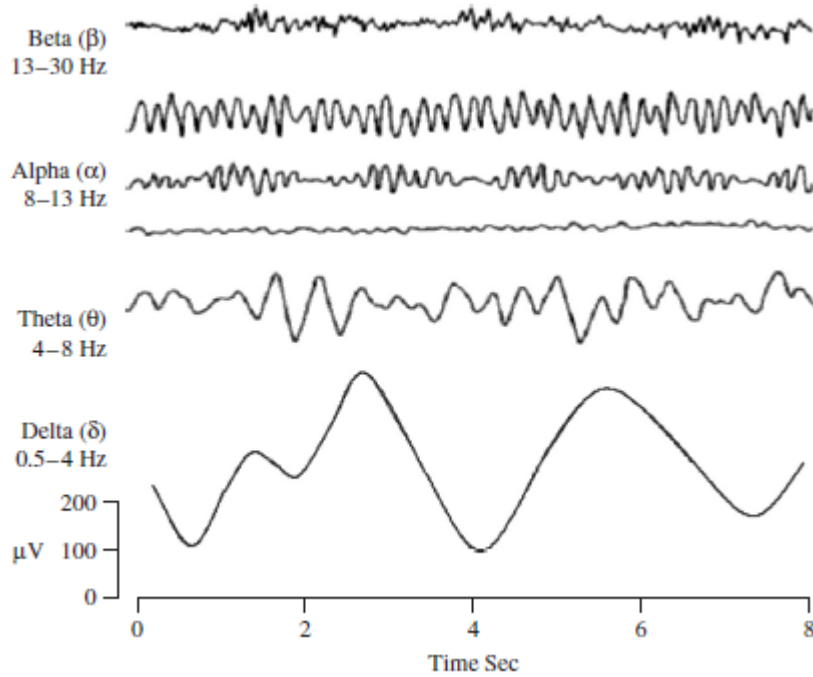
Delta Bandı: 0.5-4 Hz bandındaki dalgalar olarak tanımlanırlar. Bu ritimler, beynin derin bölgelerinde oluşan genliği yüksek dalgalarıdır. Derin uyku, uyanmaya başlama gibi düşük beyin aktiviteleriyle ilişkilidirler [13].

Teta Bandı: 4-8 Hz aralığındaki beyin dalgalarıdır. Çocuklarda sıklıkla görülen bu dalga çeşidi yaratıcı düşünce, stres ve derin meditasyonla ilişkilidir [14].

Alfa Bandı: 8-13 Hz aralığındaki ritimlerdir ve dalga biçimleri genel olarak sinüzoidaldir. Genellikle gözler kapalıyken ve rahatlama durumlarında görülür. Literatürde düşünsel veya fiziksel motor hareketiyle ilgili olarak gözlemlenen alfa dalgasına mu dalgası adı verilir [15].

Beta Bandı: 13-30 Hz aralığında bulunan ve genlikleri nadiren 30 μ V üzerine çıkan dalgalarıdır. Yoğun düşünme, konsantrasyon ve odaklanma durumunda ortaya çıkarlar [16].

Gama Bandı: 30 Hz'in üstü (genelde 45 Hz'e kadar) olarak tanımlanan gama dalgalarının nasıl oluştuğu tam olarak bilinmemektedir. Bu ritimlerin genlikleri çok düşüktür.

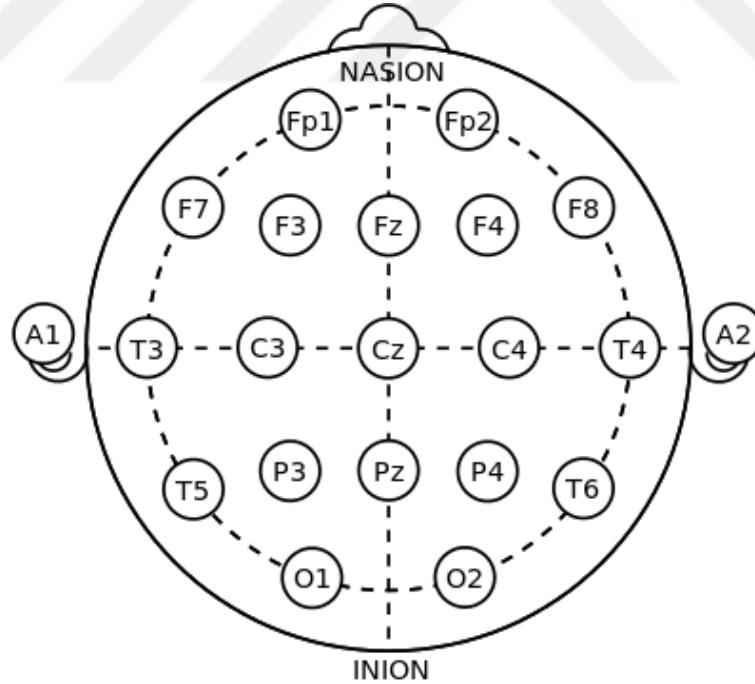


Şekil 1. EEG dalga türleri [17].

1.2.2. EEG Sinyallerinin Elde Edilmesi

EEG verileri elektrotların kafa derisi üzerine yerleştirilmesi suretiyle kaydedilir. Verilerin daha sağlıklı bir şekilde alınabilmesi için elektrotlar iletken jeller kullanılır. En yaygın olan AgCl' dir. Ölçüm alınmadan önce elektrotlarda aşınma olup olmadığı kontrol edilmelidir. Daha sonra elektrotlar yıkanıp, alkolle dezenfekte edilmelidir.

Elektrotlar kafa derisi üzerine belirli bir düzende yerleştirilir. Bunlardan birisi 10-20 düzenidir. Şekil 2'de bu düzen görülmektedir. Burada "Nasion" kafanın ön kısmını, "Inion" ise arka kısmını tanımlar. 10-20 sisteminde tek sayı ile numaralandırılmış elektrotlar beynin sol tarafını, çift sayı ile numaralandırılmış elektrotlar ise beynin sağ tarafını tanımlar. Sayıların önlerinde bulunan harfler ise beynin ilgili bölgelerinin isimlerinin kısaltılmış halidir (Fp = frontopolar; F = frontal; C = central; P = parietal; O = occipital; T = temporal). "Cz" ise beynin tam ortasından gelen sinyalleri tanımlamaktadır. EEG işaretlerinin genlik, faz ve frekansları sürekli değişmektedir. Bu nedenle anlamlı bir veri elde etmek için ölçümlerin uzun bir sürede yapılması gerekir.



Şekil 2. Elektrotların yerleşim düzeni (10-20 sistemi) [18].

1.2.3. Bozucu Etkiler

EEG verisinde bozucu etkiler meydana getiren dış gürültüler “artefakt” olarak adlandırılır. Aşağıda, veriyi kirli hale getiren bazı yapaylıklar listelenmiştir [19]:

1. Kas faaliyetleri: Kas faaliyetleri elektrik akımı ürettiğinden, deneyi yapan kişinin kas faaliyetleri mümkün olduğunda azaltılmalıdır. İnsan kalbinin sebep olduğu Elektrokardiyografi (EKG) gürültüsü de EEG kayıtlarına etki etmektedir.
2. Göz hareketleri: Göz retinasında bulunan çok sayıda nöron elektromanyetik alan üretmektedir. Dikey göz hareketleri EEG sinyallerinin sinüs fonksiyonu gibi görünmesine, yatay göz hareketleri ise kutu şeklinde görünmesine sebep olmaktadır. Göz kırpmak da EEG üzerinde bir tür yapaylığa sebep olmaktadır.
3. Elektrotların oynaması: Elektrotların oynaması veri kaydında gözle görülen yapaylıklar oluşturabilir. Bu sebeple ölçüm düzeneğinin bağlantılarına dikkat edilmelidir.
4. Hat gürültüsü: Şebeke frekansına (ABD için 60 Hz, Avrupa için 50 Hz) yakın spektral bileşenler hat gürültüsünden etkilenecektir.
5. Sallanma: Kafanın sallanması, vücuttaki suyun dağılımını etkilediğinden elektrik alanı da etkileyecektir.

1.2.4. EEG Yönteminin Avantajları

EEG analizinin kullanışlı olmasının sebeplerinden bazıları şu şekildedir:

1. EEG, yüksek zaman çözünürlüğüne sahiptir. Bu sayede EEG, bilişsel faaliyeti gerçekleştiği anda yakalayabilmektedir [20].
2. EEG, ucuz, hafif ve taşınabilir bir tekniktir.
3. EEG analizi ile beyin hangi bölgelerin aktif olduğu ve aktif olan bölgelerin birbirleriyle nasıl iletişime geçtiği açıklanabilmektedir.

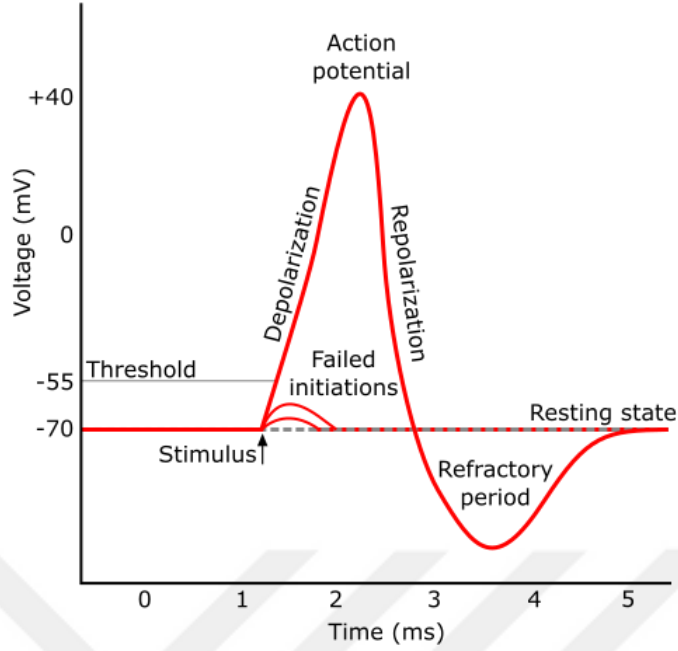
1.3. Elektromiyografi (EMG) Nedir?

Elektromiyografi (EMG), kaslara iletilen elektriksel sinyalin kas yüzeyinden elektrotlar veya içine batırılan iğneler yardımıyla ölçülmesi sonucu elde edilen işaretlerdir [21]. EMG sinyalinin kaydedilmesi sırasında, çevresel etkilerden veya ölçüm cihazından kaynaklı sebeplerden ötürü oluşan gürültüler EMG sinyalinin karakteristiğini etkiler [22]. EMG sinyallerinin EEG sinyallerine göre daha lineer bir karakteristiği mevcuttur. Sinyal genliği birkaç yüz mikro volttan birkaç mili volta kadar çıkabilmektedir [23].

1.3.1. Motor Üniteleri ve Aksiyon Potansiyeli

Sinir hücrelerinden gelen elektriksel uyarılar kaslarda kasılmaya sebep olduğu gibi, kaslardaki kasılmalarda bir tür elektrik alan oluşturur [24]. EMG, bu işaretlerin elektrotlar vasıtasıyla kaydedilmesi ile elde edilir. Kasların kasılması, motor nöronlar tarafından oluşturulan Motor Ünite Aksiyon Potansiyelleri (MÜAP) sayesinde gerçekleşir. Motor ünite, bir motor nöron, onun aksonları ve aksonların bağlandığı kas liflerinden oluşmaktadır. Motor nöronların içinde ve dışında bulunan Na^+ ve K^+ iyonlarının miktarı hücre içi ile dışı arasındaki potansiyel farkı belirler. Hiçbir uyarının bulunmadığı durumda bu potansiyel fark sabittir ve dinlenme potansiyeline (-70 mV) eşittir [25].

Hücre uyarıldığında, hücre zarı Na^+ iyonlarının hücre içine girmesini sağlar. Böylece hücre içi ile dışı arasındaki potansiyel fark artar. Eğer bu fark eşik değerine (-55 mV) ulaşmazsa uyarım bir süre sonra sönümlenir, aksi durumda iyon geçişleri daha da artmaya başlar. Bu durum “depolarizasyon” olarak adlandırılır [26]. Aksiyon potansiyeli belirli bir üst sınıra ulaştıktan sonra Na^+ geçişini sağlayan kanallar kapanmaya, K^+ geçişini sağlayan kanallar ise açılmaya başlar. Bu sayede hücre içi ve dışı arasındaki fark yeniden dinlenme potansiyeli seviyesine düşer. Bu olaya ise “repolarizasyon” denir. Repolarizasyondan sonra bu düşüş bir süre daha devam eder. Ancak bir müddet sonra hücre tekrar dinlenme seviyesine ulaşır [27]. Şekil 3’te bir motor nöronunun uyarılma sürecine ait aşamalar gösterilmiştir.



Şekil 3. Aksiyon potansiyeli [28].

Motor nöronu depolarizasyon ve repolarizasyon olayları sırasında geçen birkaç milisaniye süre boyunca hücre tekrar uyarılamaz [29]. Ayrıca bu süre zarfında aksiyon potansiyeli sıfır geçişleri yapmaktadır. Dolayısıyla aksiyon potansiyeli dönüşleri ile sıfır geçişleri arasında doğrusal bir ilişkiden söz edilebilir [30].

1.3.2. EMG Gürültü Kaynakları

EMG sinyalini etkileyen gürültü kaynaklarından birkaçı aşağıda belirtilmiştir:

Ortam gürültüsü: Yakında bulunan elektronik cihazlar, ışık kaynakları, güç kabloları çevresine elektromanyetik gürültüler yaymaktadır. Bu gürültünün genliği EMG sinyali genliğinin 1-3 katı kadar olabilir [31]. Şebeke gürültüsü de bu gruba dahil edilebilir.

Ölçüm cihazı: EMG kaydının yapıldığı ölçüm cihazı da elektronik bir cihaz olduğu için gürültü üretmektedir.

Hareket: EMG elektrotlarının yüzeyindeki oynamalar, ölçüm kablolarının hareket etmesi sinyal üzerinde gürültü oluşturur.

Doğal kararsızlık: EMG sinyalinin 0-20 Hz arasındaki bileşenleri motor nöron aktivitelerinden etkilendiği için kararsız yapıda olup, gürültü olarak değerlendirilmektedir [32].

1.4. Makine Öğrenmesi Nedir?

Makine öğrenme algoritmaları iki grup içinde incelenir. Bunlar sırasıyla:

1. Gözetimli Öğrenme (Supervised Learning)
2. Gözetimsiz Öğrenme (Unsupervised Learning)

Gözetimli öğrenme algoritmaları etiketlenmiş eğitim verilerini kullanır. Bir başka ifadeyle; eğitim verisinde hangi girdinin hangi çıktıyı verdiği bellidir. Algoritmanın görevi girdiler ile çıktılar (etiketler) arasındaki ilişkiyi bulmaktır. Böylelikle gelecekte eklenecek yeni verinin hangi çıktıyı vereceği tahmin edilebilir. Gözetimsiz öğrenme algoritmaları ise etiketlenmiş verileri kullanmaz. Bunun yerine eğitim verisinin özelliklerine bakarak veriyi gruplara (kümelere) ayırır ve her gruba bir etiket bilgisi ekler. Gelecekte eklenecek yeni veri bu gruplardan birine dahil edilir veya yeni bir grup oluşturmak için kullanılır.

Makine öğrenmesi geniş bir çalışma alanıdır ancak genellikle birkaç temel aşamadan oluşmaktadır. Bu aşamalar Şekil 4'te de gösterildiği üzere; ham verinin işlenmesi, özniteliklerin çıkarılması, sınıflandırılması, elde edilen model başarısının ölçülmesi ve model başarısının kaydedilmesidir. Günümüzde yaygın olarak kullanılan birçok sınıflandırma algoritması mevcuttur. Bunlardan bazıları, aynı zamanda bu tez çalışmasında da kullanılan, K-En Yakın Komşular, Karar Ağaçları, Yapay Sinir Ağları, Destek Vektör Makineleri ve Naif Bayes gibi gözetimli sınıflandırma algoritmalarıdır.



Şekil 4. Makine öğrenmesinde izlenecek adımlar

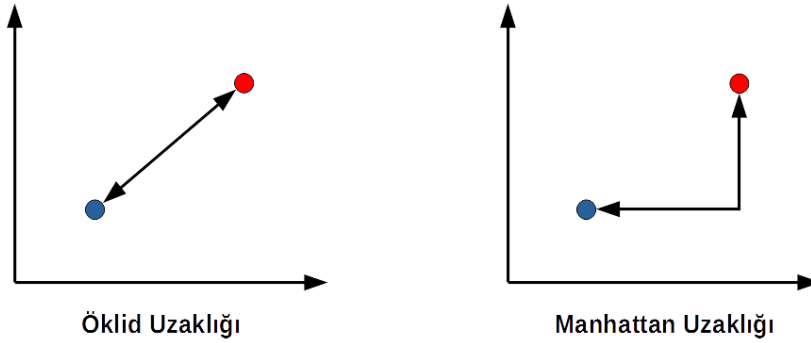
1.5. Sınıflandırma Algoritmaları

1.5.1. K-En Yakın Komşular Yöntemi

K-En Yakın Komşular (K-Nearest Neighbours veya kısaca KNN) algoritması en çok kullanılan gözetimli öğrenme algoritmalarından biridir. Test edilecek verinin sınıf tahmini için bu verinin diğer verilere olan uzaklıklarına bakılır. Test verisine en yakın mesafede k adet örnek tespit edilir ve k adet örnek içinde hangi sınıf çoğunlukta ise test örneği o sınıfa dahil edilir. KNN yöntemi için çeşitli uzaklık hesaplama yöntemleri kullanılabilir. Bunlardan en çok kullanılanları öklid ve manhattan uzaklıklarıdır. Varsayılan uzaklık bulma yöntemi öklid uzaklığıdır ancak çok sayıda veri ile çalışırken bilgisayarın daha hızlı sonuç vermesi için manhattan uzaklığı kullanılabilir. Verilen \mathbf{p} ve \mathbf{q} noktaları arasındaki öklid uzaklığı denklem 1’de, manhattan uzaklığı ise denklem 2’deki gibi hesaplanır. Ayrıca bu uzaklıkların geometrik gösterimi Şekil 5’te verilmiştir.

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

$$d(\mathbf{p}, \mathbf{q}) = d(\mathbf{q}, \mathbf{p}) = \sum_{i=1}^n |p_i - q_i| \quad (2)$$



Şekil 5. Uzaklık fonksiyonlarının geometrik ifadeleri

1.5.2. Destek Vektör Makineleri (DVM)

Destek Vektör Makineleri (DVM) yöntemi sınıflar arası marjin genişliğini en büyük yapan hiper düzlemi bulmaya çalışır. Bunun anlamı hiper düzlemin terimlerinin katsayılarını veren ağırlık vektörünün en küçük katsayılara sahip (normu en küçük) olması gerektiğidir. DVM yöntemi marjinin tanımlanabilmesi için her sınıfın sınır hatlarından geçen hiper düzlemlere ihtiyaç duyar. Sınır hatlarını oluşturan veri noktaları “destek vektörleri” olarak adlandırılır. DVM adını buradan almaktadır.

DVM çok boyutlu veriler üzerinde gayet iyi sonuçlar verebilir. İki sınıflı verileri doğrusal olarak ayırmak için geliştirilmiştir ancak çok sınıflı ve doğrusal olmayan veriler üzerinde de kullanılabilir. Bir hiper düzlemin genel denklemi aşağıdaki gibi tanımlanır:

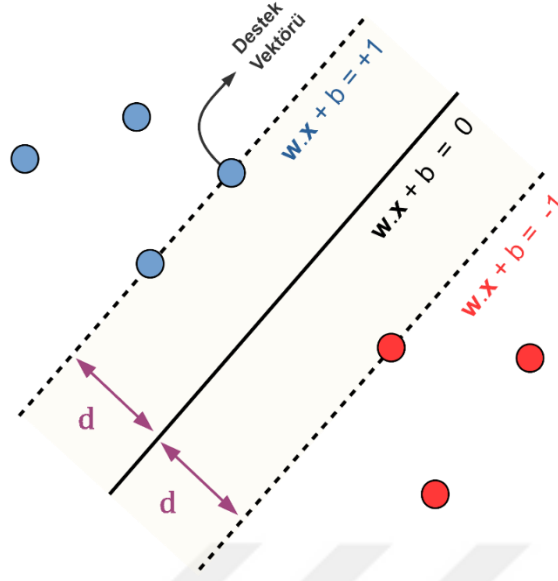
$$w \cdot x + b = 0 \quad (3)$$

1.5.2.1. Doğrusal DVM

Sınıflandırıcı hiper düzlemin, iki sınıfa da mümkün olduğunca uzak olması gerekmektedir. Bunun için marjinin maksimize edilmesi gerekmektedir. d mesafesi destek vektörlerinin hiper düzleme olan dik uzaklığına, ağırlık vektörünün normunun 1'e bölümüne, eşittir. Buna göre marjin Şekil 6'da görüldüğü üzere, aşağıdaki eşitlikteki gibi tanımlanabilir.

$$m = 2d \quad (4)$$

$$m = \frac{2}{\|w\|} \quad (5)$$



Şekil 6. Destek vektör makineleri yöntemi

Denklem 5'e göre marjin genişliğinin, ağırlık vektörünün normuyla ters orantılı olduğu görülmektedir. Buna göre, marjinin maksimize edilmesi için ağırlık vektörü normunun minimize edilmesi gereklidir. Destek vektör makineleri yönteminde sadece en küçük normlu ağırlık vektörünün bulunması yeterli gelmeyecektir. Aynı zamanda marjinin tanımı gereği sınırları içinde hiçbir veri girdisi bulunmamalıdır. Bu sebeple tüm veri girdilerin $(x_i, \forall i)$ aşağıdaki şartı sağlaması beklenir (Destek vektör makineleri yönteminde genellikle sınıflar +1 ve -1 olarak etiketlenirilir ($y_i = \{+1, -1\}$)).

$$\begin{cases} \mathbf{w} \cdot \mathbf{x}_i + b \geq 1, & y_i = 1 \\ \mathbf{w} \cdot \mathbf{x}_i + b \leq -1, & y_i = -1 \end{cases} \quad (6)$$

Yukarıdaki ifade daha basit haliyle şu şekilde yazılabilir:

$$y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 \quad (7)$$

Destek vektör makineleri denklem 7'deki koşul dahilinde çözülmesi gereken bir optimizasyon problemidir ve bu koşul "Lagrange Katsayıları" yöntemini DVM için kullanılabilir kılar.

Lagrange minimizasyonu için karesel (quadratic) yapıda bir hedef fonksiyonu $h(\mathbf{w}, b)$ tanımlayalım:

$$h(\mathbf{w}, b) = \min_{\mathbf{w}, b} \frac{\|\mathbf{w}\|^2}{2} \quad \text{ve} \quad y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 \quad \forall i \quad (8)$$

Buna göre, Lagrange denklemi ilkel (primal) formda aşağıdaki gibi yazılır:

$$\mathcal{L} = \frac{1}{2} \mathbf{w} \cdot \mathbf{w} + \sum_{i=1}^n \alpha_i (1 - y_i(\mathbf{w} \cdot \mathbf{x}_i + b)) \quad (9)$$

Bu üç bilinmeyenli denklemde α_i , Lagrange katsayısı olarak adlandırılır. Ayrıca bu denklemin \mathbf{w} ve b ye göre kısmi türevleri 0 (sıfır) olacaktır. Yani;

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^n \alpha_i y_i \mathbf{x}_i = 0 \quad (10)$$

$$\frac{\partial \mathcal{L}}{\partial b} = - \sum_{i=1}^n \alpha_i y_i = 0 \quad (11)$$

Buna göre aşağıdaki eşitlikler elde edilir:

$$\mathbf{w} = \sum_{i=1}^n \alpha_i y_i \mathbf{x}_i \quad (12)$$

$$\sum_{i=1}^n \alpha_i y_i = 0 \quad (13)$$

Denklem 12'deki eşitlik denklem 9'da kullanılırsa bilinmeyen sayısı bire düşürülebilir.

$$\mathcal{L} = -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j + \sum_{i=1}^n \alpha_i \quad (14)$$

Böylece Lagrange denkleminin dual formu elde edilir Dikkat edilirse bu fonksiyonunun sadece Lagrange katsayılarına bağlı olduğu görülmektedir. Tüm veri gidileri için Lagrange katsayıları bulunduğu anda, hiper düzlemin ağırlık vektörü elde edilebilir. Ayrıca Lagrange katsayılarının iki özelliğinden biri sıfıra eşit veya sıfırdan büyük olmasıdır. Yani,

$$\alpha_i \geq 0 \quad , \quad \forall i \quad (15)$$

İkinci bir özellik ise tüm katsayıların Karush-Kuhn-Tucker şartını sağlamasıdır. Buna göre her katsayının doğrusal koşulu ile çarpımı sıfır olmaktadır.

$$\alpha_i(1 - y_i(\mathbf{w} \cdot \mathbf{x}_i + b)) = 0 \quad , \quad \forall i \quad (16)$$

Dual form için optimizasyon problemi artık Lagrange katsayılarının maksimize edilmesidir. Ayrıca tüm katsayılar, sıfır ve sıfırdan büyük olmak koşulunu sağlamalıdır. Dual form için hedef fonksiyonunu tanımlayalım:

$$H(\boldsymbol{\alpha}) = \max_{\boldsymbol{\alpha}} \left(\sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j \right) \quad , \quad \alpha_i \geq 0 \quad \forall i \quad (17)$$

$$\sum_{i=1}^n \alpha_i y_i = 0 \quad (18)$$

Dual form için türetilen optimizasyon problemi bir “karesel programlama” problemidir ve global maksimumlar her zaman için bulunabilir.

$$W(\boldsymbol{\alpha}) = \max_{\boldsymbol{\alpha}} \left(\frac{1}{2} \mathbf{a}^T \mathbf{H} \mathbf{a} + \mathbf{c}^T \boldsymbol{\alpha} \right) \quad (19)$$

$$h_{ij} = -y_i y_j \mathbf{x}_i^T \mathbf{x}_j \quad (20)$$

$$\mathbf{c} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}_{N \times N} \quad , \quad \alpha_i \geq 0 \quad \forall i \quad (21)$$

Birçok α_i değeri sıfır olarak bulunur ve hiper düzlemin belirlenmesine herhangi bir katkıda bulunmazlar. Diğer katsayılar ise destek vektörlerine ait katsayılar olacaktır ve bu katsayılar yardımıyla ağırlık vektörü aşağıdaki eşitlikteki gibi elde edilir.

$$\mathbf{w} = \sum_{i, \alpha_i > 0} \alpha_i y_i \mathbf{x}_i \quad (22)$$

Ağırlık vektörü bulunduğuna göre bias (b) değeri de bulunabilir. Bias, Karush-Kuhn-Tucker şartı yardımıyla bulunan b_i sayılarının belli oranlarda toplamıyla elde edilir.

$$\alpha_i (1 - y_i (\mathbf{w}^T \mathbf{x}_i + b_i)) = 0, \quad \alpha_i > 0 \quad (23)$$

$$y_i (\mathbf{w}^T \mathbf{x}_i + b_i) = 1 \quad (24)$$

$$\begin{aligned} b_i &= \frac{1}{y_i} - \mathbf{w}^T \mathbf{x}_i \\ &= y_i - \mathbf{w}^T \mathbf{x}_i \end{aligned} \quad (25)$$

Bias (b) buluan b_i ve α_i sayılarının çarpımlarının toplamı ile α_i katsayılarının toplamına oranı şeklinde tanımlanabilir.

$$b = \frac{\sum \alpha_i b_i}{\sum \alpha_i} = \frac{\sum \alpha_i (y_i - \mathbf{w}^T \mathbf{x}_i)}{\sum \alpha_i} \quad (26)$$

Modelin ağırlık vektörü ile bias değeri bulunduğuna göre test aşamasına geçilebilir. Bir \mathbf{z} test verisi $f(\mathbf{z})$ fonksiyonuna girdi olarak verilir. Eğer $f(\mathbf{z}) > 0$ ise, test verisi +1 olarak, $f(\mathbf{z}) < 0$ ise -1 olarak sınıflandırılır. $f(\mathbf{z})$ fonksiyonu aşağıda verilmiştir.

$$\begin{aligned} f(\mathbf{z}) &= \mathbf{w}^T \mathbf{z} + b \\ &= \sum_{i, \alpha_i > 0} \alpha_i y_i \mathbf{x}_i^T \mathbf{z} + b \end{aligned} \quad (27)$$

Karesel programlama metotları kullanılırken Lagrange katsayılarının önceden belirlenmiş bir C değerine göre üst limiti belirlenebilir. C limiti azaldıkça hiper düzlemin belirlediği sınırların etkisi azalır ve daha yumuşak geçişlere sahip bir hiper düzlem elde edilir, C arttıkça sınırların keskinliği artar.

$$0 \leq \alpha_i \leq C \quad \forall i \quad (28)$$

1.5.2.2. Çekirdek Fonksiyonları ile Sınıflandırma

Doğrusal olarak sınıflandırılmayan veri kümeleri daha yüksek boyutlu özellik uzaylarında yeniden tanımlanarak, doğrusal olarak sınıflandırılabilir hale getirilebilir. Bir veri kümesini daha yüksek boyutlu veri kümelerine eşlemek için kullanılan fonksiyonlara “haritalama fonksiyonları” adı verilir ve genel olarak $\phi(\mathbf{x})$ şeklinde gösterilir.

$$\phi(\mathbf{x}): \mathbb{R}^d \rightarrow \mathbb{R}^k \quad (29)$$

Doğrusal olmayan veri kümesi uygun bir haritalama fonksiyonu yardımıyla yüksek boyutlu bir özellik uzayına eşlendikten sonra DVM yöntemi uygulanır. Ancak kullanılan haritalama fonksiyonunun sadece o veri boyutuna uygun olarak hazırlanmış olması ve farklı boyuttaki girdiler için farklı haritalama fonksiyonlarının tanımlanması gerekliliği bu yöntemin etkinliğini sınırlandırmaktadır. Bu sebeple dönüştürülmüş verileri hesaplamaya gerek kalmadan, iç çarpımlarından elde edilen sonucu doğrudan kullanabileceğimiz özel fonksiyonlara ihtiyaç duyulmaktadır. Mercer Teoremi’ne göre böyle bir fonksiyon tanımlamak mümkündür. Bu yöntem “çekirdek hilesi (kernel trick)” olarak bilinir. Bir çekirdek fonksiyonu $K(\mathbf{x}_i, \mathbf{x}_j)$ şeklinde gösterilir ve aşağıdaki gibi tanımlanır:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j) \quad (30)$$

Buna göre doğrusal olmayan DVM için hedef fonksiyonu şu şekilde yazılır:

$$H(\boldsymbol{\alpha}) = \max_{\boldsymbol{\alpha}} \left(\sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \right) \quad (31)$$

En çok kullanılan çekirdek fonksiyonları şunlardır:

- Doğrusal Çekirdek Fonksiyonu

$$K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i \cdot \mathbf{x}_j \quad (32)$$

- Polinom Çekirdek Fonksiyonu

$$K(\mathbf{x}_i, \mathbf{x}_j) = (1 + \mathbf{x}_i \cdot \mathbf{x}_j)^d \quad (33)$$

- Radyal Tabanlı Çekirdek Fonksiyonu

$$K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2} \quad (34)$$

- Sigmoid Çekirdek Fonksiyonu

$$K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(-\kappa \mathbf{x}_i \cdot \mathbf{x}_j + \theta) \quad (35)$$

1.5.2.3. Çok Sınıflı Verilerde Sınıflandırma

Destek vektör makineleri yöntemi iki sınıflı verilerin sınıflandırılmasında kullanılmak için geliştirilmiştir. Ancak bazı yaklaşımlar sayesinde çok sınıflı (ikiden fazla sınıflı) verileri sınıflandırmak mümkündür. Uygulanabilecek yaklaşımlardan ilki “Bire Bir Yaklaşımı (One to One)” dir. Bu yaklaşımda N tane sınıfın mümkün olan tüm ikili sınıflandırmaları yapılır. Buna göre $\frac{N(N-1)}{2}$ adet DVM eğitilmelidir. Sınıflandırılacak veri eğitilen DVM sınıflandırıcılarına girdi olarak verilir ve etiketler belirlenir. Test girdisi en çok hangi sınıfa dahil edilmişse o sınıfa ait olarak kabul edilir.

Diğer bir yaklaşım ise “Bire Çok Yaklaşımı (One to Many)” dir. Bu yöntemde bir sınıfın verilerinin, diğer verilere göre sınıflandırması yapılır. Bire bir yaklaşıma göre hesaplama maliyeti daha az olan bu yaklaşımda N adet DVM eğitilir. Test verisi, sınıflandırıcı fonksiyonlara girdi olarak verilir ve en yüksek sonucun elde edildiği sınıfa dahil edilir.

1.5.3. Karar Ağaçları

Karar Ağaçları bir gözetimli öğrenme algoritmasıdır. Bu yöntemde temel amaç verilerin en az işlem adımıyla, hiyerarşik bir düzen içinde alt kümelere ayırmaktır. Bunun için her özelliğin (özniteliğin) sınıflandırmaya olan etkisi ölçülür ve sıralanır. En üst sıradaki özelliğin sınıflandırmadaki rolünün en fazla olduğu ve böylece sınıflandırmaya bu özellikten başlandığı takdirde, sonuca daha az işlem adımıyla ulaşılabileceği varsayılır. Karar ağaçları adı altında çeşitli algoritmalar bulunmaktadır. Bunlardan biri olan J48 en çok bilinen karar ağacı algoritmasıdır. J48 algoritmasında bir özniteliğin bilgi miktarını belirlemek için; bilgi kazancı, bilgi kazancı oranı, veya gini indeksi yöntemleri kullanılabilir.

1.5.3.1. Bilgi Kazancı

Bir veri setinde kullanılan her bir öznitelik veya özellik F olarak tanımlansın. Buna göre ilgili özelliğin bilgi kazancı aşağıdaki gibi hesaplanır:

$$IG_F = Info_F - Expected Info_F \quad (36)$$

Denklem 36'da geçen $Info_F$ şu şekilde bulunur:

$$Info_F = - \sum_i \frac{N_i}{N} \log_m \frac{N_i}{N} \quad (37)$$

m : Toplam sınıf sayısı

N_i : i sınıfındaki örnek sayısı

N : Toplam örnek sayısı

Bir F özelliğinin her bir alt eleman k olarak tanımlansın ($k \in F$). Buna göre F özelliğine ait $Expected Info_F$ şu şekilde bulunur:

$$Expected Info_F = \sum_{k \in F} \frac{N_k}{N} Info_k \quad (38)$$

N_k : k elemanının örnek sayısı

1.5.3.2. Bilgi Kazancı Oranı

Bir veri setinde kullanılan her bir öznelik veya özellik F olarak tanımlansın. Buna göre ilgili özelliğin bilgi kazancı oranı aşağıdaki gibi hesaplanır:

$$IG\ Ratio_F = \frac{IG_F}{Split\ Info_F} \quad (39)$$

Denklem 39'da geçen IG_F ilgili özelliğin bilgi kazancını temsil etmektedir. Bilgi kazancının nasıl hesaplandığı bir önceki başlıkta ifade edilmiştir. Denklemde yer alan $Split\ Info_F$ ifadesi ise şu şekilde bulunmaktadır:

$$Split\ Info_F = - \sum_{k \in F} \frac{N_k}{N} \log_2 \left(\frac{N_k}{N} \right) \quad (40)$$

N_k : F özelliğine ait k elemanın örnek sayısı

1.5.3.3. Gini İndeksi

Bir veri setinde kullanılan her bir öznelik veya özellik F olarak tanımlansın. Buna göre ilgili özelliğin gini indeksi aşağıdaki gibi hesaplanır:

$$Expected\ Gini_F = \sum_{k \in F} \frac{N_k}{N} Gini_k \quad (41)$$

Bir F özelliğini oluşturan her bir eleman k olarak tanımlansın ($k \in F$). Buna göre her k elemanına ait $Gini_k$ şu şekilde bulunur:

$$Gini_k = 1 - \sum_i \left(\frac{N_i}{N_k} \right)^2 \quad (42)$$

N_i : k alt kümesinde i sınıfında olan örnek sayısı

N_k : F özelliğine ait k elemanın örnek sayısı

1.5.3.4. Karar Ağacının Oluşturulması

Karar ağacı oluşturulurken aşağıdaki adımlar izlenir:

1. Adım: Seçilen bilgi ölçme yöntemine göre, tüm özelliklerinin bilgi kazançları (*IG*), bilgi kazancı oranı (*IG Ratio*) veya beklenen gini değeri (*Expected Gini*) bulunur.
2. Adım: Seçilen bilgi ölçme yöntemine göre, en büyük bilgi kazancına, en büyük bilgi kazancı oranına veya en küçük beklenen gini değerine sahip özellik bulunur.
3. Adım: Bu özellik karar ağacının düğümü olarak seçilir. Bu düğümden her k elemanı için alt düğümler oluşturulur. Her alt düğüm F özelliğinde ve sadece k elemanını içerecek şekilde belirlenmiş (ancak F özelliği çıkarılmış) alt veri gruplarıdır.
4. Adım: Her alt düğüm için örneklerin sınıfları aynı ise, kendisinden sonra başka bir özellik kalmamış ise veya kalan özellikler aynı elemandan oluşuyorsa; bu veri grubu için başka bir işlem yapılmaz.
5. Adım: Tüm düğümler bu şartlardan en az birini sağlıyorsa algoritma sonlandırılır aksi halde geri kalan düğümler için yukarıdaki işlemler tekrarlanır.

1.5.4. Naif Bayes

Bayes yöntemi, verilerin içeriğinden çok dağılım özelliklerinden yararlanan gözetimli bir sınıflandırma yöntemidir. Dolayısıyla sayısal veya metin içerikli veriler üzerinde sınıflandırma yapılabilir. Bayes yöntemi veri sayısının az olması durumunda bile başarılı tahminler yapabilmektedir. Çünkü az sayıda veriden bile sınıfların olası dağılımları hesaplanabilmektedir. Bu yöntemde sınıfların gauss dağılımına sahip olduğu varsayılır. Bu sebeple sınıfların dağılımlarının gauss dağılımına benzerliği ölçüsünde sınıflandırma performansı söz konusudur. Bayes sınıflandırıcı herhangi bir girdiyi etiketlendirirken veri setindeki her bir sınıfın olasılık yüzdesini hesaplar ve en büyük olasılığı veren sınıfa bu girdiyi atar. Bunu yaparken bayes kuralından yararlanır. d Boyutlu eğitim verisinde bir x noktasının ω_i sınıfına ait olma olasılığı şu şekilde bulunur ($i = 1,2,3, \dots, n$):

$$P(x|\omega_i) = \frac{1}{(2\pi)^{\frac{d}{2}} \sqrt{|\det \Sigma_i|}} e^{-\frac{1}{2}(x-\mu_i) \cdot \Sigma_i^{-1} \cdot (x-\mu_i)^T} \quad (43)$$

μ_i : ω_i sınıfına ait ortalama vektörü

$\Sigma_i^{-1}, |\Sigma_i|$: ω_i sınıfına ait kovaryans matrisinin tersi ve determinanı

Aşağıda verilen eşitlik “Bayes Teoremi” olarak bilinir.

$$P(A|B) = \frac{P(B|A).P(A)}{P(B)} \quad (44)$$

$P(A|B)$: B koşulu sağlandığı takdirde A'nın gerçekleşme olasılığı

$P(A)$: A'nın gerçekleşme olasılığı

$P(B|A)$: A koşulu sağlandığı takdirde B'nin gerçekleşme olasılığı

$P(B)$: B'nin gerçekleşme olasılığı

Bayes teoremini S adında sınırlı bir örnek uzayı için tanımlayalım ve bu uzay A_i ($i = 1, 2, 3, \dots, n$) sonlu olaylarının toplamından oluşsun. Bu durumda Bayes teoremini her bir A_i olayı için yazalım:

$$P(A_i|B) = \frac{P(B|A_i).P(A_i)}{P(B)} \quad (45)$$

Yukarıdaki ifadede $P(A_i)$ olasılığı, A_i olay kümesinin eleman sayısının tüm örnek uzayın eleman sayısına oranıdır. Ayrıca B olayı da S örnek uzayının bir alt kümesi olsun. Bu durumda B olayı aynı zamanda kendisinin tüm A_i parçalarıyla olan kesişimlerinden oluşmaktadır. Bu bakış açısı “Toplam olasılık prensibi” olarak bilinir.

$$P(B) = \sum_{i=1}^n P(B \cap A_i) \quad (46)$$

Bu ifadeyi Bayes teoremine göre yeniden yazarsak aşağıdaki eşitliği elde ederiz:

$$P(B \cap A_i) = P(B|A_i).P(A_i) \quad (47)$$

$$P(B) = \sum_{i=1}^n P(B|A_i).P(A_i) \quad (48)$$

$P(B)$ olasılığı da bilindiğine göre Bayes teoremini şu şekilde güncelleyebiliriz:

$$P(A_i|B) = \frac{P(B|A_i) \cdot P(A_i)}{\sum_{i=1}^n P(B|A_i) \cdot P(A_i)} \quad (49)$$

Eğitim verisini sonlu sayıda (ω_i , $i = 1, 2, \dots, n$) sınıflarından oluşmuş bir örnek uzay olarak tanımlayalım. Bu durumda örnek uzayda bilinen bir noktanın (\mathbf{x}) ω_i sınıfına ait olma olasılığı $P(\omega_i|\mathbf{x})$ şeklinde ifade edilebilir. Bayes teoremine göre bu olasılık şu şekilde hesaplanabilir:

$$P(\omega_i|\mathbf{x}) = \frac{P(\mathbf{x}|\omega_i) \cdot P(\omega_i)}{\sum_{i=1}^n P(\mathbf{x}|\omega_i) \cdot P(\omega_i)} \quad (50)$$

Burada $P(\mathbf{x}|\omega_i)$ olasılığı önceki kısımlarda anlatıldığı gibi ω_i sınıfının olasılık yoğunluk fonksiyonunu tanımlamaktadır ve aşağıdaki gibidir:

$$P(\mathbf{x}|\omega_i) = \frac{1}{(2\pi)^{\frac{d}{2}} \sqrt{|\det \Sigma_i|}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i) \cdot \Sigma_i^{-1} \cdot (\mathbf{x}-\boldsymbol{\mu}_i)^T} \quad (51)$$

$P(\omega_i)$ olasılığı ise basitçe ω_i sınıfının veri sayısının, tüm uzayın veri sayısına oranıdır.

Diskriminant fonksiyonları test girdisinin hangi sınıfa ait olduğunu bulmak için kullanılan fonksiyonlardır. Bu fonksiyonlar her bir sınıf için tanımlanır daha sonra test girdisinin hangi fonksiyonda en büyük değeri aldığı belirlenir son olarak test girdisi o fonksiyonunun sınıfına dahil edilir. Diskriminant fonksiyonları $g_i(\mathbf{x})$ şeklinde gösterilir ($i = 1, 2, 3, \dots, n$). Diskriminant fonksiyonları Bayes sınıflandırma yönteminde birçok şekilde tanımlanabilir. Örneğin $P(\omega_i|\mathbf{x})$ olasılığı sınıflandırma kriteri olarak kullanılabilir.

$$g_i(\mathbf{x}) = P(\omega_i|\mathbf{x}) = \frac{P(\mathbf{x}|\omega_i) \cdot P(\omega_i)}{\sum_{i=1}^n P(\mathbf{x}|\omega_i) \cdot P(\omega_i)} \quad (52)$$

Ancak sınıflandırmada paydadaki kısım tüm diskriminant fonksiyonlarında aynı olduğundan; paydanın sınıflandırmaya katkısı yoktur.

Paydadaki ifade sadece sonuçların normalize edilmesini sağlamaktadır. Bu sebeple sınıflandırma kriteri olarak sadece pay da kullanılabilir:

$$g_i(\mathbf{x}) = P(\mathbf{x}|\omega_i).P(\omega_i) \quad (53)$$

Diskriminant fonksiyonu üstel ifadeler içerdiğinden işlem kolaylığı için logaritması alınarak aşağıdaki şekilde de tanımlanabilir. Bu dönüşüm sonucu değiştirmeyecektir.

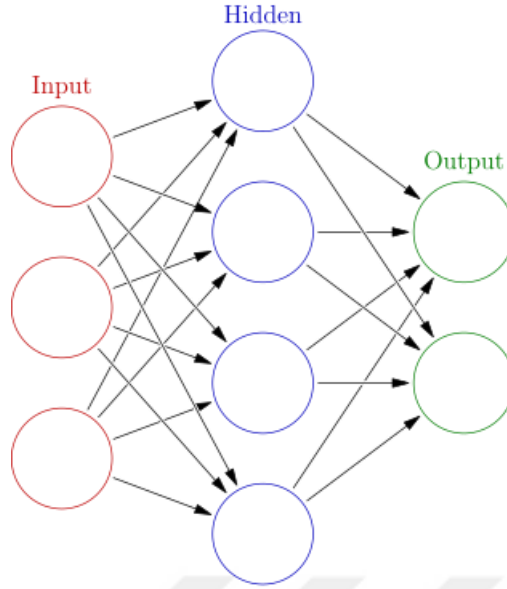
$$g_i(\mathbf{x}) = \ln(P(\mathbf{x}|\omega_i).P(\omega_i)) \quad (54)$$

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)\boldsymbol{\Sigma}_i^{-1}(\mathbf{x} - \boldsymbol{\mu}_i)^T - \frac{d}{2}\ln(2\pi) - \frac{1}{2}\ln|\boldsymbol{\Sigma}_i| + \ln P(\omega_i) \quad (55)$$

1.5.5. Yapay Sinir Ağları

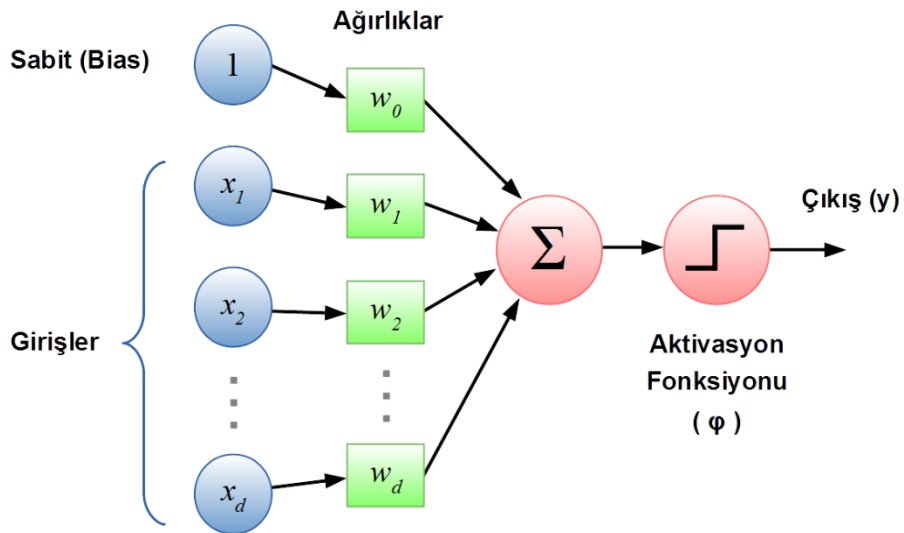
Yapay Sinir Ağları (YSA), sınıflandırmada sıkça kullanılan gözetimli bir sınıflandırma algoritmasıdır. Yapay sinir ağları, insan beyninin sinir ağı yapısından esinlenerek tasarlanmıştır. Biyolojik sinir ağlarının, nöronlardan oluşmasına benzer şekilde, YSA da perceptron adı verilen küçük birimlerden oluşmuştur. Bu birimler üç katman halinde sıralanmışlardır. Bu katmanlar sırasıyla giriş katmanı, gizli katmanlar ve çıkış katmanıdır. Bu katmanların bir temsili Şekil 7'de gösterilmiştir.

Katmanlar arası her bir perceptron birbirine tıpkı sinir hücreleri gibi bağlıdır ve her bir bağlantı veya her bir giriş bir ağırlık katsayısı (\mathbf{w}) ile çarpılarak bir sonraki perceptrona bilgi iletir. Bir perceptron kendisine gönderilen bilgileri toplar ve bir aktivasyon fonksiyonu aracılığıyla çıkış üretir. Eğer elde edilen çıkış istenilenden farklıysa bu hata olarak adlandırılır. Bu sistem Şekil 8'de gösterilmiştir. YSA, hatayı küçültmek için türev tabanlı yöntemler yardımıyla ağırlık katsayılarını günceller. Güncelleme işleminde ağırlıkların ne oranda değişeceği öğrenme katsayısı adı verilen bir parametre ile belirlenir. Her güncelleme sonrası çıkışlar yeniden hesaplanır. Bu döngü belli bir deneme sayısı boyunca devam eder.



Şekil 7. YSA katmanları [33].

Bir YSA modeli eğitilmeden önce belirlenmesi gereken birtakım parametreler bulunmaktadır. Bu parametreler ağı kaç girişli ve kaç çıkışlı olacağı, gizli katmanların sayısı ve bu katmanlardaki hücre sayısı, öğrenme oranı ve deneme sayısıdır [34].



Şekil 8. Perceptronun yapısı

1.6. Sınıflandırmada Doğrulama Metodları

1.6.1. Hold-Out Metodu

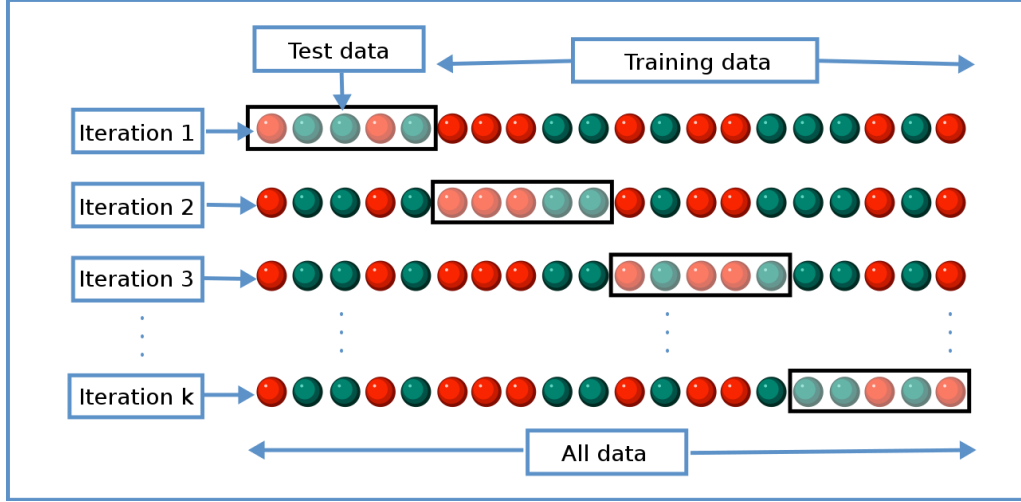
Hold-Out metodunda modelin başarısını ölçmek için, eğitim setinin bir kısmı (genellikle %20 si) doğrulama işlemi için ayrılır. Eğitim verisinin geri kalan kısmı ile öğrenme sağlandıktan sonra doğrulama verisi ile model test edilir. Model başarısı, sınıflandırıcının doğru tahmin ettiği örnek sayısının, tüm örneklerin sayısına oranı şeklinde tanımlanır. Bu durumda tüm örnekler doğru tahmin edildiğinde; model başarısı 1, yanlış tahmin edildiğinde ise 0 olmaktadır.

$$0 \leq \text{Model Başarısı} = \frac{\text{Doğru Tahminler}}{\text{Tüm Tahminler}} \leq 1 \quad (56)$$

Bu yöntemde eğitim verisi bölündüğü için öğrenme başarısı, eğitim verisinin bölünmediği duruma göre daha düşük çıkabilir. Çünkü öğrenmeye katkısı daha fazla olan veriler eğitim verisinden çıkarılabilmektedir. Bu sorun öğrenmenin eksik olduğu anlamına gelir. Eksik öğrenmede model başarısı (eğitim başarısı) ile test başarısı düşüktür. Bu yüzden hold-out yöntemi pek kullanılmamaktadır.

1.6.2. K Katlamalı Çapraz Doğrulama Metodu

Çapraz doğrulama metodunda eğitim verisi rastgele bir şekilde karıştırılarak eş k parçaya bölünür. Genellikle k sayısı 5 veya 10 olarak belirlenir. Bu yöntemde modelin k defa eğitimi söz konusudur. Şekil 9'da gösterildiği üzere ilk iterasyonda parçalardan biri doğrulama verisi olarak seçilirken, diğer parçalar eğitim için kullanılır. Eğitim sonunda doğrulama verisi ile model başarısı hesaplanır. İkinci iterasyonda daha önce seçilmeyen veri grubu doğrulama için kullanılırken, geri kalanlar eğitim için kullanılır. Böylece k adet iterasyon sonunda elde edilen model başarılarının ortalaması alınarak çapraz doğrulama başarısı hesaplanır. Bu yöntemde tüm eğitim seti öğrenmede kullanıldığından hold-out yöntemine göre daha kararlıdır.



Şekil 9. Çapraz doğrulama metodu

1.1. Temel Bileşenler Analizi

Boyut azaltma için kullanılan yaygın algoritmalarından birisi “Temel Bileşen Analizi (TBA)” veya “Principal Component Analysis (PCA)” dır. TBA yönteminin amacı verinin en küçük varyanslı özneliği bulmak ve veri setinden çıkartmaktır. Bu sayede iki boyutlu veri tek boyuta, üç boyutlu bir veri iki boyuta daha genel bir ifadeyle d boyutlu bir veri d-1 boyuta indirgenebilir.

1.1.1. Ortalama Vektörü

Bir verinin ortalama vektörü; her özneliğin ortalama değerleriyle oluşturulmuş bir vektör olup 1xd uzunluğundadır. X verisine ait ortalama vektör \bar{X} ile gösterilir.

$$\bar{X}_i = \frac{\sum_{k=1}^n X_{ki}}{n} \quad (i = 1, 2, \dots, d) \quad (57)$$

Ortalama vektörü aşağıdaki gibi de tanımlanabilir.

$$\bar{X} = \frac{J_{1,n} \cdot X}{n} \quad (58)$$

$J_{1,n}$: 1xn uzunluğunda tüm elemanları 1 olan vektör

1.1.2. Standart Sapma ve Varyans

Standart sapma (s), verinin ortalama vektörü etrafındaki dağılımının bir ölçüsüdür. Verinin elemanları ortalama vektörüne yaklaştıkça verinin standart sapması azalır, uzaklaştıkça artar. Varyans ise standart sapmanın karesidir. Denklem 59'da \mathbf{X} verisine ait her özneliğin standart sapmalarının hesaplanması yer almaktadır.

$$s_i = \sqrt{\frac{\sum_{k=1}^n (X_{ki} - \bar{X}_i)^2}{n-1}} \quad (i = 1, 2, \dots, d) \quad (59)$$

1.1.3. Kovaryans Matrisi

Kovaryans matrisi, veri setindeki herhangi bir özneliğin diğer özneliklere göre nasıl değiştiğini (veriler birlikte artabilir, azalabilir veya bağımsız olabilirler) görmemizi sağlayan matristir. d boyutlu bir veri setinin kovaryans matrisi $d \times d$ boyutlarında simetrik bir matristir. Kovaryans matrisi Sigma (Σ) ile gösterilir.

Kovaryans matrisinin i . satır ve i . sütundaki değeri (esas köşegen elemanları) veri setinin i . boyutuna ait varyansı verir ($i=1,2,\dots,d$). Hangi boyutun varyansı daha büyükse verinin o boyutta daha geniş yer kapladığı anlaşılır. Eğer veri setindeki i . boyuta ait veriler ile (veya i . özneliğe) j . boyuttaki veriler arasındaki ilişkiye bakacak isek, kovaryans matrisinin i . satır – j . sütuna (veya j . satır – i . sütuna) ait değere bakılır. Eğer sayı pozitif ise her iki değişken birlikte artmakta veya azalmaktadır, sayı negatif ise bir değişken artarken diğeri azalmaktadır, sıfır ise bu iki değişken arasında artış-azalış yönünden bir ilişki yoktur. Bir d boyutlu n elemanlı \mathbf{X} verisine ait kovaryans matrisi şu şekilde tanımlanır:

$$\Sigma = \frac{\mathbf{Z}^T \cdot \mathbf{Z}}{n} \quad (60)$$

Burada \mathbf{Z} matrisinin matematiksel ifadesi denklem 61'de verilmiştir.

$$\mathbf{Z} = \mathbf{X} - \frac{\mathbf{J}_n \cdot \mathbf{X}}{n} \quad (61)$$

\mathbf{J}_n : $n \times n$ boyutlarında tüm elemanları 1 olan matris

1.1.4. Özdeğerler ve Özvektörler

Kovaryans matrisinin özdeğeri Lamda (λ) olarak tanımlanır ve aşağıdaki eşitlik yardımıyla bulunur.

$$\det(\Sigma - \lambda.I) = 0 \quad (62)$$

Denklem 62'nin kökleri ilgili özdeğerleri verir. En büyük özdeğere ait özvektör verideki en büyük varyansı verecek şekilde bir doğrultu belirlememizi sağlar. Bu sayede veriye ait noktaların bu doğrultu üzerine izdüşümü (projeksiyonu) yapılabilir. Dolayısıyla bu doğrultuya dik yöndeki eksen yok edilmiş olur. Lamda (λ) özdeğerine ait özvektör \mathbf{v} olarak tanımlansın. Buna göre, \mathbf{v} vektörü aşağıdaki denklem sisteminin çözümüyle elde edilir:

$$\Sigma.\mathbf{v} = \lambda.\mathbf{v} \quad (63)$$

1.1.5. Temel Bileşenler Analizi Uygulaması

Verilen veri setinde (\mathbf{X}) boyut azaltması yapılırken veri setinin etiket bilgisine bakılmaksızın aşağıdaki adımlar uygulanır:

1. Adım: Veri setinin ortalama vektörü ($\bar{\mathbf{X}}$) bulunur:

$$\bar{\mathbf{X}} = \frac{J_{1 \times N} \cdot \mathbf{X}}{N} \quad (64)$$

N : Veri setinin eleman sayısı

$J_{1 \times N}$: $1 \times N$ boyutlarında tüm elemanları 1 olan vektör

2. Adım: Kovaryans matrisi (Σ) hesaplanır. Bunun için öncelikle \mathbf{Z} matrisi bulunur.

$$\mathbf{Z} = \mathbf{X} - \frac{J_N \cdot \mathbf{X}}{N} \quad (65)$$

J_N : $N \times N$ boyutlarında tüm elemanları 1 olan matris

$$\Sigma = \frac{\mathbf{Z}^T \cdot \mathbf{Z}}{N} \quad (66)$$

3. *Adım:* Kovaryans matrisinin özdeğerleri ve bu özdeğerlerin özvektörleri bulunur.
4. *Adım:* Veriyi c boyuta indirgemek için öz değerler büyük olandan küçük olana doğru sıralanır ve ilk c tanesinin öz vektörleri ile W dönüşüm matrisi oluşturulur (c , verinin boyut sayısından küçük bir sayıdır).
5. *Adım:* Veriye aşağıda tanımlanan dönüşüm uygulanır.

$$PC = Z.W \quad (67)$$



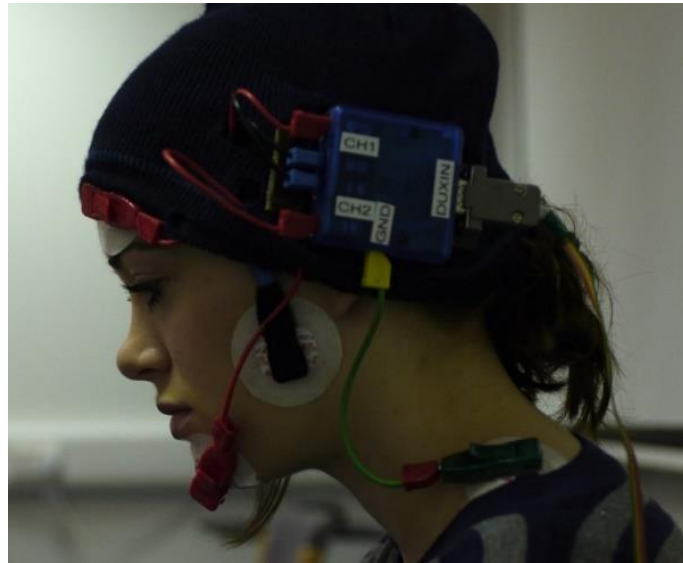
2. YAPILAN ÇALIŞMALAR

2.1. Kullanılan Veri Seti

Tez çalışmasında, akran kız ve erkek 20 öğrenciye ait eşzamanlı kaydedilmiş EEG ve EMG ölçümlerini içeren bir veri seti [35] kullanılmıştır. Katılımcıların Tablo 1’de verilen 11 gündelik aktiviteyi yaparken kaydedilmiş bu veri seti 1 kHz örnekleme frekanslı bir kanal EEG, 1 kanal EMG verilerinden oluşmaktadır. EEG kaydı Fp1 elektrotu üzerinden alınırken, EMG kaydı çene üzerinden alınmıştır. Deneye düzeneğine ait bir görüntü Şekil 10’da verilmiştir.

Tablo 1. Deney aktiviteleri ve sınıf numaraları

No	Aksiyonlar	No	Aksiyonlar
1	Göz kırpmak	7	Yatarken okuma
2	Göz yummak	8	Otururken okuma
3	Flow mobil oyunu oynama	9	Sudoku çözme
4	Çene sıkamak	10	Temple Run mobil oyunu oynama
5	Dinlenme	11	Kelime araştırması yapmak
6	Kaşları kaldırmak		



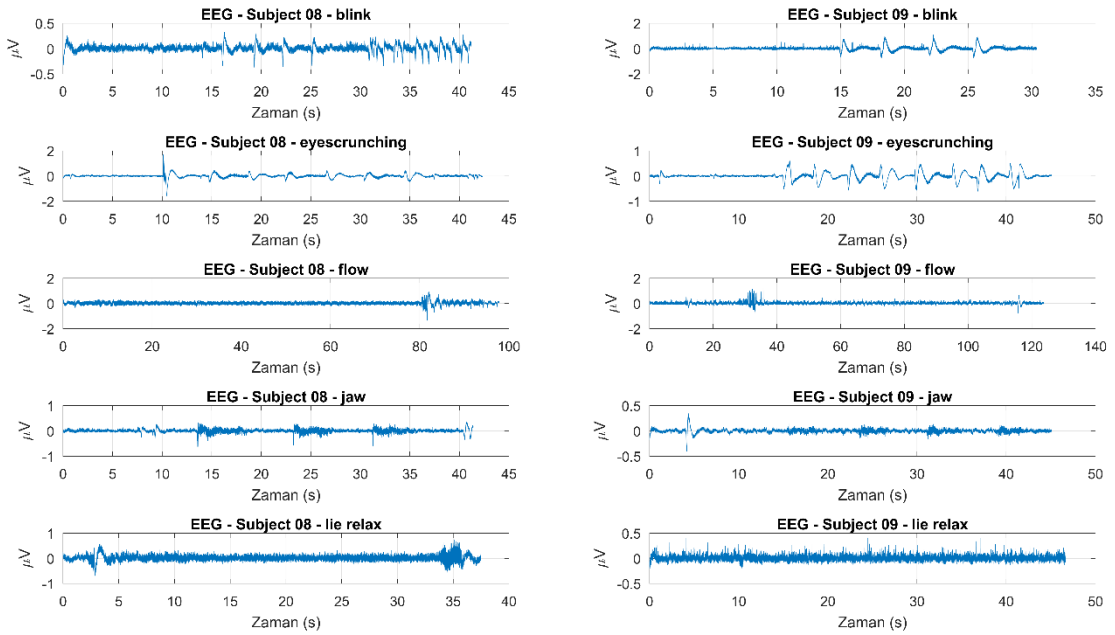
Şekil 10. Deney düzeneği

EEG ve EMG ölçümleri 500 kat yükseltılarak kaydedilmiştir. Tablo 2’de yukarıda adı geçen aktivitelerin EEG kayıtlarına ait istatistiksel özellikler ve toplam örnek süreleri verilmiştir.

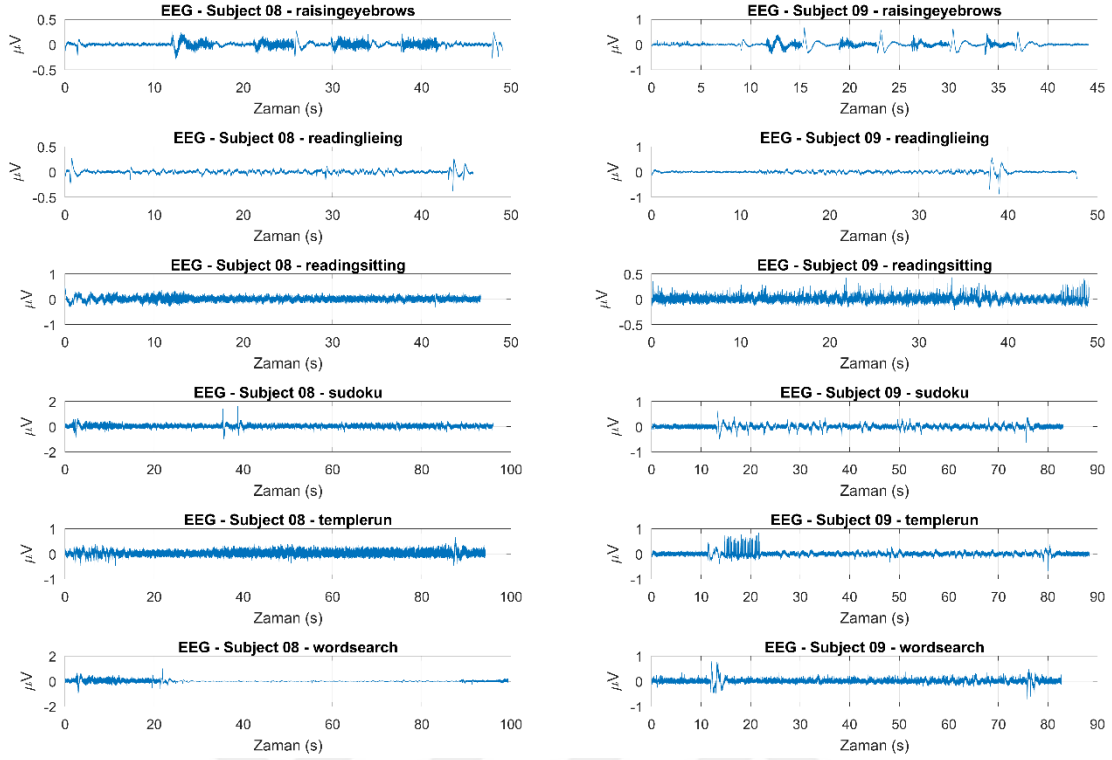
Tablo 2. EEG verisine ait istatistiksel özellikler

Sınıf Numarası	Minimum Değer (V)	Maksimum Değer (V)	Ortalama Değer (V)	Standart Sapma (V)	Toplam Süre (s)
1	-0.1308	0.1982	-23.0E-06	0.0184	166.49
2	-0.1600	0.1887	160E-06	0.0154	363.98
3	-0.0587	0.0783	8.40E-06	0.0052	981.47
4	-0.3661	0.1754	-15.0E-06	0.0122	256.07
5	-0.0002	0.0002	0.03E-06	0.0000	146.03
6	-0.2963	0.3701	-52.2E-06	0.0148	257.07
7	-0.0695	0.1165	-7.94E-06	0.0067	410.56
8	-0.0758	0.1131	-7.84E-06	0.0071	416.06
9	-0.0825	0.0975	1.51E-06	0.0060	1043.4
10	-0.0911	0.1369	5.03E-06	0.0054	1300.5
11	-0.0682	0.1006	-5.61E-06	0.0048	1089

Şekil 11 ve 12’de çalışmaya katılan iki kişiye ait EEG kayıtları gösterilmektedir.



Şekil 11. İlk 5 aktivitenin EEG kaydı (Katılımcı 8 solda, Katılımcı 9 sağda)



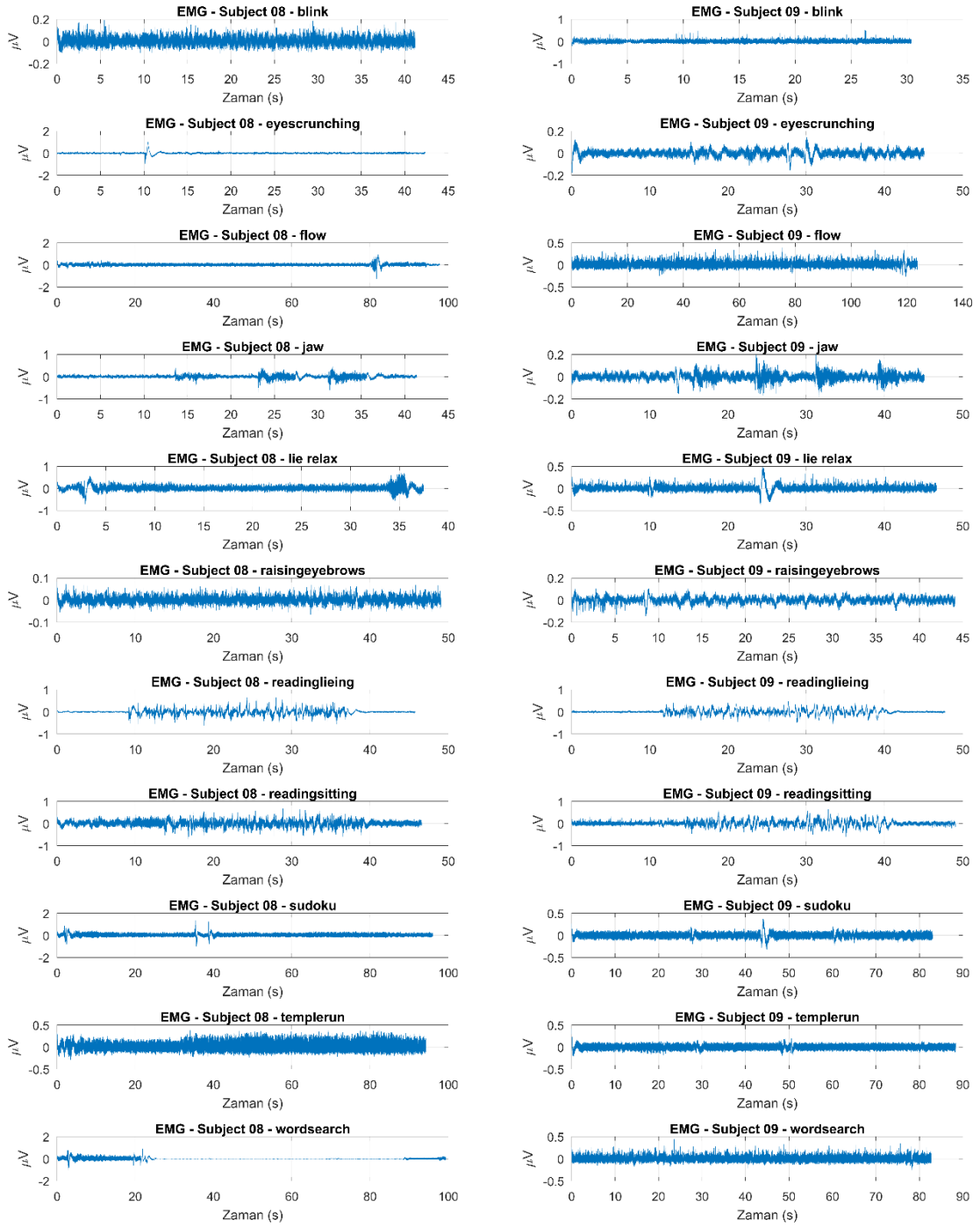
Şekil 12. Son 6 aktivitenin EEG kaydı (Katılımcı 8 solda, Katılımcı 9 sağda)

Benzer şekilde Tablo 3'te de her aktivitenin EMG kayıtlarına ait istatistiksel özellikler ve toplam örnek süreleri verilmiştir.

Tablo 3. EMG verisine ait istatistiksel özellikler

Sınıf Numarası	Minimum Değer (V)	Maksimum Değer (V)	Ortalama Değer (V)	Standart Sapma (V)	Toplam Süre (s)
1	-0.0447	0.0917	-16.6E-06	0.0036	166.49
2	-0.1026	0.0865	-4.85E-06	0.0052	363.98
3	-0.0831	0.0964	9.71E-06	0.0053	981.47
4	-0.2336	0.1952	-61.0E-06	0.0125	256.07
5	-0.0001	0.0001	0.02E-06	0.0000	146.03
6	-0.0481	0.0769	-19.4E-06	0.0042	257.07
7	-0.2048	0.1596	-9.23E-06	0.0141	410.56
8	-0.2089	0.1515	3.70E-06	0.0139	416.06
9	-0.1300	0.1463	-4.67E-06	0.0064	1043.4
10	-0.0899	0.0967	-7.12E-06	0.0046	1300.5
11	-0.1478	0.1950	2.55E-06	0.0062	1089

Şekil 13'te çalışmaya katılan iki kişiye ait EMG kayıtları gösterilmektedir.



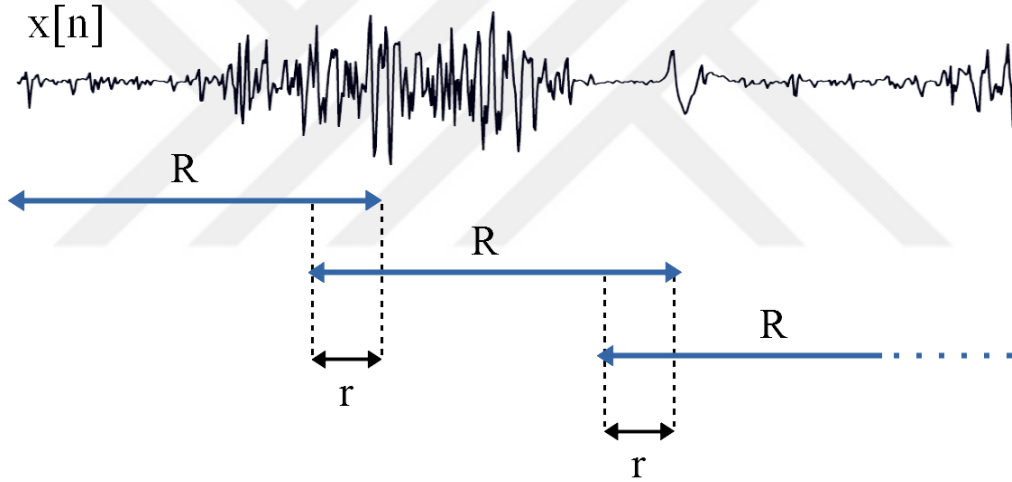
Şekil 13. İki katılımcıya ait EMG kayıtları (Katılımcı 8 solda, Katılımcı 9 sağda)

2.2. Pencereleme Yöntemleri

Pencereleme yöntemleri veri setindeki örneklerden daha fazla sayıda örnek elde etmeye yarayan veri bölütleme işlemleridir. Veri miktarını ve detayını artırmak sınıflandırma performansını artırabilir. Özellikle örnek sayısı az olan veri setleri için bu işlem tavsiye edilir. Pencereleme yöntemleri ikiye ayrılır:

2.2.1. Bitişik (Adjacent) Pencereleme

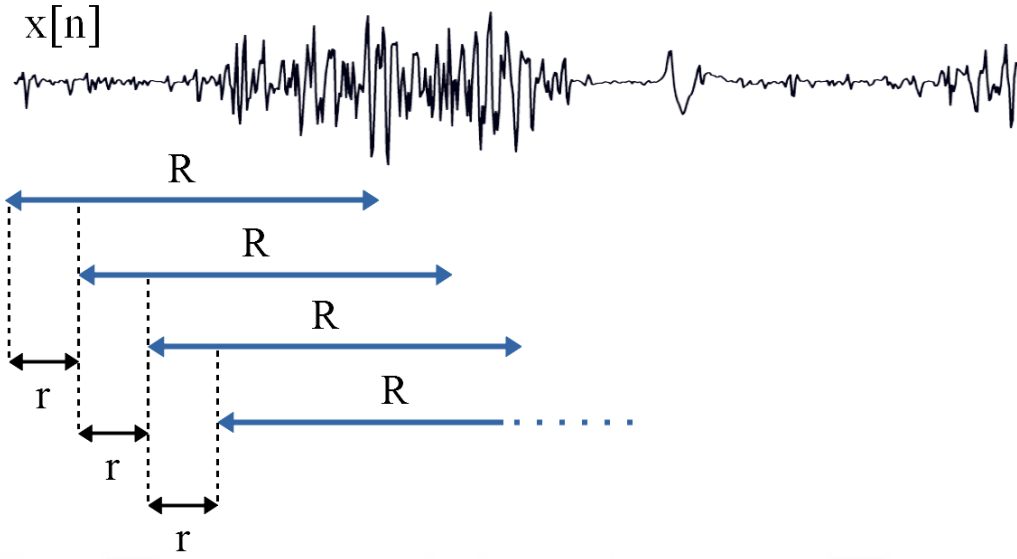
Bitişik pencereleme yönteminde işlenecek sinyalden R süresi uzunluğunda örnekler art arda ve birbirleriyle r süresi ($r < R$) boyunca kesişecek şekilde alınır. Bu işlem Şekil 14'te gösterilmiştir.



Şekil 14. Bitişik pencereleme yöntemi

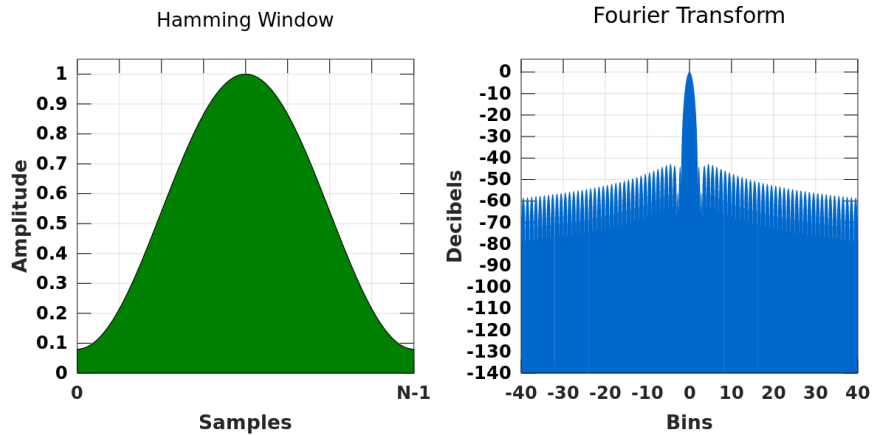
2.2.2. Üst Üste (Overlap) Pencereleme

Üst üste pencereleme yönteminde ise işlenecek sinyalden R süresi uzunluğunda örnekler sinyalin başlangıcından itibaren r süresi ($r < R$) boyunca kaydırılarak alınır. Bu işlem Şekil 15'te gösterilmiştir. Bu yöntem bitişik pencereleme yöntemine göre daha fazla örnek sayısı sağmasına karşın örnekler arası benzerlik daha fazladır.



Şekil 15. Üst üste pencereleme yöntemi

Bu çalışmada üst üste pencereleme yöntemi tercih edilmiştir. Pencere süresi (R) 2 saniye, kaydırma miktarı (r) 0.1 saniye olarak belirlenmiştir. Ayrıca elde edilen örnekler sonlu sinyaller olduklarından periyodik olarak tekrarlandıklarında süreksizlikler meydana getirir. Bu süreksizlikler sinyallerin frekans spektrumlarında istenmeyen bileşenlerin oluşmasına sebep olur. Bu bileşenlerin etkisini azaltmak için örneklerin sınırlarına yakın kısımlar çeşitli pencere fonksiyonları yardımıyla bastırılır. Çeşitli pencere fonksiyonları mevcuttur. Bunlardan bazıları Hamming, Hanning, Blackman-Harris fonksiyonlarıdır. Bu tez çalışmasında Hamming fonksiyonu kullanılmıştır. Hamming fonksiyonunun görüntüsü Şekil 16'da gösterilmiştir.



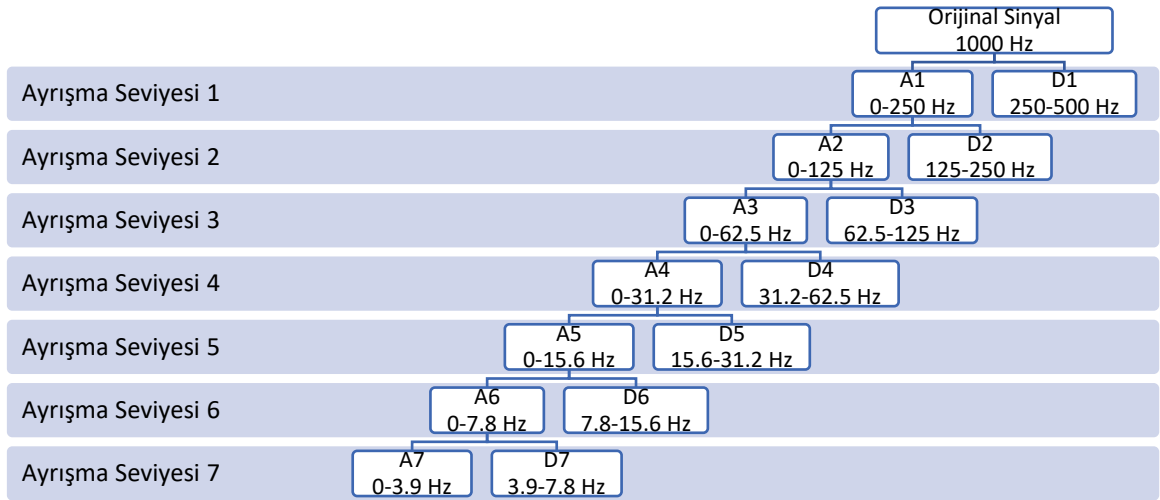
Şekil 16. Hamming fonksiyonu zaman ve frekans domeni görüntüsü [36].

2.3. Dalgacık Dönüşümü

Biyomedikal sinyallerin işlenmesinde en çok kullanılan yöntemlerden birisi dalgacık dönüşümüdür. Dalgacık dönüşümü zaman sinyallerinin değişken frekans bileşenlerinin frekans–zaman düzleminde görüntüsünü verir. Fourier dönüşümünden farklı olarak elde edilen frekans bileşenleri sabit değil, zaman eksenini boyunca değişkendir. Dalgacık dönüşümünün bu özelliği EEG gibi durağan olmayan frekans bileşenlerine sahip sinyallerin işlenmesine olanak verir. Bu sayede sinyal süresince meydana gelen spektral değişimler yakalanabilir.

2.3.1. Ayrık Dalgacık Dönüşümü

Ayrık dalgacık dönüşümü (ADD), bir sinyalin spektral bileşenlerini bir alçak geçiren ve bir de yüksek geçiren filtre yardımıyla iki parçaya ayırıp, düşük frekanslı bileşenlere bu işlemi tekrar eden bir dalgacık dönüşümüdür [20]. Bu işlem seçilen ayrışma seviyesi boyunca tekrar eder. Ayrışma seviyesi biyomedikal sinyalin örnekleme frekansına göre belirlenebilir. Nyquist teoremine göre ayrık sinyalin frekans spektrumu en fazla örnekleme frekansının yarısına kadar çizilebilir. Bu çalışmada EEG ve EMG sinyallerinin örnekleme frekansı 1 kHz olduğundan ayrık sinyalden elde edilebilecek en büyük frekans bileşeni 500 Hz olmaktadır. Bu frekans aralığı (0-500 Hz) bir EEG sinyalinin en düşük frekans bandına kadar 7 ayrışma seviyesinde bölünmektedir. Bu durum Şekil 17’de gösterilmiştir.



Şekil 17. 1000 Hz'lik bir sinyalin alt bantlarına ayrışması

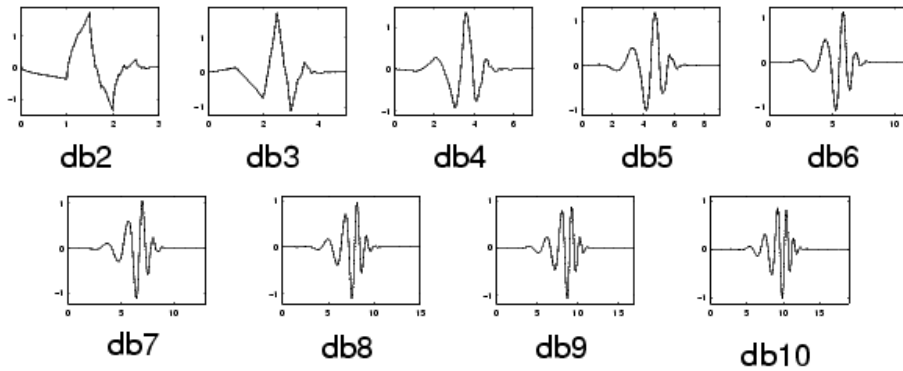
Şekil 17’de belirtilen yapıya göre, EEG bileşenlerine karşılık gelen dalgacıklar Tablo 4’te gösterilmiştir.

Tablo 4. EEG sinyalinin dalgacık dönüşümü ile elde edilen frekans bantları

Dalgacık Tipi	Frekans Aralığı (Hz)	EEG Bileşeni
D1	250-500	Gürültü
D2	125-250	Gürültü
D3	62.5-125	Gürültü
D4	31.2-62.5	γ
D5	15.6-31.2	β
D6	7.8-15.6	α
D7	3.9-7.8	θ
A7	0-3.9	δ

Tablo 4’e göre D1, D2 ve D3 sinyallerine ait frekans aralığı, EEG sinyalinin anlamlı olduğu aralığın dışında kaldığından gürültü olarak değerlendirilebilir. Bu sebeple çalışmanın geri kalan kısmında değerlendirilmeye alınmayacaklardır. Geri kalan alt bantlar sırasıyla gama (γ), beta (β), alfa (α), teta (θ) ve delta (δ) bileşenlerine karşılık gelmektedir.

Ayrık dalgacık dönüşümünde ayrıştırılan sinyal dalgacık adı verilen küçük sinyallerin bileşimi olarak tanımlanır. Bu dalgacıklardan bazıları Daubechies, Coiflet, Symmlet ve Meyer dalgacıkları olarak bilinir [37]. Bu çalışmada 4 seviyeli Daubechies dalgacığı kullanılmıştır. Şekil 18’de Daubechies dalgacık ailesi gösterilmektedir.



Şekil 18. Daubechies dalgacık ailesi [38].

2.4. Sınıflandırmada Kullanılan Öznitelik Fonksiyonları

Sınıflandırma işleminde ham veri örneklerini kullanmak yerine örneklerin çeşitli özelliklerini içeren daha düşük boyutlu temsillerini kullanmak sınıflandırma performansını artıracaktır. Öznitelik çıkarımında kullanılan fonksiyonlar zaman düzleminde, frekans düzleminde veya zaman-frekans düzleminde tanımlı olabilir. Bu çalışmada kullanılan öznitelik fonksiyonları zaman düzleminde elde edilmiştir. Bunlar sırasıyla ortalama mutlak değer, Hurst fraktal boyut indisi, Hjort parametreleri, verinin dağılımıyla ilgili olan basıklık değeri ve sıfır geçiş sayısıdır. Bu fonksiyonlar EEG, EEG alt frekans bileşenlerine (D4, D5, D6, D7 ve A7) ve son olarak EMG örneklerine uygulanmak suretiyle eğitim verileri elde edilmiştir.

2.4.1. Ortalama Mutlak Değer

Bir sinyalin ortalama mutlak değeri her örneğin mutlak değerinin toplamının örnek sayısına bölünmesiyle elde edilir. Matematiksel ifadesi denklem 68'de verilmiştir.

$$\bar{x} = \frac{\sum_{k=1}^n |x_k|}{n} \quad (68)$$

2.4.2. Hjort Parametreleri

Hjort parametreleri, EEG ve EMG sinyallerinin özellik çıkarımında tercih edilen fonksiyonlardan biridir. Bu parametreler sırasıyla etkinlik (activity), değişkenlik (mobility) ve karmaşıklık (complexity) parametreleridir [39]. Belirtilen parametrelerin matematiksel gösterimleri aşağıda verilmiştir.

$$Etkinlik(x) = var(x) \quad (69)$$

$$Değişkenlik(x) = \sqrt{\frac{Etkinlik\left(\frac{dx}{d}\right)}{Etkinlik(x)}} \quad (70)$$

$$Karmaşıklık(x) = \frac{Değişkenlik\left(\frac{dx}{d}\right)}{Değişkenlik(x)} \quad (71)$$

Etkinlik parametresi işaretin varyansı olarak tanımlanmıştır ve sinyal gücünü temsil eder. Değişkenlik, işaretin ortalama frekansını temsil eder. Son parametre olan karmaşıklık ise frekans değişiminin temsilidir ve sinyalin sinüs sinyaline olan benzerliğinin bir ölçüsüdür [40]. Bu tez çalışmasında değişkenlik parametresinin kullanılması tercih edilmiştir.

2.4.3. Basıklık

Basıklık, istatistiksel dağılım ölçülerinden bir tanesidir. Dağılımının tepe değerinin keskinliği basıklık özelliğiyle tanımlanır [41]. Matematiksel gösterimi aşağıda verilmiştir.

$$k = \frac{\sum_{i=1}^N (x_i - \bar{x})^4}{N\sigma^4} \quad (72)$$

2.4.4. Fraktal Boyutlar

Fraktallar, parçalarında kendisini tekrar eden yapılar olarak tanımlanabilir. Fraktal yapıları tanımlayan çeşitli matematiksel modeller bulunmaktadır. Bunlardan bir tanesi “Fraktal Brownian Hareketi” dir. Bu modelde üretilecek fraktal yapının derecesi “Hurst Parametresi” olarak tanımlanan bir sayı ile belirlenir ($0 < H < 1$) [42]. Fraktal analizdeki amaç ise incelenen veriye ait fraktal örüntü derecesini belirleyen parametreleri bulmaktır.

EEG sinyalleri incelendiğinde güç spektrumlarının fraktal yapılara benzediğini gösteren çalışmalar mevcuttur [43]. Fraktal boyut analizi bu sebeple EEG sinyallerinden özellik çıkarımı konusunda güçlü bir araç olmaktadır. Bu çalışmada Matlab kütüphanesinde bulunan `wfbmesti(x)` kestirim fonksiyonun kullanılmıştır. Kullanılan fonksiyon üç farklı kestirim yöntemiyle elde edilen Hurst indislerini verir. Eğitim verisinde kullanılan öznelik ise “Ayrık İkinci Türev Tabanlı Yaklaşım” metodu ile bulunan indistir [44].

2.4.5. Sıfır Geçiş Sayısı

Sıfır geçiş sayısı bir işaretin belirli bir zaman aralığında sıfır noktasına kaç kez uğradığının sayısıdır. Matematiksel gösterimi aşağıda verilmiştir.

$$sgs = \sum_{i=1}^N \text{sgn}(-x_i \times x_{i+1}) > 0 \quad (73)$$

2.5. Eğitim Verisinin Normalizasyonu

Hazırlanan veri setinin sınıflandırma aşamasından önce normalize edilmesi sınıflandırma performansı açısından önemlidir. Özelliklerin farklı genişlikte veya farklı konumda dağılımlara sahip olması bazı özelliklerin öğrenme aşamasına olan katkısını olumsuz yönde etkileyebilir. Böyle bir durumla karşılaşmamak için özelliklerin aynı sınırlar içinde olması sağlanmalıdır.

Bu çalışmada veriler minimum-maksimum yöntemiyle normalize edilmiştir. Böylelikle tüm özelliklerin 0-1 aralığına yayılmıştır. Minimum-maksimum normalizasyonu x_i özellik vektörüne aşağıdaki şekilde uygulanır.

$$x_{norm\ i} = \frac{x_i - \min(x_i)}{\max(x_i) - \min(x_i)} \quad (74)$$

2.6. Başarı Değerlendirme Ölçütleri

Sınıflandırma algoritmalarının değerlendirilmesinde kullanılan başarı ölçütleri karmaşıklık matrisi üzerinden elde edilebilir. Karmaşık matrisi, sınıflandırıcının yaptığı tahminlerin ne kadarının doğru ne kadarının yanlış olduğuyla ilgili bilgiler içeren bir tablodur. Karmaşıklık matrisinde her bir satır bir sınıfa ait tahmin sonuçlarını içerir. Pozitif olarak tanımlanan bir sınıf için yapılan doğru tahminlerin sayısı doğru pozitiflerin sayısını, yanlış tahminlerin sayısı ise yanlış pozitif sayısını oluşturur. Diğer sınıflar açısından düşünüldüğünde, negatif bir sınıfın yine kendisi olarak tahmin edilme sayısı doğru negatiflerin sayısını verirken, pozitif olarak tahmin edilme sayısı ise yanlış negatiflerin sayısını verir. Karmaşıklık matrisine ait örnek bir gösterim Tablo 5'te verilmiştir.

Tablo 5. Karmaşıklık matrisi (Confusion matrix)

Tahmin Gerçek	P (Pozitif)	N (Negatif)
P (Pozitif)	DP	YP
N (Negatif)	YN	DN

DP: Doğru Pozitif Sayısı

DN: Doğru Negatif Sayısı

YP: Yanlış Pozitif Sayısı

YN: Yanlış Negatif Sayısı

Bu çalışmada sınıflandırıcıların performanslarını değerlendirmek amacıyla doğruluk, doğru pozitif oranı, yanlış pozitif oranı, kesinlik (precision) ve f-skoru (f-score) metrikleri kullanılmıştır.

2.6.1. Doğruluk

Doğruluk, yapılan tüm doğru sınıflandırmaların tüm sınıflandırmalara oranı şeklinde tanımlanır. Matematiksel gösterimi aşağıdaki eşitlikte verilmiştir.

$$\text{Doğruluk} = \frac{DP + DN}{DP + DN + YP + YN} \quad (75)$$

2.6.2. Doğru Pozitif Oranı (DP Oranı)

Doğru pozitif oranı, pozitif olarak sınıflandırılmış örneklerin kaç tanesinin doğru olarak sınıflandırıldığına göstergesidir. Aynı zamanda duyarlılık (recall) veya hassaslık (sensitivity) olarak da bilinir. Matematiksel gösterimi aşağıdaki eşitlikte verilmiştir.

$$\text{DP Oranı} = \frac{DP}{DP + YN} \quad (76)$$

2.6.3. Yanlış Pozitif Oranı (YP Oranı)

Yanlış pozitif oranı, negatif olarak sınıflandırılmış örneklerin kaç tanesinin yanlış olarak sınıflandırıldığıнын göstergesidir. Mümkün olduğunca küçük olması istenir. Matematiksel gösterimi aşağıdaki eşitlikte verilmiştir.

$$YP \text{ Oranı} = \frac{YP}{YP + DN} \quad (77)$$

2.6.4. Kesinlik

Kesinlik, sınıflandırıcının herhangi bir sınıfa ait yaptığı tahminlerin ne kadarını doğru yaptığıнын oranıdır. Matematiksel gösterimi aşağıdaki eşitlikte verilmiştir.

$$Kesinlik = \frac{DP}{DP + YP} \quad (78)$$

2.6.5. F-Skoru (F-Score)

F-Skoru, doğru pozitif oranı (duyarlılık) ile kesinlik değerlerinin harmonik ortalamasıdır. Matematiksel gösterimi aşağıdaki eşitlikte verilmiştir.

$$F - Skoru = \frac{2DP}{2DP + YP + YN} \quad (79)$$

2.7. Tez Çalışması Akış Şeması

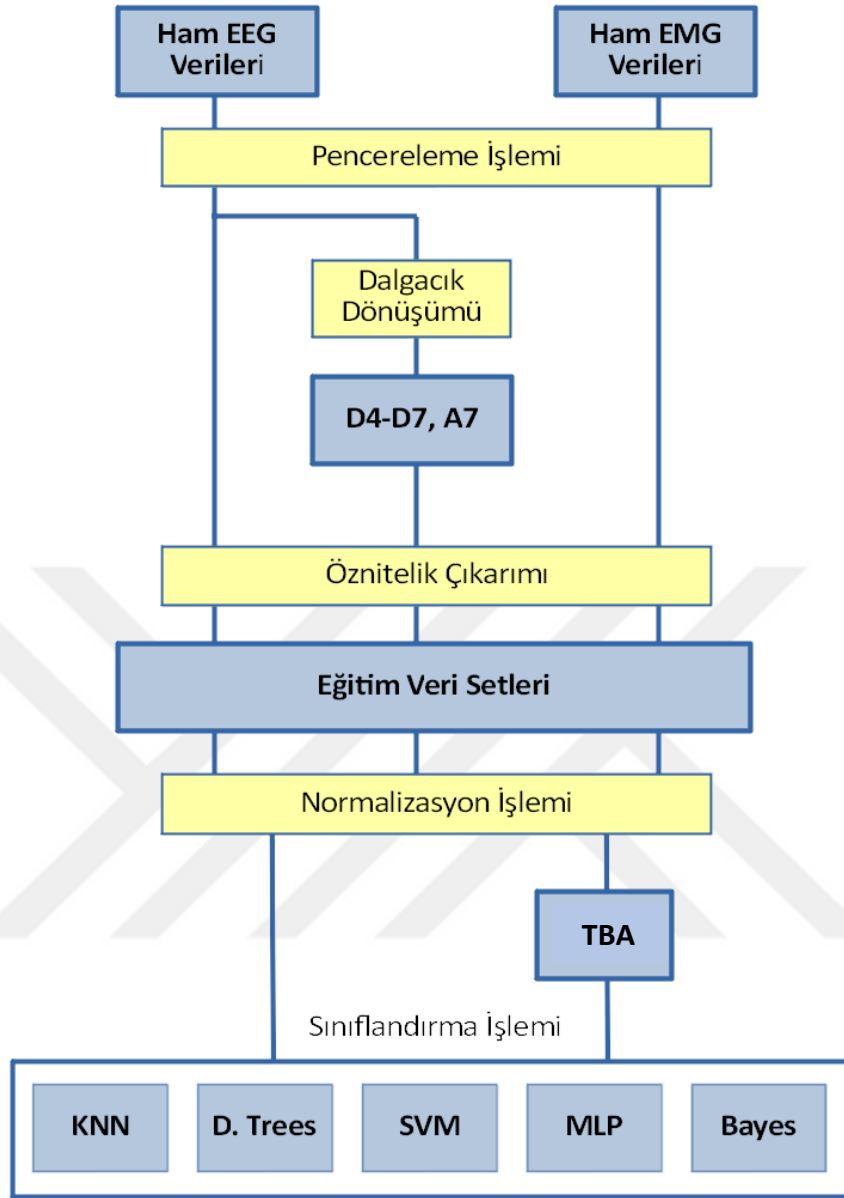
Tez çalışması temel olarak iki aşamadan oluşmaktadır. Bu aşamalar eğitim veri setinin hazırlanması ve sınıflandırma aşamalarıdır. Eğitim verisinin hazırlanmasında MATLAB, sınıflandırılmasında ise Weka programı kullanılmıştır. Sınıflandırma aşamasından önce EEG ve EMG kayıtları 50 Hz çentik filtre ile hat gürültüsünden arındırılmıştır. Ayrıca anlamlı bileşenler 0.5 Hz frekansından sonra başladığı için 0.5 Hz kesim frekanslı butterworth yüksek geçiren filtre uygulanmıştır. Daha sonra ham verilere üst üste pencereleme yöntemi uygulanmıştır.

Yapılan çeşitli ön sınıflandırma sonuçlarına göre yüksek başarılar üst üste pencereleme yönteminde, minimum 2 saniye uzunluğundaki pencere boyutlarında ve %90 örtüşme oranında sağlanmıştır. Çalışmanın geri kalan kısmında işlem süresini minimize etmek amacıyla bu parametrelerin kullanılması tercih edilmiştir. EEG işaretlerinden alfa, beta, gama, delta ve teta bileşenlerinin elde edilebilmesi için 7 seviyeli dalgacık dönüşümü uygulanmıştır. Sonraki aşamada EEG, EMG ve alt bileşenlere ait öznitelikler çıkarılmıştır. Çalışmada kullanılan yöntemler sırasıyla, ortalama mutlak değer, Hjorth tanımlayıcılarının değişkenlik parametresi, basıklık, Hurst fraktal indisi ve sıfır geçişleridir.

EEG, EMG ve EEG alt bileşenlerinden elde edilen öznitelik vektörleri sınıflandırılmadan önce normalize edilmiş ve bu sayede tüm veriler [0-1] aralığında ölçeklenmiştir. Oluşturulan öznitelik vektörleri KNN, J48, DVM, YSA ve Naif Bayes sınıflandırıcılarıyla sınıflandırılmak üzere kullanılmıştır. DVM sınıflandırıcısının kullanıldığı çalışmalar incelendiğinde; genellikle radyal tabanlı çekirdek fonksiyonun daha başarılı sınıflandırma performansına sahip olduğu görülmüştür [45]. Bu sebeple DVM sınıflandırıcısında radyal tabanlı çekirdek fonksiyonu tercih edilmiştir. Daha sonra veri kümelerinin birleşiminden elde edilen yeni veri setlerinin sınıflandırma sonuçları değerlendirilmiştir. Ayrıca sınıflandırma performansını iyileştirmek ve gürültü oranını azaltmak amacıyla Temel Bileşenler Analizi yöntemi kullanılmış ve sınıflandırmaya olan etkisi gözlemlenmiştir.

Tez çalışmasında yapılan sınıflandırmalar tüm katılımcılardan elde edilen verilerin birlikte kullanıldığı genel sınıflandırma çalışmalarıdır. Ancak biyomedikal sinyallerin kişiden kişiye oldukça değişkenlik göstermesi göz önüne alındığında, bireysel bazda yapılan sınıflandırmaların daha kararlı sonuçlar vereceği düşünülebilir. Bu sebeple Bulgular bölümünün son kısmına, 20 katılımcıya ait bireysel bazlı sınıflandırma sonuçlarının eklenmesi uygun görülmüştür.

Sınıflandırma sonuçlarını etkileyecek diğer bir problem veri setinin dengesiz bir yapıda olmasıdır. Dolayısıyla dengesiz verilerin sınıflandırmaya olan etkisinin araştırılması gerekmektedir. Bu amaçla dengeli veriler oluşturmak için “undersampling” metodu kullanılmıştır. Bu yöntemde her sınıfın örnek sayısı, en küçük örnek sayısına eşitlenir. Bu sebeple veri seti dengeli hale getirilmiş, sınıflandırma sonuçları ölçülmüş ve raporlanmıştır. Tez çalışmasında izlenecek adımlar Şekil 19’da verilen tez akış şemasında özetlenmiştir.



Şekil 19. Tez çalışması akış şeması

3. BULGULAR

Ham EEG ve EMG verisinden şebeke gürültüsü 50 Hz çentik filtre ile giderildi. Sınıflandırma işleminde aktivitelerin gerçekleştiği zaman dilimlerinden seçilen veriler kullanıldı. Daha sonra bu verilere üst üste pencereleme işlemi uygulandı. Pencere süresi 2 saniye, kaydırma süresi ise 0.1 saniye olarak belirlendi. Ayrıca her pencere Hamming fonksiyonu ile yeniden şekillendirildi. EEG verilerine 7 seviye dalgacık dönüşümü ile alt bantlar elde edildi.

Veri gruplarına öznitelik fonksiyonları uygulanmak suretiyle eğitim verileri oluşturuldu. Bu aşamada iki tür uygulama yapılarak ikinci bölümde belirtilen öznitelikler ile literatürde kullanımı yaygın diğer öznitelikler karşılaştırıldı. Eğitim verilerine minimum-maksimum normalizasyon işlemi uygulandı. Sınıflandırma işlemleri Weka programı kullanılarak yapıldı. Veri boyutunun uzunluğundan dolayı sınıflandırma algoritmaları 5 katlı çapraz doğrulama ile test edildi. Sınıflandırma başarısı en yüksek olan iki grup birleştirildi. Elde edilen veri setine ait sonuçlar değerlendirildi.

Ayrıca sınıf dağılımının sınıflandırmaya olan etkisini gözlemlemek amacıyla eğitim verisinin sınıf dağılımı undersampling yöntemiyle eşitlenmiştir. Hem dengesiz hem de dengeli veri setlerine ait sınıflandırma sonuçları sunulmuştur. Son olarak sınıflandırma başarısını iyileştirmek ve veri boyutunu küçültmek amacıyla Temel Bileşen Analizi uygulanmıştır. Elde edilen sonuçlara ait karmaşıklık matrisleri hazırlanmıştır.

3.1. Dengelenmemiş Verilerin Sınıflandırma Sonuçları

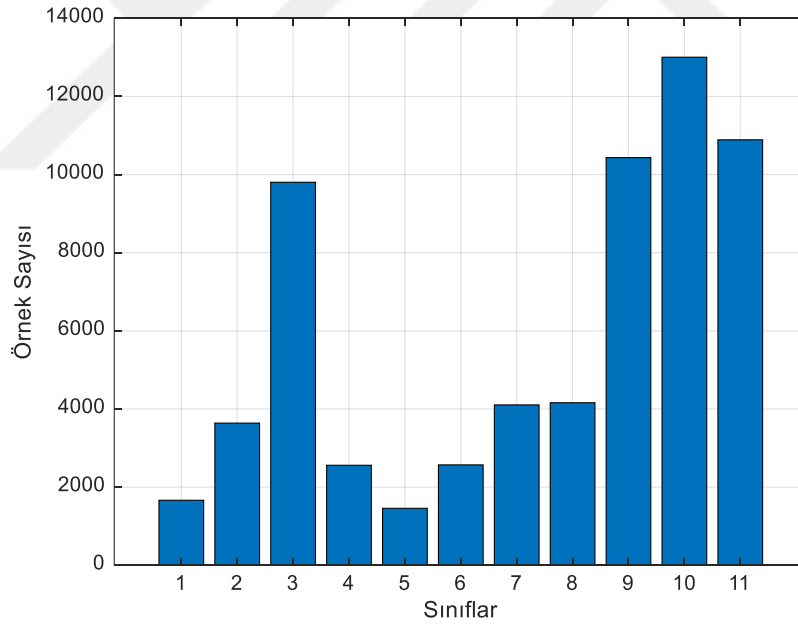
3.1.1. Uygulama 1

Bu uygulama bir önceki bölümde yer alan öznitelik fonksiyonları kullanıldı. Kullanılan öznitelikler Tablo 6'da verilmiştir.

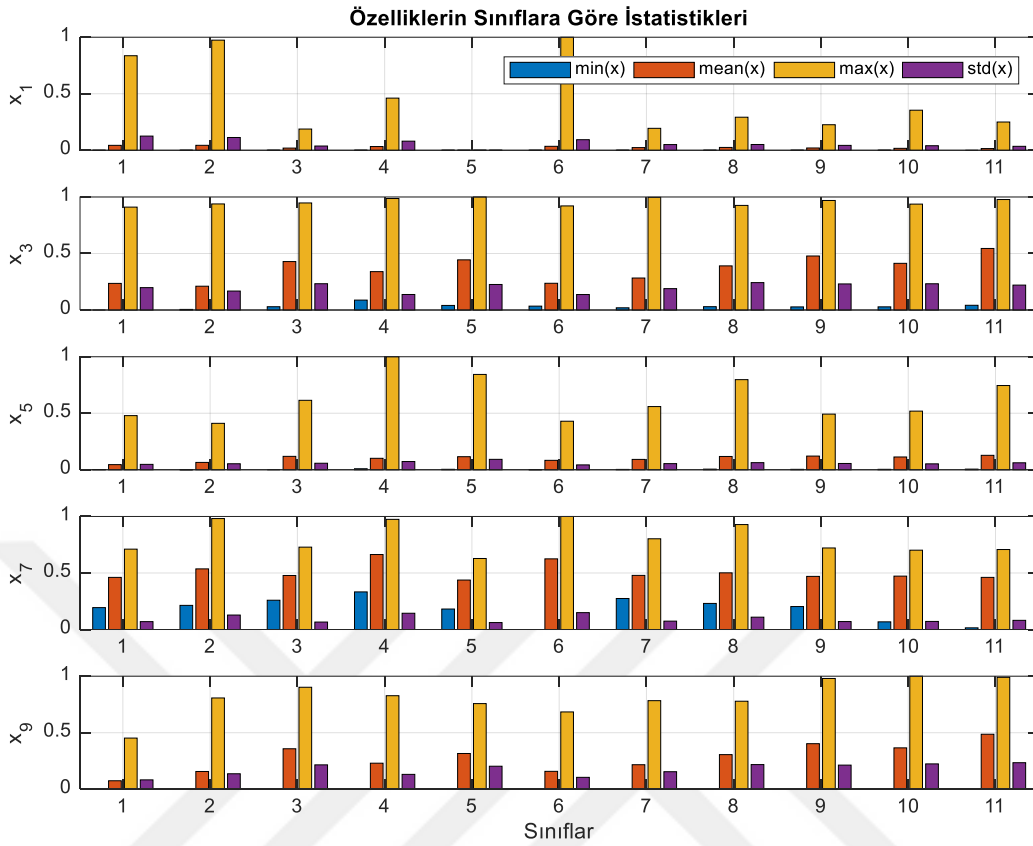
Tablo 6. Uygulama 1 için kullanılan öznitelikler

EEG Veri Seti		EMG Veri Seti	
Kısaltma	Öznitelik Fonksiyonu	Kısaltma	Öznitelik Fonksiyonu
x ₁	Ortalama Mutlak Değer	x ₂	Ortalama Mutlak Değer
x ₃	Değişkenlik	x ₄	Değişkenlik
x ₅	Basıklık	x ₆	Basıklık
x ₇	Fraktal Boyut İndisi	x ₈	Fraktal Boyut İndisi
x ₉	Sıfır Geçiş Sayısı	x ₁₀	Sıfır Geçiş Sayısı

Ayrıca sınıfların dağılımı Şekil 20'deki grafikte gösterilmiştir. Uygulamada öncelikle EEG verilerinin sınıflandırılması yapılmıştır. EEG verilerinden elde edilen özniteliklere ait istatistikler Şekil 21'deki grafiklerde sunulmuştur.



Şekil 20. Sınıfların dağılımı



Şekil 21. EEG veri seti özneliklerine ait istatistikler

EEG verilerini sınıflandırmada KNN, J48, DVM, YSA ve Naif Bayes algoritmaları kullanılmıştır. KNN yönteminde en yakın komşu sayısı (k) 1 olarak alınmıştır. DVM (libSVM) yönteminde radyal tabanlı çekirdek fonksiyonu (gamma=2) kullanılmıştır. YSA yönteminde gizli katmanlar 30,48 olarak seçilmiş, öğrenme oranı 0.1 alınmış ve eğitim 250 iterasyonda yapılmıştır. Tüm algoritmalar 5 katlı çapraz doğrulama ile test edilmiştir. 10 katlı çapraz doğrulama yerine 5 katlı çapraz doğrulama seçilmesinin sebebi yüksek veri boyutundan ötürü uzayan işlem süresini azaltmaktır. Elde edilen sonuçlar Tablo 7'de verilmiştir.

Tablo 7. Uygulama 1 EEG verisine ait sınıflandırma sonuçları

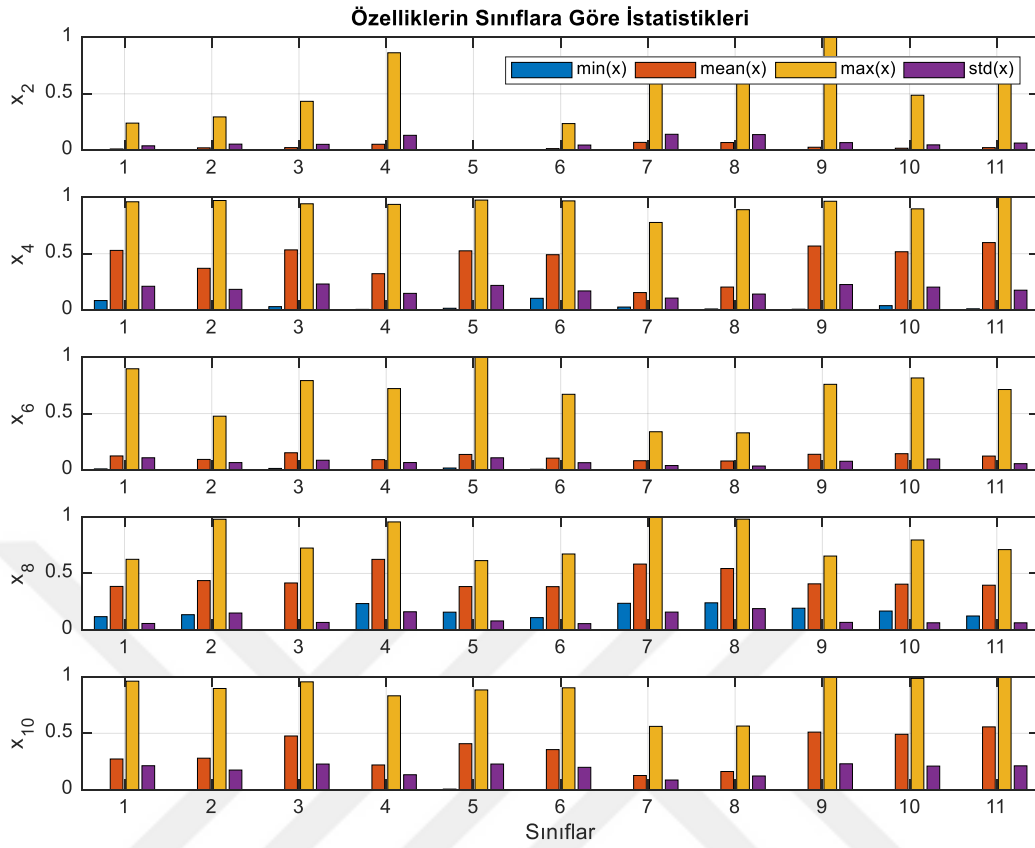
	Doğruluk	Duyarlılık	YP Oranı	Kesinlik	F-Skoru
KNN	0,587	0,587	0,066	0,587	0,587
J48	0,673	0,673	0,053	0,673	0,672
DVM	0,393	0,393	0,119	0,424	0,380
YSA	0,473	0,473	0,092	0,475	0,468
N. Bayes	0,115	0,115	0,047	0,241	0,119

Bir sonraki işlemde EEG verisine ait alt bantlar J48 algoritmasıyla sınıflandırılmıştır. J48 algoritmasının kullanılmasındaki amaç EEG verisine göre daha yüksek başarı elde edilip edilemeyeceğinin gözlemlenmesidir.

Tablo 8. EEG alt bantlarına ait sonuçlar (J48)

	Doğruluk	Duyarlılık	YP Oranı	Kesinlik	F-Skoru
D4	0,306	0,306	0,110	0,305	0,305
D5	0,303	0,303	0,111	0,302	0,302
D6	0,263	0,263	0,115	0,263	0,263
D7	0,209	0,209	0,124	0,208	0,208
A7	0,251	0,251	0,121	0,248	0,249

EEG alt bantları düşük sınıflandırma performansına sahiptir. Dolayısıyla çalışmanın bundan sonraki kısmında kullanılmaları tercih edilmemiştir. Uygulamada daha sonra EMG verilerinin sınıflandırılması yapılmıştır. EMG verilerinden elde edilen özneliklere ait istatistikler Şekil 22'deki grafiklerde sunulmuştur.



Şekil 22. EMG veri seti özneliklerine ait istatistikler

EMG verilerini sınıflandırmada KNN, J48, DVM, YSA ve Naif Bayes algoritmaları kullanılmıştır. KNN yönteminde en yakın komşu sayısı (k) 1 olarak alınmıştır. DVM (libSVM) yönteminde radyal tabanlı çekirdek fonksiyonu (gama=2) kullanılmıştır. YSA yönteminde gizli katmanlar 30,48 olarak seçilmiş, öğrenme oranı 0.1 alınmış ve eğitim 250 iterasyonda yapılmıştır. Tüm algoritmalar 5 katlı çapraz doğrulama ile test edilmiştir. Elde edilen sonuçlar Tablo 9’da verilmiştir.

Tablo 9. Uygulama 1 EMG verisine ait sınıflandırma sonuçları

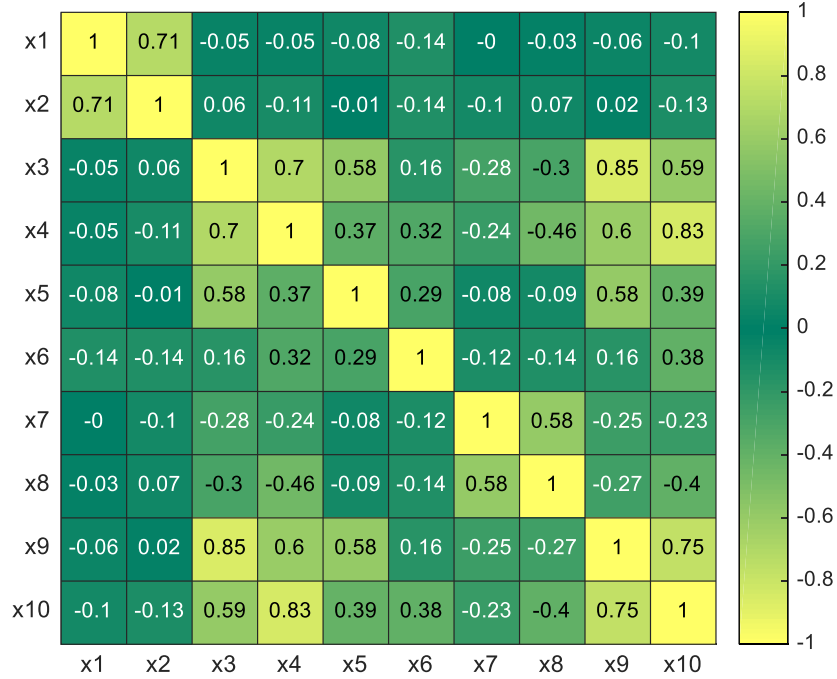
	Doğruluk	Duyarlılık	YP Oranı	Kesinlik	F-Skoru
KNN	0,665	0,665	0,052	0,665	0,665
J48	0,741	0,741	0,041	0,741	0,741
DVM	0,452	0,452	0,100	0,465	0,440
YSA	0,531	0,531	0,080	0,527	0,524
N. Bayes	0,119	0,119	0,048	0,189	0,103

Elde edilen veri gruplarından en yüksek başarıya sahip ilk iki grup EEG ve EMG verileridir. Bu sebeple EEG ve EMG öznitelikleri birlikte sınıflandırılmıştır. Elde edilen sonuçlar Tablo 10’da verilmiştir. Ayrıca özniteliklerin birbiriyle olan korelasyonları Şekil 23’te sunulmuştur.

Tablo 10. Uygulama 1 EEG+EMG verisine ait sınıflandırma sonuçları

	Doğruluk	Duyarlılık	YP Oranı	Kesinlik	F-Skoru
KNN	0,947	0,947	0,008	0,946	0,946
J48	0,853	0,853	0,024	0,853	0,853
DVM	0,689	0,689	0,057	0,692	0,688
YSA	0,681	0,681	0,056	0,681	0,679
N. Bayes	0,169	0,169	0,040	0,295	0,175

Tablo 10’da verilen sonuçlar incelendiğinde en yüksek başarı %94.7 ile KNN yönteminde sağlanmıştır. KNN sınıflandırma sonuçlarından elde edilen karmaşıklık matrisi Şekil 24’te gösterilmiştir.



Şekil 23. Uygulama 1 EEG+EMG özelliklerinin Pearson korelasyonu

Confusion Matrix

True class	blink	86.5% 1439	3.1% 52	1.6% 27	1.2% 20	0.6% 10	1.4% 23	0.2% 4	0.4% 6	1.4% 24	1.8% 30	1.7% 29
	eyescrunching	1.2% 45	94.2% 3428	0.6% 21	0.1% 5	0.4% 14	1.0% 35	0.5% 17	0.2% 7	0.4% 16	0.9% 33	0.5% 18
	flow	0.3% 28	0.2% 15	93.7% 9189	0.2% 16	0.3% 30	0.2% 17	0.1% 12	0.1% 8	1.9% 189	1.8% 173	1.3% 126
	jaw	0.9% 23	0.2% 6	0.7% 19	94.3% 2413	0.2% 5	0.5% 12	0.6% 16	0.5% 14	0.9% 23	0.4% 10	0.7% 19
	lie relax	1.0% 14	0.3% 5	2.1% 30	0.5% 7	91.2% 1330	0.3% 5	0.2% 3	0.3% 4	1.6% 23	1.3% 19	1.3% 19
	raisingeyebrows	0.8% 21	1.4% 37	0.7% 19	0.3% 7	0.2% 4	94.9% 2438	0.1% 2		0.4% 9	0.7% 19	0.5% 14
	readinglieing	0.1% 3	0.3% 11	0.2% 8	0.3% 11	0.1% 3	0.1% 3	97.8% 4013	0.9% 35	0.0% 2	0.2% 7	0.2% 9
	readingsitting	0.2% 9	0.1% 5	0.1% 6	0.5% 19	0.1% 6		1.1% 47	96.9% 4033	0.3% 14	0.3% 12	0.2% 9
	sudoku	0.2% 22	0.1% 12	1.6% 167	0.1% 10	0.3% 29	0.1% 15	0.1% 7	0.2% 25	93.9% 9801	1.9% 194	1.4% 151
	templerun	0.2% 21	0.2% 30	1.2% 156	0.1% 12	0.2% 20	0.2% 20	0.1% 14	0.1% 15	1.3% 167	95.3% 12398	1.2% 150
	wordsearch	0.3% 28	0.1% 14	1.2% 130	0.1% 16	0.2% 24	0.2% 20	0.1% 7	0.1% 12	1.2% 126	1.4% 148	95.2% 10364
			blink	eyescrunching	flow	jaw	lie relax	raisingeyebrows	readinglieing	readingsitting	sudoku	templerun
		Predicted class										

Şekil 24. Uygulama 1 EEG+EMG verisine ait KNN karmaşıklık matrisi

3.1.1. Uygulama 2

Bu uygulama, literatürde en yaygın kullanılan özniteliklerin Uygulama 1 ile kıyaslanması amacıyla yapılmıştır. Kullanılan öznitelikler Tablo 11’de verilmiştir. Bu öznitelikler sırasıyla veri örneklerinin minimum değeri, maksimum değeri, ortalama değeri, standart sapması ve karekök ortalama fonksiyonlarıdır. Bir sinyalin karekök ortalama denklem 80’de tanımlandığı gibi hesaplanır. Sınıflandırma algoritmalarında kullanılan parametreler ilk uygulama ile aynıdır. Test metodu olarak 5 katlı çapraz doğrulama kullanılmıştır.

$$x_{rms} = \sqrt{\frac{\sum_{i=1}^n x_i^2}{n}} \quad (80)$$

Tablo 11. Uygulama 2 için kullanılan öznitelikler

EEG Veri Seti		EMG Veri Seti	
Kısaltma	Öznitelik Fonksiyonu	Kısaltma	Öznitelik Fonksiyonu
x ₁	Minimum Değer	x ₂	Minimum Değer
x ₃	Ortalama Değer	x ₄	Ortalama Değer
x ₅	Maksimum Değer	x ₆	Maksimum Değer
x ₇	Standart Sapma	x ₈	Standart Sapma
x ₉	RMS Değeri	x ₁₀	RMS Değeri

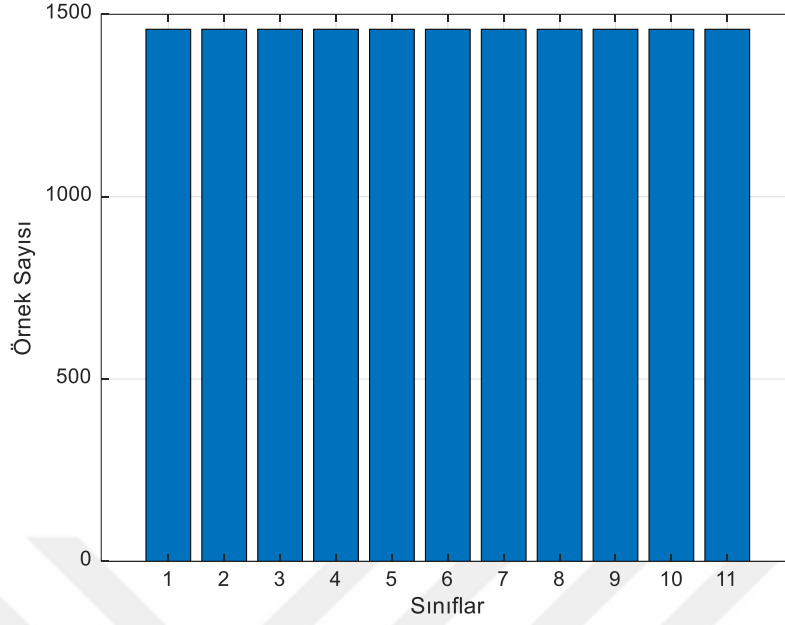
Uygulama 1 de en yüksek başarı hem EEG hem de EMG verilerinin kullanıldığı zaman elde edilmiştir. Bunun üzerine kullanılan özniteliklerin etkisini kıyaslamak amacıyla bu verilere Tablo 11’de verilen öznitelikler uygulanmıştır. Sınıflandırma sonuçları Tablo 12’de gösterilmiştir.

Tablo 12. Uygulama 2 EEG+EMG verisine ait sınıflandırma sonuçları

	Doğruluk	Duyarlılık	YP Oranı	Kesinlik	F-Skoru
KNN	0,745	0,745	0,044	0,745	0,745
J48	0,314	0,314	0,168	0,618	0,268
DVM	0,271	0,271	0,179	0,571	0,196
YSA	0,256	0,256	0,168	0,492	0,197
N. Bayes	0,060	0,060	0,048	0,240	0,053

3.2. Dengelenmiş Verilerin Sınıflandırma Sonuçları

Sınıf dağılımlarının sınıflandırma sonuçlarına olan etkisini gözlemlemek için ilk uygulamada en yüksek doğruluk oranının elde edildiği veri setine sınıf dengeleme yöntemi uygulanmış ve yeniden sınıflandırma işlemi gerçekleştirilmiştir. Bu yöntemde tüm sınıflardan en az örneğe sahip sınıfın eleman sayısı kadar örnek rastgele olarak seçilmiştir. Elde edilen dağılım Şekil 25’te verilmiştir.



Şekil 25. Dengeli sınıfların dağılımı

Bu aşamada öncelikle EEG verilerinin sınıflandırılması yapılmıştır. EEG verilerini sınıflandırmada KNN, J48, DVM, YSA ve Naif Bayes algoritmaları kullanılmıştır. KNN yönteminde en yakın komşu sayısı (k) 1 olarak alınmıştır. DVM (libSVM) yönteminde radyal tabanlı çekirdek fonksiyonu (gama=2) kullanılmıştır. YSA yönteminde gizli katmanlar 30,48 olarak seçilmiş, öğrenme oranı 0.1 alınmış ve eğitim 250 iterasyonda yapılmıştır. Tüm algoritmalar 5 katlı çapraz doğrulama ile test edilmiştir. Elde edilen sonuçlar Tablo 13'te verilmiştir.

Tablo 13. Dengeli EEG verisine ait sınıflandırma sonuçları

	Doğruluk	Duyarlılık	YP Oranı	Kesinlik	F-Skoru
KNN	0,540	0,540	0,046	0,537	0,538
J48	0,597	0,597	0,040	0,594	0,595
DVM	0,432	0,432	0,057	0,442	0,412
YSA	0,432	0,432	0,057	0,429	0,422
N. Bayes	0,224	0,224	0,078	0,293	0,181

Daha sonra EMG verilerinin sınıflandırılması aynı algoritmalar ve aynı parametreler ile yapılmıştır. Tüm algoritmalar 5 katlı çapraz doğrulama ile test edilmiştir. Elde edilen sonuçlar Tablo 14'te gösterilmiştir.

Tablo 14. Dengeli EMG verisine ait sınıflandırma sonuçları

	Doğruluk	Duyarlılık	YP Oranı	Kesinlik	F-Skoru
KNN	0,588	0,588	0,041	0,587	0,587
J48	0,651	0,651	0,035	0,650	0,650
DVM	0,413	0,413	0,059	0,419	0,407
YSA	0,443	0,443	0,056	0,452	0,443
N. Bayes	0,207	0,207	0,079	0,238	0,143

Son olarak EEG ve EMG verileri birleştirilerek elde edilen yeni veri seti sınıflandırılmıştır. Elde edilen sonuçlar Tablo 15'te verilmiştir.

Tablo 15. Dengeli EEG+EMG verisine ait sınıflandırma sonuçları

	Doğruluk	Duyarlılık	YP Oranı	Kesinlik	F-Skoru
KNN	0,831	0,831	0,017	0,828	0,829
J48	0,751	0,751	0,025	0,749	0,750
DVM	0,706	0,706	0,029	0,702	0,702
YSA	0,667	0,667	0,033	0,664	0,663
N. Bayes	0,284	0,284	0,072	0,381	0,256

3.3. Boyutu Azaltılmış Verilerin Sınıflandırma Sonuçları

Sınıflandırılma sonuçlarında yüksek doğruluk oranı elde edilmesine karşın üzerinde çalışılan veri 10 adet öznelikten oluşmaktadır. Sınıflandırma performansı özneliklerin sayısına, taşıdığı bilgi miktarına ve gürültülü olup olmamasına göre değişmektedir. Bu sebeple eğitim verisindeki gereksiz bileşenlerin ve gürültünün etkisini azaltmak amacıyla boyut azaltma işleminin uygulanması tercih edilmiştir. Böylelikle sınıflandırma algoritmalarının daha kararlı ve hızlı çalışması hedeflenmiştir. Çalışmada dengelenmemiş verilere TBA boyut azaltma yöntemi uygulanmış ve başarıda meydana gelen değişimler incelenmiştir.

Bu aşamada öncelikle EEG verilerine TBA boyut azaltma yöntemi uygulanmıştır daha sonra sınıflandırması yapılmıştır. TBA yönteminde verilerin toplam varyansının yaklaşık %95' inin korunması istenmiştir. 5 Boyutlu EEG verisinin işlem sonrası 4 boyuta indirildiği görülmüştür. Sınıflandırma aşamasında KNN, J48, DVM, YSA ve Naif Bayes algoritmaları kullanılmıştır. KNN yönteminde en yakın komşu sayısı (k) 1 olarak alınmıştır. DVM (libSVM) yönteminde radyal tabanlı çekirdek fonksiyonu (gamma=2) kullanılmıştır. YSA yönteminde gizli katmanlar 30,48 olarak seçilmiş, öğrenme oranı 0.1 alınmış ve eğitim 250 iterasyonda yapılmıştır. Tüm algoritmalar 5 katlı çapraz doğrulama ile test edilmiştir. Elde edilen sonuçlar Tablo 16'da verilmiştir.

Tablo 16. Boyutu azaltılmış EEG verisine ait sınıflandırma sonuçları

	Doğruluk	Duyarlılık	YP Oranı	Kesinlik	F-Skoru
KNN	0,463	0,463	0,085	0,463	0,463
J48	0,442	0,442	0,088	0,442	0,442
DVM	0,529	0,529	0,082	0,534	0,527
YSA	0,426	0,426	0,102	0,430	0,421
N. Bayes	0,259	0,259	0,153	0,301	0,216

Daha sonra EMG verilerine TBA yöntemi uygulanmıştır. TBA yönteminde verilerin toplam varyansının yaklaşık %95' inin korunması istenmiştir. 5 Boyutlu EMG verisinin işlem sonrası 4 boyuta indirildiği görülmüştür.

Elde edilen veri setinin sınıflandırılması Uygulama 1 de kullanılan aynı algoritmalar ve aynı parametreler ile yapılmıştır. Tüm algoritmalar 5 katlı çapraz doğrulama ile test edilmiştir. Elde edilen sonuçlar Tablo 17’de gösterilmiştir.

Tablo 17. Boyutu azaltılmış EMG verisine ait sınıflandırma sonuçları

	Doğruluk	Duyarlılık	YP Oranı	Kesinlik	F-Skoru
KNN	0,527	0,527	0,074	0,526	0,527
J48	0,507	0,507	0,077	0,506	0,506
DVM	0,579	0,579	0,074	0,581	0,574
YSA	0,496	0,496	0,088	0,491	0,486
N. Bayes	0,257	0,257	0,125	0,210	0,195

Son olarak EEG ve EMG verileri birleştirilerek elde edilen 10 boyutlu veri setine TBA uygulanmıştır. TBA yönteminde verilerin toplam varyansının yaklaşık %95’ inin korunması istenmiştir. 10 Boyutlu veri setinin işlem sonrası 8 boyuta indirildiği görülmüştür. Daha sonra Uygulama 1 de kullanılan aynı parametreler ile sınıflandırılmıştır. Elde edilen sonuçlar Tablo 18’de verilmiştir.

Tablo 18. Boyutu azaltılmış EEG+EMG verisine ait sınıflandırma sonuçları

	Doğruluk	Duyarlılık	YP Oranı	Kesinlik	F-Skoru
KNN	0,910	0,910	0,015	0,910	0,910
J48	0,726	0,726	0,045	0,727	0,727
DVM	0,913	0,913	0,016	0,914	0,913
YSA	0,681	0,681	0,056	0,680	0,679
N. Bayes	0,339	0,339	0,119	0,356	0,310

EEG ve EMG öznelikleri birleştirilip boyut azalma işlemi uygulandıktan sonra yapılan sınıflandırmada en yüksek başarı %91.3 ile DVM yönteminde sağlanmıştır. Boyut azaltma sonrası DVM ve Naif Bayes yöntemlerinden elde edilen doğruluk oranlarının yükseldiği, KNN ve J48 algoritmalarından ise bir miktar düştüğü gözlemlenmiştir. DVM sınıflandırma sonuçlarından elde edilen karmaşıklık matrisi Şekil 26’da verilmiştir.

Confusion Matrix

True class \ Predicted class	blink	eyescrunching	flow	jaw	lie relax	raisingeyebrows	readinglieing	readingsitting	sudoku	templerun	wordsearch
blink	84.1% 1399	3.1% 52	2.2% 36	0.5% 8	0.5% 8	1.3% 21	0.2% 4	0.4% 6	1.4% 23	4.4% 74	2.0% 33
eyescrunching	1.2% 43	92.9% 3379	0.7% 27	0.2% 8	0.2% 8	0.9% 33	0.2% 9	0.2% 8	0.8% 29	2.1% 78	0.5% 17
flow	0.2% 24	0.2% 24	88.7% 8700	0.1% 12	0.2% 21	0.1% 10	0.1% 8	0.1% 7	3.6% 349	4.5% 440	2.1% 208
jaw	0.5% 12	0.4% 11	0.7% 17	93.6% 2397	0.1% 3	0.5% 13	0.7% 18	0.3% 8	0.7% 19	1.8% 46	0.6% 16
lie relax	1.0% 14	0.5% 7	2.2% 32	0.3% 5	88.2% 1287	0.3% 4	0.3% 5	0.5% 7	1.6% 23	3.3% 48	1.9% 27
raisingeyebrows	0.6% 15	2.2% 56	1.2% 32	0.2% 5	0.1% 2	92.0% 2365	0.1% 2	0.1% 2	0.8% 20	2.2% 56	0.7% 17
readinglieing		0.3% 12	0.2% 8	0.2% 9	0.0% 1	0.0% 2	95.5% 3920	2.1% 85	0.2% 8	1.3% 53	0.2% 7
readingsitting	0.1% 4	0.3% 12	0.2% 7	0.1% 4	0.1% 6	0.0% 1	2.3% 95	95.2% 3961	0.2% 7	1.3% 52	0.3% 11
sudoku	0.2% 19	0.2% 25	2.6% 273	0.1% 7	0.2% 20	0.2% 18	0.0% 5	0.1% 10	90.0% 9394	3.9% 402	2.5% 260
templerun	0.2% 21	0.4% 53	2.2% 290	0.1% 11	0.2% 23	0.2% 27	0.1% 13	0.1% 16	2.6% 338	92.0% 11960	1.9% 251
wordsearch	0.2% 21	0.1% 14	2.2% 244	0.2% 18	0.2% 24	0.1% 14	0.0% 4	0.1% 8	2.6% 280	2.9% 321	91.3% 9941

Şekil 26. Boyutu azaltılmış EEG+EMG verisine ait DVM karmaşıklık matrisi

3.4. Bireysel Bazda Yapılan Sınıflandırma Sonuçları

Bu bölümde Tablo 6'da belirtilen EEG ve EMG öznelikleriyle hazırlanmış veri setinin kişi bazlı sınıflandırmaları yapılmıştır. Sınıflandırmalar 5 katlı çapraz doğrulama ile yapılmış ve aynı parametreler kullanılmıştır. Tablo 19'da 9 aktiviteyi gerçekleştiren 2 katılımcının sınıflandırmalarına ait doğruluk yüzdeleri yer almaktadır.

Tablo 19. Birey bazlı sınıflandırma sonuçları (9 aktivite)

Denek Numarası	KNN	J48	DVM	YSA	N. Bayes
4	98.6662	96.5655	98.9663	96.7322	85.5952
6	98.0612	96.8878	98.3673	94.7449	85.051
Ortalama	98.3637	96.7267	98.6668	95.7386	85.3231

Tablo 20’de 10 aktiviteyi gerçekleştiren 4 katılımcının sınıflandırmalarına ait doğruluk yüzdeleri yer almaktadır.

Tablo 20. Birey bazlı sınıflandırma sonuçları (10 aktivite)

Denek Numarası	KNN	J48	DVM	YSA	N. Bayes
2	98.0495	90.5882	93.7461	89.226	68.6997
3	98.8663	97.7326	99.2151	97.936	90.1163
5	98.5771	93.197	97.1543	93.3304	81.414
22	99.7452	98.4713	99.7962	93.4268	89.8089
Ortalama	98.8095	94.9973	97.4779	93.4798	82.5097

Tablo 21’de 11 aktiviteyi gerçekleştiren 14 katılımcının sınıflandırmalarına ait doğruluk yüzdeleri yer almaktadır.

Tablo 21. Birey bazlı sınıflandırma sonuçları (11 aktivite)

Denek Numarası	KNN	J48	DVM	YSA	N. Bayes
8	98.2916	94.0395	94.7988	92.407	86.2946
9	98.4779	93.9422	96.1644	89.5282	82.4049
10	97.6692	91.2782	95.4887	91.5038	79.5865
13	96.4163	92.2608	94.472	83.3397	76.9729
14	97.1028	92.8037	93.4579	88.8162	83.1776
16	95.5014	95.3009	96.6189	94.4699	91.5186
17	98.6536	95.6522	98.1206	93.4923	92.23
18	94.2731	88.37	88.7445	85.9471	72.1145
20	96.4343	90.226	95.1289	91.5632	73.7982
21	97.7067	94.1867	98.24	94.48	72.4267
23	97.6143	89.6321	92.7982	84.2586	65.641
24	96.4399	84.2547	90.4197	83.9363	61.8234
26	97.1244	92.1486	93.9257	88.0129	77.2213
28	98.0033	92.991	98.1255	90.057	83.1296
Ortalama	97.1221	91.9348	94.7503	89.4152	78.4528

4. TARTIŞMA

4.1. EEG ve EMG Verilerinin Karşılaştırılması

Tablo 6’da verilen özniteliklerin kullanıldığı birinci uygulamada EEG ve EMG verileri ayrı olarak sınıflandırılmıştır. Tablo 7’de verilen EEG veri seti ile Tablo 9’da verilen EMG verisine ait sınıflandırma sonuçları incelendiğinde her iki sinyal türü için de J48 algoritmasının en yüksek doğruluk oranları (sırasıyla %67.3 ve %74.1) verdiği görülmektedir. Her iki veri grubunda da en düşük doğruluk oranını %11 ile Naif Bayes algoritması vermektedir. Ayrıca EMG verilerinin sınıflandırılmasında elde edilen sayısal değerlerin EEG verisine göre daha yüksek olduğu görülmektedir.

EEG ve EMG veri setleri birleştirildiğinde başarı oranı tüm sınıflandırıcılarda artmıştır. KNN yönteminde tek başına %58.7’lik doğruluk sağlayan EEG ile %66.5 doğruluk sağlayan EMG veri setleri birlikte sınıflandırıldığında, bu başarı %94.7’ye çıkmıştır. Çalışmada elde edilen bir diğer bulgu ise şudur; tek tip biyomedikal sinyallerden (EEG veya EMG) elde edilen veri kümelerinde en yüksek doğruluğu J48 algoritması veriyorken, iki veri tipi birleştiğinde, bu sefer KNN yönteminden daha yüksek skor elde edilmektedir. Bunun sebebi boyut sayısının artmasıyla birlikte gürültünün KNN üzerindeki etkisinin azalması olabilir. [46]. Sonuç olarak, Larraz’ın çalışmasında [9] da belirtildiği üzere EEG ve EMG öznitelikleri ile oluşturulmuş karma bir BBA sisteminin daha yüksek başarı performansı sağladığı söylenebilir.

Tablo 8’de verilen sonuçlara göre, EEG alt bileşenlerinin, ham EEG verisinden elde edilen başarıyı yakalayamadığı görülmektedir. Ayrıca alt bantların frekansları azaldıkça doğruluk oranlarının da giderek azaldığı görülmüştür. Dalgacık dönüşümü sonuçlarının sınıflandırma üzerindeki etkisini tek kanal üzerinden görmek yeterli değildir. Beynin hangi aktivitelerde hangi bölgeleri daha çok kullandığını izlemek için deney daha fazla elektrot ile tekrarlanabilir.

4.2. Kullanılan Özniteliklerin Değerlendirilmesi

Sınıflandırmayı etkileyen bir başka konu öznitelik seçimidir. Literatürde yaygın olarak kullanılmış olan en küçük, en büyük, ortalama, standart sapma gibi özniteliklerin Tablo 12’de verilen sınıflandırma sonuçları incelendiğinde kullanılan özniteliklerin ilk uygulamaya göre daha düşük doğruluk sağladığı görülmüştür. Bu sonuca göre, KNN yönteminden elde edilen doğruluk oranı %74.5 tir. KNN dışındaki diğer sınıflandırıcılar oldukça yetersiz bir performans göstermişlerdir.

4.3. Sınıflandırıcı Performanslarının Karşılaştırılması

Tablo 10’da verilen sonuçlar incelendiğinde en yüksek başarı %94.7 ile KNN için sağlanmıştır. Bu sonucun elde edilmesinde pencereleme türü ve süresi oldukça önemlidir. KNN yönteminde pencereleme türü bitişik olarak değiştirildiğinde başarının düştüğü gözlemlenmiştir. Nurwulan ve Jiang çalışmasında [46] olduğu gibi üst üste pencereleme yönteminde, örtüşme süresi arttıkça başarı artmaktadır. Ayrıca örnek sayısının artması yüzünden sınıflandırma süresi de artmaktadır. Ancak Nurwulan ve Jiang’ın çalışmalarının aksine pencere süresinin daha fazla kısaltılması başarı oranını düşürmektedir. KNN yönteminde en yüksek başarı oranları $k=1$ olarak alındığı durumda elde edilmiştir. J48 algoritması da bu veride yeterli bir doğruluk oranına sahiptir. DVM ve YSA yöntemlerinden elde edilen sonuçlar ise birbirine yakın (%68) elde edilmiştir.

Yapılan tüm sınıflandırmalarda doğruluk, hassaslık ve f skoru ölçütlerinin neredeyse aynı oldukları gözlemlenmiştir. Dolayısıyla değerlendirmeler çoğunlukla doğruluk üzerinden yapılmıştır.

4.4. Dengeli ve Dengesiz Verilerin Kıyaslanması

Deneyisel çalışmada ölçümü gerçekleştirilen her aktivitenin deneyimlenmesi farklı süreler gerektirdiğinden ötürü aktivitelerden elde edilen toplam örnek sayısı da her sınıf için farklı olmaktadır. Bu sebeple eğitim verisi dengesiz bir yapıda olmakta ve bu durum sınıflandırma sonuçlarını etkileyebilmektedir. Örneğin sınıflandırıcı algoritmalar sayıca daha baskın olan sınıfın hata oranını azaltma yoluna giderek doğruluğu artırmaya çalışırken azınlık sınıfları göz ardı edebilmektedir [47]. Dolayısıyla genel olarak yüksek bir doğruluk elde edilse dahi azınlık sınıflardaki hata oranı yüksek çıkabilmektedir.

KNN gibi bazı örnek bazlı algoritmalar sınıflar arası dengesizlikten daha az etkilenirken, Naif Bayes gibi sınıf dağılımına duyarlı algoritmalar ise bu durumdan daha çok etkilenmektedir. Şekil 24'te verilen KNN yöntemine ait karmaşıklık matrisine göre hem baskın hem de azınlık sınıflardan elde edilen doğruluk oranları gayet yüksek ve birbirine yakındır. Ortalama doğruluk ise %94.7 dir. Ancak Naif Bayes yönteminde elde edilen ortalama başarı %16.7 dir.

Bu çalışmada her sınıftan rastgele ve eşit sayıda örnekler seçmek suretiyle dengeli veri kümeleri sınıflandırılmıştır. EEG özniteliklerinin Tablo 13'te verilen sınıflandırma sonuçları incelendiğinde KNN, YSA ve J48 algoritmalarının doğruluk oranları Tablo 7 sonuçlarına göre %4-8 arasında azalma gösterirken DVM ile Naif Bayes sınıflandırıcılarının doğruluk oranı yükselmiştir. Tablo 14'te verilen EMG verilerinde de benzer oranlarda değişim söz konusudur. Ancak Tablo 15'e göre birleştirilmiş EEG ve EMG özniteliklerini incelediğimizde KNN sınıflandırıcısından elde edilen %94.7 lik başarının %83.1 e gerilediği görülmektedir. Ancak bu oran 11 sınıflı bir veriye göre oldukça yeterlidir. Benzer bir düşüş J48 için de geçerlidir. Dengelenmiş veri kümelerinde verinin büyük bir kısmının kaybı söz konusudur. Bu daha az bilgiyle öğrenme anlamına gelmesine rağmen genel başarı oranlarında büyük bir değişim gözlemlenmemiştir. Ayrıca daha dengeli bir dağılım elde edildiğinden dolayı DVM ve Naif Bayes algoritmalarında doğruluk oranlarının genel olarak yükseldiği görülmüştür.

4.5. Boyut Azaltmanın Sınıflandırmaya Etkisi

EEG özniteliklerinden oluşan veri setine TBA yöntemi uygulandığında DVM ve Naif Bayes sınıflandırıcılarının doğruluk oranlarının %14 arttığı gözlemlenmiştir. Diğer sınıflandırıcılarda ise gözle görülür bir düşüş söz konusudur. EMG özniteliklerinden oluşan veri setinde de benzer sonuçlar elde edilmiştir. Bu iki veri seti birleştirildiğinde ise DVM yönteminde %69 olan başarının %91.3'e çıktığı görülmüştür. Ayrıca DVM yönteminde en yüksek başarı oranları her zaman için radyal tabanlı çekirdek fonksiyonu ile elde edilmiştir. Bu çekirdek fonksiyonun biyomedikal sinyallerde daha başarılı sonuçlar vermesi literatür ile uyumludur [45].

4.6. Bireysel Bazda Yapılan Sınıflandırmanın Değerlendirilmesi

Bireysel sınıflandırma sonuçları incelendiğinde oldukça yüksek sonuçlar alındığı görülmektedir. Deneye katılan kişilerden 14 tanesi tüm aktiviteleri gerçekleştirirken, 4 kişi 10 aktiviteyi, 2 kişi ise 9 aktiviteyi gerçekleştirmiştir. Dolayısıyla bireysel bazlı sınıflandırmalar üç kategoride yapılmıştır. Tablo 19, Tablo 20 ve Tablo 21 de verilen doğruluk oranları birbirine yakın olup; KNN yönteminde %97.1-98.8, J48 karar ağacında %91.9-96.7, Destek Vektör Makineleri sınıflandırıcısında %94.7-98.6, Yapay Sinir Ağlarında %89.4-95.7, Naif Bayes yönteminde ise %78.4-85.3 arası doğruluk elde edilmiştir. KNN yönteminin hem genel sınıflandırmada hem de bireysel sınıflandırmada iyi sonuçlar verdiği gözlemlenmiştir. Naif Bayes sınıflandırıcısının genel sınıflandırmaya göre oldukça yüksek başarı elde ettiği görülmektedir. Genel sınıflandırma sonuçlarının, birey bazlı sınıflandırmaya göre daha düzensiz bir yapıda olması; EEG sinyallerinin kişiden kişiye değişkenliği ve aktiviteyi gerçekleştirenlerin uygulama sırasında odaklanma durumlarının farklılığı ile açıklanabilir [4].

5. SONUÇLAR

Bu tez çalışmasında EEG ve EMG kayıtlarından elde edilen öznitelikler ile 11 gündelik aktivitenin sınıflandırılması yapılmıştır. Eğitim verisinin hazırlanmasında MATLAB, sınıflandırılmasında ise Weka programı kullanılmıştır. EEG ve EMG kayıtlarına sınıflandırma aşamasından önce 50 Hz çentik filtre ile 0.5 Hz butterworth yüksek geçiren filtre uygulanmıştır. Daha sonra ham verilere üst üste pencereleme yöntemi uygulanmıştır. Pencere uzunluğu 2 saniye ve kaydırma miktarı 0.1 saniye olarak belirlenmiştir. EEG işaretlerinden alfa, beta, gama, delta ve teta bileşenlerinin elde edilebilmesi için 7 seviyeli dalgacık dönüşümü kullanılmıştır. Sonraki aşamada EEG, EMG ve alt bileşenlere ait öznitelikler çıkarılmıştır. Çalışmada kullanılan öznitelikler sırasıyla, ortalama mutlak değer, Hjorth tanımlayıcılarının değişkenlik parametresi, basıklık, Hurst fraktal indisi ve sıfır geçişleridir. Oluşturulan öznitelik vektörleri KNN, J48, DVM, YSA ve Naif Bayes algoritmalarıyla sınıflandırılmıştır.

En yüksek başarı %94.7 ile KNN yönteminde, EEG ve EMG verilerinin birlikte kullanıldığı veri setinde sağlanmıştır. J48 algoritması da %85.3 ile bu veride yeterli bir doğruluk oranına sahiptir. DVM ve YSA yöntemlerinden elde edilen sonuçlar ise birbirine yakın (%68) elde edilmiştir. Naif Bayes sınıflandırıcısı hazırlanan tüm veri setlerinde en düşük başarıyı göstermiştir. Ayrıca EEG alt bileşenlerinin sınıflandırma performanslarının yetersiz olduğu gözlemlenmiştir. Sınıflandırma performansını iyileştirmek ve gürültü oranını azaltmak amacıyla Temel Bileşenler Analizi (TBA) yöntemi kullanılmıştır. Boyut azaltma sayesinde DVM yönteminde %69 olan başarının %91.3 e çıktığı görülmüştür.

Elde edilen bulgulara göre DVM ve Naif Bayes algoritmalarının gürültülü ve yüksek boyutlu verilere karşı hassas olduğu söylenebilir. Ayrıca KNN algoritmasının dengesiz verilere karşı bağımsız olduğu ve yüksek boyutlu, büyük veri setleri için yavaş bir sınıflandırıcı olduğu elde edilen sonuçlardan bazılarıdır. Diğer çalışmalara kıyasla, sadece iki kanal elektrot ile kaydedilmiş 11 sınıflı bir veri kümesinin yüksek doğrulukla sınıflandırılması bu çalışmanın en önemli farkını oluşturmaktadır. Aynı zamanda EEG ile EMG verilerinin sınıflandırılmasında aynı öznitelik grubunun kullanılması ayrıca sınıflandırmanın sadece EEG verileri, sadece EMG verileri ile EEG ve EMG verileri olacak şekilde üç veri grubu üzerinden değerlendirilmesi de çalışmanın bir diğer önemli yanıdır. Çalışmada bahsi geçen göz kırpmak, kaş veya çeneyi hareket ettirmek gibi basit aktivitelerin

yüksek doğrulukla sınıflandırılabilir olması, bu çalışmanın BBA uygulamalarında kullanılabilirliğini göstermektedir. Ayrıca fiziksel ve zihinsel aktivitelerin sınıflandırılmasında EEG ve EMG verilerinin birlikte kullanılması, daha iyi sistemlerin geliştirilmesi için önemlidir.

Tez kapsamında elde edilen sonuçlar genel kullanıcılara hitap eden bir sınıflandırma modelinin yanında aynı zamanda bireysel bazlı modellerin de yüksek doğruluk oranları verdiğini göstermiştir. Bireysel bazlı modellerin daha kararlı ve yüksek sonuçlar verdiğini gözlemlenmiştir. Örneğin Naif Bayes algoritması hızlı çalışabilmesine karşın genel sınıflandırma açısından yeterli değildir. Ancak bireysel bazlı sınıflandırma performansının yüksek olması özellikle gerçek zamanlı uygulamalarda tercih edilmesini sağlayabilir.



6. ÖNERİLER

Verilerin pencereleme sınıflandırma performansını oldukça etkilemektedir. Ancak hangi pencere tipinin, süresinin veya kaydırma miktarının belirlenmesinde geçerli bir kural yoktur. Ancak gerçek zamanlı uygulamalar hedeflenmiyorsa pencere süresini geniş bir skalada değerlendirmek sınıflandırma performansını yükseltme konusunda avantaj sağlayabilir.

Öznitelik seçiminde veya çıkarılması gereken özniteliklerin belirlenmesinde; özniteliklerin birbiri ile olan korelasyonları incelenebilir. Eğer iki öznitelik arasında korelasyon yüksek ise veya saçılım diyagramı bir hat üzerinde yığılmış bir görünümde ise, bu iki özelliğin sınıflandırma performanslarının birbirine yakın olduğu düşünülebilir. Bu durumda sınıflandırma performansı en düşük olan öznitelik veri setinden çıkarılabilir.

Benzer şekilde iki özneliğin etiket vektörü arasındaki korelasyonu da çıkarılacak özneliği belirlemede yardımcı olabilir. Etiket vektörü ile arasında yüksek korelasyon olan öznitelikler veriyi daha iyi temsil edebilmektedir.

Literatürde genellikle 10 katlı çapraz doğrulama kullanılmaktadır. Ancak yüksek boyutlu bir veri seti üzerinde çalışırken, sınıflandırma süresinin verimli kullanılması amacıyla 5 katlı çapraz doğrulama kullanılması tercih edilebilir.

KNN yöntemi dengesiz ve gürültülü verilere bağımsız olması sebebiyle ön sınıflandırmalarda tercih edilebilir.

Yapay sinir ağlarında hatanın bir noktada sabit kalması, iterasyon sayısının belirlenmesinde yardımcı olabilir. Böyle bir durumda sınıflandırıcı ne kadar süre geçerse geçsin hatayı daha fazla küçültememektedir.

TBA gibi boyut azaltma algoritmaları sınıflandırma performansını artırabilir. Aynı zamanda veri boyutunu da azaltır. Bu sebeple boyut azaltma yöntemleri sınıflandırma konusunda güçlü bir araç olmaktadır.

Sınıflandırma problemlerinde herhangi bir algoritmanın diğerine üstünlüğü söz konusu değildir. Sınıflandırma işlemi verinin yapısı ile doğrudan ilişkilidir. Dolayısıyla en yüksek başarıyı hangi algoritmanın sağlayacağı deneme yapmadan bilinemez.

7. KAYNAKLAR

1. Aydemir, Ö., ve Kayıkçıoğlu, T., EEG Tabanlı Beyin Bilgisayar Arayüzleri, XI. Akademik Bilişim Konferansı, Şubat 2009 Harran Üniversitesi, Şanlıurfa, Bildiriler Kitabı: 11-13.
2. Akbari, H., Khalighinejad, B., Herrero, J. L., Mehta, A. D., ve Mesgarani N., Towards Reconstructing Intelligible Speech from the Human Auditory Cortex, Sci Rep, 9,874 (2019) 1-12.
3. Koçyiğit, M., Güven, A., ve Furkan, Ç. A. M., Elektroensefalografi Sinyallerinden Zihinsel Aritmetik Tabanlı Görev Ayırımı Sınıflandırması, Electronic Letters on Science and Engineering, 15,3 (2019) 57-65.
4. Kutlu, Onay, F., Motor ve Motor Olmayan Hayali Aktivite EEG Sinyallerinin Metne Dönüştürülmesi, Doktora Tezi, K.T.Ü. Fen Bilimleri Enstitüsü, Trabzon, 2020.
5. Tosun, M., Erginli, M., Kasım, Ö., Uğraş, B., Tanrıverdi, Ş. ve Kavak, T., EEG Verileri Kullanılarak Fiziksel El Hareketleri ve Bu Hareketlerin Hayalinin Yapay Sinir Ağları ile Sınıflandırılması, Sakarya University Journal of Computer and Information Sciences, 1,2 (2018) 1-9.
6. Başçıl, M., EEG Üzerinde Ortaya Çıkan Beyin Dalgalarının Çene Hareketleri ile İlişkisi, Bozok Tıp Dergisi, 9,3 (2019) 45-49.
7. Aydın, E. A., EEG Sinyalleri Kullanılarak Zihinsel İş Yükü Seviyelerinin Sınıflandırılması, Politeknik Dergisi, 24,2 (2021) 681-689.
8. Çalışkan, A., EMG Sinyalleri İçin Hibrid Öznitelik Çıkarma Yöntemi Geliştirilmesi, Niğde Ömer Halisdemir Üniversitesi Mühendislik Bilimleri Dergisi, 8,2 (2019) 652-664.
9. Lóopez-Larraz, E., Birbaumer, N., ve Ramos-Murguialday, A., A Hybrid EEG-EMG BMI Improves the Detection of Movement Intention in Cortical Stroke Patients with Complete Hand Paralysis, 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Temmuz 2018, Honolulu, Havai, Bildiriler Kitabı: 2000-2003.
10. Xie, P., Chen, X., Ma, P., Li, X., ve Su, P., Identification Method of Human Movement Intention based on the Fusion Feature of EEG and EMG, Proceedings of the World Congress on Engineering, Temmuz 2013, Londra, Bildiriler Kitabı II: 1340-1344.
11. Erin, K., ve Boru, B., EMG Based Human Computer Interaction, Journal of Smart Systems Research, 1,1 (2020) 11-17.
12. Teplan, M., Fundamentals Of EEG Measurement, Measurement Science Review, 2,2 (2002) 1-11.

13. Sanei, S., ve Chambers, J.A., EEG Signal Processingi 1st edition, Wiley-Interscience, 2013.
14. Donchin, E., Spencer, K. M., ve Wijesinghe, R., The Mental Prosthesis: Assessing the Speed of a P300-Based Brain-Computer Interface, IEEE Transactions on Rehabilitation Engineering, 8,2 (2000) 174-179.
15. Siuly, S., Li, Y., ve Zhang, Y., EEG Signal Analysis and Classification: Techniques and Applications, Springer International Publishing, 2016.
16. Niedermeyer, E., ve Da Silva, F.H.L., Electroencephalography: Basic Principles, Clinical Applications and Related Fields, Lippincott Williams and Wilkins, 2005.
17. Sanei, S., ve Chambers, J. A., EEG Signal Processing, John Wiley & Sons, 2007.
18. [https://en.wikipedia.org/wiki/10%E2%80%9320_System_\(EEG\)](https://en.wikipedia.org/wiki/10%E2%80%9320_System_(EEG)) 10–20 System (EEG). 8 Mayıs 2021.
19. Farnsworth, B., iMotions_EEG_Guide__2019.pdf, <https://imotions.com/blog/eeg/> 8 Mayıs 2021.
20. Uygun, B. N., Sezer, E., ve Ünalır, M. O., EEG Sinyallerinin Analizi ve Sınıflandırılmasında Yazılım Sistemlerinin Kullanımına Genel Bir Bakış, 13. Ulusal Yazılım Mühendisliği Sempozyumu, Eylül 2019, İzmir, Bildiriler Kitabı: 1-14.
21. Stashuk, D., EMG Signal Decomposition: How Can It Be Accomplished and Used, Journal of Electromyography and Kinesiology, 11,3 (2001) 151–173.
22. Taşar, B., Emg Sinyallerinin Anlamlandırılması İçin Radyal Fonksiyonlu Yapay Sinir Ağı Tasarımı, International Journal of Engineering Research and Development, 10,1 (2018) 153-159.
23. Görür, A., Bruxism, EEG-EMG İlişkisi, Klinik Tıp Bilimleri, 5,3 (2017) 29-34.
24. http://megep.meb.gov.tr/mte_program_modul/moduller_pdf/Kas-sinir%20Sinyal%20%C4%B0zleyicilerde%20Kurulum.pdf Kas Sinir Sinyal İzleyicilerde Kurulum. 9 Mayıs 2021.
25. Yaman M.O., Fiziyojji. https://cdn-acikogretim.istanbul.edu.tr/auzefcontent/20_21_Bahar/fizyoloji/index.html 9 Mayıs 2021.
26. Altın, C., Dört Rotorlu İnsansız Hava Aracının Görsel Geri Beslemeli Birleştirilmiş EEG ve EMG Biyolojik Sinyalleri ile Kontrolü, Doktora Tezi, Bozok Üniversitesi Fen Bilimleri Enstitüsü, Yozgat, 2017.
27. Koz, M., İstirahat Membran Potansiyeli & Aksiyon Potansiyeli. https://acikders.ankara.edu.tr/pluginfile.php/8344/mod_resource/content/1/4.1.AksiyonPotansiyeli.pdf 9 Mayıs 2021.

28. https://en.wikipedia.org/wiki/Action_potential Action Potential. 9 Mayıs 2021.
29. Kaçoğlu, C., ve Kale, M., Elektriksel Kas Uyarımlarının Biyokimyasal, Fizyolojik ve Nöral Mekanizması, Spor metre Beden Eğitimi ve Spor Bilimleri Dergisi, 17,1 (2019) 1-19.
30. Yazgan, E. ve Korürek, M., Tıp Elektroniği, İstanbul Teknik Üniversitesi Elektrik-Elektronik Fakültesi Ofset Baskı Atölyesi, 1996.
31. Ayaz, F., EMG Sinyallerinin Sınıflandırılması, Yüksek Lisans Tezi, İnönü Üniversitesi Fen Bilimleri Enstitüsü, Malatya, 2018.
32. Friesen, G. M., Jannett, T. C., Jadallah, M. A., Yates, S. L., Quint, S. R., ve Nagle, H. T., A Comparison of the Noise Sensitivity of Nine QRS Detection Algorithms, IEEE Transactions on Biomedical Engineering, 37,1 (1990) 85-98.
33. https://en.wikipedia.org/wiki/Machine_learning Machine Learning. 10 Mayıs 2021.
34. Tekin, R., Kaya, Y., ve Tağluk, M. E., K-means ve YSA temelli Hibrit Bir Model ile Epileptik EEG İşaretlerinin Sınıflandırılması, Elektrik-Elektronik ve Bilgisayar Sempozyumu (FEEB 2011), Ekim 2011, Fırat Üniversitesi, Elazığ, Bildiriler Kitabı: 277-283.
35. Porr, B., Measuring EMG and EEG Simultaneously to Determine the EMG Pollution in the EEG During a Variety of Tasks. <http://researchdata.gla.ac.uk/676/> 18 Kasım 2020.
36. <https://vru.vibrationresearch.com/lesson/tables-of-window-function-details/> Table of Window Function Details. 4 Mayıs 2021.
37. Türkoğlu, M., EEG Sinyallerinin Analizinde Performansı Yüksek Olan Dalgacık Tipinin Belirlenmesi, Yüksek Lisans Tezi, TOOB E.T.Ü., Fen Bilimleri Enstitüsü, Ankara, 2010.
38. http://matlab.izmiran.ru/help/toolbox/wavelet/ch01_31a.html Daubechies. 4 Mayıs 2021.
39. Çerçi, Ç., EMG İşaretlerinin Özniteliklerinin Çıkarılması, KNN ve YSA Yöntemleri ile Sınıflandırılması, Yüksek Lisans Tezi, İ.T.Ü., Fen Bilimleri Enstitüsü, İstanbul, 2017.
40. https://en.wikipedia.org/wiki/Hjorth_parameters Hjorth Parameters. 6 Mayıs 2021.
41. Akıncı., T. Ç., Taşkın, S., Şeker, S. ve Kaymaz, H., Dalgacık Dönüşümü Kullanılarak Bir Yolcu Otobüsü Alternatörünün Sıcaklık Değişiminin İstatistiksel Analizi, Elektrik, Elektronik ve Bilgisayar Mühendisliği Sempozyumu ve Fuarı, Aralık 2006, Bursa, Bildiriler Kitabı: 187-191.

42. Khosrowabadi, R., ve bin Abdul Rahman, A. W., Classification of EEG Correlates on Emotion Using Features from Gaussian Mixtures of EEG Spectrogram, Proceeding of the 3rd International Conference on Information and Communication Technology for the Moslem World (ICT4M), Aralık 2010, Bildiriler Kitabı: 102-107.
43. Pritchard, W. S., The Brain in Fractal Time: 1/f-like Power Spectrum Scaling of the Human Electroencephalogram, International Journal of Neuroscience, 66, 1-2 (1992) 119-129.
44. <https://www.mathworks.com/help/wavelet/ref/wfbmesti.html> Parameter Estimation of Fractional Brownian Motion. 7 Mayıs 2021.
45. Ayhan, S. ve Erdoğan, Ş., Destek Vektör Makineleriyle Sınıflandırma Problemlerinin Çözümü İçin Çekirdek Fonksiyonu Seçimi, Eskişehir Osmangazi Üniversitesi İktisadi ve İdari Bilimler Dergisi, 9,1 (2014) 175-201.
46. Nurwulan, N., ve Jiang, B. C., Window Selection Impact in Human Activity Recognition, International Journal of Innovative Technology and Interdisciplinary Sciences, 3,1 (2020) 381-394.
47. Koçoğlu, F. Ö., ve Özcan, T., Dengeli-Dengesiz Veri Seti Dağılımının Aşırı Öğrenme Makinesi Yöntemi Performansına Etkisi, Mühendislik ve Teknoloji Yönetimi Zirvesi, 2018, Türkiye, Bildiriler Kitabı: 201-209.

8. EKLER

Ek 1. Eğitim Verisinin Hazırlanmasında Kullanılan MATLAB Programı

% FEATURE EXTRACTOR (©2021 Taner YURDUSEVER)

```
clc; clear; close all; tic;
currentFolder=pwd; % Programın bulunduğu dizin
data_path = '\dataset_676\experiment_data\';

% ===== Parametreler =====
% Sampling Rate (Hz)
fs = 1000;
% Veri Setine Eklenecek Aksiyonlar
chosen_acts={'blink','eyescrunching','flow','jaw','lie_relax',...
'raisingeyebrows','readinglieing','readingsitting','sudoku',...
'templerun','wordsearch'};
% Veri Setine Eklenecek Deneklerin Numaraları
subj_nums={'08','09','10','13','14','16','17','18','20','21','23','24',...
'26','28','02','03','04','05','06','22'};
% Feature Functions
featureFuncs={
    @(x)min(x)
    @(x)mean(abs(x))
    @(x)max(x)
    @(x)std(x)
    @(x)hjorth(x,'complexity')
    @(x)hjorth(x,'mobility')
    @(x)skewness(x)
    @(x)kurtosis(x)
    @(x)fractalDim(x,1)
    @(x)powerDen(x)
    @(x)spectDen(x)
    @(x)rms(x)
    @(x)sifir(x)
    @(x)dfu(x)
    @(x)median(x)
};

% Veri Yapısı
% Data Types: 'eeg' 'emg' 'd1' ..... 'd7' 'a1' ..... 'a7'
% NOT: Verinin frekans spektrumu 'f_veritipi' tanımlayıcısıyla elde edilir.
dy={
    % 1. min() uygulanacak veriler
    {}
    % 2. absmean() uygulanacak veriler
    {'eeg' 'emg'}
```

Ek 1 'in devamı

```

% 3. max() uygulanacak veriler
{}
% 4. std() uygulanacak veriler
{}
% 5. complexity() uygulanacak veriler
{}
% 6. mobility() uygulanacak veriler
{'eeg' 'emg'}
% 7. skewness() uygulanacak veriler
{}
% 8. kurtosis() uygulanacak veriler
{'eeg' 'emg'}
% 9. fractalDim(1) uygulanacak veriler
{'eeg' 'emg'}
% 10. powerDen() uygulanacak veriler
{}
% 11. spectDen() uygulanacak veriler
{}
% 12. rms() uygulanacak veriler
{}
% 13. sifir() (sifir geiř sayısı) uygulanacak veriler
{'eeg' 'emg'}
% 14. dfu() (dalga formu uzunluęu) uygulanacak veriler
{}
% 15. median() (dalga formu uzunluęu) uygulanacak veriler
{}
};
% Pencereleme
windowing=1;
wintype=2; % 1 = adjacent windowing, 2 = overlap windowing
r=0.1; % (s)
R=2.0; % (s)
normalize=1; % Eęitim verisi zelliklerine min-max normalizasyonu uygula
exportFileName='veri.txt';
saveAsArff=1; % Datayı ayrıca .arff formatında kaydet
% =====

if wintype==1
    winsize=abs(R-r)*fs;
    offset=r*fs;
else
    winsize=r*fs;
    offset=abs(R-r)*fs;
end

dataNames={'emgeeg.dat','artefact.dat','dataok.dat',...
'zero_time_data.dat','zero_time_video.dat'}; % İřlenecek veri dosyaları

```

Ek 1 'in devamı

```

subj_nums=unique(subj_nums);
chosen_acts=unique(chosen_acts);

% Eğitim Verisi Boyut Sayısı
dimNum=0;
for i=1:length(dy)
    dimNum=dimNum+length(dy{i});
end

allFileNums=length(chosen_acts)*length(subj_nums);

% Etiket Vektörü
labels=1:length(chosen_acts); % Aksiyonların sırası etikettir.

levelNums=[1 2];
sampleNum=0;
sampleRow=1;
sampleSum=0;
LabelTimes=zeros(1,length(labels));
sayac1=0; sayac2=0; sayac3=0; process=0;
gecis1=0; gecis2=0;
for level=levelNums % level 1: Örnek sayısı tespiti içindir.
    % level 2: Eğitim verisi oluşturulması içindir.
    for i=1:length(chosen_acts)
        s=1;
        for j=1:length(subj_nums)
            cd(data_path);
            folder1=['subj' subj_nums{j}];
            folder2=chosen_acts{i};
            if isfolder(folder1)
                hedef=[data_path folder1 '\'];
                cd(hedef);
            if isfolder(folder2)
                hedef=[data_path folder1 '\' folder2 '\'];
                cd(hedef);

            % Veri Dosyalarının Kontrolü
            if isfileNumOk(dataNames)
                % Files exists
                if level==1
                    load('artefact.dat');
                    sampleNum=sampleNum+size(artefact,1);
                    LabelTimes(i)=LabelTimes(i)+...
                        sum(artefact(:,2)-artefact(:,1));
                elseif level==2
                    % =====
                    % Eğitim Verisini Oluştur

```

Ek 1 'in devamı

```

if sampleRow==1
    Data=zeros(sampleNum,dimNum+1);
end

load('emgeeg.dat');

% Bazı Frekanslar Bastırılıyor.
emgeeg=filteredData(emgeeg,fs);

load('artefact.dat');
load('zero_time_video.dat');
load('zero_time_data.dat');

fark=zero_time_video-zero_time_data;

for smpl=1
    if smpl==1 % Aktivite verilerinden özellik vektörleri eldesi
        time=artefact-fark;
        etk=labels(i); % Etiket
    end
    pos=round(time*fs);
    pos(pos<=0)=1;
    pos(pos>length(emgeeg(:,1)))=length(emgeeg(:,1));
    for p1=1:size(pos,1)
        pack=emgeeg(pos(p1,1):pos(p1,2),:);
        sampleSum=sampleSum+size(pack,1); % İşlenen toplam örnek sayısı
        % ===== Pencereleme =====
        if windowing==1
            edgebins=1:winSize:length(pack(:,1));
            if length(edgebins)==1
                edgebins=[1 length(pack(:,1))];
            end
            for edge=1:length(edgebins)-1
                eStart=round(edgebins(edge));
                eEnd=round(edgebins(edge+1)+offset);
                if eEnd>length(pack(:,1))
                    eEnd=length(pack(:,1));
                end
                mini_pack=pack(eStart:eEnd,:);
                % Hamming Window
                hamming_win=hamming(length(mini_pack),'symmetric');
                mini_pack=mini_pack.*hamming_win;
                [sample,~]=makeSample(mini_pack,dy,featureFuncs);
                Data(sampleRow,:)= [sample etk]; % Eğitim Veri Seti
                sampleRow=sampleRow+1;
            end
        end
    end
    % =====

```

Ek 1 'in devamı

```

else
    [sample,~]=makeSample(pack,dy,featureFuncs);
    Data(sampleRow,:)= [sample etk]; % Eğitim Veri Seti
    sampleRow=sampleRow+1;
end
end
end
% =====
end
else
    % Folder1 and Folder2 exists but missing File(s)
    if level==1
        if exist('loss','var')==0
            loss=0;
        end
        fileNum=0;
        for lst=1:length(dataNames)
            if isfile(dataNames{lst})
                fileNum=fileNum+1;
            end
        end
        loss=loss+length(dataNames)-fileNum;
    elseif level==2
        if sayac1==0
            emptyFiles=cell(loss,3);
        end
        for lst=1:length(dataNames)
            if isfile(dataNames{lst})==0
                sayac1=sayac1+1;
                emptyFiles{sayac1,1}=folder1;
                emptyFiles{sayac1,2}=chosen_acts{i};
                emptyFiles{sayac1,3}=dataNames{lst};
            end
        end
    end
end

end
else
    % Folder1 exists but Folder2 (act) does not exist
    if gecis1==1
        emptyFolder2=cell(sayac2,2);
        sayac2=0; gecis1=0;
    end
    sayac2=sayac2+1;
    if level==2
        emptyFolder2{sayac2,1}=folder1;
        emptyFolder2{sayac2,2}=chosen_acts{i};
    end
end

```


Ek 1 'in devamı

```

        end

        end
    else
        % Folder1 (subj) does not exist
        if gecis2==1
            emptyFolder1=cell(sayac3,1);
            sayac3=0; gecis2=0;
        end
        sayac3=sayac3+1;
        if level==2
            emptyFolder1 {sayac3}=folder1;
        end
    end

    end
    if level==1
        clc; disp('Örnekler İşleniyor!');
    elseif level==2
        process=process+1;
        process_yuzde=(process/allFileNums)*100;
        clc;
        disp(['İlerleme (%) : ' sprintf('%1.1f', process_yuzde)]);
        yuzde(process_yuzde,50); disp(' ');
    end
end

end
if level==1
    gecis1=1; gecis2=1;
end
end

% Örneklerin Sırası Düzenleniyor
if sampleRow>0
    Data=unique(Data,'rows'); % Tekrarlayan örnekleri çıkar.
else
    Data=[];
end

% Sınıfların Histogram Grafiği
if sampleRow>0
    figure;
    labels=unique(Data(:,end));
    classNums=zeros(1,length(labels));
    for i=1:length(labels)
        classNums(i)=sum(Data(:,end)==labels(i));
    end
end

```

Ek 1 'in devamı

```

bar(labels,classNums)
xlabel('Sınıflar'); ylabel('Örnek Sayısı');
title('Sınıfların Dağılımı')
grid on;
box on;
end

% SONUÇLAR

disp(' '); disp('<strong> RAPOR:</strong>');

disp(' '); disp(['<strong> num2str(size(Data,1)) ...
'</strong>adet örnek elde edildi!']);

if sayac3>0
emptyFolder1=unique(emptyFolder1);
b=[num2str(length(emptyFolder1)) ' adet eksik denek tespit edildi!'];
disp(' '); disp(['<strong> b '</strong>'])
for i=1:length(emptyFolder1)
disp(' '); disp([emptyFolder1{i} ' eksik olarak tespit edildi.']);
end
end

if sayac2>0
b=[num2str(sayac2) ' adet eksik aksiyon tespit edildi!'];
disp(' '); disp(['<strong> b '</strong>'])
for i=1:sayac2
disp(' '); disp([emptyFolder2{i,1} ' => ' emptyFolder2{i,2}...
' eksik olarak tespit edildi.']);
end
end

if sayac1>0
b=[num2str(sayac1) ' adet eksik dosya tespit edildi!'];
disp(' '); disp(['<strong> b '</strong>'])
emptyFiles=emptyFiles(1:sayac1,:);
for i=1:sayac1
disp(' '); disp([emptyFiles{i,1} ' => ' emptyFiles{i,2}...
' => ' emptyFiles{i,3} ' eksik olarak tespit edildi.']);
end
end

cd(currentFolder);

% Min-Max Normalizasyonu Uygula
if normalize==1
min_vec=min(Data(:,1:end-1));

```

Ek 1 'in devamı

```

    max_vec=max(Data(:,1:end-1));
    den=max_vec-min_vec;
    den(den==0)=1;
    Data(:,1:end-1)=(Data(:,1:end-1)-min_vec)./den;
end

% Eğitim Verisini Kaydet
save(exportFileName,'Data','-ascii','-tabs')

% Eğitim Verisini .arff Olarak Kaydet
if saveAsArff==1
    % Özellik İsimleri
    colNames=cell(1,length(Data(1,:)));
    ind=1;
    for i=1:length(featureFuncs)
        if ~isempty(dy{i})
            for j=1:length(dy{i})
                veri_tipi=dy{i}{j};
                val=strrep(char(featureFuncs{i}),'@(x)',");
                val=strrep(val,char("''"),");
                colNames{ind}=strrep(val,'(x',[(' veri_tipi)];
                ind=ind+1;
            end
        end
    end
    colNames{end}='class';
    % Etiket Vektörü
    classNames=chosen_acts;
    % Kaydet
    arffSaver(exportFileName(1:end-4),Data,colNames,classNames)
end

disp(' '); disp('<strong>TÜM İŞLEMLER TAMAMLANDI!</strong>');
disp(' '); disp(['Geçen Süre: ' num2str(toc) ' sn']); disp(' ');

% PROGRAM FONKSİYONLARI

% Tüm dosyaların kontrolünü yapan fonksiyon
function sonuc=isfileNumOk(dataNames)
p=0;
for i=1:length(dataNames)
    if isfile(dataNames{i})
        p=p+1;
    end
end
if p==length(dataNames)
    sonuc=true;
end

```

Ek 1 'in devamı

```

else
    sonuc=false;
end
end

% EEG ve EMG verilerinde bazı frekansları filtreleyen fonksiyon.
% eegemg: emgeeg.dat dosyasına kaydedilmiş veridir.
% eegemg = [time_vector eeg(fp1) emg(chin) trigger]
function sonuc = filteredData(eegemg,fs)
% Scaling by the gain of the pre-amp
eeg = eegemg(:,2)/500; % Amp. Gain = 500
emg = eegemg(:,3)/500; % Amp. Gain = 500
trigger = eegemg(:,4);
% Filters
% High Pass 0.5 Hz
[bhp,ahp] = butter(4,0.5/fs*2,'high');
eeg = filter(bhp,ahp,eeg);
emg = filter(bhp,ahp,emg);
% Filter 50 Hz
[bfilt50hz,afilt50hz] = butter(2,[49/fs*2 51/fs*2],'stop');
eeg = filter(bfilt50hz,afilt50hz,eeg);
emg = filter(bfilt50hz,afilt50hz,emg);
% Filter 80 Hz
[bfilt80hz,afilt80hz] = butter(2,[78/fs*2 82/fs*2],'stop');
eeg = filter(bfilt80hz,afilt80hz,eeg);
emg = filter(bfilt80hz,afilt80hz,emg);
% Filter 100 Hz
[bfilt100hz,afilt100hz] = butter(2,[95/fs*2 105/fs*2],'stop');
eeg = filter(bfilt100hz,afilt100hz,eeg);
emg = filter(bfilt100hz,afilt100hz,emg);
sonuc=[eegemg(:,1) eeg emg trigger/5000];
end

% Bu fonksiyon bir eeg sinyalinin temel bileşenlerini elde eder.
% Kullanım:
% type = {'D1','D2','D3','D4','D5','A5'};
% a = eegComp(eeg,type{ 1 })
function sonuc = eegComp(eeg,type)
waveletFunction = 'db4';
[C,L] = wavedec(eeg,7,waveletFunction);
% Sinyal Tipi
stype = char(lower(string(type)));
% D or A
st1 = stype(1);
% Level
st2 = str2double(stype(2));
sonuc = wrcoef(st1,C,L,waveletFunction,st2);

```

Ek 1 'in devamı

```
end
```

```
% Verinin frekans spektrumu bileşenlerini elde eden fonksiyon.
```

```
% f: Frekans bileşenlerini içeren vektör (Hz)
```

```
% G: Frekans bileşenlerine karşılık gelen genlik değerleri
```

```
function [f,G] = fftData(data,fs)
```

```
L = length(data); % Length of signal
```

```
Y = fft(data);
```

```
P = abs(Y/L);
```

```
G = P(1:round(L/2)+1);
```

```
G(2:end-1) = 2*G(2:end-1);
```

```
f = fs*(0:(round(L/2)))/L;
```

```
end
```

```
% Bu fonksiyon kaynak veriden, dy parametresinde belirtilen bileşenlerini
```

```
% veya dönüşümlerini üretir daha sonra elde edilen vektör featureFuncs
```

```
% içine kaydedilmiş fonksiyonlara göre skaler bir sayıya eşlenir.
```

```
% Bu sayılar kaydedilerek kaynak verinin özellik vektörü oluşturulur.
```

```
function [Samples,Signals] = makeSample(eegemg,dy,featureFuncs)
```

```
num=0;
```

```
for i=1:length(dy)
```

```
    num=num+length(dy{i});
```

```
end
```

```
Samples=zeros(1,num);
```

```
Signals=cell(1,num);
```

```
sayac=1;
```

```
spectrum=0;
```

```
for i=1:length(featureFuncs)
```

```
    func=featureFuncs{i};
```

```
    if ~isempty(dy{i})
```

```
        for j=1:length(dy{i})
```

```
            veri_tipi=dy{i}{j};
```

```
            % Veri Domeni Tespit Ediliyor
```

```
            if string(veri_tipi(1:2))=="f_" % Fourier dönüşümü uygula
```

```
                spectrum=1;
```

```
                veri_tipi=veri_tipi(3:end);
```

```
            end
```

```
            % Veri Tanımlanıyor
```

```
            if string(veri_tipi)=="eeg"
```

```
                veri=eegemg(:,2);
```

```
            elseif string(veri_tipi)=="emg"
```

```
                veri=eegemg(:,3);
```

```
            elseif string(veri_tipi)=="xcor"
```

```
                veri=xcorr(eegemg(:,2),eegemg(:,3));
```

```
            else
```

```
                veri=eegComp(eegemg(:,2),veri_tipi); % Dalgacık dönüşümü
```

```
            end
```

Ek 1 'in devamı

```

    % Frekans Domeni İçin Dönüşüm
    if spectrum==1
        fs=1/(eegemg(2,1)-eegemg(1,1));
        [~,G]=fftData(veri,fs);
        veri=G;
    end
    % Veriden Öznitelikler Elde Ediliyor
    Samples(sayac)=func(veri);
    Signals{sayac}=veri;
    sayac=sayac+1;
end
end
end

% x yüzdesine göre ilerleme çubuğunu ekrana yazdır.
% step, ilerleme çubuğunun karakter uzunluğu
function yuzde(x,step)
d=floor(x/(100/step));
e=step-d;
fprintf('#')
for i=1:d
    fprintf('=');
end
for i=1:e
    fprintf('-');
end
fprintf('#')
end

% Matris tipi verileri .arff formatında kaydeden fonksiyon
function arffSaver(name,data,colNames,classNames)
% Dosyanın Oluşturulması
name=char(name);
if string(name(end-4:end))~=".arff"
    name=[name '.arff'];
end
fid=fopen(name,'wt');
line1=['%\n%% Date: ' date '\n%\n\n'];
fprintf(fid,line1);
line2=['@RELATION ' char(name(1:end-5)) '\n\n'];
fprintf(fid,line2);
C=string.empty(0,length(classNames));
for i=1:length(classNames)
    if i<length(classNames)
        C(i)=string([char(" ") classNames{i} char(",")]);
    else

```

Ek 1 'in devamı

```

        C(i)=string([char('') classNames{i} char('')]);
    end
end
C=['{' char(strjoin(C)) '}'];
for i=1:length(colNames)
    if i<length(colNames)
        line3=['@ATTRIBUTE ' char('') colNames{i} char('') ' REAL\n\n'];
        fprintf(fid,line3);
    else
        line3=['@ATTRIBUTE ' char('') colNames{i} char('') C '\n\n'];
        fprintf(fid,line3);
    end
end
end
line4='@DATA\n';
fprintf(fid,line4);
% Veri Yazım Formatının Oluşturulması
num=size(data,2);
format="; % Verilerin yazım şekli
ek=['%f ','];
for i=1:num-1
    format(end+1:end+length(ek))=ek;
end
% Verilerin Yazılması
for i=1:size(data,1)
    fprintf(fid,[format char('"%s\n")'],data(i,1:end-1),...
        classNames{data(i,end)});
end
fclose(fid);
end

```

% ÖZİNİTELİK FONKSİYONLARI

```

% Bu fonksiyon giriş sinyalinden hjorth parametrelerini hesaplar. Activity,
% frekans bölgesine ait güç spektrumunun yüzeyini belirler. Sinyalin yüksek
% frekans bileşenleri çok sayıda ise activity büyük olarak, az sayıda ise
% bu değer küçük olarak elde edilmektedir Mobility, güç spektrumun standart
% sapmasını vererek, sinyalin enerjisini temsil eder. Complexity, sinyalin
% saf sinüs dalgası ile benzerliğini hesaplamak için kullanılmaktadır.
% Karmaşıklık değeri 1 değerine yaklaştıkça, sinyal o kadar saf sinüs
% dalgasına benzemektedir.
% Kullanım:
% type = {'activity','mobility','complexity'}
% a = hjorth(eeg,type{1})
function sonuc = hjorth(data,parType)
% Hjorth Activity
act=@(x) var(x);
% Hjorth Mobility

```

Ek 1 'in devamı

```

mob=@(x) sqrt(act(diff([0;x]))/act(x));
% Hjorth Complexity
comp=@(x) mob(diff([0;x])/mob(x));
if string(parType)=="activity"
    sonuc=act(data);
elseif string(parType)=="mobility"
    sonuc=mob(data);
else
    sonuc=comp(data);
end
end

```

```

% Sinyalin fraktal boyut indisini hesaplayan fonksiyon
function sonuc=fractalDim(data,num)
w=wfbmesti(data);
sonuc=w(num);
end

```

```

% Sinyalin güç yoğunluğunu hesaplayan fonksiyon
function p=powerDen(data)
n=length(data);
e=sum(abs(data).^2);
p=e/(2*n+1);
end

```

```

% Sinyalin rms değerini hesaplayan fonksiyon
function r=rms(data)
n=length(data);
e=sum(data.^2);
r=sqrt((1/n)*e);
end

```

```

% Sinyalin dalga formu uzunluğunu hesaplayan fonksiyon
function d=dfu(data)
top=0;
for i=2:length(data)
top=top+data(i)-data(i-1);
end
d=top;
end

```

```

% Sinyalin sıfır geçiş miktarını hesaplayan fonksiyon
function s=sifir(data)
esik=0;
top=0;
for i=1:length(data)-1
    if sign(data(i)*data(i+1))==-1 && abs(data(i+1))>esik

```


Ek 1 'in devamı

```
        top=top+1;
    end
end
s=top;
end
```

% Sinyalin spektral güç yoğunluğunu hesaplayan fonksiyon

```
function sd=spectDen(data)
```

```
P=fft(data);
```

```
n=length(data);
```

```
P=abs(P/n);
```

```
sd=sum(abs(P).^2);
```

```
end
```

ÖZGEÇMİŞ

Keşap Fen Lisesi'nden mezun oldu. 2018 yılında Süleyman Demirel Üniversitesi Elektronik ve Haberleşme Mühendisliği Bölümü'nden "Derin Öğrenme Tabanlı Video İndeksleme" konulu bitirme ödevi ile mezun oldu. 2019 yılında Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü Elektrik-Elektronik Mühendisliği Anabilim Dalı'nda lisansüstü öğrenimine başladı. Yabancı dil olarak İngilizce bilmektedir.

