T.C.

YILDIZ TEKNİK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

NÖRODEJENERATİF HASTALIKLARIN ENTROPİ TABANLI ÖZNİTELİKLER KULLANILARAK MAKİNE ÖĞRENMESİ YÖNTEMLERİYLE SINIFLANDIRILMASI

Dilara ARAPOĞLU

YÜKSEK LİSANS TEZİ

İstatistik Anabilim Dalı

İstatistik Programı

Danışman

Prof. Dr. Ersoy ÖZ

Mart, 2023

T.C.

YILDIZ TEKNİK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

NÖRODEJENERATİF HASTALIKLARIN ENTROPİ TABANLI ÖZNİTELİKLER KULLANILARAK MAKİNE ÖĞRENMESİ YÖNTEMLERİYLE SINIFLANDIRILMASI

Dilara ARAPOĞLU tarafından hazırlanan tez çalışması 02.03.2023 tarihinde aşağıdaki jüri tarafından Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü İstatistik Anabilim Dalı, İstatistik Programı **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Prof. Dr. Ersoy ÖZ
Yıldız Teknik Üniversitesi
Danışman

Jüri Üyeleri

Prof. Dr. Ersoy ÖZ, Danışman
Yıldız Teknik Üniversitesi _____

Doç. Dr. Öyküm Esra YİĞİT, Üye
Yıldız Teknik Üniversitesi _____

Prof. Dr. Semra ERPOLAT TAŞABAT, Üye
Mimar Sinan Güzel Sanatlar Üniversitesi _____

Danışmanım Prof. Dr. Ersoy ÖZ sorumluluğunda tarafımca hazırlanan Nörodejeneratif Hastalıkların Entropi Tabanlı Öznitelikler Kullanılarak Makine Öğrenmesi Yöntemleriyle Sınıflandırılması başlıklı çalışmada veri toplama ve veri kullanımında gerekli yasal izinleri aldığımı, diğer kaynaklardan aldığım bilgileri ana metin ve referanslarda eksiksiz gösterdiğimi, araştırma verilerine ve sonuçlarına ilişkin çarpıtma ve/veya sahtecilik yapmadığımı, çalışmam süresince bilimsel araştırma ve etik ilkelerine uygun davrandığımı beyan ederim. Beyanımın aksinin ispatı halinde her türlü yasal sonucu kabul ederim.

Dilara ARAPOĞLU

İmza

Eymen ve biricik ailem,

yanımda varlığını daimi hissettiğim arkadaşlarıma

TEŞEKKÜR

Çalışma süreci boyunca beni destekleyen başta danışmanım Prof. Dr. Ersoy ÖZ'e, bu süreçte beni yalnız bırakmayıp anlayışı ile destek olan ailem ve arkadaşlarıma teşekkürlerimi sunarım.

Dilara ARAPOĞLU

İÇİNDEKİLER

SİMGE LİSTESİ vii
KISALTMA LİSTESİ viii
ŞEKİL LİSTESİ ix
TABLO LİSTESİ x
ÖZET xii
ABSTRACT xiv
1 GİRİŞ 1
1.1 Makine Öğrenmesi 1
1.2 Nörodejeneratif Hastalıklar 1
1.3 Literatür Özeti 2
1.4 Tezin Amacı 6
1.5 Hipotez 7
2 NÖRODEJENERATİF HASTALIKLAR VE YÜRÜYÜŞ ANALİZİ 8
2.1 Nörodejeneratif Hastalıklar 8
2.1.1 Parkinson Hastalığı 8
2.1.2 Amyotrofik Lateral Skleroz (ALS) Hastalığı 9
2.1.3 Huntington Hastalığı 12
2.2 Yürüyüş Analizi 13
3 MATERYAL VE METOTLAR 15
3.1 Makine Öğrenmesi 15
3.1.1 Destek Vektör Makineleri 15
3.1.2 K En Yakın Komşu 20
3.1.3 Lojistik Regresyon 24
3.1.4 Rastgele Orman 26
3.1.5 Artırma Algoritmaları 28
3.1.6 Gradyan Artırma Algoritmaları 30
3.1.7 XGBOOST 30
3.1.8 LightGBM 32
3.1.9 Saf Bayes Sınıflandırıcısı 33
3.2 Entropi Tabanlı Sistemler 37
3.2.1 Bilgi Teorisi 37

3.2.2 Permütasyon Entropisi 39
3.2.3 Yaklaşık Entropi 41
3.2.4 Örnek Entropi 44
3.2.5 Bulanık Entropi 46
4 UYGULAMA **50**
4.1 Veri Seti 50
4.2 Çalışmada Kullanılan Öznitelikler 52
4.3 Sonuçlar 53
5 SONUÇ **68**
KAYNAKÇA **70**
TEZDEN ÜRETİLMİŞ YAYINLAR **79**

SİMGE LİSTESİ

s	Benzerlik Ölçüsü
K	Çekirdek Fonksiyonu
y_i	Etiket
m	Gömülü Boyut
α	Hassas Düzeltme Faktörü
H	Hilbert Uzayı
L_d	Lagranj Çarpımı
L_p	Lagranj Fonksiyonu
μ_i	Lagranj Parametresi
$d(i)$	Serilerin Farkı
b	Skaler
d	Uzaklık
τ	Zaman Gecikmesi

KISALTMA LİSTESİ

AH	Alzheimer Hastalığı
ALS	Amyotrofik Lateral Skleroz
CD	Chebyshev Uzaklığı
CORD	Korelasyon Uzaklığı
COSD	Kosinüs Uzaklığı
DVM	Destek Vektör Makineleri
ECAS	Edinburgh Bilişsel ve Davranışsal ALS Taraması
ED	Öklid Uzaklığı
EFB	Exclusive Feature Bundling
EURALS	Avrupa ALS Epidemiyoloji Konsorsiyumu
GBM	Gradyan Artırma Makinesi
GOOS	Gradient-based One-Side Sampling
H&Y	Hoehn ve Yahr
HH	Huntington Hastalığı
KKT	Karush-Kuhn-Tucker
KLE	Kullback Leibler Entropi
KNN	K-En Yakın Komşu
MAP	Maksimum A Posteriori
MD	Manhattan Uzaklığı
MS	Multipl Skleroz
ÖR	Örnek Entropi
PEAD	Peearson Uzaklığı
PH	Parkinson Hastalığı
UPDRS	Evrensel Parkinson Hastalığı Derecelendirme Ölçeği
WHO	Dünya Sağlık Örgütü
YE	Yaklaşık Entropi

ŞEKİL LİSTESİ

Şekil 3.1 Destek Vektör Makineleri'nde lineer olarak ayrılabilen durum için optimal ayrılan hiperdüzlem 16

Şekil 3.2 Destek Vektör Makineleri'nde lineer olarak ayrılamayan durum için optimal ayrılan hiperdüzlem 18

Şekil 4.1 Yürüyüş döngüsü 51

TABLO LİSTESİ

Tablo 1.1 Yürüyüş analizi yapan bazı makalelerin özeti 5

Tablo 2.1 Nörodejeneratif hastalıklar ve yürüyüş özellikleri 14

Tablo 3.1 γ, r, τ ve d çekirdek parametreleri ile literatürde en çok kullanılan çekirdek fonksiyonları 20

Tablo 4.2 Veri kümesinde bu çalışmada kullanılacak değişkenler 51

Tablo 4.3 Özniteliklere ait veri kümesi 52

Tablo 4.4 Tek öznitelik kümesi kullanılarak 30% test 70% eğitim ile elde edilen sınıflandırma sonuçları 54

Tablo 4.5 Tek öznitelik kümesi kullanılarak 20% test 80% eğitim ile elde edilen sınıflandırma sonuçları 54

Tablo 4.6 Tek öznitelik kümesi kullanılarak 25% test 75% eğitim ile elde edilen sınıflandırma sonuçları 55

Tablo 4.7 Tek öznitelik kümesi kullanılarak 15% test 85% eğitim ile elde edilen sınıflandırma sonuçları 55

Tablo 4.8 Tek öznitelik kümesi kullanılarak 10% test 90% eğitim ile elde edilen sınıflandırma sonuçları 56

Tablo 4.9 İkili öznitelik kümesi kullanılarak 30% test 70% eğitim ile elde edilen sınıflandırma sonuçları 56

Tablo 4.10 İkili öznitelik kümesi kullanılarak 20% test 80% eğitim ile elde edilen sınıflandırma sonuçları 57

Tablo 4.11 İkili öznitelik kümesi kullanılarak 25% test 75% eğitim ile elde edilen sınıflandırma sonuçları 58

Tablo 4.12 İkili öznitelik kümesi kullanılarak 15% test 85% eğitim ile elde edilen sınıflandırma sonuçları 59

Tablo 4.13 İkili öznitelik kümesi kullanılarak 10% test 90% eğitim ile elde edilen sınıflandırma sonuçları 60

Tablo 4.14 Üçlü öznitelik kümesi kullanılarak 30% test 70% eğitim ile elde edilen sınıflandırma sonuçları.....61

Tablo 4.15 Üçlü öznitelik kümesi kullanılarak 20% test 80% eğitim ile elde edilen sınıflandırma sonuçları.....62

Tablo 4.16 Üçlü öznitelik kümesi kullanılarak 25% test 75% eğitim ile elde edilen sınıflandırma sonuçları.....63

Tablo 4.17 Üçlü öznitelik kümesi kullanılarak 15% test 85% eğitim ile elde edilen sınıflandırma sonuçları.....65

Tablo 4.18 Üçlü öznitelik kümesi kullanılarak 10% test 90% eğitim ile elde edilen sınıflandırma sonuçları.....66

ÖZET

Nörodejeneratif Hastalıkların Entropi Tabanlı Öznitelikler Kullanılarak Makine Öğrenmesi Yöntemleriyle Sınıflandırılması

Dilara ARAPOĞLU

İstatistik Anabilim Dalı

Yüksek Lisans Tezi

Danışman: Prof. Dr. Ersoy ÖZ

Nörodejeneratif hastalıklar son dönemde giderek yaygınlaşmaktadır ve tespit edilmeleri önem taşımaktadır. Yürüyüş anormallikleri, nörodejeneratif hastalıklarda görülen bir belirtidir. Parkinson, Amyotrofik lateral skleroz (ALS) ve Huntington gibi hastalıklar, nörodejeneratif hastalıklar olarak adlandırılır. Parkinson hastalığı, sinir sistemini ve vücudun sinirler tarafından kontrol edilen kısmını etkileyen, yaygın olarak titreme semptomu görülen bir hastalıktır. ALS hastalığı, beyin ve omurilikteki sinir hücrelerini etkileyerek kas kontrolünün kaybına neden olan ve ilerleyebilen bir sinir sistemi hastalığıdır. Huntington hastalığı, beynin bazı bölümlerinin zamanla düzgün çalışmasını engelleyen, kişinin fonksiyonel yetenekleri üzerinde geniş bir etkiye sahip olan ve genellikle hareket, düşünme gibi eylemlerde bozukluğa yol açan bir hastalıktır. Bu tezin amacı, makine öğrenmesi yöntemlerini kullanarak yürüyüş kuvveti yoluyla bir nörodejeneratif hastalık için sınıflandırma yapmaktır. Makine öğrenmesi yöntemlerinde öznitelik olarak tanımlayıcı istatistikler ve entropi tabanlı

öznitelikler kullanılmıştır. Sınıflandırmada kullanılacak algoritmalar; K-En Yakın Komşu, Destek Vektör Makineleri, Lojistik Regresyon, Saf Bayes, Rastgele Orman, XGBoost, LightGBM'dir. Sınıflandırma algoritmaları için Physionet üzerindeki Neurodegenerative Disease Database yürüyüş veri seti kullanılmıştır. Bu veri setinde her denekten 5 dakika boyunca herhangi bir tekerlekli sandalye ya da yardımcı cihaz bulunmadan, ayak bileklerine kayıt cihazı takılarak normal hızlarında yürümeleri istenmiştir. ALS deneği olan 11, Huntington deneği 20, Parkinson deneği 15, kontrol grubu olan sağlıklı deneklerde 16 örnek bulunmaktadır. Bu veri setinden entropi tabanlı öznitelikler yaratılacak olup, bu öznitelikler istatistiksel tabanlı öznitelikler ile sınıflandırma amacıyla kullanılacaktır. Tezin sonucunda ise makine öğrenmesi algoritmaları ile 4 grubun entropi tabanlı öznitelik çıkarımları ve istatistiki öznitelikler ile başarılı bir şekilde sınıflandırıldığı görülmüştür.

Anahtar Kelimeler: Nörodejeneratif hastalıklar, entropi, yürüyüş kuvveti, sınıflandırma

ABSTRACT

Classification of Neurodegenerative Diseases by Machine Learning Methods Using Entropy Based Features

Dilara ARAPOĞLU

Department of Statistics

Master of Science Thesis

Supervisor: Prof. Dr. Ersoy ÖZ

Neurodegenerative diseases have become increasingly common in recent years and their detection is important. Gait abnormalities are a symptom of neurodegenerative diseases. Diseases such as Parkinson's, Amyotrophic lateral sclerosis (ALS), and Huntington's are called neurodegenerative diseases. Parkinson's disease is a common symptom of tremors that affects the nervous system and the part of the body controlled by the nerves. ALS is a nervous system disease that can progress and cause loss of muscle control by affecting nerve cells in the brain and spinal cord. Huntington's disease is a disease that prevents some parts of the brain from working properly over time, has a wide effect on the functional abilities of the person, and often causes impairment in actions such as movement and thinking. The aim of this thesis is to classify a neurodegenerative disease through gait force using machine learning methods. Descriptive statistics and entropy based features are used as features in machine learning methods. Algorithms to be used in classification; K-Nearest Neighbor, Support Vector Machines, Logistic Regression, Naive Bayes, Random Forest, XGBoost, LightGBM. The gait dataset of Neurodegenerative Disease Database on

Physionet was used for classification algorithms. In this data set, each subject was asked to walk at their normal speed for 5 minutes without any wheelchair or assistive device, with a recording device attached to their ankles. There are 11 samples in the ALS subject, 20 in the Huntington subject, 15 in the Parkinson's subject, and 16 in the healthy subjects as the control group. Entropy based features will be created from this data set and these features will be used for classification with statistical based features. As a result of the thesis, it was seen that 4 groups were successfully classified with entropy-based feature extraction and statistical features with machine learning algorithms.

Keywords: Neurodegenerative diseases, entropy, gait force, classification.

1
GİRİŞ

1.1 Makine Öğrenmesi

Makine öğrenmesi, örnek verileri veya geçmiş deneyimleri kullanarak, performansı optimize etmek için bilgisayarları programlama işlemine verilen isimdir. Makine öğrenmesi modelleri, gelecek tahminlerinde bulunmak için öngörücü olabilme veya verilerden bilgi elde etmek için tanımlayıcı olabilme gibi özelliklere sahiptir.

Makine öğrenmesi, matematiksel modeller oluştururken istatistik teorisini kullanır çünkü temel görev bir örnekten çıkarımlar yapmaktır. Bilgisayar biliminin buradaki rolü iki yönlüdür: İlk olarak, eğitimde, genellikle sahip olunan çok büyük miktardaki verileri depolamak ve işlemek kadar optimizasyon problemini çözmek için verimli algoritmalara ihtiyaç duyulur. İkinci olarak, bir model bir kez öğrendiğinde, modelin çıkarımı için algoritmik çözümünün de anlamlı ve etkili olması gerekir. Bazı uygulamalarda, öğrenme veya çıkarım algoritmasının etkinliği, yani uzay ve zaman karmaşıklığı, tahmin doğruluğu kadar önemli olabilir [1].

1.2 Nörodejeneratif Hastalıklar

Nörodejenerasyon, merkezi sinir sistemindeki nöronların ve aksonların yavaş ve ilerleyici işlev bozukluğu ve kaybı anlamına gelmektedir [2]. Alzheimer hastalığı (AH), Parkinson hastalığı (PH) ve Amyotrofik lateral skleroz (ALS), başlıca nörodejeneratif hastalıklardan üçüdür. Bu hastalıkların prevalansı ve insidansı yaşla birlikte önemli ölçüde artmaktadır; bu nedenle birçok ülkede yaşam süreleri uzamaya devam ettikçe vaka sayısının yakın gelecekte artması beklenmektedir [3]. Alzheimer Hastalığı (AH), Multipl Skleroz (MS), Parkinson hastalığı (PH), Huntington hastalığı (HH) vb. gibi nörodejeneratif hastalıklar tedavi edilemez. Dünya Sağlık Örgütü (WHO), 2030 yılında nörolojik bozuklukların dünya çapında ikinci önde gelen ölüm nedeni olacağını tahmin

etmiştir [4]. Şu anda mevcut tedaviler, yalnızca hastalığın hızlı ilerlemesini sınırlayabilir. Nörodejeneratif hastalıklar, ilerleyici bilişsel gerilemeyi içeren, günlük işlevsel yetenekleri sınırlayan ve yürüyüş ve dengedeki eksiklikler dahil olmak üzere motor işlev bozukluklarına yol açan semptomları paylaşır [5,6].

1.3 Literatür Özeti

Nörodejeneratif hastalıkların tespitinde makine öğrenmesi algoritmaları kullanılması sınıflandırmaya yardımcı olacaktır. Özellikle yürüyüş bozuklukları nörodejeneratif hastalıklar için bir klinisyen tarafından, nörodejeneratif hastalığın ilerlemesini teşhis etmek ve değerlendirmek için kullanılan ana semptomdur. Bunlara ilişkin sinyaller hastalıkların tespitinde ve sınıflandırılmasında önem arz etmektedir. Sınıflandırma için gerekli etiketlerin uzmanlar tarafından değerlendirilmesi gerekmektedir. Etiketler belirlendikten sonra model oluşturulabilir.

Hausdorff vd. (1997), adım aralığı korelasyonuna göre yaşlanma ve belirli hastalık durumlarıyla ilişkili nörolojik bağlantıların olabileceğini varsaymıştır. Bu hipotezi test etmek için sağlıklı, yaşlı denekler ve genç kontrol denekleri ile Huntington hastalığı olan deneklerin ve sağlıklı kontrollerin adım aralığı zaman serisini karşılaştırılmıştır. Sağlıklı deneklere göre hasta bireylerin adım aralığı korelasyonunun değiştiği görülmüştür [7]. Nguyen vd. (2020), Yürüyüş Nörodejeneratif Veri Tabanı veri kümesinde yer alan sol ve sağ ayağın hareketlerine ilişkin sinyaller kullanmıştır. Çalışmanın amacı, çok ölçekli örnek entropi ve makine öğrenimi modellerini kullanarak yürüme kuvveti yoluyla bir Nörodejeneratif Hastalık sınıflandırma algoritması geliştirmektir. Uygulama sonucunda, çok ölçekli örnek entropi ve makine öğrenimi sınıflandırıcılarına dayalı bir nörodejeneratif hastalık yürüyüş sınıflandırmasını başarıyla geliştirilmiştir [8]. Setiawan vd. (2022), hastalıklara göre sağ ve sol ayak korelasyonunu incelemiş ve Huntington hastalarının sağ ve sol ayakları birbirini desteklediği için dengede, ALS'li hastaların Huntington hastalarına göre ayak korelasyonunun daha zayıf, Parkinson hastalarının ise alt vücudu dengede tutabilmek için koordine olduğu görülmüştür. Bu çalışmada kullanılan algoritma ile yürüme kuvveti sinyallerinin yürüyüş modelini etkili bir şekilde ayırt ettiği

görülmüştür [9]. Moumita vd. (2022), Nörodejeneratif Hastalık Veri Tabanındaki 12 yürüyüş sinyalini (Elapsed time hariç) kullanarak Backpropagation Neural Network algoritması ile sınıflandırma yapmıştır. Bu özniteliklere ek olarak her sinyalin ortalama ve standart sapması da öznitelik olarak eklenmiştir. Ayrıca deneklerin yaş, boy, ağırlık ve cinsiyet ve yürüme hızları da öznitelik olarak eklenmiştir. Neighborhood Rough Set yöntemini öznitelik çıkarım yöntemi olarak kullanmış ve özniteliklerden en önemli 11 tanesi seçilmiştir. Sadece Parkinson ve kontrol grubu kullanılmıştır. Bunun sonucunda makine öğrenmesi sınıflandırıcısıyla 90.3% başarı elde edilmiştir. WiSARD ve BNN'nin önerilen hibrit modeli, 90.3% doğruluk ile Parkinson'u kontrol deneklerinden etkili bir şekilde ayırır. Önerilen hibrit model, ağırlıksız sinir ağları için doğrusal ve rastgele haritalar kullanılarak gerçekleştirilmiştir [10]. Berke Erdaş, Ç. vd. (2022), hastalık şiddetini derecelendiren bir sistem yaratmak için, veri setinde yer alan 12 yürüyüş sinyali özniteliği ve klinik performans ve kapasite alanını değerlendirmek için kullanılan UHDRS ölçeğine ait öznitelikleri kullanarak, Quick Response modeli kurmuşlardır. Rastgele orman, Çok Katmanlı Algılayıcı ve K-en yakın komşu algoritmaları ile sınıflandırma yapılmıştır [11]. Wu ve Krishnan (2009), sadece ALS hastalarına dair sinyalleri ve kontrol grubu olarak sağlıklı grubu kullanmıştır. Sadece salınım aralığı öznitelikleri hem ALS grubu hem de kontrol grubu için kullanılmış ve Fisher'in Doğrusal Diskriminant Analizi ve En küçük Kareler Destek Vektör Makinesi yöntemleri ile çapraz doğrulama yapılmıştır. En küçük Kareler Destek Vektör Makinesi sınıflandırıcısı ile bu sınıflandırmada 89,66% oranında başarı elde edilmiştir [12]. Wu ve Ng (2010), ALS hastalarını ve kontrol grubuna ait sinyalleri kullanmıştır. Yürüyüş sinyallerinin sınıflandırılması için istatistiksel bir yöntem olan olasılık yoğunluk fonksiyonu tahmin edilmeye çalışılmıştır. Bu tahmin ile, ALS hastalarına ve kontrol grubuna ait sinyaller ile sınıflandırma yapabilmişlerdir [13]. Bilgin (2017), ALS, Huntington, Parkinson ve kontrol grubunun hepsini kullanarak dalgacık dönüşümü yöntemi ile elde ettiği özniteliklerle, Saf Bayes kullanarak sınıflandırma yapmıştır [14]. Bilgin ve Akın (2018), en iyi ayrım yöntemini belirlemek için, nörodejeneratif hastalıklara ilişkin sinyalleri kullanarak önce dalgacık dönüşümü ile öznitelikler elde

etmişler, ardından çok katmanlı algılayıcı sinir ağları, radyal tabanlı fonksiyon sinir ağları, genelleştirilmiş regresyon sinir ağları, destek vektör makineleri ve karar ağacı sınıflandırıcıları ile sınıflandırma yapmışlardır [15]. Alaskar ve Hussain (2018), ALS hastalığının sınıflandırılması için yürüyüş sinyallerini kullanmış, doğrusal diskriminant sınıflandırıcı (linear discriminant classifier), kuadratik diskriminant sınıflandırıcı (quadratic discriminant classifier), Saf Bayes ve destek vektör makineleri ile sınıflandırma sonuçlarını incelemiştir [16]. Aşuroğlu vd. (2018), Parkinson hastalığına dair motor semptomlarını incelemişler ve Yerel Ağırlıklı Rastgele Orman regresyon modelini sınıflandırma amacıyla kullanmışlardır. GRF sinyalleri sayesinde çok sayıda sensör aracılığı ile yürüyüş analizinden hastalık tahmini sağlanabilir. Evrensel Parkinson Hastalığı Derecelendirme Ölçeği (UPDRS) ve Hoehn ve Yahr (H&Y) açısından Parkinson semptomlarının şiddetini tahmin etmek için giriş sinyallerinden çıkarılan sayısal öznitelikleri kullanmışlardır. Amaç, motor bozuklukların ciddiyetini tanımlayan kategorik bir sonuçtan ziyade semptom seviyesinin kesin bir değerinin elde edilmesidir. Bu yaklaşımlar sonucunda etkili bir sınıflandırma yapılmıştır [17]. Zeng vd. (2016), yürüyüş analizi ile Parkinson hastalığını deterministik öğrenme teorisi ile sınıflandırmak için yeni bir yöntem sunmuşlardır. Yapay sinir ağları kullanılarak yapılan bu çalışmada, 93 Parkinson hastasının ve 73 sağlıklı kontrolün yürüme paternleri beşli çapraz doğrulama yöntemiyle sınıflandırılmış ve etkili bir şekilde Parkinson hastalarının yürüme paternlerinin ayırt edildiği görülmüştür [18]. Huang vd. (2019), Huntington hastalığını yürüyüş sinyalleriyle tespit etmek için makine öğrenmesi yöntemi geliştirmiştir. Standart sapma ve çeyrekler arası aralık ile altı çeşit sinyal kullanarak 5 farklı makine öğrenimi modeli için özellikleri ve hiperparametre alt kümelerini optimize etmek üzere kapsamlı bir öznitelik seçimi uygulanmıştır. Sınıflandırma sonucu olarak Destek Vektör Makineleri en yüksek doğruluğu elde etmiştir [19]. Hausdorff vd. (1997), Nörodejeneratif Veri Tabanında yer alan hastalıklar ve içerdikleri sinyallere göre, sağlıklı kontrol grubuna kıyasla ALS'li hastaların yürüyüşünün daha az sabit ve geçici olarak daha düzensiz olduğunu bulmuşlardır [7]. Huang vd. (2021), entropi tabanlı yöntemleri ile yürüyüş stabilitesi arasındaki ilişkiyi araştırmış ve Ortalama Entropi yönteminin statik ayakta duruş stabilitesini, Çok

Ölçekli Entropi ve Entropi Entropisinden daha iyi ölçebileceğini bildirmiştir. Burada farklı veri setleri birleştirilerek bir analiz yapılmıştır. Entropiler ve istatistiksel öznitelikler ile Kuadratik Diskriminant Analizi uygulanmıştır. Ortalama Entropi sisteminin hastalıkları ayırt etmede daha iyi performans sergilediği görülmüştür [20].

Tablo 1.1 Yürüyüş analizi yapan bazı makalelerin özeti

Yürüyüş Analizi Yapan Bazı Makalelerin Özeti			
Makale	Öznitelik çıkarımı	Sınıflandırma modeli	Entropi kullanımı
Nguyen vd. (2020) [8]	Çok ölçekli örnek entropi, istatistiksel öznitelikler	KNN, Destek Vektör Makineleri	Evet
Steiwan vd. (2022) [9]	Wavelet analizi, temel bileşenler analizi	Yapay sinir ağları	Hayır
Moumita vd. (2022) [10]	Her sinyalin ortalama ve standart sapması, deneklerin yaş, boy, ağırlık ve cinsiyet ve yürüme hızları, Neighborhood Rough Set	Yapay sinir ağları	Hayır
Berke Erdaş vd. (2022) [11]	UHDRS ölçeği	Rastgele orman, Çok Katmanlı Algılayıcı ve KNN	Hayır
Wu vd. (2009) [12]	Belli bir saniyede dönen salınım sayısı, ortalama adım aralığı	Doğrusal Diskriminant Analizi ve En küçük Kareler Destek Vektör Makinesi	Hayır
Wu vd. (2010) [13]	Sol ayak adım aralığının ortalaması ve değiştirilmiş Kullback-Leibler sapması	Destek vektör makineleri	Hayır
Bilgin (2017) [14]	Dalgacık dönüşümü	Saf Bayes	Hayır
Bilgin vd. (2018) [15]	Dalgacık dönüşümü	Çok katmanlı algılayıcı sinir ağları, radyal tabanlı fonksiyon sinir ağları, genelleştirilmiş regresyon sinir ağları, destek vektör makineleri ve karar ağacı	Hayır
Alaskar vd. (2018) [16]	Standart sapma, minimum, maksimum, karekök ortalama, periyodik faaliyetlerin zamansallığı	Doğrusal diskriminant sınıflandırıcı, Kuadratik diskriminant sınıflandırıcı, Saf Bayes ve destek vektör makineleri	Hayır

Tablo 1.1 Yürüyüş analizi yapan bazı makalelerin özeti (devamı)

Aşuroğlu vd. (2018) [17]	Evrensel Parkinson Hastalığı Derecelendirme Ölçeği (UPDRS) ve Hoehn ve Yahr (H&Y) ölçeklerindeki sonuçlar	Rastgele Orman	Hayır
Zeng vd. (2016) [18]	Yürümeye dair öznitelik vektörleri	Yapay sinir ağları	Hayır
Huang vd. (2019) [19]	Standart sapma ve çeyreklik	Destek vektör makineleri	Hayır
Huang vd. (2021) [20]	Ortalama Entropi, Çok Ölçekli Entropi ve Entropi Entropisi	Diskriminant Analizi	Evet
Li vd. (2014) [21]	Biyometrik öznitelikler	Destek vektör makineleri ve radyal tabanlı çekirdek	Hayır
Ren vd. (2016) [22]	İndeks ρ ve Koşullu Entropi	Yapay Sinir Ağları (Çok Katmanlı Algılayıcı), Rastgele Orman ve Saf Bayes	Evet
Yang vd. (2009) [23]	Biyometrik özellikler, zaman ve frekans alanından öznitelikler	Kuadratik Bayes, Destek vektör makineleri	Hayır
Liao vd. (2008) [24]	Örnek Entropi	Sınıflandırma yapılmamış	Evet
Cirugeda-Roldan vd. (2014) [25]	Bulanık Entropi, Örnek Entropi, Yaklaşık Entropi ve DFA	Sınıflandırma yapılmamış	Evet
Xia vd. (2015) [26]	İstatistiksel değerler, Bulanık Entropi	DVM, Yapay Sinir Ağları ve KNN	Evet
Prabhu vd. (2020) [27]	RQA parametreleri, istatistiksel değerler	DVM, Olasılıksal Yapay Sinir Ağları	Hayır
Lin vd. (2020) [28]	Tekrarlama grafiği, Temel Bileşen Analizi	Evrişimsel Sinir Ağları	Hayır
Pham vd. (2018) [29]	Bulanık tekrarlama grafiği, gri seviye birlikte oluşum matrisi	En Küçük Kareler Destek Vektör Makineleri, Doğrusal Diskriminant Analizi	Hayır

1.4 Tezin Amacı

Nörodejeneratif hastalıkların tespit edilmesi ve teşhis konulmasında, doktor ya da klinisyen görüşü haricinde yürüyüş sinyalleri de belirleyici bir özellik olduğundan, bu çalışmada temel amaç hastalıkları yürüyüş sinyalleri ile başarılı bir şekilde sınıflandırmak ve bu sinyaller ile nörodejeneratif hastalıkları tespit etmektir. Sınıflandırmalar için öznitelikler kullanılırken, entropi tabanlı özniteliklerin sınıflandırmaya katkısı görülmüştür. Bu veri setinin entropi ile çalışıldığı örnekler literatürde yer alsa da bu çalışmada entropi öznitelikleri ile istatistiksel öznitelikler birleştirilerek elde edilmesi amaçlanmaktadır.

1.5 Hipotez

Bu çalışmada nörodejeneratif hastalıkların entropi tabanlı ve istatistiksel tabanlı öznitelikler ile başarılı bir şekilde sınıflandırılıp sınıflandırılmayacağı anlaşılmaya çalışılmıştır.

2
NÖRODEJENERATİF HASTALIKLAR VE YÜRÜYÜŞ ANALİZİ

2.1 Nörodejeneratif Hastalıklar

2.1.1 Parkinson Hastalığı

Parkinson hastalığı en yaygın nörodejeneratif hastalıklardan biridir. Hastalığın Parkinson'u bir hareket bozukluğu haline getiren motor belirtileri, hastalığın ayırt edici özelliklerini temsil etmektedir. Parkinson hastalığına teşhis konulmasının zorluğuna yardımcı olması için, modern görüntüleme veya laboratuvar testleri kullanılabilir. Nüfusun her binde 1-2'sini etkileyen Parkinson, ileri yaş gruplarında daha sık görülür [30].

2.1.1.1 Parkinson Hastalığının Tarihçesi

Parkinson hastalığının bileşenlerine çok eski belgelerde rastlansa da ilk net tıbbi tanım 1817'de James Parkinson tarafından yazılmıştır. 1800'lerin ortalarında özellikle Jean-Martin Charcot, bu tanımın genişletilmesinde ve Parkinson hastalığı hakkındaki bilgilerin uluslararası düzeyde yayılmasında etkili olmuştur. Jean-Martin Charcot, Parkinson hastalığını multipl skleroz ve titreme ile nitelendirilen diğer nörodejeneratif bozukluklardan ayırdı. Daha sonra Parkinsonizm-plus sendromlar arasında sınıflandırılabilecek vakaları tanımladı. Parkinson hastalığının ilk tedavileri ampirik gözlemlere dayanıyordu ve antikolinerjik ilaçlar on dokuzuncu yüzyılın başlarında ilk kez kullanılmaya başlandı. Tarihsel olarak önemli diğer anatomik, biyokimyasal ve fizyolojik çalışmalar, Parkinson hastalığı için ek farmakolojik ve beyin cerrahisi alanındaki hedefleri belirledi ve modern klinisyenlerin hala tedavi edilemeyen bu hastalıkta fonksiyonları iyileştirmeyi amaçlayan bir dizi terapi sunmasına imkân sağladı [31].

Parkinson hastalığı, tüm dünyada erkeklerde kadınlara göre 1,5-2 kat daha sık görülmektedir ve batı ülkelerinde, Asya ülkelerine göre daha yaygındır. Aynı

ülkelerde bile, farklı etnik popülasyonlardaki insidansta farklılıklar gözlemlenmiştir.

Patil (2022)'ye göre bu bulgu, Parkinson hastalığının gelişme riski üzerindeki genetik etkiye işaret etmektedir. Aynı zamanda, kendi ülkelerindeki benzer popülasyonlarla karşılaştırıldığında, Amerika'daki Japonlar ve Afrikalılar arasında daha yüksek Parkinson insidansı, Parkinson hastalığının oluşmasına dair çevresel bir rolün olabileceğini öne sürmektedir. Son yirmi yılda, bilim adamları Parkinson ile ilişkilendirilebilecek birkaç çevresel faktör gözlemlemiştir. Örnek olarak, sigara içme, kahve içme, şiddetli egzersiz, ibuprofen kullanımı ve plazma ürat ile Parkinson hastalığı ters ilişkiler içinde iken, genel pestisit maruziyeti (belirli pestisitlerin kullanımı ve travmatik beyin hasarı) ile pozitif ilişki içindedir [32]. Sertlik, titreme, bradikinezi ve bozulmuş denge gibi klasik PD semptomlarına, beyindeki dopamin üreten hücrelerin 60%-80%'ine veya daha fazlasına zarar verildikten sonra tanık olunur. Parkinson hastalığı olan bireylerin sayısı, Geçtiğimiz nesil boyunca, dünya çapında 6 milyondan fazla kişiyi geçmiştir [33]. 1990'dan bu yana, hastalığın yaygınlık oranının dünya çapında 74% arttığı görülmüştür. Parkinson'un hastalık yükündeki artışın nedenleri, dünya genelinde artan sanayileşmenin sonucunda ortaya çıkan çevresel faktörlere bağlanabilir [34]. Genel olarak, sağlığın iyi olması, daha iyi sosyoekonomik durum ile pozitif ilişkilidir, ancak Parkinson hastalığı için bunun tersi geçerlidir [35].

2.1.1.2 Parkinson Hastalığının Teşhisi

Parkinson hastalığının teşhis ve semptomlarının takibi için altın standart, birkaç sübjektif bileşen içeren klinik değerlendirmeye dayanmaktadır. Teşhis ve semptomların izlenmesi için objektif ve kantitatif biyobelirteçlerin olmaması, doğrudan ve dolaylı sağlık hizmetleri maliyetine yol açar [33]. Mevcut teşhis kriterlerine [34] dayalı olarak, teşhis hatası oranı %20 civarındadır [35]. Ayrıca Parkinson hastalık sırasında semptomlar değişebildiği için, tedavinin sürekli ayarlanmasını gerektiren dinamik bir hastalıktır [33].

2.1.2 Amyotrofik Lateral Skleroz (ALS) Hastalığı

Motor nöron hastalığı olarak da bilinen amyotrofik lateral skleroz (ALS), hem üst hem de alt motor nöronların dejenerasyonu olarak tanımlanır, bu dejenerasyon zamanla kas zayıflığına ve sonunda da felce yol açar. Çoğu hastada, ALS'nin nedeni bilinmemektedir, ancak bazı kişilerde motor olmayan hücrelerdeki roller de dahil olmak üzere çok çeşitli işlevlere sahip genlerdeki mutasyonlarla ilişkili ailesel hastalık mevcuttur. ALS'nin ortaya çıkış şekli hastalar arasında değişebilir; bazıları spinal başlangıçlı hastalık (uzuvlarda kas güçsüzlüğünün başlangıcı) ile başvurur, ancak diğer hastalar, dizartri (konuşma güçlüğü) ve disfaji (yutma güçlüğü) ile tanımlanan bulbar başlangıçlı hastalık ile başvurabilir [39]. ALS'nin sınıflandırılması, kullanılan kriterlere bağlı olarak değişebilir. ALS alt gruplarının geleneksel tanımları, üst ve alt motor nöronların tutulum derecesine dayanır, ancak diğer sınıflandırma sistemleri, başlangıç yeri (yani, bulbar başlangıçlı veya spinal başlangıçlı hastalık), seviye gibi farklı parametreler içerir. Bugüne kadar, bu sınıflandırma sistemlerinin hiçbiri bilişsel veya davranışsal semptomları içermemiştir ve her sınıflandırma sisteminde bir dizi alt fenotip ve klinik yol gözlemlenebilir [40].

2.1.2.1 ALS Hastalığının Epidemiyolojisi

ALS'nin toplum temelli epidemiyolojik çalışmalarının çoğu, yüksek kaliteli Avrupa hasta kayıtlarından alınmıştır. Avrupa'da görülme sıklığı 100.000 kişi başına 2 ila 3 vaka arasında değişmektedir. Tanımlanmış coğrafi alanlar, insidansı ve prevalansı tahmin etmek ve ALS ile ilgili hizmetlerin riski, klinik gidişatı, sonucu ve kullanımı ile ilgili daha ayrıntılı çalışmaları desteklemek için tercih edilir [41]. Bu nüfusa dayalı kayıtlar, Avrupa ülkeleri arasında ALS insidansını karşılaştıran verileri sağlayan Avrupa ALS Epidemiyoloji Konsorsiyumu'nu (EURALS) oluşturmak için birleştirildi [42]. Hardiman vd. (2017) çalışmasında, ALS nadir görülen bir hastalık olduğundan, birden fazla tespit kaynağına sahip popülasyona dayalı bir yaklaşımın, tüm fenotipik spektrumu tanımlamanın en iyi yolunun olduğundan, çünkü nüfusa dayalı kayıtların hastalık hakkında, genellikle daha genç hastalar ve daha az şiddetli hastalığı olanlar lehine önyargılı olan uzman kliniklerden alınan veri setlerinden

daha eksiksiz bilgi sağlamakta olduğundan söz etmişlerdir [43]. Daha önceki varsayımların aksine, ALS insidansı atasal kökene göre farklılık gösterir; Avrupa kökenli popülasyonlarda yapılan araştırmalar insidansın 100.000 kişide minimum 3 vaka olduğunu göstermiştir [44, 45] ancak insidans Doğu Asya'da (100.000 kişide yaklaşık 0.8 vaka) ve Güney Asya'da (100.000 kişide yaklaşık 0.7 vaka) daha düşüktür. Bazı bölgelerde (Japonya'nın Guam ve Kii yarımadası gibi) bildirilen insidans çok yüksektir ancak belirsiz kalan nedenlerle son 30 yılda önemli ölçüde azalmıştır. Farklı ata popülasyonlarının birbirine yakın yaşadığı bölgelerde (Kuzey Amerika'da olduğu gibi), yerli popülasyonlarda ALS insidansı düşüktür (100.000 kişi başına 0,63 vaka) [46]. Nispeten homojen nüfusa sahip bölgelerde (İrlanda, İskoçya ve Faroe Adaları gibi) bildirilen insidanslar yüksektir (100.000 kişide 2,6 vaka) [42].

Ek olarak, farklı ata popülasyonlarında ALS'nin fenotipi ve doğal seyrindeki farklılıklar bildirilmiştir; bildirilen hayatta kalma süresi Avrupa'da (24 ay) Orta Asya'dakinden (48 ay) daha kısadır [47, 48]. Avrupa'da çoğu erkekte spinal başlangıçlı hastalık vardır ve kadınların bulbar başlangıçlı hastalığa eğilimi daha yüksektir. Eldeki verilere göre Avrupa'da tanı yaşı ve ilk belirtiler Asya ve Güney Amerika'ya göre daha yüksektir. Avrupa'da başlangıç yaşı 65 yaşında doruğa ulaşır [42].

2.1.2.2 ALS Hastalığına Sebep Olabilecek Durumlar

- Genetik

  ALS, bazı durumlarda Mendel kalıtım modeline sahip, ancak geri kalanında fark edilebilir bir aile öyküsü olmayan karmaşık bir genetik bozukluk olarak kabul edilir. Nüfusa dayalı kayıtlar kullanılarak geliştirilen matematiksel modeller, ALS'li bireylerin, hastalığın oluşmasına yol açan en az altı kavramsal adımlık bir dizi aracılığıyla çevresel faktörlerle etkileşime giren bir dizi 'risk altındaki' varyantları taşıyabileceklerini öne sürdü. Bu adımlardan birinin genetik risk (doğumdan itibaren) olduğu düşünülmektedir, ancak geri kalan adımlara yol açan çevresel faktörlerin etkileşimi henüz tanımlanmamıştır [43].
- Çevresel faktörler ve yaşam tarzı faktörleri

Epidemiyolojik vaka kontrol çalışmaları, ALS'nin çevresel nedenlerini belirlemeyi amaçlamıştır. ALS, genel popülasyona göre sporcu grupları arasında daha yüksek bir sıklıkta rapor edilmiştir [43].

- Bilişsel ve davranışsal eksiklikler

  ALS için standart teşhis ve sınıflandırma parametreleri, hastanın bilişsel veya davranışsal durumunu içermez. Klinikte bilişsel ve davranışsal değişiklikleri olan ALS'li hastaları belirlemek için birkaç dilde geçerliliği olan ve yaygın olarak kullanılan Edinburgh Bilişsel ve Davranışsal ALS Taraması (ECAS) gibi yüksek duyarlılık ve düşük özgüllüğe sahip çeşitli tarama araçları tasarlanmıştır [49].

  Bilişsel ve davranışsal değişikliklerin saptanması, ALS'li hastalar ve bakıcıları için önemlidir çünkü yürüme bozukluğu hastalık gidişatının daha hızlı olması ile ilişkilidir [50].

2.1.3 Huntington Hastalığı

Huntington hastalığı, yerleşik bir genetik kökene ve beyin hastalığının belirli bölgelerine atıfta bulunabilen semptomlara sahip, ilerleyebilen bir nörodejeneratif bozukluktur. Hücresel ve moleküler teknikler, hastalığın ilerleyişinin nasıl olabileceğini hızla aydınlatmakta ve akılcı tedaviler geliştirmeye yönelik yaklaşımlara öncülük etmektedir. Bu nedenle Huntington, anormal beyin fonksiyonunun ağırlıklı olarak genetik faktörlerden kaynaklandığı düşünülen psikiyatrik bozuklukların gelecekteki çalışmaları için bir model görevi görür. Huntington, motor, bilişsel ve duygusal bozuklukların bir üçlüsü olarak tanımlanabilir [51]. Semptomlar genellikle 35 ila 50 yaşları arasında başlar, ancak başlangıç çocukluktan yaşlılığa kadar herhangi bir zamanda ortaya çıkabilir. Ölüm, semptomların ilk ortaya çıkmasından ortalama 15 ila 20 yıl sonra meydana gelir; bazı hastalar düşme veya intihar nedeniyle daha erken hayatını kaybedebilirken, diğerleri 30 ila 40 yıl hayatta kalabilir. Huntington hastalığında hareket bozukluğu iki bileşenden oluşur: istemsiz hareketler ve anormal istemli hareketler. Kore veya koreoatetoz, Huntington ile en sık ilişkilendirilen hareket anormalliğidir. Sürekli ve düzensiz sarsıntılı hareketlerden veya kıvranma hareketlerinden oluşur. Bununla birlikte, istemli hareket bozuklukları, hastalığın beyindeki boyutuna göre, fonksiyonel engellilik

ve hastalık ciddiyeti ile daha yüksek oranda ilişkilidir. Huntington hastalığında gözlemlenen düzensiz istemli hareketler şunlardır: yavaş, hipometrik sakkadlar (göz testi) ve akılda kalıcı takip gibi anormal göz hareketleri; koordinasyonsuz, aritmik ve yavaş ince motor hareketler; disfaji ve dizartri; katılık; ve yürüyüş bozuklukları [52]. Huntington için ortalama başlangıç yaşı yaklaşık 45'tir. Etkilenen bireylerin yaklaşık üçte ikisi ilk önce nörolojik belirtiler gösterir; diğerleri psikiyatrik değişiklikler gösterir. Teşhisi takip eden erken aşamalarda, belirtiler göz hareketlerinde, koordinasyonda, küçük istemsiz hareketlerde, zihinsel planlamada güçlükte ve sıklıkla depresif veya sinirli bir ruh halindeki küçük değişiklikleri içerir. Etkilenen bireyler genellikle olağan aktivitelerinin çoğunu gerçekleştirebilir ve çalışmaya devam edebilir [53, 54].

Doğru prevalans tahminleri, hastalık başlangıcının nörolojik değerlendirmesiyle birlikte kapsamlı genetik testlere bağlıdır. Hem genetik hem de klinik teşhis standartlarını içeren yaygınlık çalışmaları, Batı popülasyonlarında 100.000'de 10.6-13.7 kişinin veya 7.300'de 1'in bu hastalıktan etkilendiğini göstermektedir [55, 56].

Genel olarak, Huntington hastalığı olan kişiler arasındaki yürüyüş, kore ile mekansal ve zamansal yürüyüş değişkenliği ile karakterizedir. Ek olarak, Huntington hastaları, çeşitli dikkat dağınıklığı, dış işaret ve hız değişimi olduğu durumda yürüyüşü kontrol etme yeteneklerini kaybetmişlerdir. Premanifest bireyler arasında, genel olarak, bradikinezi, akinezi, yürüme değişkenliği ve bozulmuş dinamik denge, Huntington hastalığı için tespit edilebilen temel özelliklerdir.

Huntington hastalığında, düşme yaygındır ve Huntington hastalarının yaşam kalitesini büyük ölçüde etkiler. Egzersiz müdahaleleri ve rehabilitasyona ilişkin çalışmalardan elde edilen kanıtlar, bu müdahalelerin bazı olumlu etkilerle birlikte Huntington'a özgü yürüyüş ve denge eksikliklerini ele almak için güvenli, uygulanabilir ve kabul edilebilir olduğunu göstermektedir [57].

2.2 Yürüyüş Analizi

Yürüyüş analizi hem insanların hem de hayvanların yürüme şekillerini inceler [58, 59]. Bir yürüyüş döngüsü, yürüme sırasında alt uzuvların hareketini içeren

bir dizi fiziksel eylemdir. Yürüyüş döngüsü, aynı ayağın birbirini izleyen iki topuk vuruşu arasındaki aralık olarak tanımlanır ve buna da adım denir. Adımlar, iki aşamadan oluşur: basma fazı ve salınım fazı. Basma fazı, ayağın topuktan parmağa temas dizisini içerir. Salınım fazı, ayak havada asılıyken devam eder. Ortalama olarak, basma fazı yürüyüş döngüsünün 60%'ını, salınım fazı ise 40%'ını oluşturur. Ayrıca, her faz bir dizi Çift Destek (her iki ayağın da yerle temas halinde olduğu durum) ve Tek Destek (yalnızca bir ayağın yerle temas halinde olduğu durum) alt fazlarını içerir. Bu tanımlar yürüyüş analizi ile ilgili tüm çalışmalar için geçerlidir [60].

Yürüyüş analizi ile ilgili olarak nörodejeneratif hastalıklarla semptomlar ve yürüme karakteristiği Tablo 2.1'de verildiği gibi incelenebilir:

Tablo 2.1: Nörodejeneratif hastalıklar ve yürüyüş özellikleri [60]

Hastalıklar	Semptomlar	Yürüme Karakteristiği
Parkinson Hastalığı	Hipokinetik hareket, bradikinezi, hipotoni, titreme, duruş bozukluğu, istemsiz adım hızlanması, duruş reflekslerinin kaybı ve donma	Yürüme hızı düşüşü Yükselen kadans Düşen adım aralığı Düşen salınım süresi Daha yüksek çift destek süresi
Amyotrofik Lateral Skleroz	Dalgalanma dinamiklerinde düzensizlikler, değişen yürüyüş ritmi, bacaklarda, ayaklarda veya ayak bileklerinde zayıflık.	Düşen yürüyüş hızı Artan adım süresi değişkenliği Artan adım süresi
Huntington Hastalığı	Kontrol edilemeyen hareketler, duygusal problemler, psikiyatrik hastalıklar ve düşünme yeteneği kaybı	Düşen yürüyüş hızı Düşen adım aralığı Artan salınım fazı Azalan tek destek süresi

3
MATERYAL VE METOTLAR

3.1 Makine Öğrenmesi

3.1.1 Destek Vektör Makineleri

Öncelikle ikili sınıflandırma için tasarlanmış sınıflama algoritmasıdır. Destek Vektör Makineleri (DVM'ler) sınıflandırma, regresyon ve diğer öğrenme görevleri için popüler bir makine öğrenme yöntemidir [61]. Yapısal risk minimizasyonuna ve istatistiksel öğrenme teorisine dayalı olarak geliştirilmiştir. Destek vektör makineleri yüz tanıma, görsel sınıflamada, biyoenformatikte, el yazısı tanımada kullanılabilir. Yöntem olarak iki sınıfı birbirinden ayırabilmek için en uygun hiperdüzlemin bulunması gerekir, eğer veriler lineer olarak ayrılabiliyorsa bu lineer DVM, lineer olarak ayrılamıyorsa lineer olmayan DVM şeklinde nitelenebilir. Lineer olmayan DVM'de haritalama yöntemi kullanılır, böylece orijinal veriler daha yüksek boyuta taşınarak en iyi lineer ayırıcı hiperdüzlem aranır [62].

3.1.1.1 Lineer Olarak Ayrılabilen Durum

n adet örnek için eğitim kümesi X ile denklem 3.1'deki gibi gösterilebilir.

$$X = \left\{(x_1, y_1), \dots, (x_n, y_n), x_i \in R^d, y_i \in \{-1, +1\}\right\} \tag{3.1}$$

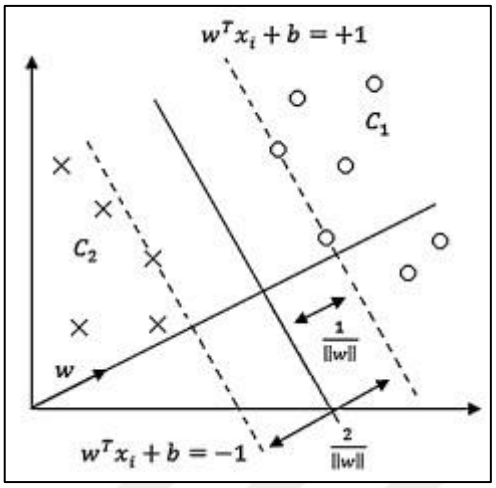
Burada x_i girdi vektörü ve y_i, x_i'ye karşılık gelen etiket olarak tanımlanır [63].

$$f(x) = w^T x_i + b, \qquad w \in R^d \; ve \; b \in R \tag{3.2}$$

$f(x)$ hiperdüzlem ile ilgili diskriminant fonksiyonu ve karar fonksiyonu ise signum$\big(f(x)\big)$'tir. Buradaki amaç, Şekil 3.1'de gösterilen optimum hiperdüzlemde birlikte tanımlanan w vektörünü ve b skalerini belirlemektir. Hiperdüzlem denklem 3.3 ve denklem 3.4 ile verilen kısıtları sağlamalıdır:

$$w^T x_i + b \geq +1, y_i = +1 \; için \tag{3.3}$$

$$w^T x_i + b \leq -1, y_i = -1 \; için \tag{3.4}$$



Şekil 3.1 Destek Vektör Makineleri'nde lineer olarak ayrılabilen durum için optimal ayrılan hiperdüzlem [1]

Bu kısıtlar tek kısıt olarak denklem (3.5) formunda yazılabilir:

$$y_i(w^T x_i + b) \geq +1, \quad i = 1, \ldots, n \tag{3.5}$$

$w^T x_i + b = +1$ ve $w^T x_i + b = -1$ hiperdüzlemleri arasındaki bölge marj aralığı olarak tanımlanır [3]. Bu marj aralığının genişliği $\frac{2}{||w||}$ ile hesaplanabilir. Marj aralığının maksimum olması örneklerin olabildiğince iyi ayrıldığını gösterir. Dolayısıyla marj aralığını maksimum yapacak bir amaç fonksiyonu tanımlanabilir. Bu amaç fonksiyonu marj aralığının tersinin minimizasyonu olarak düşünülebilir ve böylece aşağıda verilen lineer kısıtlı kuadratik optimizasyon problemi oluşturulur.

Amaç fonksiyonu denklem (3.6)'da gösterildiği gibi oluşturulabilir.

$$\frac{min}{2\|w\|^2} \tag{3.6}$$

$$y_i(w^T x_i + b) \geq +1 \tag{3.7}$$

Pozitif örneklerin hiperdüzlemin bir tarafında, negatif örneklerin hiperdüzlemin diğer tarafında yer alacağı bir optimal hiperdüzlem vardır ve tektir [64].

Denklem 3.6 ve denklem 3.7 ile verilen optimizasyon problemi α_i Lagranj Çarpanlarını göstermek üzere, Lagranj Çarpanları Yöntemi ile kısıtsız optimizasyon problemine çevrildiğinde denklem (3.8) elde edilir.

$$L_p = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{n} \alpha_i [y_i(w^T x_i + b) - 1] \tag{3.8}$$

L_p ile ifade edilen Lagranj fonksiyonu $\alpha_i \geq 0$ için maksimize, w ve b için minimize edilmelidir. Karush-Kuhn-Tucker(KKT) koşulları kullanılarak, L_d dual problem maksimizasyon olacak şekilde, denklem (3.9) ve denklem (3.10)'da gösterildiği gibi oluşturulabilir.

$$\frac{\partial L_p}{\partial w} = 0 \rightarrow \sum_{i=1}^{n} \alpha_i y_i x_i = w, \tag{3.9}$$

$$\frac{\partial L_p}{\partial b} = 0 \rightarrow \sum_{i=1}^{n} \alpha_i y_i = 0. \tag{3.10}$$

Dual problem denklem (3.9) ve denklem (3.10)'un, denklem (3.8)'de yerine yazılması ile kuadratik programlama probleminin duali olarak aşağıdaki şekilde, denklem (3.11) ve denklem (3.12)'de gösterildiği gibi elde edilir:

$$L_d = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j {x_i}^T x_j \tag{3.11}$$

$$\sum_{i=1}^{n} \alpha_i y_i = 0 \ ve \ \alpha_i \geq 0 \tag{3.12}$$

Bu kuadratik programlama problemi α_i'yi belirlemek için herhangi standart kuadratik optimizasyon metoduyla çözülebilir. n tane değişken olmasına rağmen, bunların çoğu göz ardı edilebilir. α_i pozitif değerlerine karşılık gelen x_i örnekleri destek vektörler olarak adlandırılır. Sonuç olarak w vektörü denklem 3.9 kullanılarak belirlenir. Tüm destek vektörler için son bilinmeyen b parametresi

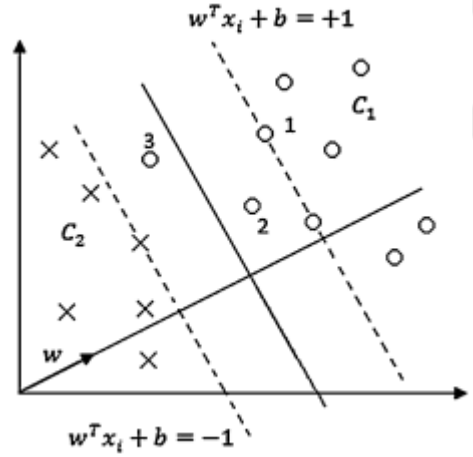
$$b = \ y_i - w^T x_i \tag{3.13}$$

denkleminin ortalaması alınarak belirlenir [1].

3.1.1.2 Lineer Olarak Ayrılamayan Durum

Bir önceki kısımda farklı bölgelerde bulunan farklı sınıf etiketlerinin veri örneği bakımından, verinin kusursuzca ayrılabildiği varsayıldı. Ama pratikte, verideki örneklerin dağılımları nedeniyle böyle bir hiperdüzlem bulunmayabilir. Bu lineer ayrılabilirliği zorlaştırır çünkü basit lineer karar sınırları genellikle modelleri yüksek doğrulukla sınıflamak için yeterli değildir. Eğer veri lineer olarak ayrılamıyorsa, denklem (3.5)'teki kısıtlama etkisiz olabilir. Bu şuna karşılık gelir, Şekil 3.2'deki gibi veri örneği marj içine ya da karar sınırına düşer. Ayrılamayan verilerin ele alınabilmesi için, her bir örnek için ξ aylak değişkeni olarak tanıtılan değişkenin eklenmesiyle formülasyon genelleştirilir.

$$y_i(w^T x_i + b) \geq 1 - \xi_i, \qquad i = 1, \dots, n, \xi_i > 0 \tag{3.14}$$



Şekil 3.2 Destek Vektör Makineleri'nde lineer olarak ayrılamayan durum için optimal ayrılan hiperdüzlem [63]

Yeni optimizasyon problemi iki amacın birleşimi olarak tanımlanır: marj maksimizasyonu ve hata minimizasyonu. Eğer $\xi = 0$ ise x_i örnekleri düzgünce sınıflandırılmıştır. (Şekil 3.2'deki Nokta 1) Eğer $0 < \xi < 1$ ise x_i örnekleri yine düzgünce sınıflandırılmıştır ama marj içindedirler. (Şekil 3.2'deki Nokta 2) Son olarak, eğer $\xi \geq 1$ ise x_i örnekleri yanlış sınıflandırılmıştır. (Şekil 3.2'deki Nokta 3) Hatalı sınıflandırmaların sayısı, ξ'nin 1'den büyük olanlarının sayısıdır. Diğer

taraftan ayrılamayan noktaların sayısı pozitif ξ sayısıdır. Sonuç olarak yumuşak hata $\sum_{i=1}^{n} \xi_i$. denklem (3.8)'in başına eklendiğinde denklem (3.15) elde edilir:

$$L_p = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i[y_i(w^T x_i + b) - 1] - \sum_{i=1}^{n} \mu_i \xi_i \qquad (3.15)$$

Lagranj parametresi μ_i ile ξ değerlerinin pozitif olması garanti edilir. C parametresi, destek vektörlerin sayısı ile ayrılamayan noktaların sayısı arasındaki ağırlığı ayarlamak için kullanılır. Diğer bir deyişle marj aralığı içindeki örnekler, testte daha iyi bir genelleme elde etmek için hatalı sınıflandırılmış örneklerle birlikte cezalandırılır.

Dual problem:

$$L_d = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j {x_i}^T x_j \qquad (3.16)$$

Kısıt:

$$\sum_{i=1}^{n} \alpha_i y_i = 0. \qquad (3.17)$$

α_i'nin tutulduğu örnekler destek vektör değildir. Denklemin kalan kısmı, w ve b parametrelerinin benzer şekilde belirlenebileceğini tanımlar [1].

3.1.1.3 Lineer Olarak Ayrılamayan Durum

Veri, lineer olarak ayrılamayan durumdaysa, lineer destek vektör makineleri kullanılamaz. Bu durum için lineer olmayan destek vektör makineleri kullanmak daha iyidir. Bu girdiyi dönüştürmek amacıyla, lineer olmayan destek vektör makineleri önce lineer olmayan çekirdek kullanır, ardından destek vektör makineleri kullanır. Lineer olmayan çekirdek fonksiyonu, büyük ihtimalle büyük bir matristir. Çekirdek fonksiyonu girdiyi yüksek boyutlu bir uzaya dönüştürür. Bu işleme haritalama denir. Lineer olmayan destek vektör makinelerinde Çekirdek fonksiyonu kullanıldığında haritalama uzun süre alabilir. Lineer olarak ayrılamayan veriyi, lineer olarak ayrılabilen veriye dönüştürmek için, lineer olmayan ϕ fonksiyonu kullanılarak $\phi: R^n \rightarrow H$ formunda, daha yüksek boyutlu öznitelik uzayı olan Hilbert uzayına haritalanır. Sonuç olarak, maksimum marjlı hiperdüzlem, lineer olmayan destek vektör makinelerinin yardımı ile öznitelik

uzayına uyar. Bu öznitelik uzayının içinde bir lineer sınıflandırma problemi formüle edilmiştir [65]. Buna bağlı olarak, dual optimizasyon probleminin Lagranjını işaret eden denklem 3.11 ve denklem 3.12 aşağıdaki şekilde değişmelidir:

$$L_d = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K\left(x_i{}^T x_j\right) \tag{3.18}$$

Çekirdek Fonksiyonu girdi vektörlerini içerir ve denklem (3.19)'da gösterildiği şekilde ifade edilir:

$$K\left(x_i{}^T x_j\right) = \phi(x_i)^T \phi(x_i) \tag{3.19}$$

Çekirdek fonksiyonlarının, dönüştürülen öznitelik uzayında karşılık gelen bir iç çarpıma sahip olması gerekir. Bu Mercer'in Teoremi olarak adlandırılır [66]. Böylece dual problemin çözümü basitleşir çünkü dönüşmüş uzaydaki iç çarpım için hesaplama yapmaya gerek yoktur. Sıklıkla kullanılan çekirdekler Tablo 3.1'de gösterilmiştir.

Tablo 3.1 γ, r, τ ve d çekirdek parametreleri ile literatürde en çok kullanılan çekirdek fonksiyonları

Çekirdek	$\boldsymbol{K\left(x_i{}^T x_j\right)}$
Lineer	$x_i{}^T x_j$
Radyal Tabanlı Fonksiyon	$\exp\left(-\gamma \left\|x_i - x_j\right\|^2\right), \gamma > 0$
Polinomiyal	$\left(\gamma x_i{}^T x_j + R\right)^d, \gamma > 0$
Sigmoid	$\tanh\left(\gamma x_i{}^T x_j + R\right)^d, \gamma > 0$
Çoklu Kareli	$\left(\left\|x_i - x_j\right\| + \tau\right)^{\frac{1}{4}}$
Çoklu Kareli Tersi	$\left(\left\|x_i - x_j\right\| + \tau\right)^{-\frac{1}{4}}$

3.1.2 K En Yakın Komşu

K-En Yakın Komşu (KNN), ilk ortaya çıktığından beri örüntü sınıflandırmasında en iyi bilinen denetimli öğrenme algoritmalarından biri olmuştur [67, 68]. KNN'nin avantajları arasında basitlik, etkililik, sezgisellik ve birçok alanda rekabetçi sınıflandırma performansı gibi özellikler ye almaktadır [68]. KNN algoritması, yeni nesnelerin hedef çıktısını, bir eğitim kümesinin özellik

uzayındaki en yakın örneklerin veya birkaç yakın nesnenin sonucuna göre çıkarır. KNN algoritması, tüm makine öğrenimi algoritmalarının en basitlerinden biri olarak hem sınıflandırma hem de regresyon problemlerini yönetebilen parametrik olmayan bir yöntemdir: bir örnek, en yakın komşuları arasında en yaygın olan sınıfa yeni nesne atanarak, komşularının çoğunluk oyu tahmin edilerek sınıflandırılır [69, 70]. Daha büyük k değerleri, eğitim veri seti içindeki gürültülü noktaların etkilerini azaltmaya yardımcı olur ve k seçimi genellikle çapraz doğrulama yoluyla gerçekleştirilir. Parametrik yöntemlerde, yoğunluk tahmini veya sınıflandırma için, tüm girdi uzayı için geçerli bir model varsayılır. Sınıflandırmada normal bir yoğunluk varsayıldığında, sınıftaki tüm örneklerin bu aynı yoğunluktan çekildiği varsayılmaktadır. Ancak pratikte bu varsayım her zaman geçerli değildir ve büyük bir hataya sebebiyet verebilir. Eğer bu varsayımlar gerçekleştirilemiyorsa ve parametrik bir model oluşturulamıyorsa, yarı parametrik model kullanılabilir [71].

3.1.2.1 Uzaklık Metrikleri

Prasatha vd. (2017) göre mesafe, varlıkların ne kadar uzakta olduğunun sayısal bir açıklamasıdır [72]. İki x ve y vektörü arasındaki mesafe fonksiyonu, her iki vektör arasındaki mesafeyi negatif olmayan bir gerçek sayı olarak tanımlayan bir $d(x, y)$ fonksiyonudur. Bu işlev, aşağıdakileri içeren belirli sayıda özelliği karşılıyorsa bir metrik olarak kabul edilir [73]:

1. Negatif olmama: x ile y arasındaki mesafe her zaman sıfıra eşit ya da sıfırdan büyük bir değer olmalıdır [72].
$$d(x, y) \geq 0 \tag{3.20}$$
2. Ayırt edilemez noktaların belirlenmesi: Ancak ve ancak x, y'ye eşitse, x ve y arasındaki mesafe sıfır olabilir [72].
$$d(x, y) = 0 \text{ eğer } x = y \tag{3.21}$$
3. Simetri: x ile y arasındaki mesafe, y ile x arasındaki mesafeye eşit olmalıdır [72].
$$d(x, y) = d(y, x) \tag{3.22}$$

4. Üçgen eşitsizliği: Üçüncü bir z noktasının varlığı göz önüne alındığında, x ve y arasındaki mesafe her zaman x ve z arasındaki mesafe ile y ve z arasındaki mesafenin toplamından küçük veya ona eşittir [72].

$$d(x,y) \leq d(x,z) + d(z,y) \tag{3.23}$$

Mesafe [0,1] aralığında olduğunda, karşılık gelen benzerlik ölçüsünün $s(x,y)$ hesaplanması aşağıdaki gibidir:

$$s(x,y) = 1 - d(x,y) \tag{3.24}$$

Mesafe ölçümleri, yeni bir veri noktası ile mevcut eğitim veri seti arasındaki mesafeyi bulmaya yönelik bir yöntemdir.

- **Minkowski uzaklığı:**
  Bu mesafe ailesi, bu güç mesafesi için farklı p değerlerine karşılık gelen Minkowski mesafesinin özel durumları olan üç mesafe ölçüsü içerir. Lp normu olarak da bilinen Minkowski mesafesi genelleştirilmiş bir metriktir.

$$D_{M_{\text{ink}}}(x,y) = \sqrt[p]{\sum_{i=1}^{n} |x_i - y_i|^p} \tag{3.25}$$

  şeklinde tanımlanabilir. p burada pozitif bir değerdir. $p = 2$ olduğunda, bu denklem Öklit uzaklığına dönüşür. $p = 1$ olduğunda ise Manhattan uzaklığına dönüşür.
- **Öklid uzaklığı:** Pisagor teoreminin bir uzantısı olan bu mesafe ölçüsü, vektörlerdeki zıt değerler arasındaki farkların karelerinin toplamının kökünü temsil eder.

$$ED(x,y) = \sqrt{\sum_{i=1}^{n} |x_i - y_i|^2} \tag{3.26}$$

- **Chebyshev uzaklığı:** Chebyshev mesafesi, maksimum değer mesafesi [74], Lagrange [75] olarak da bilinir. Bu mesafe, herhangi bir boyutta farklı olan iki nesnenin farklı olarak tanımlanacağı durumlarda uygundur [76]. İki vektör arasındaki mesafenin herhangi bir koordinat boyutu boyunca farklarının en büyüğü olduğu bir vektör uzayında tanımlanan bir metriktir.

$$CD(x,y) = max_i \ |xi - yi| \tag{3.27}$$

- **Manhattan uzaklığı:** 19. yüzyıl Almanya'sında Hermann Minkowski tarafından kabul edilen $L1$ normu, Doğrusal mesafe veya Şehir bloğu mesafesi olarak da bilinen Manhattan uzaklığıdır. Bu uzaklık, vektörlerdeki zıt değerler arasındaki mutlak farkların toplamını temsil eder.

$$MD(x,y) = \sum_{i=1}^{n} |x_i - y_i| \tag{3.28}$$

- **Kosinüs Uzaklığı:** Açısal mesafe olarak da adlandırılan Kosinüs mesafesi, iki vektör arasındaki açıyı ölçen kosinüs benzerliğinden türetilir; burada Kosinüs mesafesi, kosinüs benzerliğini birden çıkararak elde edilir.

$$\text{Cos}\, D(x,y) = 1 - \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}} \tag{3.29}$$

- **Pearson Uzaklığı (PeaD):** Pearson mesafesi, iki vektör arasındaki doğrusal ilişkiyi ölçen Pearsons korelasyon katsayısından türetilir [77]. Bu uzaklık, Pearsons korelasyon katsayısı birden çıkarılarak elde edilir.

$$\text{PeaD}(x,y) = 1 - \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n} (x_i - \bar{x})^2 \sum_{i=1}^{n} (y_i - \bar{y})^2}} \tag{3.30}$$

- **Korelasyon Uzaklığı (CorD):** Korelasyon mesafesi, Pearson mesafesinin sıfır ile bir aralığında bir mesafe ölçüsü elde etmek için ölçeklendiği Pearson mesafesinin bir versiyonudur.

$$\text{Cor}\, D(x,y) = \frac{1}{2}\left(1 - \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n} (x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n} (y_i - \bar{y})^2}}\right) \tag{3.31}$$

3.1.2.2 K En Yakın Komşu Hesaplaması

$T = \{(x_i, y_i)\}_{i=1}^{N}$ eğitim veri kümesini göstersin. Burada $x_i \in \Re$ m-boyutlu öznitelik uzayında eğitim vektörüdür ve y_i de buna karşılık gelen sınıf etiketidir.

Bir x' sorgusu verildiğinde, bilinmeyen y' sınıfı iki adımda atanabilir.

İlk olarak, x sorgusu için k adet benzer etiketli hedef komşu kümesi tanımlanır. $T' = \{(x_i{}^{NN}, y_i{}^{NN})\}_{i=1}^{k}$ bu seti göstersin. $x_i \; ile \; x_i{}^{NN}$ arasındaki Öklid mesafesi $d(x', x_i^{NN})$ açısından artan bir sırada düzenlenmiştir:

$$d(x', x_i^{NN}) = \sqrt{(x' - x_i^{NN})^T (x' - x_i^{NN})} \tag{3.32}$$

İkincisi, sorgunun sınıf etiketi, en yakın komşularının çoğunluk oyu ile tahmin edilir.

$$y' = \arg\max_{y} \sum_{(x_i^{NN}, y_i^{NN}) \in T'} \delta(y = y_i^{NN}) \tag{3.33}$$

Burada y sınıf etiketidir. $y_i{}^{NN}$ ise k en yakın komşusu arasında i'inci en yakın komşunun sınıf etiketidir [68].

3.1.3 Lojistik Regresyon

Doğrusal modeller, bir bağımlı yanıt değişkeniyle ilişkiyi tanımlayan bir veya birden çok bağımsız değişkenden oluşur. Niteliksel veya niceliksel girdi özelliklerinin finansal, biyolojik veya sosyolojik veriler gibi tahmin edilmeye çalışılan bir hedef değişkenle eşlenmesi, etiketler biliniyorsa makine öğrenimi terminolojisinde denetimli öğrenme olarak bilinir. Diskriminant analizi için en yaygın kullanılan doğrusal istatistiksel modellerden biri lojistik regresyondur.

$$\pi_i = \beta_0 + \beta_1 X_1 + \cdots + \beta_n X_n \tag{3.34}$$

Lojistik regresyonun basitliği ve birlikte çalışabilirliği, bazen topluluk öğrenicileri veya destek vektör makineleri gibi diğer gelişmiş doğrusal olmayan modellerden daha iyi performans göstermesine yol açabilir. Ancak bağımlı değişkenin küçük bir örneklem büyüklüğünden alınması durumunda lojistik regresyon modelleri yetersiz kalmakta ve ikili yanıtlar için düşük performans göstermektedir [78]. İkili sınıflandırma veri tiplerini modellemek için bir dizi öğrenme algoritması uygulanabilir. Lojistik regresyon durumunda, bağımlı değişken niceldir. Lojistik regresyon için bağımlı değişken, ikili veya çok sınıflı bir değişkenin i. grubunda sınıflandırılma olasılıklarının logaritmasıdır [79].

Lojistik regresyon, bağımsızlık, açıklayıcı değişkenin bir alt popülasyonunun her seviyesindeki bağımlı değişkenlerin (logit) normal dağıldığı ve bağımlı değişken

ile açıklayıcı değişkenin tüm değerleri arasındaki sabit varyans gibi birkaç varsayım yapar. Sezgisel olarak, 0 ile 1 arasında sınırlanan çıktı sınıfları üzerinde sürekli bir olasılık dağılımı elde etmek için bağımlı değişkene bir dönüşüm uygulanır; bu dönüşüm "lojistik" veya "sigmoid" işlevi olarak adlandırılır, burada 'z' logitin, log olasılıklarına bölünmesi sonucu çıkan sonuca karşılık gelir [80]. Parametre tahminleri, diğer tüm açıklayıcı değişkenleri sabit tutarken, açıklayıcı değişkenlerden birinde bir birimlik artış veya azalış tarafından tahmin edilecek yanıt değişkeninin tahmin edilen log olasılıklarında bir artış veya azalma olup olmadığını bildirir.

$$\sigma(Z) = \frac{1}{1+\exp^{-z}} \tag{3.35}$$

Bir ikili yanıt için, lojistik regresyon modeli, Denklem (3.34) ve (3.35)'te gösterildiği gibi, girdi özelliklerinin doğrusal kombinasyonları ve karşılık gelen bir ağırlık (w) artı bir yanlılık terimi (b) üzerinden toplanarak ifade edilebilir:

$$p\left(y^{(i)} = 1 \mid x^{(i)}, w\right) = 1 - \frac{1}{1 + \exp^{\left(w^T x^{(i)} + b\right)}} \tag{3.36}$$

$$p\left(y^{(i)} = 0 \mid x^{(i)}, w\right) = 1 - \frac{1}{1 + \exp^{\left(w^T x^{(i)} + b\right)}} \tag{3.37}$$

Amaç, gradyan iniş veya stokastik gradyan iniş gibi optimizasyon teknikleri kullanılarak tanımlanan eğitim seti üzerinde negatif log olasılığının en aza indirildiği bir ağırlık seti bulmaktır [80]. Negatif log olasılığının en aza indirilmesi, aynı zamanda parametre tahmininin (pi) doğru sınıfı seçme olasılığının veya olasılığının en üst düzeye çıkarılması anlamına gelir. Temel gerçek etiketi ile tahmin edilen sınıf etiketi arasındaki farkı ölçen kayıp fonksiyonu, çapraz entropi olarak adlandırılır. Tahmin, gerçek etiketine çok yakınsa, kayıp değeri düşük olacaktır. Bununla birlikte, tahmin gerçek etiketten uzaksa, ortaya çıkan log kaybı daha yüksek olacaktır [81].

$$J(\theta) = -\frac{1}{m}\sum\ p_i \log(y_i) + (1 - p_i) \log(1 - y_i) \tag{3.38}$$

Girdi vektörü belirtilmek için $X = (X_1, \ldots, X_n)$ kullanıldığında, sınıf etiketi olan Y'nin koşullu olasılığı $P(Y = y \mid X = x)$ şeklinde X üzerindeki bir dizi gözleme

dayalı olarak hesaplanabilir. Ve Y, Bernoulli tipi bir rastgele değişken olduğundan, koşullu olasılık denklem (3.40) şeklinde ifade edilebilir:

$$p(y \mid x) = f(x)y\big(1 - f(x)\big)1 - y \tag{3.40}$$

burada $f(x)$, girdi vektörünün fonksiyonudur ve Bernoulli dağılımının parametresidir:

$$f(x) = p(y = 1 \mid x) = E(y \mid x)f(x) = p(y = 1 \mid x) = E(y \mid x) \tag{3.41}$$

Ayrıca, $f(x)$'in lojistik fonksiyon ve girdi özelliklerinin lineer dönüşümü kullanılarak hesaplanabileceğini varsayılmaktadır, burada θ lojistik regresyon sınıflandırıcısının parametre vektörüdür [82].

3.1.4 Rastgele Orman

Rastgele orman, her biri belirli bir sınıf için oy sağlayan karar ağaçlarından oluşan bir sınıflandırıcıdır. Çok sayıda ağacın rastgele bir ormanda birleştirilmesi, daha güvenilir tahminlere yol açabilirken, tek bir karar ağacı verilere gereğinden fazla uyum sağlayabilir [83].

Rastgele Orman, varyansı azaltmak için değiştirme ile yeniden örnekleme yöntemi olan önyükleme toplama fikri üzerine inşa edilmiştir. Rastgele Orman, bir tahmin yaparken terminal yaprak düğümlerinde ortalama (gerileme) veya çoğunluk oylarını hesaplamak (sınıflandırma) için birden çok ağaç kullanır. Karar ağaçları fikrini temel alan rastgele orman modelleri, 'n' sayıda ağaç büyüterek tek bir ağaca kıyasla tahmin doğruluğunda önemli gelişmeler sağlamıştır; eğitim setindeki her ağaç değiştirilmeden rastgele örneklenir [84]. Karar ağaçları, en üstteki düğümün, kökten terminal düğüme veya karar düğümüne ulaşılana kadar bir dizi karar düğümünde yinelemeli olarak bölünmüş ağacın kökü olarak kabul edildiği ağaç benzeri bir yapıdan oluşur. Karar ağacı algoritması, veri setini daha küçük alt kümelere bölen yukarıdan aşağıya "açgözlü" bir yaklaşımdır. Açgözlü algoritmalar, genellikle daha karmaşık olan en uygun çözüm yerine en basit çözümü alanlardır. Karar ağacının tepesi kök düğüm olarak bilinir ve bu en iyi tahmin değişkenine karşılık gelir. Her bir karar düğümünde öznitelikler iki dala ayrılır ve bu işlem son tahminin yapıldığı yaprak düğümlere ulaşılana kadar tekrarlanır. Her düğümde hangi özelliğin

bölüneceğini belirlemek için entropi hesaplanır. Entropi, alt küme verilerinin homojenliğini ölçer; entropi bire eşitse, sınıf etiketleri eşit olarak bölünürken sıfır entropi, örneğin tamamen homojen olduğu anlamına gelir (Denklem 3.42). Yalnızca iki etiketli ikili sınıflandırma durumunda olduğu gibi, ayırma sınıf etiketlerinin hepsinin 1 veya 0 olmasıyla sonuçlanırsa, entropi sıfır olacaktır. Benzer şekilde, etiketlerin yarısı 1 veya 0 ise, 1'den daha yüksek bir entropisi gözlenir. Entropi her değişken için hesaplanır ve ardından her değişkende bölünmeden önceki (örneğin ana düğüm) ve bölünmeden sonraki (örneğin alt düğüm) entropi arasındaki fark gerçekleştirilir. Sınıf etiketlerini bölümlere ayırmada en yararlı olan değişken, entropilerde en yüksek farkı verecektir. Bu fark aynı zamanda, modeldeki bir girdi özelliğinin ne kadar önemli olduğunu ölçmek için bir yöntem olan elde edilen bilgilere de atıfta bulunur.

$$\text{Entropi} = -\text{plog}\,2(p) - \text{qlog}\,2(q) \tag{3.42}$$

Ağaç benzeri bir öğrenme algoritması kullanmanın avantajlarından biri, niceliksel ve niteliksel girdi değişkenlerine ek olarak büyük veri kümelerinde eğitim modellerine izin vermesidir. Ek olarak, ağaç tabanlı modeller, gereksiz değişkenlere veya yüksek korelasyona sahip değişkenlere karşı bağışık olabilir ve bu da diğer öğrenme algoritmalarında aşırı uyuma yol açabilir. Ağaçların ayrıca modeli eğitirken ayarlanması gereken çok az parametresi vardır ve bir veri kümesindeki aykırı değerler veya eksik değerlerle nispeten iyi performans gösterir. Ancak, ağaçlar zayıf tahmin performansına eğilimlidir; karar ağaçlarının kendileri, sonuçta yüksek varyanslı sonuçlara yol açan bir eğitim setinde aşırı uyum gürültüsüne eğilimlidir. Başka bir deyişle, bu, modelin eğitildiği aynı verileri doğru bir şekilde tahmin edebileceği, ancak eğitim kümesindeki benzer modeller ve varyasyonlar olmadan veri kümelerinde aynı performansı gösteremeyeceği anlamına gelir. Tamamen büyümüş karar ağaçları bile aşırı uydurma konusunda pek tutarlı olmayabilir ve görünmeyen verilere pek iyi genelleme yapmaz; rastgele orman, ağaçtaki değerlerin rastgele, bağımsız bir örnek olduğu karar ağaçlarının bir kombinasyonunu veya "topluluğunu" kullanarak aşırı uyum sorununu çözer. Değiştirme ile rastgele örnekleme, torbalama olarak bilinir ve bu, üzerinde eğitim yapılacak farklı bir ağacın

oluşturulmasıyla sonuçlanır; 'n' ağaç sayısından elde edilen sonuçların ortalaması, varyansın azalmasına ve daha yumuşak bir karar sınırının oluşturulmasına neden olacaktır [80]. Örneğin, sınıflandırma için rastgele orman kullanılırken, her ağaç sınıf etiketinin olasılığını tahmin edecek, olasılıkların 'n' ağaç üzerinden ortalaması alınacak ve en yüksek verim, tahmin edilen sınıf etiketini verecektir. Torbalama veya önyükleme toplamaya ek olarak, karar sınırındaki varyansı daha da azaltmak için ağaçların tamamen ilişkisiz olması ve yalnızca önyükleme yönteminin olması gerekir [81].

Rastgele Ormanlar, genel olarak, karar ağacı sınıflandırıcılarının bir kombinasyonudur; öyle ki, her ağaç, bağımsız olarak örneklenen ve ormandaki tüm ağaçlar için aynı dağılıma sahip rastgele bir vektörün değerlerine bağlıdır. İki sınıfa ait N örneğinden ve F özelliklerinden oluşan bir X eğitim seti verildiğinde, bir Rastgele Orman çok yollu sınıflandırıcısı $\Theta(x)$, her bir ağacın bir tür rastgeleleştirme kullanılarak büyütüldüğü bir dizi karar ağacından oluşur; burada x bir giriş örneğidir. Her ağacın yaprak düğümleri, veri sınıfı etiketleri üzerindeki sonsal dağılımın tahminleriyle etiketlenir [85]. Her dahili düğüm, sınıflandırılacak veri alanını en iyi şekilde bölen bir test içerir. Yeni, görünmeyen bir örnek, her ağaca gönderilerek ve ulaşılan yaprak dağılımları toplanarak sınıflandırılır. Her ağaç aşağıda açıklanan şekilde büyütülür:

Eğitim setindeki vaka sayısı N ise, orijinal verilerden rastgele ancak değiştirilerek N vaka örneği alınır. Bu örnek, ağacı büyütmek için eğitim seti olacaktır.

- F giriş özellikleri varsa, her düğümde, m değişkenleri F'den rastgele seçilecek ve bu m'deki en iyi bölme düğümü bölmek için kullanılacak şekilde bir $m < F$ sayısı belirtilir. Orman büyümesi sırasında m değeri sabit tutulur.
- Her ağaç mümkün olduğu kadar büyütülür. Bu nedenle herhangi bir budama işlemi uygulanmaz [83].

3.1.5 Artırma Algoritmaları

Artırma, etiketli eğitim örnekleri verildiğinde, bir temel veya zayıf sınıflandırıcı üreten bir temel veya zayıf öğrenme algoritmasının geçerli olduğunu varsayar. Artırmanın amacı, zayıf öğrenme algoritmasını bir alt program gibi tekrar tekrar

çağrılabilen, ancak iç kısımları gözlemlenemeyen veya manipüle edilemeyen bir "kara kutu" olarak ele alırken performansını iyileştirmektir. Varsayım, zayıf sınıflandırıcıların, hata oranlarının, her tahmini rastgele bir tahmin olan bir sınıflandırıcıdan biraz daha iyi olmasının aslında önemsiz olmadığıdır. Bu nedenle, istenmeyen e-posta filtreleme örneğindeki pratik kurallar gibi, zayıf sınıflandırıcılar sert ve kısmen yanlış olabilir, ancak tamamen önemsiz ve bilgisiz olmayabilir. Temel öğrenicinin, üzerinde eğitildiği örnekler üzerinde rastgele tahminden biraz daha iyi olan zayıf bir hipotez ürettiğine dair bu varsayım, zayıf öğrenme varsayımı olarak adlandırılır ve güçlendirme çalışmasının merkezinde yer alır.

Diğer makine öğrenmesi algoritmaları gibi, artırma algoritması da her x_i X'ten bir örnek olacak şekilde $(x_1, y_1), \ldots, (x_m, y_m)$ şeklinde bir girdi vektörü alır ve her yi sınıf etiketini belirtir. Bir hızlandırma algoritmasının verilerden öğrenmesinin tek yolu, temel öğrenme algoritmasına yapılan çağrılardır. Bununla birlikte, temel öğrenici, her zaman aynı eğitim verileriyle tekrar tekrar çağrılırsa çıkacak sonuçta ekstrem bir durumdan söz edilemeyebilir; bunun yerine, aynı veya hemen hemen aynı temel sınıflandırıcının tekrar tekrar üretilmesi beklenebilir, böylece temel öğreniciyi yalnızca bir kez çalıştırmaktan çok az şey kazanılır. Bu, güçlendirme algoritmasının, eğer temel öğreniciyi geliştirmek istiyorsa, kendisine beslediği verileri bir şekilde manipüle etmesi gerektiğini gösterir. Aslında, güçlendirmenin arkasındaki ana fikir, temel öğrenen için eğitim setlerini, her çağrıldığında veriler hakkında yeni bir şey çıkarım yapmaya zorlayacak şekilde seçmektir. Bu, önceki temel sınıflandırıcıların performansının çok zayıf, hatta normal zayıf performanslarından bile daha düşük olmasını makul bir şekilde beklenebileceği eğitim setleri seçilerek başarılabilir.

Bu başarılabilirse, o zaman temel öğrenicinin öncekilerden önemli ölçüde farklı olan yeni bir temel sınıflandırıcı çıkarması beklenebilir. Bunun nedeni, temel öğrenicinin zayıf ve vasat bir öğrenme algoritması olarak düşünülmesine rağmen, yine de önemsiz olmayan tahminler yapan sınıflandırıcılar üretmesinin beklenmesidir [86].

3.1.6 Gradyan Artırma Algoritmaları

Gradyan artırma algoritmaları, regresyon için artırma benzeri bir algoritmadır [87].

$D = \{\mathrm{x}_i, y_i\}_{\mathrm{i}=1}^{N}$ şeklinde bir test veri seti verildiğinde, gradyan artırmanın amacı $F^*(\mathrm{x})$ fonksiyonundan türetilmiş olan $\hat{F}(\mathrm{x})$ için bir yaklaşım bulmaktır, bu fonksiyon ayrıca belirli bir kayıp fonksiyonunun beklenen değerini $L\big(y, F(\mathrm{x})\big)$ en aza indirerek x örneklerini, çıktı değerleri olan y ile eşler.

Gradyan artırma, fonksiyonların ağırlıklı toplamı olarak $F^*(\mathrm{x})$'in toplamsal bir yaklaşımını denklem (3.43)'te gösterildiği gibi oluşturur:

$$F_m(\mathrm{x}) = F_{m-1}(\mathrm{x}) + \rho_m h_m(\mathrm{x}) \tag{3.43}$$

Burada ρ_m , $h_m(\mathrm{x})$'in $m.$ fonksiyonunun ağırlığıdır. Bu işlevler, topluluk modelleridir (örneğin, karar ağaçları). Yaklaşım, yinelemeli olarak oluşturulur. İlk olarak, sabit bir $F^*(\mathrm{x})$ yaklaşımı denklem (3.44)'te ifade edilen şekilde elde edilir:

$$F_0(\mathrm{x}) = \arg\min_{\alpha} \sum_{i=1}^{N} L(y_i, \alpha) \tag{3.44}$$

Sonraki modellerin minimize edilmesi beklenir, buna ilişkin denklem (3.45)'te yer almaktadır:

$$\big(\rho_m, h_m(\mathrm{x})\big) = \arg\min_{\rho,h} \sum_{i=1}^{N} L\big(y_i, F_{m-1}(\mathrm{x}_i) + \rho h(\mathrm{x}_i)\big) \tag{3.45}$$

Bununla birlikte, optimizasyon problemini doğrudan çözmek yerine, her h_m, F^* için gradyan iniş optimizasyonunda açgözlü bir adım olarak görülebilir. Bunun için her model h_m,yeni bir veri kümesi olan $D = \{\mathrm{x}_i, r_{mi}\}_{i=1}^{N}$ üzerinde eğitilir; burada sözde artıklar, r_{mi}, denklem (3.46)'da gösterildiği şekilde hesaplanır:

$$r_{mi} = \left[\frac{\partial L\big(y_i, F(\mathbf{x})\big)}{\partial F(\mathbf{x})}\right]_{F(\mathbf{x})=F_{m-1}(\mathbf{x})} \tag{3.46}$$

ρ_m değeri daha sonra bir satır arama optimizasyon problemini çözerek hesaplanır.

Yinelemeli süreç düzgün bir şekilde düzenlenmezse, bu algoritma aşırı uyumdan dolayı zarar görebilir [88]. Bazı kayıp fonksiyonları için (örneğin ikinci dereceden kayıp), eğer model h_m sözde kalıntılara mükemmel bir şekilde uyarsa, ardından bir sonraki yinelemede sözde kalıntılar sıfır olur ve süreç erken sona erer. Gradyan artırmanın eklemeli sürecini kontrol etmek için birkaç düzenlileştirme hiper parametresi dikkate alınır. Gradyan artırmayı düzenli hale getirmenin doğal yolu, $v = (0,1.0]$ ile $F_m(\mathrm{x}) = F_{m-1}(\mathrm{x}) + \nu\rho_m h_m(\mathrm{x})$ her gradyan düzgün adımını azaltmak için büzülme uygulamaktır. v değeri genellikle 0.1 olarak ayarlanır. . Bununla birlikte, yöntem olarak eğitilen modellerin karmaşıklığı sınırlanırsa daha fazla düzenleme yapılabilir. Karar ağaçlarını bu duruma uyarlarsak, ağaçların derinliğini veya bir düğümü bölmek için gereken minimum örnek sayısını sınırlayabiliriz. Rastgele ormanın aksine, gradyan artırmadaki bu hiper parametreler için varsayılan değerler, ağaçların ifade gücünü sert bir şekilde sınırlayacak şekilde ayarlanmıştır. Son olarak, gradyan artırmanın farklı versiyonlarında yer alan başka bir hiper parametre ailesi de temel öğrenicileri rastgele hale getirenlerdir ve bu da değiştirme olmadan rastgele alt örnekleme gibi topluluğun genelleştirilmesini daha da geliştirebilir.

Gradyan artırma yönteminin özellikleri aşağıdaki gibidir:

- Öğrenme oranı veya v büzülmesi.
- Ağacın maksimum derinliği (Rastgele ormanda oluşturulan ağaçlarla aynı anlama gelir.)
- Rastgele örneklerin boyutu için alt örnekleme oranı (Rastgele ormanın aksine, bu genellikle değiştirme olmadan gerçekleştirilir)
- En iyi bölmeyi ararken dikkate alınması gereken özelliklerin sayısı (rastgele ormandaki gibi.)
- Bir iç düğümü bölmek için gereken minimum örnek sayısı (rastgele ormanda olduğu gibi.) [89]

3.1.7 Ekstrem Gradyan Artırma (XGBOOST)

XGBoost, yüksek oranda ölçeklenebilir olacak şekilde tasarlanmış, gradyan artırmaya dayalı bir karar ağacı topluluğudur [90, 91]. XGBoost, gradyan artırma makinesi (GBM) için en popüler artırma ağacı algoritmalarından biridir. Problem çözmedeki yüksek performansı ve özellik mühendisliği için minimum gereksinim nedeniyle endüstride yaygın olarak kullanılmaktadır. Derin öğrenme algoritmalarıyla karşılaştırıldığında, XGBoost'un CPU üzerinde çalışan küçük veri kümeleri için kullanımının daha kolay olduğu kabul edilmektedir [92]. Gradyan artırmaya benzer şekilde, XGBoost, bir kayıp işlevini en aza indirerek amaç işlevinin ek bir genişlemesini oluşturur. XGBoost'un temel sınıflandırıcılar olarak yalnızca karar ağaçlarını baz aldığı düşünülürse, ağaçların karmaşıklığını kontrol etmek için kayıp fonksiyonunun bir varyasyonu kullanılır.

$$L_{xgb} = \sum_{i=1}^{N} L\left(y_i, F(\mathrm{x}_i)\right) + \sum_{m=1}^{M} \Omega(h_m) \tag{3.47}$$

$$\Omega(h) = \gamma T + \frac{1}{2}\lambda \parallel w \parallel^2 \tag{3.48}$$

Burada T ağacın yaprak sayısıdır ve w, yaprakların çıktı puanıdır. Bu kayıp fonksiyonu, bir ön budama stratejisine yol açan karar ağaçlarının bölünme kriterine eklenebilir. Daha yüksek γ değerleri daha basit ağaçlarla sonuçlanır. γ değeri bir düğümü bölmek için gereken minimum kayıp azaltma kazancını kontrol eder. XGBoost'taki ek bir düzenlileştirme hiper parametresi, ek genişletmedeki adım boyutunu azaltan büzülmedir. Son olarak, ağaçların karmaşıklığı, ağaçların derinliği vb. gibi diğer stratejiler kullanılarak da sınırlandırılabilir. Ağaç karmaşıklığının azaltılmasının ikinci yararı, modellerin daha hızlı eğitilmesine olanak sunması ve diğer yöntemlere göre daha az depolama alanı gerektirmesidir. Hem fazla uyumu azaltmak hem de eğitim hızını artırmak için XGBoost'ta rastgeleleştirme teknikleri de uygulanmaktadır [91].

3.1.8 LightGBM

LightGBM gradyan artırmayı uygulayan ve bazı değişkenler öneren kapsamlı bir kitaplıktır [93]. LightGBM'de gradyan artırmanın uygulanması, özellikle,

XGBoost'ta olduğu gibi, özniteliklerin histogramının önceden hesaplanmasına dayanan hesaplama açısından verimli bir algoritma oluşturmaya odaklanmıştır. Kitaplık ayrıca bu modelin çok çeşitli senaryolarda çalışmasına izin veren onlarca öğrenme hiper parametresi içerir: Uygulama hem GPU'da hem de CPU'da çalışır, temel gradyan artırma olarak çalışabilir ve birçok rastgeleleştirme türüne sahiptir (sütun rastgeleleştirme, sütun rastgeleleştirme, önyükleme alt örneklemesi vb.). LightGBM uygulaması ayrıca, ağaç içeren topluluklar oluşturmak için eğitim setleri oluşturmak için kullanılan bir alt örnekleme tekniği olan Gradient-based One-Side Sampling (GOSS) ve Exclusive Feature Bundling (EFB)) [93] gibi çoğunlukla yeni özellikler sağlar. Bu yöntemin amacı, daha yüksek eğime sahip örnekler olarak tanımlanan sınıflandırma belirsizliği daha yüksek olan örneklerin anlamlılığını arttırmaktır. GOSS seçeneği belirtildiğinde, öğrencilerin ilk eğitim setleri, en büyük gradyanlı (a) örneklerin üstünden ve en küçük gradyanlı (b) örneklerin rastgele bir örnekleme fraksiyonundan oluşturulur. Orijinal dağılımdaki değişikliği telafi etmek için, bilgi kazancını hesaplarken, düşük gradyan grubundaki örnekler denklem (3.49)'daki şekilde ağırlıklandırılır:

$$\frac{\frac{1-a}{b(1-a)}}{b} \tag{3.49}$$

EFB tekniği, seyrek özellikleri tek bir özellikte toplar. Bu özellikler aynı anda sıfır olmayan değerlere sahip olmadığında herhangi bir bilgi kaybı olmadan yapılabilir [91].

3.1.9 Saf Bayes Sınıflandırıcısı

Bayes teoremi, toplam olasılık teoreminden türetilmiştir. Bu sebeple, Bayes teoremine geçmeden önce olasılık teoremi açıklanacaktır.

A ve B olayları için birleşim kuralı denklem (3.50)'deki gibi açıklanabilir:

$$P(A \cup B) = P(A) + P(B) - P(A \cap B) \tag{3.50}$$

Örnek uzayın ayrık n olaya bölünebileceği varsayılır:

1 $A_1 \cup A_2 \cup \ldots \cup A_n = \Omega$

2 $A_i \cap A_j = \emptyset$, $i \neq j$ için

3 $A_i \neq \emptyset$

Bu durumda B olayı denklem (3.51)'deki gibi ifade edilebilir:

$$B = (B \cap A_1) \cup (B \cap A_2) \cup \cdots \cup (B \cap A_n) \tag{3.51}$$

Böylece toplam olasılık teoremi denklem (3.51)'deki gibi elde edilir:

$$P(B \mid A_1)P(A_1) + \cdots + P(B \mid A_n)P(A_n) = \sum_{i=1}^{n} P(B \mid A_i)P(A_i) \tag{3.52}$$

Bu denklem yeniden yazılacak olursa denklem (3.53) elde edilir:

$$P(B) = P(B \mid A)P(A) + P(B \mid A^c)P(A^c) \tag{3.53}$$

t$|A| \neq 0$ ve $|B| \neq 0$ durumu varsayıldığında, aşağıdaki (3.54) denklemi ifade edilebilir:

$$P(A \mid B) = \frac{P(B \mid A)(A)}{P(B)} \tag{3.54}$$

Bu, Bayes teoreminin en basit formülüdür. Bayes teoremi, çıkarımsal istatistikler ve birçok gelişmiş makine öğrenimi modeli için temel öneme sahiptir. Bayesci akıl yürütme, yeni kanıtlar ışığında hipotezlerin olasılığını güncellemek için mantıksal bir yaklaşımdır ve bu nedenle bilimde çok önemli bir rol oynar [94]. Saf Bayes, niteliklerin sınıfa göre koşullu olarak bağımsız olduğuna dair güçlü bir varsayımla birlikte Bayes kuralını kullanan basit bir öğrenme algoritmasıdır. Bağımsızlık genellikle zayıf bir varsayım olsa da pratikte saf Bayes çok yönlü sınıflandırıcılarla iyi rekabet eder. Hesaplama verimliliğine sahip olan Saf Bayes diğer özellikleri ile birleştiğinde bu durum Saf Bayes'in pratikte geniş çapta uygulanmasına yol açar. Saf Bayes, x nesnesi verilen her y sınıfının sonsal olasılığını $P(y \mid x)$ tahmin etmek amacıyla örnek verilerdeki bilgileri kullanmak için bir mekanizma sağlar. Bu tür tahminlere sahip olunduğunda, sınıflandırma veya diğer karar destek uygulamaları için kullanım sağlanabilir [95]. Bayes sınıflandırıcıları, öznitelik vektörü tarafından tanımlanan belirli bir örneğe en olası sınıfı atar. Bu tür sınıflandırıcılarda öğrenme, özelliklerin verilen sınıftan

bağımsız olduğu, yani $P(\mathrm{X} \mid C) = \prod P(X_i \mid C)_{i=1}^{n}$, $\mathrm{X} = (X_1, \cdots, X_n)$ bir öznitelik vektörü ve C'nin bir sınıf olduğu varsayılarak büyük ölçüde basitleştirilebilir [96].

Saf Bayes'in özellikleri aşağıdaki biçimde verilebilir:

- Hesaplama verimliliği: Eğitim süresi hem eğitim örneği sayısı hem de öznitelik sayısı açısından doğrusaldır ve sınıflandırma süresi, öznitelik sayısı açısından doğrusaldır ve eğitim örneği sayısından etkilenmez.
- Düşük varyans: Saf Bayes aramayı kullanmadığı için, yüksek yanlılık olasılığına rağmen düşük varyansa sahiptir.
- Artımlı öğrenme: Saf Bayes, eğitim verilerinden türetilen düşük dereceli olasılık tahminlerinden çalışır. Bunlar, yeni eğitim verileri elde edildikçe kolayca güncellenebilir.
- Sonsal olasılıkların doğrudan tahmini.
- Gürültü karşısında sağlamlık: Saf Bayes her zaman tüm tahminler için tüm öznitelikleri kullanır ve bu nedenle sınıflandırılacak örneklerde gürültüye nispeten duyarsızdır. Olasılıkları kullandığı için, eğitim verilerindeki gürültüye de nispeten duyarsızdır.
- Eksik değerler karşısında sağlamlık: Saf Bayes her zaman tüm tahminler için tüm nitelikleri kullandığından, bir özellik değeri eksikse, diğer özelliklerden gelen bilgiler kullanılmaya devam edilir ve bu da performansta hafif bir düşüşe neden olur. Ayrıca, olasılık çerçevesi nedeniyle eğitim verilerindeki eksik özellik değerlerine karşı nispeten duyarsızdır [97].

Bir veri setinin p nitelikten oluşan n örnek (veya durum) x_i, $i = 1, \ldots, n$ içerdiği varsayılır, yani, $x_i = (x_{i1}, \ldots x_{ip})$'dir. Her örneğin bir (ve yalnızca bir) $y \in \{y_1, y_2, \ldots, y_m\}$ sınıfına ait olduğu varsayılır. Makine öğrenimindeki çoğu tahmine dayalı model, her x_i örneği için s ile gösterilen bir sayısal puan üretir. Bu puan, o vakanın y_j sınıfındaki sınıf üyeliğinin derecesini ölçer. Veri seti yalnızca pozitif ve negatif örnekler içeriyorsa, $y \in \{0, 1\}$, o zaman tahmine dayalı bir model bir sıralayıcı veya sınıflandırıcı olarak kullanılabilir. Sıralayıcı, örnekleri pozitif olma olasılığı en yüksekten en aza doğru sıralamak için puanları kullanır. Sıralama

puanı s(x) üzerinde $\{s(x) \geq t\} = 1$ olacak şekilde bir t eşiği ayarlayarak, sıralayıcı (net) bir sınıflandırıcı haline gelir [94].

Saf Bayes öğrenimi, bir örneğe bir son sınıf olasılığı atayan bir Bayes olasılık modelinin inşasına atıfta bulunur:

$$P(Y = y_j \mid X = \mathbf{x}_i) \tag{3.55}$$

Basit Saf Bayes sınıflandırıcısı, bir sınıfa bir örnek atamak için bu olasılıkları kullanır. Bayes teoremini (Denklem 3.54) uygulayarak ve gösterimi biraz basitleştirerek, denklem (3.56) yazılabilir:

$$P(y_j \mid \mathrm{x}_i) = \frac{P(\mathrm{x}_i \mid y_j)P(y_j)}{P(\mathrm{x}_i)} \tag{3.56}$$

Buradaki pay x_i ve y_j'nin ortak olasılığıdır. Pay, bu nedenle aşağıdaki gibi yeniden yazılabilir; burada, basitlik için i indeksi çıkarılarak sadece x kullanılacaktır:

$$P(\mathrm{x} \mid y_j)P(y_j) = P(\mathrm{x}, y_j) = P(x_1, x_2, \ldots, x_p, y_j) \tag{3.57}$$

$$= P(x_1 \mid x_2, x_3, \ldots, x_p, y_j)P(x_2, x_3, \ldots, x_p, y_j) \tag{3.58}$$

$$= P(x_1 \mid x_2, x_3, \ldots, x_p, y_j)P(x_2 \mid x_3, x_4, \ldots, x_p, y_j)P(x_3, x_4, \ldots, x_p, y_j) \tag{3.59}$$

$$= P(x_1 \mid x_2, x_3, \ldots, x_p, y_j)P(x_2 \mid x_3, x_4, \ldots, x_p, y_j) \cdots P(x_p \mid y_j)P(y_j) \tag{3.60}$$

Her x_i'nin birbirinden bağımsız olduğu varsayılsın. Bu, çoğu pratik uygulamada açıkça ihlal edilen ve bu nedenle saf olan güçlü bir varsayımdır. Saf Bayes adı buradan gelmektedir. Bu varsayım, örneğin $P(x_1 \mid x_2, x_3, \ldots, x_p, y_j) = P(x_1 \mid y_j)$ olduğunu ima eder. Böylece, x ve y_j'nin ortak olasılığı aşağıdaki gibidir:

$$\begin{aligned} P(\mathrm{x} \mid y_j)P(y_j) &= P(x_1 \mid y_j) \cdot P(x_2 \mid y_j) \cdots P(x_p \mid y_j)P(y_j) \\ &= \prod_{k=1}^{p} P(x_k \mid y_j)P(y_j) \end{aligned} \tag{3.61}$$

Denklem (3.61)'i, denklem (3.60)'ta yerine koyduğumuzda, denklem (3.62) elde edilir

$$P(y_j \mid \mathrm{x}) = \frac{\prod_{k=1}^{p} P(x_k|y_j)P(y_j)}{P(\mathbf{x})} \tag{3.62}$$

Denklem (3.62)'de payda $P(x)$ sınıfa bağlı değildir, $y_j$ve y_1 için aynı sınıfı ifade eder. $P(x)$ burada ölçeklendirme faktörü görevi görür ve $P(y_j \mid \mathrm{X})$ sonsal olasılığının uygun şekilde ölçeklenmesini sağlar, bu da 0 ile 1 arasındadır.

Kesin bir sınıflandırma kuralıyla, yani her örneği tam olarak bir sınıfa atayan bir kuralla ilgilenildiğinde, basitçe her sınıf için payın değeri hesaplanabilir ve bu değerin maksimum olduğu sınıfı seçilebilir. Bu kurala en büyük artçıl kestirimi adı verilmektedir. Ortaya çıkan "kazanan" sınıf, maksimum a posteriori (MAP) sınıfı olarak da bilinir ve x örneği için aşağıdaki gibi $\hat{y}$ olarak denklem (3.63)'teki gibi hesaplanır:

$$\hat{y} = \underset{y_j}{\operatorname{argmax}} \prod_{k=1}^{p} P(x_k \mid y_j)P(y_j) \tag{3.63}$$

Bu modele Saf Bayes sınıflandırıcısı denir. Pozitif ve negatif bir sınıf içeren görevleri sıralarken, genellikle bir modelin bir sınıfın durumlarını diğer sınıfın durumlarına göre ne kadar iyi sıraladığı daha çok önem arz eder [98]. Tahmini sınıf sonsal olasılıkları, doğal sıralama puanlarıdır. Toplam olasılık teoreminin uygulandıktan sonraki denklemi aşağıdaki gibidir:

$$P(y_j \mid \mathrm{x}) = \frac{\prod_{k=1}^{p} P(x_k \mid y_j)P(y_j)}{\prod_{k=1}^{p} P(x_k \mid y_j)P(y_j) + \prod_{k=1}^{p} P(x_k \mid \gamma_j^c)P(\gamma_j^c)} \tag{3.64}$$

3.2 Entropi Tabanlı Sistemler

3.2.1 Bilgi Teorisi

Rastgelelik, genellikle olasılıksal olarak incelenir, bir dizide farklı basamakların oluşma sıklıkları analiz edilir ve ortalama, varyans, asimetri, basıklık vb. ölçütler kullanılır. Bilgi teorisi, klasik yöntemlerin çözemeyeceği bazı problemleri çözmek için tasarlanmıştır. Rastgele seri olgusunu incelemek isteyen Chaitin yazı tura deneyini önermiştir. Bu deneye göre madeni para 20 kez atılır ve tura gelirse 1

değer, yazı gelirse 0 değeri işlenir. Olasılık açısından, bu deney iki diziyi aynı olasılıkla üretebilme imkanına sahiptir. Bu olasılık ayrıca 2 üzeri 20'de 1 olarak ifade edilebilir. Aşağıdaki A ve B dizileri rastgele diziler olarak verilmiştir.

A: 01010101010101010101

B: 01101000110111100010

İki serinin aynı olasılıkla rastgele üretildiği ve aynı ortalama ve varyans değerlerine sahip olduğu doğru olsa da, A serisinde açıklanan kalıp için gelecek sayı için tahmin yürütmek daha kolaydır. B serisine bakıldığında, A serisi gibi belirli bir kalıba sahip olmadığından, gelecekteki sayıyı tahmin etmek zorlaşır. Bu durumda daha çok bilgi gerekmektedir. Rastgelelik analiz edilmek istenirse, matematiksel ifade ile bu garanti edilemez ve rastgelelik analizi, moment istatistikleri yöntemlerinin bir alanıdır. Önceki iki diziden herhangi birini karıştırırsak, moment istatistiklerini kullanarak aynı değerleri elde ederiz, terimleri $\sum_{i=1,\dots,n}(\mathrm{x}_i - \bar{\mathrm{x}})^a$ şeklindedir, burada $\boldsymbol{a}$ hesaplanan momentin sırasıdır. Bu sınırlamanın kaynağı, farklı sıralardaki anları analiz etmeye yönelik klasik olasılık yaklaşımının, bir serinin rastgeleliğini değil, serinin üretim sürecinin rastgeleliğini analiz etmesi gerçeğinde yatmaktadır. Bir seride yer alan değerlerin ya da sayıların ölçüleri, verilerin üretilmesine ilişkin süreci incelemek için kullanılsa da üretim sıralamasının bu değerlerin ya da sayıların üzerinde etkisinin olmadığı söylenebilir. Bu sebeple rastgelelik analizi yapılmak istenirse, bilgi teorisi ve entropi bu konuda en kök analizleri sunabilir. Sadece bu alanda değil, fizik alanında da entropi kavramının bir anlamı vardır. Fizikte entropi, kökenleri Clausius ve Boltzmann tarafından tanımlanmış termodinamik ve istatistiksel mekanikte ifade edilen bir değerdir. İstatistik alanında da karmaşıklığın ölçüsü olarak entropi kullanılır. Özünde, entropi bilgi miktarını nicelleştirdiği için sistemdeki rastgelelik derecesini de ölçer. Bilgi teorisinin ilk başlarda tasarlanma amacı, gürültülü bir kanal aracılığıyla mesaj gönderme sürecini analiz etmek ve mesajın düşük hata olasılığıyla nasıl yeniden yapılandırılacağını anlamaktır. Bilgi teorisinin temel fikirlerinden biri, bir mesajda yer alan bilgilerin nicel olarak ne ifade ettiğinin ölçülebilmesidir. Bu miktarın nasıl ölçüleceğini incelerken, Bell laboratuvarlarında o esnada çalışma

yapmakta olan iki araştırmacı [99, 100], logaritmik fonksiyonun sahip olduğu özelliklerden dolayı bilgi miktarını ölçmek için iyi bir seçim olduğu sonucuna varmıştır. Böylelikle bu durum, telgrafla yapılan iletişimlerin analizini inceleyen araştırmacıların bu türden bilgileri ölçmek için $H = nlogS$ formülünü bulmalarına yol açmıştır. S burada olası sembollerin sayısını belirtirken, n iletilen sembollerin sayısıdır. Shannon, entropiyi tek bir rastgele değişkenin bilgisinin ölçüsü olarak önererek bu formülasyonu bu kısıtlamayı atlayacak şekilde genişletmiştir. Bu bağlamda rastgele bir değişken için olasılıksal açıdan değerlendirme yapıldığında iki sonuca ulaşılabilir: değişkenin olası sonuçlarının her birinin belirli bir oluşma olasılığı vardır ve sonuçla ilişkili değer olasılığa bağlıdır. X, $\{x_1, \dots, x_n\}$ serisinden değerleri alabilir ve $x \in X$ olduğu durumda $p(x)$ verilen değerlerle ilişkili olasılık anlamına gelir, bu durumda entropi şu şekilde tanımlanır:

$$H(X) = -\sum_{x \in X} p(x) \log p(x) \tag{3.65}$$

Logaritma tabanı olarak genellikle 2 seçilir, ancak başka bir taban seçilmek istendiğinde $H_b(X) = \log_b(a)[H_a(X)]$ eşitliği kullanılabilir. Ayrıca X değeri sıfır olduğunda, $lim_{p \to 0} p \cdot log_2(p) = 0$ eşitliğinden dolayı, entropi sıfıra eşittir:

$$H(0) = 0 \tag{3.66}$$

Bu büyüklük, matematiksel beklenti olarak belirlenen bir olayın ortalama veya beklenen bilgisi olan $E\left[\log\left(\frac{1}{p(X)}\right)\right]$ formülünü belirler. Entropinin her bir sembolün ortaya çıkma olasılığına dayandığını anlamak, diğer entropi metotlarını anlamak için önemlidir, yani entropi, serilerin değerlerinin bir fonksiyonu değil, olasılıklarının bir fonksiyonudur. En genel anlamıyla, araştırmacı her bir sembolün gerçek olasılık dağılımını önemsemez ve olasılığı tahmin etmek için veri serisindeki değerlerin oluşma sıklığını hesaplar [101].

3.2.2 Permütasyon Entropisi

Permütasyon entropisi, zaman serilerini komşu değerlere dayanarak ölçen bir karmaşıklık ölçüsüdür. Christoph Bandt ve Bernd Pompe tarafından 2002 yılında

tanıtılmıştır [102]. Verilen bir zaman serisindeki karmaşıklığı ölçmek, veri analizinde önemli bir sorun teşkil eder [103]. Genellikle karmaşıklığı, düzensizliği ölçmek, bazı entropi ölçümleri kullanılarak gerçekleştirilebilir. Bir zaman serisinin permütasyon entropisi, zaman serisindeki sıralı desenlerin dağılımının Shannon entropisidir. Sıralı vektör türlerini tanımlayan bu tür sıralamalar, permütasyonlar ile kodlanır [104]. Klasik metotların çoğu, bir zaman serisindeki değerlerin zamansal sıralamasını göz ardı eder (örneğin klasik Shannon entropisi) ve büyük boyutlu bir zaman serisi gerektirebilir. Bandt ve Pompe, 2002 yılında basit ve güçlü bir sistem olmakla birlikte bu sorunların üstesinden gelebilen permütasyon entropisini geliştirdiler [103]. Permütasyon entropisi, ekonofizik, biyomedikal, arıza teşhisleri, sinyal analizi, borsa analizi gibi alanlarda kullanılabilir. Diğer entropi yaklaşımlarının aksine permütasyon entropisi kaotik ve gürültülü zaman serileri için kolay hesaplama, düşük hesaplama maliyeti ve sağlamlılık gibi avantajlara sahiptir [105]. Shannon ve diğer klasik ölçütler zaman serilerinin değerleri arasındaki zamansal ilişkileri ihmal eder, böyle olunca süreçte mevcut olan yapı ve olası zamansal desenler hesaba katılmaz. Diğer bir deyişle iki zaman serisi $x_1 = \{0,0,1,1\}$ ve $x_2 = \{0,1,0,1\}$ olarak tanımlanırsa, bu $S[P(x_1)] = S[P(x_2)]$ anlamına gelmektedir [98]. Permütasyon entropisinin arkasındaki fikir, desenlerin aynı oluşum olasılığına sahip olamayacağı ve bu nedenle olasılığın, temeldeki sistemle ilgili bilgileri açığa çıkarabileceğidir. Aykırı bir durum ise forbidden pattern ile yani zaman serilerinde hiç görülmeyen desenler kullanılarak temsil edilmesidir [106].

$X = \{x_i\} \quad i = 1, \dots, N$ denklemi verilsin. Her sembol sırasının olasılık dağılımları $p_1, p_2, \dots, p_k$ ile gösterilsin, $\sum_{j=1}^{k} p_j = 1$ ve $x_i,\ i = 1,2, \dots, N$ zaman serisinin permütasyon entropisi, k her sembol sırası için, Shannon entropisi olarak tanımlanabilir.

Bandt ve Pompe'ye göre permütasyon entropisi hesabı denklem (3.67)'deki gibi yapılır [103]:

$$H_n = \sum_{j=1}^{n!} p'_j log_2(p'_j) \tag{3.67}$$

Her sembol için permütasyon entropisi ise denklem (3.68)'de verilmiştir:

$$h_n = \frac{-1}{n-1}\sum_{j=1}^{n!} p'_j log_2(p'_j) \tag{3.68}$$

m boyutlu gömülü vektör i zamanında tanımlanırsa [107]

$$x_i^m = \left[x(i), x(i+\tau), \dots, x\big(i+(m-1)\big)\tau\right] \tag{3.69}$$

m gömülü boyut ve τ zaman gecikmesi olarak ele alınsın. x_i^m aşağıdaki koşul ve kısıtları sağlarsa denklem (3.70)'te yer alan duruma göre,

$$x(t+r_0\tau) \leq x(t+r_1\tau) \leq \cdots \leq x(t+r_{m-1}\tau) \tag{3.70}$$

$0 \leq r_i \leq m$ ve $r_i \neq r_j$, $\pi_{r_0}\pi_{r_1\dots}\pi_{r_{m-1}}$ permütasyonuna sahip olduğunu söyleyebiliriz. Tüm olası permütasyonlar, aynı olasılıkta göründüğünde maksimum $HPE(m)$ değeri log(m!)'dir. Bu nedenle, normalize edilmiş permütasyon entropisi denklem (3.71)'deki gibi elde edilir:

$$H_{NPE}(m) = \frac{H_{PE}(m)}{\ln(m!)} \tag{3.71}$$

Her zaman serisi için $0 \leq H_{NPE}(m) \leq 1$ koşulu sağlanmalıdır. Bu ölçümle ilişkili olarak, Kullback Leibler Entropi (KLE), $1 - H_{NPE}$ olarak tanıtılmıştır. Sıralı desenlerin olasılık dağılımı ve düzgün dağılım arasındaki mesafeyi ölçmektedir [106].

3.2.3 Yaklaşık Entropi

Kaotik sistemleri analiz etmek için geliştirilen hesaplamalar, karmaşık bir sistem için yetersiz kalabilir. Ayrıca bu hesaplamalar sınırlı, gürültülü ve stokastik olarak türetilmiş zaman serileri için kullanıldığında, sonuçların çok başarılı olmadığı söylenebilir. Pincus, deneysel veri setleri için yeni bir istatistik geliştirmiştir. Bu istatistiği geliştirme sebebi, deterministik ve stokastik öğeleri içeren sistemleri karşılaştırmaktır [108]. Bazı stokastik öğeler sistemde mevcutsa, diğer yöntemlerin istatistiksel kesinliği riske atıldığı için Yaklaşık Entropi (YE) algoritması daha kullanışlıdır.

YE, düşük YE değerlerinin sistemin çok kalıcı, tekrarlayan ve öngörücü olduğunu yansıtması anlamında korelasyonu, kalıcılığı veya düzenliliği ölçen, seri boyunca kendini tekrarlayan kalıplarla ölçüm yapan bir parametredir; işlem sonucunda yüksek değer elde edilmesi veriler arasında bağımsızlık, düşük sayıda tekrarlanan kalıp ve rastgelelik anlamına gelir. Bulgusal anlamda YE, bir sonraki artımlı karşılaştırmada yakındaki desen çalışmalarının yakın kalmasının

logaritmik olasılığını ölçer. YE, düzenliliğin kesin istatistiklerinin zorlukları olmadan, değişken karmaşıklık kavramını nicelleştirir. YE'nin gelişiminin arkasındaki ana fikir, bir sistemin dinamiklerini tam olarak belirlemenin sadece tek algoritma ile mümkün olmamasıydı. Bunun yerine YE, sistemleri sınıflandırmak ve karmaşıklıklarının gelişimini incelemek için uygun bir algoritmadır, bu sebeple sınıflandırma yapmak için sistem dinamiklerinin tamamen yeniden yapılandırılması gerekmez. Matematiksel ifade ile, Pincus'un çalışmasının ana fikir, "İki sistemin her birini tanımlayan yeniden yapılandırılmış dinamikler için ortak olasılık ölçütleri farklıysa, o zaman sabit bir bölümdeki koşullu olasılıklar tarafından verilen marjinal olasılık dağılımlarının muhtemelen farklı olacaktır. Ortak olasılıklar oldukça farklıyken, iki sürecin marjinal olasılıklarının eşit olabileceğine dikkat etmek önemlidir. Ancak, eğer iki ölçü belirgin şekilde farklı marjinal olasılıklara sahipse, bu tek başına ölçüleri ayırt etmek için yeterlidir" [109].

YE algoritmasında giriş parametreleri olarak m, r çifti seçilmelidir. İstatistiksel olarak, $m - inci$ mertebesinin koşullu olasılıklarını tahmin etmek, durumların uzayını r genişliğindeki hücrelere bölmeye eşdeğer olacaktır.

Farklı vektör karşılaştırmalarının penceresinin uzunluğu m ile ifade edilir ve r bir gürültü filtresidir. Daha yüksek m ve küçük r, daha keskin (olasılıksal) parametrelerin ayrıntılarını tanımlar. Bununla birlikte, stokastik süreçlerle uğraşırken, koşullu olasılıkların analizi, büyük m değerlerinin veya minimum r değerlerinin istatistiksel olarak eksik tahminler üretmesine neden olur.

Yaklaşık Entropi algoritmasının matematiksel olarak hesaplanması aşağıdaki biçimdedir.

N uzunluğunda $u = \{u(1), u(2), \dots, u(N)\}$ bir sayı dizisi verilsin. M negatif olmayan bir tamsayı, $m \leq N$ ve pozitif bir reel sayı olan r ile $x(i) = \{u(i), u(i+1), \dots, u(i+m-1)\}$ ve $x(j) = \{u(j), u(j+1), \dots, u(j+m-1)\}$ arasındaki mesafe, $d[x(i), x(j)] = \max_{k=1,2,\dots,m} (|u(i+k-1) - u(j+k-1)|)$ gibi hesaplanır. $j \leq N - m + 1$ sayısı öyle ki $C_i^m = d[x(i), x(j)] \leq r)/(N - m + 1)$ hesaplanır. C_i^m'nin payı, r çözünürlüğü içinde, belirli bir bloğa benzeyen ardışık uzunluk m değerlerine sahip blok sayısını sayar.

Hesaplama denklem (3.72)'deki gibi yapılır:

$$\phi^m(r) = \frac{1}{N-m+1} \sum_{i=1}^{N-m+1} \log C_i^m(r) \tag{3.72}$$

YE hesaplanma şekli aşağıdaki gibi tanımlanabilir:

$m \geq 1$ ve $\mathrm{YE}(0,r,N)\,(u) = -\phi^1(r)$ iken,

$$\mathrm{YE}(m,r,N)\,(u) = \phi^m(r) - \phi^{m+1}(r) \tag{3.73}$$

$\mathrm{YE}(0,r,N)\,(u)$, birbirine yakın olan m boylam bloklarının bir sonraki konum için bir arada kaldığı veya farklı bir şekilde yerleştirildiği logaritmik frekansı ölçer, YE'nin negatif değeri şu şekilde tanımlanır:

$-\mathrm{YE}(m,r,N)\,(u) = \phi^{m+1}(r) - \phi^m(r)$ =logaritmanın i üzerinden ortalaması $\mathrm{YE}(m,r,N)$, $\mathrm{YE}(m,r)$ parametresinin istatistiksel tahmincisidir:

$$\mathrm{YE}(m,r) = \lim_{N\to\infty} [\phi^m(r) - \phi^{m+1}(r)] \tag{3.74}$$

YE, m gözlemleri için yakın olan kalıpların sonraki artımlı karşılaştırmalarda yakın kalma olasılığını ölçer. Düzenliliği ima eden yakın kalma olasılığı daha yüksek, daha küçük YE değerleri üretir ve bunun tersi de geçerlidir [110].

YE özellikleri aşağıdaki gibidir:

- YE istatistiği herhangi bir modelden bağımsızdır, bu bağımsızlık durumu onu verilerden başka hiçbir şeyi hesaba katmadan veri serilerinin analizi için uygun kılar.
- YE, Shannon entropisi gibi büyük ve seyrek aykırı değerlere veya sayısal eserlere karşı çok kararlıdır. Büyük aykırı değerler, genel ölçüme küçük bir katkı sağlayan düşük olasılıklı olaylardır. Ancak bu olaylar, değerler ve ortalama arasındaki farklar ölçüldüğü için moment istatistiklerinde kritik öneme sahiptir. Değişkenlik, ortalama değerden sapmaları tespit edebilir, ancak verilerin düzenliliği konusunda bir amacı yoktur. Bir serinin analizi için en iyi yöntemlerden biri, düzenlilik istatistikleri ile moment istatistiklerinin birlikte kullanılması olacaktır. Bu anlamda, YE'nin moment istatistiği yaklaşımlarının önemli farklılıklar gösteremediği

durumlarda normal ve anormal verileri ayırt ettiği gösterilmiştir [101, 110].

- YE'in yapısı ve Bilgi Teorisindeki temelleri nedeniyle, YE'nin değeri negatif değildir. Stokastik süreçler ve gürültülü deterministik süreçler için sonlu bir yapıya sahiptir. YE, tamamen düzenli seriler için 0 değerine sahiptir
- Yaklaşık Entropi, tamamen rastgele ikili diziler için $ln2$ değerini ve genel olarak k sembolden oluşan seriler için lnk değerini alır.
- Yaklaşık Entropi'nin tanımı olarak, sıralamayı korumak göreceli bir özelliktir. YE, m ve r seçimi ile önemli ölçüde değişebilir, bu nedenle mutlak bir ölçü değildir. YE'nin en önemli faydası, göreliliğinin sistemleri ayırt etmek için yeterli olmasıdır. Pincus'a göre, genel olarak, iki veri serisi A ve B verildiğinde, $YE\ (m_1, r_1)(A) \leq YE\ (m_1, r_1)(B)$ ise $YE\ (m_2, r_2)(A) \leq YE\ (m_2, r_2)(B)$ denkleminden söz edilebilir. Bununla birlikte, bu YE'nin genel bir özelliği değildir, bazı çiftler tamamen tutarlıdır, ancak diğerleri tutarlı olmayabilir [101,110].
- Doğrusal olmama durumu, daha büyük bir YE değerine neden olur [101, 110].

3.2.4 Örnek Entropi

YE'nin uygulanmasında, yanlılığının iki önemli sonucu vardır. Birincisi, göreceli tutarlılığın garanti edilmemesi ve r'nin değerine bağlı olarak sonuçların farklı olabilmesidir. İkincisi, YE değerinin veri setinin uzunluğu ile ilişkili olmasıdır. Bu iki problemden kaçınmak için, Richman ve Moorman (2000) kendi kendini saymayan bir istatistik olan Örnek Entropiyi tanımlamışlardır. Örnek Entropi(m, r, N), iki benzer m noktası dizisinin bir sonraki $m + 1$ noktasında benzer kalması ve her vektörü kendisi dışındaki diğer tüm vektörler üzerinde sayması durumunda ortaya çıkan koşullu olasılığın logaritmasının negatif değeridir. Bu, Örnek Entropi'nin göreceli tutarlılığını koruduğu ve aynı zamanda çoğunlukla verilen serinin uzunluğundan bağımsız olduğu anlamına gelir [112].

Richman ve Moorman'a göre, YE'nin yanlılığı, sonuçların gerçekte olduğundan daha fazla düzenlilik önermesini sağlar. Her vektörün kendisini saymasına izin

vererek, tüm $C_i^m(r)$fonksiyonları pozitif kalır ve bir log(0) hesaplanması gereken durumdan kaçınır. $C_i^m(r)$ fonksiyonları, temel olarak, tüm hedef vektörler arasındaki (eşleşmelerin) / (olası vektörlerin toplamı) toplamı olarak hesaplanan koşullu olasılıklardır ve kendi kendine sayma, en az bir eşleşmeye ve bir olası vektöre izin verir. Olası vektörlere B_i ve tüm eşleşmelere A_i dersek, gerçekte YE algoritması, her zaman ölçülmesi gereken önyargısız miktardan $\frac{A_i}{B_i}$ daha büyük olan $\frac{A_i+1}{B_i+1}$'i hesaplar. Bu yanlılık, birkaç N noktasına sahip örnekler için daha önemlidir. Her bir vektörün kendisiyle sayılmasını önleyerek bu önyargıyı ortadan kaldırmak, YE'yi birçok durumda kararsız hale getirecek ve her vektör en az bir eşleşme bulamazsa belirsiz bırakacaktır.

Örnek Entropi, bu yanlılık problemini ortadan kaldırmak için kendi kendine sayma problemini çözmeyi hedefler. Yazarlarına göre, "entropinin bilgi üretim hızının bir ölçüsü olarak tasarlandığı ve bu bağlamda verilerin kendileriyle karşılaştırılmasının anlamsız olduğu" göz önüne alındığında, kendi kendini saymanın ortadan kaldırılması mantıklı görülebilir [112]. Örnek entropi, değerini belirlemek için YE ile aynı model yaklaşımını kullanmaz, yöntem olarak tüm seriyi birlikte kullanır ve yalnızca bir örnek vektörün tanımlanacak $m + 1$ uzunluğunda bir eşleşme bulmasını gerektirir. Her örnek vektörünün tanımlanacak bir eşleşme bulması gereken YE ile çelişir.

Her örnek vektörü için hesaplayarak olası toplam vektör sayısı denklem (3.75)'te verilmiştir.

$$B_i^m(r) = \frac{1}{N-m-1} \times \sum_{j=1, j\neq \mathrm{i}}^{N-m} = \frac{1}{N-m-1} \tag{3.75}$$

Ve tüm örnek vektörler eklendiğinde:

$$B^m(r) = \frac{1}{N-m} \sum_{i=1}^{N-m} B_i^m(r) = \frac{1}{N-m-1} \frac{1}{N-m} \sum_{i=1}^{N-m} \sum_{j=1, j\neq i}^{N-m} \tag{3.76}$$

denklemi elde edilir.

Aynı şekilde, her model vektör için hesaplanarak bulunan toplam eşleşme sayısı aşağıdaki gibi tanımlanabilir:

$$A_i^m(r) = \frac{1}{N-m-1} \times \frac{1}{N-m-1} \sum_{j=1, j\neq i}^{N-m} A_i^m(r) \tag{3.77}$$

Ve aşağıdaki gibi eklendiğinde:

$$A^m(r) = \frac{1}{N-m} \sum_{i=1}^{N-m} A_i^m(r) = \frac{1}{N-m-1} \cdot \frac{1}{N-m} \sum_{i=1}^{N-m} \sum_{j=1, j\neq i}^{N-m} \tag{3.78}$$

denklem (3.78) elde edilmektedir.

Bu nedenle $B^m(r)$, iki dizinin m noktaları (olasılıklar) için benzer olma olasılığıdır, $A^m(r)$ ise iki dizinin $m + 1$ noktaları (eşleşmeler) için benzer olma olasılığıdır. Her zaman olası vektör sayısından küçük veya ona eşit olan eşleşme sayısına göre, $\frac{A^m(r)}{B^m(r)}$ oranı bu hesaplamaya göre birden küçük koşullu bir olasılıktır.

Böylece, ÖE ve YE arasında üç farkın olduğu görülebilir:

- ÖE, YE yaparken kendi kendine saymaya $(j \neq i)$ izin vermez.
- Tüm örnek vektörlerinin toplamına bakıldığında, ÖE'de örnek vektörlerinin toplamı logaritmanın içinde ve YE'de örnek vektörlerinin toplamı dışta yer alır. Bu, ÖE'nin tüm seriyi dikkate aldığını ve bir örnek bir eşleşme bulursa, ÖE'nin zaten tanımlanmış olduğunu, YE'nin ise her örnek için bir eşleşmeye ihtiyacı olduğunu ima eder. Jensen'in eşitsizliği bize $\log(\sum) > \sum log$ olduğunu söyler, böylece bu terim ÖE'de, YE'den daha büyüktür.
- YE, bu istatistiği serinin boyutuna bağlı kılan bir faktör, yani $1/(N-m)$ faktörünü içerirken, ÖE bu faktörü içermez.

Bu duruma göre YE'de ortaya çıkabilecek olan sorunlar, Örnek Entropisi sayesinde ortadan kaldırılmıştır [101].

3.2.5 Bulanık Entropi

Bulanık küme teorisi, son zamanlarda karmaşık süreçlerle etkili bir şekilde başa çıkabilen sistemler tasarlamak için insan muhakemesinin güçlü yeteneğini taklit

etmek için geliştirilmiştir. Tanım olarak, bir bulanık küme, değişen üyelik derecelerine sahip elemanları içeren bir kümedir [113]. Bu, öğelerin o kümede tam üyeliğe sahip olduğu (yani, üyeliklerine 1 değeri atanan) klasik (net) kümelerden farklıdır. Bir bulanık kümenin elemanları, fonksiyon-teorik bir form kullanılarak bir üyelik değerleri evrenine eşlenir. Fonksiyon, bir bulanık kümenin öğelerini [0,1] aralığına ait gerçek bir değere eşler. Bulanık küme teorisi, karmaşık ve kesin olmayan sistemlerin modellenmesinde çok faydalıdır. Ayrıca bir karar verme sistemi olarak kontrol alanında da çok etkin bir şekilde kullanılmıştır [114, 115]. Bir bulanık mantık denetleyicisi dört ana bileşenden oluşur: bulanıklaştırma arayüzü, bilgi tabanı, bulanık karar verme ve durulaştırma arayüzü. Bulanıklaştırma arabirimi, girdi değişkenlerinin değerlerini karşılık gelen söylem evrenlerine aktaran bir ölçek eşlemesi gerçekleştirir. Bilgi tabanı, süreç tanımını içeren bir veri tabanından ve kontrol hedeflerini yansıtan bir dilbilimsel (bulanık) kontrol kuralı tabanından oluşur. Üçüncü bileşen olan bulanık karar verme, Bulanık Mantık Kontrolü'nün çekirdeğidir. İnsanın karar verme mantığını simüle eder ve çıkarım kurallarına dayalı kontrol eylemleri üretir. Bulanıklaştırma arayüzünün aksine, durulaştırma bileşeni net değerli bir kontrol eylemi üretir. Literatürde birkaç durulaştırma yöntemi önerilmiştir. En yaygın olanları ağırlık merkezi veya alan merkezi yöntemi, toplamların merkezi yöntemi ve maksimumların ortalaması yöntemidir.

Belirsizlik ölçüsü, bir bilgi ölçüsü olarak benimsenmiştir. Bu nedenle, bulanıklık ölçüleri bulanık bilgi ölçüleri olarak bilinir [116, 117].

Bir bulanık kümeden veya bulanık bir sistemden elde edilen bulanık bilgi miktarının ölçüsü, bulanık entropi olarak bilinir. Burada bulanık entropinin anlamının klasik Shannon entropisinden oldukça farklı olduğunun altı çizilmelidir çünkü onu tanımlamak için herhangi bir olasılık kavramına gerek yoktur [118-120]. Bunun nedeni, bulanık entropinin muğlak belirsizlikleri içermesi, Shannon entropisinin ise rastgelelik belirsizliği (olasılıksal) içermesidir. Bulanık entropi, üyelik fonksiyonu kavramı kullanılarak tanımlanır. 1972'de De Luca ve Termini [121] Bulanık entropiyi Shannon'ın işlevine göre tanımladılar ve bulanık bir entropinin onları karşılaması gereken bir dizi özellik yayınladılar. Bulanık entropinin özellikleri yaygın olarak kabul görmektedir ve herhangi bir

yeni bulanık entropiyi tanımlamak için bir kriter haline gelmiştir. De Luca ve Termini tarafından önerilen bulanık entropi denklem (3.79) 'de gösterilmiştir. N üyelik fonksiyonunun (μ_i) bulunduğu üyelik fonksiyonu kavramına göre tanımlanır.

$$H_A = -K \sum_{i=1}^{n} \{\mu_i \log(\mu_i) + (1-\mu_i) \log(1-\mu_i)\} \tag{3.79}$$

Bulanık entropinin özelliği aşağıdaki gibidir:

1. A bir keskin küme ise ($\mu_i = 0$ or $1 \forall x_i \in A$) $H_a = 0$'dır.
2. $\mu_i = 0.5 \; \forall x_i \in A$ ise H_a maksimumdur.
3. $H \geq H*$ burada $H*, A*$'nın entropisidir, A'nın keskinleştirilmiş versiyonudur.
4. $H = \bar{H}$ burada H, A'nın tamamlayıcı versiyonunun entropisidir.

Sander [122], resmi olarak Shannon entropisine benzeyen bulanık bir entropinin bir nasıl elde edildiğini sunmuştur.

$X = \{x_1, x_2, x_3, \dots, x_n\}$ kesikli veri seti ve $F_p, p \in [0,1]$ sürekli bulanık kümesi verildiğinde, $F_p(x_i) = p$ eşitliği tüm $x_i \in X, i = 1,2, \dots, n.$ için mümkündür.

Bir bulanık kümenin gücü $f \in [0,1]^X$ denklem (3.80)'deki gibi tanımlanabilir:

$$P(f) = \sum_{i=1}^{n} f(x_i) \tag{3.80}$$

Sander, bulanık entropinin herhangi bir haritalama $d: [0,1]^X \rightarrow [0, \infty)$ işlemi olduğu sonucuna vardı. Shannon Bulanık Entropisi denklem (3.81)'deki gibi tanımlanabilir:

$$H(f) = -c \sum_{i=1}^{n} f(x_i) \ln f(x_i), c > 0 \tag{3.81}$$

f ve g bulanık kümeleri için, VE operatörü minimum, VEYA operatörü ise maksimum operatörü temsil eder.

Sander, tüm $f \in [0,1]^X$ için $d(f) = H(f)$ olması için aşağıdaki özelliklerin bulanık entropi d'ye uygulanması gerektiğini kanıtlamıştır [123]:

1. Keskinlik: $d(f) = 0 \Leftrightarrow f(X) \subset \{0,1\}, f \in [0,1]^X$
2. Değerleme: $d(f \wedge g) + d(f \vee g) = d(f) + d(g), f, g \in [0,1]^X$
3. Genelleştirilmiş katkı: Burada iki haritalama mevcuttur: s *ve* t: $[0,\infty) \rightarrow [0,\infty)$

   X ve Y sınırlı kümeyken her $f \in [0,1]^X, g \in [0,1]^Y$ için öyle ki $d(f \times g) = d(f)t\big(P(g)\big) + s\big(P(f)\big)d(g)$

4
UYGULAMA

4.1 Veri Seti

Bu çalışmada veri seti olarak Physionet üzerinde yer alan Gait in Neurodegenerative Disease Database isimli nörodejeneratif hastalıklara ilişkin sinyalleri içeren veri seti kullanılmıştır [124]. Bu veri seti Hausdorff tarafından yayınlanmıştır. Veri seti içeriği olarak ALS, Parkinson, Huntington hastalığına sahip denekler ile kontrol grubu olan sağlıklı gruba ait yürüyüş sinyalleri ele alınmıştır. Bu veri setinde her denekten 5 dakika boyunca herhangi bir tekerlekli sandalye ya da yardımcı cihaz bulunmadan, ayak bileklerine kayıt cihazı takılarak normal hızlarında yürümeleri istenmiştir. ALS deneği olan 11 kişinin (8 erkek, 3 kadın) yaş ortalaması 55'tir. ALS teşhisi konulduğundan beri geçen ay sayısı 5.5 ve 54 aralığındadır. Huntington deneklerinde ise 20 kişi (6 erkek, 14 kadın) bulunmaktadır, yaş ortalamaları ise 47'dir. Parkinson deneği olan 15 kişinin (10 erkek, 5 kadın) yaş ortalamaları 67 iken, kontrol grubu olan sağlıklı deneklerde 16 kişinin (2 erkek, 14 kadın) yaş ortalaması 39'dur. Bu süreç boyunca sinyaller ölçülüp sayısal değerlere çevrilmiştir.

Veri setinde 13 başlık bulunmaktadır. Bunlar Tablo 4.1'deki gibidir:

Tablo 4.1: Veri kümesi açıklaması

Sütun	İçerikler
1	Geçen zaman (saniye)
2	Sol Adım Aralığı (saniye)
3	Sağ Adım Aralığı (saniye)
4	Sol Salınım Aralığı (saniye)
5	Sağ Salınım Aralığı (saniye)
6	Sol Salınım Aralığı (adıma göre % oranı)
7	Sağ Salınım Aralığı (adıma göre % oranı)
8	Sol Basma Aralığı (saniye)

Tablo 4.1: Veri kümesi açıklaması (devamı)

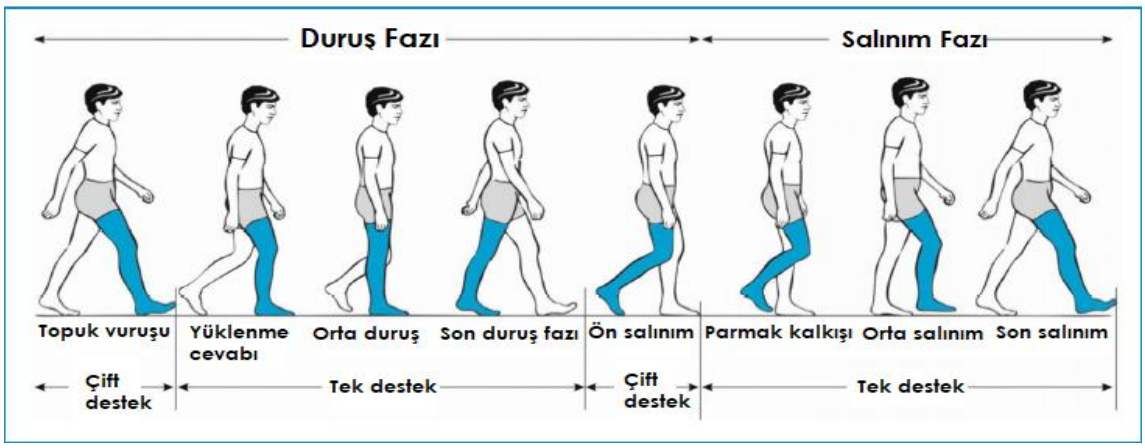
9	Sağ Basma Aralığı (saniye)
10	Sol Basma Aralığı (adıma göre % oranı)
11	Sağ Basma Aralığı (adıma göre % oranı)
12	Çift Destek Aralığı (saniye)
13	Çift Destek Aralığı (adıma göre % oranı)

Bu çalışmada veri seti kullanılırken, yüzde değerlere sahip olan değişkenler, zaten var olan değişkenlerden türetildikleri için kullanılmamıştır. Geçen zaman değişkeni de belirtme amaçlı veride var olduğundan kullanılmamıştır. Veri setinde kullanılan değişkenler Tablo 4.2'deki gibidir:

Tablo 4.2: Veri kümesinde bu çalışmada kullanılacak değişkenler

Sütun	İçerikler	Veri Tipi
1	Sol Adım Aralığı (saniye)	Nümerik
2	Sağ Adım Aralığı (saniye)	Nümerik
3	Sol Salınım Aralığı (saniye)	Nümerik
4	Sağ Salınım Aralığı (saniye)	Nümerik
5	Sol Basma Aralığı (saniye)	Nümerik
6	Sağ Basma Aralığı (saniye)	Nümerik
7	Çift Destek Aralığı (saniye)	Nümerik
8	Etiket	Kategorik

Yürüyüş analizinde kullanılan metoda ilişkin görsel aşağıdaki şekilde verilmiştir:



Şekil 4.1: Yürüyüş döngüsü [11]

4.2 Çalışmada Kullanılan Öznitelikler

Bu çalışmada entropi bazlı sistemler hesaplanıp, öznitelikler oluşturulmuştur. Entropi yöntemleri olarak Yaklaşık Entropi, Bulanık Entropi, Permütasyon Entropisi ve Örnek Entropi kullanılmıştır. Çalışmada sadece entropi yöntemleri değil, aynı zamanda istatistiksel yöntemler de kullanılmıştır. Veri kümesinde yer alan her değişken için (7 değişken) ortalama, standart sapma, Yaklaşık Entropi, Bulanık Entropi, 3 boyutlu Permütasyon Entropisi, 10 boyutlu Permütasyon Entropisi ve Örnek Entropi öznitelikleri çıkarılmıştır. Bu öznitelikler çıkarılırken Python kullanılmıştır. Her değişkenin özniteliği Python programında hesaplanmıştır. Hesaplanan değerler bir Excel dosyasında veri kümesi olarak tutulup, ardından Python'da sınıflandırma yapmak için kullanılmıştır. Permütasyon Entropisi için 3 ve 10 olmak üzere iki boyut seçilmiş ve uygulanmıştır.

Özniteliklerin çıkarılmasına ilişkin oluşturulan kümeler aşağıda gösterilmektedir:

Tablo 4.3: Öznitelikler için oluşturulan kümeler

Sıra	Öznitelik Kümesi	Öznitelik Karşılığı
1	Xa	Her bir deneğin Sol Adım Aralığı, Sağ Adım Aralığı, Sol Salınım Aralığı, Sağ Salınım Aralığı, Sol Basma Aralığı, Sağ Basma Aralığı, Çift Destek Aralığı değişkenlerinin aritmetik ortalaması
2	Xb	Her bir deneğin Sol Adım Aralığı, Sağ Adım Aralığı, Sol Salınım Aralığı, Sağ Salınım Aralığı, Sol Basma Aralığı, Sağ Basma Aralığı, Çift Destek Aralığı değişkenlerinin standart sapması
3	Xc	Her bir deneğin Sol Adım Aralığı, Sağ Adım Aralığı, Sol Salınım Aralığı, Sağ Salınım Aralığı, Sol Basma Aralığı, Sağ Basma Aralığı, Çift Destek Aralığı değişkenlerinin 3 boyutlu Permütasyon Entropisi
4	Xd	Her bir deneğin Sol Adım Aralığı, Sağ Adım Aralığı, Sol Salınım Aralığı, Sağ Salınım Aralığı, Sol Basma Aralığı, Sağ Basma Aralığı, Çift Destek Aralığı değişkenlerinin 10 boyutlu Permütasyon Entropisi

Tablo 4.3: Öznitelikler için oluşturulan kümeler (devamı)

5	Xe	Her bir deneğin Sol Adım Aralığı, Sağ Adım Aralığı Sol Salınım Aralığı, Sağ Salınım Aralığı, Sol Basma Aralığı, Sağ Basma Aralığı, Çift Destek Aralığı değişkenlerinin Yaklaşık Entropisi
6	Xf	Her bir deneğin Sol Adım Aralığı, Sağ Adım Aralığı, Sol Salınım Aralığı, Sağ Salınım Aralığı, Sol Basma Aralığı, Sağ Basma Aralığı, Çift Destek Aralığı değişkenlerinin Bulanık Entropisi
7	Xg	Her bir deneğin Sol Adım Aralığı, Sağ Adım Aralığı, Sol Salınım Aralığı, Sağ Salınım Aralığı, Sol Basma Aralığı, Sağ Basma Aralığı, Çift Destek Aralığı değişkenlerinin Örnek Entropisi

Uygulama yapılırken, tekli, ikili ve üçlü veri kümesi kullanılmıştır. İkili veri kümeleri örneğin, ortalama ve standart sapmanın bir arada öznitelik olarak verildiği model anlamına gelmektedir. Bunlar, kombinasyonlar ile çoğaltılıp, her koşul model ile kullanılmıştır. Üçlü veri kümeleri örneğin, ortalama, standart sapma ve 3 boyutlu Permütasyon Entropisi'nin bir arada öznitelik olarak verildiği model anlamına gelmektedir. İkili veri kümelerinde olduğu gibi, burada da kombinasyonlar ile çoğaltılıp, her koşul model ile kullanılmıştır.

4.3 Sonuçlar

Sonuçlar kısmında sınıflandırma sonucu olarak doğruluk değeri kullanılmıştır. Bu, başarı oranı olarak temsil edilmiş olup, her sonuç için tek veri kümesi ve değişen test yüzdeleri bu tezde gösterilmiştir.

4.1.1 Tek Öznitelik Kümesi ile Elde Edilen Sınıflandırma Doğruluk Sonuçları

Tek veri kümesi girdi olarak kullanıldığında elde edilen sonuçlar aşağıdaki tablolarda verilmiştir. 30% test 70% eğitim, 25% test 75% eğitim, 20% test 80% eğitim, 15% test 85% eğitim, 10% test 90% eğitim kombinasyonları denenmiş ve çıkan sonuçlara göre maksimum 60% başarıya ulaşılmıştır.

Tablo 4.4: Tek öznitelik kümesi kullanılarak 30% test 70% eğitim ile elde edilen sınıflandırma sonuçları sınıflandırma sonuçları

Test %30		Başarı Oranı						
		KNN	Saf Bayes	Lojistik Regresyon	DVM_ Lineer	Rastgele Orman	XGBoost	LightGBM
Kullanılan öznitelik kümesi	Xa	0.40	0.50	0.35	0.30	0.40	0.45	0.40
	Xb	0.40	0.45	0.30	0.20	0.35	0.45	0.35
	Xc	0.45	0.40	0.20	0.20	0.35	0.35	0.30
	Xd	0.40	0.45	0.20	0.20	0.30	0.30	0.45
	Xe	0.35	0.25	0.15	0.25	0.15	0.20	0.20
	Xf	0.40	0.45	0.20	0.20	0.40	0.50	0.35
	Xg	0.30	0.30	0.20	0.15	0.20	0.15	0.25

Tek veri kümesi kullanıldığında elde edilen sonuçlara göre en yüksek sınıflandırma sonucu 50% ile XGBoost ve Saf Bayes sınıflandırıcılarına aittir. XGBoost sınıflandırıcısı için verilen girdi vektörü Xf (Bulanık Entropi), Saf Bayes sınıflandırıcısı için verilen girdi vektörü ise Xa (Ortalama)'dır.

Tablo 4.5: Tek öznitelik kümesi kullanılarak 25% test 75% eğitim ile elde edilen sınıflandırma sonuçları

Test %25		Başarı Oranı						
		KNN	Saf Bayes	Lojistik Regresyon	DVM_ Lineer	Rastgele Orman	XGBoost	LightGBM
Kullanılan öznitelik kümesi	Xa	0.38	0.44	0.38	0.25	0.38	0.44	0.38
	Xb	0.38	0.44	0.38	0.25	0.44	0.44	0.50
	Xc	0.50	0.31	0.25	0.25	0.25	0.31	0.25
	Xd	0.38	0.44	0.25	0.25	0.19	0.19	0.38
	Xe	0.31	0.19	0.13	0.31	0.19	0.19	0.25
	Xf	0.38	0.44	0.25	0.25	0.44	0.44	0.56
	Xg	0.31	0.38	0.25	0.31	0.25	0.13	0.38

Tek veri kümesi kullanıldığında elde edilen sonuçlara göre en yüksek sınıflandırma sonucu 50% ile KNN ve LightGBM sınıflandırıcılarına aittir.

XGBoost sınıflandırıcısı için verilen girdi vektörü Xf (Bulanık Entropi), Saf Bayes sınıflandırıcısı için verilen girdi vektörü ise Xa (Ortalama)'dır.

Tablo 4.6: Tek öznitelik kümesi kullanılarak 20% test 80% eğitim ile elde edilen sınıflandırma sonuçları

Test %20		Başarı Oranı						
		KNN	Saf Bayes	Lojistik Regresyon	DVM_Lineer	Rastgele Orman	XGBoost	LightGBM
Kullanılan öznitelik kümesi	Xa	0.38	0.46	0.38	0.23	0.31	0.46	0.3846
	Xb	0.31	0.46	0.31	0.23	0.31	0.3	0.46
	Xc	0.38	0.38	0.23	0.23	0.31	0.31	0.3846
	Xd	0.38	0.46	0.23	0.23	0.15	0.23	0.3846
	Xe	0.23	0.15	0.15	0.31	0.15	0.08	0.23
	Xf	0.31	0.46	0.23	0.23	0.31	0.46	0.54
	Xg	0.31	0.38	0.31	0.31	0.15	0.15	0.38

Tek veri kümesi kullanıldığında elde edilen sonuçlara göre en yüksek sınıflandırma sonucu 54% ile LightGBM sınıflandırıcısına aittir. XGBoost sınıflandırıcısı için verilen girdi vektörü Xf (Bulanık Entropi)'dir.

Tablo 4.7: Tek öznitelik kümesi kullanılarak 15% test 85% eğitim ile elde edilen sınıflandırma sonuçları

Test %15		Başarı Oranı						
		KNN	Saf Bayes	Lojistik Regresyon	DVM_ Lineer	Rastgele Orman	XGBoost	LightGBM
Kullanılan öznitelik kümesi	Xa	0.60	0.50	0.50	0.20	0.40	0.60	0.40
	Xb	0.40	0.60	0.40	0.20	0.40	0.30	0.50
	Xc	0.30	0.40	0.20	0.20	0.20	0.20	0.30
	Xd	0.50	0.60	0.20	0.20	0.20	0.20	0.50
	Xe	0.30	0.20	0.10	0.30	0.20	0.20	0.50
	Xf	0.30	0.60	0.20	0.20	0.40	0.50	0.60
	Xg	0.30	0.40	0.50	0.40	0.10	0.10	0.40

Tek veri kümesi kullanıldığında elde edilen sonuçlara göre en yüksek sınıflandırma sonucu 60% ile KNN, LightGBM, Saf Bayes ve XGBoost sınıflandırıcılarına aittir. LightGBM sınıflandırıcısı için verilen girdi vektörü Xf (Bulanık Entropi), Saf Bayes sınıflandırıcısı için verilen girdi vektörü ise Xb (Standart Sapma), Xd (10 boyutlu Permütasyon Entropisi) ve Xf (Bulanık Entropi), XGBoost sınıflandırıcısı için verilen girdi vektörü Xa (Ortalama)'dır.

Tablo 4.8: Tek öznitelik kümesi kullanılarak 10% test 90% eğitim ile elde edilen sınıflandırma sonuçları

Test %15		Başarı Oranı						
		KNN	Saf Bayes	Lojistik Regresyon	DVM_ Lineer	Rastgele Orman	XGBoost	LightGBM
Kullanılan öznitelik kümesi	Xa	0.43	0.43	0.29	0.29	0.29	0.57	0.43
	Xb	0.29	0.57	0.29	0.29	0.14	0.43	0.43
	Xc	0.43	0.43	0.29	0.29	0.57	0.43	0.43
	Xd	0.43	0.57	0.29	0.29	0.43	0.43	0.29
	Xe	0.43	0.14	0.14	0.43	0.14	0.29	0.43
	Xf	0.29	0.57	0.29	0.29	0.43	0.43	0.57
	Xg	0.43	0.14	0.57	0.43	0.29	0.29	0.43

Tek veri kümesi kullanıldığında elde edilen sonuçlara göre en yüksek sınıflandırma sonucu 57% ile LightGBM, Saf Bayes, XGBoost, Lojistik Regresyon ve Rastgele Orman sınıflandırıcılarına aittir. LightGBM sınıflandırıcısı için verilen girdi vektörü Xf (Bulanık Entropi), Saf Bayes sınıflandırıcısı için verilen girdi vektörü ise Xb (Standart Sapma), Xd (10 boyutlu Permütasyon Entropisi) ve Xf (Bulanık Entropi), XGBoost sınıflandırıcısı için verilen girdi vektörü Xa (Ortalama), Lojistik Regresyon sınıflandırıcısı için verilen girdi vektörü Xg (Örnek Entropi), Rastgele Orman sınıflandırıcısı için verilen girdi vektörü Xc (3 boyutlu Permütasyon Entropisi)'dir.

4.1.2 İkili Veri Kümesi ile Elde Edilen Sınıflandırma Doğruluk Sonuçları

İkili veri kümeleri girdi olarak kullanıldığında elde edilen sonuçlar aşağıdaki tablolarda verilmiştir. 30% test 70% eğitim, 25% test 75% eğitim, 20% test 80% eğitim, 15% test 85% eğitim, 10% test 90% eğitim kombinasyonları denenmiş ve çıkan sonuçlara göre maksimum 90% başarıya ulaşılmıştır.

Tablo 4.9: İkili öznitelik kümesi kullanılarak 30% test 70% eğitim ile elde edilen sınıflandırma sonuçları

Test %30		Başarı Oranı						
		KNN	Saf Bayes	Lojistik Regresyon	DVM_ Lineer	Rastgele Orman	XGBoost	LightGBM
	Xa, Xb	0.35	0.55	0.35	0.35	0.40	0.55	0.45
	Xa, Xc	0.40	0.65	0.35	0.25	0.40	0.55	0.55
	Xa, Xd	0.40	0.60	0.35	0.25	0.40	0.40	0.45
	Xa, Xe	0.25	0.45	0.20	0.25	0.30	0.35	0.30

Tablo 4.9: İkili öznitelik kümesi kullanılarak 30% test 70% eğitim ile elde edilen sınıflandırma sonuçları (devamı)

Kullanılan öznitelik kümesi	Xa, Xf	0.40	0.60	0.40	0.25	0.60	0.55	0.50
	Xa, Xg	0.30	0.40	0.30	0.35	0.55	0.55	0.60
	Xb, Xc	0.50	0.50	0.30	0.20	0.40	0.35	0.45
	Xb, Xd	0.40	0.35	0.30	0.20	0.50	0.50	0.40
	Xb, Xe	0.25	0.35	0.30	0.20	0.45	0.40	0.40
	Xb, Xf	0.40	0.50	0.40	0.25	0.35	0.45	0.35
	Xb, Xg	0.35	0.45	0.35	0.40	0.40	0.40	0.30
	Xc, Xd	0.30	0.60	0.20	0.20	0.35	0.30	0.35
	Xc, Xe	0.35	0.40	0.15	0.25	0.25	0.30	0.25
	Xc, Xf	0.40	0.50	0.20	0.20	0.35	0.45	0.50
	Xc, Xg	0.30	0.50	0.15	0.20	0.40	0.30	0.40
	Xd, Xe	0.35	0.40	0.15	0.25	0.25	0.15	0.30
	Xd, Xf	0.45	0.65	0.20	0.20	0.55	0.55	0.45
	Xd, Xg	0.30	0.50	0.15	0.20	0.50	0.40	0.55
	Xe, Xf	0.30	0.50	0.25	0.25	0.40	0.45	0.40
	Xe, Xg	0.25	0.30	0.20	0.20	0.20	0.20	0.30
	Xf, Xg	0.30	0.50	0.25	0.20	0.40	0.50	0.40

İkili veri kümesi kullanıldığında elde edilen sonuçlara göre en yüksek sınıflandırma sonucu 65% ile Saf Bayes sınıflandırıcısına aittir. Saf Bayes sınıflandırıcısı için verilen girdi vektörü ise {Xa, Xc} (Ortalama, 3 boyutlu Permütasyon Entropisi) ve {Xd, Xf} (10 boyutlu Permütasyon Entropisi, Bulanık Entropi)'dir.

Tablo 4.10: İkili öznitelik kümesi 25% test 75% eğitim ile elde edilen sınıflandırma sonuçları

Test %25		**Başarı Oranı**						
		KNN	Saf Bayes	Lojistik Regresyon	DVM_ Lineer	Rastgele Orman	XGBoost	LightGBM
Kullanılan öznitelik kümesi	Xa, Xb	0.19	0.50	0.31	0.25	0.56	0.56	0.56
	Xa, Xc	0.38	0.63	0.31	0.25	0.44	0.50	0.63
	Xa, Xd	0.38	0.50	0.38	0.25	0.31	0.44	0.38
	Xa, Xe	0.19	0.38	0.25	0.25	0.38	0.38	0.31
	Xa, Xf	0.44	0.56	0.44	0.25	0.56	0.56	0.50
	Xa, Xg	0.19	0.44	0.56	0.38	0.50	0.50	0.63
	Xb, Xc	0.38	0.50	0.38	0.25	0.38	0.38	0.44
	Xb, Xd	0.38	0.38	0.38	0.25	0.44	0.56	0.38
	Xb, Xe	0.25	0.31	0.31	0.25	0.38	0.38	0.38

Tablo 4.10: İkili öznitelik kümesi 25% test 75% eğitim ile elde edilen sınıflandırma sonuçları (devamı)

	Xb, Xf	0.38	0.44	0.38	0.25	0.44	0.44	0.50
	Xb, Xg	0.25	0.44	0.31	0.38	0.38	0.44	0.44
	Xc, Xd	0.25	0.44	0.25	0.25	0.38	0.56	0.44
	Xc, Xe	0.31	0.38	0.13	0.31	0.25	0.25	0.25
	Xc, Xf	0.38	0.50	0.25	0.25	0.38	0.56	0.50
	Xc, Xg	0.31	0.50	0.38	0.38	0.31	0.38	0.44
	Xd, Xe	0.31	0.38	0.13	0.31	0.25	0.25	0.44
	Xd, Xf	0.38	0.63	0.25	0.25	0.56	0.56	0.44
	Xd, Xg	0.31	0.56	0.31	0.31	0.44	0.44	0.63
	Xe, Xf	0.31	0.44	0.25	0.25	0.38	0.44	0.38
	Xe, Xg	0.13	0.31	0.31	0.31	0.25	0.25	0.25
	Xf, Xg	0.31	0.44	0.38	0.31	0.50	0.44	0.50

İkili veri kümesi kullanıldığında elde edilen sonuçlara göre en yüksek sınıflandırma sonucu 63% ile Saf Bayes ve LightGBM sınıflandırıcılarına aittir. Saf Bayes sınıflandırıcısı için verilen girdi vektörü ise {Xa, Xc} (Ortalama, 3 boyutlu Permütasyon Entropisi) ve {Xd, Xf} (10 boyutlu Permütasyon Entropisi, Bulanık Entropi)'dir. LightGBM sınıflandırıcısı için verilen girdi vektörü ise {Xa, Xc} (Ortalama, 3 boyutlu Permütasyon Entropisi), {Xa, Xg} (Ortalama, Örnek Entropi) ve {Xd, Xg} (10 boyutlu Permütasyon Entropisi, Örnek Entropi)'dir.

Tablo 4.11: İkili öznitelik kümesi 20% test 80% eğitim ile elde edilen sınıflandırma sonuçları

Test %20		**Başarı Oranı**						
		KNN	Saf Bayes	Lojistik Regresyon	DVM_ Lineer	Rastgele Orman	XGBoost	LightGBM
Kullanılan öznitelik kümesi	Xa, Xb	0.23	0.62	0.31	0.23	0.54	0.46	0.54
	Xa, Xc	0.46	0.62	0.38	0.23	0.54	0.54	0.62
	Xa, Xd	0.38	0.54	0.38	0.23	0.15	0.46	0.38
	Xa, Xe	0.23	0.38	0.23	0.23	0.23	0.23	0.31
	Xa, Xf	0.46	0.62	0.46	0.23	0.54	0.62	0.54
	Xa, Xg	0.15	0.46	0.54	0.38	0.54	0.54	0.69
	Xb, Xc	0.31	0.54	0.31	0.23	0.31	0.31	0.54
	Xb, Xd	0.31	0.46	0.31	0.23	0.54	0.46	0.38
	Xb, Xe	0.23	0.38	0.31	0.23	0.31	0.31	0.31
	Xb, Xf	0.31	0.46	0.31	0.23	0.31	0.38	0.54
	Xb, Xg	0.23	0.46	0.31	0.38	0.38	0.46	0.46
	Xc, Xd	0.31	0.38	0.23	0.23	0.54	0.54	0.62

Tablo 4.11: İkili öznitelik kümesi 20% test 80% eğitim ile elde edilen sınıflandırma sonuçları (devamı)

	Xc, Xe	0.23	0.38	0.08	0.31	0.23	0.15	0.23
	Xc, Xf	0.31	0.46	0.23	0.23	0.31	0.38	0.54
	Xc, Xg	0.31	0.38	0.38	0.31	0.38	0.46	0.46
	Xd, Xe	0.23	0.46	0.15	0.31	0.23	0.08	0.38
	Xd, Xf	0.31	0.69	0.23	0.23	0.54	0.46	0.54
	Xd, Xg	0.31	0.54	0.31	0.31	0.46	0.38	0.54
	Xe, Xf	0.31	0.46	0.23	0.23	0.31	0.31	0.38
	Xe, Xg	0.08	0.31	0.31	0.15	0.23	0.15	0.23
	Xf, Xg	0.31	0.46	0.38	0.31	0.38	0.46	0.46

İkili veri kümesi kullanıldığında elde edilen sonuçlara göre en yüksek sınıflandırma sonucu 69% ile Saf Bayes sınıflandırıcısına aittir. Saf Bayes sınıflandırıcısı için verilen girdi vektörü {Xd, Xf} (10 boyutlu Permütasyon Entropisi, Bulanık Entropi)'dir.

Tablo 4.12: İkili öznitelik kümesi 15% test 85% eğitim ile elde edilen sınıflandırma sonuçları

Test %15		**Başarı Oranı**						
		KNN	Saf Bayes	Lojistik Regresyon	DVM_ Lineer	Rastgele Orman	XGBoost	LightGBM
Kullanılan öznitelik kümesi	Xa, Xb	0.20	0.70	0.30	0.30	0.70	0.50	0.40
	Xa, Xc	0.70	0.70	0.50	0.20	0.60	0.50	0.80
	Xa, Xd	0.60	0.60	0.50	0.20	0.30	0.50	0.40
	Xa, Xe	0.30	0.50	0.30	0.20	0.40	0.30	0.50
	Xa, Xf	0.70	0.80	0.50	0.20	0.60	0.60	0.60
	Xa, Xg	0.20	0.60	0.60	0.30	0.70	0.70	0.80
	Xb, Xc	0.40	0.60	0.40	0.20	0.30	0.30	0.40
	Xb, Xd	0.30	0.60	0.40	0.20	0.70	0.40	0.30
	Xb, Xe	0.30	0.50	0.30	0.20	0.30	0.30	0.40
	Xb, Xf	0.30	0.60	0.30	0.30	0.30	0.40	0.50
	Xb, Xg	0.20	0.50	0.30	0.40	0.50	0.40	0.50
	Xc, Xd	0.40	0.70	0.20	0.20	0.50	0.30	0.50
	Xc, Xe	0.30	0.50	0.10	0.30	0.20	0.40	0.30
	Xc, Xf	0.30	0.60	0.20	0.20	0.30	0.40	0.50
	Xc, Xg	0.30	0.50	0.50	0.40	0.40	0.40	0.50
	Xd, Xe	0.30	0.60	0.10	0.30	0.30	0.20	0.50
	Xd, Xf	0.30	0.90	0.20	0.20	0.60	0.40	0.50
	Xd, Xg	0.30	0.70	0.50	0.40	0.60	0.30	0.70
	Xe, Xf	0.40	0.60	0.30	0.30	0.30	0.40	0.40

Tablo 4.12: İkili öznitelik kümesi 15% test 85% eğitim ile elde edilen sınıflandırma sonuçları (devamı)

	Xe, Xg	0.10	0.30	0.30	0.30	0.10	0.10	0.40
	Xf, Xg	0.30	0.60	0.40	0.40	0.40	0.50	0.60

İkili veri kümesi kullanıldığında elde edilen sonuçlara göre en yüksek sınıflandırma sonucu 90% ile Saf Bayes sınıflandırıcısına aittir. Saf Bayes sınıflandırıcısı için verilen girdi vektörü {Xd, Xf} (10 boyutlu Permütasyon Entropisi, Bulanık Entropi)'dir.

Tablo 4.13: İkili öznitelik kümesi 10% test 90% eğitim ile elde edilen sınıflandırma sonuçları

Test %10		**Başarı Oranı**						
		KNN	Saf Bayes	Lojistik Regresyon	DVM_ Lineer	Rastgele Orman	XGBoost	LightGBM
Kullanılan öznitelik kümesi	Xa, Xb	0.29	0.57	0.43	0.29	0.57	0.57	0.43
	Xa, Xc	0.43	0.57	0.29	0.29	0.71	0.71	0.86
	Xa, Xd	0.43	0.57	0.29	0.29	0.43	0.57	0.29
	Xa, Xe	0.29	0.29	0.43	0.29	0.57	0.43	0.29
	Xa, Xf	0.43	0.71	0.43	0.43	0.57	0.71	0.43
	Xa, Xg	0.29	0.43	0.57	0.43	0.57	0.57	0.57
	Xb, Xc	0.29	0.57	0.29	0.29	0.43	0.43	0.57
	Xb, Xd	0.29	0.57	0.29	0.29	0.57	0.57	0.43
	Xb, Xe	0.29	0.57	0.29	0.29	0.29	0.29	0.29
	Xb, Xf	0.29	0.57	0.29	0.29	0.29	0.29	0.43
	Xb, Xg	0.29	0.43	0.43	0.43	0.29	0.43	0.57
	Xc, Xd	0.43	0.57	0.29	0.29	0.43	0.43	0.71
	Xc, Xe	0.43	0.43	0.14	0.43	0.43	0.43	0.57
	Xc, Xf	0.29	0.57	0.29	0.29	0.43	0.43	0.43
	Xc, Xg	0.43	0.43	0.57	0.57	0.57	0.57	0.57
	Xd, Xe	0.43	0.57	0.14	0.43	0.29	0.29	0.43
	Xd, Xf	0.29	0.86	0.29	0.29	0.57	0.43	0.29
	Xd, Xg	0.43	0.57	0.57	0.43	0.71	0.57	0.71
	Xe, Xf	0.43	0.57	0.29	0.29	0.29	0.29	0.43
	Xe, Xg	0.29	0.29	0.29	0.29	0.14	0.14	0.14
	Xf, Xg	0.43	0.57	0.43	0.43	0.43	0.43	0.57

İkili veri kümesi kullanıldığında elde edilen sonuçlara göre en yüksek sınıflandırma sonucu 86% ile Saf Bayes ve LightGBM sınıflandırıcılarına aittir. Saf Bayes sınıflandırıcısı için verilen girdi vektörü {Xd, Xf} (10 boyutlu

Permütasyon Entropisi, Bulanık Entropi)'dir. LightGBM sınıflandırıcısı için verilen girdi vektörü {Xa, Xc} (Ortalama, 3 boyutlu Permütasyon Entropisi)'dir.

4.1.3 Üçlü Veri Kümesi ile Elde Edilen Sınıflandırma Doğruluk Sonuçları

Üçlü veri kümeleri girdi olarak kullanıldığında elde edilen sonuçlar aşağıdaki tablolarda verilmiştir. 30% test 70% eğitim, 25% test 75% eğitim, 20% test 80% eğitim, 15% test 85% eğitim, 10% test 90% eğitim kombinasyonları denenmiş ve çıkan sonuçlara göre maksimum 90% başarıya ulaşılmıştır.

Tablo 4.14: Üçlü öznitelik kümesi kullanılarak 30% test 70% eğitim ile elde edilen sınıflandırma sonuçları

Test %30		Başarı Oranı						
		KNN	Saf Bayes	Lojistik Regresyon	DVM_ Lineer	Rastgele Orman	XGBoost	LightGBM
Kullanılan öznitelik kümesi	Xa, Xb, Xc	0.35	0.60	0.35	0.40	0.45	0.55	0.50
	Xa, Xb, Xd	0.35	0.50	0.35	0.35	0.70	0.55	0.45
	Xa, Xb, Xe	0.25	0.40	0.25	0.30	0.50	0.45	0.45
	Xa, Xb, Xf	0.35	0.60	0.35	0.30	0.50	0.50	0.55
	Xa, Xb, Xg	0.35	0.50	0.35	0.40	0.60	0.55	0.70
	Xa, Xc, Xd	0.40	0.55	0.35	0.25	0.40	0.60	0.60
	Xa, Xc, Xe	0.25	0.65	0.20	0.25	0.35	0.45	0.50
	Xa, Xc, Xf	0.40	0.60	0.40	0.25	0.50	0.50	0.55
	Xa, Xc, Xg	0.30	0.70	0.30	0.35	0.55	0.45	0.60
	Xa, Xd, Xe	0.25	0.55	0.20	0.25	0.30	0.30	0.30
	Xa, Xd, Xf	0.40	0.65	0.40	0.25	0.60	0.55	0.45
	Xa, Xd, Xg	0.30	0.50	0.30	0.35	0.50	0.55	0.60
	Xa, Xe, Xf	0.20	0.60	0.30	0.25	0.45	0.45	0.40
	Xa, Xe, Xg	0.30	0.45	0.30	0.30	0.55	0.55	0.55
	Xa, Xf, Xg	0.30	0.60	0.30	0.40	0.60	0.65	0.70
	Xb, Xc, Xd	0.40	0.55	0.30	0.20	0.60	0.55	0.45
	Xb, Xc, Xe	0.25	0.50	0.30	0.20	0.45	0.25	0.40
	Xb, Xc, Xf	0.45	0.50	0.40	0.25	0.40	0.40	0.50
	Xb, Xc, Xg	0.35	0.50	0.35	0.40	0.45	0.35	0.35
	Xb, Xd, Xe	0.25	0.40	0.30	0.20	0.45	0.40	0.40
	Xb, Xd, Xf	0.40	0.65	0.40	0.25	0.50	0.45	0.45
	Xb, Xd, Xg	0.35	0.40	0.35	0.40	0.45	0.40	0.45
	Xb, Xe, Xf	0.25	0.50	0.30	0.25	0.45	0.45	0.35
	Xb, Xe, Xg	0.25	0.40	0.30	0.25	0.35	0.35	0.45
	Xb, Xf, Xg	0.35	0.50	0.35	0.40	0.40	0.45	0.40
	Xc, Xd, Xe	0.35	0.60	0.15	0.25	0.35	0.30	0.30
	Xc, Xd, Xf	0.50	0.60	0.20	0.20	0.55	0.45	0.50
	Xc, Xd, Xg	0.30	0.60	0.20	0.25	0.40	0.45	0.55
	Xc, Xe, Xf	0.30	0.50	0.25	0.25	0.40	0.40	0.50
	Xc, Xe, Xg	0.25	0.30	0.20	0.20	0.30	0.30	0.45
	Xc, Xf, Xg	0.30	0.55	0.20	0.20	0.50	0.50	0.35

Tablo 4.14: Üçlü öznitelik kümesi kullanılarak 30% test 70% eğitim ile elde edilen sınıflandırma sonuçları (devamı)

	Xd, Xe, Xf	0.30	0.65	0.25	0.25	0.50	0.50	0.45
	Xd, Xe, Xg	0.25	0.45	0.20	0.20	0.45	0.35	0.45
	Xd, Xf, Xg	0.30	0.65	0.25	0.20	0.65	0.50	0.65
	Xe, Xf, Xg	0.25	0.50	0.20	0.15	0.55	0.50	0.55

Üçlü veri kümesi kullanıldığında elde edilen sonuçlara göre en yüksek sınıflandırma sonucu 70% ile Saf Bayes ve LightGBM sınıflandırıcılarına aittir. Saf Bayes sınıflandırıcısı için verilen girdi vektörü {Xa, Xc, Xg} (Ortalama, 3 boyutlu Permütasyon Entropisi, Örnek Entropi)'dir. LightGBM sınıflandırıcısı için verilen girdi vektörü {Xa, Xb, Xg} (Ortalama, Standart Sapma, Örnek Entropi) ve {Xa, Xf, Xg} (Ortalama, Bulanık Entropi, Örnek Entropi)'dir.

Tablo 4.15: Üçlü öznitelik kümesi kullanılarak 25% test 75% eğitim ile elde edilen sınıflandırma sonuçları

Test %25		**Başarı Oranı**						
		KNN	Saf Bayes	Lojistik Regresyon	DVM_ Lineer	Rastgele Orman	XGBoost	LightGBM
Kullanılan öznitelik kümesi	Xa, Xb, Xc	0.19	0.56	0.31	0.25	0.50	0.56	0.56
	Xa, Xb, Xd	0.19	0.50	0.31	0.25	0.56	0.50	0.50
	Xa, Xb, Xe	0.19	0.38	0.31	0.25	0.44	0.44	0.56
	Xa, Xb, Xf	0.31	0.56	0.31	0.25	0.56	0.50	0.50
	Xa, Xb, Xg	0.25	0.44	0.50	0.50	0.69	0.63	0.69
	Xa, Xc, Xd	0.38	0.56	0.31	0.25	0.44	0.50	0.69
	Xa, Xc, Xe	0.19	0.56	0.25	0.25	0.25	0.44	0.44
	Xa, Xc, Xf	0.44	0.56	0.44	0.25	0.56	0.56	0.63
	Xa, Xc, Xg	0.19	0.63	0.56	0.38	0.50	0.56	0.56
	Xa, Xd, Xe	0.19	0.50	0.25	0.25	0.31	0.31	0.31
	Xa, Xd, Xf	0.44	0.63	0.44	0.25	0.56	0.56	0.50
	Xa, Xd, Xg	0.19	0.50	0.56	0.38	0.50	0.56	0.63
	Xa, Xe, Xf	0.19	0.56	0.31	0.25	0.50	0.44	0.44
	Xa, Xe, Xg	0.19	0.44	0.44	0.31	0.44	0.50	0.56
	Xa, Xf, Xg	0.19	0.56	0.56	0.44	0.56	0.63	0.63
	Xb, Xc, Xd	0.38	0.50	0.38	0.25	0.63	0.44	0.38
	Xb, Xc, Xe	0.25	0.50	0.31	0.25	0.38	0.38	0.44
	Xb, Xc, Xf	0.38	0.44	0.38	0.25	0.38	0.44	0.50
	Xb, Xc, Xg	0.25	0.50	0.31	0.38	0.44	0.44	0.44
	Xb, Xd, Xe	0.25	0.38	0.31	0.25	0.63	0.38	0.44
	Xb, Xd, Xf	0.38	0.63	0.38	0.25	0.56	0.50	0.44
	Xb, Xd, Xg	0.25	0.44	0.31	0.38	0.63	0.63	0.56
	Xb, Xe, Xf	0.25	0.44	0.31	0.25	0.38	0.44	0.38
	Xb, Xe, Xg	0.13	0.38	0.25	0.25	0.44	0.38	0.44
	Xb, Xf, Xg	0.25	0.44	0.38	0.38	0.44	0.44	0.50

Tablo 4.15: Üçlü öznitelik kümesi kullanılarak 25% test 75% eğitim ile elde edilen sınıflandırma sonuçları (devamı)

	Xc, Xd, Xe	0.31	0.56	0.13	0.31	0.38	0.31	0.38
	Xc, Xd, Xf	0.38	0.63	0.25	0.25	0.56	0.50	0.50
	Xc, Xd, Xg	0.31	0.63	0.44	0.38	0.50	0.56	0.63
	Xc, Xe, Xf	0.31	0.50	0.25	0.25	0.38	0.50	0.38
	Xc, Xe, Xg	0.13	0.31	0.31	0.31	0.31	0.38	0.44
	Xc, Xf, Xg	0.31	0.50	0.38	0.31	0.44	0.44	0.56
	Xd, Xe, Xf	0.31	0.63	0.25	0.25	0.50	0.50	0.50
	Xd, Xe, Xg	0.13	0.50	0.31	0.31	0.44	0.38	0.50
	Xd, Xf, Xg	0.31	0.63	0.38	0.31	0.56	0.63	0.63
	Xe, Xf, Xg	0.13	0.44	0.31	0.19	0.44	0.44	0.56

Üçlü veri kümesi kullanıldığında elde edilen sonuçlara göre en yüksek sınıflandırma sonucu 69% ile Rastgele Orman ve LightGBM sınıflandırıcılarına aittir. Rastgele Orman sınıflandırıcısı için verilen girdi vektörü {Xa, Xb, Xg} (Ortalama, Standart Sapma, Örnek Entropi)'dir. LightGBM sınıflandırıcısı için verilen girdi vektörü {Xa, Xb, Xg} (Ortalama, Standart Sapma, Örnek Entropi) ve {Xa, Xc, Xd} (Ortalama, 3 boyutlu Permütasyon Entropisi, 10 boyutlu Permütasyon Entropisi)'dir.

Tablo 4.16: Üçlü öznitelik kümesi kullanılarak 20% test 80% eğitim ile elde edilen sınıflandırma sonuçları

Test %20		**Başarı Oranı**						
		KNN	Saf Bayes	Lojistik Regresyon	DVM_ Lineer	Rastgele Orman	XGBoost	LightGBM
Kullanılan öznitelik kümesi	Xa, Xb, Xc	0.23	0.62	0.31	0.23	0.46	0.46	0.62
	Xa, Xb, Xd	0.23	0.54	0.31	0.23	0.62	0.46	0.54
	Xa, Xb, Xe	0.23	0.38	0.31	0.23	0.38	0.46	0.46
	Xa, Xb, Xf	0.31	0.62	0.31	0.23	0.54	0.46	0.54
	Xa, Xb, Xg	0.23	0.54	0.54	0.54	0.62	0.69	0.69
	Xa, Xc, Xd	0.46	0.54	0.38	0.23	0.54	0.54	0.62
	Xa, Xc, Xe	0.23	0.54	0.23	0.23	0.38	0.38	0.38
	Xa, Xc, Xf	0.46	0.62	0.46	0.23	0.54	0.54	0.62
	Xa, Xc, Xg	0.15	0.62	0.54	0.38	0.69	0.69	0.62
	Xa, Xd, Xe	0.23	0.54	0.23	0.23	0.31	0.23	0.38
	Xa, Xd, Xf	0.46	0.69	0.46	0.23	0.54	0.62	0.54
	Xa, Xd, Xg	0.15	0.54	0.54	0.38	0.54	0.46	0.69
	Xa, Xe, Xf	0.23	0.62	0.38	0.23	0.54	0.31	0.62
	Xa, Xe, Xg	0.15	0.46	0.46	0.38	0.54	0.38	0.54
	Xa, Xf, Xg	0.15	0.62	0.62	0.38	0.54	0.62	0.62
	Xb, Xc, Xd	0.31	0.46	0.31	0.23	0.46	0.46	0.46
	Xb, Xc, Xe	0.23	0.54	0.31	0.23	0.31	0.31	0.38

Tablo 4.16: Üçlü öznitelik kümesi kullanılarak 20% test 80% eğitim ile elde edilen sınıflandırma sonuçları (devamı)

	Xb, Xc, Xf	0.31	0.46	0.31	0.23	0.31	0.38	0.46
	Xb, Xc, Xg	0.23	0.54	0.31	0.38	0.46	0.38	0.54
	Xb, Xd, Xe	0.23	0.38	0.31	0.23	0.46	0.38	0.46
	Xb, Xd, Xf	0.31	0.69	0.31	0.23	0.54	0.46	0.54
	Xb, Xd, Xg	0.23	0.46	0.31	0.38	0.62	0.54	0.54
	Xb, Xe, Xf	0.23	0.46	0.31	0.23	0.31	0.38	0.38
	Xb, Xe, Xg	0.08	0.38	0.23	0.23	0.38	0.38	0.38
	Xb, Xf, Xg	0.23	0.46	0.38	0.38	0.38	0.38	0.46
	Xc, Xd, Xe	0.23	0.62	0.15	0.31	0.46	0.23	0.38
	Xc, Xd, Xf	0.31	0.69	0.23	0.23	0.54	0.46	0.54
	Xc, Xd, Xg	0.31	0.62	0.38	0.31	0.54	0.62	0.62
	Xc, Xe, Xf	0.31	0.46	0.31	0.23	0.31	0.38	0.46
	Xc, Xe, Xg	0.08	0.23	0.31	0.23	0.38	0.38	0.38
	Xc, Xf, Xg	0.31	0.46	0.38	0.31	0.38	0.46	0.54
	Xd, Xe, Xf	0.31	0.69	0.23	0.23	0.46	0.38	0.62
	Xd, Xe, Xg	0.08	0.54	0.31	0.15	0.38	0.38	0.54
	Xd, Xf, Xg	0.31	0.69	0.38	0.31	0.54	0.54	0.62
	Xe, Xf, Xg	0.08	0.54	0.31	0.15	0.38	0.38	0.54

Üçlü veri kümesi kullanıldığında elde edilen sonuçlara göre en yüksek sınıflandırma sonucu 69% ile Saf Bayes, Rastgele Orman, XGBoost, LightGBM sınıflandırıcılarına aittir. Saf Bayes sınıflandırıcısı için verilen girdi vektörü {Xa, Xd, Xf} (Ortalama, 10 boyutlu Permütasyon Entropisi, Bulanık Entropi), {Xb, Xd, Xf} (Standart Sapma, 10 boyutlu Permütasyon Entropisi, Bulanık Entropi), {Xc, Xd, Xf} (3 boyutlu Permütasyon Entropisi, 10 boyutlu Permütasyon Entropisi, Bulanık Entropi), {Xd, Xe, Xf} (10 boyutlu Permütasyon Entropisi, Yaklaşık Entropi, Bulanık Entropi), {Xd, Xf, Xg} (10 boyutlu Permütasyon Entropisi, Bulanık Entropi, Örnek Entropi)'dir. LightGBM sınıflandırıcısı için verilen girdi vektörü {Xa, Xb, Xg} (Ortalama, Standart Sapma, Örnek Entropi) ve {Xa, Xc, Xd} (Ortalama, 3 boyutlu Permütasyon Entropisi, 10 boyutlu Permütasyon Entropisi)'dir. Rastgele Orman sınıflandırıcısı için verilen girdi vektörü {Xa, Xc, Xg} (Ortalama, 3 boyutlu Permütasyon Entropisi, Örnek Entropi)'dir. XGBoost sınıflandırıcısı için verilen girdi vektörü {Xa, Xb, Xg} (Ortalama, Standart Sapma, Örnek Entropi)'dir.

Tablo 4.17: Üçlü öznitelik kümesi kullanılarak 15% test 85% eğitim ile elde edilen sınıflandırma sonuçları

Test %15		Başarı Oranı						
		KNN	Saf Bayes	Lojistik Regresyon	DVM_ Lineer	Rastgele Orman	XGBoost	LightGBM
Kullanılan öznitelik kümesi	Xa, Xb, Xc	0.20	0.60	0.30	0.30	0.50	0.60	0.80
	Xa, Xb, Xd	0.20	0.60	0.30	0.30	0.60	0.40	0.60
	Xa, Xb, Xe	0.30	0.50	0.30	0.30	0.40	0.40	0.60
	Xa, Xb, Xf	0.30	0.80	0.40	0.30	0.60	0.50	0.60
	Xa, Xb, Xg	0.30	0.60	0.50	0.50	0.60	0.80	0.80
	Xa, Xc, Xd	0.70	0.60	0.50	0.20	0.60	0.50	0.80
	Xa, Xc, Xe	0.30	0.60	0.30	0.20	0.50	0.50	0.70
	Xa, Xc, Xf	0.70	0.80	0.50	0.20	0.50	0.70	0.60
	Xa, Xc, Xg	0.20	0.60	0.60	0.30	0.80	0.90	0.80
	Xa, Xd, Xe	0.30	0.60	0.30	0.20	0.30	0.30	0.50
	Xa, Xd, Xf	0.70	0.90	0.50	0.20	0.60	0.60	0.60
	Xa, Xd, Xg	0.20	0.70	0.60	0.30	0.80	0.60	0.80
	Xa, Xe, Xf	0.30	0.80	0.50	0.20	0.40	0.40	0.60
	Xa, Xe, Xg	0.20	0.50	0.50	0.50	0.60	0.60	0.50
	Xa, Xf, Xg	0.20	0.80	0.70	0.40	0.60	0.80	0.70
	Xb, Xc, Xd	0.30	0.70	0.40	0.20	0.40	0.50	0.50
	Xb, Xc, Xe	0.30	0.60	0.30	0.20	0.30	0.40	0.40
	Xb, Xc, Xf	0.30	0.60	0.30	0.30	0.20	0.30	0.50
	Xb, Xc, Xg	0.20	0.60	0.30	0.40	0.50	0.40	0.50
	Xb, Xd, Xe	0.30	0.50	0.30	0.20	0.40	0.40	0.50
	Xb, Xd, Xf	0.30	0.80	0.30	0.30	0.60	0.40	0.60
	Xb, Xd, Xg	0.20	0.60	0.30	0.40	0.70	0.60	0.60
	Xb, Xe, Xf	0.30	0.60	0.30	0.30	0.30	0.40	0.40
	Xb, Xe, Xg	0.10	0.40	0.20	0.20	0.30	0.30	0.50
	Xb, Xf, Xg	0.20	0.60	0.30	0.40	0.30	0.40	0.50
	Xc, Xd, Xe	0.30	0.60	0.10	0.30	0.30	0.40	0.50
	Xc, Xd, Xf	0.30	0.80	0.20	0.20	0.50	0.50	0.60
	Xc, Xd, Xg	0.30	0.60	0.50	0.40	0.60	0.70	0.60
	Xc, Xe, Xf	0.40	0.60	0.30	0.30	0.30	0.50	0.40
	Xc, Xe, Xg	0.10	0.30	0.30	0.30	0.40	0.50	0.60
	Xc, Xf, Xg	0.30	0.60	0.50	0.40	0.30	0.40	0.40
	Xd, Xe, Xf	0.40	0.90	0.30	0.30	0.40	0.30	0.70
	Xd, Xe, Xg	0.10	0.70	0.30	0.30	0.60	0.30	0.60
	Xd, Xf, Xg	0.30	0.90	0.40	0.40	0.60	0.70	0.70
	Xe, Xf, Xg	0.10	0.60	0.30	0.20	0.40	0.40	0.60

Üçlü veri kümesi kullanıldığında elde edilen sonuçlara göre en yüksek sınıflandırma sonucu 90% ile Saf Bayes ve XGBoost sınıflandırıcılarına aittir. Saf Bayes sınıflandırıcısı için verilen girdi vektörü {Xa, Xd, Xf} (Ortalama, 10 boyutlu Permütasyon Entropisi, Bulanık Entropi), {Xd, Xe, Xf} (10 boyutlu Permütasyon Entropisi, Yaklaşık Entropi, Bulanık Entropi), {Xd, Xf, Xg} (10

boyutlu Permütasyon Entropisi, Bulanık Entropi, Örnek Entropi)'dir. XGBoost sınıflandırıcısı için verilen girdi vektörü {Xa, Xc, Xg} (Ortalama, 3 boyutlu Permütasyon Entropisi, Örnek Entropi)'dir.

Tablo 4.18: Üçlü öznitelik kümesi kullanılarak 10% test 90% eğitim ile elde edilen sınıflandırma sonuçları

Test %10		Başarı Oranı						
		KNN	Saf Bayes	Lojistik Regresyon	DVM_ Lineer	Rastgele Orman	XGBoost	LightGBM
Kullanılan öznitelik kümesi	Xa, Xb, Xc	0.29	0.57	0.43	0.29	0.57	0.43	0.71
	Xa, Xb, Xd	0.29	0.43	0.43	0.29	0.71	0.57	0.43
	Xa, Xb, Xe	0.29	0.43	0.43	0.29	0.57	0.43	0.43
	Xa, Xb, Xf	0.43	0.71	0.43	0.29	0.57	0.57	0.43
	Xa, Xb, Xg	0.43	0.57	0.57	0.71	0.71	0.71	0.57
	Xa, Xc, Xd	0.43	0.43	0.29	0.29	0.71	0.71	0.86
	Xa, Xc, Xe	0.29	0.43	0.43	0.29	0.57	0.43	0.71
	Xa, Xc, Xf	0.43	0.71	0.43	0.43	0.57	0.71	0.71
	Xa, Xc, Xg	0.29	0.57	0.57	0.43	0.71	0.86	0.86
	Xa, Xd, Xe	0.29	0.57	0.43	0.29	0.29	0.43	0.43
	Xa, Xd, Xf	0.43	0.86	0.43	0.43	0.57	0.71	0.43
	Xa, Xd, Xg	0.29	0.57	0.57	0.43	0.57	0.43	0.71
	Xa, Xe, Xf	0.43	0.71	0.43	0.29	0.57	0.43	0.57
	Xa, Xe, Xg	0.29	0.29	0.57	0.57	0.57	0.43	0.43
	Xa, Xf, Xg	0.29	0.71	0.71	0.57	0.57	0.71	0.71
	Xb, Xc, Xd	0.29	0.57	0.29	0.29	0.71	0.43	0.57
	Xb, Xc, Xe	0.29	0.57	0.29	0.29	0.43	0.43	0.43
	Xb, Xc, Xf	0.29	0.57	0.29	0.29	0.29	0.43	0.57
	Xb, Xc, Xg	0.29	0.57	0.43	0.43	0.57	0.57	0.57
	Xb, Xd, Xe	0.29	0.29	0.29	0.29	0.57	0.43	0.43
	Xb, Xd, Xf	0.29	0.71	0.29	0.29	0.57	0.43	0.43
	Xb, Xd, Xg	0.29	0.57	0.43	0.43	0.71	0.57	0.57
	Xb, Xe, Xf	0.29	0.57	0.29	0.29	0.29	0.29	0.29
	Xb, Xe, Xg	0.29	0.29	0.29	0.29	0.29	0.43	0.43
	Xb, Xf, Xg	0.29	0.57	0.43	0.43	0.43	0.29	0.57
	Xc, Xd, Xe	0.43	0.43	0.14	0.43	0.71	0.43	0.71
	Xc, Xd, Xf	0.29	0.71	0.29	0.29	0.57	0.43	0.57
	Xc, Xd, Xg	0.43	0.43	0.57	0.57	0.71	0.57	0.71
	Xc, Xe, Xf	0.43	0.57	0.29	0.14	0.29	0.43	0.57
	Xc, Xe, Xg	0.29	0.14	0.29	0.29	0.57	0.57	0.57
	Xc, Xf, Xg	0.43	0.57	0.57	0.57	0.43	0.57	0.71
	Xd, Xe, Xf	0.43	0.86	0.29	0.29	0.57	0.29	0.43
	Xd, Xe, Xg	0.29	0.57	0.29	0.29	0.43	0.43	0.29
	Xd, Xf, Xg	0.43	0.86	0.43	0.43	0.57	0.43	0.57
	Xe, Xf, Xg	0.29	0.57	0.29	0.29	0.29	0.43	0.43

Üçlü veri kümesi kullanıldığında elde edilen sonuçlara göre en yüksek sınıflandırma sonucu 86% ile Saf Bayes ve LightGBM sınıflandırıcılarına aittir.

Saf Bayes sınıflandırıcısı için verilen girdi vektörü {Xa, Xd, Xf} (Ortalama, 10 boyutlu Permütasyon Entropisi, Bulanık Entropi), {Xd, Xe, Xf} (10 boyutlu Permütasyon Entropisi, Yaklaşık Entropi, Bulanık Entropi), {Xd, Xf, Xg} (10 boyutlu Permütasyon Entropisi, Bulanık Entropi, Örnek Entropi)'dir. LightGBM sınıflandırıcısı için verilen girdi vektörü {Xa, Xc, Xd} (Ortalama, 3 boyutlu Permütasyon Entropisi, 10 boyutlu Permütasyon Entropisi), {Xa, Xc, Xg} (Ortalama, 3 boyutlu Permütasyon Entropisi, Örnek Entropi)'dir.

5
SONUÇLAR

Yapılan araştırmada, nörodejeneratif hastalıklara sahip kişilerin yürüyüş sinyalleri kullanılarak sınıflandırma yapılmıştır. Bu sınıflandırma sonucunda ikili veri kümesiyle yapılan 15% test, 85% eğitim koşulu sağlayan analizde, {Xd, Xf} (10 boyutlu Permütasyon Entropisi, Bulanık Entropi) girdilerinin Saf Bayes ile 90% başarı elde ettiği görülmüştür. Üçlü veri kümesi kullanıldığında elde edilen sonuçlara göre en yüksek sınıflandırma sonucu 90% ile Saf Bayes ve XGBoost sınıflandırıcılarına aittir. {Xa, Xd, Xf} (Ortalama, 10 boyutlu Permütasyon Entropisi, Bulanık Entropi), {Xd, Xe, Xf} (10 boyutlu Permütasyon Entropisi, Yaklaşık Entropi, Bulanık Entropi) girdileri Saf Bayes algoritmasında 90% sınıflandırma başarısı elde etmişlerdir. {Xa, Xc, Xg} (Ortalama, 3 boyutlu Permütasyon Entropisi, Örnek Entropi) girdisi ise XGBoost algoritması ile 90% başarı elde etmiştir. Bu durumda, ikili ve üçlü veri kümelerinde Saf Bayes algoritmasının iyi bir sınıflandırıcı olduğu görülmektedir.

Tezin sonucuna göre, istatistiksel tabanlı öznitelikler ile entropi tabanlı öznitelikler bir arada girdi vektörü olarak kullanıldığında başarı oranının arttığı söylenebilir. Tezin amacı olarak verilen, istatistiksel öznitelikler ile entropi tabanlı özniteliklerin kullanılarak başarılı bir sınıflandırma yapılması koşulu, 90% başarı ile sağlanmıştır. Literatürde farklı entropi türleri ile çalışmalar olsa da Bulanık Entropi, Örnek Entropi ve Yaklaşık Entropi'nin bir arada kullanıldığı bir vaka bulunmamaktadır. Bu noktada tez, diğer çalışmalardan ayrışmaktadır. Ayrıca tezin sonucu olarak, 15% test, 85% eğitim ile yapılan sınıflandırmalarda diğer test ve eğitim gruplarına göre daha iyi sonuç elde edildiği görülmüştür. Nitekim, 90% başarı olarak görülen sınıflandırma sonuçlarının hepsi 15% test, 85% eğitim ile yapılan sınıflandırma sonucuna aittir. İkili veri kümesi kullanılarak yapılan analizde 10 boyutlu Permütasyon Entropisi ve Bulanık Entropi'nin birlikte iyi sonuç verdiği görülmektedir. Üçlü veri kümesi ile yapılan analizde ise 90% başarıya ulaşan üç durum vardır. {Xa, Xd, Xf} (Ortalama, 10

boyutlu Permütasyon Entropisi, Bulanık Entropi) durumunda, istatistiksel tabanlı öznitelikler ile entropi tabanlı öznitelikler birleştiğinde iyi bir sınıflandırma başarısı elde edildiği söylenebilir. {Xd, Xe, Xf} (10 boyutlu Permütasyon Entropisi, Yaklaşık Entropi, Bulanık Entropi) durumunda ise sadece entropi tabanlı öznitelikler bir arada kullanıldığında iyi bir sınıflandırma başarısı elde edilmiştir. Bu iki durum Saf Bayes sınıflandırıcısıyla başarılı bir şekilde çalışmıştır. İkili ve üçlü veri kümelerinde ortak olarak Permütasyon Entropisi ve Bulanık Entropi öznitelikleri bulunmaktadır. Diğer yandan tek veri kümesi kullanıldığında başarı oranı ikili ve üçlü veri kümelerine göre daha düşüktür. Öz E. Vd. (2022)'e göre boyutluluk problemi, ilgili sınıflandırıcıya girdi olarak kullanılmak üzere bir öznitelik vektörü oluşturulmak istendiğinde ortaya çıkar. Bu sorunu aşmak için özellik vektörünün her bir elemanı aynı boyutta tanımlanmalıdır. Entropi ölçümleri, farklı uzunluklara sahip karmaşık yapılı sinyalleri özetlemek için etkili bir şekilde uygulanabilir [125]. Bu tezde entropi yöntemlerinin kullanılması ile farklı uzunluklara sahip veriler uygulanabilmiştir. Aynı zamanda bu tezde ham veriler kullanılmış olup, verilerde dönüştürme işlemi uygulanmamıştır.

KAYNAKÇA

[1] Alpaydin, E. (2020). Introduction to machine learning. MIT press.

[2] Amor, S., Puentes, F., Baker, D., & Van Der Valk, P. (2010). Inflammation in neurodegenerative diseases. Immunology, 129(2), 154-169.

[3] Checkoway, H., Lundin, J. I., & Kelada, S. N. (2011). Neurodegenerative diseases. IARC scientific publications, (163), 407-419.

[4] Neurological Disorders: Public Health Challenges, World Health Organization (WHO), 2006.

[5] L. Bahureksa et al., "The impact of mild cognitive impairment on gait and balance: A systematic review and meta-analysis of studies using instrumented assessment", Gerontology, vol. 63, no. 1, pp. 67-83, 2017.

[6] R. McArdle, R. Morris, J. Wilson, B. Galna, A. J. Thomas and L. Rochester, J. Alzheimer's Dis., vol. 60, no. 4, pp. 1295-1312, 2017.

[7] Hausdorff, J. M., Mitchell, S. L., Firtion, R., Peng, C. K., Cudkowicz, M. E., Wei, J. Y., & Goldberger, A. L. (1997). Altered fractal dynamics of gait: reduced stride-interval correlations with aging and Huntington's disease. Journal of applied physiology (Bethesda, Md. : 1985), 82(1), 262–269. https://doi.org/10.1152/jappl.1997.82.1.262

[8] Nam Nguyen, Q. D., Liu, A.-B., & Lin, C.-W. (2020). Development of a Neurodegenerative Disease Gait Classification Algorithm Using Multiscale Sample Entropy and Machine Learning Classifiers. Entropy, 22(12), 1340.

[9] Setiawan, F., Liu, A. B., & Lin, C. W. (2022). Development of Neuro-Degenerative Diseases' Gait Classification Algorithm Using Convolutional Neural Network and Wavelet Coherence Spectrogram of Gait Synchronization. IEEE Access, 10, 38137-38153.

[10] Pramanik, M., Borah, S., Pradhan, R., Nandy, P., Kalam, A., Barsocchi, P., & Bhoi, A. K. Weightless Neural Network and Neighborhood Rough Set for Parkinson's Detection on Gait Movement.

[11] Berke Erdaş, Ç., Sümer, E., & Kibaroğlu, S. (2022). CNN-based severity prediction of neurodegenerative diseases using gait data. Digital Health, 8, 20552076221075147.

[12] Wu, Y., & Krishnan, S. (2009). Computer-aided analysis of gait rhythm fluctuations in amyotrophic lateral sclerosis. Medical & biological engineering & computing, 47(11), 1165-1171.

[13] Wu, Y., & Ng, S. C. (2010, August). A PDF-based classification of gait cadence patterns in patients with amyotrophic lateral sclerosis. In 2010 Annual International Conference of the IEEE Engineering in Medicine and Biology (pp. 1304-1307). IEEE.

[14] Bilgin, S. (2017). The impact of feature extraction for the classification of amyotrophic lateral sclerosis among neurodegenerative diseases and healthy subjects. Biomedical Signal Processing and Control, 31, 288-294.

[15] BİLGİN, S., & Akin, Z. E. (2018). Gait pattern discrimination of ALS patients using classification methods. Turkish Journal of Electrical Engineering and Computer Sciences, 26(3), 1367-1377.

[16] Alaskar, H., & Jaafar Hussain, A. (2018, August). Data mining to support the discrimination of amyotrophic lateral sclerosis diseases based on gait analysis. In International Conference on Intelligent Computing (pp. 760-766). Springer, Cham.

[17] Aşuroğlu, T., Açıcı, K., Erdaş, Ç. B., Toprak, M. K., Erdem, H., & Oğul, H. (2018). Parkinson's disease monitoring from gait analysis via foot-worn sensors. Biocybernetics and Biomedical Engineering, 38(3), 760-772.

[18] Zeng, W., Liu, F., Wang, Q., Wang, Y., Ma, L., & Zhang, Y. (2016). Parkinson's disease classification using gait analysis via deterministic learning. Neuroscience letters, 633, 268-278.

[19] Huang, X., Khushi, M., Latt, M., Loy, C., & Poon, S. K. (2019, December). Machine Learning Based Method for Huntington's Disease Gait Pattern Recognition. In International Conference on Neural Information Processing (pp. 607-614). Springer, Cham.

[20] Huang, H. P., Hsu, C. F., Mao, Y. C., Hsu, L., & Chi, S. (2021). Gait stability measurement by using average entropy. Entropy, 23(4), 412.

[21] Li, Z., Chen, W., Wang, J., & Liu, J. (2014, June). An automatic recognition system for patients with movement disorders based on wearable sensors. In 2014 9th IEEE Conference on Industrial Electronics and Applications (pp. 1948-1953). IEEE.

[22] Ren, P., Zhao, W., Zhao, Z., Bringas-Vega, M. L., Valdes-Sosa, P. A., & Kendrick, K. M. (2015). Analysis of gait rhythm fluctuations for neurodegenerative diseases by phase synchronization and conditional entropy. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 24(2), 291-299.

[23] Yang, M., Zheng, H., Wang, H., & McClean, S. (2009, April). Feature selection and construction for the discrimination of neurodegenerative diseases based on gait analysis. In 2009 3rd International Conference on Pervasive Computing Technologies for Healthcare (pp. 1-7). IEEE.

[24] Liao, F., Wang, J., & He, P. (2008). Multi-resolution entropy analysis of gait symmetry in neurological degenerative diseases and amyotrophic lateral sclerosis. Medical engineering & physics, 30(3), 299-310.

[25] Cirugeda-Roldan, E., Cuesta-Frau, D., Miro-Martinez, P., & Oltra-Crespo, S. (2014). Comparative study of entropy sensitivity to missing biosignal data. Entropy, 16(11), 5901-5918.

[26] Xia, Y., Gao, Q., & Ye, Q. (2015). Classification of gait rhythm signals between patients with neuro-degenerative diseases and normal subjects:

Experiments with statistical features and different classification models. Biomedical Signal Processing and Control, 18, 254-262.

[27] Prabhu, P., Karunakar, A. K., Anitha, H., & Pradhan, N. (2020). Classification of gait signals into different neurodegenerative diseases using statistical analysis and recurrence quantification analysis. Pattern Recognition Letters, 139, 10-16.

[28] Lin, C. W., Wen, T. C., & Setiawan, F. (2020). Evaluation of vertical ground reaction forces pattern visualization in neurodegenerative diseases identification using deep learning and recurrence plot image feature extraction. Sensors, 20(14), 3857.

[29] Pham, T. D. (2017). Texture classification and visualization of time series of gait dynamics in patients with neuro-degenerative diseases. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 26(1), 188-196.

[30] Tysnes, O. B., & Storstein, A. (2017). Epidemiology of Parkinson's disease. Journal of neural transmission, 124(8), 901-905.

[31] Goetz, C. G. (2011). The history of Parkinson's disease: early clinical descriptions and neurological therapies. Cold Spring Harbor perspectives in medicine, 1(1), a008862.

[32] Patil, R. R. (2022). Epidemiology of Parkinson's Disease—Current Understanding of Causation and Risk Factors. In Techniques for Assessment of Parkinsonism for Diagnosis and Rehabilitation (pp. 31-48). Springer, Singapore.

[33] Gupta, B. M., & Bala, A. (2013). Parkinson's disease in India: An analysis of publications output during 2002-2011. International Journal of Nutrition, Pharmacology, Neurological Diseases, 3(3), 254.

[34] Feigin, V. L., Abajobir, A. A., Abate, K. H., Abd-Allah, F., Abdulle, A. M., Abera, S. F., ... & Nguyen, G. (2017). Global, regional, and national burden of neurological disorders during 1990–2015: a systematic analysis for the Global Burden of Disease Study 2015. The Lancet Neurology, 16(11), 877-897.

[35] Darweesh, S. K., Raphael, K. G., Brundin, P., Matthews, H., Wyse, R. K., Chen, H., & Bloem, B. R. (2018). Parkinson matters. Journal of Parkinson's Disease, 8(4), 495-498.

[36] Letizia, C. M., Ahmar, S. S., Ricci, L., & Di Lazzaro, V. (2020). Gait Analysis in Parkinson's Disease: An Overview of the Most Accurate Markers for Diagnosis and Symptoms Monitoring.

[37] Gibb, W. R., & Lees, A. (1988). The relevance of the Lewy body to the pathogenesis of idiopathic Parkinson's disease. Journal of Neurology, Neurosurgery & Psychiatry, 51(6), 745-752.

[38] Rizzo, G., Copetti, M., Arcuti, S., Martino, D., Fontana, A., & Logroscino, G. (2016). Accuracy of clinical diagnosis of Parkinson disease: a systematic review and meta-analysis. Neurology, 86(6), 566-576.

[39] Hardiman, O., Al-Chalabi, A., Chio, A., Corr, E. M., Logroscino, G., Robberecht, W., ... & Van Den Berg, L. H. (2017). Amyotrophic lateral sclerosis. Nature reviews Disease primers, 3(1), 1-19.

[40] Al-Chalabi, A., Hardiman, O., Kiernan, M. C., Chiò, A., Rix-Brooks, B., & van den Berg, L. H. (2016). Amyotrophic lateral sclerosis: moving towards a new classification system. The Lancet Neurology, 15(11), 1182-1194.

[41] Rooney, J. P., Brayne, C., Tobin, K., Logroscino, G., Glymour, M. M., & Hardiman, O. (2017). Benefits, pitfalls, and future design of population-based registers in neurodegenerative disease. Neurology, 88(24), 2321-2329.

[42] Logroscino, G., Traynor, B. J., Hardiman, O., Chiò, A., Mitchell, D., Swingler, R. J., ... & Beghi, E. (2010). Incidence of amyotrophic lateral sclerosis in Europe. Journal of Neurology, Neurosurgery & Psychiatry, 81(4), 385-390.

[43] Hardiman, O., Al-Chalabi, A., Brayne, C., Beghi, E., van den Berg, L. H., Chio, A., ... & Rooney, J. (2017). The changing picture of amyotrophic lateral sclerosis: lessons from European registers. Journal of Neurology, Neurosurgery & Psychiatry, 88(7), 557-563.

[44] Chiò, A., Logroscino, G., Traynor, B. J., Collins, J., Simeone, J. C., Goldstein, L. A., & White, L. A. (2013). Global epidemiology of amyotrophic lateral sclerosis: a systematic review of the published literature. Neuroepidemiology, 41(2), 118-130.

[45] Marin, B., Hamidou, B., Couratier, P., Nicol, M., Delzor, A., Raymondeau, M., ... & French register of ALS in Limousin (FRALim). (2014). Population-based epidemiology of amyotrophic lateral sclerosis (ALS) in an ageing E urope–the F rench register of ALS in L imousin (FRAL im register). European Journal of Neurology, 21(10), 1292-e79.

[46] Gordon, P. H., Mehal, J. M., Holman, R. C., Rowland, L. P., Rowland, A. S., & Cheek, J. E. (2013). Incidence of amyotrophic lateral sclerosis among American Indians and Alaska natives. JAMA neurology, 70(4), 476-480.

[47] Joensen, P. (2012). Incidence of amyotrophic lateral sclerosis in the Faroe Islands. Acta neurologica scandinavica, 126(1), 62-66.

[48] Marin, B., Logroscino, G., Boumédiene, F., Labrunie, A., Couratier, P., Babron, M. C., ... & Beghi, E. (2016). Clinical and demographic factors and outcome of amyotrophic lateral sclerosis in relation to population ancestral origin. European journal of epidemiology, 31(3), 229-245.

[49] Abrahams, S., Newton, J., Niven, E., Foley, J., & Bak, T. H. (2014). Screening for cognition and behaviour changes in ALS. Amyotrophic lateral sclerosis and frontotemporal degeneration, 15(1-2), 9-14.

[50] Burke, T., Elamin, M., Galvin, M., Hardiman, O., & Pender, N. (2015). Caregiver burden in amyotrophic lateral sclerosis: a cross-sectional investigation of predictors. Journal of neurology, 262(6), 1526-1532.

[51] Folstein, S. E. (1989). Huntington's disease: a disorder of families. Johns Hopkins University Press.

[52] Ross, C. A., & Margolis, R. (2020). Huntington disease. Medicine, 76, 305-338.

[53] Bates, G. P., Dorsey, R., Gusella, J. F., Hayden, M. R., Kay, C., Leavitt, B. R., ... & Tabrizi, S. J. (2015). Huntington disease. Nature reviews Disease primers, 1(1), 1-21.

[54] Ross, C. A., Aylward, E. H., Wild, E. J., Langbehn, D. R., Long, J. D., Warner, J. H., ... & Tabrizi, S. J. (2014). Huntington disease: natural history, biomarkers and prospects for therapeutics. Nature Reviews Neurology, 10(4), 204-216.

[55] Fisher, E. R., & Hayden, M. R. (2014). Multisource ascertainment of Huntington disease in Canada: prevalence and population at risk. Movement Disorders, 29(1), 105-114.

[56] Evans, S. J., Douglas, I., Rawlins, M. D., Wexler, N. S., Tabrizi, S. J., & Smeeth, L. (2013). Prevalence of adult Huntington's disease in the UK based on diagnoses recorded in general practice records. Journal of Neurology, Neurosurgery & Psychiatry, 84(10), 1156-1160.

[57] Vuong, K., Canning, C. G., Menant, J. C., & Loy, C. T. (2018). Gait, balance, and falls in Huntington disease. Handbook of clinical neurology, 159, 251-260.

[58] Aristotle,. (2004). On the Gait of Animals.

[59] G. Borelli, "On the Movement of Animals", 1989

[60] Cicirelli, G., Impedovo, D., Dentamaro, V., Marani, R., Pirlo, G., & D'Orazio, T. R. (2021). Human gait analysis in neurodegenerative diseases: a review. IEEE Journal of Biomedical and Health Informatics, 26(1), 229-242.

[61] Chang C.-C. ve Lin, C.-J. (2011) "LIBSVM: A library for support vector machines", ACM Trans. Intell. Syst. Technol. 2, 3, Article 27 1-27.

[62] Büyüklü, A. H., Aşkın, Ö. E., Filiz, E., & Öz, E., (2017). DOĞRUSAL OLMAYAN DESTEK VEKTÖR MAKİNELERİ İLE BANKACILIKSEKTÖRÜ ÜZERİNE BİR SINIFLANDIRMA UYGULAMASI. XVIII. Uluslararası Ekonometri, Yöneylem Araştırması ve İstatistik Sempozyumu (pp.258). Trabzon, Turkey

[63] Campbell C., Ying Y., (2001) Learning with Support Vector Machines. Morgan & Claypool Publishers, San Rafael

[64] Vapnik N.V., (1998) Statistical Learning Theory. Wiley, New York

[65] Cherkassky V., Mulier F., (1998) Learning From Data: Concepts, Theory, Methods. Wiley-Interscience, New York

[66] Scholkopf B., Smola A., (2001) Learning with Kernels. MIT Press, Cambridge

[67] E. Fix, J. L. Hodges, Discriminatory Analysis, Nonparametric Discrimination: Consistency Properties, Technique Report No. 4, U. S. Air Force School of Aviation Medicine, Randolf Field Texas, 1951, 238-247

[68] Gou, J., Du, L., Zhang, Y., & Xiong, T. (2012). A new distance-weighted k-nearest neighbor classifier. J. Inf. Comput. Sci, 9(6), 1429-1436.

[69] Triguero, I, García-Gil, D, Maillo, J, Luengo, J, García, S, Herrera, F. Transforming big data into smart data: An insight on the use of the k-nearest neighbors algorithm to obtain quality data. WIREs Data Mining Knowl Discov. 2019; 9:e1289.

[70] Cover, T. M., &Hart, P. E.(1967).Nearest neighbor pattern classification.IEEE Transactions on Information Theory,13(1),21-27.

[71] Islam, M. J., Wu, Q. J., Ahmadi, M., & Sid-Ahmed, M. A. (2007, November). Investigating the performance of naive-bayes classifiers and k-nearest neighbor classifiers. In 2007 international conference on convergence information technology (ICCIT 2007) (pp. 1541-1546). IEEE.

[72] Prasatha, V. S., Alfeilate, H. A. A., Hassanate, A. B., Lasassmehe, O., Tarawnehf, A. S., Alhasanatg, M. B., & Salmane, H. S. E. (2017). Effects of distance measure choice on knn classifier performance-a review. arXiv preprint arXiv:1708.04321, 56.

[73] Deza, M. M., & Deza, E. (2009). Encyclopedia of distances. In Encyclopedia of distances (pp. 1-583). Springer, Berlin, Heidelberg.

[74] Grabusts, P. (2011). The choice of metrics for clustering algorithms. Environment. Technology. Resources , 70-76.

[75] Todeschini, R., Ballabio, D., & Consonni, V. (2015). Distances and other dissimilarity measures in chemometrics. Encyclopedia of Analytical Chemistry.

[76] Verma, J. P. (2012). Data Analysis in Management with SPSS Software. Springer

[77] Fulekar, M. H. (2009). Bioinformatics: Applications in life and environmental sciences. Springer

[78] Anastasiy Motrenko, Vadim Strijov, Gerhard-Wilhelm Weber: Sample size determination for logistic regression. Journal of Computational and Applied Mathematics Volume 255, Pages 743-752. 2014

[79] Hastie, T., Tibshirani, R., Friedman, J. The elements of statistical learning: data mining, inference and prediction. Springer. 2009

[80] Ng, Andrew. CS229 Lecture Notes. Stanford University. 2012

[81] Kirasich, K., Smith, T., & Sadler, B. (2018). Random forest vs logistic regression: binary classification for heterogeneous datasets. SMU Data Science Review, 1(3), 9.

[82] Li, N., & Jimenez, R. (2018). A logistic regression classifier for long-term probabilistic prediction of rock burst hazard. Natural Hazards, 90(1), 197-215.

[83] Devetyarov, D., & Nouretdinov, I. (2010, October). Prediction with confidence based on a random forest classifier. In IFIP international

conference on artificial intelligence applications and innovations (pp. 37-44). Springer, Berlin, Heidelberg.

[84] Breiman, L. Random Forests. 2001

[85] Strobl, C., et al.: Bias in random forest variable importance measures: illustrations, sources and a solution. BMC Bioinformatics 8, 25 (2007)

[86] Schapire, R. E., & Freund, Y. (2013). Boosting: Foundations and algorithms. Kybernetes.

[87] Freund, Y., Schapire, R., & Abe, N. (1999). A short introduction to boosting. Journal-Japanese Society For Artificial Intelligence, 14(771-780), 1612.

[88] Friedman JH (2001) Greedy function approximation: a gradient boosting machine. Ann Stat 29(5):1189-1232

[89] Friedman, J. H. (2002). Stochastic gradient boosting. Computational statistics & data analysis, 38(4), 367-378.

[90] Chen, T., & Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining (pp. 785-794).

[91] Bentéjac, C., Csörgő, A., & Martínez-Muñoz, G. (2021). A comparative analysis of gradient boosting algorithms. Artificial Intelligence Review, 54(3), 1937-1967.

[92] Dong, W., Huang, Y., Lehane, B., & Ma, G. (2020). XGBoost algorithm-based prediction of concrete electrical resistivity for structural health monitoring. Automation in Construction, 114, 103155.

[93] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., ... & Liu, T. Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. Advances in neural information processing systems, 30.

[94] Berrar, D. (2018). Bayes' theorem and naive Bayes classifier. Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics, 403.

[95] Sammut, C., & Webb, G. I. (2017). Encyclopedia of machine learning and data mining. Springer Publishing Company, Incorporated.

[96] Rish, I. (2001, August). An empirical study of the naive Bayes classifier. In IJCAI 2001 workshop on empirical methods in artificial intelligence (Vol. 3, No. 22, pp. 41-46).

[97] Webb, G. I., Keogh, E., & Miikkulainen, R. (2010). Naïve Bayes. Encyclopedia of machine learning, 15, 713-714.

[98] Berrar, D., & Flach, P. (2012). Caveats and pitfalls of ROC analysis in clinical microarray research (and how to avoid them). Briefings in bioinformatics, 13(1), 83-97.

[99] Nyquist, H. (1924). Certain Factors Affecting Telegraph Speed, Bell Labs Technical Journal, April.

[100] Hartley, R. V. (1928). Transmission of information 1. Bell System technical journal, 7(3), 535-563.

[101] Delgado-Bonal, A., & Marshak, A. (2019). Approximate entropy and sample entropy: A comprehensive tutorial. Entropy, 21(6), 541.

[102] Bandt C.; Pompe B. (2002) “Permutation entropy: A natural complexity measure for time series.” Phys. Rev. Lett., 88, 174102–1–174102–4.

[103] Riedl M., Müller A., Wessel N., (2013) "Practical considerations of permutation entropy", Eur. Phys. J. Special Topics 222, 249-262

[104] Keller K, Mangold T., Stolz I., Werner J. (2017) "Permutation Entropy: New Ideas and Challanges", Entropy, 19,3, 134

[105] Dostál O., Vysata O., Pazdera L., Procházka A., Kopal J., Kuchy?ka J., Valis M., (2018) "Permutation Entropy and Signal Energy Increase the Accuracy of Neuropathic Change Detection in Needle EMG," Computational Intelligence and Neuroscience, vol. 2018

[106] Zanin M., Zunino L., Rosso O.A., & Papo D. (2012). "Permutation Entropy and Its Main Biomedical and Econophysics Applications: A Review." Entropy, 14, 1553-1577.

[107] Wu S.D., Wu P.H., Wu C.W., Ding J.J., Wang C.-C. (2012) "Bearing Fault Diagnosis Based on Multiscale Permutation Entropy and Support Vector Machine." Entropy 14, 1343-1356.

[108] Pincus, S. M. (1991). Approximate entropy as a measure of system complexity. Proceedings of the National Academy of Sciences, 88(6), 2297-2301.

[109] Pincus, S. (1995). Approximate entropy (ApEn) as a complexity measure. Chaos: An Interdisciplinary Journal of Nonlinear Science, 5(1), 110-117.

[110] Pincus S.M., Goldberger A.L. Physiological time-series analysis: what does regularity quantify? Am. J. Physiol. Heart Circul Physiol. 1994;266:H1643–H1656. doi: 10.1152/ajpheart.1994.266.4.H1643.

[111] Pincus, S. M., & Huang, W. M. (1992). Approximate entropy: statistical properties and applications. Communications in Statistics-Theory and Methods, 21(11), 3061-3077.

[112] Richman, J. S., & Moorman, J. R. (2000). Physiological time-series analysis using approximate entropy and sample entropy. American Journal of Physiology-Heart and Circulatory Physiology, 278(6), H2039-H2049.

[113] D. Dubois and H. Prade, Fuzzy Sets and Systems: Theory and Applications, New York:Academic Press, 1980.

[114] C. W. de Silva, Intelligent Control Fuzzy Logic Applications, CRC Press, 1995.

[115] L. X. Wang, A Course in Fuzzy Systems and Control, Prentice Hall, 1997.

[116] A. Ramer. Fuzziness vs probability. Int. Journal on general systems, 17:241-248, 1990.

[117] I. Rudas and M. Kaynak. Entropy-based operations on fuzzy sets. IEEE transactions on fuzzy systems, 6(1):33-39, 1998.

[118] F. Criado and T. Gachechiladze. Entropy of fuzzy events. Fuzzy sets and systems, 88:99-106, 1997.

[119] J. Herencia and M. Lamata. Entropy measure associated with fuzzy basic probability assignment. In IEEE int. Conf. On fuzzy systems, volume 2, pages 863-868, 1997.

[120] B. Reecan and D. Markechova. The entropy of fuzzy dynamical systems, general scheme and generator. fizzy sets and systems, 96:191-199, 1998.

[121] A. D. Luca and S. Termini. A definition of non probabilistic entropy in the setting of fuzzy set theory. Information and Control, 20:301-312, 1972.

[122] Sander, W. (1989). On measures of fuzziness. Fuzzy sets and systems, 29(1), 49-55.

[123] Al-Sharhan, Salah & Karray, Fakhri & Gueaieb, W. & Basir, Otman. (2001). Fuzzy entropy: A brief survey. IEEE International Conference on Fuzzy Systems. 3. 1135 - 1139. 10.1109/FUZZ.2001.1008855.

[124] Hausdorff, J. M., Lertratanakul, A., Cudkowicz, M. E., Peterson, A. L., Kaliton, D., & Goldberger, A. L. (2000). Dynamic markers of altered gait rhythm in amyotrophic lateral sclerosis. Journal of applied physiology.

[125] YİĞİT, Ö. E., & Ersoy, Ö. Z. (2022). Feature extraction for DNA capillary electrophesis signals based on discrete wavelet transform combined with multi-scale permutation entropy. Sigma Journal of Engineering and Natural Sciences, 40(3), 475-490.

TEZDEN ÜRETİLMİŞ YAYINLAR

Arapoğlu D., (2022). Nörodejeneratif Hastalıkların Entropi Tabanlı Öznitelikler Kullanılarak Makine Öğrenmesi Yöntemleriyle Sınıflandırılması, EGE 7th INTERNATIONAL CONFERENCE ON APPLIED SCIENCES (pp. 103). İzmir, Türkiye