



T.C.
KONYA TEKNİK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ



EVRIŞİMSEL SİNİR AĞI KULLANARAK
KAN PARAMETRELERİNDEN COVID-19
TESPİTİ

Gizemnur EROL DOĞAN

YÜKSEK LİSANS

Bilgisayar Mühendisliği Anabilim Dalı

Mayıs-2023
KONYA
Her Hakkı Saklıdır

TEZ BİLDİRİMİ

Bu tezdeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

DECLARATION PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Gizemnur EROL DOĞAN

Tarih: 26/05/2023

ÖZET

YÜKSEK LİSANS TEZİ

EVİRİŞİMSEL SİNİR AĞI KULLANARAK KAN PARAMETRELERİNDEN COVID-19 TESPİTİ

Gizemnur EROL DOĞAN

**Konya Teknik Üniversitesi
Lisansüstü Eğitim Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı**

Danışman: Dr. Öğr. Üyesi Betül UZBAŞ

2023, 89 Sayfa

Jüri

**Dr. Öğr. Üyesi Betül UZBAŞ
Prof. Dr. Mustafa Servet KIRAN
Doç. Dr. Hüseyin HAKLI**

COVID-19, çok hızlı yayılabilen pandemik bir solunum yolu hastalığıdır. Enfekte bireylerde COVID-19'un asemptomatik olarak kendini göstermesi, hastalığın yayılımının durdurulamamasına neden olmaktadır. Yayılımın durdurulamamasının diğer bir sebebi olan virüsün çok fazla genomik varyant değiştirmesi ise virüsün bulaşıcılık hızını arttırarak hastalığın yayılımını daha kolay hale getirmektedir. COVID-19'un kısa vadede kesin bir tedavisi mümkün olmadığından temel hedef hastalığın bulaşıcılığını azaltmaktır. Hastalığın bulaşıcılığını azaltmadaki öncelikli adım ise enfekte bireylerin tespiti yapılarak hızlı sonuç almaktır. Evrişimsel Sinir Ağı (ESA), akciğer röntgenleri üzerinden hızlı ve başarılı bir şekilde pozitif bireyleri teşhis ettiğinden COVID-19 çalışmalarında aktif olarak kullanılmaktadır. Buna ek olarak, bulaşıcı hastalıklar hakkında önemli klinik bilgiler içeren ve ulaşılması kolay olan kan parametreleri, COVID-19 teşhisinde önemli bir yere sahiptir.

Bu tez çalışmasında, bireylerin bir boyutlu (1B) kan parametrelerinden iki boyutlu (2B) görüntü verileri elde edilerek ESA ile COVID-19 tespit modeli geliştirilmiştir. Özellik vektörlerinin 2B düzleme aktarılması için T-Dağıtılmış Stokastik Komşu Gömme (T-Distributed Stochastic Neighbor Embedding - T-SNE) yöntemi uygulanmıştır. Görüntü verileri elde etmek için, tüm veriler Dışbükey Gövde Algoritması (DGA) ve Minimum Sınırlayıcı Dikdörtgen (MSD) ile çerçevelenmiş ve piksel haritalama yapılarak elde edilen görüntü verileri, önerilen 3 hatlı ESA mimarisine sunulmuştur. Bu çalışmada, düşük maliyetli ve ulaşılabilirliği hızlı olan kan parametreleri ile görüntü verilerinde yüksek başarı getiren ESA mimarisinin kombinasyonu sağlanarak %94.85 başarı ile COVID-19 tespiti yapılmış, kan parametrelerinden COVID-19 tespiti için etkin ve başarılı bir model önerilmiştir.

Anahtar Kelimeler: COVID-19, 1B Verileri 2B Verilere Dönüştürme, Derin Öğrenme, Evrişimsel Sinir Ağı (ESA)

ABSTRACT

MS THESIS

DIAGNOSIS OF COVID-19 FROM BLOOD PARAMETERS USING CONVOLUTIONAL NEURAL NETWORK

Gizemnur EROL DOĞAN

**Konya Technical University
Institute of Graduate Studies
Department of Computer Engineering**

Advisor: Asst. Prof. Dr. Betül UZBAŞ

2023, 89 Pages

Jury

**Asst. Prof. Dr. Betül UZBAŞ
Prof. Dr. Mustafa Servet KIRAN
Assoc. Prof. Dr. Hüseyin HAKLI**

COVID-19 is a pandemic respiratory disease that can spread very quickly. The asymptomatic appearance of COVID-19 in infected individuals causes the spread of the disease to cannot be stopped. Another reason why the spread cannot be stopped is that the virus changes too many genomic variants, increasing the virulence of the disease and making the spread of the disease easier. Because there is no specific treatment for COVID-19 in the short term, the main goal is to reduce the virulence of the disease. The primary step in reducing the disease risk is to detect infected individuals and get accurate results quickly. Convolutional Neural Network (CNN) is actively used in COVID-19 studies when it quickly and successfully diagnoses positive individuals through lung X-rays. In addition, blood parameters, which contain essential clinical information about infectious diseases and are easy to reach, have an important place in COVID-19 diagnosis.

In this thesis, a COVID-19 detection model based on a CNN was developed by obtaining 2-dimensional (2D) image data from one-dimensional (1D) blood parameters of individuals. The T-Distributed Stochastic Neighbor Embedding (T-SNE) method was applied to transfer the feature vectors to the 2D plane. All data were framed with Convex Hull Algorithm (CHA) and Minimum Bounding Rectangle (MBR) to obtain image data. The image data obtained by pixel mapping was presented to the developed 3 line CNN architecture. This study proposes an effective and successful model by providing a combination of low-cost and rapidly-accessible blood parameters and CNN architecture making image data processing highly successful for COVID-19 detection. Ultimately, COVID-19 detection was made with a success rate of 94.85%.

Keywords: COVID-19, 1D to 2D Conversion, Deep Learning, Convolutional Neural Network (CNN)

ÖNSÖZ

Yüksek lisans eğitimim boyunca bana yol gösteren ve tez çalışmamda kıymetli bilgileri ile yardımlarını esirgemeyen tez danışmanım sayın Dr. Öğr. Üyesi Betül Uzbař'a ve yüksek lisans programı sürecinde tüm destekleriyle bana eşlik eden diđer üniversite hocalarıma teşekkürlerimi sunarım.

Attığım her adımda beni destekleyen ve üzerimde büyük hakları olan aileme ve her zaman yanımda olan sevgili eşime ayrı ayrı teşekkür ederim.

Gizemnur EROL DOĞAN
KONYA-2023

İÇİNDEKİLER

ÖZET	iv
ABSTRACT.....	v
ÖNSÖZ	vi
İÇİNDEKİLER.....	vii
SİMGELER VE KISALTMALAR.....	ixx
1. GİRİŞ.....	1
2. KAYNAK ARAŞTIRMASI	5
2.1. YZ Yöntemleri ile Gerçekleştirilen COVID-19 Çalışmaları.....	5
2.2. 1B Verileri 2B Verilere Dönüştürmek için Gerçekleştirilen Çalışmalar.....	8
3. MATERYAL VE YÖNTEM.....	10
3.1. COVID-19 Hastalığı.....	10
3.2. Veri Seti	12
3.3. Yapay Zeka.....	15
3.3.1. Makine Öğrenmesi.....	16
3.3.1.1. Sınıflandırma Algoritmaları.....	17
3.3.1.1.1. K-En Yakın Komşu Algoritması	18
3.3.1.1.2. Naive Bayes Algoritması	18
3.3.1.1.3. Destek Vektör Makinesi	19
3.3.1.1.4. Karar Ağacı	19
3.3.1.1.4.1. ID3 Algoritması	20
3.3.1.1.5. Kolektif Öğrenme	20
3.3.1.1.5.1. Bagging Algoritması.....	20
3.3.1.1.5.2. Adaboost Algoritması	21
3.3.1.1.5.3. Rasgele Orman Algoritması	22
3.3.2. Yapay Sinir Ağları	23
3.3.2.1. Toplama Fonksiyonu	26
3.3.2.2. Aktivasyon Fonksiyonu	27
3.3.2.3. Sınıflandırma	30
3.3.2.3.1. Hata Fonksiyonu	31
3.3.2.3.2. Geri Yayılım Algoritması	33
3.3.3. Derin Öğrenme	38
3.3.3.1. Evrişimsel Sinir Ağları	39
3.3.3.1.1. Evrişim Katmanı	40
3.3.3.1.2. Havuzlama Katmanı	42

3.3.3.1.3. Tam Bağlantı Katmanı	43
3.4. Sınıflandırma Değerlendirme Metrikleri	43
3.4.1. Hold-Out Yöntemi	43
3.4.2. Rasgele Örneklem Yöntemi	44
3.4.3. k-Katlı Çapraz Doğrulama Yöntemi	44
3.4.4. Üçlü Ayırma Yöntemi	45
3.5. Sınıflandırma Performans Metrikleri	45
3.5.1. Doğruluk	47
3.5.2. Duyarlılık, Kesinlik ve Özgüllük	47
3.5.3. F-Ölçütü	48
3.5.4. Matthews Korelasyon Katsayısı	49
3.5.5. ROC Eğrisi	49
3.6. Veri Madenciliği	51
3.6.1. Veri Ön İşleme	51
3.6.1.1. Eksik Veri Tamamlama	52
3.6.1.1.1. KNN Algoritması ile Eksik Veri Tamamlama	52
3.6.1.2. Özellik Ölçeklendirme	54
3.6.1.2.1. Standardizasyon	54
3.6.1.3. Boyut Azaltma	55
3.7. 1B Verileri 2B Verilere Dönüştürmek için Kullanılan Algoritmalar	56
3.7.1. T-Dağıtılmış Stokastik Komşu Gömme Algoritması	56
3.7.2. Dışbükey Gövde Algoritması ve Minimum Sınırlayıcı Dikdörtgen	59
3.7.3. Gradyan Temelli Rotasyon	62
4. ÖNERİLEN MODEL VE BULGULAR	64
4.1. Veri Ön İşleme	64
4.2. 1B Verileri 2B Verilere Dönüştürme	65
4.2.1. Özellik Vektörünün 2B Kartezyen Düzleme Aktarılması	65
4.2.2. Özellik Vektörünün Çerçeveselenmesi	67
4.2.3. Çerçeveselenen Özellik Vektörünün Dönürülmesi: Rotasyon	68
4.2.4. Kartezyen Koordinatlarının Piksellere Dönüştürülmesi: Piksel Haritalama	68
4.2.4.1. Özellik Vektörünün Normalizasyonu ve Koordinat Hesabı	68
4.2.4.2. Giriş Matrisleri Kullanılarak Görüntü Matrislerinin Elde Edilmesi ...	70
4.3. Önerilen Model	72
5. SONUÇLAR VE ÖNERİLER	79
KAYNAKLAR	81

SİMGELER VE KISALTMALAR

Kısaltmalar

Bilgisayar Destekli Tanı	: BDT
Bilgisayarlı Tomografi	: BT
Çapraz Entropi Kaybı	: ÇEK
Derin Öğrenme	: DÖ
Derin Sinir Ağları	: DSA
Destek Vektör Makinesi	: DVM
Dışbükey Gövde Algoritması	: DGA
Ekstra Ağaç	: EA
Evrimsel Sinir Ağı	: ESA
Geri Yayılım Algoritması	: GYA
General Line Coordinates	: GLC
GLC-Linear	: GLC-L
İki Boyutlu	: 2B
İleri Yayılım Algoritması	: İYA
K-En Yakın Komşu	: KNN
k-Katlı Çapraz Doğrulama	: k-KÇD
Karar Ağacı	: KA
Karmaşıklık Matrisi	: KM
Kolektif Öğrenme	: KÖ
Kullback Liebler	: KL
Lineer Destek Vektör Makinesi	: L-DVM
Lojistik Regresyon	: LR
Makine Öğrenme	: MÖ
Matthews Korelasyon Katsayısı	: MKK
Minimum Sınırlayıcı Dikdörtgen	: MSD
Orta Doğu Solunum Sendromu	: MERS-CoV
Ortalama Kare Hata	: OKH
Ortalama Mutlak Hata	: OMH
Rastgele Orman	: RO
Rektifiye Lineer Birim	: ReLU
Stokastik Gradyan İniş	: SGI
Şiddetli Akut Solunum Sendromu	: SARS-CoV
T-Dağıtılmış Stokastik Komşu Gömme	: T-SNE
TABular Convolution	: TAC
Tam Kan Sayımı	: TKS
Tek Boyutlu	: 1B
Uygulanabilir Ortalama Havuzlama	: UOH
Yapay Sinir Ağı	: YSA
Yapay Zeka	: YZ

1. GİRİŞ

COVID-19, Çin'in Wuhan kentinde 2019'un Aralık ayında ilk kez ortaya çıkan, 4 ay içinde dünyada 200'den fazla ülkede tespit edilerek dünyayı etkisi altına alan ve çok hızlı yayılabilen pandemik bir hastalıktır (Brinati ve ark., 2020). Bu hastalığa ait olan semptomlar, diğer birçok solunum yolu hastalığının da belirtileri olan öksürük, nefes darlığı ve ateş olup COVID-19 teşhisinde etkili olamamaktadır (Paules ve ark., 2020; Menni ve ark., 2020). Bu nedenle, genellikle asemptomatik olarak kendini gösteren COVID-19 (Day, 2020), hastalığı taşıyan ve taşımayan bireyler arasında belirtilerin net olarak seçilememesine neden olduğu için yıllardır bu hastalığın yayılımı durdurulamamaktadır. Buna ek olarak, virüsün çok fazla genomik varyant değiştirmesi, hastalığın daha kolay yayılma yeteneğine sahip olmasını sağlamaktadır (Alkhodari & Khandoker, 2022) ve aşuların etkinliğini değiştirmektedir (Barton ve ark., 2021; Harvey ve ark., 2021). COVID-19'un kısa vadede kesin bir tedavisi mümkün olmadığından temel hedef hastalığın bulaşılabilirliğini azaltmak olmuştur. Hastalığın bulaşılabilirliğini azaltmadaki öncelikli adım ise enfekte bireylerin tespit edilmesi ve bu amaç doğrultusunda gerçekleştirilen tespit çalışmalarından hızlı sonuç alınmasıdır.

İşlenmesi insan gücünü aşan büyük veri kümeleriyle başa çıkabilme yeteneği ve klinik karar desteği sağlamasından dolayı sağlık hizmetlerinde Yapay Zeka (YZ) aktif olarak kullanılmaktadır (Liang & Zheng, 2020; Davenport & Kalakota, 2019). Sağlık alanında kullanılan, uzmanların iş yükünü hafifleten ve güvenilirliği artıran YZ tabanlı Bilgisayar Destekli Tanı (BDT) sistemi, tıbbi verilerdeki anormal noktaları tespit ederek uzmanlara bir tanı sunar ve uzmanların bu tanıları kendi yorumlarıyla kısa süre içinde neticelendirmesine yardımcı olur (Gaál ve ark., 2020). Kalp rahatsızlıkları (Jahmunah ve ark., 2021; Chang ve ark., 2022), uyku bozuklukları (Zarei ve ark., 2022), kanser (Munir ve ark., 2019) ve bulaşıcı virüs hastalıkları (Andriasyan ve ark., 2019) başta olmak üzere birçok hastalık için aktif olarak kullanılan etkili bir sistemdir. BDT sayesinde uzman yükü hafifler, uzmanların muayene edebileceği hasta kapasitesi artar ve tanı işlemleri hızlı bir şekilde sonuçlanacağından hastaların tedavi aşamasına hızlıca geçilebilir. Bundan dolayı, yaşanan bu ciddi boyuttaki COVID-19 salgınıyla BDT sisteminin sağlık hizmetlerindeki önemini daha da ortaya çıkarmış ve tıbbi görüntülemeye dayalı BDT sistemlerine ilgi artmıştır (Jaiswal ve ark., 2019; Liang & Zheng, 2020).

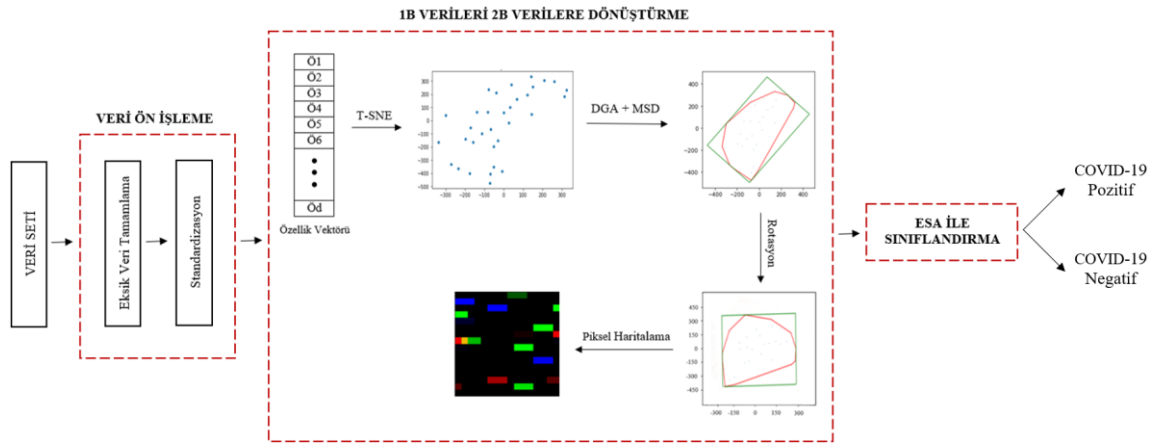
BDT'de Derin Öğrenme (DÖ) teknikleri, birçok hastalığın teşhisinde kullanılan birincil yöntemlerdir (Zhang ve ark., 2018). Son yıllarda kullanılan DÖ teknikleri,

birçok alanda olduğu gibi medikal alanda da oldukça başarılı performans sergilemektedir. DÖ modelleri yardımıyla manyetik rezonans görüntüleme, Bilgisayarlı Tomografi (BT) ve röntgen gibi tıbbi görüntüleme teknikleri ile görüntü ve sinyal verilerinin analizi gerçekleştirilmektedir ve bu analizler sonucunda da kanser ve tümör gibi birçok ciddi hastalıkların tespiti yapılmaktadır (Dorj ve ark., 2018; Shen ve ark., 2019; Saba ve ark., 2020). DÖ yöntemlerinin başarısı birçok hastalığın teşhisinde bu denli ortadayken COVID-19 tespit çalışmalarında da görüntü işleme teknikleri ile entegre edilmiş DÖ sistemleri aktif olarak kullanılmaktadır (Hakak ve ark., 2020; Iwendi ve ark., 2020). ESA, resim ve video gibi görsel veri işleme için geliştirilmiş özel bir Yapay Sinir Ağı (YSA) mimarisi olup oldukça geniş çalışma alanlarında başarılı sonuçlar getirmektedir (LeCun ve ark., 1989). ESA çok başarılı bir DÖ mimarisi olmasına rağmen, sadece 2B veriler üzerinde çalışabilmektedir. 1B büyük veri setleri ile geliştirilebilecek modeller için bu denli başarılı ve doğruluk oranı yüksek DÖ modelinin kullanılamaması, büyük bir dezavantaj olarak görülmektedir (Sharma ve ark., 2019; Sharma & Kumar, 2020).

COVID-19 enfekte bireylerin tespiti için güncel olarak kullanılmakta olan COVID-19 tespit yöntemleri: RT-PCR, BT, X-ray görüntüleri ve belirli kan parametreleridir. Yaygın olarak sürüntü testi olarak da bilinen RT-PCR, laboratuvar ortamında solunum örnekleri üzerinden COVID-19 hastalığının teşhisi amacıyla en çok kullanılan yöntemdir (Brinati ve ark., 2020). Dünya genelinde virüsün yüksek bir hızla yayılması nedeniyle RT-PCR testlerine üst düzey bir talep olmuştur. Bu talep artışı, teşhisin ortalama 2-3 saatte yapılabilmesi, sertifikalı laboratuvarlara, eğitimli personellere ve pahalı ekipmanlara ihtiyaç duyulması gibi RT-PCR yönteminin birçok kısıtlamalarını dezavantaj olarak ortaya çıkarmıştır (Li ve ark., 2020). Bunun yanı sıra, X-ray görüntüleme dünya çapında sağlık alanında ve özellikle akciğer iltihaplanması tanısında kullanılan en önemli yöntemlerden biridir (Ayan & Ünver, 2019; Jaiswal ve ark., 2019; Narayan Das ve ark., 2020). Dolayısıyla X-ray görüntüleme de, COVID-19 tespitinde kullanılan etkili yöntemlerden biri olmuştur. Hasta bireylerin BT taramaları incelendiğinde ise neredeyse tüm COVID-19 hastalarının benzer özellikler sergilediği gözlemlenmiştir (Alazab ve ark., 2020). COVID-19 tespitinde, BT taramaları ve X-ray görüntüleri kullanıldığında doğruluğu yüksek sonuçlar getirmiştir (Wu & McGoogan, 2020). Bu sebeplerden dolayı, literatürdeki COVID-19 tespit çalışmalarının çoğunun BT taramaları veya X-ray görüntüleri ile gerçekleştirildiği gözlemlenmiştir (Mei ve ark., 2020; Wynants ve ark., 2020). COVID-19 tespitinde oldukça kullanışlı bir yöntem

gibi görünen X-ray görüntülüne, bir hastalığın tanısını koymak için oldukça meşakkatli bir yöntemdir (Antin ve ark., 2017). X-ray görüntülerine bakarak COVID-19 tanısı koymak, uzman bilgisi ve deneyimi gerektirir ve dolayısıyla bu tanı ancak uzman doktorlar tarafından konulabilir (Jaiswal ve ark., 2019). Dünya üzerinde normal zamanlarda bile uzman sayısı, normal doktor sayısından daha azdır ve çoğu ülkelerde kişi başına düşen doktor sayısı da zaten yetersizdir (Narin ve ark., 2021). Bunun yanı sıra, BT ve X-ray hem pahalı hem de oldukça zaman alıcı yöntemlerdir. COVID-19 teşhisinde bu yöntemlere kıyasla daha az kullanılan kan parametreleri, BT ve X-ray'e göre maliyeti düşük, ulaşılması kolay ve hızlı sonuç getirmektedir. Buna ek olarak, kan parametreleri bulaşıcı hastalıklar hakkında önemli klinik değerlere sahiptir (Demirdal & Sen, 2018; Jan ve ark., 2019). Fakat güncel olarak gerçekleştirilen çalışmalara bakıldığında, COVID-19 teşhisinde kan parametreleri ile gerçekleştirilen çalışmalar oldukça azdır.

Bu tez kapsamında gerçekleştirilen çalışmanın genel diyagramı Şekil 1.1'de sunulmuştur. Şekil 1.1'de görüldüğü üzere, ilk olarak COVID-19 pozitif ve negatif bireylerin kan parametre bilgilerini içeren veri setine belirli veri ön işlemler uygulanmıştır: K-En Yakın Komşu (K-Nearest Neighbours - KNN) yöntemi ile eksik verileri tamamlanmış (Erol ve ark., 2022) ve $[-1 \ 1]$ aralığında standardizasyonu gerçekleştirilmiştir. Sonrasında, veri ön işlemlerden geçirilmiş 1B kan parametreleri, 2B görüntü verilerine dönüştürülmüştür. Dönüştürme işlemi için Sharma ve ark. (2019) tarafından geliştirilen, DeepInsight adlı 1B verileri 2B verilere dönüştürme metodu kullanılmıştır. Bu dönüştürme işlemi için uygulanan adımlar şu şekildedir: Veri setindeki her bir özellik vektörü T-SNE yöntemi ile 2B düzleme aktarılmıştır. Bu yöntem sayesinde, veri setinde bulunan özellik vektörlerindeki benzer özellikler bir araya, farklı olanlar daha uzağa konumlanmıştır. Sonrasında görüntü verisinin sınırlarını oluşturmak için 2B düzleme aktarılan veriler, DGA ve MSD ile çerçevelenmiştir. ESA mimarisi için görüntünün yatay ve dikey olarak çerçevelere sahip olması gerektiğinden döndürme işlemi uygulanmış ve piksel haritalama yapılarak görüntü matrisleri elde edilmiştir. Elde edilen görüntü matrisleri, RGB formata dönüştürülerek görüntü verileri elde edilmiştir. Son olarak, elde edilen görüntü verileri önerilen ESA mimarisi ile sınıflandırılmıştır.



Şekil 1.1. Tez Kapsamında Gerçekleştirilen Çalışmanın Genel Diyagramı

Bu tez kapsamında, COVID-19 pozitif ve negatif bireylerin kan parametrelerinden görüntü verileri elde ederek önerilen ESA mimarisi ile DÖ'ye dayalı bir COVID-19 tespit modeli sunulmuştur. Bu sayede, COVID-19 tespiti için maliyeti düşük ve ulaşılabilirliği hızlı olan kan parametreleri ile görüntü verilerinde yüksek başarı getiren ESA mimarisinin kombinasyonu sağlanmıştır. Bu tez çalışmasında, ESA mimarisi kullanılarak 1B COVID-19 kan parametrelerinden enfekte bireylerin tespiti yapılarak literatüre yeni bir bakış açısı kazandırılmıştır.

Bu tez çalışmasının geri kalanı şu şekilde düzenlenmiştir: Literatürde, COVID-19 tespiti için gerçekleştirilen YZ çalışmaları ve 1B verileri 2B verilere dönüştürmek için geliştirilen çalışmaların incelemesi ikinci bölümde yapılmıştır. Üçüncü bölümde, COVID-19 hastalığının, kullanılan veri setinin, YSA ve ESA'nın detaylı anlatımı sunulmuştur. Yine aynı bölümde, bu tez kapsamında kullanılan algoritmalar anlatılmıştır. Dördüncü bölümde, 1B COVID-19 kan parametre verilerini 2B görüntü verilerine dönüştürmek için uygulanan yöntemler, önerilen ESA modeli hakkında bilgiler verilmiştir. Önerilen ESA modelinin başarısı, popüler ESA modelleri, çeşitli Makine Öğrenme (MÖ) modelleri ve bu tez çalışmasında kullanılan veri seti ile literatürde gerçekleştirilmiş diğer COVID-19 çalışmaları ile karşılaştırması yine dördüncü bölümde sunulmuştur. Beşinci bölümde ise araştırmanın sonuçları özetlenmiş ve önerilerde bulunulmuştur.

2. KAYNAK ARAŞTIRMASI

Bu bölümde, YZ yöntemleri kullanılarak COVID-19 verileri üzerinde yapılan çalışmalar ve 1B veriler ile ESA mimarisinin kullanımını arttırmak için literatürde önerilmiş olan 1B verileri 2B verilere dönüştürme metotları incelenmiş ve başarıları sunulmuştur.

2.1. YZ Yöntemleri ile Gerçekleştirilen COVID-19 Tespit Çalışmaları

COVID-19 pandemisi gibi afetlerde sağlık hizmetlerinin aynı anda yoğun talep altında olması, uzman sayısı ve sağlık altyapılarındaki yetersizlikten dolayı sağlık sistemlerinin çökmesine neden olmaktadır. COVID-19 gibi ölümcül bulaşıcı hastalıklar, en çokta sağlık personellerini risk altında tutmaktadır. Bu nedenle, sağlık personellerini korumaya alacak ve sağlık hizmetlerini her koşulda aktif tutacak YZ tabanlı sistemlerin var olması oldukça önemlidir. Literatür incelendiğinde, COVID-19 teşhisinde geliştirilen çalışmalar BT ve X-ray görüntüleri üzerinden veya kan parametreleri üzerinden gerçekleştirilmiştir.

Literatürde BT ve X-ray görüntüleri üzerinden gerçekleştirilen COVID-19 tespit çalışma incelendiğinde; Jiang ve ark. (2020), şiddetli solunum ve ateş semptomları olan 51 hastanın COVID-19 tespiti için RT-PCR ile BT ve X-ray görüntülerini kıyaslayan bir çalışma gerçekleştirmiştir. Bu çalışma sonucunda, %71'lik RT-PCR testi duyarlılığına karşın, göğüs BT taramalarından %98 duyarlılık elde edilmiştir ve COVID-19 yayılımını önlemek için otomatik tespit sistemlerinin geliştirilmesinin önemi vurgulanmıştır. Öztürk ve ark.'nın (2020) X-ray görüntüleri ile gerçekleştirdiği COVID-19 tespiti yapan DarkCovidNet adlı DÖ modeli, COVID-19 pozitif ve negatif bireyleri sınıflandırırken akciğer iltihaplanmasına sahip bireyleri de sınıflandırmaya dahil etmiştir. COVID-19 pozitif ve negatif bireyler arasında yapılan sınıflandırmadan %98.08 başarı, COVID-19 pozitif ve akciğer iltihabı hastalığına sahip hastalar arasında yapılan sınıflandırmadan ise %87.02 başarı elde edilmiştir. Narin ve ark.'nın (2021) gerçekleştirdiği çalışmada ise COVID-19 tespiti için beş farklı ESA modeli sunulmuştur. Bu modellere, X-ray görüntüleri girdi olarak verilmiş ve ResNet50 model, %98 başarı getirerek en yüksek performanslı model olarak seçilmiştir. Xu ve ark. (2020), akciğer BT taramaları kullanarak COVID-19 akciğer iltihaplanmasını grip vakalarından ayırt eden, %86.7 doğruluk oranına sahip ResNet tabanlı bir DÖ modeli geliştirmişlerdir. Jaiswal ve ark.'nın (2019) gerçekleştirdiği çalışmada, göğüs BT'leri ile önceden eğitilmiş DÖ mimarileri kullanılarak enfekte olan bireylerin tespiti için

DenseNet201 modeli önerilmiştir. COVID-19 tespiti için COVID-Net modelini sunan Wang ve Wong (2020), bu model ile %92.4 başarı elde etmiştir. Bassi ve ark. (2022), COVID-19 tespiti için göğüs X-ray görüntülerini içeren veri seti üzerinden iki transferli öğrenme yaklaşımı sunmuştur. Bu çalışmada, öğrenme aktarımı tekniğini kullanarak COVID-19 tespitinde literatüre farklı bir bakış açısı kazandırmıştır. Haghaniyar ve ark. (2022), X-ray görüntü verilerine akciğer segmentasyonu ve farklı görüntü iyileştirme ön işlemleri uygulayarak öğrenme aktarım tekniği ile 431 katmanlı COVID-CXNet adlı DÖ modeli ile %96.10 başarı elde etmişlerdir.

COVID-19 için diğer bir teşhis yöntemi olan kan parametreleri, bulaşıcı hastalıklar hakkında önemli klinik değerlere sahiptir (Demirdal & Sen, 2018; Jan ve ark., 2019). Bundan dolayı, COVID-19 teşhisinde kan parametreleri tıp uzmanları tarafından kullanılan önemli verilerdir. Bu amaçla, kan parametreleri ile gerçekleştirilen COVID-19 tespit çalışmaları incelendiğinde; Sun ve ark., SPSS programını kullanarak COVID-19 teşhisinde Lökosit, Monosit ve Trombosit gibi kan değerlerinin hastalık için ayrımını istatistiksel olarak incelemişlerdir (Sun ve ark., 2020). Guan ve ark. (2019), 1000 adet COVID-19 hastasının klinik kan verilerini inceleyerek bireylerin kan değerlerinden anamoli tespiti yapmışlardır. Bunlara ek olarak, COVID-19 pozitif ve negatif bireylerin karşılaştırmalı olarak beyaz kan hücreleri, trombositler ve C-reaktif proteinler gibi birçok kan parametre değerleri analiz edilmiştir (Ferrari ve ark., 2020). Gerçekleştirilen çalışmalar, kan değerlerinin etnik ve genetik farklılıklar gösterebildiğini ortaya koymuştur (Sarnaik ve ark., 1999). Bu nedenle, COVID-19 tespitinde dünyanın farklı bölgelerindeki toplumlara göre, genetik yapılarına uygun farklı kan parametre grupları belirlemek hastalık teşhisinde belirleyici etmen olabilmektedir. COVID-19 teşhisi için belirli bir toplumun kan parametrelerini içeren bir veri seti ile geliştirilen YZ modeline farklı bir toplumun kan değerleri sunulursa genetik farklılıklardan doğan kan değerlerindeki değişikliklere bağlı olarak, geliştirilen YZ modeli farklı sonuçlar verecektir. Örneğin, bir Avrupa ülkesinden elde edilen COVID-19 hastalarının kan değerleri, YZ tabanlı COVID-19 tanı sisteminde Monosit-Lenfosit oranı en önemli bir parametre iken Asya genetiğinde bu parametrenin, Nötrofil-Lenfosit olduğu gözlemlenmiştir (Göreke ve ark., 2021; Guan ve ark., 2019). Bu durum bir dezavantaj gibi görünse de, COVID-19 kan parametrelerindeki genetik farklılıklar göz önünde bulundurularak geliştirilecek YZ tabanlı COVID-19 tanı sistemleri hem başarıyı hem de hassasiyeti oldukça arttıracaktır.

COVID-19 kan parametreleri kullanılarak gerçekleştirilen diğer çalışmalar incelendiğinde, Brinati ve ark.'ları (2020), 279 örneklili veri setine belirli veri ön işlemler uygulamış, bütün özellikleri içeren ve cinsiyet özelliğini hariç tutan iki tane MÖ modeli tasarlamıştır. Tasarlanan modeller için en başarılı sonuçlar Rastgele Orman (RO) sınıflandırıcısı ile sırasıyla %82 ve %86 olarak elde edilmiştir. ERLX modelini geliştiren Aljame ve ark.'ları (2020) ise COVID-19 için iki seviyeli bir Kolektif Öğrenme (KÖ) modeli geliştirilmiştir. 5644 hastanın kan verilerinden oluşan veri setine çeşitli veri ön işlemlerinin uygulanmasının ardından Ekstra Ağaç (EA), RO ve Lojistik Regresyon (LR) algoritmaları ile ilk seviyeli sınıflandırma, daha sonra da XGBoost ile ikinci seviye sınıflandırma gerçekleştirerek bu model ile %99.88 başarı elde etmişlerdir. Cabitza ve ark. (2020) gerçekleştirdikleri çalışmada, COVID-19 pozitif ve negatif bireylerin kan parametre bilgilerini içeren OSR veri setini COVID-19'a özgü kan parametrelerini içeren veri seti ve Tam Kan Sayımı (TKS) için özel kan parametrelerini içeren veri seti olmak üzere ikiye bölmüştür. Ardından, COVID-19 tespiti için bu üç veri seti üzerinden sınıflandırma çalışmaları yapmışlardır. Sunmuş oldukları beş farklı MÖ modeli üzerinde bu üç veri seti çalıştırılarak en başarılı sonuçlar elde edilmiştir. Alakuş ve Türkoğlu (2020), COVID-19 tespitine yönelik altı adet DÖ modelini 600 örneklili kan verileri üzerinden karşılaştırmıştır ve CNNLSTM modeli, %92.30 başarı ile en başarılı model seçilmiştir. Göreke ve ark.'nın (2021) gerçekleştirdikleri çalışmada ise kan değerlerindeki genetik farklılıklar dikkate alınarak bir COVID-19 tespit modeli geliştirilmiştir. Bu amaç doğrultusunda, genetik farklılıkları dikkate alan, kan parametrelerine dayalı yeni özellik grubu çıkaran bir algoritma geliştirilmiştir. Daha sonra bu özellik grubunu kullanılarak DÖ'ye dayalı yeni bir hibrit sınıflandırıcı modeli tasarlanarak %94.95 başarıya sahip COVID-19 tespiti gerçekleştirilmiştir. Rikan ve ark.'nın (2022) gerçekleştirdiği çalışmada, kan parametreleri içeren üç veri seti ile COVID-19'un tespiti için 8 MÖ ve 4 DÖ modeli geliştirilmiştir. Geliştirilen modellerden en yüksek, MÖ modeli olan EA ile %84.98 ve Derin Sinir Ağları (DSA) ile %93.16 en yüksek başarı elde edilmiştir. Abayomi-Alli ve ark. (2022) tarafından gerçekleştirilen çalışmada ise birinci aşama sınıflandırıcı olarak ESA modelleri, ikinci aşama sınıflandırıcı olarak denetimli MÖ algoritması kullanılarak ESA ve EA algoritmasına dayalı KÖ modeli ile %99.28 başarı elde edilmiştir. Rahman ve ark. (2022) tarafından geliştirilen QCovSML adlı MÖ modeli ise gradyan arttırma, RO ve XGBoost sınıflandırıcılarından oluşan bir yığın MÖ modelidir. 3 sınıflandırıcı ile geliştirilen bu yığın modeli ile %91.45 başarı elde edilmiştir. Hu ve ark.'larının (2022)

gerçekleştirdiği çalışmada ise diğer çalışmalardan farklı olarak TKS parametreleri kullanmak yerine kısmi oksijen basıncı, oksijen doygunluğu ve sodyum iyon konsantrasyonu gibi kan gazı parametreleri kullanılmıştır. Bu çalışma sonucunda, COVID-19 şiddetinin erken ve doğru bir şekilde değerlendirilmesi için İkili Harris Şahin Optimizasyon algoritması, speküler yansıma öğrenimi yöntemi ile geliştirilerek bir özellik seçme yöntemi sunulmuştur. Son olarak, Kistenev ve ark.'ları (2022) tarafından COVID-19 pozitif ve negatif bireylerin kan parametreleri ile gerçekleştirilen çalışmada ise RO ve Destek Vektör Makinesi (DVM) sınıflandırıcıları ile %88.00 başarı elde edilmiştir.

2.2. 1B Verileri 2B Verilere Dönüştürmek için Gerçekleştirilen Çalışmalar

YSA, çeşitli veri türlerini sınıflandırmada büyük ilerleme göstermiştir. Özellikle, görüntü verilerini sınıflandıran ESA'nın gerçekleştirdiği bazı görüntü sınıflandırma çalışmalarının doğruluğu insan düzeyindedir, hatta bazen bu düzeyi bile aşabilmektedir (Wu ve ark., 2015). Buna karşın, YSA'nın 1B verileri sınıflandırmadaki ilerlemesi oldukça az olmuştur. ESA'nın başarısı bu denli ortadayken, görüntü olmayan verileri görüntülere dönüştürerek bilgi aktarımını sağlamak çok çeşitli MÖ problemlerini çözmek için bir fırsattır (Dovhalets ve ark., 2018; Sharma ve ark., 2019). Bu şekilde, görüntü olmayan bir sınıflandırma görevi, güçlü bir ESA mimarisi tarafından çözülecek olan görüntü tanıma görevine dönüştürülür. Bu amaç doğrultusunda, 1B verileri 2B görüntü verilerine dönüştürmek için çeşitli metotlar geliştirilmiştir. Literatürde bu metotların geliştirildiği çalışmalar incelendiğinde; Buturovic ve Miljkovic (2020) tarafından geliştirilen Tabular Convolution (TAC) metodu, sabit bir temel görüntü ve filtre olarak işlenen özellik vektörlerinin konvolüsyonunu kullanarak tablo verisi olarak nitelendirdiği 1B verileri, 2B görüntü verilerine dönüştürür. Filtrelenen görüntüler daha sonra bir ESA kullanılarak sınıflandırılır. Bu çalışmada, bakteriyel veya viral bir enfeksiyonun olup olmadığını teşhis etmek amacıyla, bulaşıcı hastalığı olan 2.590 hastanın kan örneğini içeren gen ifadesi verileri TAC metodu ile görüntü verilerine dönüştürülerek ResNet ile sınıflandırılmıştır. Buna ek olarak, veri seti görüntü verilerine dönüştürülmeden çeşitli popüler MÖ modelleri ile sınıflandırılarak TAC metodunun başarısı değerlendirilmiştir. Bu çalışma, orijinal veri setini girdi olarak alan Lineer Destek Vektör Makinesi (L-DVM) ile %89.60 başarı, görüntü verilerine dönüştürülmüş veri setini girdi olarak alan ResNet modeli ile %91.10 başarı elde etmiştir. Sharma ve ark. (2019), tarafından önerilen DeepInsight metodunda, özellik vektöründeki benzer

özellikleri bir araya farklı olanları 2B düzlemde daha uzağa yerleştirerek özellik vektörlerinden görüntü matrisleri elde eder. Elde ettiği görüntü matrislerini de RGB formata dönüştürerek görüntü verilerini elde eder. Bu çalışmada, 5 farklı 1B veri seti 2B görüntü verilerine dönüştürülmüş ve bu görüntü verileri önerdikleri paralel yapı bir ESA mimarisi ile sınıflandırılmıştır. 1B verilerden 2B verilere dönüştürülen görüntüleri girdi olarak alan ESA mimarisinin başarısı, 1B veri setinin orijinal halini girdi olarak alan Karar Ağacı (KA), RO ve Adaboost MÖ sınıflandırıcıları ile karşılaştırmalı olarak değerlendirilmiştir. Gen ifadelerini içeren 60483 özellikli veri seti ile RO algoritması %96 başarı elde ederken yine aynı veri setinin DeepInsight metodu ile görüntü verilerine dönüştürülmüş halini girdi olarak alan ESA mimarisi %99 başarı elde etmiştir. Bu da Sharma ve ark.'nın (2019) önerdikleri metodun başarısını açıkça ortaya koymuştur. Kovalerchuk ve ark. (2017) tarafından geliştirilen General Line Coordinates (GLC) metotlarından biri olan GLC-Linear (GLC-L) metodu, n boyutlu veri noktalarını, herhangi bir bilgi kaybetmeden çoklu çizgiler olarak görsel temsiline dönüştürülmesine olanak tanır (Dovhalets ve ark., 2018). GLC-L, çok boyutlu verileri 2B düzlemde görmeyi sağlayan çoklu çizgilere yansır. Tek bir n-D noktasını görselleştirmek için GLC-L'nin kullanılması, bir görüntü olarak yorumlanabilen çoklu çizgileri üretir. İşlem, farklı sınıflardan n-D noktalarının bir dizi görüntüsünü oluşturarak verilen her n-D noktası için tekrarlanır. 1B verileri yani veri setinin orijinal halini girdi olarak alan 1-DC adlı ESA mimarisi ve dönüştürülmüş görüntü verilerini girdi olarak alan ayrı bir ESA mimarisi geliştirilmiştir. Sonuçta, 7 farklı veri setinin orijinal hali 1-DC ile GLC-L metodu ile görüntü verilerine dönüştürülmüş hali diğer geliştirilen ESA mimarisi ile sınıflandırılarak GLC-L metodunun başarısı karşılaştırmalı olarak sunulmuştur. Kovalerchuk ve ark.'nın (2020) geliştirdiği CPC-R metodu, yüksek boyutlu 1B bilgileri 2B olarak korur. Bir n-D noktasının özniteliklerini, 2B kartezyen koordinat değerlerinin çiftlerine ayırır ve bu çiftleri aynı 2B kartezyen koordinatlarında 2B noktalar olarak görselleştirir. Ardından, bir ısı haritası oluşturan çiftlerin sırasını kodlamak için çiftleri gri tonlama veya renk yoğunluğu değerlerine eşler. Bu çalışmada, CPC-R metodu ile görüntü verilerine dönüştürülen birçok veri seti çeşitli ESA modelleri ile sınıflandırılmış ve yüksek başarılar elde edilmiştir.

3. MATERYAL VE YÖNTEM

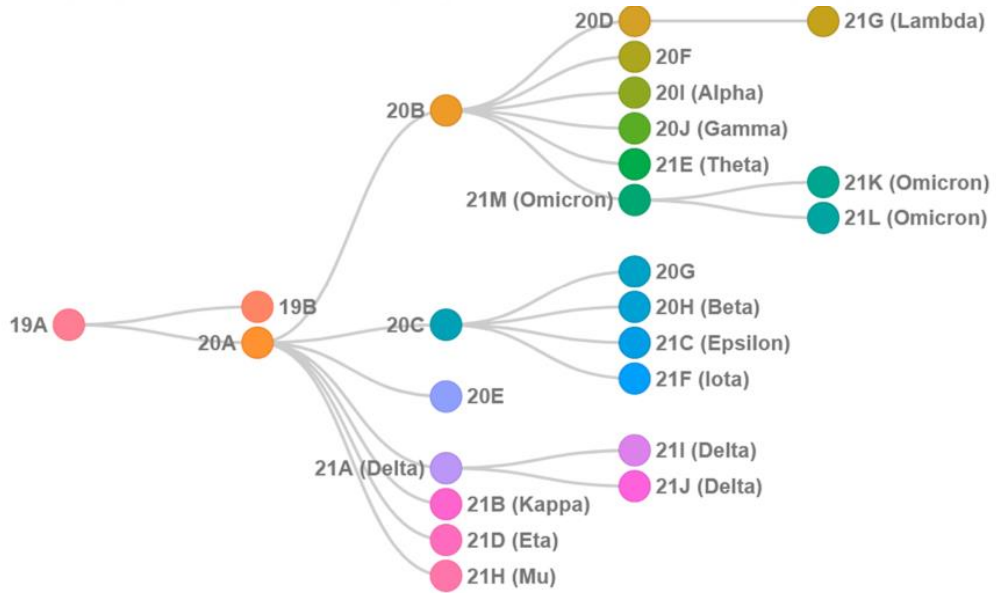
Bu bölümde ilk olarak, COVID-19 hastalığından ve kullanılan veri setinden bahsedilmiştir. Devamında ise, YZ yöntemlerine genel bir bakış yapılmış ve tez çalışmasında kullanılan yöntemler detaylı bir şekilde anlatılmıştır.

3.1. COVID-19 Hastalığı

COVID-19, Çin'nin Wuhan kentinde 2019'un aralık ayında ilk kez ortaya çıkan, 4 ay içinde dünyada 200'den fazla ülkede tespit edilerek dünyayı etkisi altına alan ve çok hızlı yayılabilen pandemik bir solunum yolu hastalığıdır (Brinati ve ark., 2020). Bu hastalığın üyesi olan koronavirüs, Orta Doğu Solunum Sendromu (MERS-CoV) ve Şiddetli Akut Solunum Sendromu (SARS-CoV) gibi çeşitli hastalıklara neden olan geniş bir virüs ailesidir (Narin ve ark., 2021). Bu virüs ailesi, solunum yolu, diyabet, kanser ve kalp hastalığı gibi kronik rahatsızlıklara sahip kişilerde ciddi etkilere neden olarak kalıcı rahatsızlıklara ve hatta ölüme neden olabilmektedir (WHO, 2020). Dünya Sağlık Örgütü'nün Mayıs 2022 raporuna göre, pandeminin ortaya çıkmasından bu yana yaklaşık olarak 767 milyon onaylanmış vaka ve 6 milyondan fazla ölüm gerçekleşmiştir (WHO, 2022). Bildirilen vakaların çoğunda ise virüs, enfeksiyon semptomları ortaya çıkmadan önce 1-14 gün arasında bir süre kuluçkada kalır (WHO, 2019) ve COVID-19 taşıyan hastalardaki bu semptomlar öksürük, nefes darlığı, ateş ve diğer akut solunum sıkıntısı sendromları olarak kendini göstermektedir (Paules ve ark., 2020; Menni ve ark., 2020). Diğer birçok solunum yolu hastalığının da belirtileri olan bu semptomlar COVID-19 teşhisinde etkili olmamaktadır. Bu nedenle, genellikle asemptomatik olarak kendini gösteren COVID-19 (Day, 2020), pozitif ve negatif bireyler arasında belirtilerin net olarak seçilememesine neden olduğu için yıllardır bu hastalığın yayılımı durdurulamamaktadır.

COVID-19' un kökeni, pandeminin başlangıcından bu yana kapsamlı bir şekilde tartışılmakta ve toplanan genetik ve epidemiyolojik veriler bu salgının zoonotik bir olay sonucu ortaya çıktığına işaret etmektedir (Rasmussen, 2021; Holmes ve ark., 2021; Zhou ve ark., 2021). Günümüzde COVID-19 ile mücadelede en büyük gelişme, COVID-19'un ilk tam genom dizisinin paylaşılması ile aşı formülasyonlarının başlaması ve bunun sonucu hastalık için birçok aşının üretilmesidir (Alexandridi ve ark., 2022). Fakat yayılımın durdurulamamasının diğer bir temel sebebi olan virüsün çok fazla genomik varyant değiştirmesi, hastalığın daha kolay yayılma yeteneğine sahip

olmasını sağlar (Alkhodari & Khandoker, 2022). Geliştirilen aşılar sayesinde, enfekte olan bireylerde hastalığın etkileri hafiflemiş gibi gözükse de virüsün çok fazla genomik varyant değiştirmesi, aşuların etkinliğini değiştirerek hastalığın yayılmasına engel olamamaktadır (Barton ve ark., 2021; Harvey ve ark., 2021). İlk olarak Hindistan’da tespit edilen Delta varyantı 193’den fazla ülkeye yayılmış ve %225 daha bulaşıcı bir virüs haline gelmiştir (Callaway, 2021). Delta varyantında hemen sonra ortaya çıkan ve ilk kez Güney Afrika’da görülen Omicron varyantının ise Delta varyantından 70 kat daha hızlı çoğalabildiği gözlemlenmiştir (Papanikolaou ve ark., 2022). Şekil 3.1’de görüldüğü üzere, Delta ve Omicron gibi COVID-19’dan türeyen daha birçok genetik varyant olup insan popülasyonunda hızla yayılmıştır.



Şekil 3.1. COVID-19’dan Türeyen Temel Genetik Varyantlar Arasındaki Evrimsel İlişkiler (Alexandridi ve ark., 2022)

Yayılmı hızı durdurulamayan bu hastalığın, bireysel sağlığı etkilemesine ek olarak ülkeleri ekonomik, sosyal ve altyapılandırma sistemleri bakımından da oldukça etkilemiştir (Dey ve ark., 2020; Fong ve ark., 2020). Bu yüzden, COVID-19’un kısa vadede kesin bir tedavisi mümkün olmadığından temel hedef hastalığın bulaşıcılığını azaltmak olmuştur. Hastalığın bulaşıcılığını azaltmadaki öncelikli adım ise enfekte bireylerin tespit edilmesi ve bu amaç doğrultusunda gerçekleştirilen tespit çalışmalarından hızlı sonuç alınmasıdır. COVID-19 taşıyıcısı kişilerin belirlenmesinde aktif rol oynayan ve yaygın olarak sürüntü testi olarak da bilinen RT-PCR, laboratuvar ortamında solunum örnekleri üzerinden COVID-19 hastalığının teşhisi amacıyla en çok

kullanılan yöntemdir (Brinati ve ark., 2020). Dünya genelinde virüsün yüksek bir hızla yayılması nedeniyle RT-PCR testlerine üst düzey bir talep olmuştur. Bu talep artışı, teşhisin ortalama 2-3 saatte yapılabilmesi, sertifikalı laboratuvarlara, eğitimli personellere ve pahalı ekipmanlara ihtiyaç duyulması gibi RT-PCR yönteminin birçok kısıtlamalarını (Li ve ark., 2020) dezavantaj olarak ortaya çıkarmıştır. Bu sebeple, gün geçtikçe COVID-19 tespit testlerine artan taleple beraber daha hızlı, alternatif teşhis yöntemlerine acil bir ihtiyaç doğmuş ve YZ tabanlı COVID-19 tespit sistemlerinin geliştirilmesi acil bir durum haline gelmiştir. Buna ek olarak, COVID-19 pandemisi gibi afetlerde sağlık hizmetlerinin aynı anda yoğun talep altında olması, uzman sayısı ve sağlık altyapılarındaki yetersizlikten dolayı sağlık sistemlerinin çökmesine neden olmaktadır ve COVID-19 gibi ölümcül bulaşıcı hastalıklar, en fazla sağlık personellerini risk altında tutmaktadırlar. Bu nedenle, sağlık personellerini korumaya alacak ve sağlık hizmetlerini her koşulda aktif tutacak YZ tabanlı sistemlerin var olması oldukça önemlidir.

3.2. Veri Seti

Bu tez çalışmasında, İtalya'nın Milano şehrindeki San Raffaele Hastanesinde COVID-19 şüphesi ile kabul edilen 1736 hasta bireyin kan parametrelerini içeren OSR (Cabitza ve ark., 2020) veri seti kullanılmıştır. Bu veri seti, her hastanın yaş, cinsiyet ve belirli kan parametrelerini içermektedir. Ayrıca COVID-19 şüphelisi olan bu hastaların sürüntü test sonuçları olan, 816'sı pozitif 920'si negatif olan RT-PCR sonuçlarını içermektedir. OSR veri seti, hem COVID-19 için özel olan kan parametrelerini hem de birçok hastalığın kontrolünde tarama testi olarak kullanılan TKS parametreleri olmak üzere, toplam 33 özelliğe sahiptir. Veri setinde bulunan özellikler, Tablo 3.1'de verilmiştir.

Tablo 3.1. OSR Veri Seti

ÖZELLİK	AKRONİM	COVID-19 KAN PARAMETRELERİ	TKS KAN PARAMETRELERİ
Yaş	-	X	X
Cinsiyet	-	X	X
Lökositler	WBC	X	X
Kırmızı Kan Hücreleri	RBC	X	X
Hemoglobin	HGB	X	X
Hematokrit	HCT	X	X
Ortalama Eritrosit Hacmi	MCV	X	X
Ortalama Eritrosit hemoglobini	MCH	X	X
Ortalama Eritrosit Hemoglobin Konsantrasyonu	MCHC	X	X
Trombositler	PLT	X	X
Nötrofil Sayısı (%)	NE	X	X
Lenfosit Sayısı (%)	LY	X	X
Monosit Sayısı (%)	MO	X	X
Eozinofil Sayısı (%)	EO	X	X
Bazofil Sayısı (%)	BA	X	X
Nötrofil Sayısı	NET	X	X
Lenfosit Sayısı	LYT	X	X
Monosit Sayısı	MOT	X	X
Eozinofil Sayısı	EOT	X	X
Bazofil Sayısı	BAT	X	X
Glikoz	GLU	X	
Kreatinin	CREA	X	
Üre	UREA	X	
Alanin Aminotransferaz	ALT	X	
Aspartat Aminotransferaz	AST	X	
Alkalin Fosfataz	ALP	X	
Gama Glutamil Transferaz	GGT	X	
Laktat Dehidrogenaz	LDH	X	
Kreatin Kinaz	CK	X	
Kalsiyum	CA	X	
C-Reaktif Protein	CRP	X	
Kell Antigen Typing	KAT	X	
Nükleik Asid Amplifikasyon Test	NAAT	X	

Enfekte olmuş COVID-19 bireylerinin birçok kan parametre değerlerinin değiştiği gözlemlenmiştir (Abayomi-Alli ve ark., 2022). Örneğin, Liu ve ark.'nın (2020) gerçekleştirdiği çalışmada, Çinli COVID-19 hastalarının transaminaz ve LDH değerlerinin yükseldiği gözlemlenmiştir. Ferrari ve ark.'nın (2020) gerçekleştirdiği çalışmada, yine enfekte bireylerde yüksek LDH değeri tespit edilmiştir. Yapılan diğer istatistiksel çalışmalarda, enfekte bireylerin kanlarında lenfosit ve bazofil değerlerinin düşük olduğu, WBC, ALT, AST, CREA ve CRP değerlerinin yüksek olduğu analiz edilmiştir (Chen ve ark., 2020; Ferrari ve ark., 2020; Huang ve ark., 2020; Lippi & Plebani, 2020; Liu ve ark., 2020; Zhang ve ark., 2020). COVID-19 için belirleyici olan bazı kan parametrelerinin bilgileri Tablo 3.2'de verilmiştir.

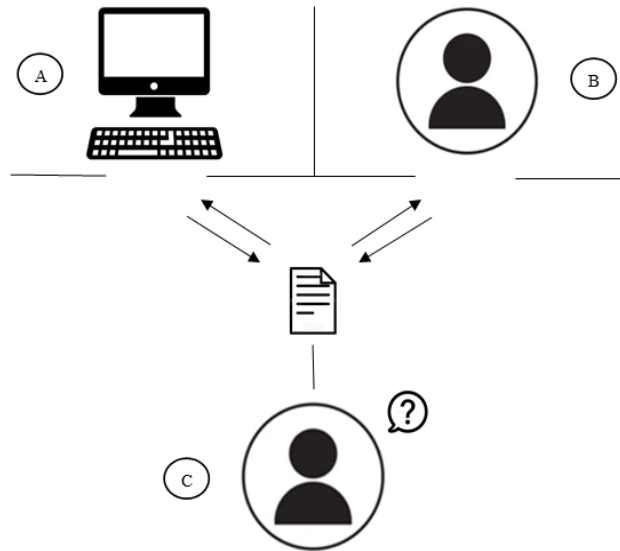
Tablo 3.2. COVID-19 İçin Belirleyici Olan Bazı Kan Parametrelerinin Görevleri

KAN PARAMETRESİ	AÇIKLAMA	REFERANS ARALIĞI
LDH	Vücutta bulunan şekerden enerji elde eden enzimdir. Genel olarak, kanda yüksek seviyeler çıkması, organ veya hücre hasarına işaretler.	~ 90-240 U/L
WBC	Kandaki beyaz kan hücrelerini temsil eder. Genel olarak, kanda WBC seviyesinin yüksek olması, bakteri ve virüs tabanlı enfeksiyonlara işaretler.	~ 4-10 mcL
ALT	Karaciğer enzimidir. Karaciğer hasarını kontrol eder. Genel olarak, kanda ALT yüksekliği, karaciğer hastalıklarına işaret eder.	~ 7-56 U/L
AST	Bir diğer karaciğer enzimidir. Yüksek AST, başta karaciğer olmak üzere kalp, beyin ve böbreklerde herhangi bir hastalık belirtisi olabilir.	~ 15-42 IU/L
CREA	Vücuttaki kas kütlelerini ve böbrek fonksiyonlarının durumunu belirtir. Genel olarak, kanda CREA yüksekliği, böbrek hastalıklarına işaret eder.	~ 0.5-1.4 mg/dL
CRP	Vücuttaki iltihabı ve iltihabın derecesini ölçen karaciğer proteindir. Sağlıklı bireylerde, CRP değeri çok düşüktür. Genel olarak, CRP değerinin yükselmesi vücutta enfeksiyon olduğuna, kalp krizi riskine veya doku ölümüne işaret eder.	~ 0-0.5 mg/dL

3.3. Yapay Zeka

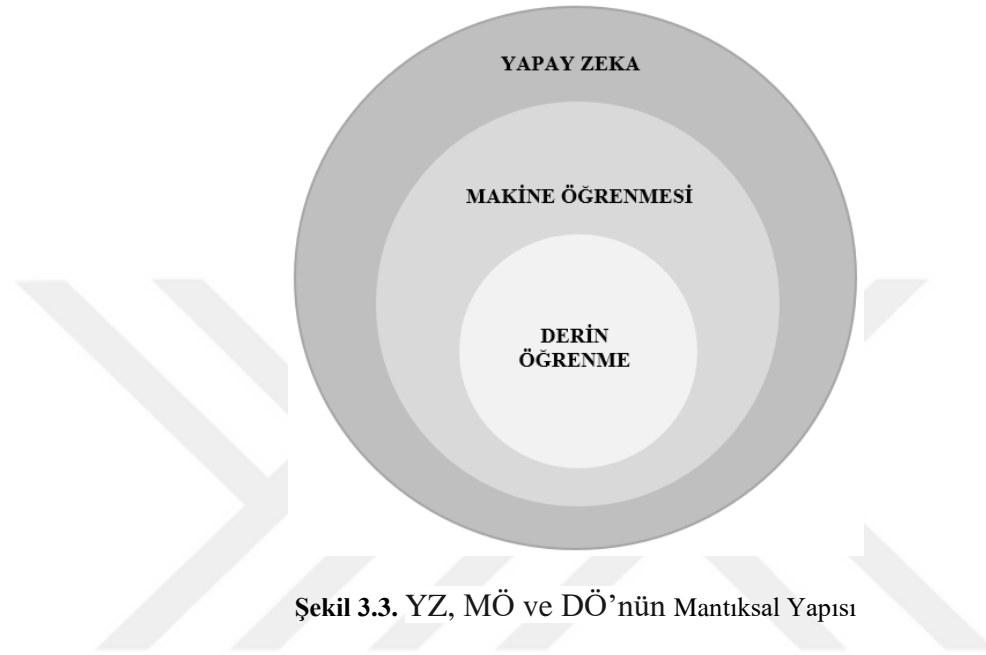
YZ, karmaşık görevleri aldığı yönergelerle insan zekasını taklit ederek çözümleyen ve hatalarını öğrenerek kendini sürekli geliştirebilen sistemlerdir. Yönergeler sunularak makinelerin belirli bilgileri öğrenmesini, öğrendiği bilgilerden deneyimler elde ederek sonraki yeni verilere uyum sağlamasını ve insan benzeri görevleri gerçekleştirmesini mümkün kılan bir teknolojidir. 1940’larda yapımına katkı verdiği elektronik bilgisayarların, potansiyelini gören Alan Turing, 1950 yılında “Makineler düşünebilir mi ?” sorusundan yola çıkarak yayınladığı makalede geliştirdiği Turing Testi ile bugünkü YZ’nin temelini atmıştır (Turing, 1950). Bu test, YZ bilimine büyük bir ivme kazandırmış ve bugünkü geldiğimiz noktaya ulaştırmıştır.

Turing Testi, bir YZ’nin insana göre gösterdiği akıl davranışını test eder ve temel amacı bir makinenin düşünebildiğini söyleyebilmenin mantıksal olarak mümkün olup olmadığıdır. Şekil 3.2’de görüldüğü üzere, Turing Testinde bir makine (A) bir insanla (B) birlikte sorgulayıcıdan (C) uzakta bir yere saklanır. Sorgulayıcıya iki taraf hakkında hiçbir bilgi verilmez ve yalnızca soru sorarak hangisinin insan hangisinin bilgisayar olduğunu tespit etmeye çalışır. Sorgulayıcının sorduğu sorular ve aldığı yanıtlar, yalnızca klavye sisteminden yazılarak verilir. Yinelemeli olarak tekrarlanan testler sonucunda sorgulayıcı, tutarlı bir şekilde insanı makineden ayırt edemediği takdirde, makine yani YZ Turing Testini geçmiş sayılır (Aydın, 2017).



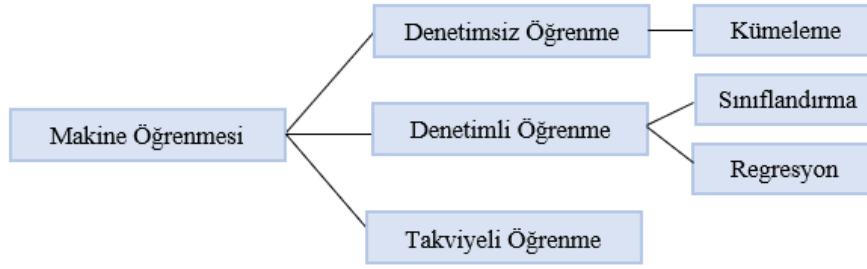
Şekil 3.2. Turing Testi

YZ'nin gnbegen geliřimi ile karmařık grevleri herhangi bir ynerge kullanmadan, sadece verilerden ve de deneyimlerinden grenerek gerekleřtiren, YZ'nin alt dalı olan M ortaya ıkmıřtır. Daha sonra, yapısındaki derin aęlar sayesinde zellik ıkarımını aę ierisinde kendi yapan ve M yntemi olan D geliřtirilmiřtir. YZ, M ve D'nn mantıksal yapısı Őekil 3.3'te sunulmuřtur.



3.3.1. Makine grenmesi

İnsanların grenme Őekillerini taklit ederek ynergelere ihtiya duymadan, verilerin algoritmalar ile entegrasyonuna odaklanan, doęruluęunu kademeli olarak arttıran YZ yntemidir. M, insanın grdę kavramları kendi kendine grenmesini ve bu greniminden sonra da tekrar bir grenime ihtiya duymadan mrnn sonuna kadar bunu kullanabilme yeteneęini temel alır. M, kendisine sunulan veri kmelerini zmseyerek yapılması istenen sınıflandırma grevini ynergeler olmadan direkt yapmayı amalar. M'y, YZ'den soyutlayarak bir alıřma alanı yapan temel prensip, ynergelere ihtiya duymadan veriler zerinden istatistiksel iřlemler ile kendi sınıflandırma karar mekanizmasını oluřturabilmesidir. M: Denetimsiz grenme, Denetimli grenme ve Takviyeli grenme olarak 3 ana kategoriye ayrılır ve Őekil 3.4'te Őematik yapısı sunulmuřtur.



Şekil 3.4. Makine Öğrenmesi Kategorileri

Denetimsiz Öğrenme: Veri setindeki örneklerin, sınıf bilgileri etiketlenmediğinde kullanılır. Sistemi eğitmek için etiketlenmemiş karışık verilerden, verileri tek tek araştırır ve verilerin yapısını tanımlamak için veri kümelerinden çıkarımlar yaparak verileri gruplar. Bu gruplamayı, etiketlenmemiş veriler arasındaki benzerliği araştırarak, aralarında bir ilişki kurarak yapar. Kümeleme problemlerinde başarılıdır (Goodfellow ve ark., 2016).

Denetimli Öğrenme: Geçmişte öğrenilen bilgilerle gelecekteki olayları tahmin etmeyi hedefleyen MÖ yöntemidir. Sınıf bilgisi etiketli örneklerden oluşan veri setinin analizini yapar. Öğrenme algoritması, çıktı değerleri hakkında tahmin yapmak için veriler ve bu verilerin sahip olduğu etiketler arasındaki ilişki ile bir çıkarım sistemi oluşturur. Bu çıkarım sistemi, minimum hata ile gelecekte herhangi bir girdiyi doğru çıktıya eşleyecek eğitimler yapar ve sistem yeterli eğitimden sonra yeni girdiler için doğru tahminler yapar hale gelir. Denetimli öğrenmenin, sınıflandırma ve regresyon problemleri için başarılı ve hızlı olduğu kanıtlanmıştır (Rende ve ark., 2016).

Takviyeli Öğrenme: Hedefe ulaşmak için doğru kararlar almayı nasıl öğreneceğini bir öğrenme sistemi ile kendi keşfeder. Öğrenme sistemi, bir karar verdiğinde bu kararın doğru olduğu durumlar için sistemi ödüllendirir, yanlış durumlarda cezalandırır. Amaç, öğrenen sistemin denediği olası durumların hedef olup olmadığının kontrolü ve sistemin kararlı hale gelmesidir.

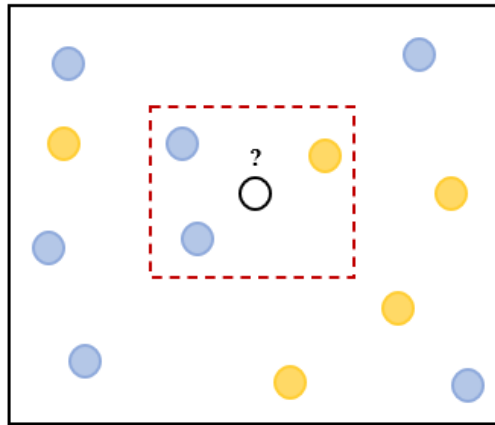
3.3.1.1. Sınıflandırma Algoritmaları

Sınıflandırma, sınıf etiketleri belli olan bir gözlem setini kullanarak yeni gözlemlerin sınıf etiketini bulma işlemidir. Sınıflandırma algoritmaları ise sınıf etiketi belli olan eğitim verilerini en iyi derecede eğiterek sınıf etiketi belli olmayan test

verilerini en doğru şekilde sınıflandırmaya çalışır. Bu algoritmaların temel amacı, veri setinde gizlenmiş gizli örüntüyü bulmaktır (Vellido ve ark., 2012). Geliştirilen birçok sınıflandırma algoritması vardır ve bu algoritmalar, ele alınan problem için sınıflandırma modelini oluşturmaktadır. Geliştirilecek bir sınıflandırma modeli için hangi sınıflandırma algoritması daha doğru tahminlerde bulunuyorsa o sınıflandırma algoritması ile geliştirilen model, sınıflandırma modeli olarak seçilir ve çeşitli performans metrikleri ile sınıflandırıcı modelin başarısı değerlendirilir. Literatürde sınıflandırma işlemi için çok fazla algoritma bulunmaktadır. Bazıları şunlardır; KNN, Naive Bayes, DVM, KA ve KÖ yöntemler olan Bagging, Adaboost ve RO.

3.3.1.1.1. K-En Yakın Komşu Algoritması

KNN, mesafeye dayalı bir sınıflandırma algoritmasıdır. Temel mantığı, sınıflandırılmak istenen yeni gözlemin sınıf etiketi, en yakınındaki k adet gözlemin sınıf etiketine bağlıdır (Ayık ve ark., 2010). Yakınlık mesafe ölçüsü olarak en sık kullanılan mesafe ölçütleri, Öklid ve Manhattan ölçütleridir. Buradaki k parametresi, en yakın gözlem sayısıdır ve bu yöntem için en önemli durum k değerinin belirlenmesidir. Şekil 3.5'te görüldüğü üzere, k=3 seçilerek sınıflandırılmak istenen yeni gözleme ait 3 yakın komşu bulunmuş ve yeni gözlemin sınıf etiketi, bu 3 gözlemin sınıf etine göre belirlenmiştir.



Şekil 3.5. KNN Algoritması Çalışma Prensipleri

3.3.1.1.2. Naive Bayes Algoritması

Naive Bayes, Bayes teoremine bağlı olasılık tabanlı sınıflandırma algoritmasıdır. Bu algoritma, eğitilmiş veriler üzerinde belirli olasılık işlemleri yapar ve test verileri, elde edilmiş bu olasılık değerlerine göre işletilerek hangi sınıf etiketine sahip olduğu

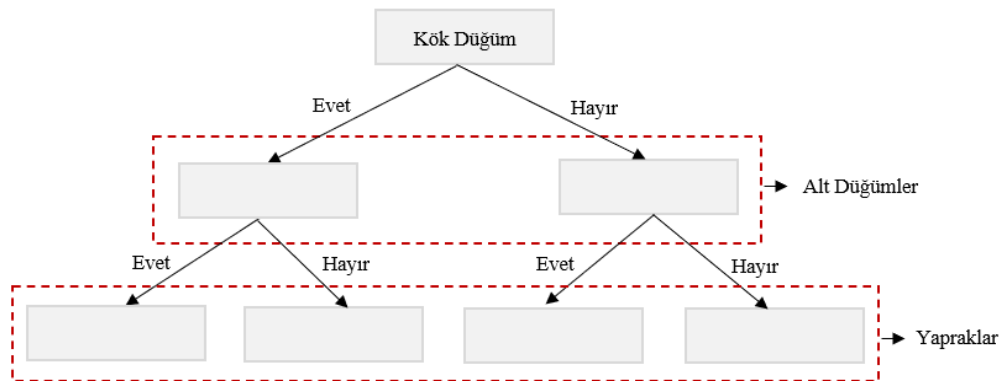
tespit edilmeye çalışılır. Bu yöntem, sınıflandırılmış veriler kullanılarak yeni gelen verinin mevcut sınıf etiketlerinden birisi olma ihtimalini hesaplayan bir yöntemdir ve bir sınıftaki belirli özelliklerin varlığının diğer herhangi bir özellik ile ilgisi olmadığı varsayılır (Cover, 1965).

3.3.1.1.3. Destek Vektör Makinesi

DVM, V.N. Vapnik (1995) tarafından geliştirilen istatistiksel öğrenme teorisine dayalı bir sınıflandırma algoritmasıdır. Son yıllarda aktif olarak kullanılan DVM, sınıflandırma işlemi yaparken yüksek başarı sağlamak için büyük boyutlu çekirdek fonksiyonları kullanmaktadır (Melgani & Bruzzone, 2004). DVM temel olarak, bir düzlemde sınıfları birbirinden ayıracak bir sınır çizerek sınıflara ait hiper-düzlem oluşturur.

3.3.1.1.4. Karar Ağacı

KA, düğüm, dal ve yapraklardan oluşan ağaç tabanlı bir sınıflandırma prensibidir. Bu prensip, düğümlerde veri setinin özelliklerini, dallarda gözlemlerin olasılık durumlarını ve yapraklarda sınıf etiketlerini tutar. Şekil 3.6'da bir KA'nın temel yapısı şematik olarak sunulmuştur. Kök düğüm, veri setini en çok açıklayan özellik olarak seçilir ve bu seçimde gini index veya entropy ölçütleri kullanılır. Kalan alt düğümlerin seçimi ise yine bu ölçütler kullanılarak veri setini en çok açıklayan özelliklerin belirlenmesi ile gerçekleştirilir. KA'nın yaprakları ise sınıf etiketlerini tutarak sonuca ulaştırır. KA, veriyi özyinelemeli olarak alt gruplara dallanma yaparak böler ve ayırma aşamasında oluşan dalların her biri, bir kuralı temsil eder (Bozkır ve ark., 2009). Literatürde birçok KA algoritması vardır. Bunlardan en popülerleri ID3 algoritmasıdır.



Şekil 3.6. Karar Ağacı Temel Yapısı

3.3.1.1.4.1. ID3 Algoritması

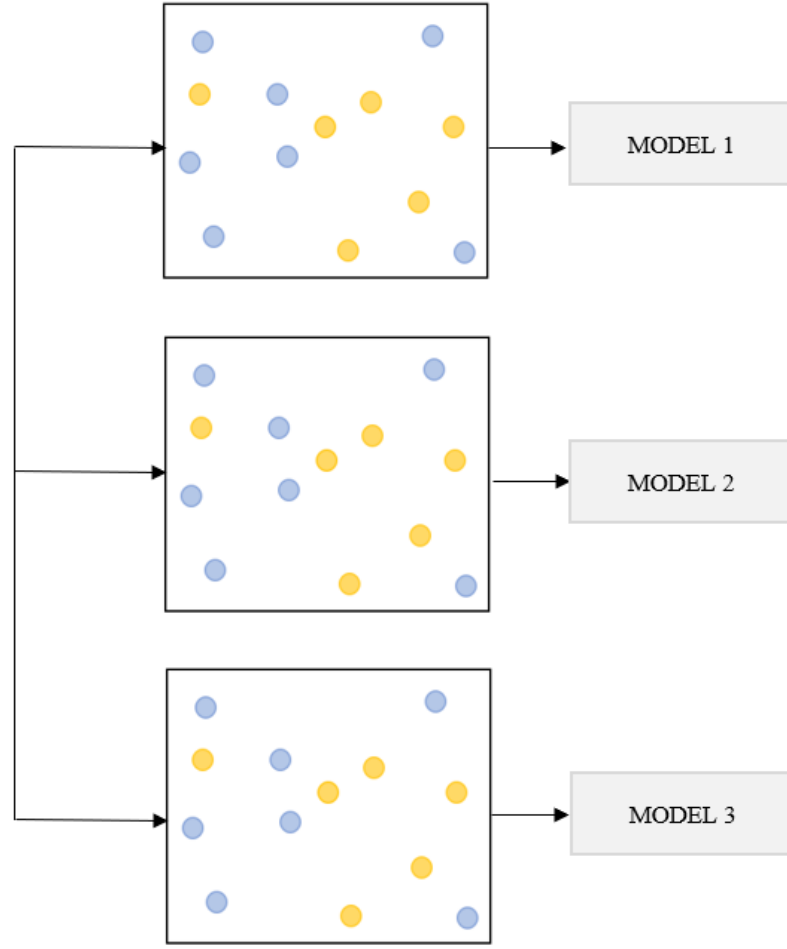
ID3 algoritması, J.R. Quinlan (1986) tarafından geliştirilen KA oluşturma algoritmasıdır. Bu algoritma, tek tek özellik vektörleri inceler ve en yüksek bilgi kazancına sahip özellik ağaçta dallanma yapmak için tercih edilir. Bu algoritma, entropy ve bilgi kazancı üzerine inşa edilmiştir (Hssina ve ark., 2014) ve sadece kategorik veriler ile çalışmaktadır. ID3 algoritması aynı zamanda, bir diğer KA oluşturma algoritması olan C4.5 algoritmasının da öncüsüdür.

3.3.1.1.5. Kolektif Öğrenme

KÖ, tek bir sınıflandırıcı modelden elde edilebilecek performanstan daha iyi sonuçlar elde etmek için birden çok modelleme algoritmasının bir araya getirilmesiyle oluşturulmuş bir sınıflandırma tekniğidir (Erol ve ark., 2022). KÖ yöntemleri, torbalama (bagging) ile varyansı azaltmak, artırma (boosting) ile önyargıyı azaltmak ve yığılma (stacking) ile daha başarılı tahminler elde etmek için kullanıldığından birçok sınıflandırıcı teknikten daha başarılı sonuçlar elde etmektedir (Wiysobunri ve ark., 2020). Literatürde birçok KÖ tabanlı sınıflandırma algoritmaları vardır: Bagging, Adaboost ve RO algoritmaları bunlardan en popülerleridir.

3.3.1.1.5.1. Bagging Algoritması

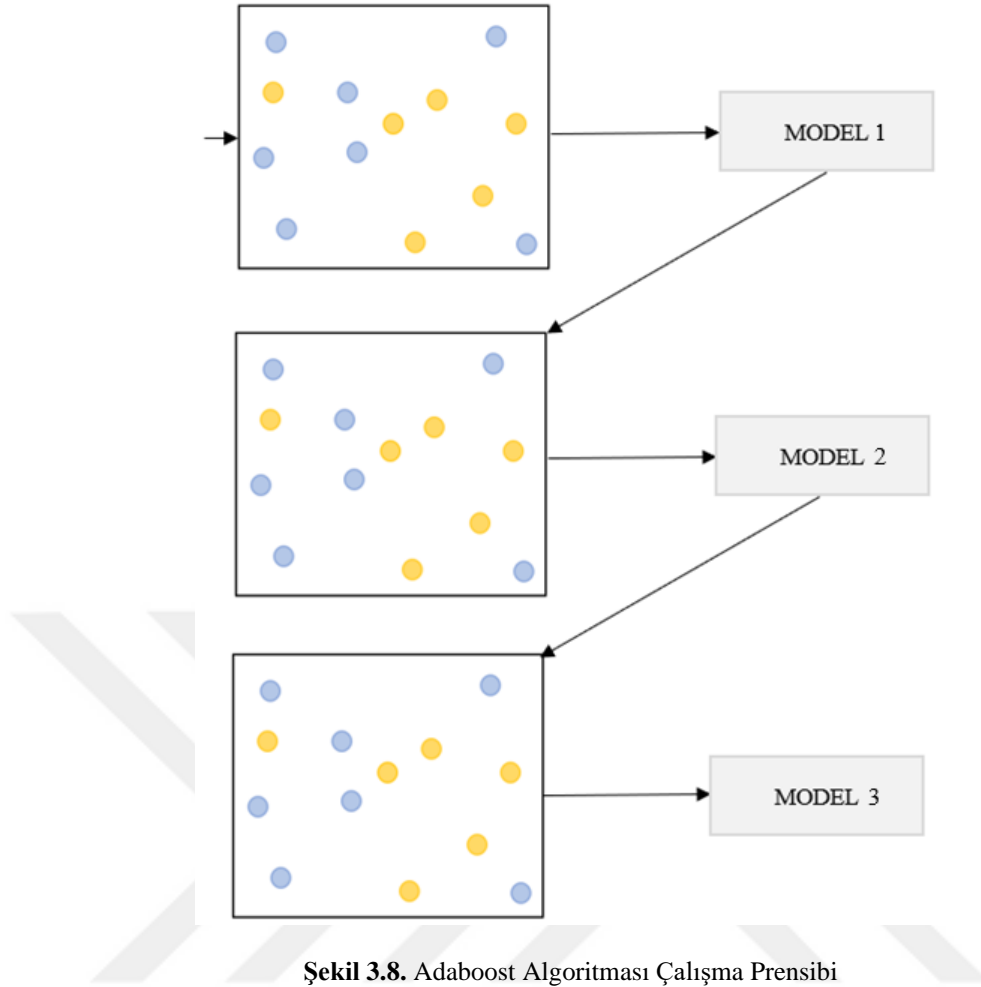
Bagging, torbalama yöntemi ile çalışan bir sınıflandırma algoritmasıdır. L. Breiman (1996) tarafından geliştirilen bu algoritma, veri setinden rastgele alt örneklemeler seçer ve bu örneklemelerin her birini bir sınıflandırıcı modeline sunarak sınıflandırıcı topluluğu oluşturur. Bu sınıflandırıcı topluluğunun her biri bir tahmin üretir. Üretilen bütün tahminlerin birleştirilmesi ile nihai sınıf tahmini üretilmiş olur. Tahminlerin birleştirilmesinde, regresyon ağaçları için ortalama alınırken sınıflandırma ağaçlarında sonuçlar çoklu oylama ile belirlenir. Bagging, n adet örnekten oluşan veri setinden yine n örnekle rastgele alt örneklemeler oluşturur. Seçilen her örnek tekrar geri veri setine konulur. Bagging'in çalışma prensibi Şekil 3.7'de sunulmuştur.



Şekil 3.7. Bagging Algoritması Çalışma Prensibi

3.3.1.1.5.2. Adaboost Algoritması

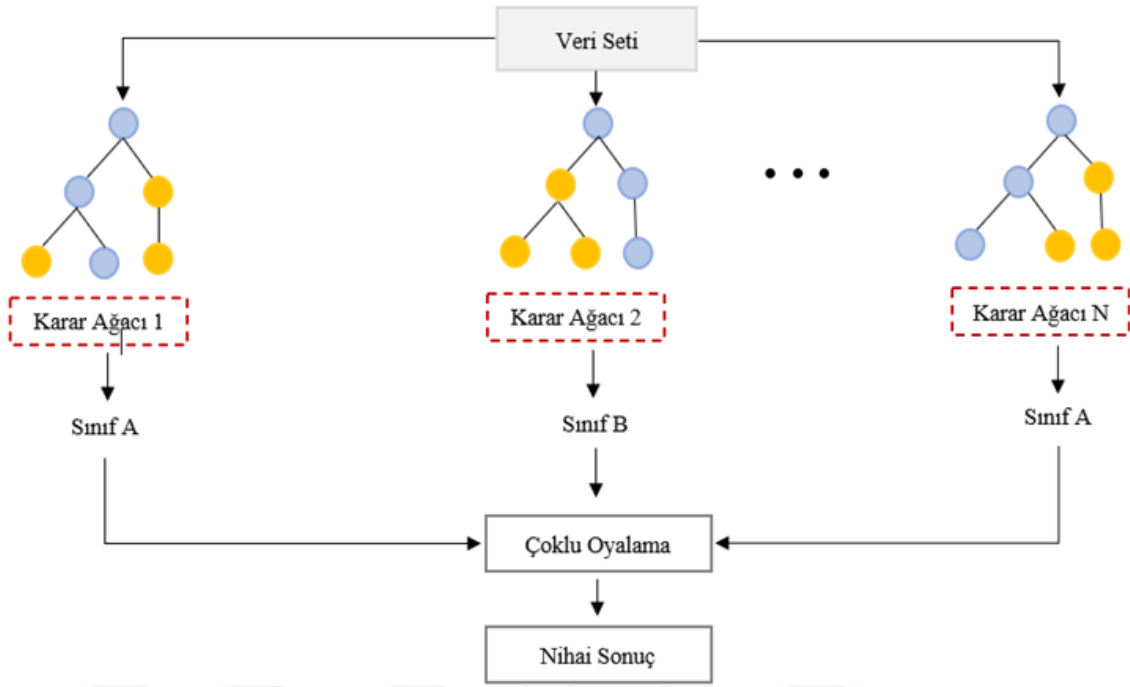
Y. Freund ve R. Schapire (1999) tarafından geliştirilen Adaboost, artırma yöntemi ile çalışan bir sınıflandırma algoritmasıdır. Artırma, birçok zayıf sınıflandırıcı modelini bir araya getirerek güçlü bir sınıflandırıcı modeli oluşturmaktır. Artırma yöntemlerinden biri olan Adaboost algoritmasında, eğitim seti öncelikle zayıf bir sınıflandırıcı ile eğitilir, sonrasında bu eğitim sonucunda yanlış olarak tahminlenen örneklere daha fazla öncelik verilerek, yani ağırlıkları artırılarak tekrar bir eğitim modeli oluşturulur. Zayıf sınıflandırıcı çıkışı diğer sınıflandırıcının girişi olacak şekilde model eğitilerek devam edilir ve en sonunda sonuçlar birleştirilerek nihai sınıflandırıcı modeli ile sınıf tahmini üretilir. Adaboost'un çalışma prensibi Şekil 3.8'de sunulmuştur.



Şekil 3.8. Adaboost Algoritması Çalışma Prensibi

3.3.1.1.5.3. Rastgele Orman Algoritması

RO, KA'ları kullanarak sınıflandırma yapan KÖ yöntemidir. RO, çok hızlı ve aşırı öğrenmeye karşıda dayanıklı bir algoritmadır (Breiman, 2001). RO, birden fazla KA oluşturarak bir orman oluşturur ve bu ormandaki her bir KA'yı farklı bir gözlem üzerinde eğiterek çeşitli modeller üretir, en son nihai bir tahmin elde etmek için bu modelleri birleştirerek sınıflandırma başarısını en yükseğe çıkarır. Şekil 3.9'da, RO algoritmasının çalışma prensibi sunulmuştur. Şekil 3.9'da görüldüğü üzere, algoritma veri setindeki her bir örnek için KA oluşturur ve her bir KA tahmini sonuç üretir. Tahmini sonuçlar içinden çoklu oylama yapılarak en çok oylanan tahmin bulunur ve sonuçta sınıflandırıcının nihai sınıf tahmini üretilmiş olur.

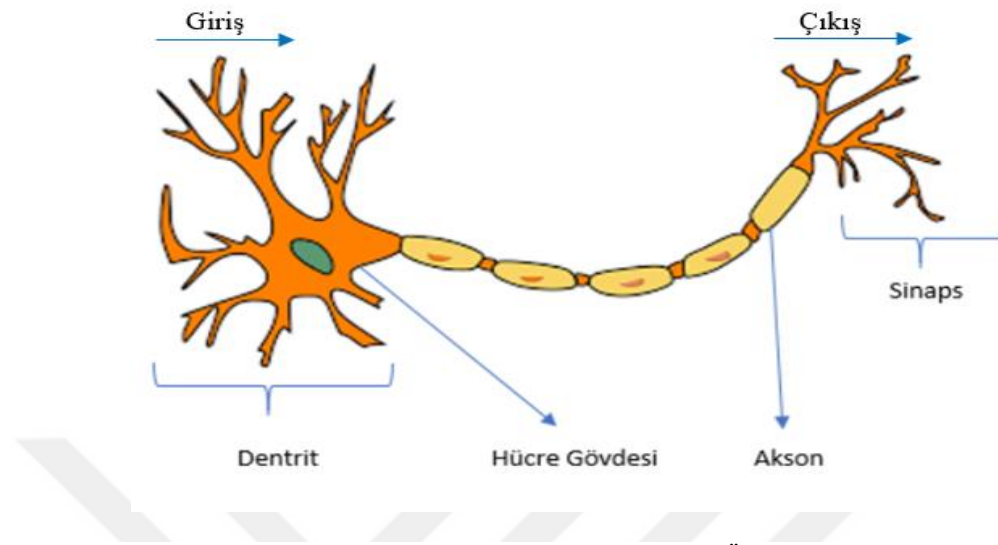


Şekil 3.9. Rasgele Orman Algoritması Çalışma Prensibi

3.3.2. Yapay Sinir Ağları

Eski bilgisayar sistemleri, hem aritmetik işlemleri yapmak hem de grafik çizmek gibi birçok işlevi yapmak yerine komutların dışarıdan verildiği sadece tek bir işlem yapan özel makineler olarak tasarlanmıştır. Bu tarz eski bilgisayarların tasarlandıkları amacın dışında başka işleri de yapmaları için bütün parçalarının tekrar baştan tasarlanması gerekiyordu. Bu da bilgi teknolojisi mantığına aykırı olarak hızdan, zamandan ve maliyetten oldukça düşük performans getirmekteydi. Ünlü matematikçi ve bilgisayar bilimcisi J. Von Neumann (1945) kendi adının verildiği Von Neumann bilgisayar mimarisini geliştirerek bilgisayar işlemcisini hafızadan ayırmış ve komutları da veriler gibi bilgisayarın içinde tutarak işlemlerin çok daha hızlı yapılmasını sağlamıştır. Dolayısıyla bilgisayara farklı bir işi yaptırmak için artık yeni komutlarla yeniden tasarlanmasına ihtiyaç kalmamış, sadece komut kısmı değiştirilerek başka işlemlerin yapılması sağlanmıştır. Fakat günümüz bilgisayarlarında da halen kullanılan Von Neumann bilgisayar mimarisi, sayısal ve sembol işlemede çok başarılı olurken algısal işlemlerde oldukça yetersiz kalmaktadır. İnsan beyni ise sayısal ve sembol işlemede daha yavaş olmasına rağmen öğrenme süreci sayesinde karmaşık algısal işlemlerde çok hızlı ve başarılıdır. Bu öğrenme süreci ise insanın sahip olduğu sinir sistemindeki sinir hücreleri (nöronlar) arasındaki bilgi aktarımı sayesinde

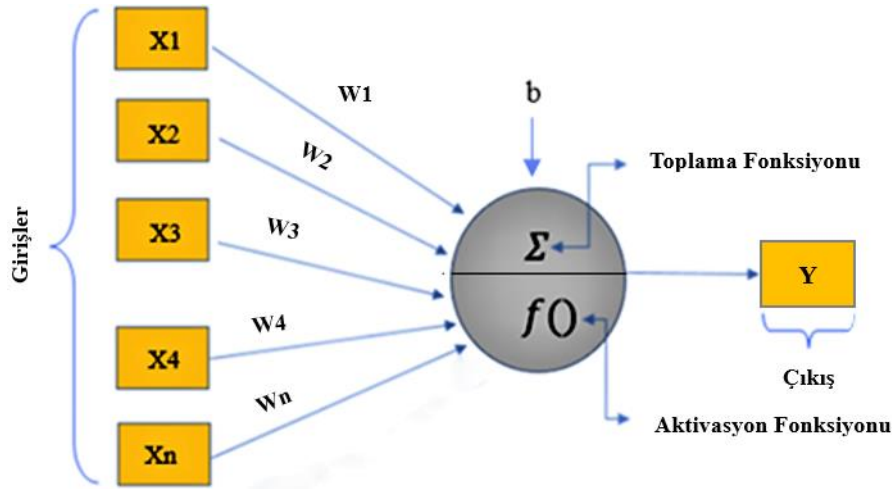
gerçekleşmektedir. Şekil 3.10’da, insanın bir şeyleri öğrenmesini sağlayan, biyolojik sinir hücresi yapısı şematik olarak gösterilmiştir.



Şekil 3.10. Biyolojik Sinir Hücresi Yapısı (Öztemel, 2012)

İnsanda bulunan sinir sistemi, nöron adı verilen sinir hücrelerinden oluşur. Nöronlar arasında bilgi aktarımı dentrit ve akson adlı iki uçtan sağlamaktadır. Dışarıdan bir uyarı geldiğinde sinir hücrelerinde “implus” adında bir sinyal oluşur, dentritler sayesinde bu sinyal algılanır ve akson ile diğer nörona iletilir. Aksonları diğer nörona bağlayan birim, sinaps olarak adlandırılır ve bu yapı öğrenmeyi sağlayan temel birimdir. İnsanların olayları tecrübe ederek deneyimlemesi sonucunda, bu sinaptik bağlantılar sürekli güncellenerek yeni bağlantılar oluşur ve bu sayede öğrenme işlemi gerçekleşir (Mishra & Srivastava, 2014).

YSA, insan beyninin biyolojik öğrenme yapısını örnek alarak bu öğrenme sürecinin matematiksel olarak modellenmiş bir YZ yöntemidir. Öğrenme işlemi örnekler yardımı ile gerçekleştiren YSA, birbirine bağlı yapay sinir hücrelerinden oluşmaktadır. Hücreler arasındaki her bir bağlantı, belirli ağırlık değerleri sahiptir ve bu da YSA’nın sahip olduğu bilginin bütün hücelere yayılmasını sağlayarak öğrenme işlemi gerçekleştirmektedir. Şekil 3.11’de, bir YSA’nın giriş değerlerinin (X_1, X_2, \dots, X_n) belirli ağırlıklar (W_1, W_2, \dots, W_n) aracılığı ile nöronlara bağlı olduğu temel bir YSA yapısı sunulmuştur. Şekil 3.11’de görüldüğü üzere, bir YSA, bilgi girişi, her bir girişin ağırlığı, toplama fonksiyonu, aktivasyon fonksiyonu ve bilgi çıkışı olmak üzere 5 ana bileşenden oluşur (Tolon & Tosunoğlu, 2008). Bunlara ek olarak, bias sabit değeri de olabilir. Bias, bir modelin tahmin edilen değer ile gerçek değer arasındaki farktır.



Şekil 3.11. Yapay Sinir Hücresi Yapısı

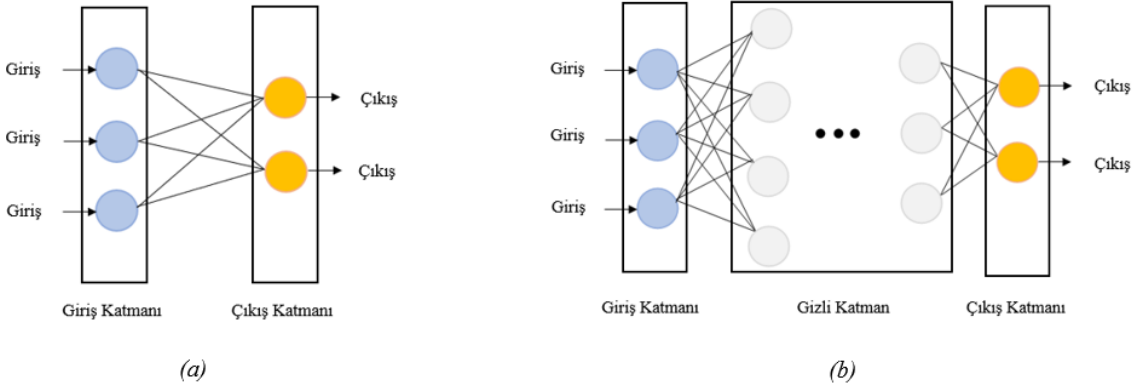
Biyolojik sinir hücresindeki yapıların, yapay sinir hücresinde karşılıkları şu şekildedir:

Dentrit	→	Giriş
Akson	→	Çıkış
Sinaps	→	Ağırlıklar
Hücre Gövdesi	→	Toplama ve Aktivasyon Fonksiyonu

YSA, birçok yapay sinir hücresinin birbirine bağlanmasıyla oluşan yapıdır. Bir YSA, temelde 3 katmandan oluşur. Bunlar; Giriş Katmanı, Gizli Katman ve Çıkış Katmanıdır. Giriş katmanı, öğrenilmesi istenen örneğin özniteliklerinin giriş verisi olarak verildiği ilk katmandır. Giriş katmanında, öğretilecek örneklerin öznitelik sayısı kadar giriş nöronu bulunmalıdır. Gizli katman, ağa gelen verilerin ağırlık değerleri kullanılarak bir çıktıya dönüşme işlemini yapan ara katmandır. Gizli katmanda gerçekleştirilen bu işlem, giriş verilerinin bağlı oldukları her nöronun ağırlık değerleri ile çarpılıp toplama fonksiyonu ile toplanarak sonrasında bir aktivasyon fonksiyonu aracılığıyla bir çıktı üretmesidir. Katman sayısı ve üzerindeki nöron sayısı problemde değişebilmektedir. Gizli katmanda elde edilen çıktı, çıkış katmanına sunulurken model tahmini bir net çıktı üretir. Yani çıkış katmanı, yapay ağda öğrenilmesi istenen örneklerin sınıf bilgisinin, çıkış olarak hesaplandığı katmandır. Ağın girdiler için doğru çıktıları üretebilmesi yani ağı öğrenimi, ağırlıkların doğru değerlere sahip olması ile gerçekleşmektedir. Bir YSA'da giriş katmanından çıkış katmanına kadar

gerçekleştirilen bütün bu ileri yönlü hesaplama akışına İleri Yayılım Algoritması (İYA) adı verilmektedir.

Tek katmanlı YSA, yalnızca giriş ve çıkış katmanlarından oluşan basit bir YSA olup doğrusal olmayan problemler için genellikle tercih edilmemektedir. Çok katmanlı YSA ise giriş katmanı, birden fazla gizli katman ve çıkış katmanından oluşmaktadır. Çok katmanlı YSA, karmaşık problemlerin çözümünde kullanılmaktadır ve dolayısıyla doğrusal olmayan problemlerde tercih edilmektedir. Gizli katman, verilerin problemin yapısına uygun fonksiyonlar ile işlenip çıkış katmanına aktarılmasını sağlar ve sayısı problemin türüne göre artırılabilir veya azaltılabilir. Tek katmanlı YSA ve çok katmanlı YSA'nın temel yapısı Şekil 3.12'de gösterilmiştir.



Şekil 3.12. (a) Tek Katmanlı YSA (b) Çok Katmanlı YSA

3.3.2.1. Toplama Fonksiyonu

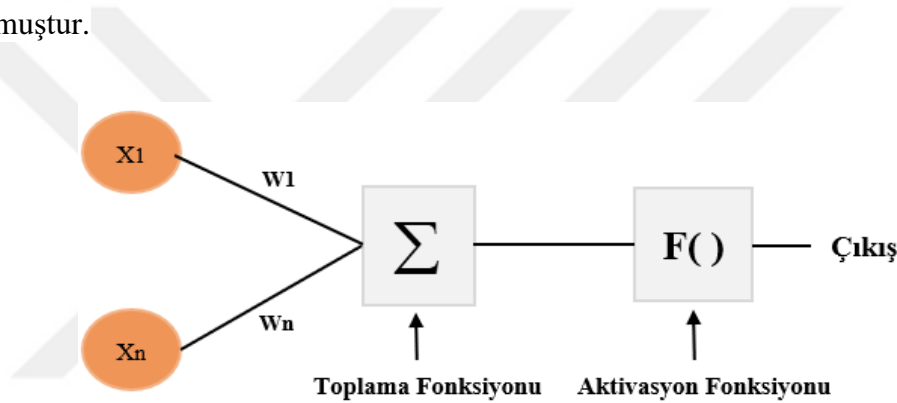
Ağırlıklar, giriş değerlerinin nöronlar üzerindeki etkisini ifade eden değerlerdir ve öğrenme işlemi, ağırlıkların en iyi değerinin bulunması ile gerçekleşir. Bundan dolayı, tüm ağırdaki ağırlıklar optimal bir değer almalıdır. Bu amaç doğrultusunda çalışan toplam fonksiyonu, ağırd net girişini yani öğrenmeye net etkisini hesaplar. Toplama fonksiyonu, her bir giriş değerinin (X_1, X_2, \dots), onu nörona bağlayan ağırlık değeriyle (W_1, W_2, \dots) çarpılıp toplanmaları ile bulunur. Toplama fonksiyonunun formülasyonu Denklem 3.1'de verilmiştir.

$$net = \sum_{i=1}^n W_i X_i + b \quad (3.1)$$

Toplama fonksiyonu ile elde edilen değer, sistemde kullanılan aktivasyon fonksiyonundan geçirilerek etkin çıkış değeri elde edilir. Aktivasyon fonksiyonunun temel görevi, toplama fonksiyonunun elde ettiği değerini bir analizini yaparak giriş sinyallerini sisteme uygun çıkış sinyallerine çevirmektir.

3.3.2.2. Aktivasyon Fonksiyonu

Bir YSA'da nöron, n sayıda girdi alan ve tek bir çıktı üreten temel matematiksel hesaplama birimidir. Bir yapay nöron, toplama fonksiyonu aracılığı ile X girdilerini W ağırlıkları ile çarpıp toplamını hesaplar ve bias değeri ekleyerek bir çıktı üretir. Aktivasyon fonksiyonu, toplama fonksiyonundan gelen çıktı değerini uygun bir şekilde işleyerek çıkışa ulaştırır. Şekil 3.13'te bir yapay sinir hücresinin temel yapısı sunulmuştur.



Şekil 3.13. Yapay Sinir Hücresi

Şekil 3.13'te görüldüğü üzere, bir yapay sinir hücresinde aktivasyon fonksiyonu bir nöron yapısının en son adımında yer almaktadır. Aktivasyon fonksiyonu, toplama fonksiyonundan gelen çıktı değerini kontrol eder ve kullanılan fonksiyon çeşidine göre ağırlıklı çıkış değerini üretir. Bir YSA'da kullanılan aktivasyon fonksiyonu, çıkıştaki anlamlı bilginin elde edilmesinde önemli bir rol oynar ve çıkışta iyi bir performans elde etmek için probleme uygun olan aktivasyon fonksiyonunun seçilmesi gerekmektedir. Veriye ve problemlere göre kullanılan aktivasyon fonksiyonu değişmektedir. Bu nedenle, nöral ağlarda çeşitli aktivasyon fonksiyonları kullanılmaktadır ve bu fonksiyonlardan en sık kullanılan üç tanesi şu şekildedir:

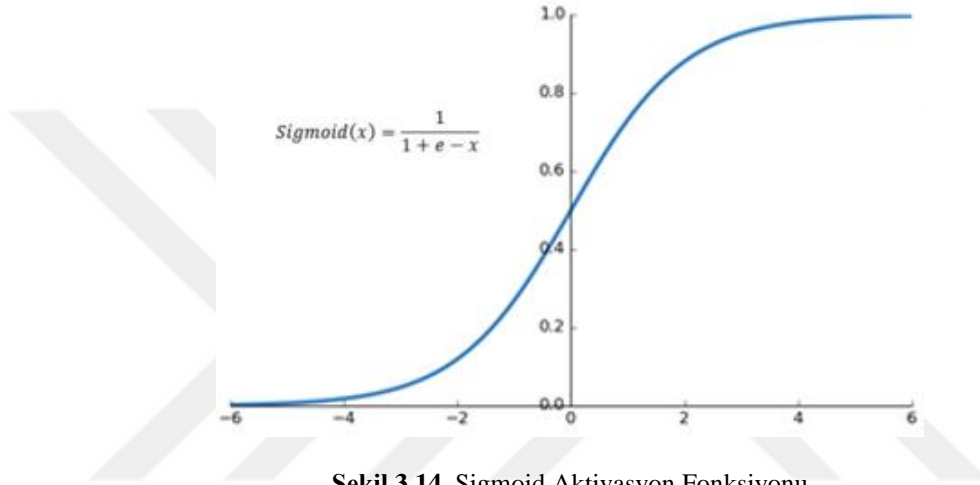
1) Sigmoid Aktivasyon Fonksiyonu:

En yaygın kullanılan, doğrusal olmayan aktivasyon fonksiyonlarından birisidir. Karar vermeye yönelik olasılıksal bir yaklaşım sunduğundan $[0,1]$ aralığında çıktı

üretmek için çıktının hangi sınıfa ait olduğuna dair olasılıksal bir değer verir. Denklem 3.2’de Sigmoid aktivasyon fonksiyonunun formülasyonu verilmiştir.

$$f(X) = \frac{1}{1 + e^{-x}} \quad (3.2)$$

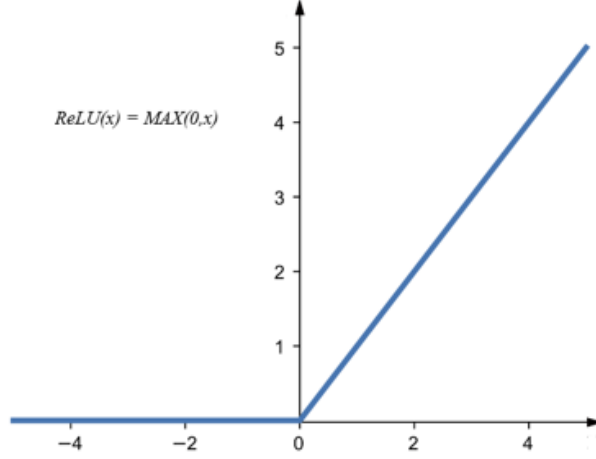
Sürekli türevlenebilir bir fonksiyon olduğundan öğrenme işlemi de gerçekleştirir. Şekil 3.14’te Sigmoid fonksiyonunun genel çizimi sunulmuştur.



Şekil 3.14. Sigmoid Aktivasyon Fonksiyonu

2) Rektifiye Lineer Birim Aktivasyon Fonksiyonu:

ESA’da çok sık kullanılan ve doğrusal olmayan Rektifiye Lineer Birim (ReLU) aktivasyon fonksiyonu, aynı anda tüm nöronları aktive etmez bundan dolayı $[0, +\infty]$ aralığında çıktı üretir. Bu da, bir nöron negatif değer üretirse aktive edilmeyeceği anlamına gelir. ReLU aktivasyon fonksiyonunun sıklıkla kullanılmasının en büyük sebebi, Şekil 3.15’te görüldüğü üzere 0’dan büyük olan girdilerin sabit türev değerine sahip olmasıdır. Böylece, tüm nöronları aktif etmeyip sadece pozitif değerli nöronları aktive ederek ağ daha hızlı çalışmaktadır. ReLU fonksiyon çıktısı olan X, sıfırdan büyük değerlerde ($X > 0$) çıktıyı X üretirken sıfırdan küçük veya eşit olduğunda ($X \leq 0$) çıktıyı 0 olarak üretmektedir.



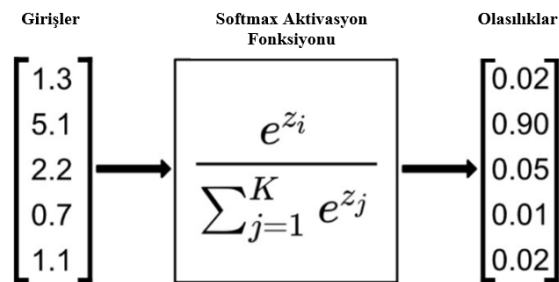
Şekil 3.15. ReLU Aktivasyon Fonksiyonu

3) Softmax Aktivasyon Fonksiyonu:

Çoklu Sigmoid olarak da bilinen Softmax, çok sınıflı sınıflandırma problemlerinde sıklıkla kullanılan bir aktivasyon fonksiyonudur (Hao, 2017). Softmax, sisteme girilen her girdinin bir sınıfa ait olma olasılığını gösteren çıktılar üretir ve bu çıktılar $[0,1]$ aralığındadır. Denklem 3.3'te Softmax aktivasyon fonksiyonunun formülasyonu verilmiştir.

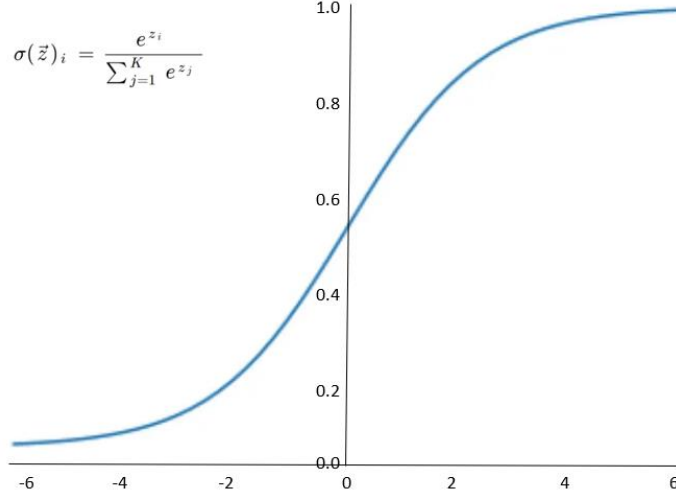
$$f(x) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}} \quad (3.3)$$

Sigmoid fonksiyonuna benzeyen bu aktivasyon fonksiyonunun Sigmoid'den farkı, Şekil 3.16'daki gibi giriş verilerinin toplamın 1 olacak şekilde bölme işlemi uygulayarak 0 ile 1 arasında normalize etmesidir. Bu normalize etme aşamasında, giriş değerlerinin önemi değişmeden $[0,1]$ aralığında olasılıklara dönüştürülmüş olur.



Şekil 3.16. Softmax Aktivasyon Fonksiyonunun Normalizasyonu

Sigmoid fonksiyonu, ikili sınıflandırma problemlerinde kullanılırken Softmax fonksiyonu ise çok sınıflandırıcı problemlerde kullanılır. Şekil 3.17’de Softmax fonksiyonunun genel çizimi sunulmuştur.



Şekil 3.17. Softmax Aktivasyon Fonksiyonu

3.3.2.3. Sınıflandırma

Canlılarda özellikle de insanlarda doğumdan itibaren öğrenilen, ilk nöroloji becerilerden biri olan sınıflandırma, bir verinin (görüntü, ses, metin vs.) girdi olarak alınmasının ardından o veri için onu en iyi tanımlayan sınıflar arasından doğru sınıf etiketi tespiti işlemidir. Bir YSA’da, giriş verileri alındıktan sonra her katmanında belirli işlemler uygulayarak eğitim süreci gerçekleştirilir. Bu eğitim sürecinin ardından, sınıflandırma yapılır ve giriş verilerine karşılık çıktı üretilir. İYA, YSA’nın giriş katmanından çıkış katmanına doğru toplama fonksiyonu ve aktivasyon fonksiyonu aracılığı ile gerçekleştirilen hesaplama işlemidir ve bu algoritma sonucunda elde edilen çıktılar, YSA’nın oluşturduğu tahmin değerleridir. Elde edilen tahmin değerler, çoğu zaman yüksek doğrulukta olmamaktadır ve tahmin değeri ile gerçek değer arasında bir fark değeri oluşmaktadır. Üretilen tahmin değeri ile gerçek değer arasında oluşan bu farkı karşılaştırma, hata fonksiyonu ile gerçekleştirilmektedir. Hata fonksiyonu, tahmin edilen çıktının gerçek değerden ne kadar uzak olduğunu hesaplar. Üretilen tahmini sonuç ile istenen sonuç arasındaki fark değeri, oluşturulan YSA modelinin “hata değeri” olarak nitelendirilir. Bu hatanın, eğitim esnasında zamanla sifira yaklaşarak modelin ürettiği tahmini çıktının, gerçek değeri üretmesi beklenir ve bunun için Geri Yayılım Algoritması (GYA) kullanılır. GYA, İYA sonucunda hesaplanan hata değerini sinir ağı

üzerinde çıktı katmanından girdi katmanına doğru çeşitli türev işlemlerine dâhil edilerek geriye doğru yayar. Bu sayede, hata geriye doğru bütün ağırlıklara aktarılarak iteratif bir şekilde ağırlıkların güncellemesi yapılır ve hata değeri azaltılmış olur. Azalan hata değeri YSA'nın doğruluğunu arttırarak gerçek değerler üretmesini sağlar.

3.3.2.3.1. Hata Fonksiyonu

YSA'nın ürettiği tahmini değer ile gerçek değer arasındaki fark, hata fonksiyonu ile hesaplanır. Hesaplanan hata değeri, sinir ağı üzerinde çıkış katmanından giriş katmanına doğru geriye doğru yayılarak ağırlıklarda güncellemeler yapılır. Güncellemeler sonucunda tekrar bir hata değeri hesaplanır ve hata değeri minimum değere gelene kadar bu işlem iteratif olarak tekrarlanır. Sinir ağları eğitilirken kullanılan bu yöntem Stokastik Gradyan İniş (SGİ) denir. SGİ, bir optimizasyon algoritmasıdır ve YSA'nın ürettiği tahminler, gerçek sonuçlardan çok fazla saparsa, hata fonksiyonu büyük sonuçları üretir. Optimizasyon fonksiyonlarının yardımıyla, hata fonksiyonu tahmindeki hatayı azaltmayı öğrenir. Birçok hata fonksiyonu vardır, bunlardan en çok kullanılanlar: Ortalama Kare Hata (OKH), Ortalama Mutlak Hata (OMH) ve Çapraz Entropi Kaybı (ÇEK). OKH ve OMH regresyon problemleri için sıklıkla kullanılırken ÇEK sınıflandırma problemleri için kullanılmaktadır.

1) Ortalama Kare Hata

OKH, YSA ile üretilen tahmini değerlerin gerçek değerlerden ne kadar farklı olduğunu, gerçek ve tahmini değerler arasındaki kare farkının ortalaması ile bulur. Bir regresyon probleminde, OKH değeri her zaman pozitif ve sifıra yakın ise modelinin performansının yüksek olduğunu söylenebilir. OKH'nin formülasyonu Denklem 3.4'te verilmiştir.

$$OKH = \frac{1}{n} \sum_{i=1}^n (Y_i - Y_i')^2 \quad (3.4)$$

OKH'nin yukarıda verilen formülasyonda: n hata sayısını, Y gerçek değeri ve Y' ağı ürettiği tahmin değerini nitelemektedir. Hesaplamalar sonucunda hata değeri sifirdan ne kadar uzaksa ağı başarısı o kadar düşük olacaktır bundan dolayı OKH değeri, ağı geri yayılarak küçültülür.

2) Ortalama Mutlak Hata

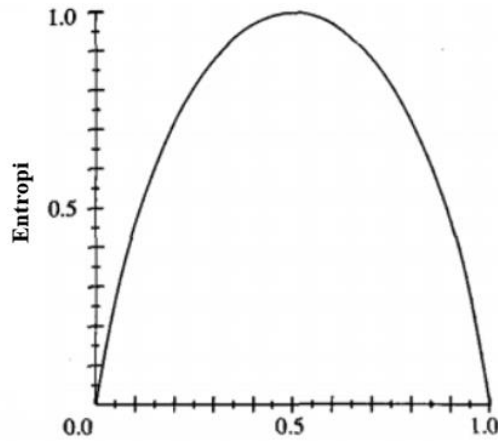
OMH, YSA ile üretilen tahmini değerlerin gerçek değerlerden ne kadar farklı olduğunu, gerçek ve tahmini değerler arasındaki mutlak farkın ortalaması ile bulur. OMH'nin formülasyonu Denklem 3.5'te verilmiştir.

$$OMH = \frac{1}{n} \sum_{i=1}^n |Y_i - Y_i'| \quad (3.5)$$

OMH'nin yukarıda verilen formülasyonda: n hata sayısını, Y gerçek değeri ve Y' ağıın ürettiği tahmin değerini nitelemektedir.

3) Çapraz Entropi Kaybı

İlk kez C. E. Shannon (1949) tarafından bir bilginin içeriğinin ölçülmesi olarak tanımlanan entropi, birden fazla sınıf bulunan bir sınıflandırma problemi için bir belirsizlik ölçüsüdür. Şekil 3.18'de görüldüğü üzere ikili sınıflandırma sistemi için entropi değeri 0 ile 1 arasında değişmektedir. Minimum değer olan 0, YSA'nın ürettiği tahminin kesinlikle gerçek değer ile aynı olmadığını; maksimum değer olan 1 ise YSA'nın ürettiği tahminin gerçek değer ile örtüşeceğini temsil etmektedir. Eğrinin tepe değerine karşılık gelen 0.5 değerinde ise belirsizlik maksimum olacaktır.



Şekil 3.18. İkili Sınıflandırma Problemi için Entropi Fonksiyonu (Mitchell, 1997)

Temel yapısı entropi mantığına dayanan ÇEK, çok sınıflı sınıflandırma problemleri için bir hata ölçüsüdür. ÇEK, 0 ile 1 arasında bir olasılık değeri üreterek sınıflandırma modelinin performansını ölçer. Tahmin edilen değer, gerçek değerden

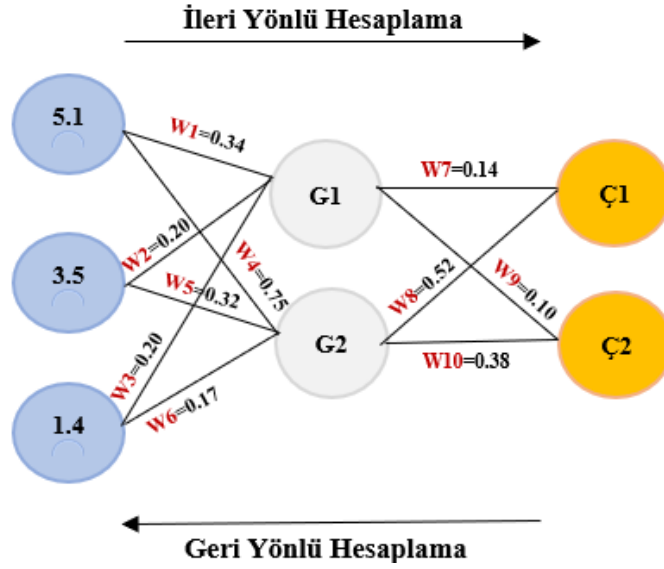
uzaklaştıkça kayıp değeri artar. Bu nedenle, ÇEK fonksiyonu 1 değerini ürettiğinde modelin gerçek değeri tahmin etmesi oldukça kötüdür. Hata fonksiyonu ÇEK olan mükemmel bir YSA modelinde, hata değeri 0 olacaktır. ÇEK'in formülasyonu Denklem 3.6'da verilmiştir.

$$\text{ÇEK} = -\sum_i^n P(Y_i) * \log P(Y'_i) \quad (3.6)$$

ÇEK'in yukarıda verilen formülasyonda: n sınıf sayısını, $P(Y)$ gerçek değerlerin olasılık dağılımını ve $P(Y')$ ağıın ürettiği tahmin değerlerinin olasılık dağılımını nitelemektedir.

3.3.2.3.2. Geri Yayılım Algoritması

GYA, İYA sonucunda hesaplanan hata değerinin, sinir ağı üzerinde çıktı katmanından girdi katmanına doğru geriye doğru yayılmasını sağlayan algoritmadır. Girişler ile çıkışlar arasındaki hata değeri bulunarak ağırlıklar bu hata değerine göre güncellenmektedir. GYA ile amaç, ağıdaki ağırlıkların her birini hata değeri aracılığı ile güncelleyerek tahmini çıktının, gerçek çıktıya daha yakın olmasını sağlamaktır. Ağırlıklar güncellenerek hata değeri minimum değere gelene kadar bu işlem iteratif olarak tekrarlanır ve bu şekilde sinir ağının başarısı yükseltilir. GYA'da her bir ağırlığın toplam hataya olan etkisinin hesaplanması için SĞİ optimizasyon algoritması kullanılır ve GYA'daki geri yönlü hesaplamaların gerçekleşebilmesi için öncelikle ileri yönlü hesaplamaların gerçekleşmesi gerekir. Şekil 3.19'da, 3 giriş ve 2 çıkışı olan 2 gizli katmanlı bir YSA verilmiştir ve bu YSA üzerinden ileri ve geri yönlü hesaplama prensibi anlatılmıştır.



Şekil 3.19. YSA'da İleri ve Geri Yönlü Hesaplama

İleri yönlü hesaplama, giriş nöron değerlerinin sahip oldukları ağırlık değerleri ile çarpılıp çarpımların toplama fonksiyonu aracılığı ile toplanması ve sonrasında, toplama fonksiyonundan elde edilen çıkışa aktivasyon fonksiyonu uygulanmasıdır. Şekil 3.19'da sunulan YSA'nın giriş katmanından gizli katmanına doğru olan ileri yönlü hesaplamaların şematik yapısı Şekil 3.20'de, Denklem 3.7'de sunulan toplama fonksiyonu ve Denklem 3.8'de sunulan Sigmoid aktivasyon fonksiyonu kullanılarak sunulmuştur.



Şekil 3.20. YSA'da Giriş Katmanından Gizli Katmana Doğru İleri Yönlü Hesaplama

$$Net = \sum_{i=1}^n X_i W_i \quad (3.7)$$

$$F(Net) = \frac{1}{1 + e^{-Net}} \quad (3.8)$$

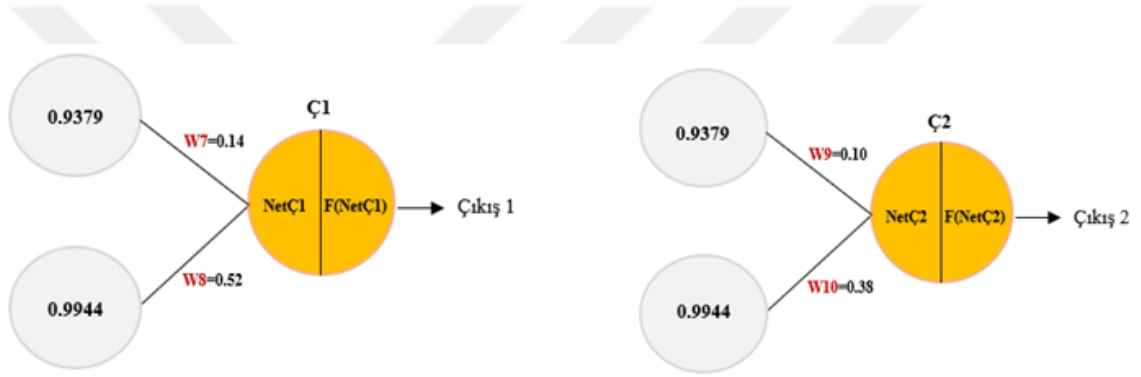
$$\text{NetG1} = 5.1*0.34 + 3.5*0.20 + 1.4*0.20 = 2.7140$$

$$F(\text{NetG1}) = 1 / (1+e^{-2.7140}) = 0.9379$$

$$\text{NetG2} = 5.1*0.75 + 3.5*0.32 + 1.4*0.17 = 5.1830$$

$$F(\text{NetG2}) = 1 / (1+e^{-5.1830}) = 0.9944$$

Giriş katmanında gizli katmana doğru uygulanan ileri yönlü hesaplama mantığı, aynı şekilde gizli katmandan çıkış katmanına doğru devam edilir. Şekil 3.19’da sunulan YSA’nın gizli katmandan çıkış katmanına doğru olan ileri yönlü hesaplamasının şematik yapısı Şekil 3.21’de, Denklem 3.7’de sunulan toplama fonksiyonu ve Denklem 3.8’de sunulan Sigmoid aktivasyon fonksiyonu kullanılarak sunulmuştur.



Şekil 3.21. YSA’da Gizli Katmandan Çıkış Katmanına Doğru İleri Yönlü Hesaplama

$$\text{NetÇ1} = 0.9379*0.14 + 0.9944*0.52 = 0.6483$$

$$F(\text{NetÇ1}) = 1 / (1+e^{-0.6483}) = 0.6566$$

$$\text{NetÇ2} = 0.9379*0.10 + 0.9944*0.38 = 0.4715$$

$$F(\text{NetÇ2}) = 1 / (1+e^{-0.4715}) = 0.6157$$

Yukarıdaki hesaplamada görüldüğü üzere, Şekil 3.19’da sunulan YSA’nın çıkış değerleri olan Çıkış 1 ve Çıkış 2 sırası ile 0.6566 ve 0.6157 olarak bulunmuştur. Bu değerler, YSA’nın ileri yönlü hesaplama ile ürettiği tahmini değerlerdir. Yani, YSA gerçek değeri 1 olan Çıkış 1 için 0.6566 ve gerçek değeri 0 olan Çıkış 2 için 0.6157 tahminlerini üretmiştir. Buradaki gerçek değer ile tahmini değer arasındaki fark, YSA’nın hata değerini vermektedir ve iki çıkışı bulunan bu YSA için iki hata değeri olacaktır. Şekil 3.19’da sunulan YSA’nın, Denklem 3.9 ile hata değeri ve Denklem 3.10 ile toplam hata değerinin hesaplanması aşağıdaki gibidir:

$$H_i = Gerçek_i - Tahmin_i \quad (3.9)$$

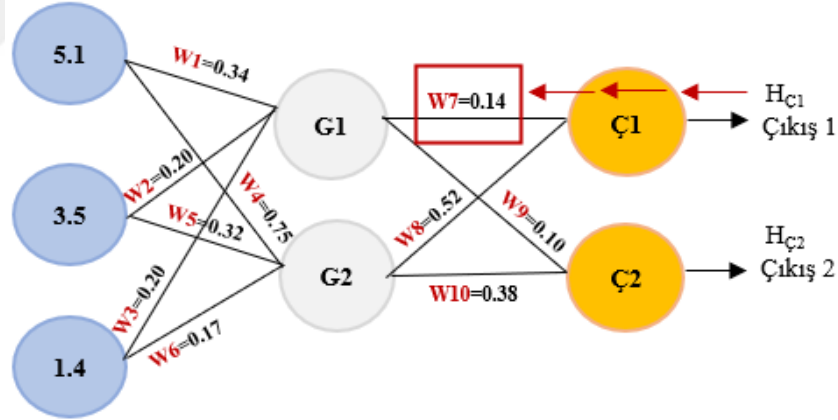
$$H_{Toplam} = \frac{1}{2} \sum_{i=1}^n H_i^2 \quad (3.10)$$

$$H_{Ç1} = 1 - 0.6566 = 0.3434$$

$$H_{Ç2} = 0 - 0.6157 = -0.6157$$

$$H_{Toplam} = (1/2) * (0.1179 + 0.3790) = 0.2484$$

Yukarıdaki hesaplamada görüldüğü üzere, Şekil 3.19'da sunulan YSA'nın toplam hatası 0.2484'tür ve bu değer, hatasızlığın ifadesi olan 0'dan büyük olduğundan hata ağı geri yayılarak ağırlık güncellemeleri ile azaltılmalıdır. Hatanın geriye yayılımı, gizli katman ile çıkış katmanı arasındaki ve gizli katmanlar arası veya giriş katmanı ile gizli katman arasındaki ağırlıkların güncellenmesi ile gerçekleştirilir (Öztemel, 2012). Şekil 3.22'de hatanın, W7 ağırlığına geri yayılımının şematik yapısı sunulmuştur.



Şekil 3.22. YSA'nın Toplam Hatasının W7 ağırlığına Geri Yayılması

Şekil 3.22'de görüldüğü üzere, Çıkış 1'in hata değeri olan $H_{Ç1}$ 'in geriye doğru yayılması ile gizli katman ile çıkış katmanı arasında bulunan W7 ağırlığının değişim miktarı bulunur. Bir gizli katmandaki i nöronunu çıkış katmanında j nöronuna bağlayan bağlantının ağırlık değişimi, herhangi bir t iterasyonunda Denklem 3.11'deki gibidir. Bu değişim miktarı, mevcut ağırlık ile toplanarak yeni ağırlık değeri bulunur.

$$\Delta W_{ij}(t) = \lambda S_j C_i + \alpha \Delta W_{ij}(t-1) \quad (3.11)$$

Denklem 3.11’de verilen gizli katman ile çıkış katmanı arasındaki ağırlık değişim formülasyonunda, λ öğrenme katsayısını ve α momentum katsayısını temsil etmektedir. Öğrenme katsayısı, ağırlıkların değişim miktarının bir sabiti iken momentum katsayısı ağırlık eğitimi aşamasında yerel optimum bir noktaya takılıp kalmamasını sağlayan bir sabittir. S_j ise j çıktı ünitesinin hatasını göstermektedir ve Sigmoid aktivasyon fonksiyonu kullanılan YSA’da Denklem 3.12’deki gibi hesaplanır.

$$S_j = C_j(1 - C_j)H_j \quad (3.12)$$

Şekil 3.22’de, Çıkış 1’in hata değeri olan H_{C1} ’in geriye doğru yayılması sonucunda gizli katmanda bulunan G1 nöronu ile çıkış katmanında bulunan Ç1 nöronunu bağlayan W7 ağırlığının güncellenmesi $\lambda=0.1$ ve $\alpha=1$ için aşağıdaki gibi olur:

$$S_{C1} = 0.6566*(1-0.6566)* 0.3434 = 0.0774$$

$$S_{C2} = 0.6157 *(1-0.6157)* (-0.6157) = -0.1456$$

$$\Delta W7(t) = 0.1*0.0774*0.9379+1*0 = 0.0072$$

$$W7_{\text{yeni}} = 0.14 + 0.0072 = 0.1472$$

Gizli katman ile çıkış katmanı arasında gerçekleştirilen ağırlık değişim miktarı ve yeni ağırlıkların hesabı benzer şekilde giriş katmanı ve gizli katman arasında da gerçekleştirilir. Burada dikkat edilmesi gereken nokta: Gizli katman ile çıkış katmanı arasındaki her bir ağırlığın güncellenmesinde ilgili çıkış katmanındaki nöronun hatası dikkate alınırken, giriş katmanı ile gizli katman arasındaki ağırlıkların güncellenmesinde çıkış katmanındaki bütün nöronların hatası dikkate alınır. Bunun temel sebebi, en son çıkış katmanına gelen bilgiler giriş ve gizli katmandan gelmektedir. Bu nedenle, giriş ve gizli katman arasındaki ağırlıkların güncellenmesinde çıkış katmanındaki bütün hatalardan payını alması gerekir (Öztemel, 2012).

Giriş katmanındaki m nöronunu ara katmandaki i nöronuna bağlayan bağlantının ağırlık değişimi, herhangi bir t iterasyonunda Denklem 3.13’teki gibidir. Bu değişim miktarı mevcut ağırlık ile toplanarak yeni ağırlık değeri bulunur.

$$\Delta W_{mi}(t) = \lambda S_i C_m + \alpha \Delta W_{mi}(t-1) \quad (3.13)$$

S_i , i çıktı ünitesinin hatasını göstermektedir ve Sigmoid aktivasyon fonksiyonu kullanılan YSA'da Denklem 3.14'deki gibi hesaplanır.

$$S_i = C_i(1 - C_i) \sum_t S_t W_{it} \quad (3.14)$$

Şekil 3.22'de verilen YSA'da, giriş katmanında bulunan 5.1 değerindeki nöron ile gizli katmanda bulunan G1 nöronunu bağlayan W1 ağırlığının güncellenmesi $\lambda = 0.1$ ve $\alpha = 1$ için aşağıdaki gibi olur:

$$S_{G1} = 0.9379 * (1 - 0.9379) * (0.14 * 0.0774 + 0.10 * (-0.1456)) = -0.0002$$

$$\Delta W1(t) = 0.1 * (-0.0002) * 5.1 + 1 * 0 = -0.0001$$

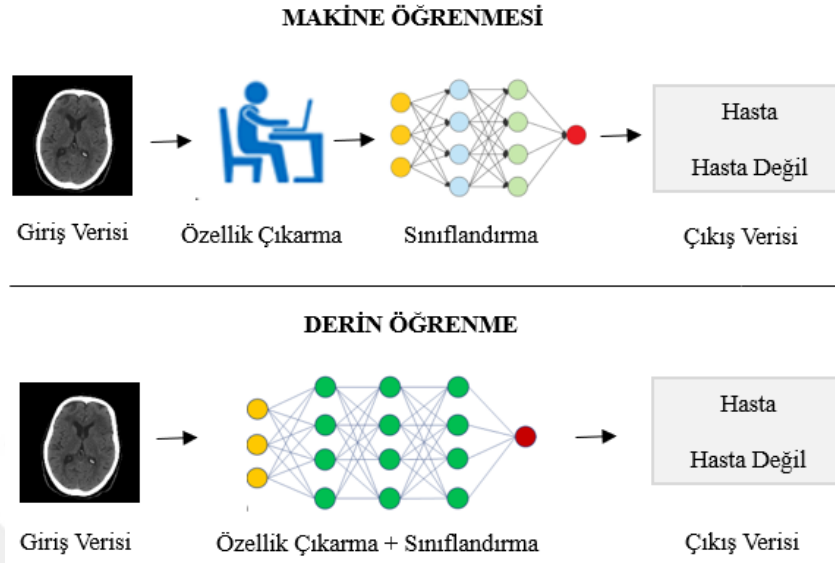
$$W1_{\text{yeni}} = 0.34 + (-0.0001) = 0.3399$$

Şekil 3.22'de verilen YSA üzerinden anlatımı sunulan GYA, YSA'daki bütün nöronlar arasındaki ağırlıklara uygulanarak güncellemeler yapılır. Güncellemeleri yapılan ağırlıklar ile YSA, tekrar İYA'ya sunularak tekrar hata değeri hesaplanır. Hata değeri sıfıra yaklaşıncaya kadar ağırlık güncelleme işlemleri iteratif olarak tekrarlanır.

3.3.3. Derin Öğrenme

Bir MÖ modelinde giriş verilerini temsil eden doğru özelliklerin çıkarılması, öğrenme başarısını etkileyen temel faktörlerden biridir. Bundan dolayı, çıkarılan özellik setleri, MÖ algoritmalarının davranışlarını ve başarılarını etkilediğinden özellik çıkarma aşaması, MÖ'de önemli bir ön işleme adımdır (Guyon & Elissee, 2003; Singh & SivaBalakrishnan, 2015). Bu şekilde, verileri doğru temsil eden özelliklerin çıkarılması zorunluluğunu ortadan kaldıran, özellik çıkarımını ağ içerisinde gerçekleştiren bir ağ yapısı önerilmiştir. Önerilen bu ağ modeli MÖ'nün bir alt dalı olan DÖ'dür. DÖ, yapısında bulundurduğu birden fazla gizli katman ile karmaşık veri kümelerinde analizler yaparak tahminleme işlemi yapan, temelinde YSA'yı kullanan bir MÖ yöntemidir. DÖ, matematiksel evrişim ve filtreleme işlemleri ile özellik çıkarımı yaparken MÖ'ye kıyasla verilerden daha derinlemesine ancak daha düşük boyutlarda özellikler elde eder. DÖ'yü diğer MÖ yöntemlerinden ayıran temel nokta, algoritmalarının öğrenme şeklidir. Şekil 3.23'te görüldüğü üzere, özellik çıkarma işlemi

MÖ'de özellik çıkarma teknikleri ile gerçekleştirilirken, DÖ'de ise bu teknikleri kullanmaya gerek kalmadan veri setinden otomatik olarak yapılmaktadır.

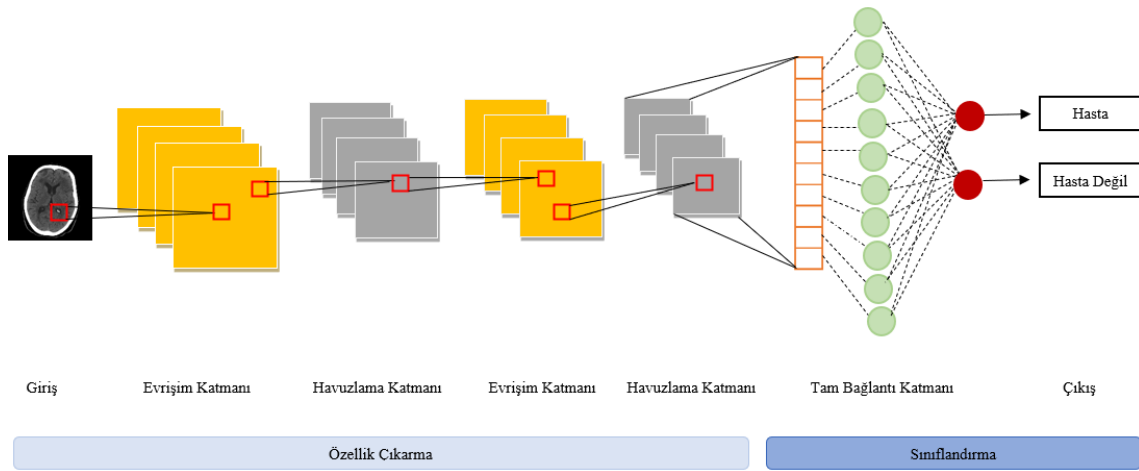


Şekil 3.23. MÖ ve DÖ

DÖ, MÖ'nün aksine etiketsiz verileri ham halde işleyebilir ve farklı veri kategorilerini birbirinden ayıran özellik kümelerini otomatik olarak belirleyebilir. DÖ'de her veri setine uygun özellik çıkarımının otomatik olarak yapılması, geliştirilecek model için en uygun özellik setinin elde edilmesine ve daha büyük veri kümelerinin kullanılmasına olanak sağlamaktadır. Bu da dolayısıyla model performansını yüksek seviyelere ulaştırmaktadır.

3.3.3.1. Evrimsel Sinir Ağları

Bilgisayarla görü alanında başarılı sonuçlar üreten birçok DÖ mimarisi geliştirilmiştir. Geliştirilen bu mimarilerden biri olan ESA resim ve video gibi görsel veri işleme için geliştirilmiş özel bir YSA mimarisidir. Bir ESA, girdi görüntüsünü alarak görüntüdeki çeşitli nesnelere önem (ağırlık ve bias) atayan ve nesnelerin birbirinden ayırt edilebilmesini sağlayan, aynı zamanda nesnelerin birbirleriyle olan ilişkilerini çıkarabilen bir DÖ mimarisidir. ESA, sahip olduğu evrişim, havuzlama, tam bağlantı ve sınıflandırma katmanları sayesinde görüntülerdeki ayırt edici özellikleri saptayarak ve uyumluluk hesaplamaları yaparak YSA yoluyla bir yapay öğrenme gerçekleştirir (Jauro ve ark., 2020). Şekil 3.24'te temel bir ESA yapısı gösterilmiştir.



Şekil 3.24. ESA'nın Temel Yapısı

ESA'nın iki temel bölümü vardır: Birincisi görüntüden özellikler çıkaran ve bu özellikleri analiz eden evrişim ve havuzlama katmanları, ikincisi ise evrişim ve havuzlama katmanlarının çıktısı olarak ürettiği veriyi giriş olarak alan, en iyi sınıf etiketini tahmin ederek sınıflandırma yapan tam bağlantı katmanıdır.

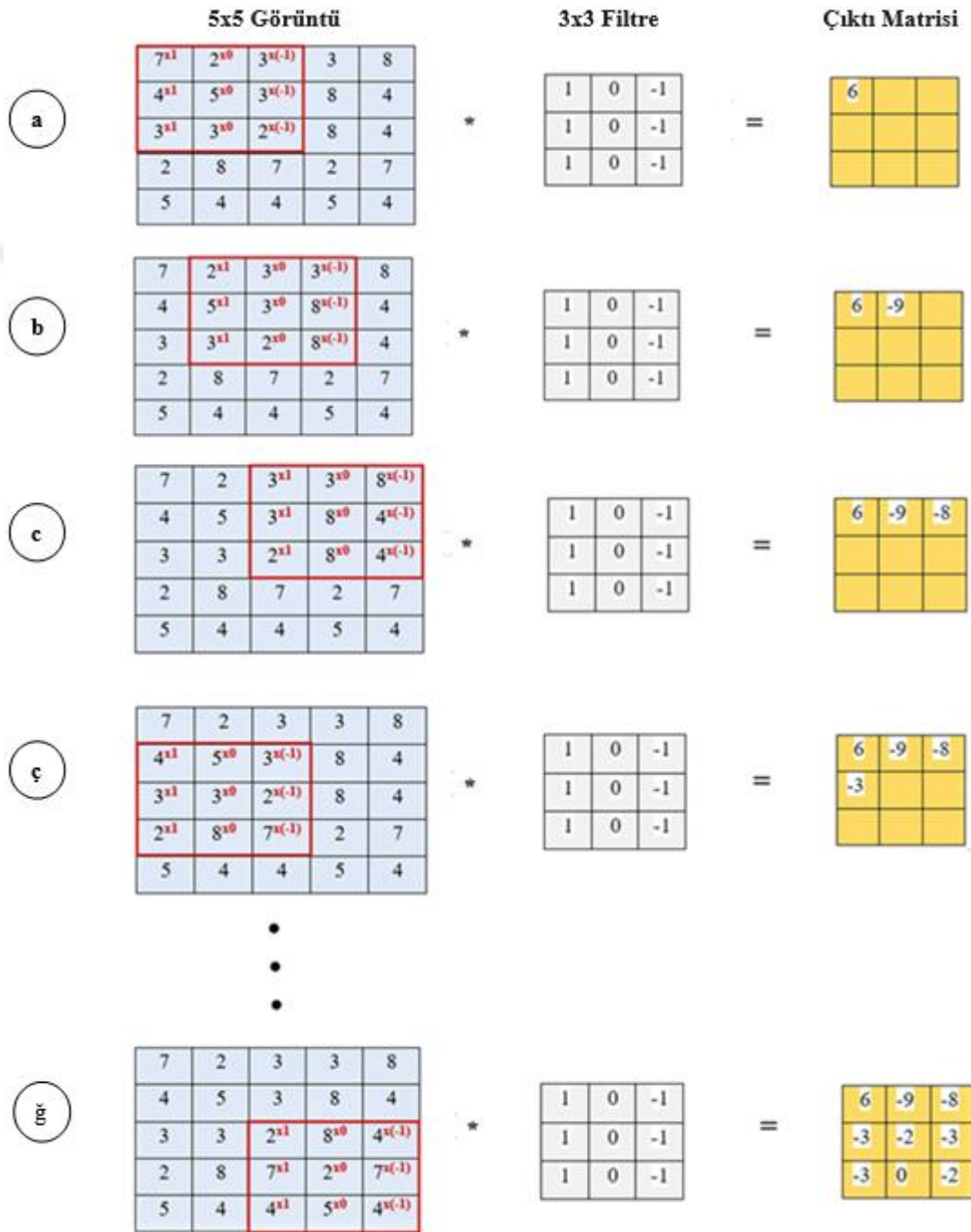
3.3.3.1.1. Evrişim Katmanı

Evrişim katmanı, ESA'da girdi görüntüsünden özellik çıkaran ilk katmandır. Dönüşüm katmanı olarak da bilinen bu katmanda temel olarak, belirli filtrelerin tüm görüntü üzerinde dolaştırılma işlemi olan evrişim yapısı (Ülker & İnik, 2017). Evrişim, girdi verilerinin her bir pikselini kullanarak görüntü özelliklerini öğrenir ve pikseller arasındaki ilişkiyi korur. Bunu, görüntü matrisleri üzerinde kaydırmalı olarak dolaşan, görüntü matrislerinden daha küçük boyutlu filtre matrisleri kullanarak gerçekleştirir. Evrişim, girdi ve filtre matrisi arasındaki matematiksel bir işlem olup, çıktısı da yine bir matristir. Görüntü piksellerinden özellikler çıkaran evrişim işlemi, Şekil 3.25'te gösterilmiştir. Şekil 3.25'te görüldüğü üzere, giriş görüntü verisi boyutu 5×5 'lik, görüntü üzerine uygulanacak filtre boyutu ise 3×3 'lük matris boyutundadır. Giriş görüntüsüne filtre uygulanırken ilk olarak filtre görüntünün sol üst köşesine yerleştirilir ve iki matris arasında örtüşen değerler aşağıdaki gibi birbiri ile çarpılır ve tüm sonuçlar toplanarak çıktı matrisinin ilk elmanı olarak yazılır (a).

$$7x1 + 4x1 + 3x1 + 2x0 + 5x0 + 3x0 + 3x(-1) + 3x(-1) + 2x(-1) = 6$$

Ardından, filtre matrisi görüntü matrisi üzerinde bir piksel sağa kaydırılarak yapılan işlem tekrarlanır (b,c). Birinci satır işlemi bittikten sonra bir piksel aşağı kaydırılarak

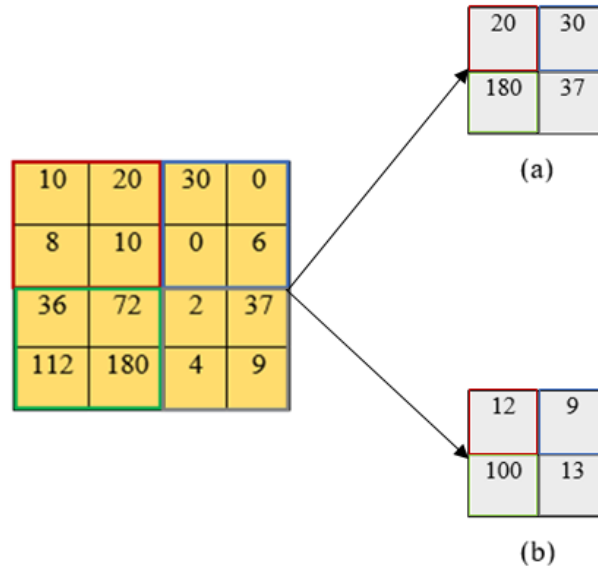
ikinci satıra geçilir ve aynı işlemler tekrarlanır (ç). Tüm işlemler bittikten sonra, çıktı matrisi tamamen oluşur (ğ). Yapılan evrişim işlemi ile girdi olarak verilen görüntü verisi üzerinden özellikler tespit edilmiş olur. ESA algoritmalarında modelin öğrendiği parametreler filtrelerdeki değerlerdir. Model sürekli olarak bu değerleri günceller ve özellikleri daha iyi tespit etmeye çalışır.



Şekil 3.25. Evrişim Katmanı Çalışma Prensibi

3.3.3.1.2. Havuzlama Katmanı

Görüntü verileri tek bir özellikten oluşmadığı için genel olarak birden fazla filtre uygulanması gerekir. Bundan dolayı, bir ESA mimarisi birden fazla evrişim katmanına sahiptir. ESA'nın bir diğer temel yapı taşı olan havuzlama katmanı ise birbirini izleyen birden fazla evrişim katmanları arasına eklenerek büyük boyutlardaki görüntülerin, parametre ve hesaplama miktarını aşamalı bir şekilde azaltarak görüntü boyutunu küçültür. Bu işlem, hem her bir haritanın boyutunu azaltır hem de görüntülerdeki rotasyonel ve konumsal değişmez olan baskın özelliklerin çıkarılmasını sağlayarak modelin etkili bir şekilde eğitim sürecine devam etmesini sağlar. Şekil 3.26'da görüldüğü üzere, iki tür havuzlama yöntemi vardır: Evrişim katmanından gelen özellik haritasının kapsadığı maksimum değeri veren maksimum havuzlama ve yine evrişim katmanından gelen özellik haritasının kapsadığı tüm değerlerin ortalamasını veren ortalama havuzlamadır.

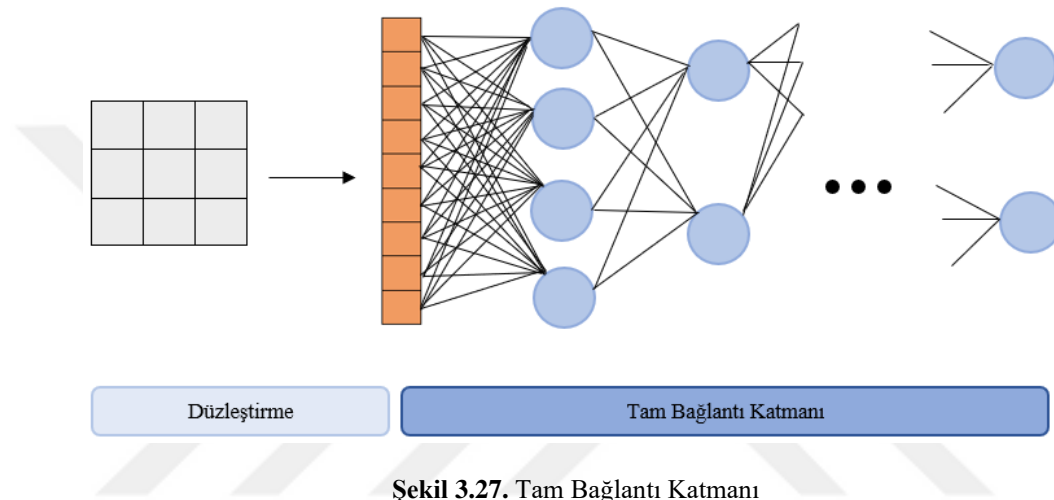


Şekil 3.26. a) Maksimum Havuzlama b) Ortalama Havuzlama

Şekil 3.26'da görüldüğü üzere, maksimum havuzlamada filtre içinde kalan görüntü bölümünde bulunan piksellerdeki maksimum değer, çıkış piksel değeridir. Ortalama havuzlamada ise filtre içindeki piksellerin ortalaması çıkış piksel değerini oluşturmaktadır.

3.3.3.1.3. Tam Bağlantı Katmanı

ESA'nın son katmanı olan tam bağlantı katmanı, son adımındaki bütün nöronları tek boyutlu bir vektöre dönüştürerek sınıflandırıcı olarak kullandığı bir YSA'ya bağlar. Sınıflandırıcı modele verilecek olan girdi, tek boyutlu bir vektör olmak zorundadır. Bu yüzden bu katmanda, birçok kez evrişim ve havuzlama katmanlarından geçen matris halindeki veriler düzleştirme (flattening) işlemi ile doğrusal bir vektör haline getirildikten sonra sinir ağları ile öğrenme işlemi gerçekleştirir. Tam bağlantı katmanının şematik yapısı Şekil 3.27'de gösterilmiştir.



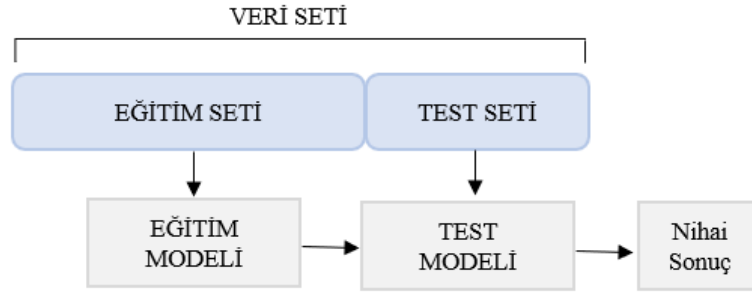
Şekil 3.27. Tam Bağlantı Katmanı

3.4. Sınıflandırma Değerlendirme Metrikleri

Sınıflandırma algoritmaları ile oluşturulan modellerin ne kadar doğru sonuçlar ürettiğinin değerlendirilmesi için belirli yöntemler ve değerlendirme metrikleri kullanılmaktadır. Bir sınıflandırıcının performansında kullanılan sınıflandırma algoritmasına ek olarak, eğitim ve test verilerinin seçim şekli de etkilidir (Haciefendioğlu, 2012). Sınıflandırma modellerinin başarısında, eğitim ve test setleri seçme şekli doğrudan etkili olduğundan çeşitli yöntemler geliştirilmiştir. Bunlar: Hold-Out, Rastgele Örnekleme, k-Katlı Çapraz Doğrulama (k-KÇD) ve Üçlü Ayırma yöntemleridir.

3.4.1. Hold-Out Yöntemi

Veri setinin belirli yüzdelerle eğitim ve test seti olmak üzere ikiye ayırma metottur ve genel olarak 2/3'ü eğitim, 1/3'ü test seti olarak ayrılır. Bu yöntemin genel çalışma prensibi Şekil 3.28'de sunulmuştur.



Şekil 3.28. Hold-Out Yöntemi Çalışma Prensipleri

Hold-Out yönteminde, eğitim seti sınıflandırıcı modelin eğitim aşamasında kullanılarak model için en iyi parametre değerleri belirlenir. Belirlenen en iyi parametrelerle eğitilen sınıflandırıcı modeline test seti sunularak modelin nihai sonucu üretilir. Bu yöntemin iki dezavantajı vardır: Birincisi, mevcut veri seti küçük ise test seti için seçilecek olan veri sayısı yetersiz olmaktadır. İkincisi dezavantaj ise, eğitim ve test setlerinin yüzdesel rastgele ayrılığı için her iki sete sınıf verilerinin dengeli olarak dağılacağına garanti yoktur.

3.4.2. Rastgele Örneklem Yöntemi

Hold-Out yönteminde gerçekleştirilen veri setinin, eğitim ve test setlerine ayırma işleminin k defa tekrarlanmasıdır. İteratif olarak k kez tekrarlan bu ayırma işleminden elde edilen k adet başarının ortalaması model başarısıdır. Bu yöntemde, veriler kullanıcının belirlediği orandaki dengede setlere dahil edilmektedir, bu da Hold-Out yönteminde doğan verilerin setlere dengesiz dağılımı sorununa çözüm niteliğindedir.

3.4.3. k-Katlı Çapraz Doğrulama Yöntemi

k -KÇD, sınıflandırıcı modelin görmediği veriler üzerinde performansını güçlendirerek model doğruluğunu arttıran bir yeniden örneklem yöntemidir. k -KÇD'de veri seti rastgele karıştırıldıktan sonra k adet gruba ayrılır. Gruplardan bir tanesi test seti kalan $k-1$ tanesi de eğitim seti olarak kullanılır. Eğitim seti kullanılarak eğitim modeli oluşturulur ve test seti ile eğitim modeli değerlendirilir. Ve bu işlemler k iterasyonda tekrarlanarak elde edilen k adet başarının ortalaması olan model başarısı elde edilir. Seçilen k kat sayısı ne kadar büyük seçilirse veri o kadar fazla parçaya ayrılır ve sınıflandırıcı tahmininin daha doğru, gerçekten sapma değeri daha küçük, varyans ve hesaplama zamanı ise büyük olacaktır (Saharidis, 2011). k değerinin en sık kullanılan

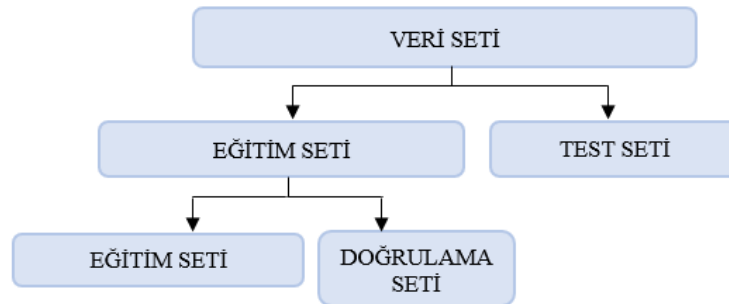
değeri 10'dur (Guil-Reyes & Daza, 2011; Martinasek, 2014; Olson & Deren, 2008). Şekil 3.29'da $k=5$ için k-KÇD yönteminin çalışma prensibi sunulmuştur.



Şekil 3.29. k-KÇD Yöntemi Çalışma Prensibi

3.4.4. Üçlü Ayırma Yöntemi

Hold-Out yöntemine benzer olan Üçlü Ayırma yöntemi, Şekil 3.30'daki gibi veri setini eğitim, doğrulama ve test seti olarak üç gruba ayırmaktadır. Hold-Out yönteminden farkı, eğitim seti ile oluşturulan eğitim modeli üzerinde doğrulama seti ile performans testi yapılarak eğitim modelinin hatası en aza indirgenir. Son olarak, test seti modele sunulurken nihai performans elde edilir.



Şekil 3.30. Üçlü Ayırma Yöntemi

3.5. Sınıflandırma Performans Metrikleri

Geliştirilen eğitim ve test setleri seçme yöntemleri ile sınıflandırıcı modellerinden en doğru ve yüksek başarı elde etme sağlanmıştır. Veri setini girdi olarak alan sınıflandırıcı modelleri, çeşitli kriterlerde performans ölçütleri ile doğrulukları hakkında bilgiler vermektedir. Çeşitli kriterlerdeki bu performans ölçütlerinin temeli, Karmaşıklık Matrisi (KM) olarak adlandırılan bir hata tablosuna dayanmaktadır (Japkowicz & Shah, 2011). KM, gerçek sonuçlarının bilindiği test setinin

sınıflandırılmasında modelin performansını görselleştirmek için kullanılan bir tablodur. KM'nin temel yapısı, ikili sınıflandırma modeli üzerinden Şekil 3.31'de sunulmuştur.

		Gerçek Sonuçlar	
		Pozitif	Negatif
Tahmini Sonuçlar	Pozitif	Doğru-Pozitif (DP)	Yanlış-Pozitif (YP)
	Negatif	Yanlış-Negatif (YN)	Doğru-Negatif (DN)

Şekil 3.31. İkili Sınıflandırma için KM

KM, tahmini sonuçların gerçek sonuçlara karşı ürettiği doğru ve yanlış veri sayısını sunan özel bir iki boyutlu tablodur. Matrisin her satırı, tahmin edilen sınıftaki örnekleri temsil ederken her sütunu o örneklerin gerçek sınıftaki örneklerini temsil eder. Şekil 3.31'de görüldüğü üzere, pozitif ve negatif etiketli iki sınıflı bir sınıflandırıcı modeli için:

Doğru-Pozitif (DP): Sınıflandırıcı tarafından doğru tahmin edilen pozitif sınıf etiketine sahip veri grubunu temsil eder.

Doğru-Negatif (DN): Sınıflandırıcı tarafından doğru tahmin edilen negatif sınıf etiketine sahip veri grubunu temsil eder.

Yanlış-Pozitif (YP): Sınıflandırıcı tarafından yanlış tahmin edilen pozitif sınıf etiketine sahip veri grubunu temsil eder.

Yanlış-Negatif (YN): Sınıflandırıcı tarafından yanlış tahmin edilen negatif sınıf etiketine sahip veri grubunu temsil eder.

KM kullanılarak bir sınıflandırıcının performansını ölçmek için bazı metrikler oluşturulmuştur. Bunlar; Doğruluk, Duyarlılık, Kesinlik ve Özgüllük, F-Ölçütü, Matthews Korelasyon Katsayısı (MKK) ve ROC Eğrisidir.

3.5.1. Doğruluk

Doğruluk performans metriği, eğitim seti kullanılarak oluşturulan sınıflandırıcı modelin, test setindeki verileri doğru sınıflandırma oranını sunar. Bu metrik, Denklem 3.15'te sunulduğu üzere, modelde doğru tahmin edilen alanların toplam veri setine oranı ile hesaplanmaktadır. Bu da, bütün sınıf etiketleri aynı derecede önemseyerek sınıflandırıcının toplam performansını değerlendirmeye yardımcı olur. Buna ek olarak, bir sınıflandırıcı modelin yanlış sınıflandırma oranına hata oranı denir. Hata oranı, yüzde yüz doğruluk ifade eden 1 sayısından hesaplanan doğruluk değerinin çıkarılması ile bulunur.

$$\text{Doğruluk} = \frac{DP + DN}{DP + DN + YP + YN} \quad (3.15)$$

Sınıflandırma çalışmalarında, veri setindeki farklı sınıflar arasında orantısız derecede örnek olması durumuna dengesiz veri seti denir. Özellikle dengesiz veri setleri için bir sınıflandırıcı modelin değerlendirmesinde doğruluk metriği tek başına yeterli olamamaktadır. Örneğin, dengesiz veri seti ile gerçekleştirilmiş ve 100 kişiden 90'ının sağlıklı olduğunu tespit eden bir tanı testi çalışmasında sınıflandırıcı modelin başarısı %90 olacaktır. Bu da, modelin bundan sonra gelecek gerçek hastaları %90 sağlıklı bulacağını göstermektedir. Bu durum, sadece doğruluk parametresi ile değerlendirilmiş tanı testlerinin hasta bireyleri sağlıklı bireylerden ayıramadığını göstermektedir. Bundan dolayı, bir sınıflandırıcı modelinin performansı değerlendirilirken diğer performans metrikleri de kullanılmalıdır.

3.5.2. Duyarlılık, Kesinlik ve Özgüllük

Pozitif ve negatif sınıfların örnekleri arasında ayırım yapmak için modelin yüksek DP ve düşük YP oranına sahip olması gerekir. Bir sınıflandırıcı model oluştururken hedeflenen standart budur. Sınıflandırıcının pozitif sınıfları saptama yeteneğine duyarlılık denir. Yani duyarlılık, pozitif sınıfa ait olduğu bilinen bir verinin, sınıflandırma sonucunda da ne kadarını pozitif sınıf olarak etiketlendiğini gösteren bir metriktir. Ve bir sınıflandırıcının duyarlılığı, KM'den elde edilen DP sayısının gerçek pozitif sınıftaki örnek sayısına bölünmesi ile elde edilir. DP oranı yani duyarlılık performans metriği Denklem 3.16'da verilmiştir.

$$Duyarlılık = \frac{DP}{DP + YN} \quad (3.16)$$

Kesinlik performans metriği ise pozitif olarak etiketlenen değerlerden gerçekte kaç tanesinin doğru olarak pozitif etiketlendiğini gösteren bir metriktir. Özetle, duyarlılık metriği sınıflandırıcının pozitif sınıftan ne kadarını pozitif olarak sınıflandırdığının ölçüsü iken kesinlik sınıflandırıcı tarafından pozitif tahminlenen verilerin gerçekte ne kadarının pozitif sınıfa ait olduğunu gösteren oranlardır. Kesinlik ve Duyarlılık, bir sınıflandırıcının test performansını ölçerek bilgi çıkarımını değerlendirmek için kullanılan ölçütlerdir ve bilgi çıkarımı gerçekleştiren YSA gibi uygulamalara uygundur (Abeel ve ark., 2009; Huang & Bader 2009). Kesinlik metriği, doğru olarak etiketlenmiş pozitif verilerin, tahmini pozitif sınıfı olarak etiketlenmiş veri sayısına oranı olarak hesaplanmaktadır ve formülasyonu Denklem 3.17’de verilmiştir.

$$Kesinlik = \frac{DP}{DP + YP} \quad (3.17)$$

Özgüllük, sınıflandırıcının negatif sınıfları saptama yeteneğini gösteren sınıflandırıcı performans metriğidir. Yani, negatif sınıfa ait olduğu bilinen bir verinin, sınıflandırma sonucunda da ne kadarını negatif sınıf olarak etiketlendiğini gösteren bir orandır. Denklem 3.17’de görüldüğü üzere, bir sınıflandırıcının özgüllüğü KM’den elde edilen DN sayısının gerçek negatif sınıftaki örnek sayısına bölünmesi ile elde edilir. Özetle, duyarlılık pozitif sınıfların doğru tahminlenmesine odaklanırken özgüllük negatif sınıfların doğru tahminlenmesine odaklanır. Bu sayede bu iki performans metriği, bir sınıflandırıcının doğru sınıfı yanlış sınıftan ne derece ayırabildiğini gösterir. Bir sınıflandırıcı için duyarlılığı ne kadar yüksekse yanlış negatiflik olasılığı o kadar azalır, özgüllüğü ne kadar yüksekse, yanlış pozitiflik olasılığı o kadar azalır.

$$Özgüllük = \frac{DN}{DN + YP} \quad (3.18)$$

3.5.3. F-Ölçütü

Kesinlik ve duyarlılık ölçütleri, sınıflandırıcı modelleri için tek başlarına anlamlı bir karşılaştırma sonucu sunmamaktadır. İki ölçütü bir arada değerlendiren F-Ölçütü,

kesinlik ve duyarlılık değerlerinin harmonik ortalamasını elde etmektedir. F-Ölçütü genel olarak, F_1 , $F_{0.5}$ ve F_2 olarak kullanılır ve 0 ile 1 arasında değer alır. Doğru tahminler yapan bir sınıflandırıcının F-Ölçütü değerinin 1'e yakın olması beklenmektedir (Japkowicz, 2011). F-Ölçütünün formülasyonu Denklem 3.19'da verilmiştir.

$$F_{\beta} - \text{Ölçütü} = \frac{(1 + \beta^2) \times (\text{Kesinlik} \times \text{Duyarlılık})}{\beta^2 + \text{Kesinlik} \times \text{Duyarlılık}} \quad (3.19)$$

Burada β değeri, genel olarak $\beta = 1$ olarak alınarak F_1 -Ölçütü ile kesinlik ve duyarlılık değerlerinin eşit ağırlıklı olarak hesaplanması sağlanmaktadır. Sadece YN ya da YP değil, tüm hata değerlerini içine alan F-Ölçütü özellikle dengesiz veri setleri için oldukça önemli bir performans metriğidir.

3.5.4. Matthews Korelasyon Katsayısı

Doğruluk, duyarlılık ve kesinlik gibi performans metrikleri dengesiz veri setleri için tek başlarına anlamlı sonuçlar üretmediğinden MKK metriği geliştirilmiştir. MKK, gerçek veriler ile tahmin edilen veriler arasındaki korelasyon ilişkisine göre performansı ölçen bir metriktir. Denklem 3.20'de formülasyonu verilen bu ölçütün, diğer metriklerin aksine KM'de yer alan bütün değerleri kullanmaktadır.

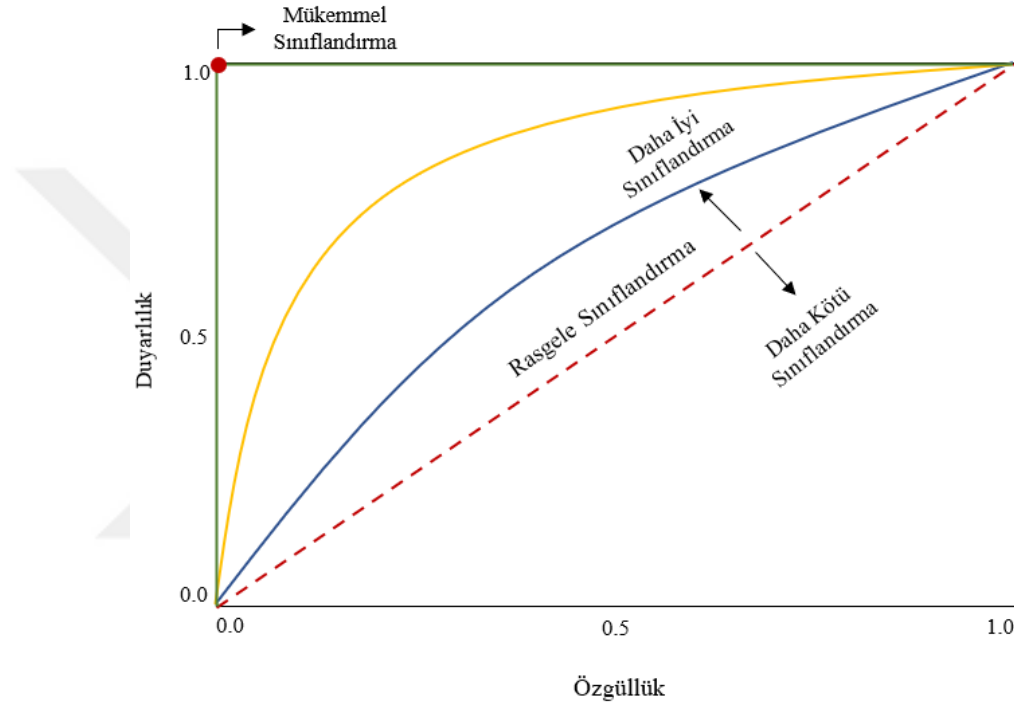
$$MKK = \frac{DP \times DN - YP \times YN}{\sqrt{(DP + YP) \times (DP + YN) \times (DN + YP) \times (DN + YN)}} \quad (3.20)$$

MKK, -1 ile +1 arasında değerler almaktadır. Sınıflandırıcı modeli, 1'e yakın değer almışsa doğru tahminlerde bulunduğunu, 0 değerini almışsa rastgele tahminlerde bulunduğunu ve -1'e yakın değer almışsa yanlış tahminlerde bulunduğunu göstermektedir (Akosa, 2017).

3.5.5. ROC Eğrisi

ROC eğrisi, sinyallerin doğru tanımlanabilmesi için geliştirilmiş bir analiz çalışmasıdır. 1967 yılında Lusted'in klinik karar desteği için ROC eğrisi kullanımını önermesi ile medikal görüntüleme cihazlarında bu analiz kullanılmıştır ve bu kullanım tıpta tanı testlerinin performans değerlendirilmesinde giderek artmıştır (Ertorsun ve ark.,

2009). ROC eğrisi, duyarlılık ve özgüllük arasındaki ilişkinin grafiksel bir gösterimidir ve bu eğri altında kalan alan, 0 ile 1 arasında sınıflandırıcıların performansını test etmek için kullanılır. ROC grafiğinin x ekseninde YP oranı yani özgüllük performans metriği yer alırken, y ekseninde DP oranı yani duyarlılık performans metriği bulunmaktadır. Farklı eşik değerleri için duyarlılık ve özgüllük değerleri hesaplanır ve ROC eğrisindeki her nokta, belirli bir karar eşğine karşılık gelen duyarlılık-özgüllük çiftini temsil eder. ROC grafiği, çeşitli ROC eğrileri ile Şekil 3.32’de sunulmuştur.



Şekil 3.32. ROC Grafiği ve ROC Eğrileri

Sınıflandırıcı modelin performans analizinde, ROC eğrisi (0,0) ile (1,1) aralığında artan bir fonksiyondur ve bu fonksiyon, sınıflandırıcının ürettiği tahminlerin ne kadar doğru olduğunun bir analizidir. Bir sınıflandırıcı, pozitif ve negatif sınıfları birbirinden tam anlamıyla ayırabiliyorsa mükemmel bir sınıflandırma gerçekleşmiş olur. ROC grafiğinde bu durum, YP oranını yani özgüllüğü 0 ve DP oranını yani duyarlılığı 1 şeklinde gösterilir. Şekil 3.32’de sunulan ROC grafiğinde, sol üst (0,1) köşesinde bu durum tanımlanmıştır. ROC grafiğinin sol üst (0,1) köşesine yaklaşıldıkça testlerin ayırt etme gücü artar, sınıflandırma mükemmelliğe yaklaşır (Alpaydin, 2014). Buna göre, ROC grafiğinde (0,0) noktası pozitif sınıflandırma yapılamadığını ve (1,1) noktası mutlak pozitif sınıflandırmaların yapıldığını göstermektedir. Grafiğin köşegeni ise,

rastgele sınıflandırma yapıldığını göstermektedir ve bu köşegen altındaki değerler kötü sınıflandırma olarak nitelendirilmektedir.

Sınıflandırıcı performans metriği olarak ROC eğrisinin altında kalan alan kullanılmaktadır. Bu alan, 0 ve 1 arasında değer üretmektedir ve alan değerinin 1'e yaklaşması sınıflandırıcının doğru tahminlerde bulunduğunu, 0.5 değerini alması sınıflandırıcının rastgele tahminlerde bulunduğunu ve 0.5'in altında aldığı değerlerde ise sınıflandırıcının doğru çalışmadığını göstermektedir (Kılıç, 2013; Hulley ve ark., 2001).

3.6. Veri Madenciliği

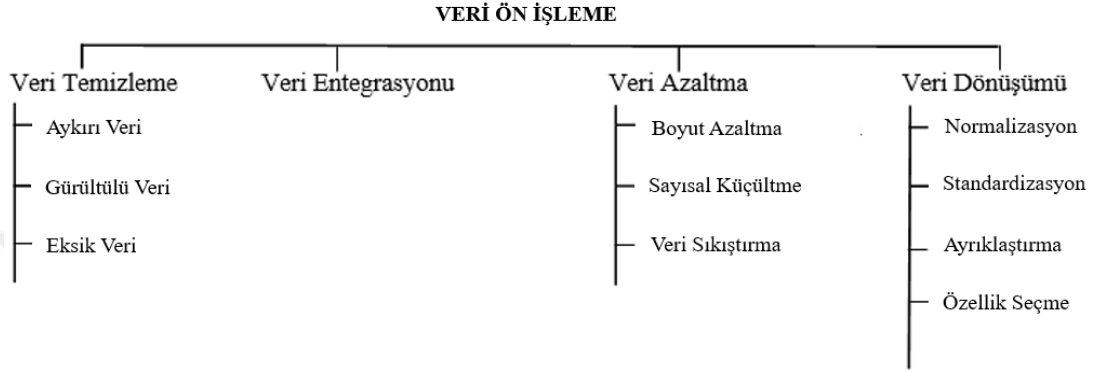
Veri Madenciliği, büyük veri yığınlarından anlamlı ilişkileri ve kuralları ortaya çıkaran bir disiplindir ve YZ yöntemlerinden yararlanır (Fayyad ve ark., 1996; Kulin ve ark., 2021). Verilerden analizler yaparak tahmini modeller oluşturmayı ve bu modelleri optimize etmeyi amaçlayan veri madenciliği, çeşitli istatistiksel yöntemler kullanır. Bundan dolayı, veri madenciliği hem YZ hem de istatistik disiplinlerini kullandığı için multidisipliner bir bilimdir. Bu bilim, verileri analiz etmeden önce onları en etkin hale getirmeyi amaçlar.

3.6.1. Veri Ön İşleme

Biyomedikal veri kümeleri kullanılarak gerçekleştirilen YZ tabanlı teşhis sistemlerinde sınıflandırma performansını azaltan bazı etmenler vardır. Bunlar genellikle veri setlerinde bulunan eksiklikler ve dengesizlikler, tekrarlı veriler veya aykırı değerler gibi durumlardır. Veri setinde bulunan, performansı etkileyen bu durumları optimize etmek başarılı ve doğru bir sınıflandırma modeli oluşturmanın temel adımlarından biridir. Bir veri analizi hem veri ön işleme adımına hem de bu amaçla seçilen tekniklere doğrudan bağlı (Mathiak ve Eckstein, 2014) olduğundan seçilen veri ön işleme tekniklerinin verilerle tümleşik bir yapıda çalışması doğrudan doğruya sınıflandırma performansını etkilemektedir. Bu sebeple, sınıflandırma performansını arttırmak amacıyla veri setlerine uygun çeşitli veri ön işleme teknikleri kullanılmaktadır.

Şekil 3.33'te görüldüğü üzere, veri ön işleme teknikleri metodolojik olarak dört gruba ayrılır. Bunlar veri temizleme, veri entegrasyonu, veri azaltma ve veri dönüştürmedir. Veri temizleme aykırı verilerin giderilmesi, gürültülü verilerin temizlenmesi ve eksik verilerin tamamlanması gibi adımları içerirken veri entegrasyonu

ise farklı kaynaklardan elde edilen verilerin sistematik bir biçimde birleştirilmesini gerçekleştirir. Veri azaltma işlemi veri setlerinde bulunan önemsiz verilerin silinmesi, etkin olmayan özelliklerin çıkarılması veya verilerin sıkıştırılması işlemlerini kapsamaktadır. Son olarak veri dönüşümü ise farklı sistem ve ölçeklerde bulunan verilerin belirli ortak bir ölçeğe taşınarak birbirleriyle ilişkilendirilebilir hale getirilmesidir.



Şekil 3.33. Veri Ön İşleme Teknikleri

3.6.1.1. Eksik Veri Tamamlama

Veri setlerinde eksik verilerin bulunması MÖ'de sıklıkla karşılaşılan temel sorunlardan biridir. Veri setlerini eksik verilerle kullanmak, veri analizi ve karar verme süreçlerini olumsuz etkilemektedir. Bundan dolayı, eksik verileri tamamlamak için birçok yöntem geliştirilmiştir. Eksik verileri tamamlayacak yöntemin başarısı verinin karakteristiği ve eksik verinin tipi gibi birçok faktörden etkilenmektedir (Albayrak ve ark., 2017). Bir veri analizi hem veri ön işleme adımına hem de bu amaçla seçilen tekniklere doğrudan bağlıdır. Bu sebeple, veri setlerindeki eksik verileri tamamlayacak uygun yöntemi seçmek, MÖ çalışmalarında önemli ve gerekli bir ön işlemdir.

3.6.1.1.1. KNN Algoritması ile Eksik Veri Tamamlama

KNN algoritması, temel olarak, mesafeye dayalı bir sınıflandırma algoritmasıdır ve eksik verilerin tamamlanmasında da bu sınıflandırma mantığını kullanmaktadır. KNN yönteminde, öncelikle k parametre değeri belirlenir. Buradaki k parametresi en yakın komşu sayısını yani en yakın gözlem sayısını belirtmektedir. k parametresi belirlendikten sonra eksik değer ve etrafında eksik değer içermeyen gözlemler arasında mesafe ölçümü yapılarak eksik değere ait k adet en yakın gözlem keşfedilir. Keşfedilen bu gözlemler, eksik veri noktasına en benzer gözlemler olarak nitelendirilir ve bu

gözlemlerin ortalaması ile eksik değer tamamlanır. Mesafe ölçümünde, Öklid yöntemi yeni veriler ve önceden bilinen sınıf verileri arasındaki mesafeyi hesaplamak için kullanılır (İdri ve ark., 2016). Eksik verileri tamamlamak için uygulanan KNN algoritmasının çalışma prensibi, Tablo 3.3 üzerinden anlatılmıştır.

Tablo 3.3. OSR Veri Setindeki 5 Örneğin 5 Özelliği Üzerinden KNN ile Eksik Veri Tamamlama

	ALT	CRP	AST	WBC	LDH
H1	18	29	36	2.9	257
H2	22	.	25	3.5	207
H3	.	193.7	31	4.6	220
H4	33	318.7	96	16.5	651
H5	14	16.5	21	4.9	.

Tablo 3.3'te görüldüğü üzere, H3 hastasının ALT özelliği eksik veri içermektedir. Bu eksik verinin KNN algoritması ile tamamlanmasında: İlk olarak, k=2 seçilerek k parametre değeri belirlenir. Sonrasında tamamlanacak olan eksik verinin bulunduğu satıra en yakın 2 komşu satır belirleneceğinden, Denklem 3.21 ve Denklem 3.22 ile mesafe ölçümleri yapılır:

$$Mesafe(X, Y) = \sqrt{Ağırlık \times \left(\sum_{i=1}^n (X_i - Y_i)^2 \right)} \quad (3.21)$$

$$Ağırlık = \frac{\text{Toplam Koordinat Sayısı}}{\text{Eksik Veri İçermeyen Koordinat Sayısı}} \quad (3.22)$$

$$H3-H1: [4/4 * ((193.7-29)^2 + (31-36)^2 + (4.6-2.9)^2 + (220-257)^2)]^{1/2} = 7.5089$$

$$H3-H2: [4/3 * ((31-25)^2 + (4.6-3.5)^2 + (220-207)^2)]^{1/2} = 16.3729$$

$$H3-H4: [4/4 * ((193.7-318.7)^2 + (31-96)^2 + (4.6-16.5)^2 + (220-651)^2)]^{1/2} = 18.6338$$

$$H3-H5: [4/3 * ((193.7-16.5)^2 + (31-21)^2 + (4.6-4.9)^2)]^{1/2} = 13.0742$$

Mesafe ölçümlerinde edilen en küçük mesafe değerleri 7.5089 ve 13.0742'dir. Bu durumda, H3 hastasının kan değerleri en çok H1 ve H5 hastalarına benzemektedir. Tablo 3.4'te görüldüğü üzere, bu iki değerlerin ortalaması eksik veri noktasının değeri olarak atanarak eksik veri tamamlanmış olur. KNN ile bütün veri setindeki eksik veriler iteratif olarak bu şekilde tamamlanır.

Tablo 3.4. KNN ile H3 Hastasının ALT Özelliğindeki Eksik Verinin Tamamlanması

	ALT	CRP	AST	WBC	LDH
H1	18	29	36	2.9	257
H2	22	.	25	3.5	207
H3	16	193.7	31	4.6	220
H4	33	318.7	96	16.5	651
H5	14	16.5	21	4.9	.

3.6.1.2. Özellik Ölçeklendirme

Veri setlerinde her bir özellik, girdi değişkeni olup farklı ölçeklere sahip olabilir. Girdi değişkenlerindeki bu ölçek farklılıkları, oluşturulacak MÖ modelinde çeşitli problemlere neden olabilmektedir. Özellik ölçeklendirmenin amacı, bütün özelliklerin yaklaşık olarak aynı ölçekte olduğundan emin olarak oluşabilecek problemleri minimuma indirmektir. Özellik ölçeklendirme sonucunda, her bir özellik eşit derecede önemli hale getirilerek MÖ algoritmaları tarafından işlenmesini kolay hale getirilmektedir. Veri ön işleme tekniklerinden biri olan standardizasyon verilerin ölçeklenmesini sağlamaktadır.

3.6.1.2.1. Standardizasyon

Standardizasyon, bir veri kümesinin standartlaştırılması, gözlenen değerlerin ortalamasının 0 ve standart sapmasının 1 olacak şekilde değer dağılımlarının yeniden ölçeklendirilmesidir. Standardizasyon, gözlemlerin uygun bir ortalama ve standart sapma ile bir Gaussian dağılımına (Karlis, 2002) uyduğunu yani her bir özellik için normal olarak dağıldığını varsayarak hesaplama yapan yaygın bir ölçekleme metodudur. Bu metot, Denklem 3.23'te görüldüğü gibi veri setindeki tüm verilerden ilgili sütunun ortalamasını çıkartıp yine ilgili sütunun standart sapmasına bölerek işlem yapar.

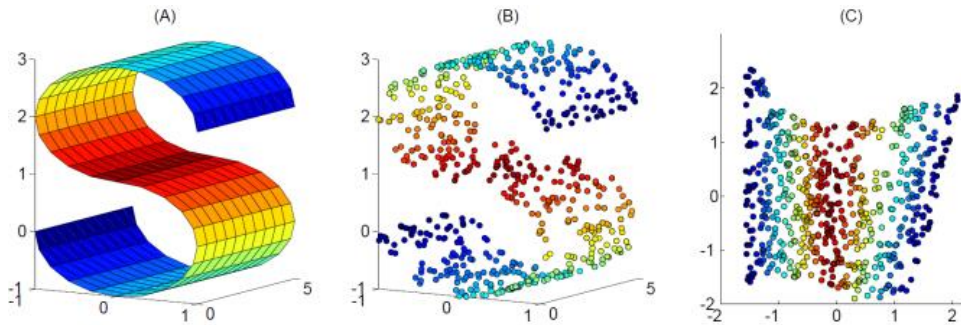
Böylelikle, veri setindeki tüm gözlem birimlerinin -1 ile 1 arasında değer almasını sağlanır.

$$X_{std} = \frac{X_i - \text{Ortalama}(X)}{\text{Standart Sapma}(X)} \quad (3.23)$$

3.6.1.3. Boyut Azaltma

Veri setlerinde bulunan örnek ve özellik sayısına veri setinin boyutu denmektedir. Boyut azaltma ise temel olarak ele alınan rastgele örnek veya özellik sayısını azaltma işlemidir. Bu işlem, büyük veri setlerindeki gereksiz verileri azaltarak hesaplama süresini indirgeyen ve eğitimi güçlendiren denetimsiz öğrenme yöntemidir. Veri setinin boyutu arttıkça eğitim setini görselleştirmek ve üzerinde işlem yapmak zor hale gelmektedir. Bundan dolayı, birçok boyut azaltma tekniği geliştirilmiştir. Bu teknikler, önemli bilgileri kaybetmeden veri setinin boyutunu azaltmayı amaçlar. Boyut azaltma tekniklerinin temel 2 bileşeni vardır. Bunlar; özellik seçme ve özellik çıkarmadır. Özellik seçme, orijinal veri setini temsil eden en önemli özellikleri bir alt küme olarak seçerek veri seti boyutunu azaltan bir yöntemdir. Özellik çıkarma ise orijinal veri setinden türetilmiş özellikler üreterek mevcut veri setini daha az boyutlu bir veri setine dönüştürme işlemidir.

Şekil 3.34'te, doğrusal olmayan üç boyutlu (3B) düzlemdeki verilere (A) doğrusal bir dönüşüm uygulanarak 2B düzleme aktarılmıştır (B). Şekil 3.34'te görüldüğü üzere, çok sayıda bulunan verileri bilgisayar sisteminde görüntüleme hem oldukça karmaşık görünmesine hem de bu karmaşıklıktan dolayı temsil ettikleri bilgilerin doğruluktan sapmasına neden olmaktadır. Verilerdeki önemli bilgileri kaybetmeden boyut azaltma tekniği uygulandığında (C) verilerin temsili daha doğru hale gelmektedir.



Şekil 3.34. Çok Boyutlu Veriler Üzerinde Boyut Azaltma (Engel ve ark., 2011)

3.7. 1B Verileri 2B Verilere Dönüştürmek için Kullanılan Algoritmalar

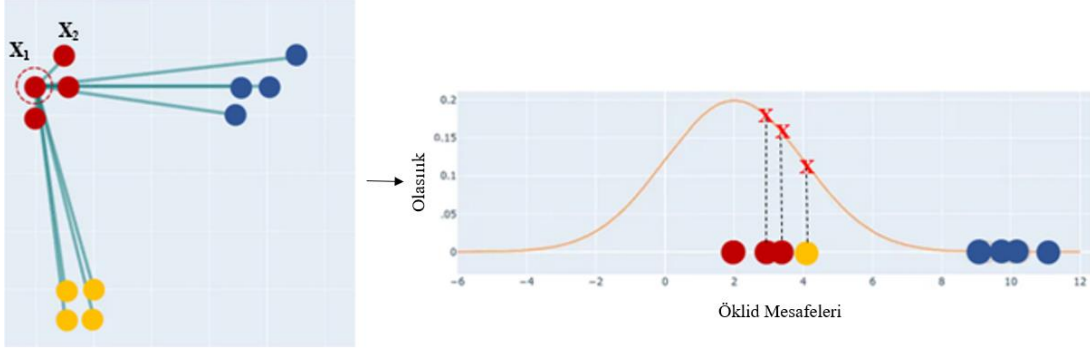
3.7.1. T-Dağıtılmış Stokastik Komşu Gömme Algoritması

T-SNE algoritması, Maaten & Hinton (2008) tarafından yüksek boyutlu veri kümelerinin görselleştirilmesi için geliştirilen, doğrusal olmayan boyut azaltma tekniğidir. T-SNE algoritmasının amacı, çok boyutlu veriler arasındaki uzaklıkları olabildiğince koruyarak insan gözlemine uygun düşük boyutlu temsillerini bulmaktır. Bu temsilleri bulurken kartezyen düzlemde benzer olan noktaları birbirine yakın tutarken benzer olmayanları ise birbirinden uzak tutmaktadır.

T-SNE algoritması, veriler arası benzerlik ölçüsünü hesaplar. Bu hesap, yüksek boyutlu uzaydaki noktaların benzerlik olasılığını, düşük boyutlu uzayda bu noktalara karşılık gelen benzerlik olasılığına dönüştürme işlemidir. T-SNE algoritması, bu dönüştürme için Öklid, Normal dağılım ve T-dağılımı yöntemlerini kullanır. Yüksek boyutlu uzayda veri noktaları arasındaki öklid mesafelerini, benzerlikleri temsil eden koşullu olasılıklara dönüştürür. Örneğin, yüksek boyutlu veri noktası X_j 'nin X_i 'ye benzerliği, P_{ji} koşullu olasılığıdır. P_{ji} , X_i merkezli normal dağılım altında olasılık yoğunluklarıyla orantılı olarak seçilirse, X_i 'nin komşusu olarak X_j 'yi seçme koşullu olasılığı veya X_j 'nin X_i 'ye benzerliği olarak yorumlanabilir. Yakın noktalar için P_{ji} yüksek olurken, uzak veri noktalarında P_{ji} küçük değere sahip olacaktır (Maaten & Hinton, 2008). T-SNE algoritması, 3 temel adımda çalışır:

1) Yüksek Boyutlu Uzayda Normal Dağılım ile Benzerlik Ölçümü

T-SNE yüksek boyutlu uzayda bulunan verilerin benzerlik ölçümünü, noktalar arası koşullu olasılıkların hesaplanması ile gerçekleştirir ve buna noktalar arası öklid mesafelerini, noktaların benzerlik olasılığına çevirerek başlar. Şekil 3.35'teki gibi, her bir veri noktası normal dağılım merkezinde merkezlenir ve kalan noktalar bu merkezlenen veri noktasına olan öklid mesafesine göre konumlandırılır. Ardından her bir veri noktasının diğer veri noktaları ile koşullu olasılığı Denklem 3.24 ile ölçülür ve normalize edilir. Bu şekilde, normal dağılım altında olasılık yoğunluğu ölçülen ve normalize edilen veri noktalarının her biri, bir olasılık değerine sahip olur. Bu olasılıklar, verilerin benzerlikleriyle orantılıdır. Yani, eğer X_i ve X_j veri noktaları bu normal dağılım altında eşit değerlere sahipse bu oranlar benzerlikleri ile aynıdır (Maaten & Hinton, 2008).



Şekil 3.35. Veriler Arası Öklid Mesafelerinin Benzerlik Olasılığına Çevrilmesi

$$P_{ji} = \frac{\exp(-\|X_i - X_j\|^2 / 2\sigma_i^2)}{\sum_{k \neq i} \exp(-\|X_i - X_k\|^2 / 2\sigma_i^2)} \quad (3.24)$$

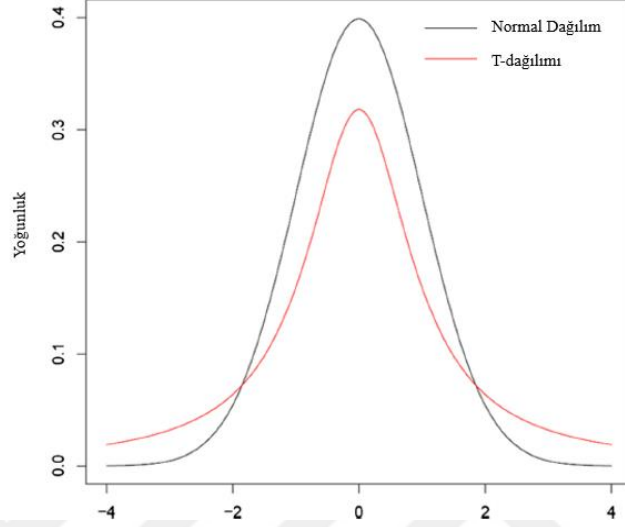
T-SNE'nin sahip olduğu *karışıklık* (*perplexity*) parametresi, bu metodun temel parametresidir ve Denklem 3.24 ile verilen formülde σ (varyans) değerini temsil etmektedir. Bu parametre, normal dağılım merkezinde merkezlenen her bir veri noktasının çevresinde çevrelenen en yakın komşu sayısı hakkında tahmini bir sayıdır. Bu parametre ile belirlenen tahmini komşu sayısına göre uygun σ değeri ayarlanmaktadır. Bu metodun bulucuları olan Maaten ve Hinton (2008), T-SNE'nin performansının bu parametreye doğrudan bağlı olduğunu ileri sürerek ondan en iyi şekilde yararlanmak için 5-50 arasında farklı karışıklık değerinin analiz edilmesini önermektedir. Denklem 3.24 ile verilen formülde k parametresi, veri setindeki veri noktaları üzerindeki endekstir.

2) Düşük Boyutlu Uzayda T-dağılımı ile Benzerlik Ölçümü

Bu adım ilk adıma benzerdir fakat benzerlik ölçümü hesabında normal dağılımı kullanmak yerine anakütlenin normal dağılım gösterdiğini varsayan T-dağılımını kullanır. Bu dağılım sayesinde T-SNE, Denklem 3.25 ile yüksek boyutlu uzaydaki noktalara karşılık gelen düşük boyutlu uzayda ikinci bir koşullu benzerlik olasılık kümesi elde eder. Denklem 3.25 ile verilen formülde k ve l parametreleri, veri setindeki veri noktaları üzerindeki endekslerdir. Veri setinde bulunan N veri noktası için endeksler 1'den N'ye kadar uzanır ve k ve l tüm veri noktalarının toplamıdır.

$$Q_{ji} = \frac{(1 + \|Y_j - Y_i\|^2)^{-1}}{\sum_{k \neq l} \exp(1 + \|Y_k - Y_l\|^2)^{-1}} \quad (3.25)$$

Şekil 3.36'da görüldüğü üzere, T-dağılımı normal dağılımdan daha uzun kuyruklara sahiptir. Bu durum, normal dağılımın sahip olduğu kısa kuyruklardan dolayı ortaya çıkan *Kalabalıklaşma (Crowding) Problemine* çözüm niteliğindedir ve daha uzak mesafedeki verileri noktalarının daha iyi modellenmesine izin vermektedir.



Şekil 3.36. Normal ve T-dağılımının Karşılaştırılması (Teramoto ve ark., 2014)

3) Benzerlik Ölçümleri Arasındaki Farkın Minimize Edilmesi

Düşük boyutlu uzayda veri noktalarının mükemmel bir temsili için, yüksek ve düşük boyutlardaki benzerlik olasılıkları arasındaki fark en aza indirgenmeye çalışılır. Bu indirgeme işlemine maliyet hesabı denir. Maliyet hesabı olan olasılık dağılımları arasındaki fark, Denklem 3.26'te verilen Kullback Liebler (KL) uzaklığı kullanılarak ölçülür.

$$C = D_{KL}(P \parallel Q) = \sum_j \sum_i P_{ji} \log \frac{P_{ji}}{Q_{ji}} \quad (3.26)$$

KL, bir tür istatistiksel mesafedir ve P olasılık dağılımının Q olasılık dağılımından ne kadar farklı olduğunun bir ölçüsüdür. KL, literatürde D_{KL} şeklinde ifade edilir ve $D_{KL}(P \parallel Q)$ ile gösterilir. KL, P_{ji} ve Q_{ji} değerlerini etkili bir şekilde karşılaştırarak olasılık dağılımlarının uyumsuzluğunu hesaplar. Ardından, Denklem 3.26'nın türemesine aracı olan Denklem 3.27 kullanılarak, düşük boyutlu uzaydaki her bir veri noktasına göre KL'nin gradyanı hesaplanarak veri noktalarının yönlerinde

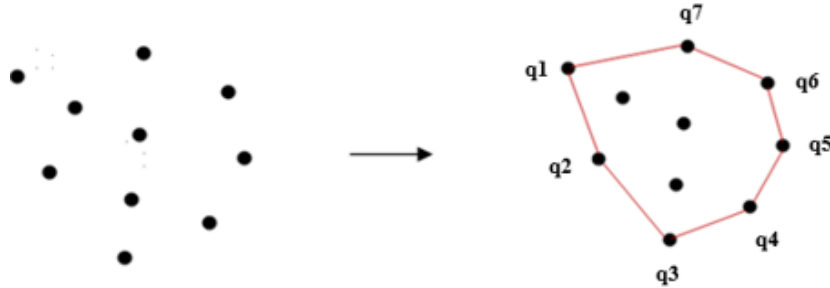
güncellemeler yapılır. Bu güncellemeler, KL değerini düşürecek yönde gerçekleştirilir ve bu işlemler en düşük KL değeri elde edilene kadar yinelemeli olarak tekrarlanır. Bu şekilde, T-SNE çok boyutlu verileri en az düzeyde bilgi kaybı ile daha düşük boyutlu bir uzaya eşler ve veri noktalarının benzerliğine dayalı kümeler tanımlayarak verilerdeki örüntüleri bulmaya çalışır.

$$\frac{\delta C}{\delta Y_j} = 4 \sum_{i \neq j} (P_{ji} - Q_{ji}) (1 + \|Y_j - Y_i\|^2)^{-1} (Y_j - Y_i) \quad (3.27)$$

3.7.2. Dışbükey Gövde Algoritması ve Minimum Sınırlayıcı Dikdörtgen

Bilgisayarlı geometri alanının en temel problemlerinden biri olan dışbükey gövde problemi, kartezyen düzlemde bütün noktaları içeren ve en küçük alanı kaplayan dışbükey çokgeni oluşturma problemidir. Bu problem için geliştirilen algoritma ise DGA'dır. Şekil 3.37'de görüldüğü üzere, DGA kartezyen düzlemdeki bütün noktaları kapsayan en küçük dışbükey çokgeni elde eder.

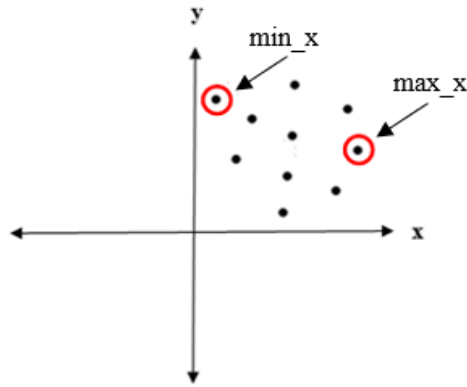
DGA, sıralama algoritmaları mantığında çalışır ve çalışma prensibi kısaca şu şekildedir: Kartezyen düzlemde n adet veri noktasından oluşan bir P veri kümesi için, bu n noktanın kartezyen koordinatları ile dışbükey gövde saat yönünün tersine sırayla veri noktalarının dairesel bağlantılı bir köşe listesi olarak temsil edilir. Yani P veri kümesinde, i'nci nokta dışbükey gövdenin bir veri noktasıysa, sonraki[i] saat yönünün tersine sonraki veri noktasının indeksidir ve önceki[i] ise bir sonraki veri noktasının saat yönünde indeksidir. Listenin başı olarak hangi köşenin tutulacağı önemli değildir ve bağlantı yönünün saat yönü veya saat yönünün tersine olma kararı ise isteğe göredir. Şekil 3.37 üzerinden incelenecek olursa, q1 veri noktası dışbükey gövdenin köşe listesinin ilk elemanıdır, sonraki[q1] = q2 ve önceki[q1] = q7'dir.



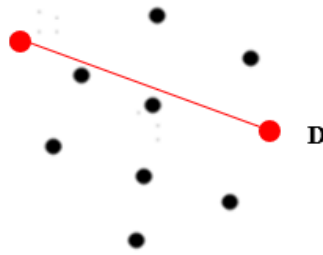
Şekil 3.37. Dışbükey Gövde Algoritması

DGA için çeşitli yöntemler geliştirilmiştir. Geliştirilen yöntemlerden en popüler olan QuickHull (Barber ve ark., 1996), sıralama algoritması olan QuickSort'a benzeyen bir Böl ve Yönet algoritmasıdır ve ortalama zaman karmaşıklığı $O(n \log(n))$ 'dir. n adet veri noktasına sahip bir veri kümesinde QuickHull, temel olarak 4 adımda çalışır:

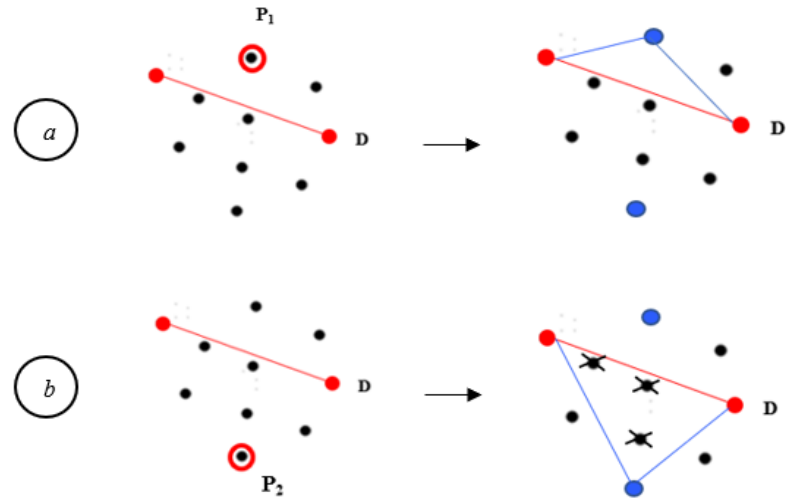
1. Minimum x koordinatına sahip nokta (min_x) ve maksimum x koordinatına sahip nokta (max_x) bulunur.



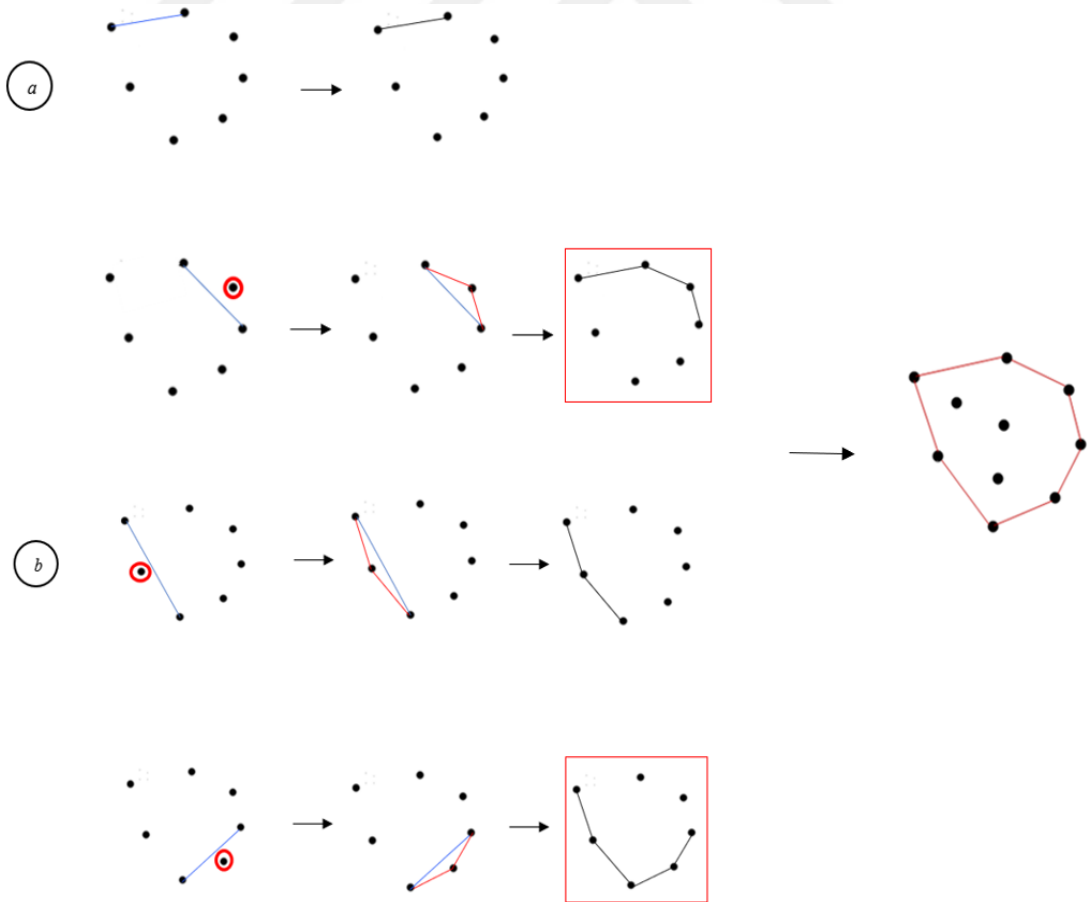
2. min_x ve max_x noktalarını birleştiren bir D doğrusu çizilir ve bu doğru, veri setini iki parçaya böler.



3. Her bir parça için, D doğrusundan maksimum uzaklığı olan P noktaları bulunur ve bu P noktaları, min_x , max_x noktalarıyla bir üçgen oluşturur. Dolayısıyla, oluşan bu üçgen içinde kalan veri noktaları, dışbükey gövdenin köşeleri olamaz.



4. P ve min_x noktalarını birleştiren doğru ile P ve max_x noktalarını birleştiren doğru yeni doğrulardır ve üçgenin dışında kalan noktalar ise güncel noktalar kümesidir. Bu şekilde problem, iki alt probleme bölünmüştür ve iteratif şekilde hiçbir nokta kalmayana kadar 2. Ve 3. adımlar tekrarlanarak noktalar dışbükey gövdeye eklenir.

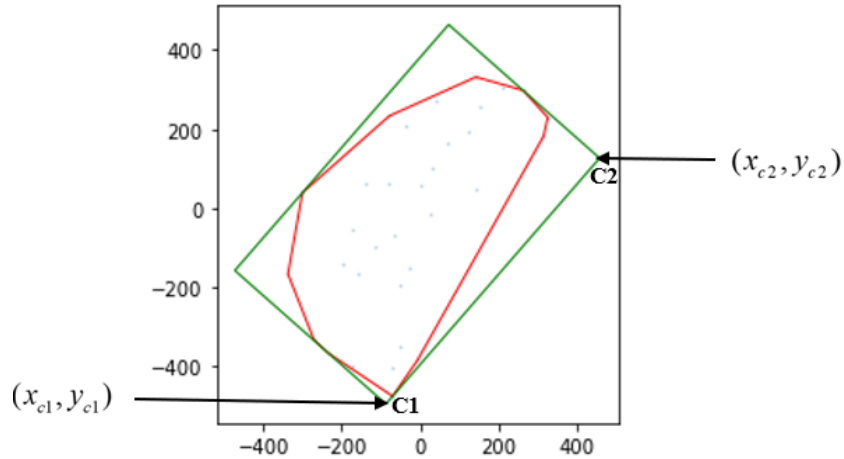


MSD, belirli bir veri kümesinin x ve y koordinatlarının minimum ve maksimum değerleri kullanılarak bir dikdörtgen ile sınırlandırılmasıdır. MSD, veri noktalarının en küçük sınırlayıcı kutusunun iki boyutlu bir halidir ve geometrik olarak bir veya birden fazla nesneyi sınırlayan en küçük dikdörtgendir. Bu dikdörtgen için ilk olarak DGA ile koordinat sisteminde elde edilen köşe listelerinden dışbükey gövdenin kenar listesi elde edilir. Elde edilen dışbükey kenarlarının her biri için trigonometrik açıları hesaplanır. Ardından, kenarların sahip olduğu açı değerleri kullanılarak MSD elde edilir.

3.7.3. Gradyan Temelli Rotasyon

Gradyan temelli rotasyon, bir görüntünün sınıflandırılabilmesi için yatay ve dikey piksel çerçevelerine sahip olması gerektiği zaman uygulanan bir yöntemdir. Bu yöntemde ilk olarak, kartezyen düzlemde bulunan ve döndürülecek olan veri kümesinin gradyan hesabı yapılır. Gradyan hesabı, dönüş açısı olan $\theta = \tan^{-1}(G_r)$ değerinin hesaplanmasında kullanılır ve bu açı değeri, R döndürme matrisinin elde edilmesini sağlar.

Şekil 3.38’de görüldüğü gibi, bir G veri kümesinin rotasyon işleminde, iki köşe koordinatları (x_{c1}, y_{c1}) ve (x_{c2}, y_{c2}) olarak tanımlanırsa gradyan G_r , Denklem 3.28’deki gibidir. R döndürme matrisi, yatay ve dikey görüntü çerçevesi sağlamak için veri kümesi ile çarpılır ve çarpma sonucunda çerçevenin A_c yatay ve B_c dikey uzunlukları elde edilir. Döndürme matrisi ve görüntünün yatay-dikey uzunlukları sırasıyla Denklem 3.29 ve Denklem 3.30’da verilmiştir.



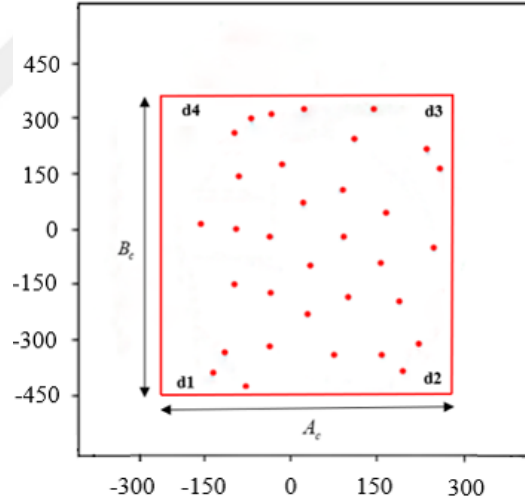
Şekil 3.38. Gradyan Temelli Rotasyon İşlemi

$$G_r = \frac{y_{c2} - y_{c1}}{x_{c2} - x_{c1}} \quad (3.28)$$

$$R = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \quad (3.29)$$

$$\begin{aligned} A_c &= |x_{d2} - x_{d1}| \\ B_c &= |y_{d3} - y_{d2}| \end{aligned} \quad (3.30)$$

Şekil 3.39’da görüldüğü üzere, burada döndürülmüş dikdörtgenin köşeleri d1, d2, d3, d4 olarak işaretlenmiştir. Dolayısıyla x_{d2} ve x_{d1} görüntünün yatay koordinat değerlerini temsil ederken y_{d3} ve y_{d2} ise dikey koordinat değerlerini temsil etmektedir. Bu noktalar A_c genişliğini ve B_c yüksekliğini hesaplamak için kullanılır.



Şekil 3.39. DGA ile Çerçvelenen Özellik Vektörünün Rotasyonu

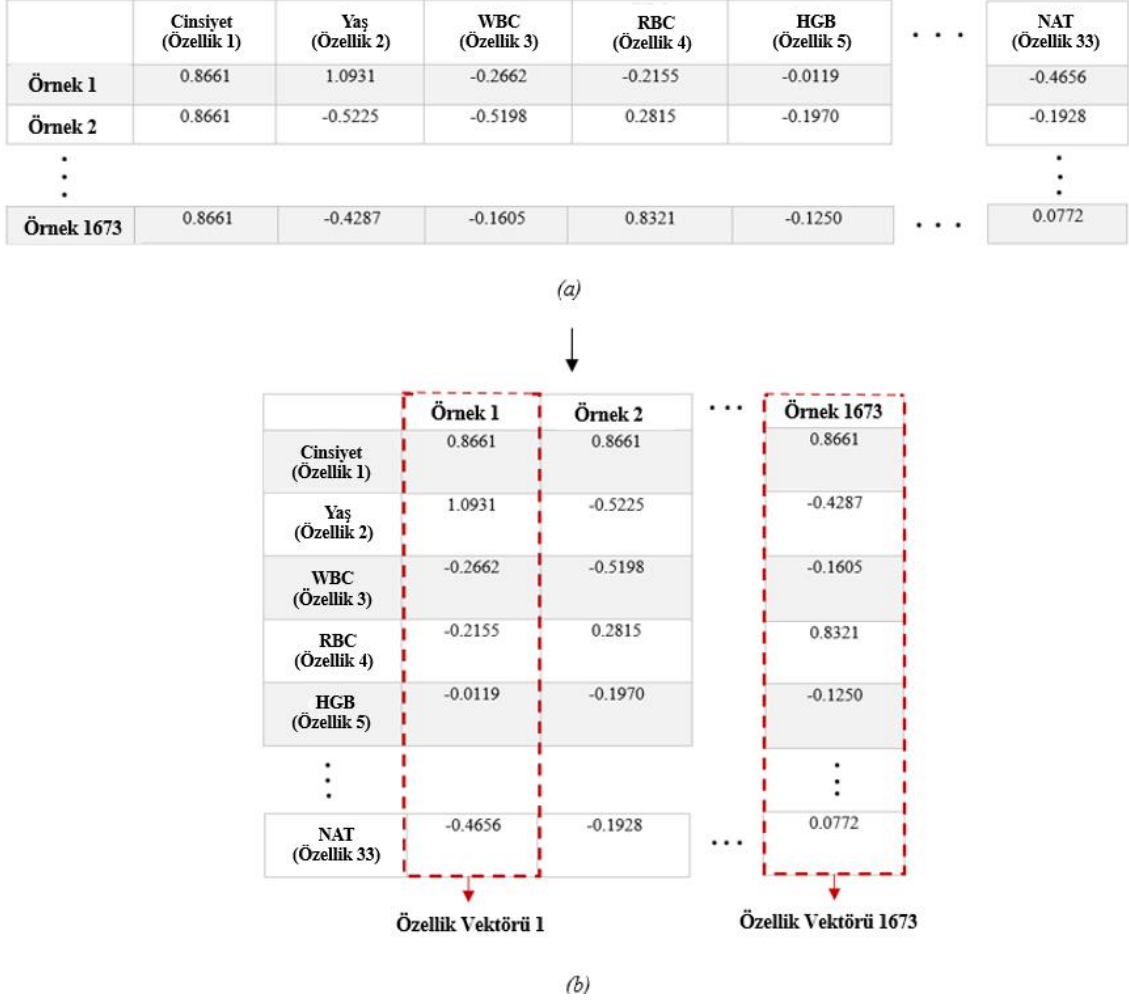
4. ÖNERİLEN MODEL VE BULGULAR

Bu tez çalışmasında, COVID-19 pozitif ve negatif bireylerin kan parametrelerinden görüntü verileri elde ederek ESA mimarisi ile COVID-19 tespiti modeli geliştirilmiştir. Buradaki temel amaç, maliyeti düşük ve ulaşılabilirliği hızlı olan kan parametreleri ile görüntü verilerinde yüksek başarıya sahip ESA mimarisinin kombinasyonu sağlanarak COVID-19 tespiti için etkin ve başarılı bir model sunmaktır. Bölüm 2’de anlatıldığı gibi, COVID-19 tespiti için birçok YZ modeli önerilmiştir. Buna ek olarak, literatürde 2B olmayan verilerin ESA ile etkin bir şekilde çalışabilirliği için çeşitli yöntemler geliştirilmiştir (Kovalerchuk & Dovhalets, 2017; Sharma ve ark., 2019; Buturovic & Miljkovic, 2020; Kovalerchuk ve ark., 2020). Günümüzde, hem daha yüksek doğrulukta COVID-19 tespiti yapabilmek için hem de bilgi kaybı yaşanmadan en doğru şekilde 1B verileri 2B verilere dönüştürmek için araştırmalar devam etmektedir. Bu tez çalışmasında, COVID-19 kan parametrelerini içeren veri setine çeşitli veri ön işlemler uygulanmış ve ardından DeepInsight (Sharma ve ark., 2019) metodu ile görüntü verilerine dönüştürülmüştür. Bu görüntü verilerini sınıflandırmak için 3 hatlı bir ESA modeli tasarlanmıştır.

Bu bölümde, tez çalışmasında kullanılan veri setine uygulanan veri ön işlem yöntemleri anlatılmıştır. Ardından, bu tez çalışmasında gerçekleştirilen 1B COVID-19 kan parametre verilerinin 2B görüntü verilerine dönüştürme adımları sunulmuştur. Son olarak önerilen 3 hatlı ESA modeli ve bu modelin başarısı, popüler ESA modelleri, çeşitli MÖ modelleri ve bu tez çalışmasında kullanılan veri seti ile gerçekleştirilmiş literatürdeki diğer COVID-19 çalışmaları ile karşılaştırılmalı olarak incelenmiştir.

4.1. Veri Ön İşleme

Bu tez çalışmasında, veri setinde bulunan 1736 örnekten %75’den fazla eksik veri içerenler silinmiştir (Cabitza ve ark., 2020). Kalan 1673 örneğin eksik verileri, k parametresi 5 seçilerek KNN algoritması ile tamamlanmıştır (Erol ve ark., 2022). [-1 1] aralığında standardizasyon işlemi gerçekleştirilmiştir ve Şekil 4.1’de görüldüğü gibi, veri setinin transpozu alınarak özellik vektörleri elde edilmiştir.

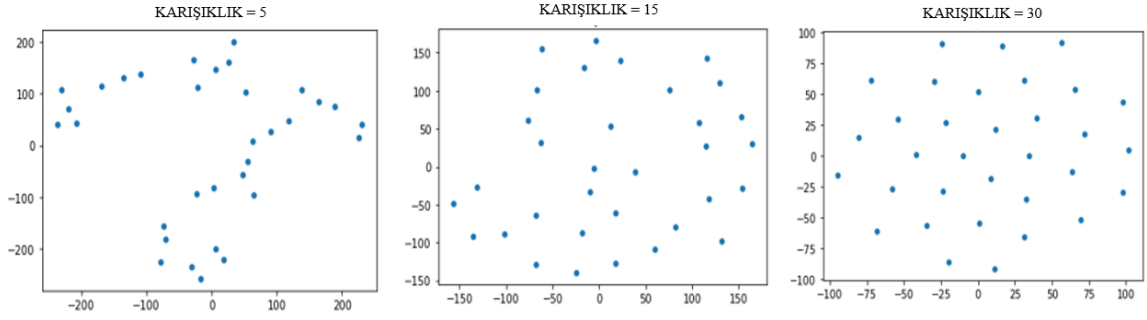


Şekil 4.1. (a) Veri Seti (b) Veri Setinin Transpoze Edilmesiyle Elde Edilen Özellik Vektörleri

4.2. 1B Verileri 2B Verilere Dönüştürme

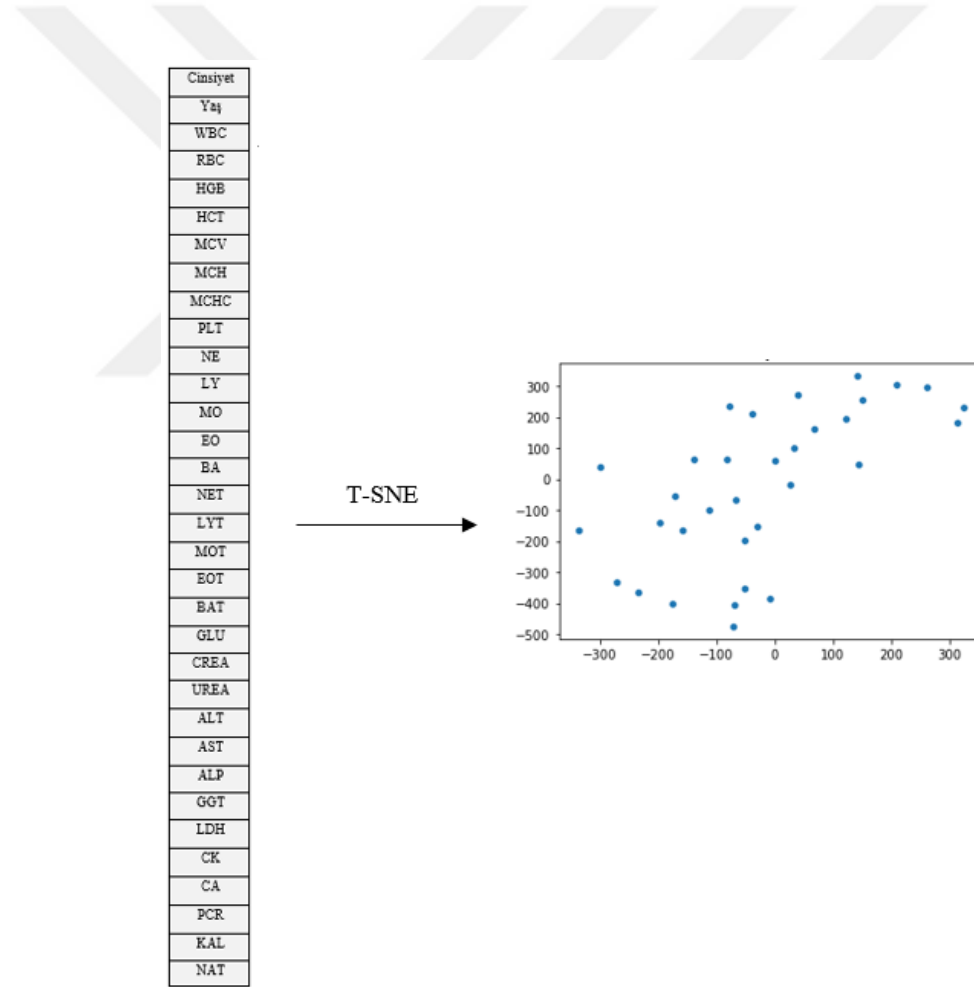
4.2.1. Özellik Vektörünün 2B Kartezyen Düzleme Aktarılması

Veri setinin görüntü verilerine dönüştürülmesi için ilk olarak veri setindeki özellik vektörleri 2B kartezyen düzleme aktarılmıştır. Tez kapsamında bu işlem, T-SNE algoritması ile benzer özellikleri birbirine yakın benzer olmayan özellikleri birbirinden uzak olacak şekilde 2B kartezyen düzlemde temsilleri bulunarak gerçekleştirilmiştir. T-SNE'nin temel parametresi olan karışıklık parametresi, her bir özellik vektörünün 2B kartezyen düzlemde sahip olduğu tahmini yakın komşu sayısını temsil etmektedir. Bu metodun bulucuları olan Maaten ve Hinton (2008), T-SNE'nin performansının bu parametreye doğrudan bağlı olduğunu ve 5-50 arasında farklı karışıklık değerlerinin analiz edilmesini önerdiğinden, bu tez çalışmasında T-SNE metodu için 5-50 arasında optimum karışıklık parametre değeri analizi yapılmıştır. Şekil 4.2'de, karışıklık değeri 5, 15 ve 30 için özellik vektörlerinin 2B düzlemde dağılımı sunulmuştur.



Şekil 4.2. T-SNE Karışıklık Parametresi 5,15 ve 30 için Özellik Vektörlerinin 2B Düzlemde Dağılımı

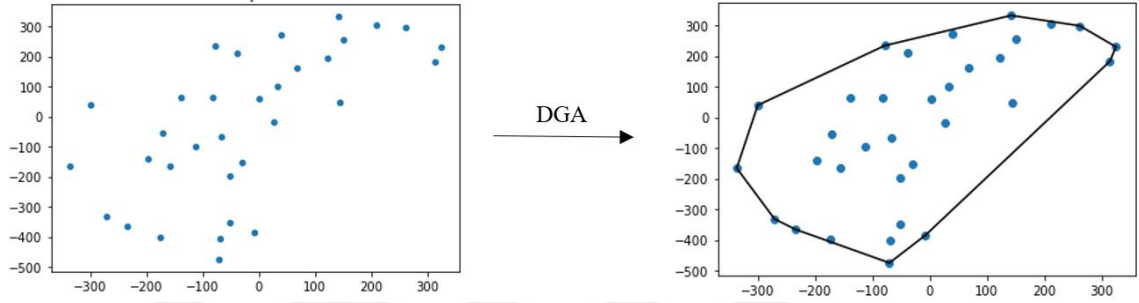
Bu tez çalışmasında, karışıklık parametresi 8 seçilerek T-SNE algoritması uygulanmıştır. Kan değerlerini içeren 33 özelliğe sahip bir özellik vektörü, T-SNE algoritması ile 2B kartezyen düzleme Şekil 4.3'teki gibi aktarılmıştır.



Şekil 4.3. Özellik Vektörlerinin T-SNE ile 2B Kartezyen Düzleme Aktarılması

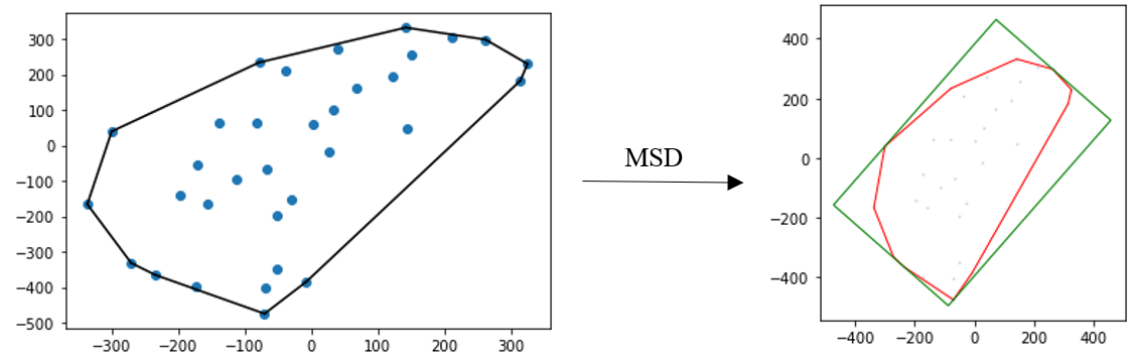
4.2.2. Özellik Vektörünün Çerçevesi

Kartezyen düzlemde bütün noktaları içeren ve en küçük alanı kaplayan dışbükey çokgeni oluşturan DGA için çeşitli yöntemler geliştirilmiştir. Bu tez çalışmasında, dışbükey çokgeni oluşturmak için kullanılan yöntem QuickHull'dur (Barber ve ark., 1996). QuickHull, T-SNE ile 2B düzleme aktarılan özellik vektörlerinin kartezyen düzlemde dış bükeyini elde eder. Bu tez kapsamında, T-SNE algoritması ile kartezyen düzleme aktarılan özellik vektörlerine uygulanan DGA Şekil 4.4'te sunulmuştur.



Şekil 4.4. T-SNE ile 2B Düzleme Aktarılan Özellik Vektörlerinin DGA Algoritması ile Dış Bükeyini Elde Etme

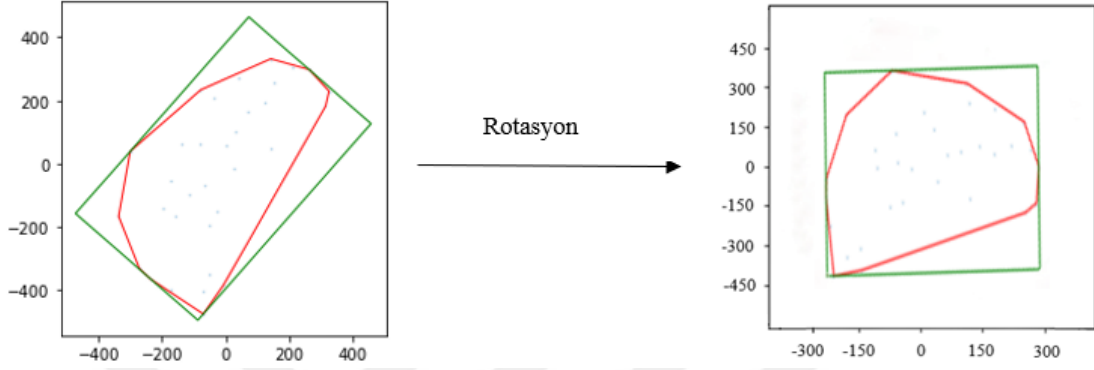
T-SNE ile 2B kartezyen düzleme aktarılan özellik vektörlerinden görüntü verileri elde etmek için bütün özellik noktalarını içeren bir çerçeveleme işlemi gerekmektedir. Şekil 4.5'te sunulduğu üzere, çalışmada özellik vektöründen DGA aracılığıyla elde edilen dışbükey ile MSD elde edilerek çerçeveleme işlemi yapılmıştır.



Şekil 4.5. DGA Algoritması ile Dış Bükeyi Elde Edilmiş Özellik Vektörünün MSD ile Çerçevesi

4.2.3. Çerçeveselenen Özellik Vektörünün Döndürülmesi: Rotasyon

ESA mimarisi için görüntünün yatay ve dikey olarak çerçevelere sahip olması gerektiğinden bu tez çalışmasında özellik vektörlerine Şekil 4.6'da sunulan gradyan temelli rotasyon işlemi uygulanır. Rotasyon işlemi için, DGA ile elde edilen dikdörtgenin iki köşe koordinatının gradyanı dikkate alınır.



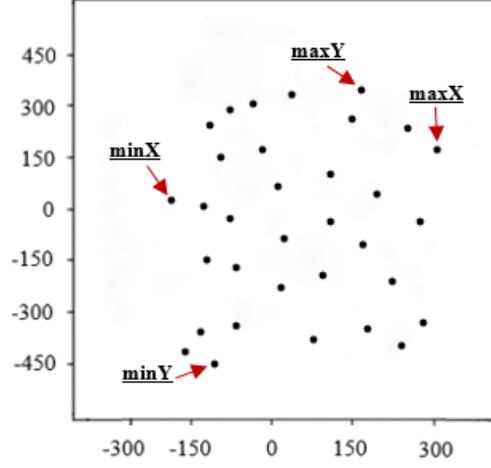
Şekil 4.6. MSD Algoritması ile Çerçeveselenen Özellik Vektörünün Rotasyonu

4.2.4. Kartezyen Koordinatların Piksellere Dönüştürülmesi: Pikel Haritalama

2B kartezyen koordinatına aktarılan özellik vektörü, çerçeveleme ve ardından döndürme işlemleri sonrasında ESA tarafından işlenebilmesi için kartezyen koordinat değerlerinin piksel formlarına dönüştürülmesi gerekmektedir. Bu dönüştürme işlemine, piksel haritalama denir. Pikel haritalama, özellik vektörlerinin normalizasyonu ile koordinat hesabı ve giriş matrisleri ile görüntü verilerinin elde edilmesi olmak üzere iki adımda gerçekleştirilir.

4.2.4.1. Özellik Vektörlerinin Normalizasyonu ve Koordinat Hesabı

ESA'nın verileri işleyebilmesi için değerlerin koordinat sistemi değerlerinde değil, piksel değerleri formlarında olması gerekmektedir. Bu tez kapsamında, özellik vektörüne uygulanan T-SNE, DGA, MSD ve rotasyon işlemleri sonucunda 2B düzlemdeki özellik vektörünün her bir özelliğinin bulunduğu x ve y koordinat değerleri için Şekil 4.7'deki gibi normalizasyon işlemine tabi tutulmuştur.



Şekil 4.7. 2B Düzlemdeki Özellik Vektörünün minX, maxX ve minY, maxY Değerlerini Bulma

Bu tez çalışmasında, özellik vektörlerine min-max normalizasyonu uygulanmıştır ve bu normalizasyonun formülü Denklem 4.1’de verilmiştir.

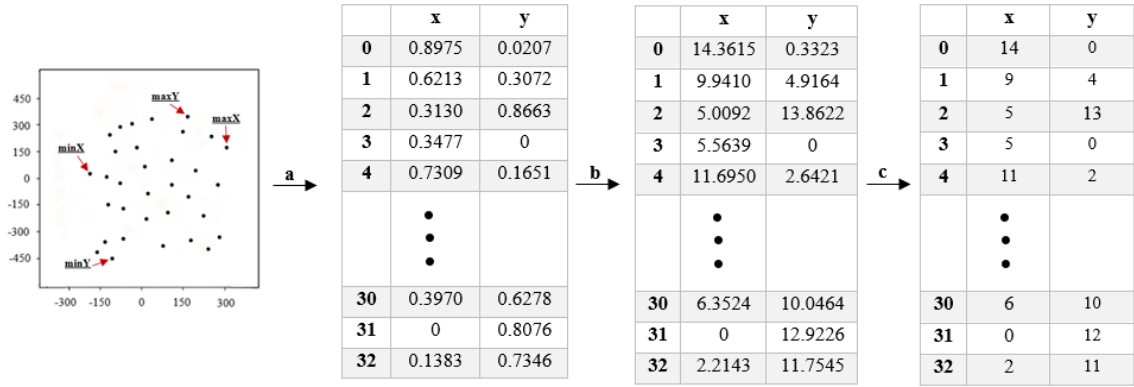
$$A_{norm} = \frac{A_i - A_{min}}{A_{max} - A_{min}} \quad (4.1)$$

Gerçekleştirilen normalizasyon işleminin çalışma prensibi şu şekildedir: Rotasyon işlemi sonucunda, Şekil 4.7’deki gibi özelliklerin hem x hem de y koordinat değerlerinden minX, maxX ve minY, maxY değerleri bulunur. Bu değerler normalizasyon işleminde kullanılacak olan, x ve y koordinatları için ayrı ayrı min-max değerleridir. Bu değerler ile Denklem 4.2 ve 4.3 kullanılarak özellik vektörünün x ve y koordinat değerleri için normalizasyon işlemi gerçekleştirilir.

$$X_{norm} = \frac{X_i - X_{minX}}{X_{maxX} - X_{minX}} \quad (4.2)$$

$$Y_{norm} = \frac{Y_i - Y_{minY}}{Y_{maxY} - Y_{minY}} \quad (4.3)$$

Normalizasyon işleminin ardından her bir özellik vektörünün koordinat hesabı yapılmıştır. Şekil 4.8’de gerçekleştirilen normalizasyon (a) ve koordinat hesabı işlemlerinin şematik yapısı 16x16 boyutlu görüntü için sunulmuştur.



Şekil 4.8. 2B Düzlemdeki Özellik Vektörünün Normalizasyonu ve Koordinat Hesabı

Şekil 4.8’de görüldüğü üzere, koordinat hesabı şu şekildedir: Normalize edilmiş x ve y koordinat değerleri, belirlenen piksel değeri ile çarpılarak özelliklerin koordinat değerlerinin, piksel domainine uygun aralıkta değerler alması sağlanır (b). Son olarak, elde edilen virgüllü koordinat değerlerinin tam kısmı piksel değeri olarak alınır (c) ve maksimum piksel değerinde olanlar piksel sınırını aşmaması için bir alt değere düşürülerek her bir özellik vektörünün x ve y koordinatlarına karşılık, $piksel_x$ ve $piksel_y$ değerleri elde edilir.

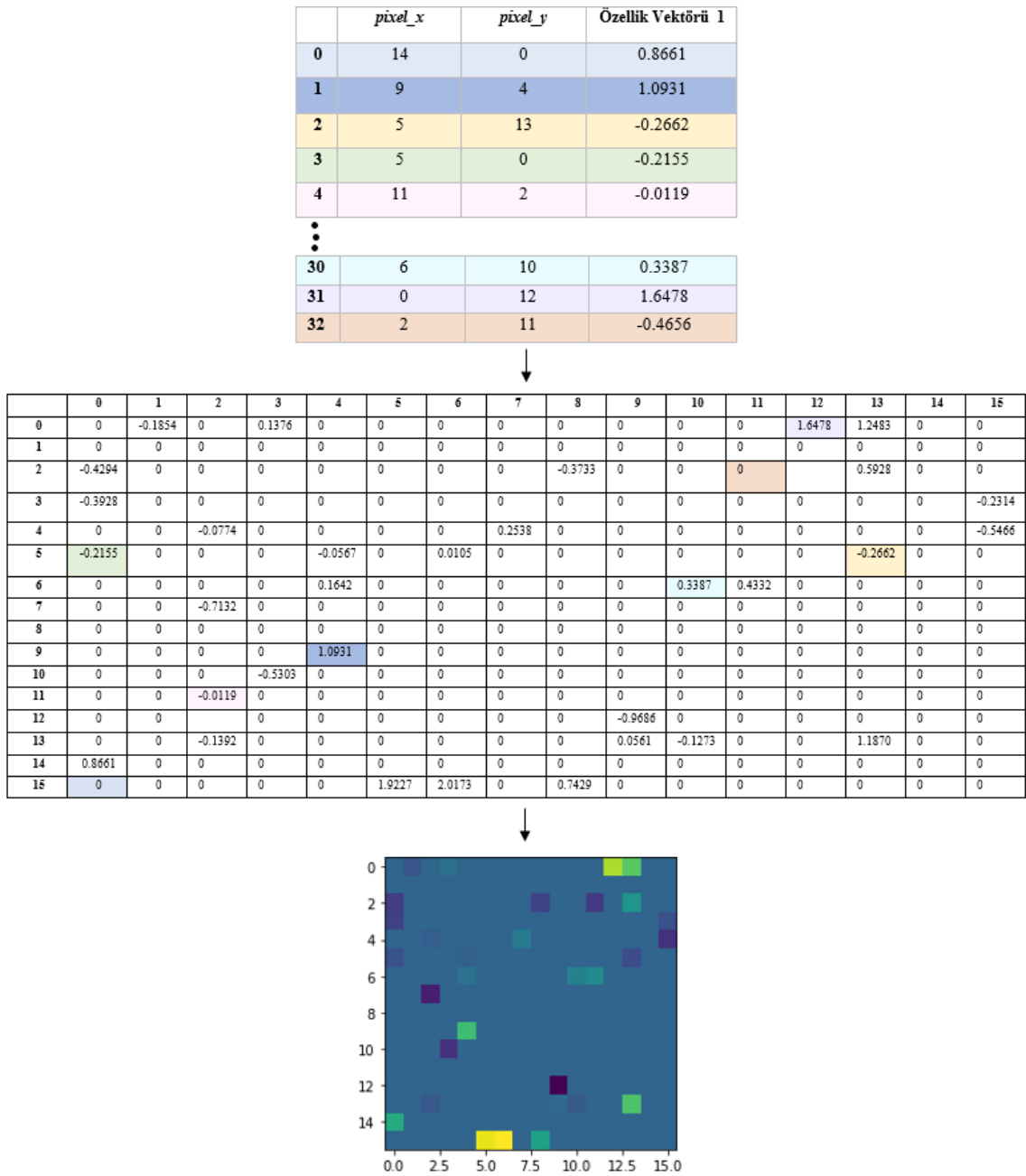
4.2.4.2. Giriş Matrisleri Kullanılarak Görüntü Matrislerinin Elde Edilmesi

Koordinat hesabı ile hesaplanan özellik vektörlerinin $piksel_x$ ve $piksel_y$ değerleri ve veri setinde bulunan özellik vektörleri ile giriş matrisleri oluşturulur. Oluşturulan giriş matrislerindeki piksel noktalarına, özellik vektörlerinin değerleri iteratif bir şekilde yerleştirilerek görüntü matrisleri elde edilir. Bu tez çalışmasında, 1673 adet görüntü matrisi elde edilmiştir. Şekil 4.9’da, 16x16 boyutlu görüntü için özellik vektörü 1 ile oluşturulan giriş matrisi verilmiştir.

	$piksel_x$	$piksel_y$	Özellik Vektörü 1
0	14	0	0.8661
1	9	4	1.0931
2	5	13	-0.2662
3	5	0	-0.2155
4	11	2	-0.0119
	•		
	•		
	•		
30	6	10	0.3387
31	0	12	1.6478
32	2	11	-0.4656

Şekil 4.9. Giriş Matrisi

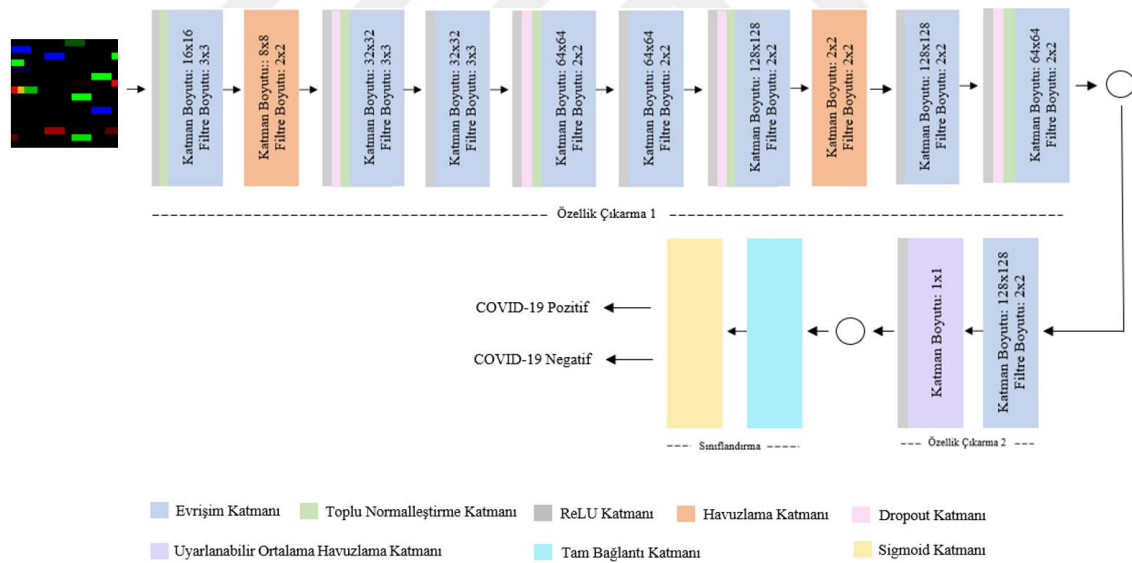
Özellik vektörlerinin giriş matrisindeki, $pixel_x$ ve $pixel_y$ noktalarına özellik vektörünün değerleri yerleştirilerek NxN boyutlu görüntü matrisleri elde edilir. N görüntü boyutu için seçilen piksel değeridir. Elde edilen görüntü matrisleri, RGB formata dönüştürülerek görüntü verileri elde edilir. Bu işlem sonucunda, özellik vektörlerinin her bir özellik noktasındaki pikseller 0-255 aralığında parlaklık değeri olarak görüntü verileri elde edilmiş olur. Şekil 4.10'da bu işlemler, 16x16 boyutlu görüntü için özellik vektörü 1 ile şematik olarak sunulmuştur.



Şekil 4.10. Piksel Haritalama ile Görüntü Verilerinin Elde Edilmesi

4.3. Önerilen Model

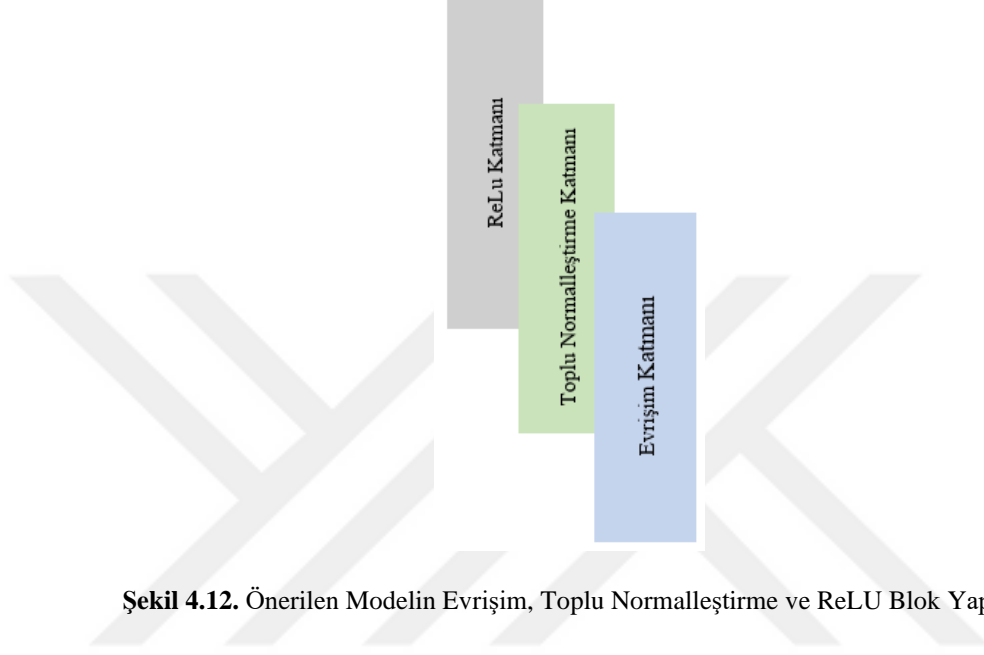
ESA, 2B verilerden çeşitli özellikler çıkarır (Albawi & Mohammed, 2017). Bu tez çalışmasında, veri ön işlemlerden geçirilmiş 1B COVID-19 kan parametrelerine sırası ile T-SNE, DGA, MSD ve piksel haritalama uygulanması sonucunda 1673 adet görüntü matrisi oluşturulmuştur. Elde edilen görüntü matrisleri, RGB formata dönüştürülerek 1673 adet görüntü verisi elde edilmiştir. Elde edilen bu görüntü verilerini sınıflandırmak için 3 hatlı bir ESA mimarisi geliştirilmiştir. 3 hatlı ESA mimarisi Şekil 4.11’de sunulmuştur. Geliştirilen 3 hatlı ESA mimarisinin ilk hattı, birden fazla evrişim katmanı sayesinde görüntülerden yoğun bir şekilde özellik çıkarımı gerçekleştirirken ikinci hattı, ilk hattan gelen özellikleri çıkarılmış verileri çeşitli katmanlar aracılığı ile daha kararlı hale getirerek veri setinin nihai özelliklerini elde eder. Son olarak, üçüncü hatta bulunan tam bağlantı ve sigmoid katmanları sayesinde sınıflandırma işlemi gerçekleştirilir. Model, PyTorch (Paszke ve ark., 2017) kullanılarak geliştirilmiştir.



Şekil 4.11. Önerilen 3 Hatlı ESA Mimarisi

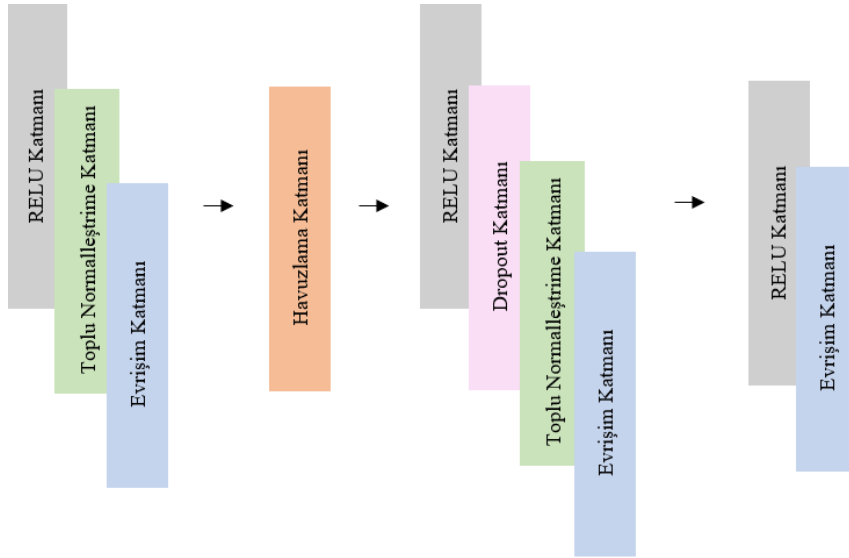
Önerilen modelde kullanılan ve Şekil 4.12’de sunulan, evrişim, toplu normalleştirme ve ReLU blok yapısı, ResNet (He ve ark., 2015) mimarisinde geliştirilen Kimlik Blok yapısından ilham alınmıştır. Bu blok yapısı, evrişim katmanında çıkarılan özellik haritasının normalleştirilip aktive edilmesi için tasarlanmıştır. Normalleştirme işlemi, toplu normalleştirme (Lofte & Szegedy, 2015) katmanı ile, aktivasyon işlemi ise

ReLU katmanı ile sağlanmıştır. Toplu normalleştirme, her evrişim işleminden sonra kullanılarak aktive edilecek girdiler yeniden ölçeklendirilmiştir. Bu işlem, modelin daha kararlı hale gelmesini sağlamıştır. Normalize edilmiş model, ReLU aktivasyon fonksiyonu ile aktive edilmiştir. Bu blok yapısı, önerilen 3 hatlı ESA modelinin ilk hattı olan, özellik çıkarma işleminin temel yapısını oluşturmaktadır.



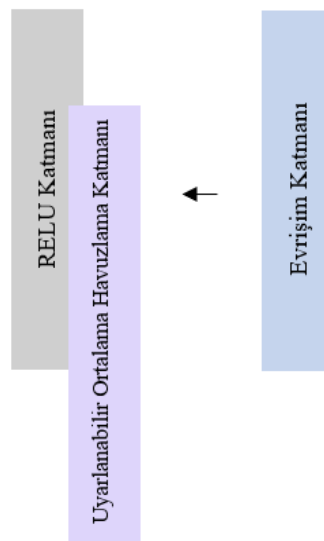
Şekil 4.12. Önerilen Modelin Evrişim, Toplu Normalleştirme ve ReLU Blok Yapısı

Önerilen 3 hatlı ESA modelinin, ilk hattı, yoğun özellik çıkarımı gerçekleştirmektedir. Şekil 4.13'te gösterildiği gibi, evrişim, toplu normalleştirme ve ReLU blok yapısından sonra bir havuzlama katmanı gelir. Bu havuzlama katmanı, maksimum havuzlamadır. Maksimum havuzlama katmanı, 2x2 filtre ile görüntünün genişlik ve yükseklik boyutlarını yarıya indirir. Havuzlama işleminden sonra, çıktı tekrar evrişim, toplu normalleştirme ve ReLU blok yapısına girdi olarak sunulur. Modelin bu aşamasında, bu blok yapısına ek olarak Dropout (Hinton ve ark., 2012) katmanı eklenmiştir. Dropout, ağ içindeki bazı bağlantıların azaltılması ile aşırı öğrenmeyi önleyen bir katmandır (Srivastava ve ark., 2014). Son olarak, aktive edilmiş bir evrişim işlemi uygulanarak 1B verilerden dönüştürülmüş görüntülerin özelliklerin saptanması sağlanmıştır. Şekil 4.13'te verilen temel model adımları, çeşitli kombinasyonlarla devam ettirilerek en yüksek başarıyı getiren ilk hat oluşturulmuştur.



Şekil 4.13. Önerilen 3 Hatlı ESA Modelin İlk Hattının Temel Adımları

Şekil 4.14’te görüldüğü üzere, önerilen 3 hatlı ESA modelinin ikinci hattı, ilk hattan girdi olarak aldığı özellik haritasına tekrar evrişim işlemi uygulayarak son özellik çıkarım işlemini gerçekleştirir. Önerilen 3 hatlı ESA modelinin ikinci hattı, Uygulanabilir Ortalama Havuzlama (UOH) katmanı sonrasında, ReLu katmanı ile sonlandırılır. UOH, verilen girdiden belirli boyutta bir çıktı üretmek için gerekli olan doğru çekirdek boyutunu hesaplayan bir havuzlama işlemidir. Önerilen 3 hatlı ESA modelinin üçüncü ve son hattı, tam bağlantı katmanı ve Sigmoid aktivasyon fonksiyonu ile sınıflandırma işlemini gerçekleştirmektedir.



Şekil 4.14. Önerilen 3 Hatlı ESA Modelin İkinci Hattı

Önerilen modelin tahmin ettiği değer ve gerçek değer arasındaki farkı yani modelin hatasını en aza indirmek için Adam (Diederik & Jimmy, 2015) optimizasyon algoritması kullanılmıştır. Modelin öğrenme hızı, $3e-3$ olarak belirlenmiştir. Veri setinin, %80 eğitim seti ve %20 test seti olarak; eğitim setinin de %10'u doğrulama seti olarak bölünmüştür. Eğitim setindeki 1204 veri kullanılarak bir eğitim modeli oluşturulmuş, doğrulama setindeki 124 veri ile bu eğitim modelinin performansı iyileştirilmiştir. 100 epok sonucunda elde edilen performans, eğitim setinin başarısını vermiştir. Eğitim ve doğrulama hata hesaplamaları için ÇEK kullanılmıştır. Eğitim setinin başarısı elde edildikten sonra bu eğitim modeli üzerinde, test setindeki 335 görüntü verileri önerilen 3 hatlı ESA modeline sunularak test işlemleri yapılmıştır.

Veri ön işlemlerden geçirilmiş ve 64x64 boyutlu görüntü verilerine dönüştürülmüş görüntüleri girdi olarak alan 3 hatlı ESA modelin ve popüler ESA modellerinin başarısı Tablo 4.1'de sunulmuştur.

Tablo 4.1. 64x64 Boyutlu Görüntüler ile Önerilen 3 Hatlı ESA Model Başarısının Popüler ESA Modelleri ile Karşılaştırması

ESA MODELİ	Doğruluk	Kesinlik	Duyarlılık	F1-Ölçütü
Önerilen 3 Hatlı ESA	86.21	88.19	82.41	85.20
LeNet-5 (LeCun ve ark., 1998)	84.05	84.89	81.37	83.09
VGG-11 (Simonyan & Zisserman, 2014)	84.55	84.56	83.10	83.82
ResNet-50 (He ve ark., 2015)	81.19	80.36	80.86	80.61

LeNet-5, LeCun ve ark.'nın (1998) geliştirdiği ilk başarılı sonuç veren ESA mimarisidir. Bu mimari, 5 katmandan oluşan temel yapıda bir ESA modelidir ve veri ön işlemlerden geçirilmiş ve 64x64 boyutlu görüntü verilerine dönüştürülmüş OSR veri setinin bu model ile sınıflandırılması sonucunda %84.05 başarı elde edilmiştir. Buna karşın, 11 katmanlı VGG-11 (Simonyan & Zisserman, 2014) ve 50 katmanlı ResNet-50 (He ve ark., 2015) modelleri ile sırasıyla %84.55 ve %81.19 başarı elde edilmiştir. Bu başarılar incelendiğinde, 3 hatlı ESA modelinin elde ettiği %86.21 başarı popüler ESA modellerine göre daha yüksektir.

Önerilen 3 hatlı ESA modeli için görüntü boyutunun sınıflandırma başarısına etkisi incelenmiştir. Tablo 4.2’de, çeşitli görüntü boyutları ile gerçekleştirilen 3 hatlı ESA modelinin sınıflandırma sonuçları sunulmuştur.

Tablo 4.2. Önerilen 3 Hatlı ESA Modelin Görüntü Boyutlarına Göre Sınıflandırma Başarısı

Görüntü Boyutu	Doğruluk	Kesinlik	Duyarlılık	F1-Ölçütü
16x16	94.85	95.92	93.27	94.58
32x32	88.03	89.20	85.51	87.32
64x64	86.21	88.19	82.41	85.20

Tablo 4.2’de veri ön işlemlerden sonra farklı boyutlarda görüntü verilerine dönüştürülmüş görüntüleri girdi olarak alan 3 hatlı ESA modelin elde ettiği başarı sonuçları sunulmuştur. Veri ön işlemlerden geçirilmiş ve 64x64 boyutlu görüntü verilerine dönüştürülmüş görüntüler ile önerilen model %86.21 ve 32x32 boyutunda görüntüler ile %88.03 başarı elde etmiştir. Buna karşın, veri ön işlemlerden geçirilmiş ve 16x16 boyutlu görüntü verilerine dönüştürülmüş görüntüleri girdi olarak alan 3 hatlı ESA mimarisi, %94.85 başarı ile en iyi performansı elde etmiştir. 1B COVID-19 kan parametrelerini içeren ve görüntü verilerine dönüştürülen özellik vektörleri, 33 özelliğe sahiptir. Bu nedenle, görüntü boyutunun küçültülmesi başarıda etkili bir faktör olmuştur. Sonuç olarak, özellik vektörlerindeki özellik sayısına uygun optimum görüntü boyutu belirlemenin önemi ortaya konmuştur.

Veri ön işlemlerden geçirilmiş ve görüntü verilerine dönüştürülmüş OSR veri setini girdi olarak alan 3 hatlı ESA modelin başarısı, veri ön işlemlerden geçirilmiş fakat görüntü verilerine dönüştürülmemiş OSR veri setini girdi olarak alan YSA, Bagging, Adaboost ve RO MÖ sınıflandırıcılarının başarıları ile karşılaştırmalı olarak Tablo 4.3’te incelenmiştir.

Tablo 4.3. Önerilen 3 Hatlı ESA Model Başarısının MÖ Modelleri ile Karşılaştırması

Sınıflandırıcı	Doğruluk	Kesinlik	Duyarlılık	F1-Ölçütü
Önerilen 3 Hatlı ESA	94.85	95.92	93.27	94.58
YSA	75.52	75.60	75.50	75.50
Bagging	81.49	81.50	81.50	81.50
Adaboost	78.50	78.70	78.50	78.50
RO	81.19	81.20	81.20	81.20

YSA, DÖ'nün temelini oluşturan önemli bir sınıflandırıcıdır. Tablo 4.3'te görüldüğü üzere, veri ön işlemlerden geçirilmiş fakat görüntü verilerine dönüştürülmemiş OSR veri setini girdi olarak alan YSA modeli %75.52 başarı elde etmiştir. KÖ yöntemleri, diğer MÖ tekniklerinden daha başarılı sonuçlar üretmektedir (Wiysobunri et al., 2020). Veri ön işlemlerden geçirilmiş fakat görüntü verilerine dönüştürülmemiş OSR veri setini girdi olarak alan ve KÖ yöntemi olan Bagging, Adaboost ve RO sınıflandırıcıları ile sırası ile %81.49, %78.50 ve %81.19 başarı elde edilmiştir. Veri ön işlemlerden geçirilmiş ve görüntü verilerine dönüştürülmüş OSR veri setini girdi olarak 3 hatlı ESA modeli ise %94.85 başarı elde etmiştir. Bu durum, 1B kan parametre verilerini 2B görüntü verilerine dönüştürerek ESA modellerinde kullanmanın başarısını açıkça ortaya koymuştur.

Önerilen 3 hatlı ESA modelin başarısı ile çalışmada kullanılan OSR veri seti ile gerçekleştirilmiş literatürdeki diğer çalışmaların başarısı karşılaştırmalı olarak Tablo 4.4'te sunulmuştur.

Tablo 4.4. Önerilen 3 Hatlı ESA Model Başarısının OSR Veri Setini Kullanarak Gerçekleştirilen Diğer COVID-19 Çalışmaları ile Karşılaştırması

Çalışma	Sınıflandırıcı	Doğruluk
Önerilen 3 Hatlı ESA	ESA	%94.85
Cabitza et al., 2020	RO	%88.00-%93.00
Rikan et al., 2022	EA	%84.98
	DSA	%93.16
Rahman et al., 2022	Yığın Modeli	%91.45
Kistenev et al., 2022	RO ve DVM	%88.00

Tablo 4.4, bu çalışmada kullanılan veri seti ile yapılan diğer çalışmaların sonuçlarını göstermektedir. Cabitza ve ark. (2020), OSR veri setini COVID-19'a özgü kan parametrelerini içeren veri seti ve TKS kan parametrelerini içeren veri seti olmak üzere ikiye bölmüştür. Daha sonra bu iki veri seti ve OSR veri seti ile çeşitli MÖ modelleri değerlendirilmiştir. Sonuç olarak, RO ile %88.00-%93.00 arasında en yüksek başarılar elde edilmiştir. Rikan ve ark.'nın (2022) gerçekleştirdiği çalışmada, kan parametreleri içeren üç veri seti ile COVID-19'un tespiti için 8 MÖ ve 4 DÖ modeli geliştirilmiştir. Geliştirilen modellerden OSR veri seti ile MÖ modeli olan EA ile

%84.98, DSA ile %93.16 en yüksek başarı elde edilmiştir. Elde edilen bu başarılar incelendiğinde, OSR veri seti için DÖ yönteminin MÖ yönteminden daha başarılı sonuç getirdiği kanıtlanmıştır. COVID-19 tespiti için Rahman ve ark. (2022), tarafından geliştirilen QCovSML adlı MÖ modeli ise Gradyan Arttırma, RO ve gradyan arttırma temelli bir KA olan XGBoost sınıflandırıcılarından oluşan bir yığın MÖ modelidir. 3 sınıflandırıcı ile geliştirilmiş bu yığın modeli, %91.45 başarı elde etmiştir. Son olarak, Kistenev ve ark.'ları (2022) tarafından gerçekleştirilen çalışmada ise OSR veri seti ile RO ve DVM sınıflandırıcılarından %88.00 başarı elde edilmiştir. Sonuç olarak, önerilen OSR veri setini görüntü verilerine dönüştürerek sınıflandıran 3 hatlı ESA modelin başarısı, OSR veri seti kullanılarak gerçekleştirilen diğer çalışma sonuçlarından daha yüksek bulunmuştur. Aynı veri setini kullanan diğer COVID-19 tespit çalışmalarına karşı önerilen 3 hatlı ESA modelin daha başarılı olması, DeepInsight (Sharma ve ark., 2019) metodunun kan parametre verilerine uygulanabilirliğini ve kan parametreleri ile COVID-19 tespitinde kullanılabilirliğini ortaya koymuştur. Bu çalışmada önerilen 3 hatlı ESA modeli, birçok MÖ modeline ve ESA modeline karşı yüksek başarı elde ederek YZ tabanlı COVID-19 tespit çalışmalarında etkin bir model olduğunu kanıtlamıştır.

5. SONUÇLAR VE ÖNERİLER

COVID-19 çeşitli varyantlarla kendini yenileyebilen pandemik bir hastalıktır. Bu hastalığı kısa vadede ortadan kaldırmak mümkün olmadığından enfekte bireylerin hızlı tespiti için yöntemler geliştirilmiş ve hastalığın gücünü kırmak için aşular üretilmiştir. Aşular hastalığın gücünü kırsa da virüsün sürekli varyant değiştirmesi aşuların etkinliğini azaltmaktadır. Bu durum sonucunda, COVID-19 için en etkili yöntemin enfekte bireyleri tespit etmek ve hastalığın yayılmasını durdurmak olduğu öngörülmüştür. YZ, özellikle bir hastaya tanı koymak için geçen sürenin neden olduğu sakıncaları en aza indirdiğinden (Clifton, 2018; Gill, 2018) COVID-19 tespitinde YZ'nin aktif kullanımı, virüs yayılımını en aza indirmek için oldukça önemlidir. YZ sistemleri ile enfekte bireylerin tespiti hızlı ve doğru bir şekilde yapılabilirse bu hastalığın yayılmasını kontrol altına almak daha kolay olacaktır.

COVID-19 tanı aşamasında ilk olarak uygulanan RT-PCR testinin, sonuçların alınmasına kadar geçen süre ve uzman gereksinimi gibi dezavantajları vardır. Diğer bir alternatif olan, BT ve X-ray görüntüleri ise hem maliyetli hem de bir uzman radyolog tarafından incelenmesi gerektiğinden COVID-19 için bir darboğaz oluşturmaktadır. Hastalığın tespitinin daha hızlı ve güvenilir bir şekilde olması, hastalığı önlemede ve risk grubu olarak tanımlanan yaşlılar ve kronik hastalığı olanlar için riski en aza indirmede fayda sağlayacaktır. Tüm bu nedenlerden dolayı, YZ tabanlı sistemlerin bu mücadelede aktif olarak kullanılması bir zorunluluktur. Bu tez çalışmasında, 1B verilerden oluşan COVID-19 kan parametreleri için önerilen veri ön işleme, 1B verilerin 2B verilere dönüşümü ve 3 hatlı ESA kombinasyonu sağlanarak COVID-19 tespiti amaçlanmıştır.

Bu tez kapsamında, COVID-19 pozitif ve negatif 1736 bireyin kan parametrelerini içeren OSR veri setinin %75'den fazlası eksik veri içeriyorsa o örnekler silinmiş (Cabitza ve ark., 2020), KNN yöntemi ile eksik veriler tamamlanmış (Erol ve ark., 2022) ve $[-1 \ 1]$ aralığında standardizasyon veri ön işlemleri uygulanmıştır. Sonrasında, 1B verileri 2B verilere dönüştürme adımları (Sharma ve ark., 2019) uygulanmıştır: İlk olarak her bir özellik vektörü, T-SNE algoritması ile 2B düzleme aktarılmıştır. Görüntü verilerinin sınırlarını oluşturmak için 2B düzleme aktarılan özellik vektörleri, DGA ve MSD ile çerçevelenmiştir. ESA mimarisi için görüntünün yatay veya dikey olarak çerçevelere sahip olması gerektiğinden rotasyon işlemi uygulanarak piksel haritalama yapılmış ve görüntü matrisleri elde edilmiştir. Elde edilen

görüntü matrisleri, RGB formata dönüştürülerek görüntü verileri elde edilmiştir. Elde edilen görüntüler, geliştirilen 3 hatlı ESA mimarisine sunularak %94.85 başarı elde edilmiştir. Elde edilen bu başarı, popüler ESA modelleri, veri ön işlemlerden geçirilmiş fakat görüntü verilerine dönüştürülmemiş OSR veri setini girdi olarak alan MÖ yöntemleri ve OSR veri seti ile gerçekleştirilmiş literatürdeki diğer COVID-19 çalışmaları ile karşılaştırmalı olarak analiz edilmiştir. Analizler sonucunda, bu çalışma ile önerilen yöntemin, kan parametre verilerine uygulanabilirliği ve kan parametreleri ile COVID-19 tespitinde kullanılabilirliği ortaya koyulmuştur. Buna ek olarak, bu çalışma ulaşılabilirliği kolay ve düşük maliyetli olan kan parametreleri ile başarı seviyesi yüksek ESA mimarisinin bir kombinasyonunu sunarak COVID-19 tespiti için bir rehber niteliğindedir. İleride yapılacak çalışmalarda, hastaların hem BT görüntüleri hem de kan parametreleri toplanarak elde edilen büyük veri setleri ile görüntü verilerine dönüştürülmüş kan parametre görüntüleri karşılaştırmalı olarak değerlendirilebilir ve daha başarılı COVID-19 tespitleri yapılabilir. Buna ek olarak, COVID-19'a benzer diğer salgın hastalıklar için de gerekli güncellemeler yapılarak çeşitli tespit modelleri geliştirilebilir.

KAYNAKLAR

- Abayomi-Alli, O.O., Damaševicius, R., Maskeliunas, R., Misra, S., 2022, An Ensemble Learning Model for COVID-19 Detection from Blood Test Samples. *Sensors*. 22(6):2224. <https://doi.org/10.3390/s22062224>
- Abeel, T., Van de Peer, Y., Saeys Y, 2009. Toward a gold standard for promoter prediction evaluation. *Bioinformatics*, 25(12):i313–i320. <https://doi.org/10.1093/bioinformatics/btp191>.
- Ai, T. , Yang, Z. , Hou, H. , Zhan, C., Chen, C. , Lv, W., et al., 2020, Correlation of chest CT and RT-PCR testing in coronavirus disease 2019 (COVID-19) in China: a report of 1014 cases. *Radiology*, 296(2):E32-E40 <https://doi.org/10.1148/radiol.2020200642>
- Akosa, J., 2017. Predictive accuracy: a misleading performance measure for highly imbalanced data. *In Proceedings of the SAS Global Forum 2017*. <https://support.sas.com/resources/papers/proceedings17/0942-2017.pdf>
- Alakuş T.B. & Turkoglu I., 2020, Comparison of deep learning approaches to predict COVID-19 infection. *Chaos Solitons Fractals*. 140(11):110120. <https://doi.org/10.1016/j.chaos.2020.110120>
- Alazab, M., Awajan, A., Mesleh, A., Abraham, A., Jatana, V., Alhyari, S., 2020, COVID-19 Prediction and Detection Using Deep Learning. *International Journal of Computer Information Systems and Industrial Management Applications*. 12:168:181. www.mirlabs.net/ijcisim/index.html
- Albawi, S. & Mohammed, T.A., 2017. Understanding of a Convolutional Neural Network, *The International Conference on Engineering and Technology 2017*. <https://doi.org/10.1109/ICEngTechnol.2017.8308186>
- Albayrak, M., Turhan, K., & Kurt, B. (2017). Kümeleme ve Maksimum Olabilirlik Yaklaşımıyla Eksik Veri Tamamlama, A Missing Data Imputation Using Clustering and Maximum Likelihood Estimation. *2017 Medical Technologies National Congress (TIPTEKNO)*, 242–245. <https://doi.org/10.1109/TIPTEKNO.2017.8238064>
- Alexandridi, M., Mazej, J., Palermo, E., Hiscott, J., 2022, The Coronavirus pandemic-2022: Viruses, variants & vaccines. *Cytokine Growth Factor Rev*. 63(2):1.9. <https://doi.org/10.1016/j.cytogfr.2022.02.002>
- Aljame, M., Ahmad, I., Imtiaz, A., & Mohammed, A., 2020, Ensemble learning model for diagnosing COVID-19 from routine blood tests. *Informatics in Medicine Unlocked*, 21(October). <https://doi.org/10.1016/j.imu.2020.100449>
- Alkhodari M., Khandoker A.H., 2022, Detection of COVID-19 in smartphone-based breathing recordings: A pre-screening deep learning tool. *PLoS ONE*, 17(1): e0262448. <https://doi.org/10.1371/journal.pone.0262448>
- Alpaydin, E., 2004. Introduction to machine learning. *The MIT Press Cambridge*, Massachusetts London, England, 433.
- Andriasyan V., Yakimovich A., Georgi F., et al., 2019, Deep learning of virus infections reveals mechanics of lytic cells. *bioRxiv*. <https://doi.org/10.1101/798074>
- Antin, B., Kravitz, J., Martayan, E., 2017, Detecting pneumonia in chest X-rays with supervised learning. <http://cs229.stanford.edu/proj2017/final-reports/5231221.pdf>
- Ayan, E. & Ünver, H.M., 2019, Diagnosis of pneumonia from chest X-ray images using deep learning. *In: Scientific meeting on electrical-electronics and biomedical engineering and computer science (EBBT)*, Istanbul, Turkey, pp 1–5. <https://doi.org/10.1109/EBBT.2019.8741582>

- Aydın, Ş.E., 2017, Yapay Zek Teknolojisi (Yapay Zekaların Dünyü Bugünü Yarını). Çukurova Üniversitesi, Sosyal Bilimler Enstitüsü İşletme ve Teknoloji Yönetimi. https://www.academia.edu/35714946/YAPAY_ZEK%C3%82_TEKNOLOJ%C4%B0S%C4%B0
- Ayık, Y. Z., Özdemir, A. ve Yavuz, U. (2007). Lise türü ve lise mezuniyet başarısının, kazanılan fakülte ile ilişkisinin veri madenciliği tekniği ile analizi. *Atatürk Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 10(2):441-454. <https://dergipark.org.tr/tr/download/article-file/32076>
- Azur, M. J., Stuart, E. A., Frangakis, C., & Leaf, P. J., 2011. Multiple imputation by chained equations: what is it and how does it work?. *International Journal of Methods in Psychiatric Research*, 20(1), 40–49. <https://doi.org/doi.org/10.1002/mpr.329>
- Barber, C.C., Dobkin, D.P., Huhdanpaa, H., 1996. The Quickhull Algorithm for Convex Hull. *ACM Transactions on Mathematical Software*. 22(4):469-483. <https://doi.org/10.1145/235815.235821>
- Barton, M.I., MacGowan, S.A., Kutuzov, M.A., et al., 2021, Effects of common mutations in the SARS-CoV-2 Spike RBD and its ligand, the human ACE2 receptor on binding affinity and kinetics. *eLife*. <https://doi.org/10.7554/eLife.70658>
- Bassi, P.R.A.S. & Attux, R., 2022, A deep convolutional neural network for COVID-19 detection using chest X-rays. *Research on Biomedical Engineering*. 38:139–148. <https://doi.org/10.1007/s42600-021-00132-9>
- Bozkır, A. S., Sezer, E., ve GÖK, Bilge.,(2009), Öğrenci seçme sınavında (öss) öğrenci başarımını etkileyen faktörlerin veri madenciliği yöntemleriyle tespiti. *5. Uluslararası İleri Teknolojiler Sempozyumu (IATS'09)*.
- Breiman L.,1996, Bagging predictors, 1996, *Machine Learning*, 24:123-140. <https://doi.org/10.1007/BF00058655>
- Breiman L.,2001, Random forests, *Machine Learning*, 45:5-32. <https://doi.org/10.1023/A:1010933404324>
- Brinati, D., Campagner, A., Ferrari, D., Locatelli, M., Banfi, G., & Cabitza, F., 2020, Detection of COVID-19 Infection from Routine Blood Exams with Machine Learning : A Feasibility Study. *Journal of Medical Systems*, 44(135). <https://doi.org/10.1007/s10916-020-01597-4>
- Buturovic, L. & Miljkovic, D., 2020, A novel method for classification of tabular data using convolutional neural networks. *bioRxiv*. <https://doi.org/10.1101/2020.05.02.074203>
- Cabitza, F., Campagner, A., Ferrari, D., et al., 2020, Development, evaluation and validation of machine learning models for COVID-19 detection based on routine blood tests. *Clinical Chemistry and Laboratory Medicine*. 59(2), 421–431. <https://doi.org/10.1515/cclm-2020-1294>
- Callaway, E., 2021, The mutation that helps Delta spread like wildfire. *Nature* 596. (7873):472–473. <https://doi.org/10.1038/d41586-021-02275-2>
- Chang V., Bhavani V. R., Xu A. Q., Hossain M., 2022, An artificial intelligence model for heart disease detection using machine learning algorithms, *Healthcare Analytics*, 2. <https://doi.org/10.1016/j.health.2022.100016>
- Chen, H., Guo, J., Wang, C., Luo, F., Yu, X., Zhang, W., ve ark., 2020. Clinical characteristics and intrauterine vertical transmission potential of COVID-19 infection in nine pregnant women: retrospective review of medical records. *Lancet*, 395(10226):809-815. [https://doi.org/10.1016/S0140-6736\(20\)30360-3](https://doi.org/10.1016/S0140-6736(20)30360-3)

- Cover, T.M., 1965, Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. *IEEE Trans. Electron. Comput.* 14(3):326–334. <https://doi.org/10.1109/PGEC.1965.264137>
- Davenport T. & Kalakota R., 2019, The potential for artificial intelligence in healthcare. *Future Healthcare Journal*, 6 (2) (2019) 94–98. <http://dx.doi.org/10.7861/futurehosp.6-2-94>
- Day, M., 2020, Covid-19 : identifying and isolating asymptomatic people helped eliminate virus in Italian village. *The BMJ*. 368(1165). <https://doi.org/10.1136/bmj.m1165>
- Demirdal T. & Sen P., 2018, The significance of neutrophil-lymphocyte ratio, platelet-lymphocyte ratio and lymphocyte-monocyte ratio in predicting peripheral arterial disease, peripheral neuropathy, osteomyelitis and amputation in diabetic foot infection. *Diabetes Res Clin Pract.* 144:118–125. <http://doi.org/10.1016/j.diabres.2018.08.009>
- Dey N., Mishra R., Fong S.J., Santosh K.C., Tan S., Crespo R.G., 2020, COVID-19: Psychological and psychosocial impact, fear, and passion. *Digit. Gov.: Res. Pract.*, 1, 1–4. <http://doi.org/10.1145/3428088>
- Diederik, P.K. & Jimmy, L.B., 2015, Adam: A Method for Stochastic Optimization. 3rd International Conference on Learning Representations (ICLR 2015). <https://arxiv.org/pdf/1412.6980.pdf>
- Dorj U.O., Lee K.K., Choi J.Y., Lee M., 2018. The skin cancer classification using deep convolutional neural network. *Multimed Tools Appl.*, 77(8):9909–9924. <http://doi.org/10.1007/s11042-018-5714-1>
- Dovhalets, D., Kovalerchuk, B., Vajda, S., Andonie, R., 2018, Deep Learning of 2-D Images Representing n-D Data in General Line Coordinates. *Intern. Symp. on Affective Science and Engineering*, 1-6. <http://doi.org/10.5057/ISASE.2018-C000025>
- Dua, S. & Chowriappa, P., 2013. Data mining for bioinformatics, *CRC Press*, ISBN: 978-1-4200-0430-4.
- Engel, D., Hüttenberger, L., Hamann, B., 2011. A survey of dimension reduction methods for high-dimensional data analysis and visualization. *VLUDS*. <http://doi.org/10.4230/OASlcs.VLUDS.2011.135>
- Erol, G., Uzbaş, B., Yücelbaş, C., Yücelbaş, Ş., 2022, Analyzing The Effect Of Data Pre-Processing Techniques Using Machine Learning Algorithms On The Diagnosis Of COVID-19, *Concurrency and Computation-Practice & Experience*, 34(28). <https://doi.org/10.1002/cpe.7393>
- Ertorsun, A.D., Bağ, B., Uzar, G. ve Turanoğlu, M. A., 2009. ROC (Receiver Operating Characteristic) eğrisi yöntemi ile tanı testlerinin performanslarının değerlendirilmesi, *XIII. Öğrenci Sempozyumu 2009*, Ankara. <http://tip.baskent.edu.tr/kw/upload/464/dosyalar/cg/sempozyum/ogrsmpznm12/10.2.pdf>
- Fayyad, U.M., Piatetsky-Shapiro, G., and Smyth, P., 1996. Knowledge discovery and data mining: towards a unifying framework. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD'96)*. AAAI Press, 82–88.
- Ferrari D., Motta A., Strollo M., Banfi G., Locatelli M., 2020, Routine blood tests as a potential diagnostic tool for COVID-19, *Clin. Chem. Lab. Med.*, 58(7), 1095-1099. <http://doi.org/10.1515/cclm-2020-0398>
- Fong S., Li G., Dey N., Crespo R.G., Herrera-Viedma E., 2020, Finding an accurate early forecasting model from small dataset: A case of 2019-nCoV novel coronavirus outbreak. *Int. J. Interact. Multimedia Artif. Intell.* 6 (1):1–10. <http://dx.doi.org/10.9781/ijimai.2020.02.002>
- Freud, Y. & Schapire, R.E. 1999, A Short Introduction to Boosting, *Journal of Japanese Society for Artificial Intelligence*. 14(5):771-780. <https://cseweb.ucsd.edu/~yfreund/papers/IntroToBoosting.pdf>

- Gaál G., Maga B., Lukács A., 2020, Attention U-Net based adversarial architectures for chest X-ray lung segmentation. arXiv. <https://arxiv.org/abs/2003.10304>
- Goodfellow, I., Bengio, Y., Courville, A. ve Bengio, Y., 2016, Deep learning, MIT press Cambridge, p.
- Göreke V., Sarı V., Kockanat S., 2021, A novel classifier architecture based on deep neural network for COVID-19 detection using laboratory findings, *Applied Soft Computing*, 106(1):107329. <https://doi.org/10.1016/j.asoc.2021.107329>
- Guil-Reyes, F., & Daza, M.T., 2011. Summarizing frequent itemsets via pignistic transformation. *In Portuguese Conference on Artificial Intelligence*, pp. 297-310. https://doi.org/10.1007/978-3-642-24769-9_22
- Guan W., Ni Z., Hu Y., Liang W., Ou C., He G.J., et al., 2019, Clinical characteristics of coronavirus disease 2019 in China, *New Engl. J. Med.*, 382(18), 1708–1720. <https://doi.org/10.1056/NEJMoa2002032>
- Guyon, I., Elissee, A., 2003. An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research*, 3 1157-1182. <https://doi.org/10.1162/153244303322753616>
- Hacıefendioğlu, Ş., 2012. Makine öğrenmesi yöntemleri ile glokom hastalığının teşhisi, Yüksek Lisans Tezi, Selçuk Üniversitesi, Fen Bilimleri Enstitüsü, Konya.
- Haghanifar, A., Majdabadi, M.M., Choi, Y., Deivalakshmi, S., Ko, S., 2022, COVID-CXNet: Detecting COVID-19 in frontal chest X-ray images using deep learning. *Multimedia Tools and Applications*, 7:1-31. <https://doi.org/10.1007/s11042-022-12156-z>
- Hakak S., Khan W. Z., Imran M., Choo K.-K. R., Shoaib M., 2020. Have You Been a Victim of COVID-19-Related Cyber Incidents? Survey, Taxonomy, and Mitigation Strategies. *IEEE Access*, 30(8), 124134-124144. <http://doi.org/10.1109/ACCESS.2020.3006172>
- Harvey, W.T., Carabelli, A.M., Jackson, B., et al., 2021, SARS-CoV-2 variants, spike mutations and immune escape. *Nat. Rev. Microbiol.* 19 (7):409–424, <https://doi.org/10.1038/s41579-021-00573-0>.
- He, K., Zhang, X., Ren, S., Sun, J., 2015. Deep Residual Learning for Image Recognition, arXiv, <https://doi.org/10.48550/arXiv.1512.03385>
- Hinton, G.E., Srivastava, N., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.R., 2012, Improving neural networks by preventing co-adaptation of feature detectors, arXiv. <https://arxiv.org/pdf/1207.0580v1.pdf>
- Holmes, E.C., Goldstein, S.A., Rasmussen, A.L., et al., 2021, The origins of SARS-CoV-2: a critical review, *Cell* 184 (19):4848–4856. <https://doi.org/10.1016/j.cell.2021.08.017>
- Hssina, B., Merbouha, A., Ezzikouri, H., Erritali, M., 2014, A comparative study of decision tree ID3 and C4. 5, *International Journal of Advanced Computer Science and Applications*, <https://doi.org/10.14569/SpecialIssue.2014.040203>
- Hu, J., Han, Z., Heidari, A.A., 2022, Detection of COVID-19 severity using blood gas analysis parameters and Harris hawks optimized extreme learning machine. *Computers in Biology and Medicine*. 142:105166. <https://doi.org/10.1016/j.combiomed.2021.105166>
- Huang, H. & Bader, J.S., 2009. Precision and recall estimates for two hybrid screens. *Bioinformatics*, 25(3):372–378. <https://doi.org/10.1093/bioinformatics/btn640>

- Huang, C., Wang, Y., Li, X., Ren, L., Zhao, J., Hu, Y., ve ark., 2020. Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China. *Lancet*, 395(10223):497–506. [https://doi.org/10.1016/S0140-6736\(20\)30183-5](https://doi.org/10.1016/S0140-6736(20)30183-5)
- Hulley, S.B., Cummings, S.R., Browner, W.S., Grady, D., Hearst, N., Newman, T.B., 2001. Studies of the Accuracy of Tests. *Designing Clinical Research An Epidemiologic Approach*. Second Edition. *Lippincott Williams&Wilkins*, 181-2.
- Idri, A., Abnane, I., Abran, A., 2016. Missing data techniques in analogy-based software development effort estimation. *The Journal of Systems and Software*, 117, 595–611. <https://doi.org/10.1016/j.jss.2016.04.058>
- Iwendi C., Bashir A.K., Peshkar A., Sujatha R., Chatterjee J.M., et al., 2020. COVID-19 Patient Health Prediction Using Boosted Random Forest Algorithm. *Front Public Health*, 3;8:357. <https://doi.org/10.3389/fpubh.2020.00357>
- Jaiswal, A.K., Tiwari, P., Kumar, S., Gupta, D., Khanna, A., Rodrigues, J.J., 2019, Identifying pneumonia in chest X-rays: a deep learning approach. *Measurement* 145:511–518. <https://doi.org/10.1016/j.measurement.2019.05.076>
- Jahmunah V., Ng E.Y.K., San T. R., Acharya U. R., 2021, Automated detection of coronary artery disease, myocardial infarction and congestive heart failure using GaborCNN model with ECG signals, *Computers in Biology and Medicine*. 134. <https://doi.org/10.1016/j.combiomed.2021.104457>
- Jan H.C., Yang W.H., Ou C.H., 2019, Combination of the preoperative systemic immune-inflammation index and monocyte-lymphocyte ratio as a novel prognostic factor in patients with upper-tract urothelial carcinoma, *Ann. Surg. Oncol.*, 26(2), 669–684. <http://doi.org/10.1245/s10434-018-6942-3>
- Japkowicz, N., Shah, M., 2011. Evaluating Learning Algorithms, *Cambridge University Press*. <https://doi.org/10.1017/CBO9780511921803>
- Jauro F., Chiroma H., Gital A.Y., Almutairi M., Abdulhamid S.M., Abawajy J.H., 2020. Deep learning architectures in emerging cloud computing architectures: Recent development, challenges and next research trend. *Appl. Soft Comput.*, 96:106582. <http://dx.doi.org/10.1016/j.asoc.2020.106582>.
- Jiang F., Deng L., Zhang L., Cai Y., Cheung C. W., Xia Z., 2020, "Review of the clinical characteristics of coronavirus disease 2019 (COVID-19)," *Journal of General Internal Medicine*, 35(5): 1545–1549. <https://doi.org/10.1007/s11606-020-05762-w>
- Karlis, D., 2002. An EM Type Algorithm for Maximum Likelihood Estimation of the Normal-Inverse Gaussian Distributions. *Statistics & Probability Letters*. 57(1): 43-52. [https://doi.org/10.1016/S0167-7152\(02\)00040-8](https://doi.org/10.1016/S0167-7152(02)00040-8)
- Kılıç S., 2013. Klinik karar vermede ROC analizi. *Journal of Mood Disorders*, (3):135-40. <https://dergipark.org.tr/tr/download/article-file/109764>
- Kovalerchuk, B., Dovhalets, D., 2017, Constructing Interactive Visual Classification, Clustering and Dimension Reduction Models for n-D Data. *Informatics*, 4(23):23. <https://doi.org/10.3390/informatics4030023>
- Kovalerchuk, B., Agarwal, B., Kal, D.C., 2020, Solving Non-image Learning Problems by Mapping to Images, *24th International Conference on Information Visualisation*. <https://doi.org/10.1109/IV51561.2020.00050>
- Kulin, Merima & Kazaz, Tarik & De Poorter, Eli & Moerman, Ingrid. (2021). A survey on machine learning-based performance improvement of wireless networks: *PHY, MAC and Network Layer*. *Electronics*. <https://doi.org/10.3390/electronics10030318>

- LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., Jackel, L. D., 1989, Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation* 1, 541-551. <https://doi.org/10.1162/neco.1989.1.4.541>
- LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998, Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278 – 2324. <https://doi.org/10.1109/5.726791>
- Li, Z., Yi, Y., Luo, X., Xiong, N., et al., 2020, Development and clinical application of a rapid IgM - IgG combined antibody test for SARS - CoV - 2 infection diagnosis. *Journal of Medical Virology*, 92(February), 1518–1524. <https://doi.org/10.1002/jmv.25727>
- Liang G. & Zheng L., 2020, A transfer learning method with deep residual network for pediatric pneumonia diagnosis. *Comput Methods Programs Biomed*, 187:104964. <https://doi.org/10.1016/j.cmpb.2019.06.023>
- Lippi, G. & Plebani, M., 2020. Laboratory abnormalities in patients with COVID-2019 infection, *Clin. Chem. Lab. Med.*, 58(7):1131-1134. <https://doi.org/10.1515/cclm-2020-0198>
- Liu, Y., Yang, Y., Zhang, C., Huang, F., Wang, F., Yuan, J., et al., 2020. Clinical and biochemical indexes from 2019-nCoV infected patients linked to viral loads and lung injury. *Sci China Life Sci*, 63(3):64–74. <https://doi.org/10.1007/s11427-020-1643-8>
- Loffe, S. & Szegedy, C., 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv*. <https://doi.org/10.48550/arXiv.1502.03167>
- Maaten, L.V.D. & Hinton, G., 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research* 9(11):2579-2605. <https://jmlr.org/papers/volume9/vandermaaten08a/vandermaaten08a.pdf>
- Martinasek, Z., Hajny, J., Malina, L., 2014. Optimization of power analysis using neural network, *Smart Card Research and Advanced Applications*, ISBN: 3-319-08301-5, 94–107. https://doi.org/10.1007/978-3-319-08302-5_7
- Mathiak, B., & Eckstein, S., 2014. Five Steps to Text Mining in Biomedical Literature Five Steps to Text Mining in Biomedical Literature. *WS on Data Mining and Text Mining for Bioinformatics at PKDD/ECML 2004. Proc. 15th European Conf. on Machine Learning and the 8th European Conf. on Principles and Practice of Knowledge Discovery in Databases, September*, 47–50. https://www.academia.edu/27467487/Five_Steps_to_Text_Mining_in_Biomedical_Literature
- Mei, X., Lee, H.C., Diao, K., Huang, M., Lin, B., Liu, C., et al. 2020, Artificial intelligence-enabled rapid diagnosis of patients with COVID-19. *Nat Med*, 26(8):1224–1228. <https://doi.org/10.1038/s41591-020-0931>
- Melgani, F. & Bruzzone, L., 2004, Classification of hyperspectral remote sensing images with support vector machines. *IEEE Transactions on Geoscience and Remote Sensing*. 42(8):1778-1790. <https://doi.org/10.1109/TGRS.2004.831865>
- Menni, C., Valdes, A.M., Freidin, M.B., et al., 2020, Real-time tracking of self-reported symptoms to predict potential COVID-19. *Nature Medicine*. 2020; 26(7):1037–1040. <https://doi.org/10.1038/s41591-020-0916-2>
- Mishra, M. & Srivastava, M., 2014, A view of Artificial Neural Network. *IEEE International Conference on Advances in Engineering & Technology Research (ICAETR - 2014)*. <https://doi.org/10.1109/ICAETR.2014.7012785>
- Mitchell, T. M., 1997. *Machine Learning (1st edition)*. McGraw-Hill Education.

- Munir K., Elahi H., Ayub A., Frezza F., Rizzi A., 2019, Cancer diagnosis using deep learning: a bibliographic review. *Cancers (Basel)*. 11(9):1235. <https://doi.org/10.3390/cancers11091235>
- Narayan Das, N., Kumar, N., Kaur, M., Kumar, V., Singh, D., 2020, Automated deep transfer learning-based approach for detection of COVID-19 infection in chest X-rays. *IRBM*. 432(2):114-119. <https://doi.org/10.1016/j.irbm.2020.07.001>
- Narin, A., Kaya, C., Pamuk Z., 2021, Automatic detection of coronavirus disease (COVID-19) using X-ray images and deep convolutional neural networks. *Pattern Anal Appl*. 24(3):1207-1220. <http://dx.doi.org/10.1007/s10044-021-00984-y>
- Neumann, J.V., 1945, First Draft of a Report on the EDVAC. *Moore School of Electrical Engineering University of Pennsylvania*. W-670-ORD-4926. <https://web.archive.org/web/20040423232125/http://www.virtualtravelog.net/entries/2003-08-TheFirstDraft.pdf>
- Olson, D.L. & Delen, D., 2008. Advanced data mining techniques. *Springer Science & Business Media*. <https://doi.org/10.5860/choice.45-6838>
- Ozturk, T., Talo, M., Yildirim, E.A., Baloglu, U.B., Yildirim, O., Rajendra, Acharya U., 2020, Automated detection of COVID-19 cases using deep neural networks with X-ray images. *Comput Biol Med*, 121:103792. <https://doi.org/10.1016/j.compbimed.2020.103792>
- Öztemel, E., 2012, Yapay Sinir Ağları (3. Basım). Papatya Yayıncılık
- Papanikolaou, V., Chrysovergis, A., Ragos, V., et al., 2022, From delta to Omicron: S1-RBD/S2 mutation/deletion equilibrium in SARS-CoV-2 defined variants. *Gene*. 814, 146134, <https://doi.org/10.1016/j.gene.2021.146134>
- Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L. ve Lerer, A., 2017. Automatic differentiation in pytorch. *31st Conference on Neural Information Processing Systems (NIPS 2017)*, 4-9.
- Paules, C.I., Marston H.D. & Fauci A.S., 2020, Coronavirus infections more than just the common cold. *JAMA*. 323(8):707-708. <https://doi.org/10.1001/jama.2020.0757>
- Quinlan, J.R., 1986, Induction of Decision Trees, 1986 *Kluwer Academic Publishers*. 1:81-106. <https://hunch.net/~coms-4771/quinlan.pdf>
- Rasmussen, A.L., 2021, On the origins of SARS-CoV-2. *Nat. Med.* 27(1):9, <https://doi.org/10.1038/s41591-020-01205-5>
- Rende, F. Ş., Bütün, G., Karahan, Ş., Enstitüsü, B. T. ve TÜBİTAK BİLGEM, G., 2016, Derin Öğrenme Algoritmalarında Model Testleri: Derin Testler, Bilişim Teknolojileri Enstitüsü, TÜBİTAK BİLGEM.
- Saba T., Mohamed A.S., El-Affendi M., Amin J., Sharif M., 2020. Brain tumor detection using fusion of hand crafted and deep learning features. *Cogn Syst Res*, 59:221-230. <https://doi.org/10.1016/j.cogsys.2019.09.007>
- Saharidis, G.K.D., Androulakis, I.P., Lerapetritou, M.G., 2011. Model building using bi-level optimization. *Journal of Global Optimization*, 49 (1):49-67. <https://doi.org/10.1007/s10898-010-9533-9>
- Sarnaik, S., Simpson P.M., Hamre M., Williams J.A., Ravindranath Y., 1999, Distributions of ethnic differences in blood counts are suggestive of genetic variability, *Pediatr. Res.*, 45, 152. <https://doi.org/10.1203/00006450-199904020-00903>

- Shannon, C. E., 1949. Communication theory of secrecy systems. *Bell Labs Technical Journal*, 28(4): 656-715. <https://doi.org/10.1002/j.1538-7305.1949.tb00928.x>
- Sharma, A. & Kumar, D., 2020. Non-image Data Classification with Convolutional Neural Networks. *arXiv*. <https://arxiv.org/abs/arXiv:2007.03218v1>
- Sharma, A, Vans E, Shigemizu D, Boroevich KA, Tsunoda T., 2019. DeepInsight: A methodology to transform a non-image data to an image for convolution neural network architecture. *Nature: Scientific Reports*, Aug 6, 9(1):1-7. <https://doi.org/10.1038/s41598-019-47765-6>
- Sharma, S., Sharma, S., Athaiya, A., 2020, Activation Functions in Neural Networks. *International Journal of Engineering Applied Sciences and Technology*, 4(12):310-316. <https://doi.org/10.33564/IJEAST.2020.v04i12.054>
- Shen, L., Margolies L.R., Rothstein J.H., Fluder E., McBride R., Sieh W., 2019. Deep Learning to Improve Breast Cancer Detection on Screening Mammography, *arXiv*. <https://arxiv.org/abs/1708.09427>
- Simonyan, K. & Zisserman, A., 2014, Very Deep Convolutional Networks for Large-Scale Image Recognition, *arXiv*, <https://arxiv.org/abs/1409.1556>.
- Singh, R. K., SivaBalakrishnan, M., 2015. Feature Selection of Gene Expression Data for Cancer Classification: A review. *2nd International Symposium on Big Data and Cloud Computing*, pp. 52-57. <https://doi.org/10.1016/j.procs.2015.04.060>
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15(1):1929-1958. <https://www.cs.toronto.edu/~rsalakhu/papers/srivastava14a.pdf>
- Sun, S., Cai X., Wang H., He G., Lin Y., Lu B., et al., 2020, Abnormalities of peripheral blood system in patients with COVID-19 in Wenzhou, China, *Clin. Chim. Acta*, 507:174–180. <http://doi.org/10.1016/j.cca.2020.04.024>
- Teramoto, R., Saito, C., Funahashi, S., 2014. Estimating causal effects with a non-paranormal method for the design of efficient intervention experiments. *BMC Bioinformatics*, 15(1):228. <https://doi.org/10.1186/1471-2105-15-228>
- Tolon, M. & Tosunoğlu, N.G., 2008. Tüketici Tatmini Verilerinin Analizi: Yapay Sinir Ağları Ve Regresyon Analizi Karşılaştırması. *DergiPark*. 10(2):247-259. <https://dergipark.org.tr/tr/download/article-file/287453>
- Turing, A.M., 1950, Computing Machinery and Intelligence. *Mind*. Volume LIX, Issue 236, 433–460. <https://doi.org/10.1093/mind/LIX.236.433>
- Ülker, E. & İnik, Ö., 2017. Derin Öğrenme ve Görüntü Analizinde Kullanılan Derin Öğrenme Modelleri, *Gaziosmanpaşa Bilimsel Araştırma Dergisi (GBAD)*. 6(3):85.104. <https://dergipark.org.tr/tr/download/article-file/380999>
- Vapnik, V.N., 1995, The Nature of Statistic Learning Theory. *Springer– Verlag*.
- Vellido, A., Martin-Guerrero, J. D., Lisboa, P. J. G., 2012, Making machine learning models interpretable, *Louvain-La-Neuve*, ISBN: 978-2-87419-049-0.
- Wang, L. & Wong A., 2020, COVID-Net: A Tailored Deep Convolutional Neural Network Design for Detection of COVID-19 Cases from Chest Radiography Images, *arXiv*. <https://arxiv.org/abs/2003.09871>
- Wiyosunri, B., Erden H., Toreyin, B. (2020). An Ensemble Deep Learning System for the Automatic Detection of COVID-19 in X-Ray Images. <https://spacing.itu.edu.tr/pdf/beltus-ytb.pdf>

- World Health Organization (WHO), 2019, Report of the WHO-China Joint Mission on Coronavirus Disease 2019 (COVID-19);. <https://www.who.int/docs/default-source/coronaviruse/who-china-joint-mission-oncovid-19-final-report.pdf>
- World Health Organization (WHO), 2020, Health topics, coronavirus. https://www.who.int/health-topics/coronavirus#tab=tab_3
- World Health Organization (WHO), 2022, WHO Coronavirus (COVID-19) Dashboard: <https://covid19.who.int/>
- Wu, R., Yan, S., Shan, Y., Dang, Q., Sun, G., 2015, Deep image: Scaling up image recognition. *arXiv*, 7(8). <https://doi.org/10.48550/arXiv.1501.02876>
- Wu, Z. & McGoogan, J.M., 2020, Characteristics of and Important Lessons From the Coronavirus Disease 2019 (COVID-19) Outbreak in China: Summary of a Report of 72 314 Cases From the Chinese Center for Disease Control and Prevention. *JAMA*, 323(13):1239-1242. <https://doi.org/10.1001/jama.2020.2648>
- Wynants, L., Van, Calster B., Collins, G.S., Riley, R.D., Heinze, G., Schuit, E., et al. 2020, Prediction models for diagnosis and prognosis of covid-19: systematic review and critical appraisal. *BMJ*, 7, 369, <http://doi.org/10.1136/bmj.m1328>
- Xu X., Jiang X., Ma C., Du P., Li X., Lv S., *et al.*, 2020, Deep learning system to screen coronavirus disease 2019 pneumonia, *arXiv*. <https://arxiv.org/abs/2002.09334>
- Zarei A., Beheshti H., Asl B. M., 2022, Detection of sleep apnea using deep neural networks and single-lead ECG signals. *Biomedical Signal Processing and Control*. 71(4):103125. <https://doi.org/10.1016/j.bspc.2021.103125>
- Zhang Q., Yang L.T., Chen Z., Li P., 2018, A survey on deep learning for big data, *Information Fusion*, 42:146-157. <https://doi.org/10.1016/j.inffus.2017.10.006>
- Zhang, J.J., Dong, X., Cao, Y.Y., Yuan, Y.D., Yang, Y.B., Yan, Y.Q., *ve ark.*, 2020. Clinical characteristics of 140 patients infected by SARS-CoV-2 in Wuhan, China. *Allergy*, 75(7):1730-1741. <https://doi.org/10.1111/all.14238>
- Zhou H., Ji, J., Chen, X., et al., 2021, Identification of novel bat coronaviruses sheds light on the evolutionary origins of SARS-CoV-2 and related viruses. *Cell*. 184 (17):4380–4391. <https://doi.org/10.1016/j.cell.2021.06.008>