

ANKARA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

DOKTORA TEZİ

DERİN ÖĞRENME İLE COVID-19, BEYAZ KAN HÜCRELERİ,
TÜBERKÜLOZ VE KALP KRİZİ RİSKİNİN ANALİZİ

Ömer SEVİNÇ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

ANKARA
2022

Her hakkı saklıdır

ÖZET

Doktora Tezi

DERİN ÖĞRENME İLE COVID-19, BEYAZ KAN HÜCRELERİ, TÜBERKÜLOZ VE KALP KRİZİ RİSKİNİN ANALİZİ

Ömer SEVİNÇ

Ankara Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. İman ASKERBEYLİ
Eş Danışman: Doç. Dr. Mehmet Serdar GÜZEL

Tıbbi görüntü verilerinin analizi, hastalıkların teşhisi ve buna bağlı olarak insan hayatı için kritik önem arz etmektedir. Son dönemde gerek akademik gerekse de tıp dünyasındaki potansiyeli sebebiyle röntgen, tomografi ve MR gibi görüntülerin ve hastanın tahlillerinden elde edilen metin tabanlı verilerin işlenerek, hastanın durumuyla ilgili tespit yapılması yapay zekâda en öne çıkmış çalışma alanlarından biridir. Bu tez çalışması kapsamında; röntgenler, mikroskopik kan hücreleri ve kalp krizi bulgularından hastalık teşhisi çalışmaları ve derin öğrenme teknikleri birleştirilerek optimize bir model üretilmiştir. Bu bağlamda çalışma boyunca üretilen model, hibrit bir modele dayalı yeni bir derin öğrenme tabanlı mimari sunmaktadır. Mimari temel olarak transfer öğrenme metodolojisinde otomatik kodlayıcı ve SVM sınıflandırıcı kullanılarak önceden eğitilmiş ağlara dayalı olarak tasarlanmıştır. Burada önerilen yöntemin temel katkıları; literatürde çoğunlukla kullanılan, önceden eğitilmiş modelleri optimize edilmiş bir şekilde birleştiren bir hibrit model önermesi ve ayrıca evrişimli sinir ağı üzerinde öznetelik iyileştirilmesi işlemi için yeni bir sezgisel algoritma modellemesi geliştirilmesidir. Kapsamlı veri setleri temel alınarak gerçekleştirilen performans karşılaştırmaları, önerilen mimarinin literatürdeki diğer gelişmiş mimarilerden daha başarılı sonuç verdiğini göstermiştir.

Haziran 2022, 109 sayfa

Anahtar Sözcükler: Derin Sinir Ağları, Derin Öğrenme, Öğrenme Transferi, Genetik Algoritma, Makina Öğrenmesi, Otomatik Kodlayıcılar, SVM

ABSTRACT

Ph. D. Thesis

ANALYSIS OF COVID-19, WHITE BLOOD CELLS, TUBERCULOSIS AND HAERT ATTACK RISK BY USING DEEP LEARNING

Ömer SEVİNÇ

Ankara University
Graduate School of Natural and Applied Sciences
Department of Computer Engineering

Supervisor: Prof. Dr. İman ASKERBEYLİ
Co-Supervisor: Assoc. Prof. Dr. Mehmet Serdar GÜZEL

The analysis of medical image and text data is critical for the diagnosis of diseases and, accordingly, for human life. In recent years, processing of images such as x-ray, tomography, MR and text-based data obtained from patients' analyzes and determination of the patient's condition is one of the most prominent areas in artificial intelligence due to its potential both in the academic and medical world. Within the scope of this thesis; An optimized model was produced by combining disease diagnostic approaches and deep learning methods from X-rays, microscopic blood cells and heart attack findings. In this context, the model provided in the study proposes a new deep learning-based architecture based on a hybrid model. The architecture is basically designed based on pre-trained networks using autoencoder and SVM classifier in transfer learning methodology. The essential contributions of the method proposed here are; Commonly used in the literature is a hybrid model proposition that combines pre-trained models in an optimized way, as well as the development of a new heuristic algorithm modeling for feature optimization operation on convolutional neural networks. Performance comparisons based on extensive datasets have shown that the proposed architecture outperforms the most advanced architectures in the literature.

June 2022, 109 pages

Keywords: Deep Neural Networks, Deep Learning, Transfer Learning, Genetic Algorithm, Machine Learning, Auto Encoders, SVM

ÖNSÖZ VE TEŞEKKÜR

Çalışmalarımı yönlendiren, araştırmalarımın her aşamasında bilgi, öneri ve yardımlarını esirgemeyen danışman hocam Prof. Dr. İman ASKERBEYLİ (Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı) ve eş danışmanım Doç. Dr. Mehmet Serdar GÜZEL'e (Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı),

Tez çalışmalarım süresince yapılan tez izleme komitesi toplantılarına katılarak fikir ve önerileriyle çalışmamın olgunlaşmasına yardımcı olan Doç. Dr. Recep ERYİĞİT (Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı) ve Dr. Öğretim Üyesi Mehmet DİKMEN' e (Başkent Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı),

Çalışmalarıma yön vererek çok değerli katkılar sağlayan Prof. Dr. Mehrübe MEHRÜBEOĞLU' na (Texas A&M Üniversitesi Bilgisayar Bilimleri Anabilim Dalı)

Sağladığı koşulsuz ve sınırsız manevi desteği için eşim Damla SEVİNÇ ve enerjisiyle bana her zaman güç veren oğlum Pamir Cem SEVİNÇ' e en içten teşekkürlerimi sunarım.

Ömer SEVİNÇ
Ankara, Haziran 2022

İÇİNDEKİLER

TEZ ONAY SAYFASI

ETİK.....	i
ÖZET.....	ii
ABSTRACT	iii
ÖNSÖZ VE TEŞEKKÜR.....	iv
SİMGELER DİZİNİ	vii
ŞEKİLLER DİZİNİ	ix
ÇİZELGELER DİZİNİ	xi
1. GİRİŞ	1
2. LİTERATÜR VE TEMEL BİLGİLER	11
2.1 Veri Kaynakları.....	13
2.2 Modelleme Yaklaşımı.....	14
2.3 Katkılar ve Anahat.....	16
2.4 Denetimli Öğrenme	17
2.4.1 Derin öğrenme	23
2.4.2 İyileştirme	25
2.4.3 Geri yayılım	28
2.5 Geleneksel Sinir Ağları.....	32
3. YÖNTEMLER	34
3.1 Evrişimli Sinir Ağları.....	34
3.2 Genetik Algoritma.....	38
3.3 Otomatik Kodlayıcı.....	42
3.3.1 Veri sıkıştırma	43
3.3.2 Özellik çıkarma	44
3.3.3 Görüntü oluşturma	44
3.4 SVM (Destek Vektör Makinaları)	44
3.4.1 SVM çekirdek işlevleri.....	47
3.5 Tekrarlayan Sinir Ağları	48
3.5.1 Uzun kısa süreli bellek LSTM.....	50
4. ÇALIŞMA VE BULGULAR.....	55

4.1 CNN ve Genetik Algoritma ile Beyaz Kan Hücrelerinin Sınıflandırılması	55
4.1.1 Lökositlerin tiplerine göre işaret ettikleri hastalıklar	60
4.1.2 CNN optimizasyonunun kullanılarak kromozom temsili.....	61
4.1.3 Veri ön işleme	61
4.1.4 Genetik algoritma.....	61
4.1.5 GA-geliştirilmiş D-CNN modeli ve sonuçları	62
4.2 Beyaz Kan Hücrelerinin Sınıflandırılması ve Covid-19 Teşhisinin Öğrenme Transferi ile Uygulanması	67
4.2.1 Öğrenme transferi.....	69
4.2.2 Önerilen model	72
4.2.3 Elde edilen sonuçlar	74
4.3 Akciger Röntgeninden Tüberküloz Teşhisi	79
4.4 Kalp Krizi Tahmini.....	81
5. SONUÇ.....	93
5.1 Gelecek Çalışmalar	94
KAYNAKLAR	96
ÖZGEÇMİŞ.....	107

SİMGELER DİZİNİ

$ \Psi p $	Ψp yamasının alan değeri
$*Pi$	Ankorun yer gerçeği etiketi (Ground-truth label)
$*ti$	Pozitif ankor ile ilişkili yer gerçeği etiketi
$\in i$	Gaussian gürültüsü değeri
$\perp$	Ortogonal (dikey) operatörü
s	Doğrusal olmayan sigmoid fonksiyonu
Z	Düzenleme fonksiyonu (Regularization function)
λc	Uzaysal frekans limiti (Spatial frequency cutoff)
Bi	Tespit için tahmin edilen kutu sınırlaması değeri
$C(p)$	p noktası merkezli Ψp yamasının güven değeri
$C(q)$	Önceden atanmış Ψp yamasının güven değeri
IRi	Örnek girdi dokunun (texture) yama değeri
$I\partial Ri$	Örnek girdi dokunun sınır bölgesi (boundary zone)
$Lreg$	Gerileme kaybı değeri
$W1$	Görünür düğümleri gizli düğümlere eşleyen bir ağırlık matrisi
$W2$	Görünür düğümü gizli düğümden yeniden yapılandıran ağırlık matrisi
Xi	Hedef bölgedeki normalleştirilmiş piksel konumları
$b1$	Gizli eğilim (bias) vektörü
$gi(z)$	$2 * n + 3$ boyutlu bir vektör
i	Mini yığın içerisindeki ankorun indeksi
k	Monotonik azalan kernel profili
np	p noktasında ön $\delta\Omega$ ortogonal olan birim vektörü değeri
pi	Ankorun tahmini olasılık değeri
$\hat{p}u(y^0)$	Taylor genişleme değerleri
qu	Hedefin renk öznitelik olasılığı
ti	Tahmini sınırlayıcı kutunun 4 parametrelili koordinatını temsil eden vektör
wi	Ağırlıklandırma veri değerleri
$x(i)$	Eğitim setinden rastgele örneklenmiş yerel yamaları
y	Çalışılan resim karesindeki lokasyon bilgisi
Ψp	Yama (patch) değeri

α	Büyültme faktörü değeri (Amplification factor)
βi	Normalize benzerlik ölçek faktörü (Normalized similarity scale factor)

Kısaltmalar

Acc	Doğruluk (Accuracy)
AE	Otomatik Kodlayıcı (Auto Encoder)
CNN	Evrişimli sinir ağı (Convolutional neural network)
CT	Bilgisayarlı Tomografi
DVM	Destek vektör makinası (Support vector machine)
GA	Genetik Algoritma
GPU	Grafik işlemci ünitesi (Graphics processing unit)
HOG	Yönelimli degradelerin histogramı (Histogram of oriented gradients)
KYM	Kırmızı, yeşil, mavi (Red green blue)
LSTM	Uzun Kısa Süreli Bellek (Long Short Time Memory)
mAP	Ortalama hassasiyet (Mean average precision)
MS	Ortalama Kayma (Mean Shift)
NIH	Ulusal Sağlık Enstitüsü
RNN	Tekrarlayan Sinir Ağları (Recurrent Neural Network)
SGD	Stokastik gradyan inişi (Stochastic gradient descent)
SVM	Destek vektör makinası (Support vector machine)
TB	Tüberküloz
X-Ray	X Işımları
YSA	Yapay Sinir Ağları

ŞEKİLLER DİZİNİ

Şekil 1.1 Nötrofil, eozinofil, monosit, lökosit, beyaz kan hücre yapıları	1
Şekil 1.2 Örnek X-Ray göğüs röntgeni	2
Şekil 1.3 Covid-19 olan (üst satır - 3 resim) ve olmayan göğüs (alt satır 3 resim) röntgenleri.....	3
Şekil 2.1 Sınıflandırma örnekleme	17
Şekil 2.2 Tipik denetimli bir öğrenme probleminde veri akışının şema sinir ağı.....	21
Şekil 2.3 Derin öğrenme modelinin makina öğreniminden farkı.....	24
Şekil 2.4 Hesaplamalı bir grafik boyunca geri yayılım örneği	30
Şekil 2.5 Tek bir nöronun arkasındaki biyolojik ilham ve yapay sinir ağı diyagramı	32
Şekil 3.1 Evrişimli sinir ağı basit model şeması	34
Şekil 3.2 Evrişimli ağ girdisi, filtre ve aktivasyon haritası örneği.....	35
Şekil 3.3 Genetik algoritma işlem süreçleri	40
Şekil 3.4 Otomatik kodlayıcı, kodlayıcı, kod ve kod çözücü mimarisi	43
Şekil 3.5 SVM destek vektörleri ve marjin	46
Şekil 3.6 RNN modelleri bire-bir, bire-çok, çoka-bir, çoka-çok	49
Şekil 3.7 RNN kullanımına bir dil modeli örneği	51
Şekil 4.1 a) Nötrofil b) Eozinofil c) Monosit d) Lenfosit	58
Şekil 4.2 Önerilen model şeması.....	59
Şekil 4.3 Test ve eğitim verileri hata grafiği.....	64
Şekil 4.4 Test ve eğitim verileri doğruluk grafiği	64
Şekil 4.5 Eğitim verileri karşılaştırma matrisi	65
Şekil 4.6 Test verileri karşılaştırma matrisi	65
Şekil 4.7 ILSVCR yarışmasında kullanılan modeller katman sayıları ve elde edilen hata oranları	68
Şekil 4.8 Resnet kısayol yaklaşımı	68
Şekil 4.9 Önerilen model Resnet50+AE+SVM veri kümesi	73
Şekil 4.10 COVID-19 X ışınları pozitif sonuçları	74
Şekil 4.11 Kan hücresi türleri	74

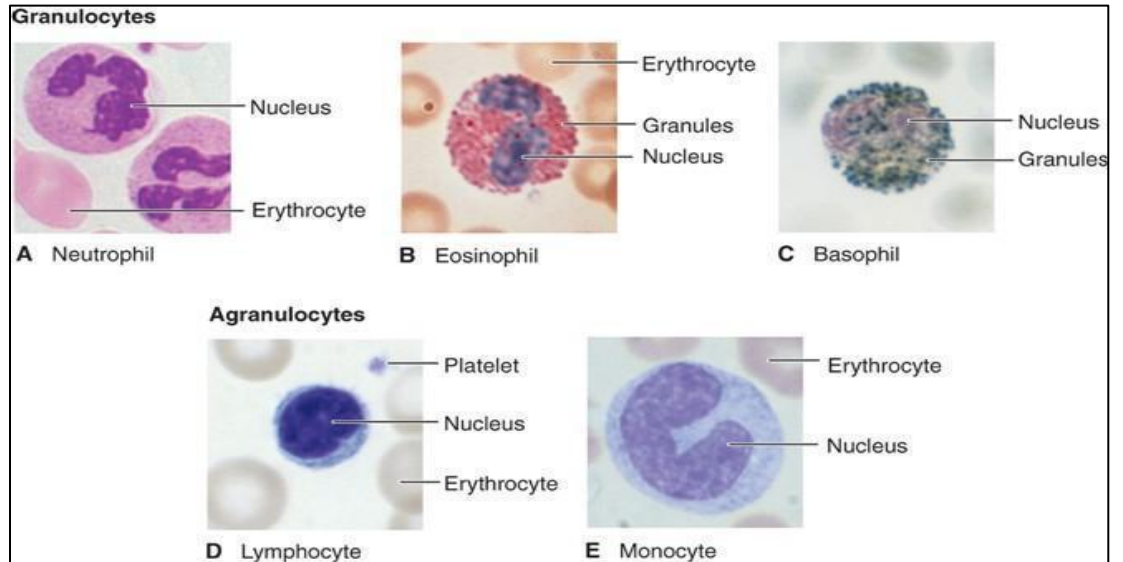
Şekil 4.12 Lökosit eğitim ve doğrulama veri seti için kayıp ve doğruluk sonuçları	76
Şekil 4.13 COVID-19 eğitim ve doğrulama veri seti için kayıp ve doğruluk sonuçları	76
Şekil 4.14 Lökosit karşılaştırma matrisi.....	77
Şekil 4.15 COVID-19 karışıklık matrisi	77
Şekil 4.16 Tüberküloz eğitim ve test veri seti kayıp değerleri.....	80
Şekil 4.17 Tüberküloz eğitim ve test veri seti doğruluk değerleri	80
Şekil 4.18 Tüberküloz veri seti için modelin karşılaştırma matrisi	81
Şekil 4.19 Kalp krizi tahmini için oluşturulan modelin şeması	83
Şekil 4.20 Kalp krizi eğitim ve test veri seti kayıp değerleri	85
Şekil 4.21 Kalp krizi eğitim ve test veri seti doğruluk değerleri	86
Şekil 4.22 Dinlenme kan basıncına göre kalp krizi riski dağılımı	87
Şekil 4.23 Kolesterol değerlerine göre kalp krizi riski dağılımı	87
Şekil 4.24 Hastanın yaşına göre kalp krizi riski dağılımı	88
Şekil 4.25 Hastanın ST eğrisi değerine göre kalp krizi riski dağılımı	88
Şekil 4.26 Hastanın EKG değerine göre Kalp Krizi Riski dağılımı	89
Şekil 4.27 Hastanın göğüs ağrısı tipine göre kalp krizi riski dağılımı	89
Şekil 4.28 Hastanın cinsiyetine göre kalp krizi riski dağılımı	90
Şekil 4.29 Hastanın egzersiz anjin tipine göre kalp krizi riski dağılımı	90
Şekil 4.30 Hastanın kalp krizi riski değerleri dağılımı	91
Şekil 4.31 Kalp krizi riski modeli ROC eğrisi	91

ÇİZELGELER DİZİNİ

Çizelge 4.1 Parametreler için kromozom genotipi.....	61
Çizelge 4.2 Parametreler için kromozom fenotipi	61
Çizelge 4.3 GA ile geliştirilmiş D-CNN modelinin katman sayısının belirlenmesi	63
Çizelge 4.4 GA destekli derin evrışimli ağ modelimizin diğer çalışmalarla kıyaslanması.....	66
Çizelge 4.5 Xception, VGG, Resnet, Inception, MobileNet ağırlık boyutları, parametreleri ve derinlikleri	70
Çizelge 4.6 Önerilen model ve diğer yöntemler için doğruluk sonuçlarının karşılaştırılması	78
Çizelge 4.7 Beyaz kan hücreleri için fl puanı ve diğer metrikler.....	78
Çizelge 4.8 COVID-19 için fl puanı ve diğer metrikler.....	78
Çizelge 4.9 Tüberküloz teşhisi için önerilen modelin performans verileri.....	81
Çizelge 4.10 Kalp krizi tahmini için kullanılan hasta bulguları.....	84
Çizelge 4.11 Kalp krizi verilerinin istatistiki dağılımını göstermektedir.....	84
Çizelge 4.12 Kalp krizi veri setinden çekilen örnek 5 veri.....	85

1. GİRİŞ

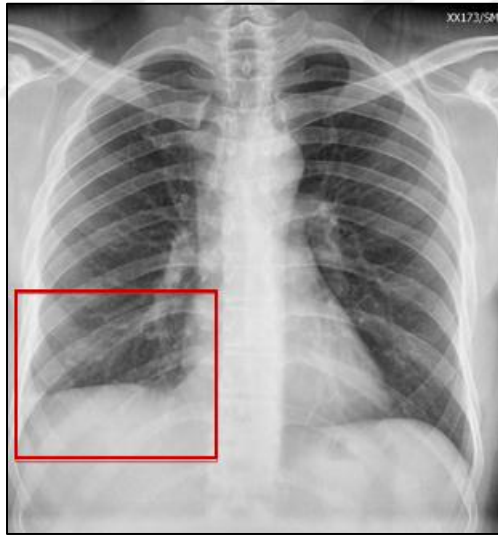
“Akıllı makinalar, özellikle de akıllı bilgisayar programları yapma bilim ve mühendisliğidir.” (McCarthy J. 1956). Yapay Zekâ, zeki insanların düşündüğü gibi bir bilgisayarı, bilgisayar kontrollü bir robotu veya bir yazılımı akıllıca düşünmesini sağlamanın bir yoludur. Yapay zekâ, insan beyninin nasıl düşündüğünü ve bir problemi çözmeye çalışırken insanların nasıl öğrendiğini, karar verdiğini ve çalıştığını inceleyerek ve daha sonra bu çalışmanın sonuçlarını akıllı yazılım ve sistemler geliştirmenin temeli olarak kullanır. Makinaların nihayetinde tamamen entelektüel alanlarda insanlarla rekabet edeceğini ümit edebiliriz (Turing A. 1950) Alan Turing’in bu sözlerine dayanarak yapay zekâ alanının hayallerinden birinin de bilgisayarların çevremizdeki zengin görsel dünyayı görmesini ve anlamasını sağlamak ve metin işlemede bizimle iletişim kurma yeteneğini oluşturmak olduğunu söyleyebiliriz. İnsanlar için karmaşık görsel tanıma içeren çok çeşitli görevleri yerine getirmek kolaydır. Aslında insan bir sahneyi tanımlarken, metin işlemede öğrendiği kavramları ve görsel öğelerle sahnedeki nesneleri tanımlamayı ve bütün bunları birleştirmeyi içeren görevleri yerine getirir. Öyle ki bir insanın görsel sahne hakkında çok fazla ayrıntıyı anlaması ve tanımlaması için bir görüntüye hızlıca bakması yeterlidir.



Şekil 1.1 Nötrofil, eozinofil, monosit, lökosit, beyaz kan hücre yapıları (Akramifard vd. 2012)

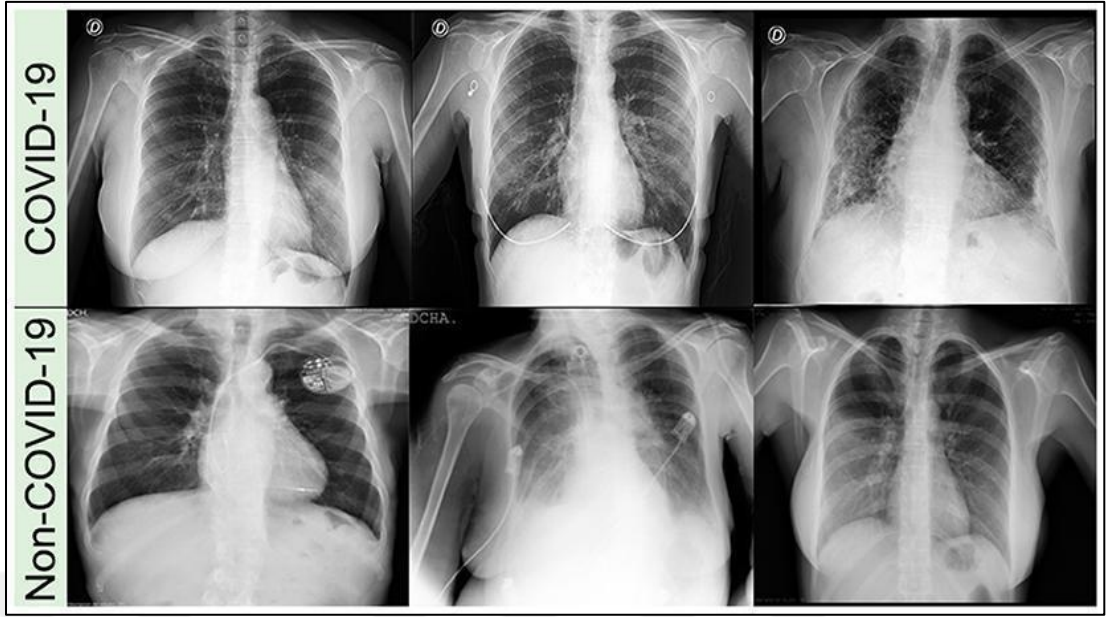
Şekil 1.1'deki örnekte, bir patolojist görüntüye bakarak çekirdek yapılarına, biçimlerine göre eozinofil, monosit, lökosit, nötrofil ve beyaz kan hücre yapılarını ayırt edebilir.

Bu yetenek bir uzman için çok normal ve kolay olabilirken bu görevin bir bilgisayar için ne kadar zor olduğu tahmin edilebilir. Bir bilgisayarda, bu görüntü bir matris olarak temsil edilir ve herhangi bir konumdaki piksel sayı dizilerinden oluşur. Sıradan bir görüntüde bu piksellerden birkaç milyon olabilir ve bir bilgisayar bu parlaklık değerleri desenlerini yüksek düzeye yani kavramlara dönüştürmelidir. Dahası, farklı kamera açısında ya da aydınlatma koşullarında parlaklık değerlerinin deseni tamamen farklı olabilir. Ya da tanımlanacak nesnenin farklı açılardan alınmış görüntüleri için yine desen temsil değerleri değişebilir. Ya da bunun yerine çok benzer hücre yapılarını birbirinden ayırt ederken birbirine yakın seviyeli istatistiklere sahip desenler zorluk oluşturabilir. Bilgisayar açısından ele aldığımız bu zorluklar dil anlama tarafında da benzer şekildedir.



Tani : Sağda plevral efuzyon tespit edildi.

Şekil 1.2 Örnek X-Ray göğüs röntgeni (Erdoğu ve Metin 2013)



Şekil 1.3 Covid-19 olan (üst satır - 3 resim) ve olmayan göğüs (alt satır 3 resim) röntgenleri (De Moura vd. 2021)

Görüntülerle birlikte açıklamalarının olması şüphesiz kavramlarla ilgili ek bilgiler elde edilmesini kolaylaştıracaktır. Bunlardan birisi resimdeki desenlerden bu bilgiyi çıkarmak iken diğeri de insanlar tarafından eklenen resmi tanımlayıcı açıklamalardan bilgi çıkarmak şeklinde olmaktadır. Örneğin Şekil 1.2’deki “Sağda plevral efüzyon tespit edildi” ifadesi bilgisayarda bir tamsayı dizisi olarak gösterilecektir. Bir görüntünün farklı kısımlarını işaret etmek ve adlandırmak yapay zekâ çalışmalarında çokça yapılan işlemlerdendir. Birkaç milyonluk bir matrisin belirgin alt kümelerini tanımlamak için karmaşık bir örüntü tanıma süreci gerekir. Yine Şekil 1.3 te COVID-19 belirtileri olan ve olmayan hastaların göğüs röntgenleri görülmektedir. Hastanın bulgularını içeren metinsel veri setlerinin analizi de hastalık teşhisi için çok büyük önem arz etmektedir. Hastanın yaş, boy, kilo, cinsiyet bilgileri, kan ve idrar tahlilinden, kalp EKG ölçümlerinden, ultrasonla elde edilen bulgular kanser, diyabet, kalp krizi gibi hastalıklar teşhis edilirken birer gösterge olarak kullanılmaktadır. Örneğin kalp krizi teşhisinde hastanın yaşı, cinsiyeti, göğüs ağrısı tipi (TA: Tipik Angina, ATA: Atipik Angina, NAP: Anjinal Olmayan Ağrı, ASY: Aseptomatik), dinlenme kan basıncı (mm Hg), serum kolesterolü (mm/dl), açlık kan şekeri (1: açlık BS > 120 mg/dl ise, 0: aksi halde), istirahat elektrokardiyogram sonuçları (Normal: Normal, ST: ST-T dalga anormalliğine sahip T dalgası inversiyonları ve/veya ST yükselmesi veya > 0.05 mV

depresyonu), ulařılan maksimum kalp atıř hızı (60 ile 202 arasındaki sayısal deęer), egzersize baęlı anjin (Y: Evet, N: Hayır), oldpeak ST (Depresyonda ölçülen sayısal deęer), zirve egzersiz ST segmentinin eğimi (Yukarı: eğimli, Düz: düz, Ařaęı: eğimli) gibi sayısal ve kategorik veri içeren metin veri seti kullanılmaktadır. Bu tezde saęlık alanındaki metinsel ifadeler genellikle dil işleme gerektirmeyen ancak metin verilerini özellikler olarak kullanarak sınıflandırma yapabilen modeller çalışılmıştır. Bu çalışmada görüntü işleme, metin veri setlerinden yapılan çıkarımlar ve hastalık teşhisleri ele alınmıştır. Metin veya görüntü veri setleri içeren örneklemeler üzerinde aynı tipte geliştirilmiş derin öğrenme modeli çalıştırılmış ve sonuçlar bu veri setleri için ayrıca ele alınmıştır.

Tıp alanında derin öğrenme çeşitli hastalıkların teşhislerinde görüntü analizi sonucunda doğru ve hızlı sonuçlar ürettięi için yaygın olarak kullanılmaya başlanmıştır. Bu hastalık ve analizlerin öne çıkanlarından bazıları ele alınmış olup derin öğrenmenin nasıl uygulandığı açıklanmıştır. Örneęin; tüberküloz (TB), dünyadaki çoęu ülke için önemli bir saęlık sorunudur. Tüberküloz, her yıl 1,5 milyon ölüm ve her yıl bildirilen 10 milyon yeni vaka ile dünya genelinde beşinci önde gelen ölüm nedenidir (Pasa vd. 2019). Dünya Saęlık Örgütü, hastalığın yönetimini kolaylařtırmak için geniş ve sistematik tarama önermektedir. Tüberküloz gibi akcięer hastalıklarını izlemek ve teşhis etmek için kullanılan en yaygın ölçü, uygun fiyatlı ve kolay erişilebilir oldukları için göęüs röntgenidir (CXR). Derin öğrenme gibi modern teknolojilerin gelişimi ve evrimi, bilim adamlarının ve radyologların röntgen görüntülerinden akcięer hastalıklarını otomatik olarak tespit etmesine olanak tanır (Wang vd. 2017). Derin öğrenme, etiketlenmemiş, yapılandırılmamış ve denetlenmeyen verileri öğrenebilen aęa sahip bir YSA' ya dayalı bir makina öğrenimi ailesinin parçasıdır. Göęüs Röntgeni görüntülerinden tüberkülozu tahmin etmek ve ölümleri azaltmak için, evriřimli Sinir Aęları (CNN) gibi derin öğrenme yöntemleri kullanılabilir (Gozes ve Greenspan 2019). Bu çalışmada CNN, RNN ve SVM gibi derin öğrenme tekniklerini ve göęüs röntgeni görüntülerini kullanarak tüberküloz tahmini yapılmıştır. Tüberküloz taramasında kullanılan en yaygın ölçü röntgendir. Bununla birlikte, kullanımlar uzun ve karmařık bir süreci içerir. Özellikle gelişmekte olan ülkelerin çoęu, röntgenden elde edilen sonuçları yorumlamada yeterli radyoloęa sahip deęildir (Stirenko vd. 2018). Benzer řekilde,

Lakhani ve Sundaram (2017), dünyanın birçok bölgesinde tüberkülozu yorumlamak için gerekli olan radyoloji uzmanlığının eksikliğinin olduğunu ve bunun da sonuç olarak çalışma çabalarını ve tarama çabalarını olumsuz etkilediğini söylemektedir. Derin öğrenme teknolojisi, bilgisayarların, insanların ilişkileri incelemek için mevcut verileri kullanma gibi görevleri yerine getirmelerine yardımcı olabileceği yapay zekâyı tanımlar (Glorot ve Bengio 2010). Böylece derin öğrenme teknikleri, tüberküloz tahmininde doğruluğu artırdıkları ve uzmanlara olan bağımlılığı azalttığı için etkin bir çözüm sağlar (Sathitratanacheewin ve Pongpirul 2018).

Ayrıca, tüberküloz geliştirme riski yüksek kişilerin belirlenmesinde, kullanımı kolay ve tahminde bulunmada doğru olan görüntüleme tekniklerine duyulan ihtiyaç derin öğrenmenin tıp ve insan sağlığı için önemini ortaya koymaktadır. Görüntüleme, esas olarak, hastalar arasında akciğer tüberkülozu olduğundan şüphelenilen kişiler için önemli bir tarama önlemidir (Lakhani ve Sundaram 2017). Bu nedenle, uygun maliyetli ve etkili otomatikleştirilmiş yöntemler olan derin öğrenme teknikleri, tüberkülozun taranmasına ve erken teşhisine yardımcı olur ve kolaylaştırır.

Yine 2019'un sonunda Wuhan şehrinde (Hubei, Çin) nedeni bilinmeyen birçok zatürre vakası ortaya çıktı (Lu vd. 2020). Birkaç gün sonra, neden olduğu virüs tespit edildi ve Uluslararası Virüs Taksonomisi Komitesi, virüsü “Şiddetli Akut Solunum Sendromu Coronavirüs 2” (SARS-CoV-2) (Rabi vd. 2020) olarak adlandırdı. Koronavirüsler, Coronaviridae alt ailesinin RNA virüsüdür ve dört ana alt grubu (α , β , γ ve δ) vardır (Cui vd. 2019). İnsanları enfekte edebilen sadece yedi koronavirüs vardır ve SARS-CoV-2 yedincisidir.

SARS-CoV-2'nin neden olduğu koronavirüs hastalığı (COVID-19) oldukça bulaşıcı bir hastalıktır ve Dünya Sağlık Örgütü 11 Mart 2020'de bu virus sebebiyle bir pandemi ilan etmiştir (Andersen vd. 2020). Ters transkripsiyon polimeraz zincir reaksiyonu, SARS-CoV-2'yi tespit etmek için yaygın olarak kullanılan bir testtir. Ancak bu test maliyetli, manuel, karmaşık ve zaman alıcıdır. COVID-19'u tespit etmenin diğer yolları klinik belirti muayenesi, patoloji testleri ve akciğer röntgenidir (Lu 2020).

Sağlık alanında hastalardan elde edilen görüntüler ve hastayı tanımlamak için kullanılan ifadeler derin öğrenme için iyi birer veri kaynağı olmaktadır. Röntgen ya da hasta verilerini okumak zaman ve uzmanlık gerektirirken, derin öğrenme eldeki görüntülerden teşhislere olan bağlantıyı öğrendiği için sonraki örneklemelerde kusursuza yakın teşhis ve sınıflandırma yapabilmektedir. Örneğin beyaz kan hücrelerinin mikroskopik görüntülerinden doğru sınıflandırma ve sayımının yapılarak vücutta bir hastalığa karşı antikor üretilip üretilmediği, değerlerin normalin altında ya da üstünde olması teşhis edilebilmektedir. Görüntülerin işlenmesinde görüntülerin elde edilmesinden, işlenip ölçeklenerek özelliklerinin çıkarılması süreçlerinin zorlu görevler olduğu aşikârdır, ancak derin öğrenme alanındaki gelişmeler ile bu işlemler çok daha kolay ve hızlı şekilde yapılabilmektedir. Evrişimli sinir ağları ile (CNN), son zamanlarda hızlı görsel tanıma alanında çok iyi düzeyde ilerlemeler kaydedilmiştir. Özellikle son görüntü tanıma modelleri derin evrişimli sinir ağlarına dayanan (Kaiming He vd. 2014) çalışmalar görüntüleri yüksek doğruluklarda karşılaştırabilmekte ve kategorilere sınıflandırmaktadır (Krizhevsky vd. 2012). Bölümlere ayrıştırma ve nesne tespitinde de benzer şekilde çok başarılı sonuçlar elde edilmiştir (Sermanet 2014). Bu gelişmelerle birlikte yapay zekâ kendisini birçok yüz tanıma, kişisel fotoğraf arama, hastalık teşhisi gibi gerçek dünya uygulamalarında etkin şekilde göstermiştir.

Bu uygulamaların çoğunda baskın yaklaşım görsel tanıma problemini gerçekleştirmek, görüntüleri önceden belirlenmiş kategorilere sınıflandırmaktadır. Örneğin, ImageNet organizatörler tarafından manuel olarak seçilmiş kategorilerde sınıflandırma yapılabilmesine imkân tanır. Benzer şekilde, PASCAL VOC (Fang vd. 2015) ile nesne algılama karşılaştırması için manuel olarak seçilen 20 farklı kategoride sınıflandırma yapılabilir. Medikal veri seti (MNIST Medikal), 64x64 boyutunda basit tıbbi görüntülerdir; orjinalleri ilgili veri kümelerinden alınmış ve işlenmiştir. 6 sınıfa ait 58954 medikal görsel bulunmaktadır. Kaggle üzerinde röntgen görüntüleri, mikroskopik kan örneklem resimleri, iskelet tomografi görüntüleri gibi birçok medikal veri kaynağı analiz edilerek hastalıkların teşhisini yapan ve sınıflandıran yapay zekâ modelleri geliştirilmektedir. Diğer veri setleri metin sınıflandırması, nitelik sınıflandırması için eldeki metin verilerini kullanarak kendi kategorilerinde sınıflandırmalar yaparlar.

Belirlenmiş görsel kategoriler için ideal yapay zekâ modelleme varsayımları başarılı şekilde oluşturulabilmektedir.

Bu tezde de bilgisayar görme görevleri ve metin verisi için evrişimli sinir ağları ile tekrarlayan yapay sinir ağları ele alınmış ve bu teknikler üzerinden modeller geliştirilmiştir. Bunlardan bir tanesi evrişimli yapay sinir ağlarını kullanarak beyaz kan hücrelerini sınıflandırmak iken bir diğerinde hasta verilerinden kalp krizi tespitinde evrişimli sinir ağları ile metin işleme tekniklerini birleştirerek sınıflandırma yapmaktır. Başka bir deyişle, model veri setlerine göre düzenlenerek çoğunlukla resim tabanlı ve ek olarak da metin tabanlı çıkarım yapılması hedeflenmiştir.

Son yıllarda, makina öğrenimi uygulamalarının gerçek dünya sorunlarına, özellikle derin öğrenme ile birlikte önemli oranda çözüm ürettiği görülmüştür (Schmidhuber, 2015). Derin öğrenme modellerinden biri olan Evrişimli sinir ağları (CNN), bilgisayarlı görme ve konuşma tanıma gibi alanlarda kullanılan en etkili ve popüler araçlardan biri olduğunu kanıtlamaktadır.

Evrişimli sinir ağlarının, iyi çözümler bulma konusundaki olağanüstü yetenekleri ve gürültülü ve belirsiz verileri tolere edebilme özelliği ile birçok sorunu çözmek için aday olmaya başlamıştır. Buna ek olarak, büyük boyutlu veri kümelerinde çalışmak için CNN uygulanması, değiştirilmesi ve test edilmesi Python gibi üst düzey programlama dilleri ve Keras gibi yardımcı kütüphaneler sayesinde daha kolay ve kullanışlı hale gelmektedir (Chollet 2016).

CNN nöronlar adı verilen, katmanlı bir şekilde konumlandırılmış ve ağırlıklı bağlantılar kullanarak birbirleriyle etkileşime giren basit işleme elemanlarından oluşan biyolojik olarak ilham verici hesaplama sistemi olan özel bir ileri beslemeli yapay sinir ağıdır (Henry 1992). Bir CNN mimarisi iki ana bölümden oluşur, birincisi özellik çıkarma ve ikincisi sınıflandırmadır. Birincisi, her biri gerçek değerleri koruyan probleme özgü boyutun bir tensörü olan bir dizi filtre içerir ve ikincisi aslında tamamen bağlı bir ileri beslemeli yapay sinir ağıdır.

Bu tezdeki hedef, kısa vadede değerli uygulamalar sunmak iken uzun vade de akıllı makinalara doğru yol almakta katkı sağlamaktır. Öncelikle, YSA teknikleri, özellikle metin işlemede bilgisayarlarla etkileşime girebileceğimiz geleceğe doğru bir adımdır. Ayrıca, yapay zekâlı ajanlara yönelik çalışmak bilgisayarlara dünyamızın nasıl çalıştığı hakkında büyük miktarda bilgi sağlayacaktır. Somut olarak, üzerinde kullanabileceğimiz iki büyük bilgi kaynağı vardır. Örneğin sağlık alanında olup bitenler hakkında bilgi içeren görüntüler, olaylar, nesneler ve etkileşimleri içeren fiziksel bilgiler birincil kaynaktır. Diğer taraftan ise ikincil kaynak olarak, yalnızca fiziksel alandan çıkarılamayan bilgiler ki bu bilgileri işleyebileceğimiz çok miktarda semantik içeren internetin dijital alanındaki bilgilerdir (Chen vd. 2016, Mikolov vd. 2013). Görüntüler yapay zekâ modellerinde en yoğun kullanılan veri kaynaklarından biridir. Sağlık bilgisine ulaşılabilir teknikler geliştirmek çok önemlidir ve her biri bağımsız olarak ya da ilişkilendirilerek işlenebilir. Bu tezde akciğer röntgenlerinden hastalık tespitlerinde görüntü işleme ile kalp krizi teşhisinde metin veri setleri kullanılarak da, derin öğrenme ile başarılı sonuçlar elde edilmiştir. Daha ileri bir gelecekte, bilgisayarlar teşhisleri veya geçmişteki tanıları okuyabilir ve daha önce anlaşılamamış ve gizemli kalmış olayların temellerini anlayabilir. Ayrıca yalnızca metinde belirtilmeyen kritik çıkarımların görüntüler ile sadece görüntülerde olmayan çıkarımların metinler ile kilidi açabilir.

Kısa vadeli motivasyonlar için ise görüntü veya metin işleme kullanılarak hastalık teşhislerinin hızlı ve doğru yapılabilmesi için etkin pratik çözümler sağlanacaktır. Nesne tanıma, sahne sınıflandırması, öznitelik tanımlama bilgisayarla görmede ayrı alanlar olarak ele alınıp incelenmekte ve bu alanlarda sınıflandırma, eylem tanıma gibi kapsamlı ve önemli çalışmalar yapılmaktadır. Ek olarak, metin veriler dil işleme üzerine veriler olmasının yanında kategorik ya da rakamsal değerlerden de oluşabilir. Örneğin, bu tezde çalışılan kalp krizi belirtileri 13 farklı metin verisi içermekte ve bunlar özellikler olarak evrişimli ağ modeline girdi olarak verildiğinde hasta için kalp krizi riski tahmini yapmaktadır. Bu çalışmada amaç, belirlenen özelliklerin kalp krizi tanısı ile ilişkisini incelemek ve kalp krizi tanısını en yüksek düzeyde doğru olarak tahmin etmektir. Tezde üzerinde çalıştığımız 13 farklı kalp krizi belirtisi, metin veri setlerinde tanımlanan göğüs ağrısı tipi, egzersize bağlı anjin ve egzersize bağlı ST depresyonu gibi

nitelikler evrişimli ağ üzerinden metin işleme ile kalp krizi tanısını ortaya koyarak vaka verilerine algoritmik ve istatistiksel yöntemler uygulanarak tespit edilmiştir. Resim ve metin verilerinin birlikte ele alınması mevcut yaklaşımlardan biri olup, bu yaklaşım bazı zorlukları da beraberinde getirir. Bir teşhis cümlesi tanımlandığında bu cümlenin görüntü ile ne kadar tutarlı ve ne kadar doğru olduğunu değerlendirmek daha zor olabilir. Uzmanlar tarafından yazılmış bazı referans cümleler ile her bir tahmin karşılaştırılabilir, ancak bu karşılaştırmanın açıkça tanımlanması da zor olabilir. Dolayısı ile metin üzerinde ve görüntüler üzerinde ayrıca çalışarak her iki veri seti için sınıflandırma yapan modeller daha pratik ve doğruluğu daha yüksek sonuç verecek şekilde uygulanabilmektedir. Bundan ötürü çalışmalardaki motivasyon, sağlık alanı için kritik önem arz eden yüksek doğrulukla teşhis yapılabilmesi olduğundan görüntü ve metin tabanlı analizler ayrı şekilde ele alınmıştır. Günümüzde insan değerlendirmelerinin de katkısı ile en yüksek bağlantılar kurulabilmekte ve en gelişmiş otomatik değerlendirme yöntemleri kullanılmaktadır. Ancak bunların gelecekte daha da iyileştirilmesi ile ilerleme kaydedileceği beklenmektedir.

Tez çalışmasında kullanılan modeller resim veri setlerini ayrı ele alarak sınıflandırmayı amaçlamaktadır. Bu veri setleri ele alınırken model üzerinde ufak değişiklikler yapılması yeterli olmaktadır. Öte yandan, bu görevlerin ele alınması, görüntü işlenirken hangi özelliklerin çıkarılması gerektiğini açıkça belirtmek zorunda kalmadan, metin ve görüntüleri eğitim sırasında tüm ara temsillerini otomatik olarak keşfeden tek bir model ile formüle edilebilmesini sağlamaktadır. Bu tezde, görüntüleri sınıflandırmak, hiper parametrelerin iyileştirilmesi ve metin işleme için modeller geliştirilmiştir. Ayrıca, böylelikle medikal görüntüler üzerinden tanı koyacak yeni bir model önerilmiştir. İlgili modelin parametreleri meta sezgisel yöntemler ile en uygun değerlere yaklaştırılmıştır. Özetle, sağlık alanında görüntüleri işleyip, sınıflandırmak için derin öğrenme modelleri ve modelin iyileştirilmesi için genetik algoritma gibi iyileştirme teknikleri uygulanmış ve ayrıntıları ile açıklanmıştır. Diğer taraftan hem görüntü hem de metin işlemeyi ele alarak görüntü veri kümelerinde uçtan uca eğitilen sinir ağı mimarileri uygulanmış ve açıklanmıştır.

2. bölümde, denetimli öğrenme, geri besleme, iyileştirme ve sinir ağları için ilgili matematiksel altyapı anlatılarak, özellikle evrişimli (CNN), ve 3. bölümde metni işlemek için yaygın olarak kullanılan tekrarlayan sinir (RNN) ağları gibi mimari tasarım modelleri açıklanmıştır. 4. bölümde doktora süresince yapılan çalışmalar ele alınmış olup; beyaz kan hücreleri görüntülerinin sınıflandırılması ve modelin hiper parametre iyileştirilmesi ile nasıl daha etkin olabildiği anlatılmıştır. Bunun yanında X ışınları görüntülerinden COVID-19 ve tüberküloz teşhisi yapan CNN modelleri geliştirilmiş ve modeller SVM sınıflandırıcı ve otomatik kodlayıcı ile desteklenmiştir. Yine tez konusunu temel alan metin tabanlı çıkarım ve tahmini için kalp krizini hasta bulgularından analiz ederek doğru tahminler yapabilen model CNN ile geliştirilip, sonuçlardan elde edilen bulgular ve detayları ele alınmıştır.

2. LİTERATÜR VE TEMEL BİLGİLER

Sağlık alanında ve diğer alanlarda görüntü işleme üzerine yapılmış birçok çalışma vardır. Bu tezde, görüntü işleme ve metin işleme modelleri incelenmiş ve ilgili çalışmalar ortaya konulmuştur. İlk çalışmalardan bazıları, ortak oluşum istatistiklerinden yararlanan ve istatistiksel modellere dayanan isimlerin görüntü bölgesi özellikleriyle ilişkilendirilmesi sorununa odaklanmıştır (Kiros vd. 2015, (Kong C. vd. 2014) Beyaz kan hücreleri çekirdek segmentasyonunda CNN modeli kullanılmış ve %96 başarı elde edilmiştir (Banik P vd. 2020). Yine Habibzadeh vd. (2013) evrişimli sinir ağları modeli kullanarak düşük çözünürlüklü beyaz kan hücreleri görüntülerinden alt grupların sayımını yapmıştır. ‘Tiny Yolo’ evrişimli sinir ağı kullanılarak kan hücre sayımı yapılmıştır (Sharon ve Therse 2018) . COVID-19 ve zatürre hastalarının göğüs röntgenleri ve bulgularını incelenmiş antimikrobiyal kullanımındaki artışı tespit etmiştir (De Moura vd. 2021). Bayes iyileştirmesi ile iyileştirilmiş ‘SqueezeNet’ mimarisi kullanılarak göğüs röntgenlerinden COVID-19 teşhisi yapılmış %98 doğru sınıflandırma başarısı göstermiştir (Ucar ve Korkmaz 2020). Bir başka çalışmada kalp krizi riski tahmini için evrişimli sinir ağları modeli geliştirilmiş ve sonuçta %86 doğruluk oranında başarı elde edilmiştir (Shankar vd. 2020). Kalp EKG verilerinin CNN ve LSTM modeli üzerinde eğitilmesi sonucunda ise %97 doğrulukla kalp krizi riski tahmini yapılmıştır (Petmezas vd. 2021).

Derin öğrenme, klinisyenlerin röntgenlerden tüberküloz varlığı hakkında sınıflandırma yeteneklerini geliştirir. CNN ve RNN gibi yöntemler, doğru sonuçlar için büyük veri setlerinin mevcudiyetine bağlıdır. Ne yazık ki, Ulusal Sağlık Örgütü tarafından sağlanan yalnızca birkaç büyük ölçekli tıbbi veri seti bulunmaktadır. Ayrıca, profesyonel eğitim gerektirdiğinden, kitle kaynak kullanımı gibi stratejilerin uygulanması tıbbi görüntülerin toplanmasında kullanılamaz.

Evrişimli sinir ağlarının kullanımıyla ilgili bir detay, bazıları denetlenen ve bazıları denetlenmeyen mevcut geniş verilerle doğru görüntüleme sonuçları üretme yeteneğidir (Hsu vd. 2018). Etiketsiz büyük görüntüleme veri kümelerinde derin öğrenmenin gerçekleştirilip gerçekleştirilemeyeceğini raporlamak için araştırılmıştır. Bunun yanı sıra

COVID-19'un erken evresinde radyografik görüntülerin çoğunda hem akciğer tutulumu hem çok odaklı buzlu cam opasiteleri incelenmiştir (Salehi vd. 2020, Huang vd. 2020, Zhang vd. 2020). COVID-19 (Shrock vd. 2020), tıbbi görüntüleme yöntemleri arasında ilk tercih olarak röntgeni tercih etmiş ve araştırmalarında, röntgen şiddet skoru üzerinden çıkarımlar yapılmıştır.

Otomatik tanı koymada makina öğrenmesi tabanlı çözümler popülerlik kazanmış ve araştırmacılar bunları birçok sağlık sorununda kullanmaktadır (Waring vd. 2020). Özellikle görüntü işleme alanında önceden eğitilmiş CNN modeli ResNet (Szegedy vd. 2017) o kadar etkili ve verimli oldu ki yarışmalarda öne çıkan modellerden biri olduğunu kanıtladı. Derin CNN modelleri ile görüntülerin katmanlar aracılığıyla elde edilen kısımları etkin bir şekilde öğrenilir ve ayrıntılardan çıkarımlar yapılır (Sun vd. 2019). Böylece model, incelenen görüntüleri (Zeiler ve Fergus 2014) doğru bir şekilde sınıflandırabilir. Modeldeki artımlı katman sayıları, teoride görüntünün özelliklerini temsil etme yeteneğini geliştirir. Bu nedenle, daha fazla katmanın potansiyel olarak başarılı bir şekilde çalışabilen gelişmiş ağ mimarileriyle sonuçlanabileceği sonucuna varılabilir.

Diğer çalışmalar, benzer kıyaslama veri kümelerini kullanan önerilen modellerle karşılaştırılmıştır. Karşılaştırılan modeller, kan hücrelerini ve X ışınlarını sınıflandırmak için hibrit CNN ve RNN modelleriyle Kaggle'daki beyaz kan hücresi veri setini ve göğüs röntgenlerini kullanır (Kansara vd. 2018, Habibzadeh vd. 2013, Horne vd. 2005, Sharon ve Therse 2018).

Ek olarak, önceden eğitilmiş CNN mimarileri, öğrenme transferi yoluyla tıbbi görüntüler üzerinde eğitilmiştir. MRI ve PET'i sınıflandırmak için önerilen model, %90'ın üzerinde bir doğruluk oranı başardı (Liang vd. 2018). Önerilen yönteminde divNet mimarisini kullandılar. Sıradan bir makina öğrenimi algoritmasına dayalı olarak yapılan başka bir sınıflandırma, bir hastanın Alzheimer hastalığı olup olmadığıyla ilgili özelliklerden tanımlamayı içeriyordu. SqueezeNet modeli ile yaklaşık %98 doğrulukla CT görüntülerinden COVID-19 sınıflandırması için bir model kabul edilebileceği öne sürülmüştür (Ucar ve Korkmaz 2020). Özel CNN modelleri tıbbi görüntüler üzerinde

çalıştırılmış ve yaklaşık %94 doğruluk sonuçları rapor edilmiştir (Ben Atitallah vd. 2022). COVID-19'u %98 doğrulukla sınıflandırmak için bir CNN-LSTM modeli kullanılmıştır. Yöntem, sınıflandırma başarı oranını artırmak için LSTM ile COVID-19'u teşhis etmek için CNN'den çıkarılan öznitelikleri kullanmıştır (Islam vd. 2020). Akciğer CT görüntülerinde nodül tespiti için yapılan başka bir çalışmada, CNN modelleri nodül tespiti için tasarlanmış ve yaklaşık %97 doğruluk elde edilmiştir (Su vd. 2021). Son olarak, Hamidian vd. (2020) COVID-19 teşhisi için CNN-AE kullanmış ve %96.05 doğruluk bildirmiştir. Bu tez çalışmasında önerilen model, ResNet-50 üzerinde otomatik kodlayıcı ve SVM sınıflandırıcıyı birleştirir ve hem beyaz kan hücreleri hem de göğüs röntgeni görüntülerinde daha yüksek doğrulukta sonuçlar üretmek için öğrenme transferi yönteminden faydalanılmıştır.

Bu tezde hem metin hem de görüntü sınıflandırılması görevleri ele alınmaktadır, derin öğrenme ile üstün sonuçlar sunan yeni teknikler ve modeller geliştirilmiştir. Görüntülerin ve metinlerin derin öğrenme modelleri ile nasıl başarılı şekilde sınıflandırılabilirdiği, dikkat edilmesi gereken parametre tanımlamaları, iyileştirmeleri ve boyut indirgeme ele alınmıştır. Ayrıca bu tezde ön eğitilmiş CNN modelleri ele alınmış olup veri setlerine göre uyarlanan çalışmalar ile anlatılmıştır. Yapılan çalışmalar bir görüntünün tüm belirgin bölgelerini algılamak, tanımlamak ve sınıflandırmak ve yine metin sınıflandırılması için de CNN modelinin başarılı sonuçları verdiğini göstermiştir.

2.1 Veri Kaynakları

Bu alandaki modelleme yaklaşımlarının büyük çoğunluğu, modelin elde edilen görüntü verilerinden öğrendiği yöntemleri içermektedir. Bu nedenle, mevcut veri kümelerini vurgulamak ve istatistiklerini dikkatlice incelemek önemlidir. Sağlık alanında birçok görsel ve metin veri kümesi vardır. Kaggle, NIH üzerinde beyaz kan hücreleri mikroskopik görüntüleri, tüberküloz, zatürre ve Covid-19 teşhisi için göğüs röntgenleri, kalp krizi öngörüsü için metin veri kümelerini kullanılmıştır (Mooney 2018). Örneğin beyaz kan hücreleri görüntüleri monosit, lökosit, lenfosit, eozinofil gibi alt hücre gruplarını içermekte iken Covid-19 X ışınları görüntüleri kişinin akciğer detaylarını içermektedir (Sivaramakrishnan vd. 2018). Yine tüberküloz ve zatürre için akciğer

görüntüleri teşhis için belirleyici veriyi içermektedir (Li vd. 2020). Kalp krizi teşhisinde hastanın yaş, boy, kilo gibi bilgileri ile hastadan elde edilen bulguların bir arada olduğu metin veri seti kalp krizi riskini belirlemede kullanılmaktadır. İnsanların kemik grafisinden elde edilen kafatası, bilek, el bölümünden kimlik, cinsiyet ve yaş teşhisi yapılabilmekte ve adli vakalarda kimlik tespitinde kullanılabilmektedir. Diyabet hastalarının bulguları ve tanım bilgilerini içeren metin veri setlerinden diyabet tipi, seviyesi ve risk grubu belirlenebilmektedir. Yine tomoğrafi görüntülerinde akciğer kanseri tanısı yapılabilmekte ve hatta 3 boyutlu görüntülerden elde edilen kesitler CNN ile sınıflandırılarak teşhise yardımcı bulgular yapay zekâ yöntemleri ile elde edilebilmektedir (Sivari vd. 2022).

2.2 Modelleme Yaklaşımı

Bu alanda yapılan ve üzerinde tartışılan çalışmalarda modelleme yaklaşımlarının çoğu, özellik ayıklama, nesne algılama ve nitelik sınıflandırması adımlarından oluşur. Bu yaklaşımın dezavantajları, modelin hızlı bir şekilde karmaşık hale gelmesidir. Bu dezavantajlar ön işleme işlemleri, ara hesaplamalar, ara katmanlar oluşturulması ve çok sayıda hiper parametre olmasıdır. Nesne, sahne veya öznitelik algılayıcıları da genellikle kendi istatistikleri ve yanlılıkları (bias) olan farklı veri kümeleri üzerinde eğitilir ve nihai olarak ilgilenilen etki alanına aktarım sınırlıdır (Lin vd. 2014). Son zamanlarda, büyük ölçekli veri kümelerinin kullanılabilirliği ve daha hızlı hesaplama nedeniyle derin evrişimli sinir ağlarının kullanımı ile görüntü tanıma hızı ileriye taşınmıştır (Vedantam vd. 2015). Bu modeller, 1940'lı yıllardan başlayarak geliştirilen bir dizi tekniğe dayanmaktadır. “Sibernetik” adı altında, (Rosenblatt, 1958) ayarlanabilir sinaptik güçlü nöronlar ve öğrenme kuralları içeren modelleri geliştirilmiştir. 1980'lerde “bağlantılık” ve “paralel dağıtılmış işleme” adı, (Rumelhart vd. 1986) bununla kullanılan nöronların eğitim ağı için geriye yayılım kullanımını yaygınlaştırmıştır. Zincir kuralının özyinelemeli uygulaması ve kayıp fonksiyonunun parametrelere göre gradyanının belirlenmesine dayanan son derece etkili bir kredi tahsis algoritması geliştirilmiştir. Özellikle, görsel olarak bu tekniklerin ilk başarılı uygulamalarından biri olan benzeyen rakam tanıma için (LeCun vd. 2015), kafes yapısında nöronlar ve uzayda yerel bir bağlantı düzenlenmiştir. Bu modern evrişimli

sinir ağıının ilk açıklamasıydı, ancak büyük çaptaki başarısı önemli ölçüde daha fazla hesaplama ve eğitim verisi ile derin öğrenme görsel tanıma problemlerini çözmesi ile (ImageNet) yıllar sonra mümkün olmuştur. Evrişimli ağlar, bir girdi alanından bir çıktı alanına bir işlevi tanımlar ve bu işlevin parametreleri tüm nöronların bağlantılarının güçlü yönleri geniş bir etiketli görüntü koleksiyonu kullanılarak eğitilir. Bu yaklaşımda model “uçtan uca”, yani girdilerden çıktılara kadar optimize edilir ve tüm hesaplama süreci aynı son hedefi paylaşır. Bu yaklaşım yaygın olarak kabul edilmektedir ki bu da modellerin pratikte çok iyi çalışmasını sağlar. Bu modellerin avantajı, tek bir veri seti üzerinde eğitildikten sonra sabit bir şekilde kullanılabilmesidir. Derin öğrenme modelleri aynı zamanda görüntüler için özellik çıkarıcıdır, ancak daha da önemlisi farklı bir veri kümesindeki parametreleri iyileştirerek nihai performansta önemli gelişmeler sağlar. Bu noktadan hareketle çalışmalarda evrişimli sinir ağları kullanıldı.

Görüntüler için bir çekirdek işleme kısmı olan yapay sinir ağları, onları daha büyük sinirsel ağ mimarilerinin bir parçası yaparak görüntüleri sınıf kategorileriyle ilişkilendirmektedir.

Görsel tanımda ilerlemenin yanı sıra, metin işlemede de benzer başarılarla tanık olunmuştur (Mikolov vd. 2013). Kelimelerin dağılımları vektör uzayında benzer bağlamların yakınlarda bulunur. Örneğin, “tüberküloz” ve “zatürre” kelimeleri benzer anlamlara sahip olup vektör uzayında yakın konumdadır. Evrişimli ağlara benzer şekilde, bu kelime düğümlerini yeniden tasarlayabiliriz, diğer uygulamalarda ve daha sonra yeni görevlerde ince ayarlar yaparak modeli çözüme uyarlayabiliriz. Yapay sinir ağı mimarilerini formüle eden yapay sinir dili modelleri, (Pascanu vd. 2013) çalışma kelime dizileri üzerinde olasılık dağılımları çözümlerine nispeten avantaj sağlamıştır. Daha yakın zamanda, tekrarlayan yapay sinir ağlarının (Gregor vd. 2015) dil modelleme görevinde özellikle etkili olduğu kanıtlanmıştır (Mao vd. 2014). Daha büyük mimarilerde cümle ve cümle dağılımı üzerinden dil modelleri ve teknikleri kullanılır.

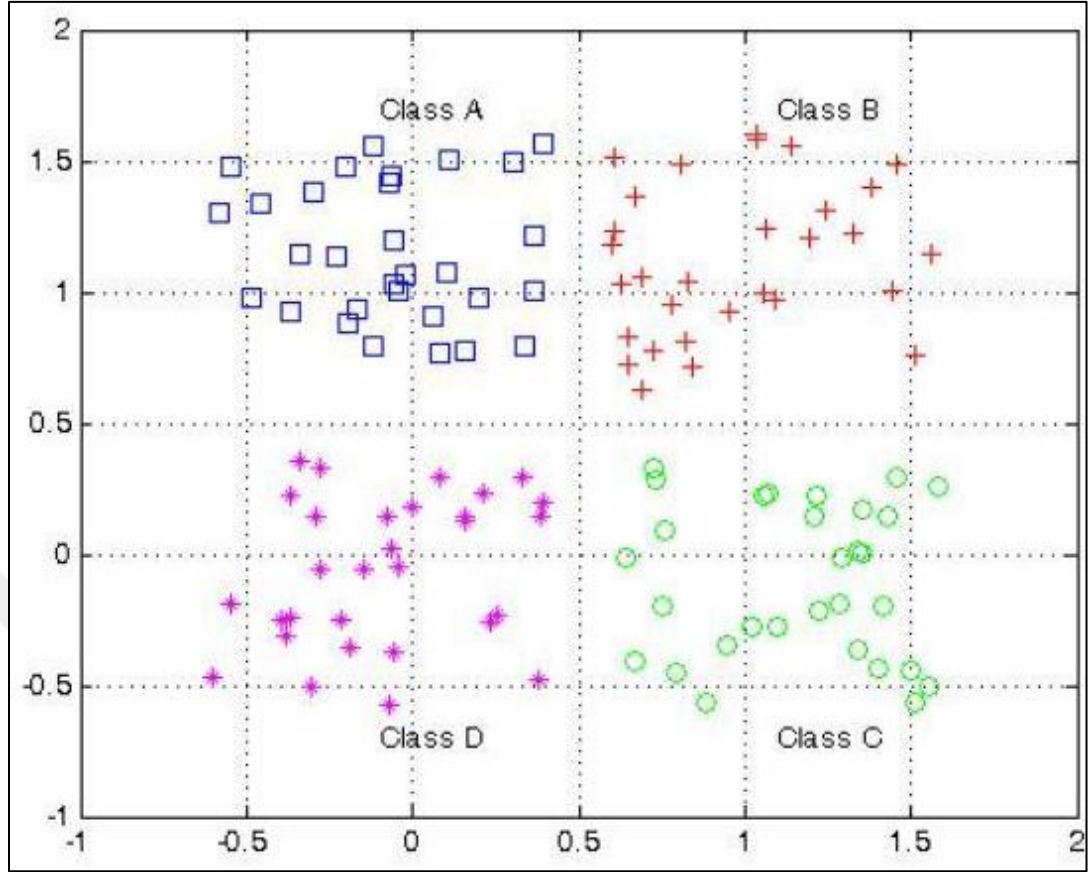
Özetle, bu tezde uçtan uca öğrenme değerler dizisi ve evrişimli sinir ağları benimsenmiş, ön eğitilmiş derin öğrenme modelleri kullanılmış, hiper parametrelerin iyileştirilmesi ve kendiliğinden belirlenmesi sağlanmıştır. Ek olarak görüntü destekli

modellerle sınıflandırma çalışması ele alınmıştır. Sonuç olarak, geliştirilen modeller evrişimli sinir ağlarının avantajlarını ve ön eğitilmiş modellerin ağırlıklarını kullanarak parametrelerin iyileştirilmesi ve oto kodlayıcı (AE) desteği ile beyaz kan hücreleri mikroskopik görüntüleri, Covid-19 göğüs röntgenleri ve kalp krizi metin veri seti üzerinde çalıştırılarak yüksek doğrulukta sınıflandırma sonuçları elde edilmiştir.

2.3 Katkılar ve Anahat

Bu tez çalışmasının amacı bilişim dünyasındaki en son ve çığır açan yapay zekâ konusunun daha ileri ve özel bir alanı olan derin öğrenme ile resim içeriklerini analiz edip zeki çıkarımlar yapan bir sistemi oluşturmaktır. Böylece tıbbın farklı alanlarında uygulanarak daha önceden farkında olunmamış bilgiler ortaya çıkarmak ve sonucu doğrudan hesaplanamayan ya da hesaplanması uzun süren problemlerin çözümlerinin otomatik ve doğru olarak yapılması amaçlanmıştır. Bu tez çalışmasında derin öğrenme modelini sağlık alanındaki görüntü veri setlerine uygulayarak yeni çıkarımlar yapabilen zeki bir sistem ve model geliştirilmiştir. Buradan elde edilen sonuç ve bulgular değerlendirilmiştir.

Geliştirilen derin öğrenme modeli sağlık veri setlerinde; beyaz kan hücreleri, göğüs röntgenleri ve kalp krizi veri setlerinde denendi. Sağlık alanında elde edilen görüntüler üzerinden (Şekil 2.1) sınıflandırma ve tahminler yaparak hastalık teşhislerine katkıda bulunacak bir zeki sistem derin öğrenme modeli oluşturmaktır. Sonuç olarak model, görüntüleri ele alıp beyaz kan hücrelerini sınıflandırmakta ve sayılarını elde edilmektedir. Bunun yanında göğüs röntgenleri analiz edilerek COVID-19 ve tüberküloz teşhisi yapılmaktadır. Yine kalp krizi veri setinden kalp krizi riski taşıma durumu teşhis edilmektedir. Bunu gerçekleştirirken hiper parametre iyileştirmesi, öğrenme transferi oto kodlayıcı ve sınıflandırıcılar ele alınmıştır.



Şekil 2.1 Sınıflandırma örnekleme

2.4 Denetimli Öğrenme

Bir çok pratik problem, bir bilgisayarın bir $f : X \rightarrow Y$, eşlemesi yapmasını gerektirecek şekilde formüle edilebilir. Burada X bir girdi alanı ve Y bir çıktı alanıdır. Örneğin, görsel tanımda X görüntülerin alanı olmalı ve Y , görüntünün belli bir alanındaki desenin nötrofil olma olasılığını gösteren aralık $[0, 1]$ olabilir.

Ne yazık ki, çoğu durumda “f” işlevini manuel olarak belirlemek zordur ve görüntüdeki bir nötrofili tespit eden programın nasıl yazabileceği açık ve net değildir. İşte burada denetimli öğrenme değerler dizisi, eğitimden yararlanan alternatif bir yaklaşım sunar. Ayrıca istenen haritalamanın örneklerini $(x, y) \in X \times Y$ elde etmek genellikle nispeten kolaydır. Çalışılan örnekleme, her birinin uzmanlar tarafından etiketlenmesinden faydalanarak bir beyaz kan hücresinin hangi guruba dâhil olduğu tespit edilmeye çalışılmaktadır. Öncelikle verilerden üretilen bir dağıtımdan, bağımsız ve özdeş olarak

dağıtılmış bir örnek $D = \{(x_1, y_1), \dots, (x_n, y_n)\}$ eğitim veri kümesi elde edilir. Böylece D deki (x_i, y_i) bütün i ' ler için $f: X \rightarrow Y$ haritalamasının sisteme öğretilerek modelin oluşturulması amaçlanmıştır. Sonuçta modelin çıktısı olacak Y ' ler aranarak bir dizi aday fonksiyon üzerinden, eğitim örnekleriyle en tutarlı olanı bulunmaya çalışılmaktadır. Daha kesin olarak, modeli oluştururken elde edilen $\mathcal{F}$ kümesi ve gerçek bir etiketler olan y_i için fonksiyonlar ele alınarak, sayısal ya da nicel değere sahip olan ve tahmin edilen ile gerçek $\hat{y}_i = f(x_i)$ etiketliler arasındaki uyumsuzluk kayıp fonksiyonu $L(\hat{y}, y)$ ile bulunmaktadır (Goodfellow vd. 2016). Öğrenmedeki hedef ideal olarak tatmin edici $f^* \in \mathcal{F}$ leri bulmaktır:

$$f^* = \arg \min_{f \in \mathcal{F}} E_{(x,y) \sim D} L(f(x), y) \quad (2.1)$$

(Goodfellow vd. 2016)

Başka bir deyişle, D dağıtımındaki veri üretiminde beklenen kaybı en aza indiren f^* fonksiyonu aranmaktadır. Pratik uygulamalarda, bu işlevi f^* belirlediğimizde, orjinal eğitim verilerini atabilir ve yalnızca X 'in Y öğelerini eşlemek için kullandığımız öğrenilen işlevi saklanabilir. Ne yazık ki, yukarıdaki iyileştirme problemini çözmek zordur çünkü D 'nin tüm olası öğelerine erişim yoktur ve bu nedenle D , L veya f formu hakkında gerçekçi olmayan güçlü varsayımlar yapmadan beklenti değerlendirilemez ya da analitik olarak basitleştirilemez (Karpathy vd. 2014). Ancak, beklenen kaybı yaklaşık olarak örnekleyerek mevcut eğitim verileri üzerindeki kaybın ortalaması alınır:

$$f^* \approx \arg \min_{f \in \mathcal{F}} \frac{1}{n} \sum L(f(x_i), y_i) \quad (2.2)$$

(Karpathy vd. 2014)

Başka bir deyişle, kayıp sadece mevcut eğitim örnekleri üzerinden optimize edilir, ancak bunun denklem 2.1'deki gerçek hedef için iyi bir vekil hedefi olması umulmaktadır.

Ne yazık ki, denklem 2.1 yerine denklem 2.2'nin optimize edilmesi zorluklar doğurmaktadır. Örneğin, eğitim verilerindeki her bir x_i y_i ile eşleyen ancak başka bir

yerde sıfır döndüren bir f işlevi düşünülürse bu denklem 2.2'ye bir çözüm olacaktır (herhangi bir makul kayıp fonksiyonu L için $y = \hat{y}$ olduğunda minimum bir değere ulaşır), ancak eğitim setinde olmayan D 'deki diğer tüm noktalar için çok yüksek kayıp beklenir. Başka bir deyişle, bu işlevin tüm $(x, y) \sim D$ için genelleme yapması beklenmez. Diğer bir endişe, denklem 2.2 altında aynı kaybı sağlayan birçok farklı fonksiyonun olabileceğidir, ancak eğitim verilerinin dışındaki genellemeleri değişebilir. Eğer sahip olduğumuz tek şey eğitim verisi ise, o zaman denklem 2.2'de aynı kaybı sağlayan $f \in \mathcal{F}$ setinin tamamı arasından seçim yapmak gerekir (Socher vd. 2014). Bu endişelerin ikisi de hedefe düzenli bir terim R eklenerek hafifletilebilir:

$$f^* = \arg \min_{f \in \mathcal{F}} \frac{1}{n} \sum L(f(x_i), y_i) + R(f) \quad (2.3)$$

(Socher vd. 2014)

R , eğitim verilerine uymalarına bakılmaksızın, bazı işlevlerin diğerlerine göre tercihini kodlayan nicel değerli bir işlevdir. Bu ekleme kısmen Occam'ın ustura prensibini izleyerek gerekçelendirilebilir ve şu şekilde ifade edilebilir: “Bir oluşum için iki açıklama olduğunu varsayalım. Bu durumda daha basit olan genellikle daha iyidir.” Başka bir deyişle, düzenlenme bir fonksiyonun karmaşıklığının bir ölçüsüdür. Düzenleme terimiyle birlikte, denklem 2.3'teki amaç, eğitim verilerine de uygun basit çözümleri teşvik eder ve bunun amacı denklem 2.2 ile denklem 2.1'deki hedef arasındaki tutarsızlığı bir ölçüde telafi edilmektedir.

Doğrusal yakınsama: Her biri bir nicel değerle ($Y = \mathbb{R}$) açıklamalı 100 ($n = 100$) 2 boyutlu nokta ($X = \mathbb{R}^2$) veri kümesinin verildiği basit bir örnek için hipotez sınıfı $\mathcal{F}$, X 'ten Y 'ye doğrusal fonksiyonlar kümesidir (yani $\mathcal{F} = \{w^T x + b \mid w \in \mathbb{R}^2, b \in \mathbb{R}\}$). Bu durumda hipotez alanımız 3 parametre (w_1, w_2, b) ile kaplıdır, burada $w = [w_1, w_2]$ dir.

Bir regresyon ayarında yaygın olarak kullanılan bir kayıp fonksiyonu, hedef arasındaki (kare) farklılıktır ve öngörülen değer, $L(\hat{y}, y) = (\hat{y} - y)^2$ dir. Son olarak, yaygın olarak kullanılan bir düzenleme olarak, w_1, w_2 parametrelerini çok büyük olmaktan ve bir giriş özelliğinin tahminler üzerinde orantısız bir etkiye sahip olmasından dolayı $R(w, b) =$

$\lambda(w_1^2 + w_2^2)$ w_1, w_2 kullanabilir ve ayrıca düzenlemenin gücünü belirten bir parametre ile bu parçaları bir araya getirmek, problemi çözmek için denklem 2.4 deki gibi formüle edilebilir (Goodfellow vd. 2016);

$$f^* = \arg \min_{w,b} \left[\frac{1}{n} \sum_{i=1}^n (w^T x_i + b y_i)^2 \right] + [\lambda(w_1^2 + w_2^2)] \quad (2.4)$$

Özellikle, yanlılık terimlerinin normalleştirmeden çıkarılması yaygındır (Karpathy vd. 2015). Bunlar girdilerle çarpımsal olarak etkileşime girmez; sadece modelin kendi kendini kurmasına izin verir.

Sinir ağı gerilemesi kısa bir ön izleme olarak, yukarıdaki örneği sinirsel ağ F hipotez alanını daha karmaşık hale getirmek için yeterlidir. Örneğin, $f(x) = w^T x + b$ formunun doğrusal fonksiyonlarını araştırırken burada W_1, w_2, b_1, b_2 tüm parametrelerdir: $W_1, H \rightarrow 2$ boyutunda bir matristir, $b_1, f(x) = w_2 \tanh(w^T x + b_1) + b_2$, H boyutunda bir vektördür, w_2, H ve b_2 büyüklüğündeki bir vektör bir niceldir. Burada H , seçmekte özgür olduğumuz bir tamsayıdır (Örneğin 100). Genellikle gizli katmandaki nöron sayısı olarak yorumlanır ve hiperbolik tanjant ‘tanh’ eleman olarak uygulanır (değerleri $[-1, 1]$ aralığına ekler ve sinir ağıları bağlamındadır yaygın olarak doğrusallık olarak adlandırılır).

$$f^* = \arg_{w_1, b_1} \min_{w_2, b_2} \left[\frac{1}{n} \sum_{i=1}^n (w_2 \tanh(W_1^T x_i + b_1) + b_2 y_i)^2 \right] + [\lambda (||W_1||_2^2 + ||W_2||_2^2)] \quad (2.5)$$

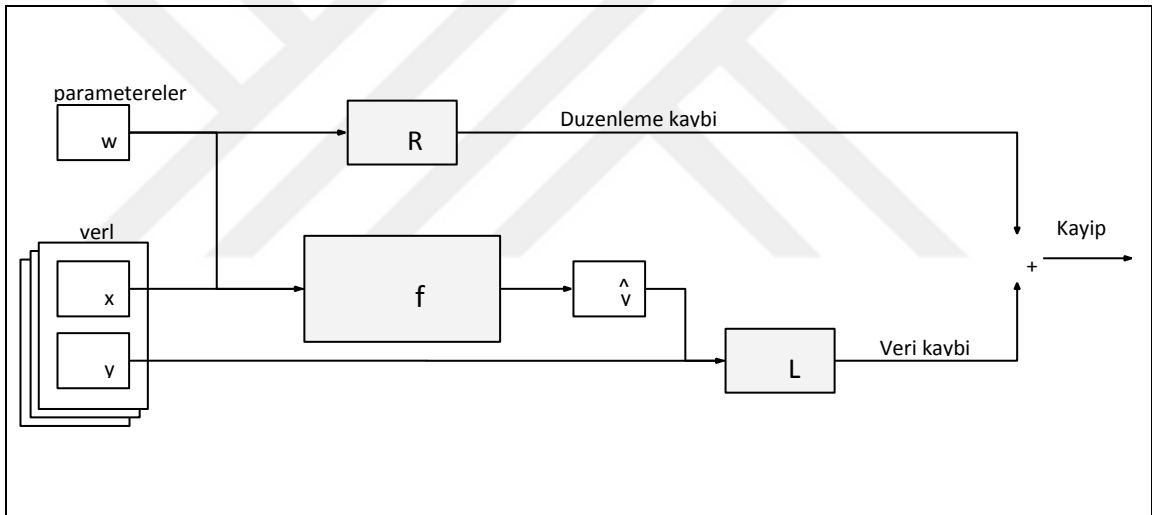
Böylece amaç sinir ağı sınıflandırması, nicel değerli bazı miktarları tahmin etmek yerine her bir girdi için ortak bir pratik model ile bir kişinin girdiyi atamak istediği, ayrı kategorilere sınıflandıran yapının oluşturulmasıdır.

Örneğin, üç olası sınıf kullanarak $f(x) = W_2 \tanh(W_1^T x + b_1) + b_2$, burada W_2 artık bir $K \in H$ matrisidir (bu örnekte $K = 3$), bu nedenle f 'nin çıkışı bir 3 boyutlu vektördür. Bu vektördeki sayıları genellikle log olarak yorumlanır ve dolayısıyla bu vektörü softmax

fonksiyonundan geçirerek üç sınıfın olasılıklarını da hesaplayabilir. Böylece bir z vektörü alınır ve aynı boyuttaki p' ye sahip bir vektör çıkarır. Hesaplama formülü şu şekildedir;

$$p_i = e^{z_i} / \sum_{k=1}^K e^{z_k} \quad (2.6)$$

z vektörünün rastgele gerçek değerli miktarlar içerebilecek iken, p vektörü normalleştirilecek şekilde tüm elemanları 0 ile 1 arasındadır ve 1'e (örneğin [0.2, 0.5, 0.3]) sahiptir. Burada p vektörü aslında ağdan tahminimiz ve doğru etiket $\hat{y}$, 3 boyutlu bir vektörü olup gerçek sınıfın dizinindeki 1 haricinde 0 dir. Örneğin, doğru sınıf üçüncüsü $y = [0, 0, 1]$ gibi.



Şekil 2.2 Tipik denetimli bir öğrenme probleminde veri akışının şema sinir ağı (Goodfellow vd. 2016)

Sınıflandırma ayarında en sık kullanılan kayıp fonksiyonu için x ve y etiket çiftlerinin (x, y) veri kümesi olmak üzere üç fonksiyon uygulanabilir.

1. x örneklerini tahmin edilen bazı etiketlerle eşleyen f işlevi $\hat{y}$ (genellikle bu tezdeki bir sinir ağı) da w bazı parametrelerinin kullanılması (bazen de θ belirtilir).
2. Öngörme arasındaki uyumsuzluğu değerlendiren $L(\hat{y}, y)$ işlevi ve gerçek etiket.
3. Eşleştirmenin karmaşıklığını değerlendiren R fonksiyonu.

Veri kaybı ve düzenleme kaybı eklenir ve tüm grafik tek bir nicel değer ölçümü üretir parametrelerin verilerimize ne kadar uygun olduğu tespit edilir. Amaç denklem 2.6'daki biçimde çapraz entropi kaybı için son kaybı en aza indiren w parametrelerini bulmaktır.

$$L(\hat{y}, y) = \sum_{k=1}^K y_k \log \hat{y}_k = \log \hat{y}_{y=1} \quad (2.7)$$

Burada ilk eşitlik iki dağılım arasındaki çapraz entropinin tanımıdır $H(p, q) = -\sum_x p(x) \log q(x)$ ve ikinci eşitlik ifadeyi basitleştirir çünkü sınıflandırma ayarlarının genellikle tüm olasılık kümesinin tek bir doğru üzerinde olduğu varsayılır. Tamsayı endeksi $y = 1$ olarak belirlendiği eleman, ağın çıktısını üç farklı sınıfın olasılıklarını içerdiğini ve aynı zamanda etkili bir şekilde en aza indirgendiğini görürüz. Doğru sınıfın negatif log olasılığı, tutarlı x girişinde koşullandırılmış y sınıfının olasılığını en üst düzeye çıkarmaktır (Goodfellow vd. 2016).

Denetimli öğrenmede n veri noktası veri kümesi verilir $\{(x_1, y_1), \dots, (x_n, y_n)\}$

$(x_i, y_i) \in X \times Y$ denkleminin kullanıldığı yerde sorunu belirlemek için üç değer belirlenir:

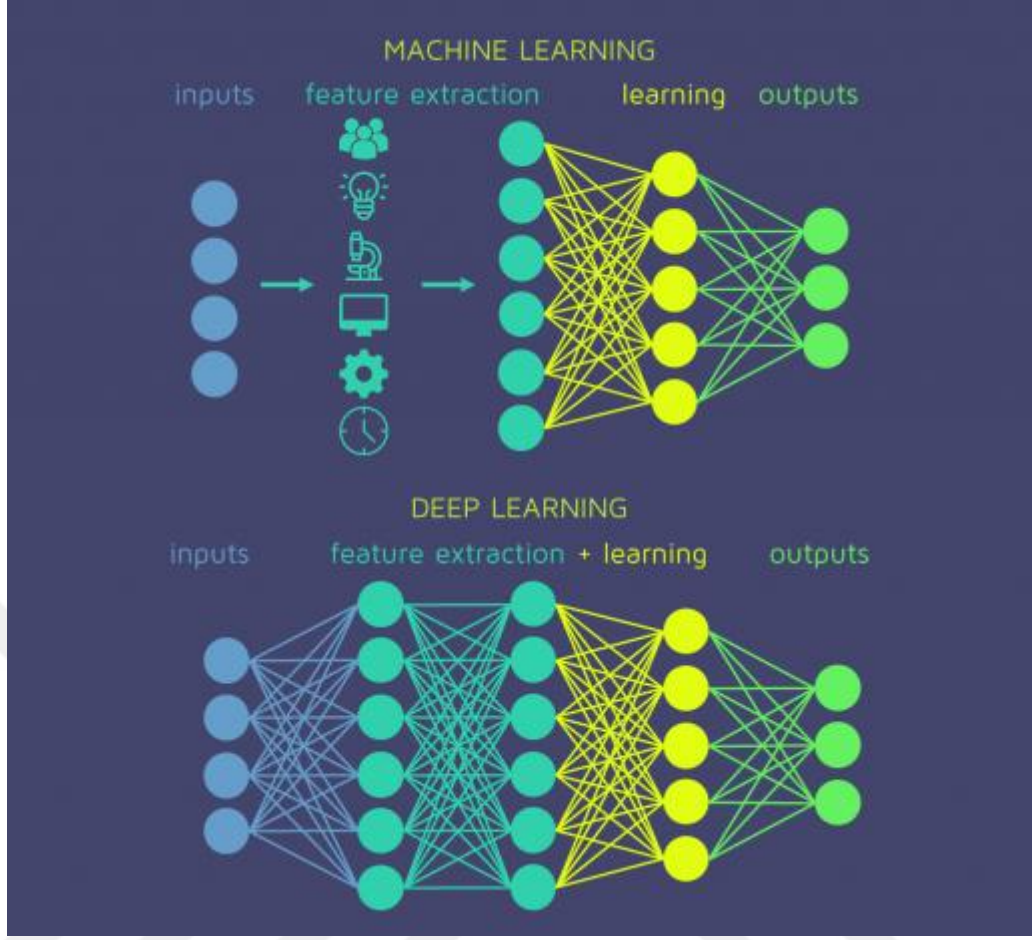
1. Her bir $f \in \mathcal{F}$ 'nin X ile Y 'yi eşleştirdiği $\mathcal{F}$ işlevlerinin arama alanı.
2. Gerçek $L(y, \hat{y})$ bir etiket y ve öngörülen bir etiket $\hat{y} = f(x)$.
3. Bir haritanın karmaşıklığını ölçen nicel değerli düzenlenme kaybı $R(f)$.

En yaygın olarak, $\mathcal{F}$ fonksiyonlarının uzayını öğrenmek, yukarıdaki örnekte görüldüğü gibi bazı parametrelerle sinir bir ağ olacaktır, L kaybı, yakınsama veya ötropropide öklid kaybı olacaktır. Sınıflandırma kaybı ve R düzensizleşmesi en yaygın olarak L_2 normudur (yani toplamı tüm ağırlıkların kareleri).

Bu seçimler yapıldıktan sonra, denetimli bir öğrenme görevi için bir model öğrenme sorunu form $\theta^* = \argmin_{\theta} g(\theta)$ genel formunun iyileştirme problemine indirgenir, burada θ bir parametredir vektör ise: $g(\theta) = \frac{1}{n} \sum_{i=1}^n L(f_{\theta}(x_i), y_i) + R(f_{\theta})$. Burada parametrelerden θ genellikle tamamen fonksiyon haritalaması f ve L, R ye aittir.

2.4.1 Derin öğrenme

Yapay zekâ çalışmaları 1960 yıllarına dayanmakta olup, bu yıllarda kredi değerlendirme sisteminde kullanılırken, 1970'lere kadar eğriye düşme (gradyan descent) ve geriye yayılım (backpropagation) yöntemleri eğitim için kullanılarak daha güçlü modeller geliştirilmiştir. 1981 senesine gelindiğinde çok katmanlı mimari olan derin öğrenme denense de pratikte başarılı olamamıştır. 1990'dan 2000 senesine kadar olan süreçte ise derin öğrenme pratiğe dökülmeye başlandı. Ancak 2009 senesine gelindiğinde denetimli derin öğrenme modeli birçok desen çözme yarışmasını kazanarak büyük yankı uyandırdı (Schmidhuber 2015). Bu başarılardan itibaren de sürekli gelişerek yeni problemlerin çözümünde de büyük başarılar elde ederek geleceğe yön veren bir çalışma alanı olmuştur. Bir diğer yönü ise bu teknolojinin artık tüm cihazlarda kullanılıyor olmasıdır. Makina öğrenme teknolojisi, modern toplumun birçok yönünü güçlendirir: web aramalarından sosyal ağlara içerik filtrelemeye, e-ticaret web sitelerindeki önerilere kadar ve kameralar ve akıllı telefonlar gibi tüketici ürünlerinde giderek daha fazla yer almaktadır (Lecun 1990).



Şekil 2.3 Derin öğrenme modelinin makina öğreniminden farkı (QuantDare 2019)

Bu da göstermektedir ki geleceğin teknolojisini insanlığa her alanda katkı sağlama potansiyeli çok yüksek olan yapay zekâ ve derin öğrenme alanları oluşturacaktır. Peki, makina nasıl öğrenir? Temsili öğrenme, bir makinanın ham verilerle beslenmesini ve tespit veya sınıflandırma için gereken gösterimleri otomatik olarak keşfetmesini sağlayan bir yöntem kümesidir (Lecun 1998). Derin öğrenme, birden çok katmanı içeren ve görüntülerin sınıflandırılmasında başarılı olan, denetimli bir derin öğrenme teknolojisi türüdür (LeCun vd. 2015). Şekil 2.3 de gösterildiği gibi derin öğrenmenin en önemli avantajı da uzmanlık gerektiren ve makina öğreniminde ayrıca yapılması gereken özellik çıkarımının model tarafından yapılmasıdır. Görüntüler gibi veri kümelerini birden çok düzeyde sunma yetenekleri, derin öğrenmenin önemli bir avantajıdır. Örneğin, derin öğrenme yüksek seviyelerdeki tüm nesneleri, orta seviyedeki nesnelerin parçalarını ve daha düşük seviyelerde kenarlara sahip pikselleri tespit edebilir. Çoklu katmanlar, veri özelliklerini çıkarma işlemlerini kolaylaştırır ve doğru

sonuçlar üretme yeteneklerini artırır. Derin öğrenme modelinde makina öğreniminin temel aldığı doğal metin, resim, ses ve video gibi veriler üzerinde çalıştırılarak başarılı sonuçlar elde edilmektedir.

2.4.2 İyileştirme

Denetimli öğrenme için bir model, öğrenme görevini indirgeyebilir. $\theta^* = \arg \min_{\theta} g(\theta)$ formunun iyileştirme probleminde θ bir parametre, vektör ve g genellikle tüm örneklerin ortalama kaybını ve bir düzenlenme cezası birleştirilir.

Türevsiz iyileştirme: İlk olarak, $g(\theta)$ herhangi bir θ için iyileştirme probleminin çözümünde etkin şekilde stokastik tepe tırmanma yöntemlerini “tahmin etme ve kontrol etme” kullanmaktır. Örneğin, bazılarında rastgele çok sayıda θ dağıtılır, her biri kontrol edilir ve g 'yi en aza indiren biri seçilir. Daha ayrıntılı bir yaklaşım tekrar tekrar küçük sapmalar yaparak bazı adayları geliştirmeye çalışır. Ne yazık ki, eğitmek istediğimiz tipik sinir ağlarında milyonlarca ya da milyar parametre vektörü olabilir, bu yaklaşımların birçoğu hesaplama açısından zorlayıcıdır.

Birinci dereceden yöntemler: Ek olarak g ile ilgili varsayımlar için iyileştirmenin verimliliğini artırabiliriz. Özellikle, yalnızca farklı işlevler kullanmakla kısıtlıysak, $\nabla_{\theta} g$ gradyanı geri yayımlı besleme ile hesaplanabilir. Gradyan, g 'nin Taylor genişlemesinde ilk sıra yaklaşımını, dolayısıyla “birinci sıra yöntemleri” adını vermemizi sağlar. Gradyan arama yönü olarak kullanabiliriz; özellikle, θ 'ya (daha düşük g 'ye ulaşma anlamında), az miktarda negatif indirgeme yönü ekleyerek onu geliştirebiliriz (çünkü g 'yi en aza indirmek istiyoruz, ancak indirgeme g 'nin artış yönünü verir).

Negatif gradyan yönünde küçük bir adım atarak parametreleri güncellenir, pratik bir değerlendirme olarak, kullandığımız veri kümeleri çok büyük olabilir, bu nedenle indirmeyi yalnızca küçük bir örnek grup kullanarak tahmin edilir.

Bu, çoğu pratik uygulamada iyi çalışan bir strateji ile daha az kesin güncelleme yerine birçok yaklaşık güncelleme yapılmasını sağlar. Sonuçta elde edilen algoritma, Stokastik İniş (SGD) Algoritma 1'de özetlenmiştir.

Algoritma 1 Stokastik Gradyan inişi.

Bir başlangıç noktası $\theta \in \text{dom}g$

Artış $\epsilon \in \mathbb{R}^+$

Tekrar et

1. Eğitim verilerinden $\{(x_1, y_1), \dots, (x_m, y_m)\}$ m örneklerinden oluşan gruplar
 2. Geri yayılım ile indirgeme tahmin $\nabla_{\theta} g(\theta) \approx \nabla_{\theta} \left[\frac{1}{m} \sum_{i=1}^m L(f_{\theta}(x_i), y_i) + R(f_{\theta}) \right]$
 3. Güncelleme yönünü hesapla: $\Delta \theta := -\epsilon \nabla_{\theta} g(\theta)$
 4. Parametre güncelleştirmesi gerçekleştirin: $\theta := \theta + \Delta \theta$
- Yakınsayana kadar.

SGD' deki kritik bir parametre ϵ adım büyüklüğüdür (öğrenme hızı olarak da adlandırılır). Çok yüksekse iyileştirme yakınsama yapamaz, hatta farklılaşmayabilir. Çok düşük ayarlanırsa öğrenme çok uzun sürer. $y = x^2$ (gradyan d_y değerine sahip) işlevini en aza indirmeyi düşürecek bir örnek için $\frac{dy}{dx} = 2x$ ile azalma, iniş $x=1$ 'den başlar. Ardından $\epsilon > 1$, inişin sonsuza sapmasına neden olur, $\epsilon = 1$, iyileştirmenin $x = 1$ ve $x = -1$ ve $\epsilon < 1$ arasında süresiz olarak salınmasına neden olur. $x = 0$ ' da minimuma yakınsamayı sağlayacaktır. Özellikle, örnekteki en iyi öğrenme oranı, ıraksama sınırında, $x = 0.9999$ ' dur. Uygulamada iyi bir sezgisel tarama, optimizasyonun ayrılmasını sağlayan (örneğin 0.1) en düşük ϵ ayarını bulmak ve ikili öğrenme oranını bir güvenlik marjı olarak bu miktardan (örn. 0.05) biraz daha küçük olacak şekilde ayarlamak için elle ikili arama yapmaktır. Eğitim sırasında bu öğrenme oranını yumuşatmak iyi bir fikirdir ve iyi bir kuraldır, iyileştirmenin sonunda bunu yaklaşık 100 faktör azaltmaktır. Bu, sabit aralıklarla bir sabitle (örneğin iki kez 0.1) çarpılarak veya bir yumuşatma çizelgesi kullanılarak yapılabilir. Genel olarak, farklı ayarlar farklı sorunlar için daha iyi veya daha kötü çalışır, bu nedenle bu bölümde daha sonra ele alınacak çapraz doğrulama kullanarak soruna özgü ayarları belirlemek yaygındır.

Gelişmiş birinci dereceden iyileştirme teknikleri: Uygulamada genellikle güncelleme yönünün hesaplamasını değiştirerek daha hızlı yakınsama elde edilebilir (Algoritma 1'de Adım 3). Örneğin, bu tekniklerden biri indirgemenin küçük ama tutarlı yönleri boyunca ilerlemeyi teşvik etmek için tasarlanmış Momentum güncellemesidir. İndirgeme için $g = \nabla_{\theta} g(\theta)$ kısa gösterimi için güncelleme $\Delta\theta$ ilk önce bir ara değişken $v := \alpha v + g$ güncellenerek hesaplanır ve ardından güncellemeyi $\Delta\theta := -\epsilon v$ olarak hesaplar. v değişkeni bir önceki gradyan yönlerinin katlanarak azalan toplamıdır. Bu güncellemenin, fizik ile yakın bir ilişkisi vardır; burada gradyan, pozisyonu θ olan bir parçacık üzerinde F kuvveti olarak yorumlanır ve doğrudan pozisyonu yerine hızını arttırır. Bu Newton'un ikinci yasası $F = \frac{dp}{dt}$ ile tutarlıdır, burada momentum $p = mv$ dir ve m kütesinin sabit olduğu varsayılır. Momentum, güncellemeyi, indirgemenin ilk anının çalışan bir tahmini ile ayarlanır. İkinci yöntemi kullanarak güncellemeyi uyarlayan bir dizi yöntem de geliştirilmiştir. Örneğin, Adagrad güncellemesi toplamda $(r := r + g \odot g)$ ara değişkenini kullanır. Kare indirgemeler ikinci ara güncellemeyi değiştirir aşağıdaki gibi formüle edilir (Duchi vd. 2011);

$$\Delta\theta := \frac{\epsilon}{\delta + \sqrt{r}} \odot g$$

Burada δ küçük bir sayıdır (örn. $1e^{-5}$), sıfıra bölünmeyi önler. RMSProp güncellemesi (Tieleman ve Hinton 2012) bunun yerine ikinci momentin bir çalışma ortalaması kullanılır: $r := \rho r + (1 - \rho)g \odot g$ Burada ρ genellikle 0,99'a ayarlanmıştır. Son olarak, Adam güncellemesi (Kingma ve Ba 2014) hem birinci hem de ikinci tahminleri anlar ve RMSProp ile Momentum'un bir bileşimi olarak görülebilir. Bu yöntemlerin parametre güncellemelerinde eşitleme etkisi vardır. Büyük indirgemeleri gören parametreler daha küçük sürer çok düşük indirgemeleri gören adımlar ve parametreler daha büyük adımlar alacaktır.

Sorun formülasyonun ve iyileştirmenin, düzenlenme gücü λ , stokastik gradyan iniş ϵ için adım boyutu, bir sinir ağındaki H gizli birimlerin sayısı gibi ayarlarının belirlenmemiş bırakılmasıdır. Bunun yerine stokastik iyileştirme tekniklerine başvurulur; yani, birkaç olasılık denenir her durumda modeli optimize edip ve son

olarak eğitim sırasında kullanılmayan saklı, onaylanmış bir dizi örnek hakkındaki tahminler değerlendirilir (Çapraz doğrulama). Bu, genelleme hatasını tahmin etmenin bir yoludur. Çok fazla örneğin bulunmadığı durumlarda, veri kümesini bazı katlara bölmek, doğrulama için bir kat kullanmak, eğitim için geri kalanı kullanmak, doğrulama katının tüm olası seçenekleri arasında geçiş yapmak ve ortalama doğrulama performansını bildirmek de mümkündür. Bu işleme çapraz-doğrulama veya k-katlı çapraz-doğrulama denir (Albawi vd. 2017).

2.4.3 Geri yayılım

Kayıp fonksiyonu, hedefin gerçek ve tahmin edilen değeri arasındaki mesafeyi ölçer. Ölçülen değer üzerinden modelin istenilen değere ne kadar yaklaştığı sonucuna ulaşılır. Kayıp fonksiyonunun gradyanı değerlendirildiğinde, onu en aza indirmek için stokastik gradyan inişi kullanılabilir ve düşük düzenleme maliyetine ulaşan $f \in F$ eşleşmeleri ve X ile Y' yi eğitim verileriyle eşleştirdiği görülür.

Geriye besleme ele alınırsa, girdilerine göre nicel değerli fonksiyonların gradyanları verimli bir şekilde hesaplandığı bir süreç olarak nitelenebilir. Geri çoğalma algoritması, zincir kuralının hesaptan tekrarlanan bir uygulamasıdır. İnişleri hesaplamakla ilgilendiğimiz fonksiyonun, örneklerin (x_i, y_i) ve θ parametrelerinin girdi setini alan g olduğu hatırlanmalıdır. Parametre güncellemesini gerçekleştirmek için parametrelerle ilgili olarak $\nabla_{\theta} g$ gradyanına odaklanılır, ancak eğer istenirse, x_i girişleri için indirgemeleri aynı işlemle de hesaplanabilir.

Örneğin $y = (5x + 4)^2$ ifadesine sahip olduğumuzu varsayarak ve $\frac{\partial y}{\partial x}$ işlemine odaklanırsak; bir sinir ağı uygulamasında x , şunları içeren bir vektör olacaktır: hem giriş verileri hem de ağı parametreleri ve y ile birlikte toplam kayıptır. Anlaşılır olması için ara değişkenlerin tanıtılmasıyla, y çıkışının $a = 5x$, $b = a + 4$ ve $y = 2b$ ara fonksiyonlarından oluşan bir dizi aracılığıyla x girişinin bir fonksiyonu olduğudur. Ayrıca, renk geçişinin girdilerine göre her bir dönüşümün yazılması kolaydır: $\frac{\partial \alpha}{\partial x} = 2$,

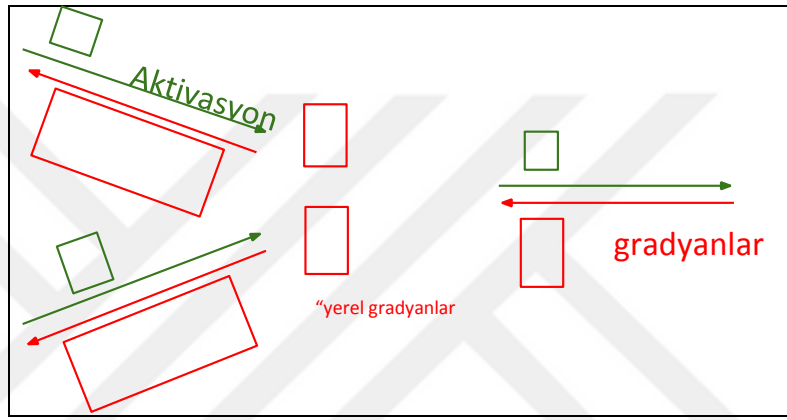
$\frac{\partial b}{\partial a} = 1$ ve $\frac{\partial y}{\partial b} = 2b$. Bu (yerel) türevleri anladıktan sonra, istenen (global) türevi $\frac{\partial y}{\partial x}$ zincir kuralını tekrar tekrar uygulayarak $\frac{\partial y}{\partial x} = \frac{\partial y}{\partial b} \frac{\partial b}{\partial a} \frac{\partial a}{\partial x}$ elde edilir.

Yukarıdaki örneğe bakıp genelleme yaparak, girdiye göre çıktının nihai türevini elde etmek için tüm ara fonksiyonların bireysel yerel türevleri birlikte çoğaltılır. Daha genel bir gösterim kullanarak, bir girdimiz olduğunu varsayalım $x_i = f_i(x_{i-1})$ bir dizi fonksiyon aracılığıyla dönüştürdüğümüz x_0 vektörü, burada $i = 1, \dots, k$ ve son x_k bir niceldir. İndirgemenin olduğunu varsayarak, Jacobi matrisi hesaplanır $\frac{\partial x_i}{\partial x_{i-1}}$ tüm ara dönüşümlerin x_i 'nin çıkış boyutu, x_{i-1} 'in her giriş boyutuna bağlıdır. Zincir kuralı ile son gradyanı ele alıp, sadece tüm Jacobi matris ürününü:

$$\frac{\partial x_k}{\partial x_0} = \prod_{i=1}^k \frac{\partial x_i}{\partial x_{i-1}} \text{ Formülü ile hesaplanır.}$$

Çoğu sinir ağı uygulamasında x_0 'ın büyük olduğunu (örn. Bir görüntünün tüm pikselleri), x_k küçük olduğunu (genellikle bir, nicel değerli bir kayıp fonksiyonumuz olduğu için) anlamak önemlidir. Bu nedenle, tüm Jacobi matrislerinin yukarıdaki çıktıları baştan sona değerlendirmek, yani başka bir yol yerine aşağıdan başlayarak hesaplamak önemli hale gelir. Bunu somut bir örnekle görmek en iyisidir; x 'in 10 boyutu varsa, yukarıdaki örneğimizde ürün üç boyut matrisinin çarpılmasını gerektirir: $[1 \times 10] \times [10 \times 10] \times [10 \times 10]$. Önden (sağdan) arkaya (sola) gitmek için $[10 \times 10] \times [10 \times 10]$ matris çarpımı ve ardından $[1 \times 10] \times [10 \times 10]$ gerekirken arkadan öne doğru yalnızca $[1 \times 10] \times [10 \times 10]$ iki kez gerektirir. Daha genel olarak, çok sayıda giriş ve tek bir çıkış varsa (çoğu sinir ağı uygulamasında olduğu gibi) öne geri gitmek her zaman daha hesaplama açısından etkilidir (buna ters mod farklılaşması denir). Tersine, tek bir giriş ve birçok çıkış olsaydı önden arkaya gitmek daha verimli olurdu (ileri mod farklılaşması). Ne yazık ki, ters mod farklılaşması, bellekte ileriye doğru geçiş sırasında hesaplanan tüm ara değerleri korumamızı gerektirmektedir, çünkü indirgemeleri daha sonra geriye doğru çoğaltıldığı için onlara ihtiyaç duyulur. Bu, hesaplama verimliliği için ödediğimiz bellek kapasitesinde bir maliyettir. Prensipite keyfi işlevler için Jacobi matrislerinin $\frac{\partial x_i}{\partial x_{i-1}}$ çok büyük olabileceği unutulmamalıdır. Bununla birlikte, çoğu pratik

uygulamada bu matrisler çok özel bir yapıya sahiptir ve bunları açıkça bellekte yaratılması gerekmez. Örneğin, 4096 boyutlu bir vektör alan ve tüm boyutlara tanh uygulayan doğrusal olmayan tanh, diyagonal boyunca 4096x4096 boyutunda bir Jacobi matrisine sahiptir. Bu matrisi bellekte açıkça oluşturmak ve tam bir matris çarpımı gerçekleştirmek verimsiz olacaktır; bunun yerine, Jacobi matrisinin sadece bu özel işlemin kimliği boyunca unsurları olduğu için, matris çarpımı sadece diyagonal boyunca elemanlarla bir çarpma yaparak çok verimli bir şekilde uygulanabilir.



Şekil 2.4 Hesaplamalı bir grafik boyunca geri yayılım örneği

İleri geçiş sırasında x ve y belirli (sayısal) değerleri alır ve z vektörü (veya nicel) bazı sabit değerler kullanılarak hesaplanır. Fonksiyon (örn. $z = x \odot y$), $\frac{\partial z}{\partial x}$ ve $\frac{\partial z}{\partial y}$ Jacobi matrisleri ve bu dönüşüm hemen hesaplanabilmektedir, çünkü z 'nin ileri geçişte hangi fonksiyonu hesaplandığı bilinir. Bunlar x ve y 'nin z değerine birinci dereceden etkisinin ne olduğunu söyler. z değeri bir hesaplama grafiğine gider ve grafiğin sonunda toplam kayıp 'g' hesaplanır (Şekil 2.4). Geriye doğru geçiş ters sırada ilerler ve grafiğin tüm girişlerinin nihai çıktı üzerindeki etkisini bulmak için zincir kuralını tekrar tekrar uygular. Özellikle, bu hesaplama birimi $\frac{\partial g}{\partial z}$, z 'nin son grafik çıktısını nasıl etkilediğini söyler. Zincir kuralı, bunu geri çoğaltmak için genel gradyanı almamız ve her girdi için $\frac{\partial g}{\partial z}$ yerel gradyanlarıyla çarpılması gerektiğini belirtir. Örneğin, x için küresel (global) gradyan, vektör olacak, daha sonra bu, tek bir matris-vektör çarpımı olacaktır. Daha sonra gradyan $\frac{\partial g}{\partial x} = \frac{\partial z}{\partial x} \frac{\partial g}{\partial z}$ girişlere ulaşılan kadar x ve y değerlerini üreten

fonksiyonlar aracılığıyla tekrar tekrar zincirlenir. Sinir ağı uygulamalarında, ilgilendiğimiz girdiler parametrelerdir ve gradyanlar, kaybı azaltmak için hangi yöne çekilmeleri gerektiğini söyler. Özetle, çıktının girdiye göre eğiminin değerlendirilmesi, zincir kuralı ile Jacobi matrislerinin bir çıktısına indirgenir. Sinir ağı uygulamalarında, önce tüm ara değerleri (daha sonra önbelleğe almak için) ve maliyeti g hesaplamak için bir grup veri ve akım parametresini θ ve “ileri” aldığımız bir ileri geçiş $\{(x_i, y_i)\}_{i=1}^m$ gerçekleştirilir. Daha sonra geri yayılma sırasında, tüm ara aşamalarda (“geriye doğru geçiş”) geriye doğru ilerleriz ve her seferinde matrisin tam çıktıda bir sonraki Jacobi matrisi ile çarpılarak yerel indirgemeler zincirlenir. Birçok durumda aslında tam Jacobi matrislerini oluşturmak gerekmez ve indirgemeleri daha hesaplamalı olarak verimli bir şekilde zincirlemek için özel yapıdan yararlanılabilir.

Bir Sinir Ağı'nın hesaplama sürecini doğrusal bir operasyon listesi olarak düşünmek yerine, girdilerden çıktıya kadar olan fonksiyonun, vektörlerin kenarlar ve düğümler boyunca aktığı farklılaşmayı temsil ettiği operasyonların yönlendirilmiş bir grafiği olarak düşünülebilir. Geri çoğaltma uygulamalarının çoğu, kod tabanını, işlemlerin ve büyük bir olası işlem koleksiyonunu koruyan bir ağ (graph) nesnesi etrafında düzenler. Hem ağ hem de düğüm nesneleri iki işlev uygular: `ileri()` ve `geri()`.

Grafiğin `ileri()` yöntemi, topolojik sırayla tüm düğümler üzerinde yinelenir ve `ileri()` 'i çağırır ve `geri()`, ters sırada yinelenir ve `geri()` olarak adlandırılır. Her Düğüm, çıkışını `ileri()` çağırısı sırasında bazı işlevlerle hesaplar ve `geri()` işlevinde, çıkış işlevine göre kayıp işlevinin gradyanı verilir ve tüm girdilerinde “zincirleme” indirgeme döndürür. “Zincirleme” ile gradyanı çıktısında almak ve kendi yerel gradyanıyla (bu dönüşümün Jacobi'ı) çarpılır. Bu indirgeme daha sonra aynı işlemi yinelemeli olarak gerçekleştiren alt düğümlere geçer. Bir düğümün çıktısı grafikte birden fazla yerde kullanılıyorsa, çok değişkenli zincir kuralı ile doğru davranış, tüm dallar boyunca geri yayılım sırasında indirgemeleri eklemektir.

Örnek olarak, bir tanh katmanı, ileri işlevini $y = \tanh x$ olarak uygular; burada tanh, eleman tabanlı uygulanır. Geri çoğaltma sırasında geriye doğru fonksiyon $\frac{\partial g}{\partial y}$ ile birlikte

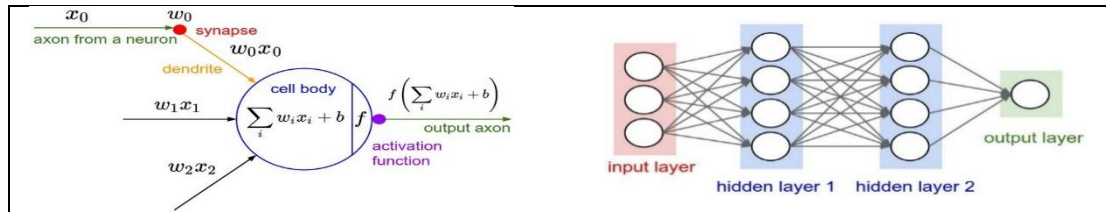
temin edilen ağ (graph) nesnesi tarafından çağrılır; burada g, grafiğin sonundaki kayıptır. Geri fonksiyonun amacı, bu katmanın Jacobi matrisi çalışan ürüne zincirlemek, bu durumda matris ürününü hesaplamaktır $\frac{\partial g}{\partial x} = \frac{\partial y}{\partial x} \frac{\partial g}{\partial y}$. Bununla birlikte, bu katmanın Jacobi matrisinin sadece diyagonal boyunca elemanlara sahip olması nedeniyle, bu matris çarpımı $\frac{\partial g}{\partial x} = eye\left(\frac{\partial y}{\partial x}\right) \odot \frac{\partial g}{\partial y}$ olarak etkili bir şekilde uygulanabilir. Burada element olarak çarpma ve $eye\left(\frac{\partial y}{\partial x}\right)$ matrisin bir vektör olarak diyagonalıdır (tanh için bu vektör olacaktır $(1 - y \odot y) \Rightarrow \frac{d}{dx} \tanh x = (1 - (\tanh x)^2)$). Doğru (Graph) nesnesi $\frac{\partial g}{\partial x}$ daha sonra döndürülen bu katmandan alınır ve bunu x hesaplanan düğüme aktarılır, özyinelemeli işlemi grafik girişlerine (veriler ve parametreler) kadar devam edilir.

2.5 Geleneksel Sinir Ağları (Vanilla Neural Networks)

Önceki bölümlerde x girişlerini tahmin edilen çıkışlara dönüştüren ve modeli stokastik gradyanlı iniş kullanarak herhangi bir değişkenlik kaybı fonksiyonuna (loss of variability) göre optimize eden keyfi olarak değişken fonksiyonları f olarak tanımlayabildiğimiz ele alındı. Şu ana kadar detaylandırılmamış f fonksiyonunun ayrıntıları incelenecektir.

X'in yapısı hakkında herhangi bir varsayımın yokluğunda, matris çarpımlarını ve element bazında lineer olmayan doğrusallıkları tekrarlayarak sinir ağları inşa ediliyor.

Örnek olarak, 2 katmanlı bir sinir ağı:



Şekil 2.5 Tek bir nöronun arkasındaki biyolojik ilham ve yapay sinir ağı diyagramı

Şekil 2.5’ de X_i girişleri w_i sinapsları ile çarparak etkileşir, hücre gövdesi toplamı biriktirilir ve sonra aktivasyon fonksiyonundan bir çıkış sinyali gönderilir. Aktivasyon sigmoid doğrusal olmama durumunda (çıkış aralığı $[0,1]$ ile), çıkış nöronu ortalama ateşleme oranı olarak yorumlanabilir. Bir katmandaki nöronlar, önceki katmandaki tüm nöronlarla bağlantılıdır, ancak birbirine bağlı değildir. Bu düzenleme, tüm nöronların aktivasyonlarını bir matris çarpımı ile tek bir katmanda etkili bir şekilde değerlendirmemizi sağlar.

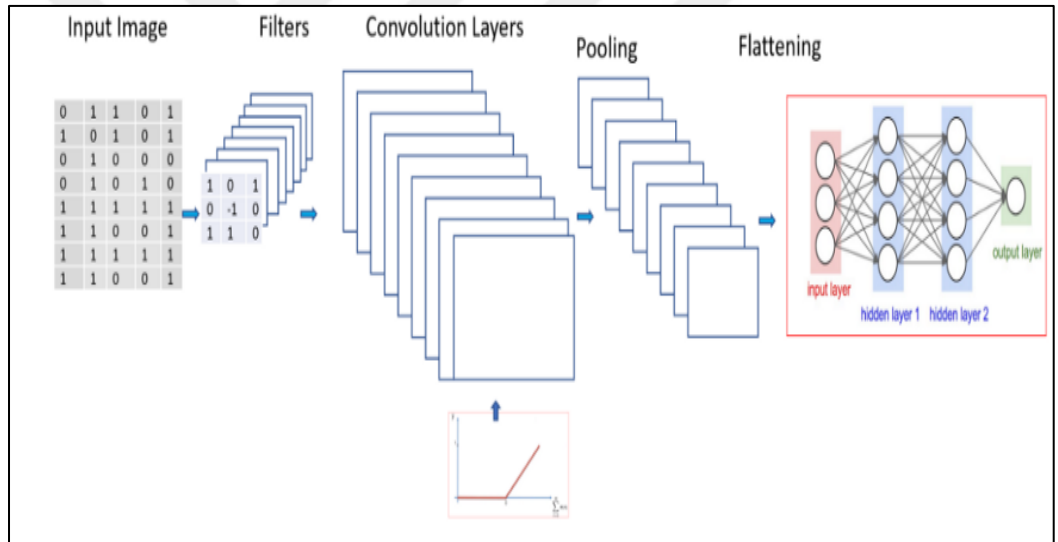
Örnek olarak, 2-katmanlı bir sinir ağı $f(x) = W_2\sigma(W_1x)$ olarak uygulanacaktır, burada W_1, W_2 matrislerdir ve element açısından doğrusal değildir (örneğin tanh). 3 katmanlı bir ağ, $f(x) = W_3\sigma(W_2\sigma(W_1x))$ biçimine sahip olacaktır. Doğrusal olmayan bir kimlik işlevi ise, tüm sinir ağının doğrusal bir işleve (temsili güçte) indirgendiği bilinmektedir. Doğrusal olmayanlar için ortak ayarlar tanh, sigmoid işlevi $1 / (1 + e^{-x})$ ve lineer doğrusal birim (ReLU) $(0, x)$ değerinin maksimumudur. Sinir ağının son katmanının normal olarak doğrusal olmasına dikkat edilmelidir. Ayrıca, 1 katmanlı bir sinir ağının basit bir doğrusal dönüşüm olduğu unutulmamalıdır.

Biyolojik ilham: Sinir ağları için biyolojik ilham tarihsel olarak biyolojik bir nöronun kaba bir modelinden ortaya çıkmıştır. Özellikle, ağırlık matrislerinin her bir sırası W bir nöronu ve onun girişine olan sinaptik bağlantı kuvvetlerini modellemektedir; pozitif ağırlıklar eksantiftir, negatif ağırlıklar inhibe edicidir ve sıfır ağırlığı hiçbir bağımlılığı temsil etmez. Girişlerin ağırlıklı toplamı hücre gövdesine ulaşır. Orjinal model, ağırlıklı toplamı $[0,1]$ aralığına sıkıştıran bir sigmoid aktivasyon fonksiyonu kullanılır ve nöronun ateşleme hızı olarak yorumlanır. Bu yorumda, bir ağırlık matrisi W ve müteakip sigmoid doğrusal olmayan fonksiyonun matris çarpımı, paralel olarak, bir katmanda düzenlenmiş birkaç nöronun ateşleme oranlarını belirler.

3. YÖNTEMLER

3.1 Evrişimli Sinir Ağları (CNN)

Evrişimli ağları, geleneksel sinir ağlarına çok benzemektedir: bunlar, öğrenilebilir ağırlık ve yanlılığa (bias) sahip nöronlardan oluşur. Her nöron, bazı girdiler alır, bir iç çarpımı yapar ve bunu isteğe bağlı olarak doğrusal olmayan bir şekilde takip eder (LeCun 1998). Tüm ağ hala tek bir ayırt edilebilir skor işlevi ifade eder: bir uçtaki ham görüntü piksellerinden diğer tarafta sınıf puanları hesaplanır. Son (tamamen bağlı) katman üzerinde bir kayıp fonksiyonuna (ör. SVM / Softmax) bağlanır. Sınıflandırma için sinir ağlarında kullanılan yapılar CNN içinde hala geçerlidir.

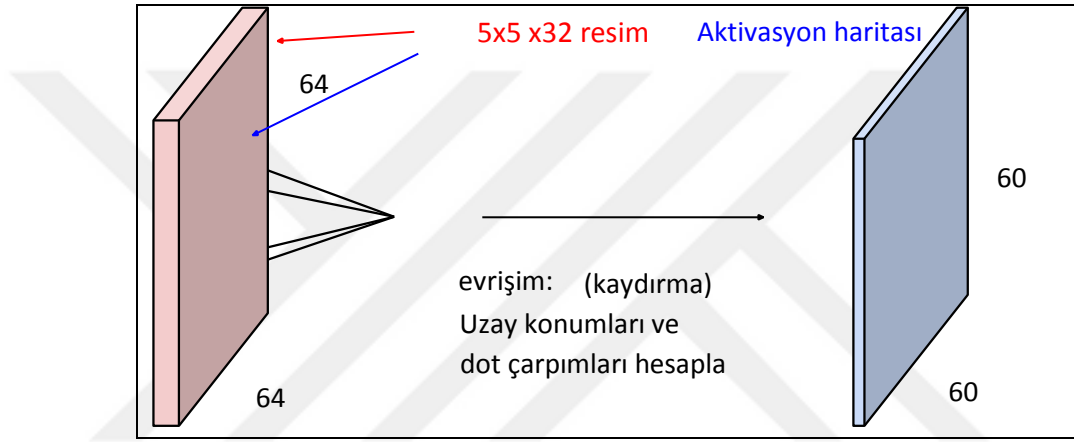


Şekil 3.1 Evrişimli sinir ağı basit model şeması

CNN metodunun resim tanıma, bölümlendirme ve tarama işlemlerinde mevcut modeller içerisinde en iyi sonuçları elde ettiği görülmüştür.

CNN, bazı uzamsal topolojiyle (örneğin görüntüler, metindeki karakter dizileri veya 3D verileri) veri işlemek için özel olarak tasarlanmış sinir ağı mimarileridir. Lopez-Garnier, Sheen ve Zimic'e (2019) göre, CNN giderek artan miktarda veri kümelerini kullanarak sınıflandırma ve algılama yeteneklerini artırdı. CNN biyolojiden ilham alır ve bilgi

katmanlari yoluyla öğrenilen pikselleri kullanarak görüntülerin konumu hakkında varsayımlar yapmak için hiyerarşik yöntemler kullanır. Katmanlar, daha yüksek katmanların karmaşık özelliklerinin algılanmasına yardımcı olur. Şekil 3.1 de basit model şeması görülen evrişimli sinir ağı diğer görevlerin yanı sıra, görüntü sınıflandırılması, tanınması ve işlenmesinde uzmanlardan daha iyi performans gösterme yeteneklerine sahip olduğunu ileri sürmektedir (Albawi vd. 2017). CNN için her bir girdi örneği x çok boyutlu bir dizidir (tensör). Asagidaki 64x64 renkli resim buna bir örnektir.



Şekil 3.2 Evrişimli ağ girdisi, filtre ve aktivasyon haritası örneği

Şekil 3.2' de filtre gezdirme adımı 1 olan ve giriş dolgusu olmayan bir 64X64X3 giriş dizisi üzerinde 5X5 filtrenin dönüştürülmesi gösterilmektedir. Filtreler her zaman uzamsal olarak küçüktür (5'e karşı 64), ancak her zaman giriş dizisinin tam derinliğini kapsar. 64 X 64 girişte 32 filtre vardır, bu nedenle kıvrım 60 X 60 aktivasyon haritası üretir, burada her bir eleman 5X5 için farklı filtre ile giriş arasındaki bir nokta ürünün sonucudur. Bir evrişimli katmanın sadece biri değil, her biri aynı Şekilde ve bağımsız olarak uygulanan ve kendi aktivasyon haritalarıyla sonuçlanan bir dizi farklı filtrenin sonucudur. 240 x 240 x 3 tensör (3 renk kanalı için kırmızı, yeşil, mavi) ile ifade edilir. Bir ses spektrogramı, $t = 1, \dots, 1000$ 'den herhangi bir zamanda 128 frekanstan herhangi birinin genliğini gösteren 1.000 X 128 büyüklüğünde bir dizi olabilir. Bazı cümle karakter düzeyinde 112 X 30 olarak temsil edilebilir, bu da 30 olası karakterden hangisinin cümledeki 112 pozisyondan herhangi birini işgal ettiğini gösterir. Bu vakaların birçoğunda girdi boyutlandırması yüksektir ve basit sinir ağlarında olduğu

gibi tamamen bağılı katmanları kullanmak işlem süresini ve hesaplama maliyetlerini artırır. Bu durumlarda, belirli yerel bağlantı ve makul parametre paylaşım şemaları kullanan yapay sinir ağı mimarileri tasarlamak tercih edilir. Bir evrişimli Sinir ağının temel hesaplamalı yapı taşı, bir girdi tensörünü alan ve girdiyi bir dizi filtre ile konvolüsyonlu hale getirerek bir çıkış tensörü üreten konvolüsyon katmanıdır (veya CONV katmanı). Bu tezde öncelikle görüntüleri işlemek üzerine çalışıldı. Girişin 240 renk yüksekliğiyle ve yani $240 \times 240 \times 3$ tensör X ile renkli bir görüntü örneği ele alınmıştır. $5 \times 5 \times 3$ filtre 75 parametresi olan filtreyi, giriş tensörünün tüm uzamsal konumlarında kaydırarak ve her bir konumdaki küçük bir X yığını ile filtre w arasında bir nokta ürün hesaplayarak konvolüsyon yapılır. Sonuç, bu durumda 236×236 boyutlarına sahip olan bir aktivasyon haritası olur. Çıktı boyutunu kontrol etmek için girdiyi sıfırlarla sınırlamak da yaygındır. Örneğin, bu durumda 2 kalınlığında sıfırlardan oluşan bir kenarlık ile dolgu yapmak, $240-240$ çıkış aktivasyon haritasına yol açar. Bazı adımlarla filtreler uygulamak da mümkündür. Örneğin, dolgu olmayan 2 adımı $240 \times 240 \times 3$ giriş tensörüne $3 \times 3 \times 3$ filtresini uygulamak, bunun yerine 118×118 boyutunda bir çıkış aktivasyon haritası verir. Son olarak, CONV katmanı sadece tek bir filtreye değil, tüm filtrelere sahiptir. Örneğin, 32 filtre kullanırsak, her biri kendi 118×118 aktivasyon haritasını hesaplar ve bunlar nihai çıkış tensörünü üretmek için istiflenir (örneğin $118 \times 118 \times 32$ ebadında). Sezgisel olarak, her filtre giriş tensörünün belirli yerel özelliklerini “arama” kapasitesine sahiptir ve filtreleri oluşturan parametreler geri yayılım ile eğitilir. Derin CNN, birden fazla katmanı içeren ve görüntülerin sınıflandırılmasında başarılı olan denetimli bir derin öğrenme teknolojisidir (LeCun vd. 2015). Birden çok düzeyde görüntüler gibi veri kümelerini sunma becerileri, derin öğrenmenin önemli bir avantajıdır. Derin CNN nesneleri daha yüksek düzeylerde, orta düzeydeki nesnelerin parçalarını ve daha düşük düzeylerde lekeleri, değerler ve kenarları içeren pikselleri içerebilir. Çoklu katmanlar veri analizini kolaylaştırır ve doğru sonuçlar üretme yeteneklerini artırır.

Daha genel olarak, görüntüler için kıvrımlı bir katman (yani, üç uzamsal boyuta sahip giriş tensörleri varsayılarak): $W_1 \times H_1 \times D_1$ boyutunda bir tensörü kabul eder. 4 hiper parametre gerektirir, bunlar K filtrelerinin sayısı, uzaysal genişlikleri F , uygulandıkları adım S ve girişin kenarlarındaki sıfır dolgu miktarı, P dir. Katlamalı tabaka $W_2 \times H_2 \times$

D_2 boyutunda bir çıktı hacmi üretir, $W_2 = (W_1 - F + 2P) / S + 1$, $H_2 = (H_1 - F + 2P) / S + 1$ ve $D_2 = K$. Her filtredeki parametre sayısı, toplam $(F * F * D_1) * K$ ağırlık ve K sapması için $F * F * D_1$ 'dir. Özellikle, filtrelerin uzamsal boyutunun boşlukta küçük olduğunu $(F \times F)$, ancak her zaman giriş tensörünün (D_1) tam derinliğini geçtiği unutulmamalıdır.

Biyolojik ilham ve yorum: Evrişimli katman, nöron benzetmesi ile aşağıdaki gibi yorumlanabilir. Belirli bir konumda bir filtreye sahip bir nokta çıktının sonucu, yalnızca giriş dizisinin belirli bir kısmına bağlantısı olan bir nöronun (aktivasyon öncesi) çıktısını temsil eder. Dahası, her filtreyi girdi üzerinde kaydırılan ve her yerde aynı ağırlıkları kullanıldığından, bir etkinleştirme haritasında komşu nöronların hepsinin aynı ağırlıkları kullandığı bir parametre paylaşım şemasını modellendi. Sonuç, her bir konvolüsyonlu katmandaki parametre sayısında büyük bir azalma görülür ve bu da aşırı sığmaya yardımcı olur. Belirli bir örnek olarak, 2 dolgu ve 1 adım kullanarak, $3 \times 3 \times 3$ boyutunda 32 filtreli bir evrişim tabakası ile işlenen $240 \times 240 \times 3$ girişini ele alındı. Çıktı bu durumda 240×240 olacaktı ve $\times 32$, tüm mekânsal konumlardaki tüm filtrelerin tetiklendiğini gösterir. Bu katman bu nedenle toplam $(3 \times 3 \times 3 \times 32 + 32 = 896)$ toplam parametre (ağırlık ve sapma) kullanılarak $240 \times 240 \times 32 = 1.843.200$ çıktı üretmektedir. Buna karşılık, eğer bu gizli tabakadaki aynı sayıda nöron ile tamamen bağlı bir katman olsaydı, $1.843.200 \times (240 \times 240 \times 3 + 1) = 318.506.803.200$ parametre (ağırlık ve yanlılık) kullanılırken astronomik değerlerde sonuçların hesaplanabildiği veya saklanabileceği varsayılsa bile, büyük ölçüde fazla taklit edilmesi beklenmektedir.

Katmanları birleştirme: Evrişimli katmanlara ek olarak, aşırı öğrenmeyi daha fazla kontrol etmek için, sabit bir alt örnekleme dönüşümüyle (yani herhangi bir parametre olmadan) temsilin boyutunu azaltan havuz katmanlarının kullanılması yaygındır. Özellikle, havuzlama katmanları her bir kanalda (aktivasyon haritası) bağımsız olarak çalışır ve bunları uzaysal olarak altörnekler. Yaygın olarak kullanılan bir ayar, her bir filtrenin maksimum işlemi (yani 4 sayıdan fazla) hesapladığı, adım 2 ile 2×2 filtre kullanmaktır. Sonuç ise bir giriş tensörünün hem genişlik hem de yükseklik açısından tam olarak 2 faktör ile küçültülmesi ve temsil boyutunun, bazı yerel uzamsal bilgilerini kaybetme pahasına 4 faktör azaltılmasıdır.

ConvNet mimarileri, evrişimli katmanların istiflenmesi ve muhtemelen mimarinin hesaplama karmaşıklığını kontrol etmek için havuzlama katmanlarının getirilmesiyle evrişimli bir ağ inşa eder. Görüntüleri işleyen tipik bir evrişimli sinir ağı mimarisi [Girdi, [CONV, CONV, POOL] x 3, FC, FC] biçimini alabilir. Burada Girdi, bir grup görüntüden oluşan bir tensörü temsil eder (örneğin, 80 adet 120x120 renkli görüntü grubu için [80x120x120x3]). CONV, 1 dolgu ve 1 adım ile uygulanan 3x3 filtreli bir evrişim katmanıdır, 'POOL' tipik bir 2x2 filtre anlamına gelir, adım 2 ile maksimum havuzlama katmanı ve 'FC' tamamen bağlı katmanlardır. Burada sonuncusu bir softmax sınıflandırmasından hemen önce farklı sınıfların loglarını hesaplanır. Bu mimaride, girdinin uzamsal boyutu her bir POOL katmanından sonra hem genişlik hem de yükseklik açısından 2 kat azaltılır, bu nedenle üçüncü POOL katmanından sonra genişlik ve yükseklik boyunca temsilin uzamsal boyutu 15 x 15 olmaktadır.

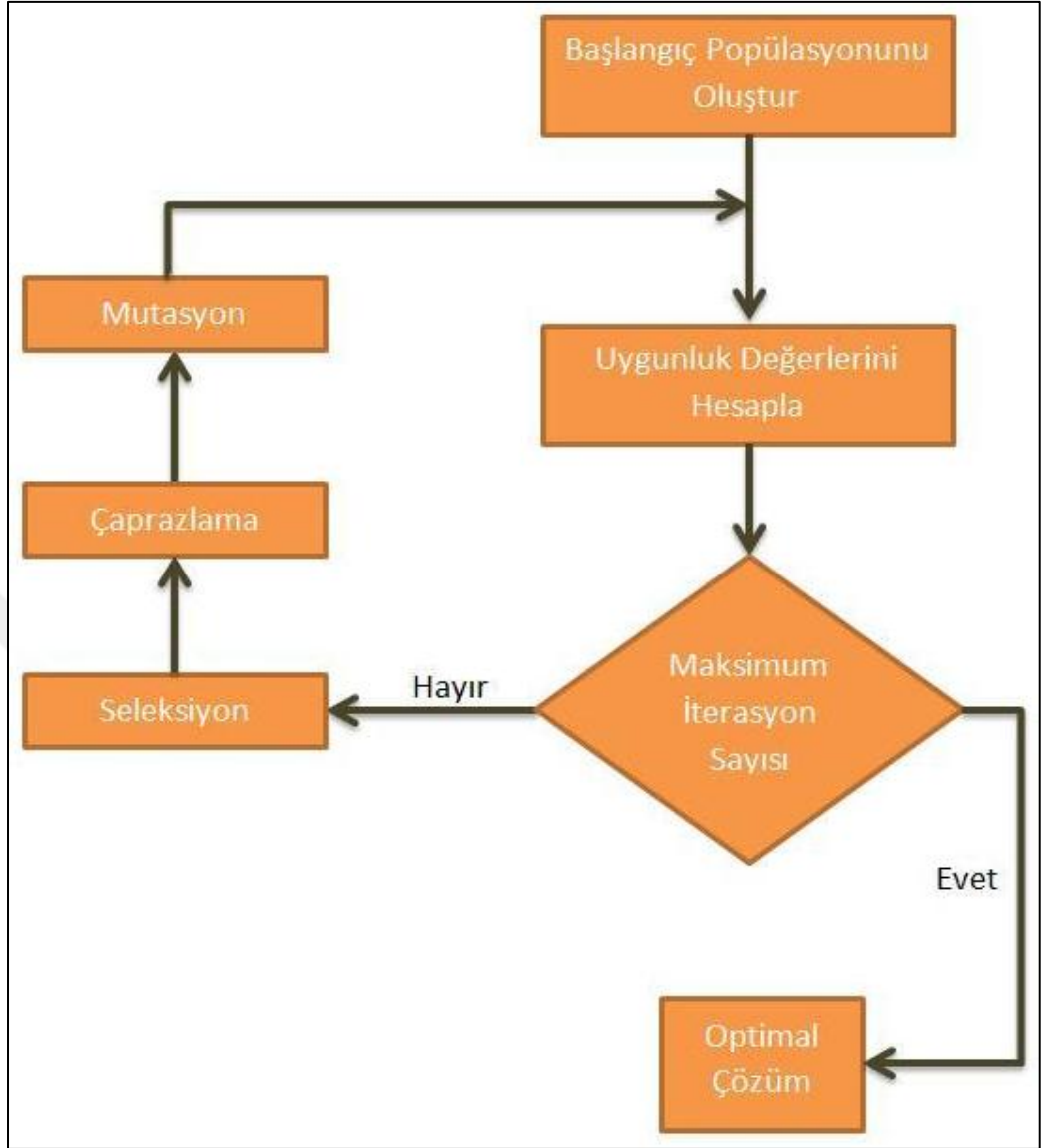
3.2 Genetik Algoritma

Bilgisayar bilimi ve yöneylem araştırması olan Genetik Algoritma (GA), Darwin'in evrim teorisi kavramına dayanarak 1960 yılında John Holland tarafından ortaya konulan doğal seçilim sürecinden esinlenen bir meta-sezgidir. GA, en çok mutasyon, geçiş ve seçim gibi biyo-ilham alan operatörlere güvenerek iyileştirme ve arama sorunlarına yüksek kaliteli çözümler üretmek için kullanılır (Miller vd. 1989).

Genetik algoritmada, bir iyileştirme problemine yönelik bir çözüm topluluğu (popülasyon, bireyler, üyeler vb.) daha iyi çözümlere doğru evrilir. Gerçek veya ikili sayılarla veya başka bir kodlamayla kodlanan her çözüm, değiştirilebilen bir kromozomla temsil edilir (Omid ve Iddo 2014). Yinelemeli bir süreç olan evrim, genellikle rastgele üretilen bireylerin bir topluluğundan başlar ve her bir yinelemedeki topluluk bir nesil olarak adlandırılır. Her nesilde, topluluktaki her bir bireyin uygunluk ölçümü değerlendirilir; bu genellikle çözülmekte olan iyileştirme problemindeki objektif fonksiyonun değeridir. Daha uygun bireyler, yeni neslin oluşumu için değiştirilecek (yeniden birleştirilecek ve muhtemelen rastgele mutasyona uğrıtılacak) mevcut topluluktan (yüksek şansla) seçilir. Yeni nesil aday çözümler daha sonra algoritmanın bir sonraki yinelemesinde kullanılır. Genel olarak algoritma, maksimum sayıda nesil

geçtiğinde veya topluluk için tatmin edici bir uygunluk seviyesine ulaşıldığında sona erer (Kattan vd. 2010).

Tipik bir genetik algoritma, çözüm bilgi alanının genetik bir temsilini ve çözüm bilgi alanı değerlendirmek için bir uygunluk fonksiyonunu gerektirir. Her çözümün standart temsili bir dizi kullanmaktır. Bu genetik temsili kolaylaştıran ana özellik, parçalarının sabit boyutlarından dolayı kolayca hizalanmasıdır, bu da basit geçiş işlemlerini mümkün kılar. Genetik temsil ve uygunluk fonksiyonu tanımlandıktan sonra, algoritma bir çözüm topluluğunu başlatmak ve daha sonra seçim, çaprazlama ve mutasyon operatörlerinin tekrar tekrar uygulanması yoluyla onu geliştirmek için ilerler (Zhen-Guo vd. 2011). Şekil 3.3 de genetik algoritma işlem adımları görülmektedir.



Şekil 3.3 Genetik algoritma işlem süreçleri

İlk topluluk genellikle rastgele üretilen birkaç yüz veya binlerce olası çözüm içerir. Bu, tüm olası çözümlerin istenen bir şekilde örneklenmesini sağlar. Bazı performans değerlendirme kurallarına göre, temsil edilen çözümün kalitesini ölçmeye dayanarak her üyeye bir uygunluk değeri atamak için çözümlerin genetik gösterimleri üzerinde bir uygunluk fonksiyonu tanımlanır. Uygunluk fonksiyonu probleme bağlıdır, bu nedenle doğruluk parametresinin tespiti, çalışan bir genetik algoritmanın ayarlanmasında önemli bir adım haline gelir. Her ardışık nesil sırasında, mevcut topluluğun tamamı veya bir kısmı yeni bir neslin yeniden üretilmesi için seçilir. Üremeye dâhil olan çözümler, daha uygun çözümlerin tipik olarak seçilme olasılığının yüksek olduğu uygunluk tabanlı bir

süreçle seçilir. Seçim stratejilerine örnek olarak uygunluk oranı seçimi (veya en yaygın olarak rulet tekerleği seçimi olarak bilinir), turnuva seçimi ve stokastik evrensel örnekleme verilebilir (Montana 1995).

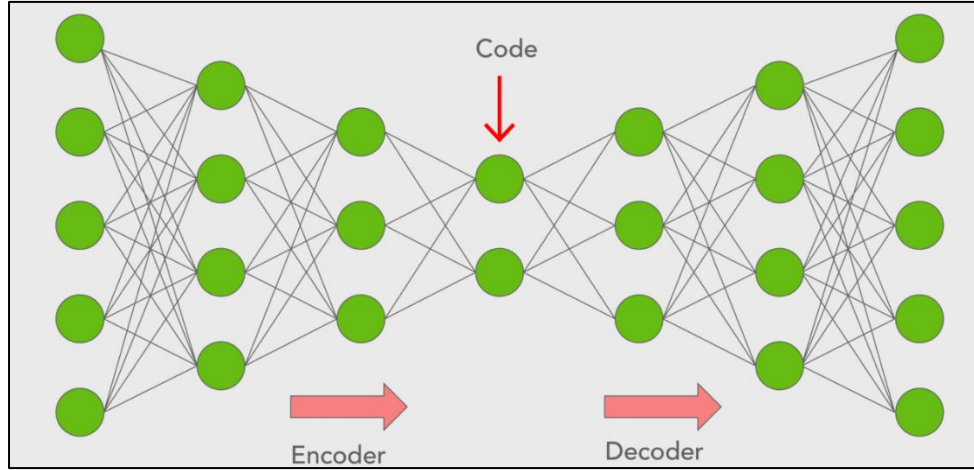
Bir sonraki adım, çaprazlama ve mutasyon gibi genetik operatörlerin birleşik kullanımı yoluyla seçilen çözümlerden yeni nesil bir çözüm popülasyonu oluşturmaktır. Her yeni çözüm üretimi için, önceden seçilmiş uygun bireylerin havuzundan üreme için bir çift "ana" çözüm (veya bazen sadece bir çözüm) seçilir. Bahsedilen üretme yöntemlerini kullanarak bir "çocuk" çözüm üreterek, çoğunlukla "ebeveyn" lerinin özelliklerinin çoğunu paylaşan yeni bir çözüm yaratılır. Yine, her yeni çocuk için yeni ebeveynler seçilir ve bu süreç uygun büyüklükte yeni bir çözüm topluluğu oluşana kadar devam eder. Yeni nesil çözümler üretmek için, bu yeni yetiştirilen çocuklar mevcut neslin üyelerinin bir kısmının veya tamamının yerini alabilir (Palmes P. vd. 2005). Bu süreçler nihayetinde önceki nesilden farklı olan yeni nesil kromozom topluluğu ile sonuçlanır. Genel olarak, bir topluluğun ortalama zindeliği bu süreç ile artar, çünkü üreme için sadece en iyi organizmalar ve az miktarda daha az uygun çözümler seçilir. Bu daha az uygun çözümler, ebeveynlerin genetik havuzunda genetik çeşitliliği garanti eder ve bu nedenle sonraki çocuk nesillerinin genetik çeşitliliğini sağlar. Geçiş ve mutasyon ana genetik operatörler olarak bilinmesine rağmen, genetik algoritmada yeniden gruplama, kopyalama veya göç gibi diğer operatörleri kullanmak mümkündür (Omid ve Iddo 2014). Üzerinde çalışılan problem sınıfı için ayarların makul bir değer kümesini bulmak için, mutasyon olasılığı, geçiş olasılığı ve topluluk büyüklüğü gibi parametrelerin uygun şekilde ayarlanması gerekir. Çok düşük bir mutasyon oranı genetik kaymaya yol açabilir. Çok yüksek bir rekombinasyon hızı genetik algoritmanın erken yakınsamasına yol açabilir. Çok yüksek bir mutasyon oranı iyi çözümlerin kaybına yol açabilir. Bu nedenle, bu parametrelerin ayarlanmasının doğru yapılandırılması, iyi bir genetik algoritma yapmak için önemli bir konudur. Yukarıdaki işlem bir miktar fesih durumuna ulaşana kadar tekrarlanır. Yaygın sonlandırma koşulu, minimum ölçütleri karşılayan, sabit sayıda nesile ulaşan veya en üst düzey çözümün uygunluğunda bir kümeye ulaşan bir çözüm bulmaktır (Oullette vd. 2004).

3.3 Otomatik Kodlayıcı

Otomatik kodlayıcı, çıktı katmanının giriş katmanı ile aynı boyuta sahip olduğu bir tür sinir ağıdır. Daha basit bir ifadeyle, çıktı katmanındaki çıktı birimlerinin sayısı, girdi katmanındaki girdi birimlerinin sayısına eşittir. Bir otomatik kodlayıcı, girdiden çıktıya veriyi denetimsiz bir şekilde çoğaltır ve bu nedenle bazen çoğaltıcı sinir ağı olarak adlandırılır (Tschannen vd. 2018). Otomatik kodlayıcılar, girdinin her boyutunu ağıdan geçirerek yeniden yapılandırır. Girdiyi çoğaltma amacıyla bir sinir ağı kullanmak önemsiz görünebilir, ancak çoğaltma işlemi sırasında girdinin boyutu daha küçük temsiline indirgenir. Sinir ağının orta katmanları, giriş veya çıkış katmanlarına kıyasla daha az sayıda birime sahiptir. Bu nedenle, orta katmanlar, girdinin azaltılmış temsili tutar. Çıktı, girdinin bu azaltılmış temsili üzerinden yeniden oluşturulur (Wang vd. 2014).

Bir otomatik kodlayıcı üç bileşenden oluşur:

Birincisi kodlayıcı; girdiyi bir gizli alan temsiline sıkıştıran ve girdi görüntüsünü sıkıştırılmış bir temsil olarak küçültülmüş bir boyutta kodlayan, ileri beslemeli, tamamen bağlı bir sinir ağıdır. Sıkıştırılmış görüntü, orjinal görüntünün bozulmuş halidir. İkinci bileşen kod; Ağın bu kısmı, kod çözücüye beslenen girişin azaltılmış temsili içerir. Üçüncü bileşen kod çözücü de kodlayıcı gibi ileri beslemeli bir ağıdır ve kodlayıcıya benzer bir yapıya sahiptir Şekil 3.4. Bu ağ, girişi koddan orjinal boyutlarına geri döndürmekten sorumludur (Zhang vd. 2020).



Şekil 3.4 Otomatik kodlayıcı, kodlayıcı, kod ve kod çözücü mimarisi

İlk olarak girdi, sıkıştırıldığı ve kod adı verilen katmanda depolandığı kodlayıcıdan geçer, ardından kod çözücü, orjinal girdiyi koddan açar. Otomatik kodlayıcının temel amacı, girdiyle aynı çıktıyı elde etmektir. Kod çözücü mimarisinin kodlayıcının ayna görüntüsü olduğu görülmektedir. Bu bir gereklilik değildir, ancak genellikle böyledir. Tek gereksinim, girdinin ve çıktının boyutunun aynı olmasıdır. Oto kodlayıcının yapabileceği görevlerden bazıları şu şekilde sıralanabilir;

3.3.1 Veri sıkıştırma

Otomatik kodlayıcılar veri sıkıştırma için tasarlanmış olsalar da pratik durumlarda bu amaç için pek kullanılmazlar. Kayıplı sıkıştırma otomatik kodlayıcının çıktısı, girdi ile tam olarak aynı değildir, yakın fakat bozulmuş bir temsildir. Otomatik kodlayıcılar, yalnızca üzerinde eğitildiklerine benzer verileri anlamlı bir şekilde sıkıştırılabilirler. Verilen eğitim verilerine özgü özellikleri öğrendikleri için jpeg veya gzip gibi standart bir veri sıkıştırma algoritmasından farklıdırlar (Cheng vd. 2018). Bu nedenle, elle yazılmış rakamlarla eğitilmiş bir otomatik kodlayıcının manzara fotoğraflarını sıkıştırmasını beklenemez.

Jpeg, LZMA, LZSS gibi daha verimli ve basit algoritmalar olduğundan, otomatik kodlayıcılar genellikle sıkıştırma için kullanılmaz. Otomatik kodlayıcıların popüler kullanım alanları genellikle, görüntü gürültü azaltma ve boyut azaltma olarak

sıralanabilir. Görüntü Gürültüsü; otomatik kodlayıcılar, görüntüleri gürültüden arındırmada çok iyidir. Bir görüntü bozulduğunda veya içinde biraz parazit olduğunda bu görüntüye gürültülü görüntü denir. Görüntünün içeriği hakkında doğru bilgiyi elde etmek için görüntü gürültü azaltma işlemi uygulanır.

Otomatik kodlayıcılar, girişi kod adı verilen orta katmanda depolanan azaltılmış bir temsile dönüştürür. Bu, girdiden gelen bilgilerin sıkıştırıldığı yerdir ve bu katman modelden çıkarılarak her bir düğüm artık bir değişken olarak ele alınabilir. Böylece, kod çözücü parçasının yok edilmesiyle, çıktının kod katmanı olduğu boyut azaltma için bir otomatik kodlayıcının kullanılabileceği sonucuna varabiliriz (Gondara 2016).

3.3.2 Özellik çıkarma

Otomatik kodlayıcıların kodlama kısmı, yeniden yapılandırma hatasını azaltma sürecinde giriş verilerinde bulunan önemli gizli özelliklerin öğrenilmesine yardımcı olur. Kodlama sırasında, orjinal özelliklerin yeni bir bileşim oluşturulur (Meng vd. 2017).

3.3.3 Görüntü oluşturma

Oto kodlayıcı, henüz model tarafından görülmeyen görüntüleri oluşturmak için kullanılan bir üretken modeldir. Buradaki fikir, yüz veya manzara görüntüleri gibi girdi görüntüleri verildiğinde, sistemin benzer görüntüler oluşturacağıdır (Chaitanya vd. 2017).

3.4 SVM (Destek Vektör Makinaları)

Destek vektör makinaları, sınıflandırma, regresyon ve aykırı değerlerin tespiti için kullanılan bir dizi denetimli öğrenme yöntemidir. Bunların hepsi makina öğreniminde ortak görevlerdir. Milyonlarca görüntüye dayalı olarak kanserli hücreleri tespit etmek için bunları kullanabilir veya iyi ayarlanmış bir regresyon modeliyle gelecekteki verileri

tahmin etmek için kullanabilir. Destek vektörü sınıflandırmasının bir uzantısı olan destek vektörü regresyonu gibi belirli makina öğrenimi sorunları için kullanılabilecek belirli SVM türleri vardır. Burada akılda tutulması gereken en önemli şey, bunların yalnızca mümkün olan en doğru cevabı mümkün olan en kısa sürede verecek şekilde ayarlanmış matematik denklemler olmasıdır (Noble 2006).

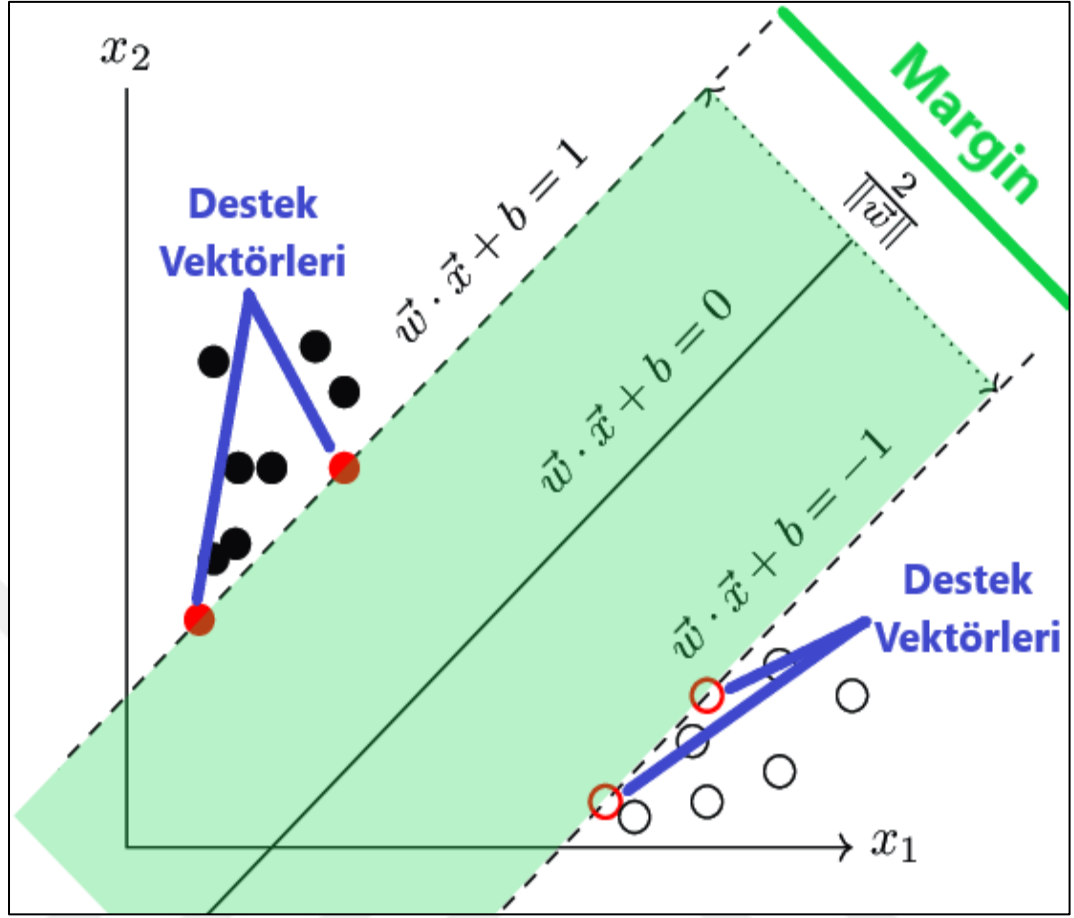
SVM'ler, tüm sınıfların en yakın veri noktalarından uzaklığı maksimize eden karar sınırını seçme biçimleri nedeniyle diğer sınıflandırma algoritmalarından farklıdır. SVM'ler tarafından oluşturulan karar sınırı, maksimum marj sınıflandırıcısı veya maksimum marj hiper düzlemi olarak adlandırılır.

Basit bir doğrusal SVM sınıflandırıcı, iki sınıf arasında düz bir doğru oluşturarak çalışır. Bu, hattın bir tarafındaki tüm veri noktalarının bir kategoriye temsil edeceği ve hattın diğer tarafındaki veri noktalarının farklı bir kategoriye yerleştirileceği anlamına gelir. Bu, aralarından seçim yapabileceğimiz sonsuz sayıda doğru olabileceği anlamına gelir (Suthaharan 2016).

Doğrusal SVM algoritmasını 'k' en yakın komşular gibi diğer bazı algoritmalarından daha iyi yapan şey, veri noktalarını sınıflandırmak için en iyi aralığı seçmesidir. Verileri ayırıp veri noktalarından mümkün olduğunca en uzak olan çizgiyi seçer (Widodo ve Yang 2007).

İki boyutlu bir örnek, tüm makina öğrenimi jargonunu anlamlandırmaya yardımcı olur. Temel olarak bir ızgara üzerinde bazı veri noktaları olduğunu varsayalım. Bu veri noktalarını uymaları gereken kategoriye göre ayırmaya çalışıldığında, yanlış kategoride herhangi bir veriye sahip olmak istenilmez. Bu, diğer veri noktalarını ayrı tutan en yakın iki nokta arasındaki çizgiyi bulmaya çalışıldığı anlamına gelir (Pisner ve Schnyer 2020).

Böylece en yakın iki veri noktası o çizgiyi bulmak için kullanılacak destek vektörlerini verir. Bu çizgiye karar sınırı denir (Şekil 3.5).



Şekil 3.5 SVM destek vektörleri ve marjin (Akca 2020)

Her biri farklı şeyler için kullanılan iki farklı SVM türü vardır:

Basit SVM: Tipik olarak doğrusal regresyon ve sınıflandırma problemleri için kullanılır.

Cekirdek SVM: İki boyutlu bir alan yerine bir hiper düzlemde daha fazla özellik eklenebildiğinden doğrusal olmayan veriler için daha fazla esnekliğe sahiptir (Joachims 1999).

SVM'ler, el yazısı tanıma, izinsiz giriş algılama, yüz algılama, e-posta sınıflandırma, gen sınıflandırma gibi uygulamalarda ve web sayfalarında kullanılır. Bu, makina öğreniminde SVM tercih edilme nedenlerinden biridir. Doğrusal ve doğrusal olmayan veriler üzerinde başarılı şekilde sınıflandırma yapabilir (Agarap 2017).

SVM'lerin tercih edilmesinin bir başka nedeni de manuel olarak çok fazla dönüşüm yapmanıza gerek kalmadan verileriniz arasında karmaşık ilişkiler bulabilmeleridir. Onlarca ila yüz binlerce özelliğe sahip daha küçük veri kümeleriyle çalışırken iyi bir alternatiftir. Küçük, karmaşık veri kümelerini işleme yetenekleri nedeniyle, diğer algoritmalarla karşılaştırıldığında genellikle daha doğru sonuçlar bulurlar (Meyer vd. 2003).

SVM'leri kullanmanın bazı artıları finansal veya tıbbi veriler gibi birden fazla özelliğe sahip veri kümelerinde etkilidir. Özellik sayısının veri noktası sayısından fazla olduğu durumlarda daha etkilidir. Destek vektörleri adı verilen karar işlevinde, belleği verimli hale getiren bir eğitim noktaları alt kümesini kullanır. Karar fonksiyonu için farklı çekirdek fonksiyonları belirlenebilir. Ortak çekirdekler kullanılabilir, ancak özel çekirdekler belirlemek de mümkündür (Byvatov ve Schneider 2003).

Eksileri ise eğer özniteliklerin sayısı veri noktalarının sayısından çok daha fazlaysa, çekirdek fonksiyonlarını ve düzenleme terimini seçerken aşırı öğrenmeden kaçınmak çok önemlidir. SVM'ler doğrudan olasılık tahminleri sağlamaz. Bunlar pahalı beş katlı çapraz doğrulaması kullanılarak hesaplanır. Yüksek eğitim süresi nedeniyle küçük numune setlerinde en iyi sonucu verir. SVM'ler herhangi bir sayıda çekirdek kullanabildiğinden, bunlardan birkaçını bilmek önemlidir (Ukil 2007).

3.4.1 SVM çekirdek işlevleri

Doğrusal: Bu tür sınıflandırma problemlerinin çoğu doğrusal olarak ayrılabilir olduğundan, bunlar genellikle metin sınıflandırması için önerilir.

Doğrusal çekirdek, çok sayıda özellik olduğunda ve metin sınıflandırma problemlerinin birçok özelliği olduğunda daha verimli çalışır. Doğrusal çekirdek işlevleri, diğerlerinden daha hızlıdır ve optimize etmek için daha az parametreye ihtiyaç duyar. Doğrusal çekirdeği tanımlayan fonksiyon: $f(x) = w^T * x + b$ şeklindedir. Bu denklemde w , minimize etmek istediğiniz ağırlık vektörüdür, x , sınıflandırmaya çalıştığınız veridir ve

b, eğitim verilerinden tahmin edilen çekirdek katsayıdır. Bu denklem, SVM'nin döndürdüğü karar sınırını tanımlar (Kim vd. 2003).

Polinom: Polinom çekirdeği, diğer çekirdekler kadar hesaplama açısından verimli olmadığı ve tahminleri doğru olmadığı için pratikte çok sık kullanılmaz. Bir polinom çekirdeğinin işlevi: $f(x_1, x_2) = (a + x_1^T * x_2)^b$ şeklinde formülize edilir.

Bu, tercih edilecek daha basit polinom çekirdek denklemlerinden biridir. $f(x_1, x_2)$, verileri ayıracak polinom karar sınırını temsil eder. x_1 ve x_2 verileri temsil eder (Pradhan 2012).

Gauss Radyal Temel Fonksiyonu (RTF): SVM'lerde en güçlü ve yaygın olarak kullanılan çekirdeklerden biri de Gauss Radyal Temel Fonksiyonudur. Genellikle doğrusal olmayan veriler için tercih edilir. RTF çekirdeği için denklem: $f(x_1, x_2) = \exp(-\gamma * ||x_1 - x_2||^2)$ şeklindedir. Bu fonksiyonda, alfa bir ağırlık vektörüdür ve C, meydana gelebilecek bazı yanlış sınıflandırmaları hesaba katan bir ofset değeridir. Kullanılabilecek birçok çekirdek bulunmaktadır (Pradhan 2012). Bu belirli hata kısıtlamalarını karşılamak, eğitim süresini denemek veya hızlandırmak, ya da hiper parametreleri ayarlamak gerektiğinde yapılması gereken bir tercihtir.

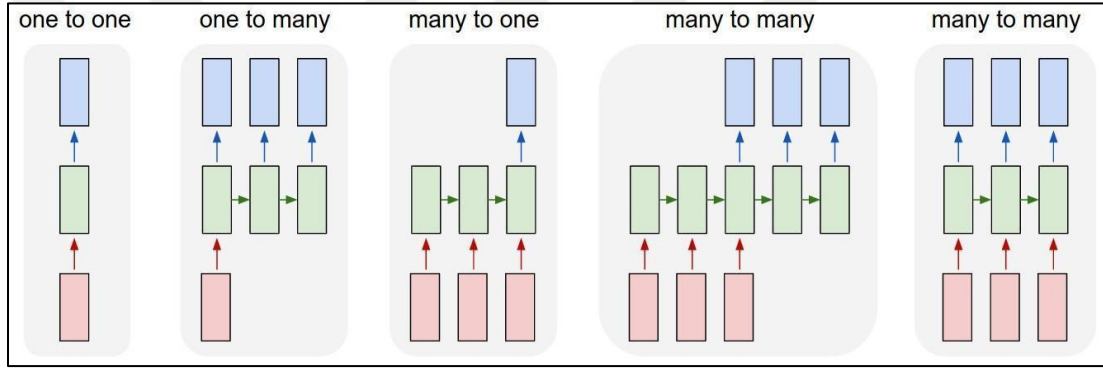
Sigmoid: Sinir ağlarında, destek vektör makinalarından daha faydalıdır, ancak ara sıra özel kullanım durumları vardır. Bir sigmoid çekirdeğin işlevi: $f(X, y) = \tanh(\alpha * X^T * y + C)$ formülü şeklindedir.

3.5 Tekrarlayan Sinir Ağları (RNN)

Birçok pratik uygulamada giriş veya çıkış boşlukları sekanslar içerir. Örneğin, cümleler genellikle bir kelime dizisi olarak modellenir, burada her kelime bir vektör olarak kodlanır (yani sabit bir kelime haznesindeki kelimenin dizisinde tek bir 1 hariç tüm sıfırların bir vektörü). Tekrarlayan bir sinir ağı, $h_t = f_\theta(h_{t-1}, x_t)$ biçiminde bir yineleme formülü kullanarak $\{x_1, ..., x_T\}$ vektörler dizisini işleyen bir bağlantı örüntüsüdür.

Ayrıca her zaman adımında aynı parametreler θ değerini kullanır, bu da rastgele sayıda vektör içeren dizilerin işlenmesine izin verir. Gizli vektör h_t , o zaman aşamasına kadar tüm vektörlerin x çalışan bir özeti olarak yorumlanabilir ve yineleme formülü bir sonraki vektöre göre özeti günceller. $h_o = \vec{0}$ Kullanmak veya h_o 'a parametre olarak davranmak ve başlangıç gizli durumunu öğrenmek yaygındır. Tekrarın kesin matematiksel formu $(h_{t-1}, x_t) \rightarrow h_t$, h_t modelden modele değişir. Geleneksel tekrarlayan sinir ağı RNN modelinin tekrarını kullanır.

$$h_t = \tanh\left(W\left(\frac{x_t}{h_{t-1}}\right)\right) \text{ Denklem} \quad (3.1)$$



Şekil 3.6 RNN modelleri bire-bir, bire-çok, çok-bir, çok-çok

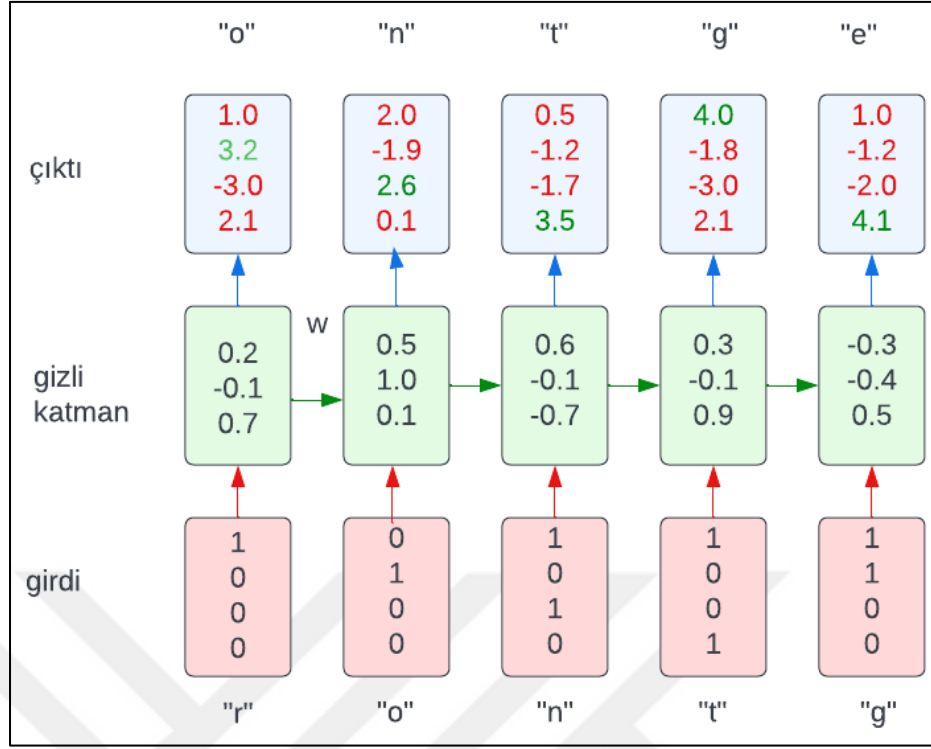
Şekil 3.6 da görüldüğü gibi sıradan bir sinir ağı (solda) bir girdi vektörü (kırmızı) alabilir, onu gizli bir katman (yeşil) boyunca dönüştürebilir ve bir çıktı vektörü (mavi) üretebilir. Bu diyagramlarda kutular vektörleri ve oklar fonksiyonel bağımlılıkları göstermektedir. Tekrarlayan sinir ağları vektör dizilerini işlememize izin verir. Bunlar: çıkışta, girişte ya da seri veya paralel olarak uygulanabilir. Önceki gizli durum h_{t-1} 'e ve mevcut girişe göre $h_t = f_{\theta}(h_{t-1}, x_t)$ sabit bir yineleme formülü kullanılarak, mevcut giriş h_t 'yi ve mevcut girişi temel alan bir dizi dâhili değişkeni manipüle eden tekrarlayan bir gizli katman ile kolaylaştırılır.

Yani, önceki gizli vektör ve akım girişi W parametreleri tarafından doğrusal olarak birleştirilir ve yukarıdaki W matrisine dönüştürülür. Bu denklemler $h_t = \tanh(W_{xh} x_t + W_{hh} h_{t-1})$ kısalık için ek sapma vektörünü ihmal eder. Tanh doğrusal olmama durumu da

ReLU ile deđiřtirilebilir. Giriř vektörleri x_t , D boyutuna ve gizli durumlar H boyutuna sahipse, W, $[H \times (D + H)]$ boyutunda bir matristir. Denklemi yorumlarken, her zaman adımımda yeni gizli durumlar x_t , h_{t-1} öğelerinin doğrusal bir fonksiyonudur ve geleneksel RNN basit bir forma sahiptir. Ancak, katkı etkileřimleri, giriřler ve gizli durumlar arasında zayıf bir bađlantıyı temsil eder. Sutskever vd. (2011) geleneksel RNN'nin fonksiyonel formu, geri yayılım sırasında istenmeyen dinamiklere yol açar (Wu Y. vd. 2016). Patlayan gradyan kaygısı, gradyanları bir miktar maksimum değere kırmak için bir bulgusal tarama ile hafıfletilebilir (Pascanu vd. 2013), ancak RNN'ler hala yok olan gradyan probleminden etkilenebilmektedir.

3.5.1 Uzun kısa süreli bellek LSTM

Tekrarlı, geleneksel RNN' nin sınırlamalarına yönelik olarak tasarlanmıřtır. Yineleme formülü, x_t ve h_{t-1} giriřlerinin çarpımsal etkileřimleri içeren daha hesaplamalı olarak karmařık bir řekilde etkileřime girmesine izin veren bir forma sahiptir. LSTM tekrarı, indirgemeleri zamanda daha etkili bir řekilde geriye doğru daha fazla yayan ilave adımlar kullanır (Hochreiter ve Schmidhuber 1997). Gizli durum vektörü h_t 'ye ek olarak, LSTM'ler bir bellek vektörü c_t 'yi de muhafaza eder. Her adımda LSTM, açık geçit mekanizmalarını kullanarak hücreden okumayı, yazmayı veya sıfırlamayı seçebilir. Güncellemenin kesin řekli ařađıdaki gibidir:



Şekil 3.7 RNN kullanımına bir dil modeli örneği

Örneğin eğitim sırası “röntgen” dir ve kelime haznesinin 7 karakteri vardır: Şekil 3.7 de görüldüğü gibi r, ö, n, t, g, e, n. Girişler, “ r, ö, n, t, g, e, n.” karakterlerinin kodlamasıdır ve RNN' nin her zaman adımında dizideki bir sonraki karakteri tahmin etmesini isteriz. RNN, üç boyutlu bir gizli duruma (yeşil) sahiptir ve çıkış vektörlerinde (mavi) 7 karakter vardır, bir sonraki karakter için loglar (yani puanlar) olarak yorumlanır. Kayıp işlevi ve indirgeme, doğru karakterlerin (yeşil renkle vurgulanmış) lojitlerin daha yüksek, diğer lojitlerin (kırmızı renkte) daha düşük olmasını teşvik eder.

$$\left(\frac{if}{og}\right) = \left(\frac{sigm}{sigm tanh}\right) W \left(\frac{x_t}{h_{t-1}}\right) \text{ Denklem} \quad (3.2)$$

$$c_t = f \odot c_{t-1} + i \odot g \quad \text{Denklem} \quad (3.3)$$

$$h_t = o \odot \tanh(c_t) \quad \text{Denklem} \quad (3.4)$$

Burada sigmoid fonksiyonu sigm ve tanh eleman olarak uygulanır ve eğer giriş boyutsallığı D ise ve gizli durum H birimleri varsa W matrisi $[4H \times (D + H)]$ boyutlarına sahiptir. Üç vektör $i, f, o \in \mathbb{R}^H$ vektörü, her bellek hücresinin güncellenip güncellenmediğini, sıfırlanıp sıfırlanmadığını ve yerel durumunun gizli vektörde sırasıyla gösterilip gösterilmediğini kontrol eden ikili kapılar olarak düşünülür. Bu kapıların aktivasyonları sigmoid fonksiyonuna dayanır ve bu nedenle modeli farklı tutmak için sıfır ile bir arasında düzgün bir şekilde değişmesine izin verilir. $G \in \mathbb{R}^H$ vektörü -1 ile 1 arasında değişir ve bellek içeriğini ilave olarak değiştirmek için kullanılır. Bu katkı etkileşimi, LSTM' nin tasarımının kritik bir özelliğidir, çünkü geri yayılım sırasında bir toplam işlemi yalnızca indirgemeleri eşit olarak dağıtır. Bu, bellek hücreleri c üzerindeki gradyanların uzun zaman aralıklarında veya en azından aktif bir unutma geçidinin çarpımsal etkileşimi ile akış kesilene kadar geriye doğru akmasına izin verir.

Karakter seviyesi dil modelleri, sıralı öğrenme için yaygın olarak incelenen yorumlanabilir bir test yatağıdır. Bu ayarda, RNN' ye giriş bazı metinden bir karakter dizisidir (örn. “X ışını koyu bölüt”) ve ağ dizideki bir sonraki karakteri tahmin etmek için eğitilir. Somut olarak, K karakterlerinden oluşan sabit bir kelime dağarcığı varsayarak, tüm karakterleri K boyutlu tek sıcak vektörlerle $\{x_t\}$, $t = 1, \dots, T$ kodlar ve H -boyutlu gizli bir dizi elde etmek için bunları ağa beslenir. Dizideki bir sonraki karakter için tahminler elde etmek için gizli durumları ayrıca bir vektör dizisine $\{y_t\}$ yansıtırız; burada $y_t = W_y h_t$ ve W_y , bir $[K \times H]$ parametre matrisidir. Bu vektörler bir sonraki karakterin loglarıdır ve amaç tüm hedefler üzerindeki ortalama çapraz entropi kaybını en aza indirmektir.

Özet olarak, bir bağlamda bir sinir ağı uygulamak için tipik bir iş akışı aşağıdaki gibidir:

İlk önce bir veri kümesi edinilir. Veri kümesinin, x' in bazı girdi örneği ve y' nin bir etiket (x, y) çiftinden oluştuğu varsayılır. Daha sonra veri kümesi genellikle eğitim, doğrulama ve test katı olmak üzere üç kata bölünür (ortak oranlar sırasıyla %80, %10, %10 olabilir). Eğitim katmanı geri yayılım ile parametreleri optimize etmek için,

doğrulama katmanı hiper parametre iyileştirmesi için ve test katmanı da değerlendirme için kullanılır.

Verilerin ön işlenmesi sinir ağlarının yakınsamasının iyileştirilmesine yardımcı olabilir (Lecun 1990). Görüntüler için yaygın ön işleme teknikleri, verilerin standartlaştırılmasını (ortalamanın çıkarılması ve x'in her giriş boyutu için standart sapmaya ayrı ayrı bölünmesi) veya en azından ortalamanın çıkarılmasını içerir. Bu istatistiklerin yalnızca eğitim verileri üzerinde tahmin edilmesi ve bu sabit istatistikleri, doğrulama ve test verilerini işlemek için kullanmak önemlidir. Çünkü bu, nihai sistemin gerçek dünyadaki uygulanmasını doğru şekilde simüle eder. Sonraki adımda keşfetmek istediğimiz bir mimarlık ailesine karar verilmelidir. Bu f fonksiyonunu oluşturan hesaplama grafiğinin iç kısımlarının tasarlanması anlamına gelir. Bu aşama bir bilimden çok bir sanattır, ancak pratikte kullanılan birkaç ortak buluşsal yöntem değerlendirilmektedir. Piksel verilerini kıvrımlarla işlemek ve sıralı verileri tekrarlayan ağlarla işlemek yaygındır. Mimarının ölçeği için, çok kaba bir kural, tam modelin eğitim veri kümesinde örneklerle yaklaşık olarak benzer sayıda parametreye sahip olması gerektiğidir. Örneğin, ImageNet sınavı (Socher R. vd. 2011) veri kümesinin 1 milyon örnekleri veya veri çoğaltmak planlanıyorsa küçük bir faktör daha vardır. ImageNet üzerinde eğitilen ConvNets genellikle 10M parametrelerine sahiptir ve aşırı öğrenmeyi önlemek için modeli daha da kısıtlamak adına düzenleme teknikleri (L2 düzenlenmesi, bırakma ve veri artırımı gibi) kullanılır (Van Laarhoven 2017).

Optimizasyon: Varsayılan bir öneri, iyileştirme için 'Adam' (Krishna R. vd. 2016) 'ı kullanmaktır, öğrenme oranları yaklaşık $1e^{-3}$, ilk anın katsayısı 0.9 ve ikinci an 0.99'dur. Eğitim sırasında öğrenme oranını, eğitimin sonuna kadar yaklaşık 100 faktör kadar optimize etmek genellikle faydalıdır, ancak eğitimin sadece yarısı bittikten sonra ilk kez bozulma yaygındır (Russakovsky vd. 2015). Kodda hata ayıklamaya yardımcı olmak için tam eğitim setini optimize etmeden önce tek bir veri kümesini (sıfır eğitim kaybına ulaşan) geçmeye çalışmak iyi bir fikirdir. Optimizasyon sırasında doğrulama performansı ve model kontrol noktasını dosyaya kaydederken $\bar{\theta}$ kullanılır. Her parametre güncellemesinden sonra bir ortalama parametre vektörünü $\bar{\theta}$ (örn. $\bar{\theta} = 0.999\bar{\theta}$

+ 0.0010) takip ettiğimiz ve hesapladığımız Polyak ortalamasını (Rumelhart vd. 1986) kullanmak neredeyse her zaman iyi bir fikirdir.

Hiperparametre iyileştirme: Stokastik gradyan ile iyileştirme, sürecin bir iç döngüsü olarak görülebilirken, hiper parametre iyileştirme, zorlanabilen veya geri çoğaltılması imkânsız olan hiper parametrelerin (öğrenme hızı veya ünite sayısı gibi) iyi değerlerini belirleyen dış döngüdür (gizli katmanlarda). Bu süreç, bazı arama aralığından hiper parametrelerin örneklenmesinden, modeli iyileştirmekten ve modeli doğrulama katında değerlendirmekten oluşur. En iyi son model, en iyi doğrulama performansını elde eden modeldir (Kingma ve Ba 2014).

En iyi eğitilmiş modeli (en düşük doğrulama kaybıyla) belirledikten sonra, model test setinde tek bir kez değerlendirilir ve performansı rapor edilir. Farklı başlatmalardan veya farklı hiper parametrelerle eğitilmiş birden fazla modeli değerlendirmenin sonuçlarını ortalayan model toplulukları kullanılarak her zaman tutarlı iyileştirmeler elde edilebilir.

Bu tezin geri kalanı, genetik algoritma destekli görüntü sınıflandırması ele almaktadır. Ayrıca öğrenme transferi ve oto kodlayıcı destekli görüntü işleme ve sınıflandırmasını analiz edilmektedir. Bununla birlikte metin veri seti işlenerek kalp krizi riski tahminleyicisi çalışması incelenmiştir. Tez boyunca yapılan her bir çalışmanın ayrıntıları sırayla açıklanacaktır.

4. ÇALIŞMA VE BULGULAR

Tez boyunca yaptığımız çalışmaları üç başlık altında toplayabiliriz; uluslararası kitap bölümü olarak yayınlanan birinci çalışmamız beyaz kan hücreleri görüntü veri setinin genetik algoritma destekli evrışimli ağlar ile sınıflandırılmasıdır. İkinci çalışmamız makale yayını olan, Covid-19 teşhisi ve beyaz kan hücreleri görüntü verilerinin kullanıldığı ön eğitilmiş ResNET-50 evrışimli ağı, oto kodlayıcı desteği ve SVM ile sınıflandırılmasıdır. Üçüncü çalışmamız ise kalp krizi metin veri setinin evrışimli ağlar ile risk gruplarına sınıflandırılmasıdır. Yine doktora sürecinde uluslararası konferans çalışması olarak yaptığımız tüberküloz teşhisi de ele alınmıştır. Birinci çalışmamızda genetik algoritma kullanılarak CNN modelinin hiper parametreleri kendiliğinden belirlenmiş ve en yüksek doğruluğu veren model elde edilmiştir. Elde edilen sonuç derin öğrenme yöntemlerinin metin ve görsel veri setleri üzerinde özellik çıkarıcı, sınıflandırıcı ve tahminleyici olarak çalışmıştır. Bu veri setlerini kullanarak tahmin edici ve sınıflandırıcı yapay zekâ modelleri %90 üzeri doğruluk oranında sonuç vermiştir. Metin ve görüntü işlemede derin öğrenme modelleri ve özellikle de CNN kullanılarak hedeflenen sonuçlar için çözümler sunulmuştur. Bu çalışmalarda kullanılan dizüstü bilgisayar Predator Triton 500 olup, i7 işlemci, 32 GB RAM, 8 GB ayrı GPU olan NVIDIA GeForce RTX 2080 ekran kartı, 512 GB SSD donanımına sahiptir.

4.1 CNN ve Genetik Algoritma ile Beyaz Kan Hücrelerinin Sınıflandırılması

Evrışimli Sinir Ağları (CNN'ler), artan gerçek dünya uygulamaları nedeniyle son yıllarda önemli bir cazibe kazanmıştır. Performansları büyük ölçüde ağ yapısına ve ağ parametrelerini ayarlamak için seçilen iyileştirme yöntemine bağlıdır. Bu çalışmada, evrışimli sinir ağlarını eğitmek için yeni ve etkili yöntemler önerilmektedir. Derin öğrenme yöntemlerinin çoğu gradyan inişine dayanmaktadır. Geleneksel CNN eğitim yöntemlerinin aksine, genetik algoritmalar (GA) dayanan yöntemleri kullanarak CNN optimize etmeyi önerildi. Ayrıca, ağların eğitiminde kullanılan genetik algoritma performansı, yakınsama oranı ve genel doğruluğu değerlendirilmiştir.

Kan örneklerinin mikroskopik incelemesi, hematik patolojiler hakkında kalitatif ve kantitatif bilgi sağlar. Beyaz kan hücreleri (WBC) yani lökositlerin sayısının hesaplanması hastalıkların teşhisinde çok önemli bir yere sahiptir. Öyle ki başta enfeksiyon olmak üzere alerjiden lösemiye, AIDS ve kansere kadar birçok hastalığın teşhisinde kullanılmaktadır. Kan akışındaki beyaz kan hücrelerinin yoğunluğu bağışıklık sistemimizin durumu ve karşılaşılabileceğimiz olası risklerle ilgili ipuçları verir. Beyaz kan hücre sayısındaki büyük bir değişim, genellikle vücudumuzun bir antijen tarafından etkilendiğinin bir işaretidir. Dahası, belirli bir lökosit tipindeki bir değişim, genel olarak belirli bir antijen tipi ile ilişkilidir. Beyaz kan hücreleri dört ana sınıfa ayrılmaktadır ve bunlar Lenfosit, Monosit, Nötrofil, Eozinofil olarak adlandırılmaktadır (Kansara vd. 2018). Beyaz kan hücrelerinin sayımı günümüzde akım taraması araçları ve lazer akım sitometreleri ile yapılmaktadır (Kickler 1999). Bu cihazlar yüksek maliyetli olup bir kan örneğinin sayımını birkaç saat içerisinde tamamlamaktadır. Bu bağlamda bizim çalışmamız bu soruna çözüm üretmektedir. Gelişen teknoloji ile birlikte yapay zekânın öne çıkması her alanda olduğu gibi sağlık alanında da önemli gelişmelerin yolunu açmıştır. Günümüzde işlemci ve grafik kartların gelişimi ve kayıtlı verilerin sayısının artması ile, yapay zekâ modelleri başarılı sınıflandırma ve analiz yetenekleri ile öne çıkmış ayrıca katmanlı yapay zekâ modeli ile derin öğrenmenin de yıldızı parlamıştır (Ohlsson 2011). Derin öğrenmeyle birlikte makina öğrenmesindeki çalışmalar öznitelik mühendisliğinden mimari mühendisliğine dönüşmüştür. Önceki çalışmalarda problemi çözebilmek için problemi ifade edecek en iyi öznitelik kümelerinin oluşturulması ve bu öznitelikler arasında temsil kabiliyeti en yüksek olanların seçilmesi üzerinde çalışılıyordu (Hall 1999). Ancak derin öğrenme yaklaşımlarının gelişimi ile çok katmanlı yapay sinir ağının nasıl tasarlandığı; kaç katmandan oluştuğu, ne kadar nöron içereceği, hangi iyileştirme algoritmasının ya da aktivasyon fonksiyonunun kullanıldığı problem çözümünde daha önemli hale geldi. Böylece derin öğrenme ile problem çözmek çok katmanlı ağ yapısını en iyi şekilde tasarlamak ile eşdeğer duruma geldi (Bergstra ve Bengio 2012).

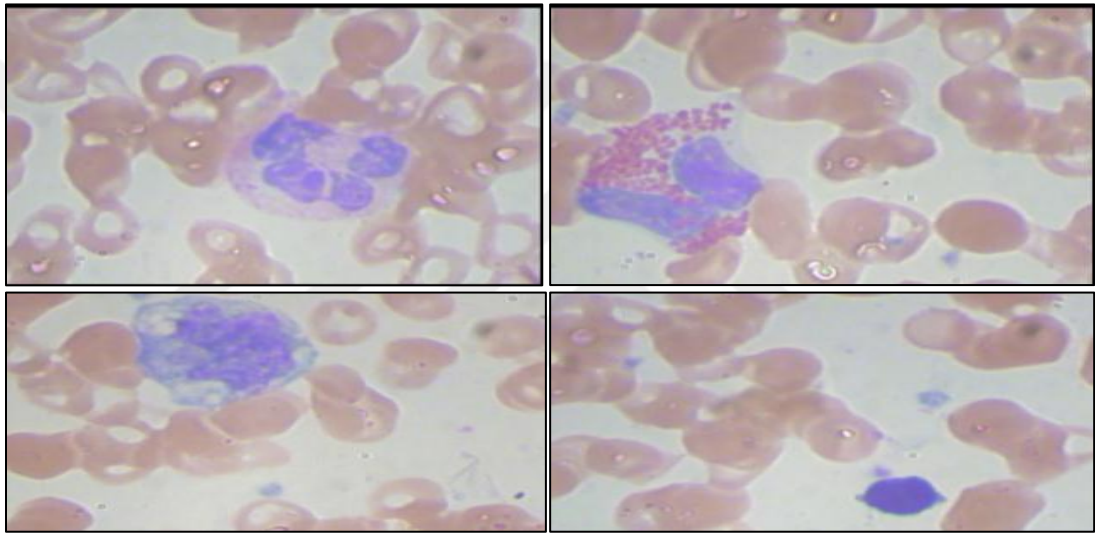
Bahsedilen parametreler derin öğrenmede hiper parametreler olarak adlandırılmakta ve modelin belirleyici unsurlarını oluşturmaktadır. Öyle ki modelin parametreleri en iyi şekilde tasarlandığında verdiği sonucun doğruluğu ve performansı da o denli artmaktadır.

Özellikle resim sınıflandırma alanında derin öğrenme modelleri 2010 yılından itibaren hızla gelişmiştir. Derin evrişimli ağ modelleri resimlerden elde ettiği kısımları katmanlar içerisinde öğrenerek detaylardan bütüne doğru çıkarımlar yapmakta ve böylece resmi iyi derecede analiz edebilmektedir. Derin evrişimli ağlarda derinlik yani katman sayısı çok önemli bir parametredir. Katman sayısının artması teorik olarak ifade etme kapasitesini artırır ve bunun da daha başarılı ağ mimarileri oluşturması beklenir. Ancak pratikte derin öğrenmede bu durum geçerli değildir çünkü geriye doğru besleme yapısı kullanıldığı ve bu işlemde ağırlık değerleri kısmi türevleri alınarak çarpıldığı için ilk katmanlara doğru çarpan etkisini yitirmekte ve sıfırlamaktadır. Bu durum bazı nöronların ölmesine yani hesaplamada etkisiz hale gelmesine neden olmaktadır. Buna kaybolan eğim (Vanishing Gradient) sorunu denmektedir. Bunun tersi de mümkünken çarpanların çok yüksek değerlere çıkması da (Exploding Gradient) mümkündür (Szegedy vd. 2017). Yine katman sayısı arttığı için parametre sayısı artmakta ve dolayısı ile hem hesaplama yani öğrenme hızı yavaşlamakta hem de aşırı öğrenme sorunu ortaya çıkabilmektedir. Tabiki bunların çözümü için hiper parametrelerin düzenlenmesi ile iyileştirmeler yapılmaktadır. Örneğin her bir girdiyi tek tek işleme tabi tutmak yerine belirli sayılarda gruplar halinde girdiler (Batch Normalization) olarak alınmakta bu da aşırı öğrenmenin önüne geçmektedir (Simon vd. 2016). Yine ağı eğitime çok katkı sağlayamayan bazı nöronlar hesap dışı bırakılmakta ve hem performans hem tutarlılık arttırılmaktadır (He vd. 2015, Gal ve Ghahramani 2016).

Bu çalışmada derin evrişimli sinir ağı modelini yarışmaya açık olan beyaz kan hücreleri veri seti üzerinde geliştirdik. Modelimizin hiper parametrelerini iyileştirerek en iyi başarıyı elde etmeye çalışıldı. Modelin çalışmasını etkileyen parametreleri değerlendirildi ve elde edilen farklı sonuçlar ortaya konuldu. Diğer derin ağ modelleri ile kıyaslayarak performans ölçütlerini analiz edildi (Hodosh vd. 2013). Beyaz kan hücre veri seti üzerinde yapılan diğer çalışmalarda (Loudon 2018, Liang vd. 2018) elde edilen başarımlarla kendi elde ettiğimiz performans değerlerini karşılaştırıldı (Mooney 2018, Kansara vd..2018, Sharon ve Therse 2018). Sonuç olarak geliştirdiğimiz model, bu veri seti ile yapılmış çalışmalarda modellerden daha yüksek doğruluk sonuçları elde edildi. Böylece beyaz kan hücre sınıflandırmasını bir yazılım üzerinden çok düşük maliyetlerle ve neredeyse anında sonuç verecek şekilde yapmak mümkün olmuştur. Derin öğrenme modelleri ve donanım alt yapı ve mimarileri geliştikçe veri setleri

artırılıp model daha da iyi öğrendikçe sürekli kendini yenileyecek bir teknolojiye sahip olunacaktır.

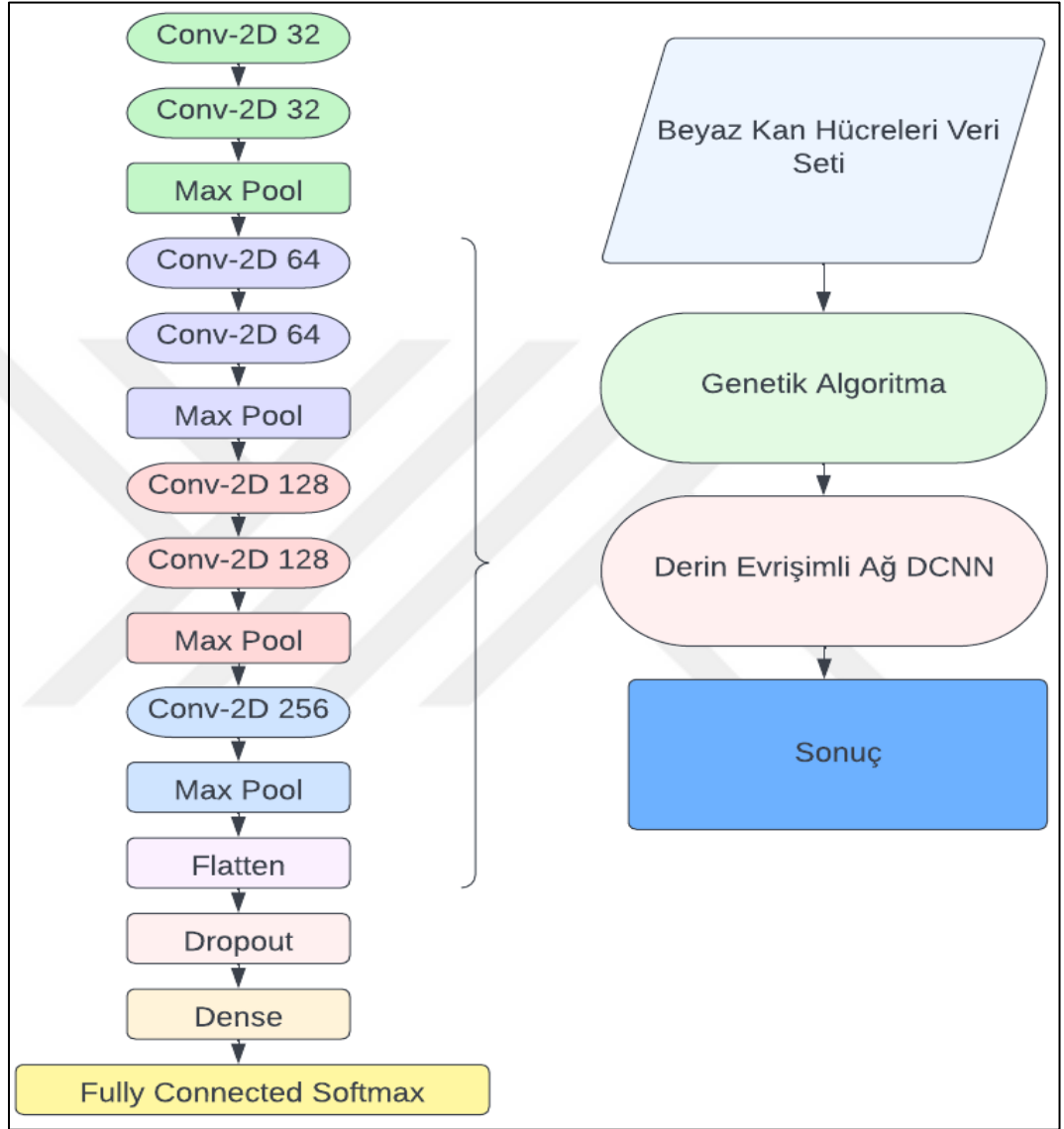
Beyaz kan hücreleri vücudumuzun bağışıklık sistemini belirleyen en önemli göstergelerden birisidir. Lökositin kanımızdaki toplam sayısı ve alt gruplarının yani Şekil 4.1.a, Şekil 4.1.b Şekil 4.1.c, Şekil 4.1.d’ de sırasıyla görülen nötrofil, eozinofil, monosit ve lenfosit sayıları bağışıklık sistemimize ve ne tür bir hastalıktan etkilendiğimize dair önemli göstergeleri taşımaktadır (Zhao vd. 2017).



Şekil 4.1 a) Nötrofil b) Eozinofil c) Monosit d) Lenfosit

Derin öğrenmeyi kullanarak bir problemi çözerken önemli bir zorluk, algoritma için hiper parametreleri belirlemek ve ayarlamaktır. Çoğunlukla, bu hiper parametreler deneyime dayalı olarak manuel olarak ayarlanır. Bu çalışmada, dört beyaz kan hücresi tipinin sınıflandırılması için yeni bir derin evrişimli sinir ağı modeli önerilmiştir. Bu modelde, hiper parametreler, modelde önemli iyileştirme sağlayan bir genetik algoritma ile kendi kendine optimize edilir. Önerilen modelin doğrulanması için beyaz kan hücreleri veri setinden elde edilen dört tip beyaz kan hücresi incelenmiştir. Model toplam 7 katmanlı olup; 32 nöronluk 2 katman, 64 nöronluk 2 katman, 128 nöronluk 2 katman ve 256 nöronluk 1 katman ile bir yoğunlaştırma katmanından oluşmaktadır. Aşırı öğrenmeyi engellemek için düşürme (dropout) katmanı kullanıldı. Beyaz kan hücresi görüntülerinin sayısı yaklaşık 12.000 adet olup 4 farklı sınıf için ortalama 3000 adet görüntü verisi bulunmaktadır.

Veriseti eğitim ve test olmak üzere sırasıyla %80'e %20 oranında bölünmüştür. Model beyaz kan hücresi veri kümesine uygulandığında, verisetindeki dört grup beyaz kan hücresi sınıflandırıldı.



Şekil 4.2 Önerilen model şeması

Genetik algoritma bloğunda 28 bitlik bir kromozom oluşturulmaktadır. Bu kromozom girdi katmanından sonraki iki katmanda bulunan gizli nöron sayılarını, katmanlarda kullanılan filtrenin genişlik ve yüksekliğini (3x3, 5x5 vb.), düşürme değerini ve girdilerde kullanılan veri grup sayısını (batch size) temsil eder. Bu parametrelerin en üst ve en alt değerleri belirlendi, örneğin ilk katman için 1-64 aralığı belirlendi. Benzer şekilde ikinci katman için 1-128, veri küme sayısı için 1-60, düşürme için 0.1 – 0.5 değer aralıkları verildi. Bu değerleri temsil eden rastgele 200 adet kromozom kümesi oluşturuldu ve bu küme ilk nesili oluşturdu. Bundan sonraki nesilleri oluşturmak için popülasyon sayısının yarısı kadar döngü çalıştırılarak mevcut popülasyon içerisinde her seferinde 3 farklı üye kromozom seçilerek bunlardan en iyi olanı bir sonraki neslin yeni üyesi olur ve bu işlem iki kez tekrar edilerek her seferinde 2 yeni kromozom elde edilir. Bu işlem sürecinde çaprazlama ve mutasyon uygulanır. Kromoz seçimi yapılırken uygunluk fonksiyonu olan doğruluk değeri sonucuna bakılarak en yüksek doğruluğu veren seçilir. Böylece tüm nesil tamamlandığında en yüksek doğruluğa sahip kromozom modeli oluşturmak için kullanılır. Burada en iyi temsil eden kromozom, 64, 128 gizli katman nöronları, 32 veri küme sayısı ve 0,25 düşürme değeri vermiştir. Bu hiper parametrelerle çalıştırılan D-CNN modeli %93'ün üzerinde sınıflandırma doğruluğu başarıyla göstermiştir.

4.1.1 Lökositlerin tiplerine göre işaret ettikleri hastalıklar

Lökosit tipleri sayılarının yüksek ya da düşük olmasına göre farklı hastalıklara işaret etmektedirler. Örneğin nötrofil sayısı akut bakteriyel iltihaplara, lenfosit sayısı kronik iltihap ve lösemilere, monosit sayısı akut, kronik iltihap, lösemi ve kollajen doku ve granümatöz hastalıklara, eozinofili ise alerji, paraziter, hematolojik hastalıklar ve intestinal hastalıklara işaret etmektedir (Zhao vd. 2017).

Örneğin, lösemi hastalarında sıklıkla, bağışıklık sisteminin zayıflığına bağlı olarak kan akışlarında belirgin şekilde daha yüksek bir lenfosit seviyesi görülür. Aynı şekilde, alerjisi olan kişilerde genellikle eozinofil sayısında bir artış olduğu görülür, çünkü bu, lökositlerin alerjenlerle savaşmada anahtarıdır.

4.1.2 CNN optimizasyonunun kullanılarak kromozom temsili

Hiper parametre değerlerini genetik algoritma aracılığıyla işlemek için, hiper parametreleri temsil eden kromozom Çizelge 4.1 ve Çizelge 4.2 'de gösterilmektedir. Özellik haritası, veri küme boyutu (batch), özellik haritası uzunluğu ve genişliği tam sayılar olarak kodlanırken, bırakma yüzdesi kayan noktalar olarak kodlanmıştır.

Çizelge 4.1 Parametreler için kromozom genotipi

integer	integer	İnteger	İnteger	float
---------	---------	---------	---------	-------

Çizelge 4.2 Parametreler için kromozom fenotipi

Özellik Haritası Sayısı (Feature Map)	Özellik Haritası Boy	Özellik Haritası En	Öbekleme sayısı (Batch)	Düşürme oranı (Dropout)
---------------------------------------	----------------------	---------------------	-------------------------	-------------------------

4.1.3 Veri ön işleme

Bu çalışmada, CNN'nin eğitim sürecini hızlandırmak için görüntüler yeniden boyutlandırılarak 60x80 yapılmıştır. Görüntüler, yeniden boyutlandırıldıktan sonra 255.0 ile bölünerek her bir piksel 0-1 aralığındaki değerlere normalize edilmiştir. Keras kütüphanesinin resim ön işleme fonksiyonları kullanılarak resimlerin boyutlanması, yatay düzlemde döndürülmesi, yakınlaştırılması, eğilmesi gibi tekniklerle veri seti düzenlenmekte ve girdi sayısı çoğaltılmıştır. Bu işlemler eğitim veri seti için uygulanmakta olup doğrulama veri seti için uygulanmaz.

4.1.4 Genetik algoritma

Genetik algoritma, yukarıda açıklandığı gibi CNN'nin hiper parametrelerini optimize etmek için kullanıldı. GA hiper parametreleri için seçilen çaprazlamanın öneminden dolayı 0.8 olasılığa ayarlanırken, mutasyon ise 0.1 olasılığa ayarlanmıştır. Turnuva seçimi ve elitizm kullanılmış olup turnuva seçimi için her seferinde 3 kromozom seçilmiştir (k=3). Model, topluluk büyüklüğü 100 olan 200 nesil için çalıştırıldı.

4.1.5 GA-geliştirilmiş D-CNN modeli ve sonuçları

Bu çalışma, sonuçları optimize etmek için hiper parametreleri genetik bir algoritma ile kendi kendini ayarlayan bir derin evrişimli sinir ağı (D-CNN) modeli geliştirmek için tasarlanmıştır. Genel amaç, tanımlanmış genetik algoritma ile optimize edilmiş D-CNN modeli kullanılarak beyaz kan hücrelerinin sınıflandırma doğruluğunu artırmak ve birçok hastalığın doğru teşhisi ile sağlık alanına önemli bir katkıda bulunmaktır. Model, Şekil-4.2' de gösterildiği gibi parametreleri bir eklenti mimarisinde uyarlanabilir ve oluşturulabilir şekilde tasarlanmıştır.

Geliştirilen modeli uygulamak için Kaggle'dan elde edilen beyaz kan hücresi görüntüleri 60x80, 120x160 ve 224x224 boyutlarında ayrı ayrı çalıştırılarak denenmiştir. Model tasarımı yapılırken CNN'nin derinliği sıkı bir model yapısından başlayarak artırılmıştır. Çizelge 4.3'de görüldüğü gibi en iyi sonuçlara ulaşıldığında modele son şekli verilmiştir. Modelin mikroskobik görüntüler kullanılırken 3x3, 5x5, 7x7 çekirdekli filtrelerden ve filtre sayısının 32-256 aralığında seçim yapması sağlanmıştır.

Evrişimli sinir ağlarında evrişim katmanı görüntüden çıkarılacak alanları tanımlar ve bu işlemden sonra görüntünün boyutları küçültülür. Resmin boyutu küçültülürken, havuza iki kez atlanarak her seferinde 2x2 kare maksimize edilerek resimdeki çizgiler vurgulanır. Bu, görüntü boyutunun azalmasına neden olur. Katmanlar derinleştikçe farklı nitelikler ortaya çıkar ve özellikler ayrıntılardan türetilir. Görüntüler ayrı girdiler veya toplu girdiler olarak ele alınabilir. Ek olarak, grupların aşırı öğrenmesi genellikle engellenir ve daha kısa bir işlem süresi elde edilir. Önerilen model, üç adet ikili konvülyasyon katmanı kullanılarak oluşturulan ve daha sonra düzleştirme, bırakma, yoğunlaştırma ve tam bağlantılı katmanlarla tamamlanan iki katmandan oluşmaktadır. Girdi olarak alınan beyaz kan hücrelerinin görüntülerinden saptanması beklenen hücre çekirdekleri görüntülerde nispeten küçük olduğu için 3x3 filtreler daha başarılı çalışmıştır. Evrişim işleminde 32 kanaldan başlanarak kanal sayısı 64'e, ardından 128'e ve daha sonra detay özelliklerini çıkarmak için 256'ya çıkarılmıştır. L2 düzenleme tekniği, öğrenme oranını iyileştirmek için test edildi, ancak başarılı grup normalizasyonunun olmaması nedeniyle sonunda kullanılmadı

(Yığın Normalleştirme) (Zhang vd. 2017). Ek olarak, öğrenme üzerinde çok az etkisi olan nöronlara seyreltme (bırakma) uygulandı.

Çizelge 4.3 GA ile geliştirilmiş D-CNN modelinin katman sayısının belirlenmesi

Katman Sayısı	Parametre Sayısı	Doğruluk (%)
3	88548	79.85
5	262308	84.47
7	584228	93.3
9	2385188	77.55

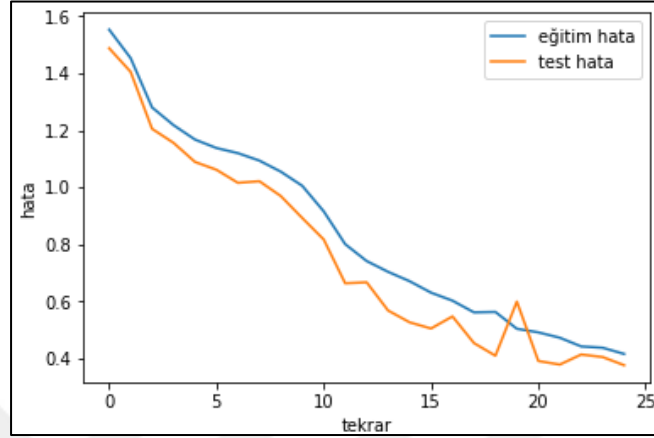
Derin CNN ile çok iyi çalışan ReLU aktivasyon fonksiyonunun başarılı olduğu kanıtlanmış dolayısı ile modelimizde ReLU tercih edilmiştir (Zhang vd. 2015). ‘Adam’ öğrenme hızı, öğrenme hızını kendine göre ayarlamak için en etkili fonksiyonlardan biri olarak seçilmiş modelimizde de kullanılmıştır (Kingma ve Ba 2014). Sınıflandırma kararı için başarılı yakınsama ile sonuçlanan fonksiyonlardan biri olan Softmax kullanıldı. Görüntüler dört beyaz kan hücresi kategorisine sınıflandırılırken, hatayı hesaplamak için kategorik çapraz bağlama işlevi kullanıldı.

Model anlatıldığı gibi tasarlandıktan sonra eğitilmiş ve test edilmiştir. Sonuçları iyileştirmek için hiper parametrelerde amaçlarına göre değişiklikler yapıldı. Süreci kolaylaştırmak için görüntü boyutu 60x80 kullanıldı. Girdilerin gruplar halinde verildiği modelde genetik algoritma grup büyüklüğünü 32 olarak seçmiştir. Yine tüm veriler modelden geçirilmiş ve 50 tur boyunca çalıştırılmıştır ve erken sonlandırma ile değerlendirilerek durdurulmuştur.

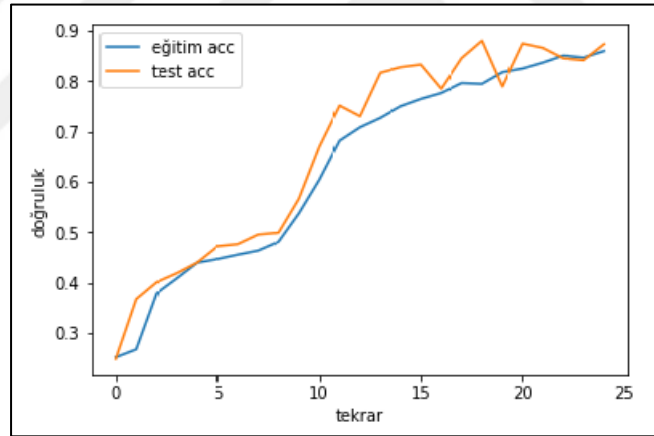
Oluşturulan model, grafik işlemci desteğinin kullanıldığı Kaggle sunucusunda çalıştırıldı. Önerilen model, burada daha önce incelenen mevcut modellerle karşılaştırıldı. Şekil-4.3, 25 tur boyunca eğitim ve test hatası grafiklerini göstermektedir. Şekil-4.4. elde edilen doğruluğu sağlar. Doğruluk değerleri aşağıdaki 4.1 ve 4.2 denklemleri kullanılarak hesaplanmıştır; burada TP doğru pozitifler, TN doğru negatifler, FP yanlış pozitifler ve FN yanlış negatiflerdir.

$$Doğruluk = (TP + TN / (TP + FN + TN + FN)) \quad (4.1)$$

$$Kesinlik = (TP / (TP + FP)) \quad (4.2)$$

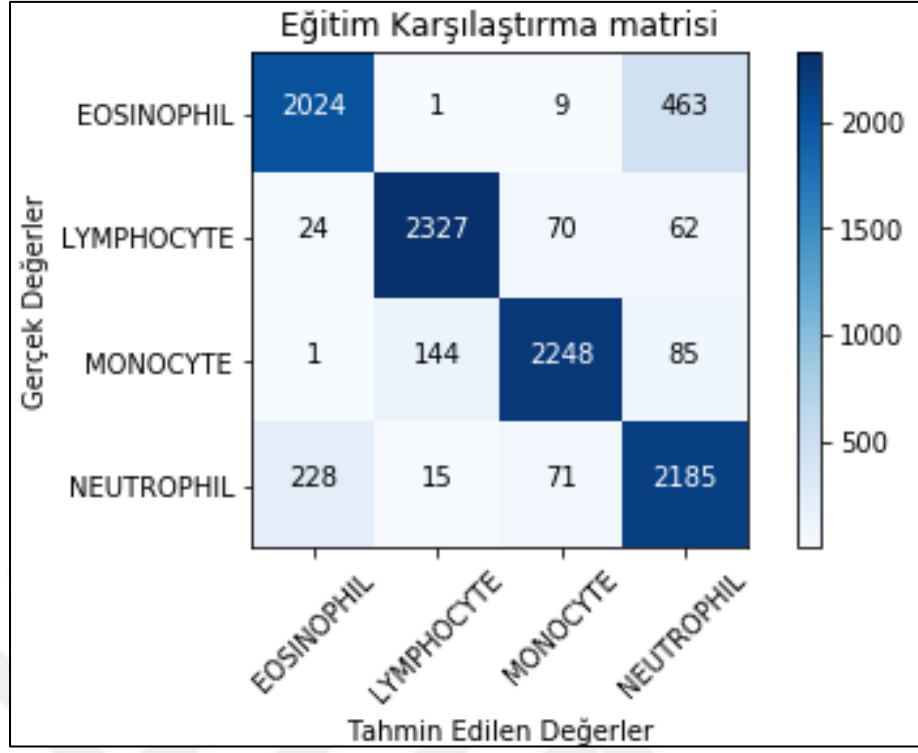


Şekil 4.3 Test ve eğitim verileri hata grafiği

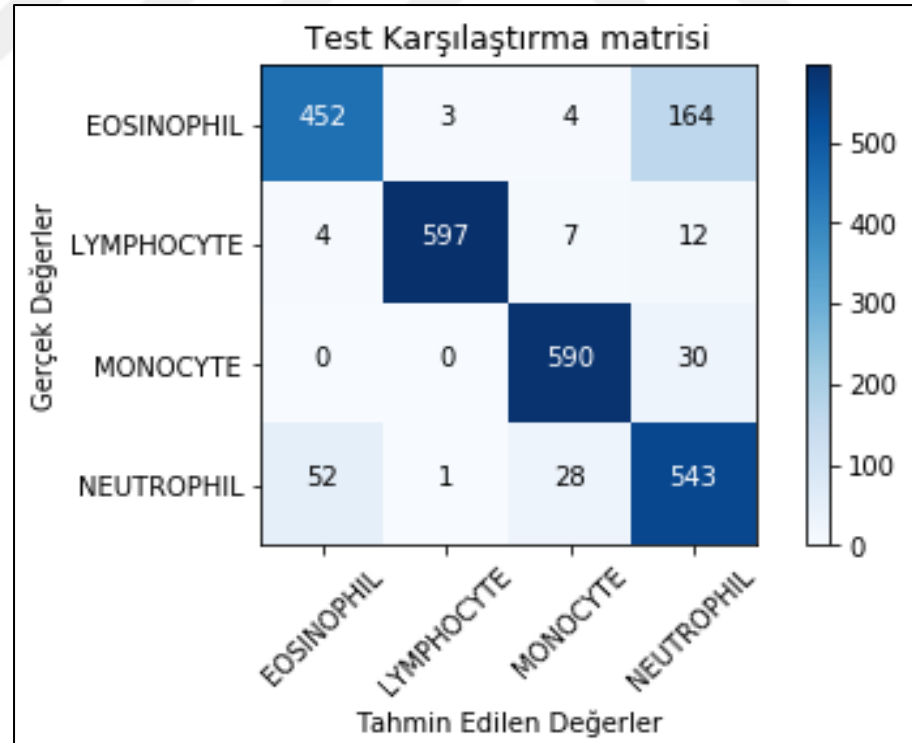


Şekil 4.4 Test ve eğitim verileri doğruluk grafiği

Yine karşılaştırma matrisinde gerçek değerler ve tahmin edilen değerler verilerek yapılan doğru ve hatalı tahminler gösterilmiştir. Eğitim seti için Şekil-4.5 ve test seti için Şekil-4.6' da karşılaştırma matrisleri verilmiştir.



Şekil 4.5 Eğitim verileri karşılaştırma matrisi



Şekil 4.6 Test verileri karşılaştırma matrisi

Çizelge 4.4 GA destekli derin evrimsel ağ modelimizin diğer çalışmalarla kıyaslanması

Diğer Çalışmalarla Kıyaslama		
Çalışma	Model	Doğruluk
Mooney (Mooney, 2018)	DCNN	%78.33
YOLO (Persson, 2018)	Tiny YOLO -CNN	%87.10
Kansara (Kansara vd., 2018)	DCNN	%83.84
Önerdiğimiz Model	DCNN-GA	%93.30

Bu çalışmada, tıbbi araştırma kurallarını yeniden yazabilecek belki de günümüzün en önde gelen teknolojik çözümleri olan sağlık alanında, yapay zekâ ve derin öğrenme yöntemlerine dayanan GA ile güçlendirilmiş bir D-CNN modeli test edilmiştir. Beyaz kan hücrelerinin sayısı, bireylerin sağlığı hakkında çok önemli ipuçları sağlar ve önemli hastalıkların teşhisinde kullanılır. Kan hücrelerini saymak için kullanılan geleneksel cihazlar, numunelerin analizi zaman aldığından hem maliyetli hem de zaman alıcıdır. Bu çalışma, dört tip beyaz kan hücresini sınıflandırmak için kullanılabilecek alternatif bir derin öğrenme modeli sunmaktadır. Bu mühendislik çalışması, derin öğrenmede bir hiper parametrik sinir ağı mimarisinin tasarımı ile sonuçlanmış, katman sayısı, iyileştirme, aktivasyon fonksiyonları gibi hiper parametrelerin nasıl ayarlanacağını belirlemiştir. Geliştirilen model ise daha derin bir yapıya doğru sığ bir yapı uygulanarak en iyi performans için tasarlanmış ve farklı parametre değerleri ile test edilmiştir. Geliştirilen model, bilinen diğer mimarilerle ve beyaz kan hücrelerini araştıran ve analiz eden diğer çalışmalarla karşılaştırıldı. Burada sunulan model, beyaz kan hücrelerinin sınıflandırılmasında %93,3 doğrulukla göstermekte ve genetik algoritma (GA) ile kendi parametrelerini belirlemektedir. Önerilen model, girdi olarak mikroskopik görüntülerle anında sonuçlar sağlayabilir. Böylece geliştirilen model kullanılarak beyaz kan hücrelerinin daha doğru ve daha hızlı analizlerinin yapılması mümkün olacaktır. Genetik algoritma ile geliştirilmiş derin evrimsel sinir ağı modeli, derin evrimsel öğrenmeyi kullanarak dört beyaz kan hücresi kategorisini başarıyla sınıflandırmaktadır.

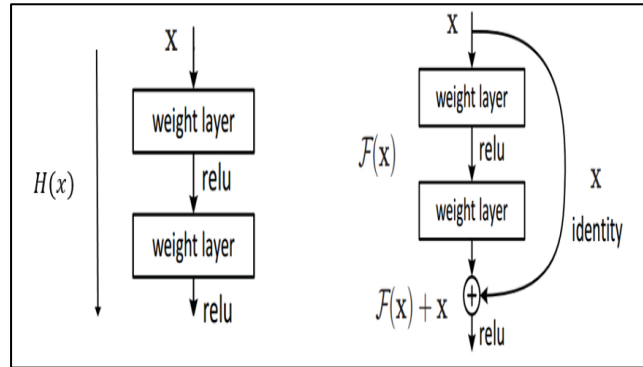
4.2 Beyaz Kan Hücrelerinin Sınıflandırılması ve Covid-19 Teşhisinin Öğrenme Transferi ile Uygulanması

Kan dolaşımındaki beyaz kan hücrelerinin sayısı, olası sağlık tehditleri ile birlikte kişinin bağışıklık sistemi ve olası hastalıklar hakkında ipuçları vermektedir (Zhao vd. 2017). Lökosit adı verilen beyaz kan hücreleri, lösemi, AIDS ve kanser gibi birçok sağlık sorununun tanımlanmasında kritik bir rol oynamaktadır (Joshi vd. 2013, Kuan vd. 2017). Diğer yandan COVID-19 virüsü çok bulaşıcı olup, tüm dünyada hızlı yayılmakta ve yüksek oranda ölümcül vakalara sebep olmaktadır. COVID-19'un ön teşhisi, sağlık uzmanlarının salgın eğrisini yavaşlatması için çok önemlidir (Velavan ve Meyer 2020). Bu çalışmada, COVID-19 hastalığının hızlı ve doğru bir şekilde otomatik olarak tespit edilmesi ve beyaz kan hücrelerinin sınıflandırılması için göğüs röntgeni görüntülerinin veri seti koleksiyonu ve beyaz kan hücreleri görüntü veri setinden kullanılmıştır. Çalışma süresini azaltmak ve doğruluğu artırmak için ön eğitilmiş modeller tercih edilmiştir. Veri setini eğitirken buradaki ağırlıkları modele yüklenerek çalıştırılabilir. Ön eğitilmiş modeller ile genel resim sınıflandırma problemlerinde çok başarılı sonuçlar elde edilmiştir. Ön eğitilmiş modellerin daha önce milyonlarca resim veri setleri üzerinde çalıştırılarak en iyi sonucu veren ağırlık değerleri, aktivasyon ve iyileştirme fonksiyonlarını içerdikleri bilinmektedir. Probleme göre son katmanlarını uyarlayarak model tasarlayıp sonuçların test edilmesi mümkündür. Ayrıca kaybolan eğitim ve artan parametre sayısının çözümü için sunulan farklı yaklaşım tarzlarından en öne çıkanı 2015 ImageNet ILSVCR yarışmasında birinci olan ResNet-50 modelidir. ImageNet milyonlarca resim içeren veri setlerine sahip olup en iyi sınıflandırmanın yapılabilmesi ve en iyi derin öğrenme modellerinin ortaya çıkarılabilmesi için her sene yarışma düzenlenmektedir. Bu yarışmada Şekil-4.7 derin evrişimli modeller tasarlanarak sırasıyla 2008 yılında LeNet, 2012'de AlexNet, 2014'te GoogleNet, VGGNet, 2015'te ise ResNet-50 modelleri derece yaparak Şekil-4.9'te görüldüğü gibi önemli başarılar elde etmişlerdir. Yarışmada her ağ modelinin getirmiş olduğu yeni yaklaşımlar var. AlexNet'te paralel iki evrişimli derin ağın iki ayrı grafik işlemcisinde (GPU) eğitilip bağlantılar kurulması, GoogleNet'teki katmanların modüler halde tasarlanıp farklı kısımlarında sonuç üretebilmesi, ResNet'teki artık bağlantılar gibi yenilikçi fikirler öne çıkmaktadır. ResNet-50' in en önemli katkılarından birisi de ağın derinliğinin artmasına rağmen önceki katmanlardan gelen değerleri bir kısa yol Şekil-4.8 ile ileri katmanlara

ekleyerek işlem karmaşıklığını ve hesap süresini azaltmasıdır. Derinlik artarken model başarılı şekilde eğitilebilmiştir.



Şekil 4.7 ILSVCR yarışmasında kullanılan modeller katman sayıları ve elde edilen hata oranları (Wang vd. 2018)



Şekil 4.8 Resnet kısayol yaklaşımı

Yapılan çalışma, tıbbi görüntü sınıflandırması için önerilen en iyi öğrenme aktarım modellerini karşılaştırmaktadır. Önerilen model, kan hücrelerinin alt türlerini sayarak sınıflandırırken, diğer tarafta göğüs röntgenlerini kullanarak COVID-19 negatif veya pozitif sınıflandırmasına yeni bir yaklaşım sunmaktadır. Önerilen model, transfer öğrenmeden gelen ağırlıkları işlenerek, tam bağlı katmandan (fully connected) önce ve

düzleştirme katmanından (flatten) sonra otomatik kodlayıcı tarafından özellikleri indirgemiş, Adam, RmsProp ve Adadelta optimizatörleri ile test edip son olarak SVM ile sınıflandırmıştır. Beyaz kan hücrelerinde eğitim veriseti 9957 adet, doğrulama veri seti ise 2487 adet görüntü içermektedir. Covid-19 veri seti 269 adet normal 70 adet Covid-19 görüntüsü içermektedir. Model L2 (çekirdek düzenleme) parametresi 0.001 olarak ayarlanıp kayıp fonksiyonu “hinge” yapıldığında softmax aktivasyon fonksiyonu SVM sınıflandırıcı şeklinde çalışmıştır. 50 katmanlı Resnet-50 mimarisinde düzleştirme katmanından sonra 512 ve 256 nöronluk 2 adet yoğunlaştırma katmanı eklenerek 2048’den 256 ya indirgenmiş ve 1000 nöronlu tam bağlı katmana bağlanmıştır. En son 4 nöronluk çıktı katmanı ile bağlanarak model elde edilmiştir. Modelin toplam parametre sayısı 75.100.804, eğitilebilen parametre sayısı 75.047.684, eğitilmeyen parametre sayısı 53.120 olmaktadır. Sonuçlar, sırasıyla COVID-19 için yaklaşık %99 ve lökosit sınıflandırması için %97,3 gibi yüksek doğruluk oranlarıyla başarı göstermiştir. Sonuçlar ortaya koyduğumuz modelin diğer modellerle karşılaştırıldığında daha yüksek doğruluk oranlarına sahip olduğunu göstermektedir. Modeldeki artımlı katman sayıları, teoride görüntünün özelliklerini temsil etme yeteneğini geliştirir; bu nedenle, daha fazla katmanın potansiyel olarak başarılı bir şekilde çalışabilen gelişmiş ağ mimarileriyle oluşturabileceği sonucuna varılabilir. Burada önerilen model, ResNet-50 üzerinde otomatik kodlayıcı ve SVM sınıflandırıcıyı birleştirerek hem lökosit hem de göğüs röntgeni görüntülerinde daha yüksek doğrulukta sonuçlar elde etmiştir.

4.2.1 Öğrenme transferi

Önceden eğitilmiş modeller ImageNet ağırlıkları ile ayarlanmış ve başlatılmıştır. Beyaz kan hücreleri ve COVID-19 X-ışınları veri seti ile eğitilmiş ve sınıflandırma sonuçları sırasıyla ele alınmıştır. Veri kümeleri, yaygın olarak kullanılan Kaggle’da herkese açıktır. Önceden eğitilmiş bir model, tahmin yapmak için önceden kalibre edilmiş ağırlıkları kullanmanıza izin verme avantajına sahiptir.

Hedefe ulaşmak için, önceden eğitilmiş modelde, tam bağlı (FC) katmanlarından önceki klinik veri kümesinin eğitildiği evrimsel katmanlar dondurulur. Lökosit ve COVID-19 Göğüs Röntgeni için etiketli veri setinin yüzde 80’ ni denetimli modeller olarak FC

katmanları üzerinde eğitilir. Önceden eğitilmiş model, sınıflandırma yapmak için hâlihazırda kalibre edilmiş ağırlıkları kullanma avantajına sahiptir. Katmanları dondurduktan ve bazılarını eğitilebilir hale getirdikten sonra, bu katmanlar veri kümeleri ile eğitilir.

Öğrenme transferi ötesindeki mantık, klinik veri kümeleri klinik olmayan veri kümelerinden farklı olsa bile, görüntülerin çoğu için alt özelliklerin ortak olduğudur (Khozeimeh vd. 2021). Bu nedenle, iletilen parametreler, eğitim süresi ve bellek maliyetinin yanı sıra büyük bir veri kümesi gereksinimini azaltan temsili bir özellikler kümesi olarak hizmet edebilir (Sharma ve Mehra 2020).

Çizelge 4.5 Xception, VGG, Resnet, Inception, MobileNet ağırlık boyutları, parametreleri ve katmanları (Singh vd. 2021)

Model	Resim boyutu	Ağırlıklar boyut	Top1 doğruluk	Top5 doğruluk	Parameterler	Katman
Xception	299x299	88MB	0.790	0.945	22.910.480	126
VGG16	224x224	528MB	0.715	0.901	138.357.544	23
VGG19	224x224	549MB	0.727	0.910	143.667.240	26
ResNet50	224x224	99MB	0.759	0.929	25.636.712	168
InceptionV3	299x299	92MB	0.788	0.944	23.851.784	159
InceptionResNetV2	299x299	215MB	0.804	0.953	55.873.736	572
MobileNet	224 x 224	17MB	0.665	0.871	4.253.864	88

Görsel Geometri Grubu, ILSVRC 2014'te VGG-16'yı kurdu ve bu, o zamanlar ImageNet'te başarılı olan iki modelden biriydi (Weiss vd. 2016). İyileştirmeler, AlexNet'in büyük çekirdek boyutlu filtrelerini, VGG-16 ve VGG-19 için sırasıyla 16 ve 19'luk evrişim katmanlarıyla, birçok küçük boyutlu çekirdek filtresinin uygulanması ile sağlandı. CNN modellerinde artan katmanlar, kaybolan gradyanlar ve aşırı öğrenme sorunlarını da içeriyordu. Artık öğrenme yaklaşımı, ResNet modelinin bu sorun için bir çözüm oluşturmasını sağladı. Diğer önceki CNN modellerinden ayrı olarak, artıklar,

özellikler yerine ResNet tarafından her evrişim katmanı için girdilerden öğrenildi. Bu, kimlik kısayol bağlantıları (Simonyan ve Zisserman 2014) adlı bir yöntem kullanılarak elde edildi. ResNet-50, ResNet-101 gibi farklı ResNet modelleri çeşitli miktarda katman kullandı. Inception-V1 modeli, doğruluğu azaltmadan bellek kullanımını ve yürütme süresini azaltarak VGGNet' in verimliliğini arttırmayı sağladı (He vd. 2016). Bu, VGGNet'in tekrar eden veya sıfır aktivasyon işlevlerini ortadan kaldırarak elde edilmiştir.

Otomatik kodlayıcı denetimsiz bir modeldir. Orjinal verileri yeniden oluşturmak için kodlamak ve çözmek için verileri daha düşük boyutlara sıkıştırır. Girdinin sıkıştırılmış gösterimi, darboğaz katmanında tutulur. Bu nedenle, girdi verileri yeniden oluşturulurken çıktı birimlerinin miktarı girdilerin miktarına eşit olmalıdır (Liu vd. 2018). Genel olarak oto kodlayıcı, bir kodlayıcı ve bir kod çözücünden oluşur. Kodlayıcı, darboğaz katmanına girdi boyutunu düşürerek sıkıştırır ve kod çözücü, sıkıştırılmış verilerden orjinal girdiyi yeniden yapılandırır (Liu vd. 2018, Pu vd. 2016).

SVM, kullanılan çekirdeğin tipine bağlı olarak, doğrusal olarak ayrılabilen ve doğrusal olarak ayrılamayan problemleri çözebilmektedir. Probleme ve yapılacak göreve göre 'Linear', 'Polynomial', 'rbf' gibi çekirdekler seçilebilir. SVM' nin rolü, bir sınıflandırıcı olarak çok sayıda görevi çözmek için en uygun karar sınırlarının veya hiper düzlemlerin seçimine yardımcı olmaktır (Gondara 2016). Boser vd. (1992), orjinal maksimum pay hiperdüzlem algoritması için doğrusal bir sınıflandırıcı oluşturdu. Doğrusal olmayan sınıflandırma için hiper düzlemleri belirtmek için SVM formülü Denklem (4.3)'de görülebilir.

$$\lambda ||w||^2 + \left[\frac{1}{n} \sum_{i=1}^n \max(0, 1 - y_i(w^T x_i - b)) \right] \quad (4.3)$$

$$\hat{f} = \arg \min_{f \in \mathcal{H}} \hat{\epsilon}(f) + R(f) \quad (4.4)$$

$$\left[\frac{1}{n} \sum_{i=1}^n \max(0, 1 - y_i(w^T x_i - b)) \right] + \lambda ||w||^2 \quad (4.5)$$

4.2.2 Önerilen model

Önerilen modelde, klinik görüntüleri doğru sınıflandırmak, test etmek ve eğitmek için tercih edilen ön eğitilmiş transfer öğrenme yöntemlerinden biri olan Resnet-50 kullanılmıştır. Özellikleri azaltmak için otomatik kodlayıcılar uygulanmıştır. Özelliklerin azaltılması, öğrenme hızını artırır ancak aşırı öğrenmeyi engeller. Optimize ediciler, sınıflandırma için doğruluk performansını değerlendirmek üzere ayrı ayrı test edilmek üzere bir diziye verilir. Rmsprop ve Adadelta optimize edicilere kıyasla Adam, derin CNN mimarilerinde görüntü sınıflandırması için en iyi optimize edicidir. Sınıflandırma katmanında önceki katman olan Tam bağlı katmanında (FC) ağırlıklar alınmış ve çıktı katmanına bağlanmıştır. Modelde ağırlıkları değiştirmeyen ancak zaman tasarrufu ve hesaplama maliyeti sağlayan ilk 4 katman dondurulmuş ve son katmanla beraber düzleştirmeden sonra eğitilebilir katmanlar eklenmiştir. Önceki katmanların boyutlarının eğitilebilir katmanlarla birleştirilmesi düzleştirilmiş bir katman kullanılarak yapılır. Otomatik kodlayıcı, tüm özellik kümelerini daha fazla küçülten özellik azaltma rolüne sahiptir. Bu da tam bağlı (FC) katmanından önce (Dense) katman eklenerek uygulanır. Sırasıyla lökosit için (12.512 x 1000) ve Göğüs Röntgeni için (339 x 1000) öznitelik vektörü elde edildi. Tam bağlı katmanına bağlanan her veri kümesi ayrı ayrı 1000 özellik ile temsil edildi. Lökositlerin sınıflandırılması için 4 farklı sınıf türü ile sonuçlanacak kategorik sınıflandırma, COVID-19 veri setini pozitif veya negatif olarak sınıflandırmak için ikili sınıflandırma kullanıldı. Model, lökosit ve COVID-19 X ışınları veri kümelerinde ayrı ayrı çalıştırıldı, bu nedenle ilk model lökosit veri seti ile eğitildi ve çıktılar işlendi, ardından X ışınları veri kümesi için de aynı adımlar uygulandı ve model farklı özellikleri işlemek için eğitildi. Aktivasyon fonksiyonu “Softmax” üzerinde parametre ayarlamaları ile SVM gibi çalıştırılarak ile son adımda görüntüler sınıflandırıldı. Ön eğitilmiş modeller tıbbi görüntüler üzerinde çalıştırıldıktan sonra doğruluk sonuçları karşılaştırıldı. SVM’ nin, tam bağlı katmanına eklendiğinde softmax sınıflandırmaya göre daha iyi doğrulukla çalıştığı ve hataları azaltabileceği değerlendirilmektedir (Lee J ve Lee Y 2021). Ön eğitilmiş modeller tıbbi görüntüler üzerinde çalıştırıldıktan sonra doğruluk sonuçları karşılaştırıldı. Sonuç olarak lökosit %97 oranında alt sınıflara, COVID-19 ise %99 oranında doğrulukla pozitif ya da negatif gruplara sınıflandırıldı. Ön eğitilmiş modeller öznitelik indirgemeleri ve farklı optimize

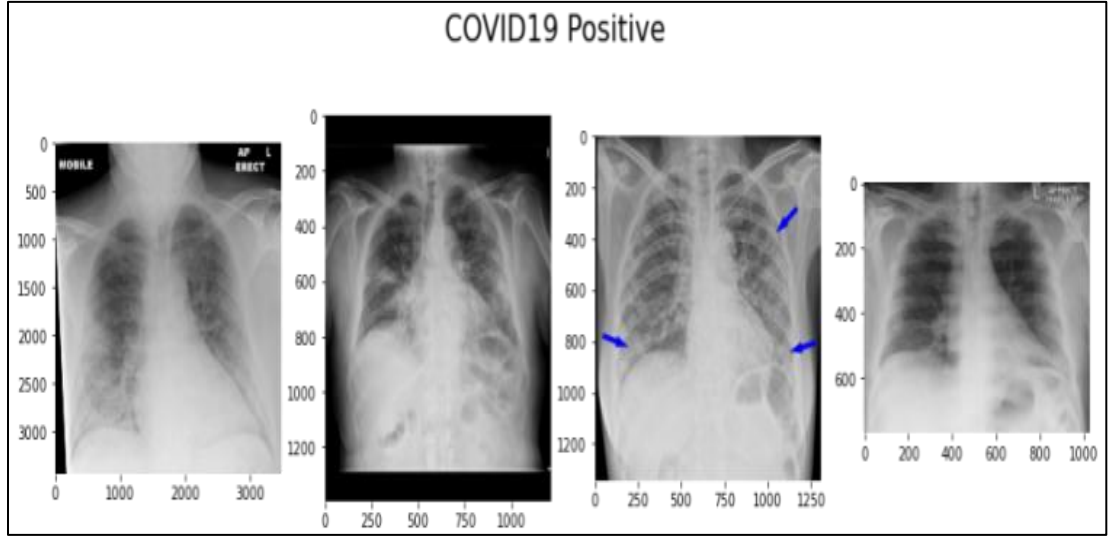
ediciler ve sınıflandırıcılarla denendiğinde; daha iyi doğruluk sonuçları veren modellerin oluşturulabileceği açıkça görülmektedir. Bizim çalışmamızda da en yüksek doğruluk sonuçları veren model geliştirildi. Özellik kümeleri ele alındıktan sonra, daha yüksek doğruluk sonuçlarıyla iyileştirilmiş sınıflandırma elde etmek için Softmax yerine SVM sınıflandırıcısı tercih edildi. Önerilen model şeması, ResNet-50-AE-SVM Şekil-4.9' da gösterilmiştir.



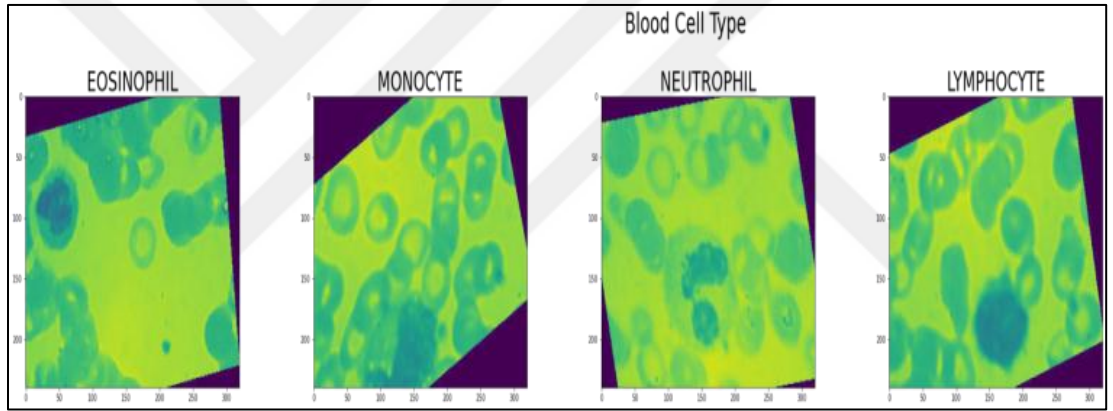
Şekil 4.9 Önerilen model Resnet50+AE+SVM veri kümesi

Bu çalışmada, sağlık alanında yapay zekâ çalışmaları için en çok tercih edilen ve erişime açık olan lökosit veri seti ve göğüs röntgeni veri seti kullanılmıştır. Eğitim için %80, test ve doğrulama için her veri setinin %20'si kullanılmıştır. Görüntüler Keras ile ön işleme tabi tutulmuş, yeniden boyutlandırılmış ve görüntülerin şekilleri (224, 224, 3) haline getirilmiştir. Daha sonra görüntüler özel dizilere (1, 224, 224, 3) dönüştürülmüş ve normalize edilmiştir.

COVID-19 göğüs röntgeni görüntülerinin bir örneği (Şekil-4.10' da) ve lökosit kan tipleri örneği (Şekil-4.11' de) verilmiştir.



Şekil 4.10 COVID-19 X ışınları pozitif sonuçları



Şekil 4.11 Kan hücresi türleri

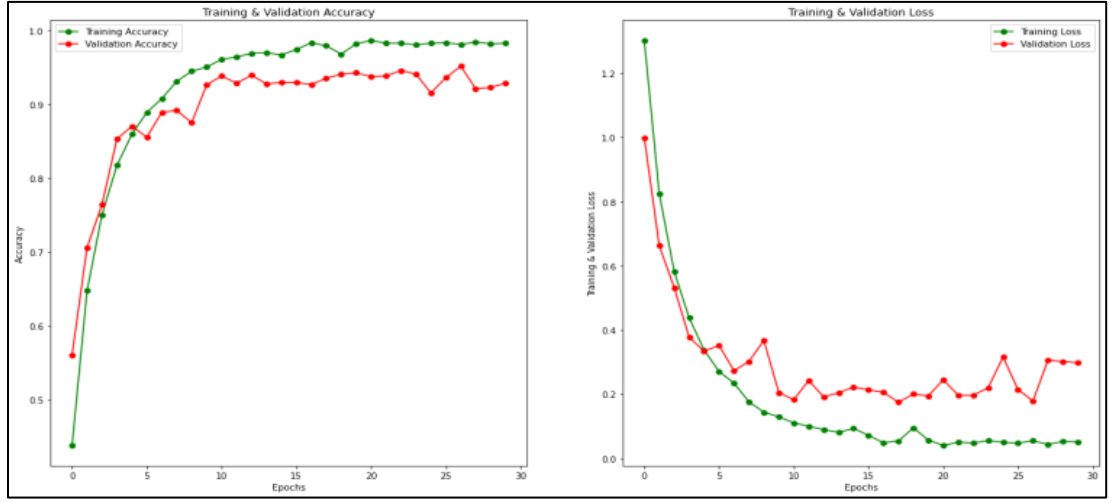
4.2.3 Elde edilen sonuçlar

Önerilen ResNet-50-AE-SVM modeli, i7 işlemcili, 32 GB RAM'li bir bilgisayarda ve Kaggle bulut sunucuları üzerinden çalıştırıldı. SVM sınıflandırıcısı kullanılarak test edildi. Deneysel sonuçların ve doğrulukların karşılaştırılması için VGG-16, ResNet-50 ve Inception-V3 olan diğer transfer öğrenme modelleri kullanılmıştır. Önerilen çerçevede, özellik çıkarımı için ResNet kullanılmıştır. Modelin üst katmanları devre dışı bırakıldı ve örnek görüntüleri eğitmek için özel yoğun katmanlarla değiştirildi. En alakalı özellikleri seçmek ve aşırı öğrenmeyi azaltmak için yoğun katmanların içinden ve sonrasında uygulanan oto kodlayıcı ile özellikler azaltıldı. Optimize edicilerden

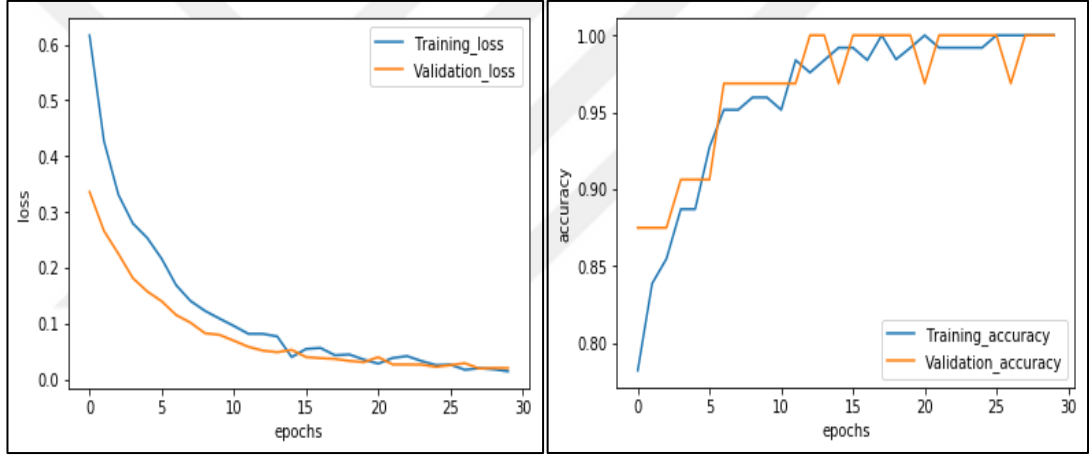
Adam, Rmsprop ve Adadelata, her biri ayrı ayrı test edilmek üzere modele bir dizi halinde eklendi. En iyi sonuçları veren aktivasyon işlevlerinden ReLu kullanıldı. Lökosit ve COVID-19 sınıflandırmaları için kayıp fonksiyonu sırasıyla kategorik ve ikili çapraz entropi olarak kullanıldı. Ancak, SVM'yi Resnet50 üzerinde uygulamak için hinge kayıp fonksiyonu ve 'l2' düzenlemesi kullanıldı.

Düzenleyiciler ve kayıp işlevleri kullanılarak, daha iyi sınıflandırma sonuçları elde etmek için modelin tam bağlantılı aşamasında SVM sınıflandırıcı uygulandı.

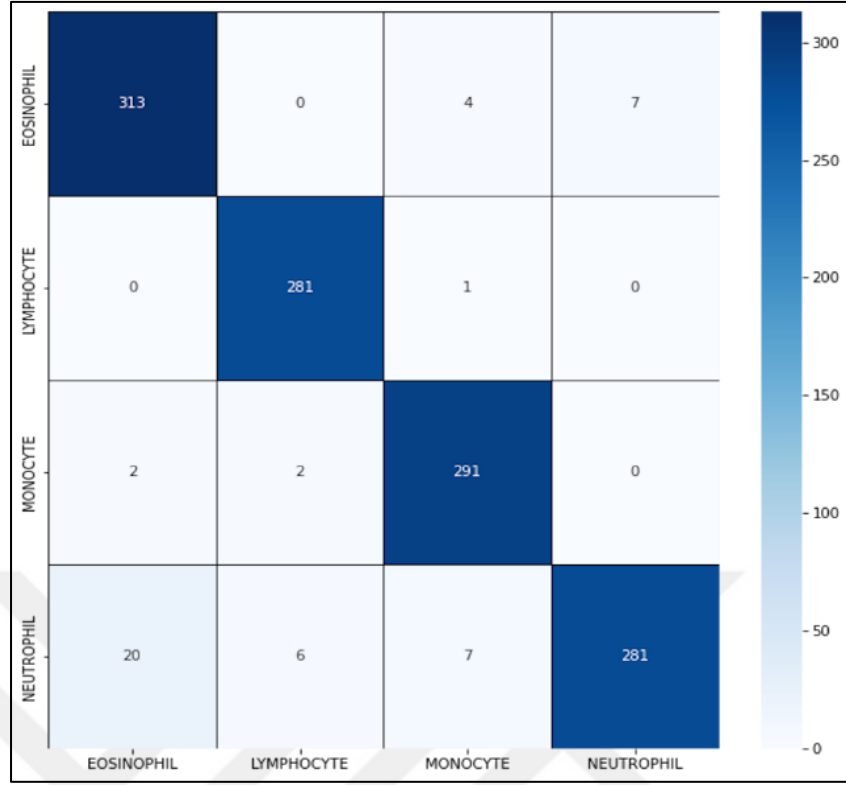
Karşılaştırma için çalıştığımız veri setinde başka transfer öğrenme modelleri de kullanılmıştır. VGG-16, ResNet-50 ve Inception-V3' te üst katmanlar devre dışı bırakıldı ve bunun yerine örnek veri kümesini eğitmek için modelin üstüne özel katmanlar eklendi. Çıkış katmanında sınıflandırma için SVM kullanılmıştır. Çizelge 4.6 ve Çizelge 4.7 da gösterildiği gibi, önerilen ResNet-50-AE-SVM modeli COVID-19 sınıflandırmasında lökosit için %97,3 ve göğüs röntgenleri için %99 olmak üzere en başarılı sonuçları elde etmiştir. Çalıştıkları tur süresi (epoch) boyunca doğruluk ve kayıp değerleri sırasıyla lökosit ve COVID-19 sınıflandırması için Şekil-4.12 ve Şekil-4.13'te gösterilmiştir. Şekil-4.14, Şekil-4.15, lökosit ve COVID-19 sınıflandırması için soldan sağa doğru karışıklık matrisi değerlerini göstermektedir. Önerilen model, en yüksek sınıflandırma doğruluğu ile optimum bir yürütme süresine sahiptir. ResNet-50, VGG-16 ve Inception-V3 modellerinin de ortaya koyduğumuz hibrit model kadar yüksek olmasa da %93'ün üzerinde doğrulukla iyi performans gösterdiği anlaşılmaktadır.



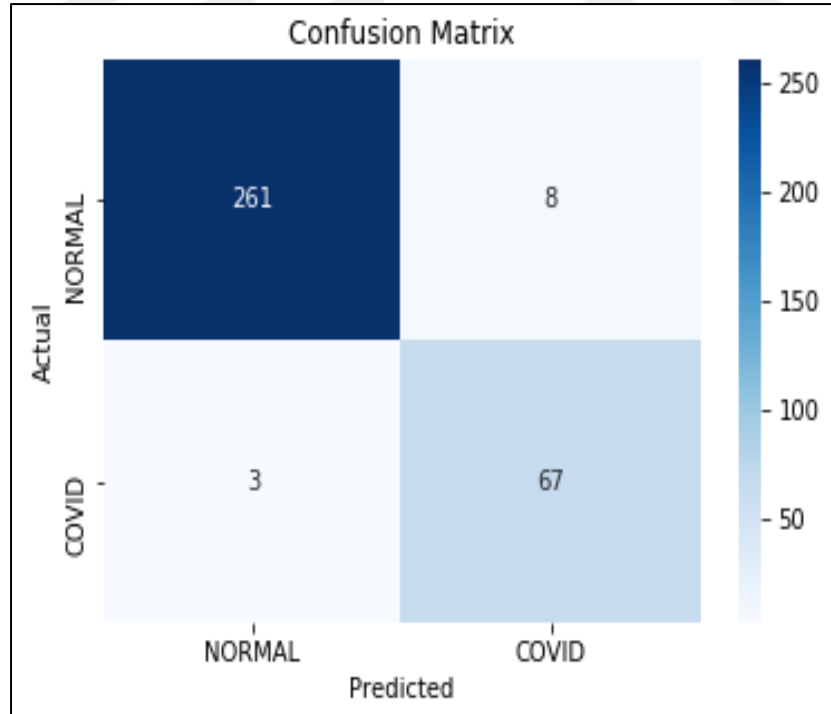
Şekil 4.12 Lökosit eğitim ve doğrulama veri seti için kayıp ve doğruluk sonuçları



Şekil 4.13 COVID-19 eğitim ve doğrulama veri seti için kayıp ve doğruluk sonuçları



Şekil 4.14 Lökosit karşılaştırma matrisi



Şekil 4.15 COVID-19 karışıklık matrisi

Çizelge 4.6 Önerilen model ve diğer yöntemler için doğruluk sonuçlarının karşılaştırılması

Veri Seti	Metot	Ortalama süre(s)	Doğruluk (test)
Lökosit	VGG16	3090.53	%93.4
COVID-19 X ışınları	VGG16	1720.10	%95.1
Lökosit	Resnet50	1210.90	%94.2
COVID-19 X ışınları	Resnet50	1121.25	%96.2
Lökosit	Inception-V3	2090	%93.2
COVID-19 X ışınları	Inception-V3	1688	%94.7
Lökosit	Resnet50+AE+SVM (Önerilen)	1250.32	%97.2
COVID-19 X ışınları	Resnet50+AE+SVM (Önerilen)	1123.45	%99

Çizelge 4.7 Beyaz kan hücreleri için f1 puanı ve diğer metrikler

Veri Seti	Kesinlik	Geri Çağırım	f1 Skoru	Veri Sayısı
Eozinofil	0.95	0.95	0.96	310
Lenfosit	0.98	1.00	0.99	320
Monosit	1.00	0.98	0.98	327
Nötrofil	0.95	0.94	0.96	295
Doğruluk			0.97	1252
Ağırlıklı ortalama	0.98	0.97	0.97	1252

Çizelge 4.8 COVID-19 için f1 puanı ve diğer metrikler

Veri Seti Covid-19	Kesinlik	Geri Çağırım	f1 Skoru	Destek
Negatif	0.99	0.98	0.99	269
Pozitif	0.98	0.96	0.97	70
Doğruluk			0.99	339
Ağırlıklı ortalama	0.98	0.98	0.98	339

Öğrenme transferi modelleri CNN mimarileri üzerinde inşa edilmiş olup, görüntülerden ayrıntılı özellikler elde etmek, daha iyi algılama ve sınıflandırma yapmak için çok sayıda konvolüsyon, havuzlama ve indirgeme (dense) blokları kullanır. Bununla birlikte, öğrenme transfer modelleri, katmanlar (dense) boyunca ve sonrasında özellik sayıları azaltılarak, farklı optimize ediciler ve sınıflandırıcılarla desteklenerek aşırı öğrenme ve sınıflandırma süresi azaltılır. Önerilen modellerde Adam, RmsProp ve Adadelta optimize edicileri ayrı ayrı test edilmiştir. Yoğun katmanlar yoluyla özellikleri azaltmak için kullanılan otomatik kodlayıcılar kullanılmıştır. Son aşamada ise bu uygulama için en etkili sınıflandırıcılardan biri olan SVM uygulanmaktadır. VGG-16, ResNet-50 ve Inception-V3 olan diğer transfer öğrenme modelleri ‘softmax’ sınıflandırıcı aktivasyonu ile aynı veri kümelerinde çalıştırıldı. Modellerden elde edilen doğruluk sonuçları %93’ün üzerindedir ve bu da

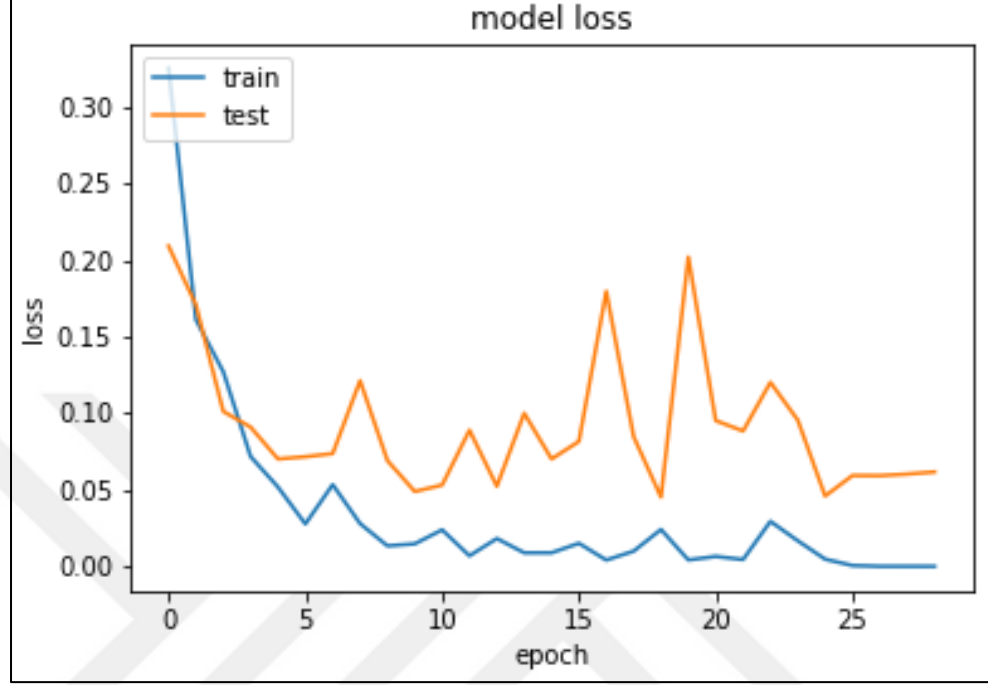
oldukça iyidir, ancak uyguladığımız yöntem en yüksek doğruluk oranlarını vermiştir. Önerilen çalışma tıbbi görüntü sınıflandırması için hızlı ve verimli bir yöntem ve modeli ortaya koymuştur.

4.3 Akciger Röntgeninden Tüberküloz Teşhisi

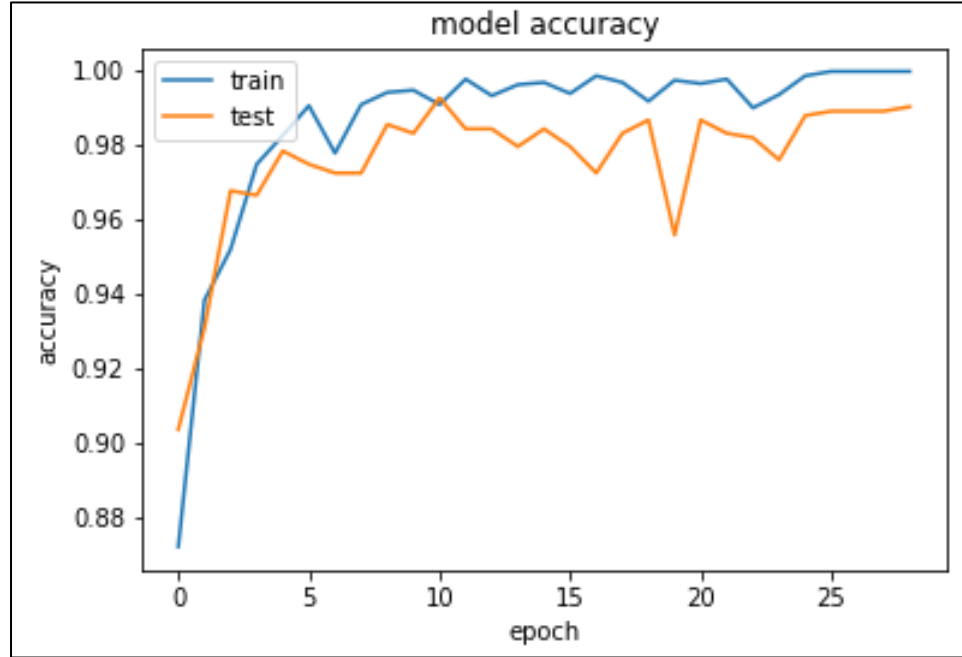
Son yıllarda yapay zekâ, sağlık sektöründe hastalıkların teşhisine yönelik çözümlerin önemli bir parçası haline geldi. Özellikle Evrişimsel Sinir Ağları (CNN) yöntemlerinin gelişmesiyle, X-ışınları, CT, MRI ve mikrobiyolojik görüntüler gibi tıbbi görüntülerden tanı koymak popülerlik kazandı (Pasa vd. 2019). Bunların yanı sıra tüberküloz tüm dünyada milyonlarca yeni vaka ve ölüme sebep olmasıyla insan sağlığını tehdit etmektedir. Neyse ki erken teşhis konulduğunda kolayca tedavi edilebilmekte, böylelikle vakaların artması ve ölüm riski de onlenmiş olmaktadır (Stirenko vd. 2018). Dolayısıyla bu çalışmada, X-ray görüntü veri setinden tüberkülozu erken teşhis etmek için yeni bir ince ayarlı CNN modeli önerilmiş ve model doğrulama verilerinde %98 oranında başarı sağlamıştır.

Önerilen yöntemde bir girdi görüntüsünün tüberkülozu gösterip göstermediğini tahmin etmek için CXR veri setini eğitip, test eden bir evrişimli ağ modeli kullanılmaktadır. Kaggle' dan alınan veri seti, 3500'ü normal ve 700'ü tüberküloz olmak üzere 4200 görüntü içermektedir. Model, veri setini %80 ile eğitmek ve %20 si ile test etmek için ayırmıştır. Görüntüler 250x250 diziye boyutlandırılmıştır. Evrişim filtreleri 32, 64, 128 (3x3) olarak ayarlandı ve düzleştirme katmanında 28800 özellik elde edildi. Düzleştirilmiş katmanda, özellikleri azaltmak için yoğun katmanlar eklendi (128) ve ardından çıktı katmanı, görüntülerin tüberküloz örneği olup olmadığını sınıflandırarak hastalık teşhisi sonuçlandırılmış oldu. Modeli oluşturmak için toplam 6 evrişimli, 2 yoğun ve 4 havuzlama (maxpool) katmanı kullanılmıştır. Optimize edici, aktivasyon fonksiyonlarının 'relu' olarak ayarlandığı ve çıktı katmanında en iyi sınıflandırma sonuçlarını elde etmek için 'softmax' olarak kullanıldığı 'Adam' olarak seçilmiştir. Çalışmamızda modelin öbekleme değeri 32, tur sayısı ise 40 olarak belirlenmiştir. Model başarıyla eğitildikten sonra, sınıflandırma için test doğruluğu sonuçları %99' un üzerinde çıktı. Karışıklık matrisi, eğitim, test doğruluğu ve kayıp değerlerinin grafikleri Şekil-4.16, Şekil-4.17 ve Şekil-4.18' de görülebilir. Model

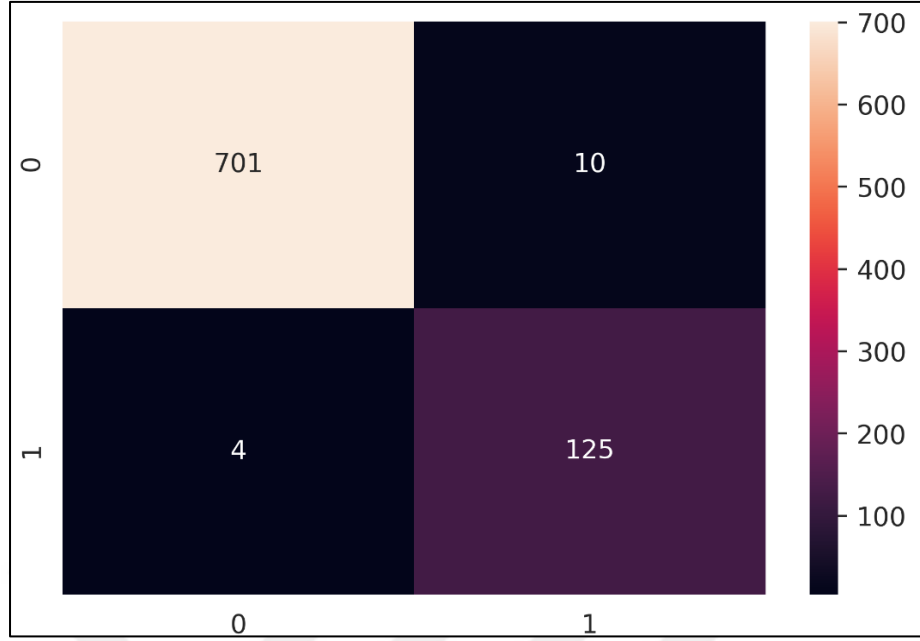
değerlendirmesi kesinlik, geri çağırma, f-skör değerleri üzerinden yapılmış olup detaylar Çizelge-4.9' da gösterilmektedir.



Şekil 4.16 Tüberküloz eğitim ve test veri seti kayıp değerleri



Şekil 4.17 Tüberküloz eğitim ve test veri seti doğruluk değerleri



Şekil 4.18 Tüberküloz veri seti için modelin karşılaştırma matrisi

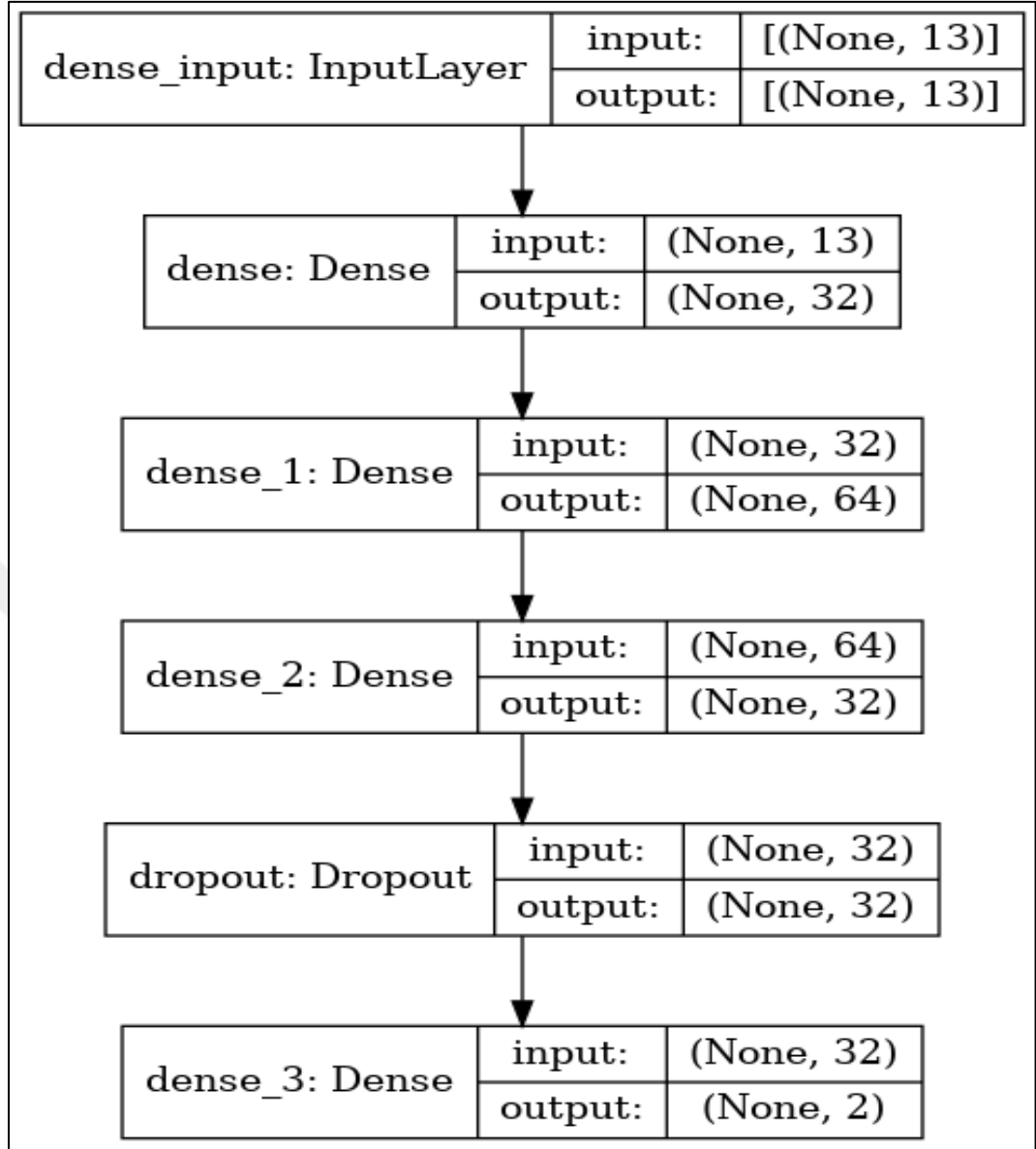
Çizelge 4.9 Tüberküloz teşhisi için önerilen modelin performans verileri

	Doğruluk	Geri Çağırım	f1-Skoru	Veri Sayısı
Normal	0.993	0.996	0.994	679
Tüberküloz	0.981	0.969	0.975	161
Doğruluk	0.990	0.990	0.990	0.990476
Makro Ort	0.987	0.982	0.985	840
Ağırlıklı Ort	0.990	0.990	0.990	840

4.4 Kalp Krizi Tahmini

Kardiyovasküler hastalıklar modern dünyada yüksek artış göstermektedir. Tüm dünyada her yıl kalp hastalıklarından 17,7 milyon ölüm meydana gelmektedir. Gelişmekte olan ülkelere göre özellikle Asya ve Afrika bölgelerinde, kalp krizinin geç tespit edilmesinin birçok insanın ölümüne neden olduğu bilinmektedir (Sayad ve Halkarnikar 2014). Kalp krizinin erken bir aşamada saptanması, atağın önlenmesine önemli ölçüde yardımcı olabilmektedir. Önerilen çalışmanın temel amacı, gerçek yaşam veri setlerini olası bir kalp krizini zamanında tahmin etmeye yardımcı olabilecek şekilde kullanmaktır. Bu

amaca hizmet etmek için çeşitli veri analizi ve veri madenciliği teknikleri vardır (Gomathi ve Priyaa 2016). Pek çok insan, daha önce keşfedilmemiş veya basitçe görmezden gelinen belirtiler yaşayarak ölür. Gerçek ortaya çıkmadan önce kalp hastalığını tahmin etmek çok büyük önem arz etmektedir. Dolayısıyla tez çalışmamızın bu kısmında kritik önem arz eden kalp krizi risk tahmini, hastaların bulgularını içeren metin veri seti üzerinde DNN (Derin yapay sinir ağı) modeli kullanılarak yapılmıştır. Bu alanda yazılım çözümü ihtiyacı da göz önünde bulundurularak derin öğrenme modellemesi ile alternatif bir model önerilmiştir. Sonuç olarak oluşturduğumuz model kalp krizi tahmininde %89 üzerinde doğrulukla tatmin edici seviyede başarı sağlamıştır. Çizelge 4.8’ de kalp krizi tahmini için kullandığımız veriler görülmektedir. Her bir veri sütunu tahminde kullanılan bir özellik olarak değerlendirilmekte olup, hedefte (target) verilen sonucu doğru tahmin etmek üzere eğitilmekte ve test verisi ile test edilmektedir. Veri seti metin verisinden oluşmakta olup tür olarak tamsayı ve ondalıklı sayılardan oluşmaktadır. Verilerden kategorik olanlar mevcut kategori sayılarına göre sayısallaştırılmış, sayısal değer içeren alanlar notmalizasyon edilerek 0-1 aralığına normalizasyon edilmiştir. 13 adet sütun girdi olarak modele eklenmekte ve sonuç olarak kalp krizi riski var ya da yok olarak 2 farklı çıktı değeri üretilmektedir.



Şekil 4.19 Kalp krizi tahmini için oluşturulan modelin şeması

Şekil-4.19’ da oluşturulan modelin tasarımı görülmektedir. Kalp krizi metin veri seti üzerinde tasarlanan evrişimli sinir ağı modeli çalıştırılmıştır. Elde edilen %89 civarındaki doğruluk sonucu evrişimli sinir ağları modellerinin metin veri setleri üzerinde çalıştırıldığında da gayet başarılı olduğunu göstermektedir.

Çizelge 4.10 Kalp krizi tahmini için kullanılan hasta bulguları

	Ozellikler	Veri Sayısı	Veri Tipi
1	Hastanın Yaşı (Ing. age)	303	int64
2	Cinsiyeti (Ing. sex)	303	int64
3	Göğüs Ağrısı Şiddeti (1-4) (Ing. cp)	303	int64
4	Dinlenme Kan Basıncı (Tansiyon) (Ing. trestbps)	303	int64
5	Kandaki Serum kolesterol miktarı (mg/dl) (Ing. chol)	303	int64
6	Açlık Kan Şekeri > 120 (mg/dl) (Ing. fbs)	303	int64
7	Dinlenme Hali Elektrokardiyografi Sonuçları (0-1-2)(Ing. restecg)	303	int64
8	Ulaşılan Maximum Kan Atış Hızı (Ing. thalach)	303	int64
9	Egzersiz Nedenli Angina (Ing. exang)	303	int64
10	Egzersize Bağlı ST Depresyonu (Ing. oldpeak)	303	float64
11	Pik Egzersiz ST Segmentinin Eğimi (Ing. slope)	303	int64
12	Floroskopi ile Renklendirilmiş Büyük Damar (0-3) Sayısı (Ing. ca)	303	int64
13	thal: 3 = normal; 6 = sabit kusur; 7 = tersinir kusur (Ing. thal)	303	int64
14	Target	303	int64

Çizelge 4.11 Kalp krizi verilerinin istatistiki dağılımını göstermektedir

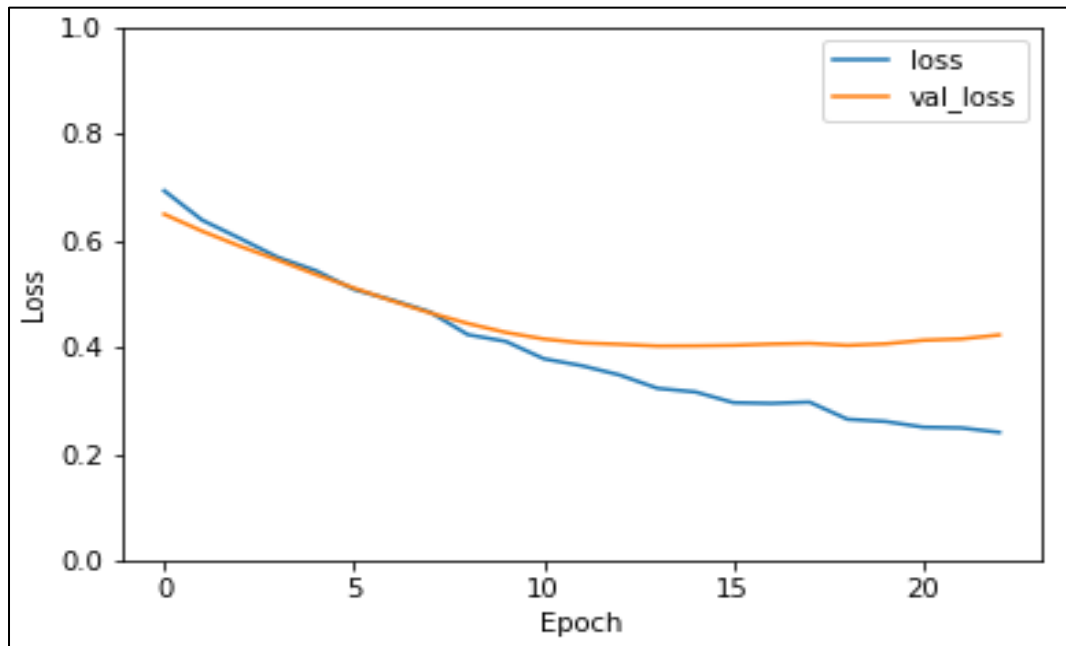
	yaş	Cinsiyet	Cp	Rps	Chol	fbs	restecg	thalach	Exang	oldpeak	Slope	ca	Thal	sonuç
#	30 3	30 3	303	30 3	30 3	303	30 3	30 3	303	30 3	30 3	303	30 3	30 3
me an	54.3 6634	0.68 3168	0.96 6997	131. 6238	246. 264	0.14 8515	0.52 8053	149. 6469	0.32 6733	1.03 9604	1.39 934	0.72 9373	2.31 3531	0.54 4554
std	9.08 2101	0.46 6011	1.03 2052	17.5 3814	51.8 3075	0.35 6198	0.52 586	22.9 0516	0.46 9794	1.16 1075	0.61 6226	1.02 2606	0.61 2277	0.49 8835
mi n	29	0	0	94	12 6	0	0	71	0	0	0	0	0	0
25 %	47. 5	0	0	12 0	21 1	0	0	13 3.5	0	0	1	0	2	0
50 %	55	1	1	13 0	24 0	0	1	15 3	0	0.8	1	0	2	1
75 %	61	1	2	14 0	27 4.5	0	1	16 6	1	1.6	2	1	3	1
ma x	77	1	3	20 0	56 4	1	2	20 2	1	6.2	2	4	3	1

Çizelge 4.11 de kalp krizi verilerinin ortalama, standart sapma, en küçük, en büyük değerleri ve %25, %50, ve %75' lik dilimlere ait ortalama verileri görülmektedir. Bu tablo verilerin dağılımı hakkında model tasarımı için önemli bilgileri içermektedir.

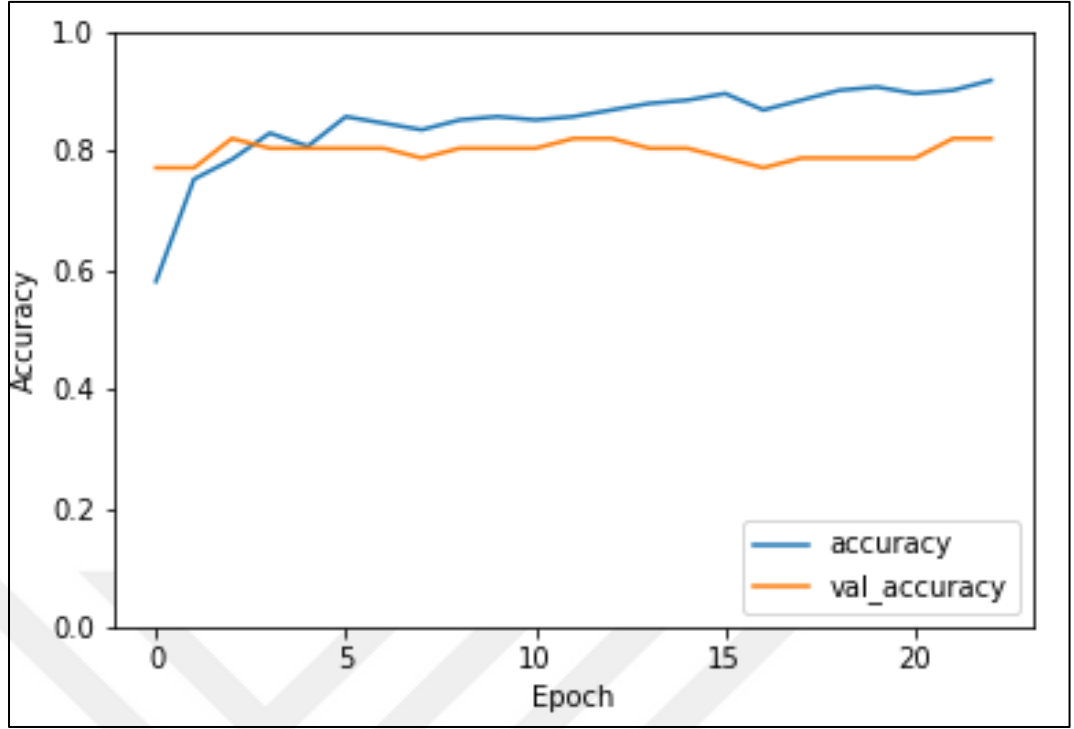
Çizelge 4.12 Kalp krizi veri setinden çekilen örnek 5 veri

Id	Yaş	Cinsiyet	Cp	Trestbps	Chol	Fbs	Restecg	Thalach	Exang	Oldpeak	Slope	ca	thal	Sonuç
1	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
2	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
3	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
4	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
5	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1

Çizelge-4.12' de örnek veriler ilk 5 kayıt listelenecek şekilde çekilip görüntülenmiştir. Model eğitim ve test verisi olarak gruplandırılmış ve eğitim seti üzerinde eğitildikten sonra test verisi üzerinden test edilmiştir. Modelin en iyi performans gösteren doğruluk ve kayıp (loss) değeri 0.3887 - doğruluk (accuracy): 0.8852 elde edilmektedir.

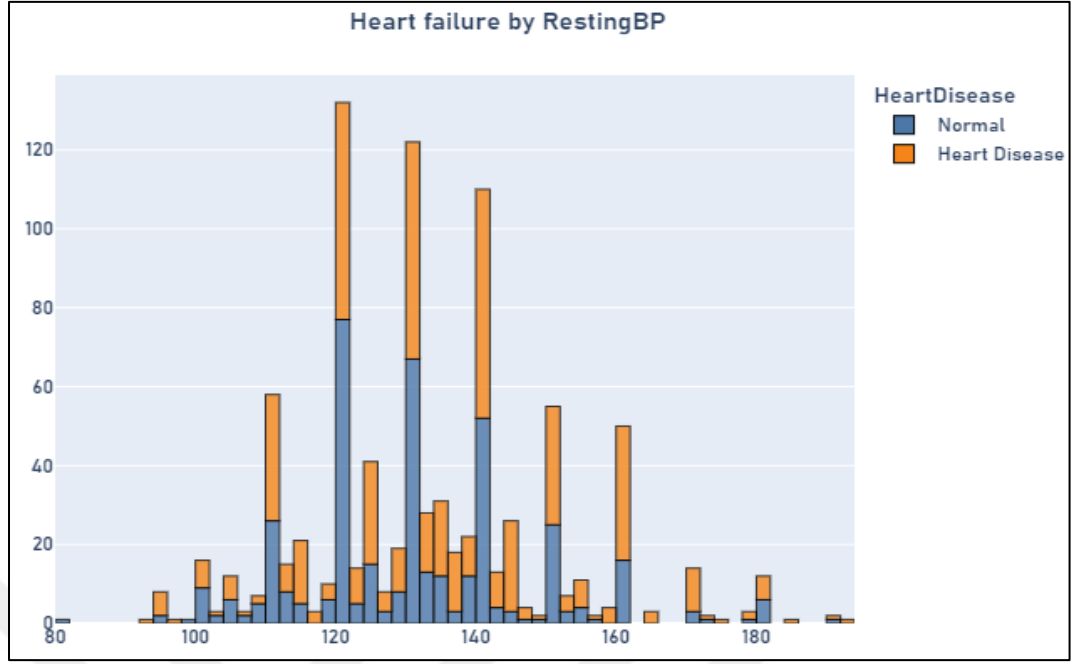


Şekil 4.20 Kalp krizi eğitim ve test veri seti kayıp değerleri

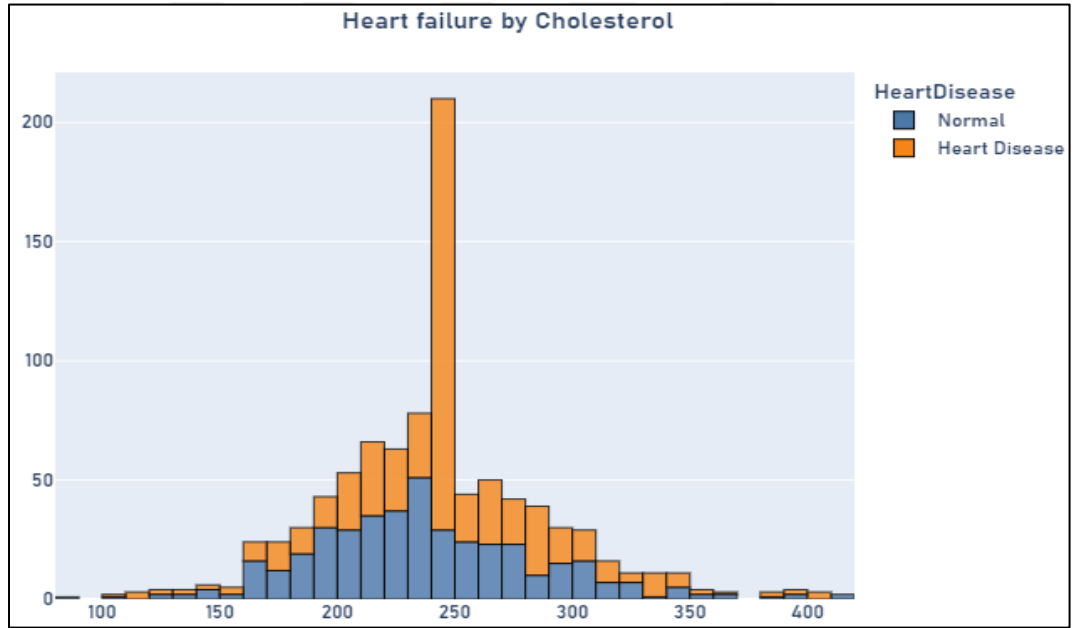


Şekil 4.21 Kalp krizi eğitim ve test veri seti doğruluk değerleri

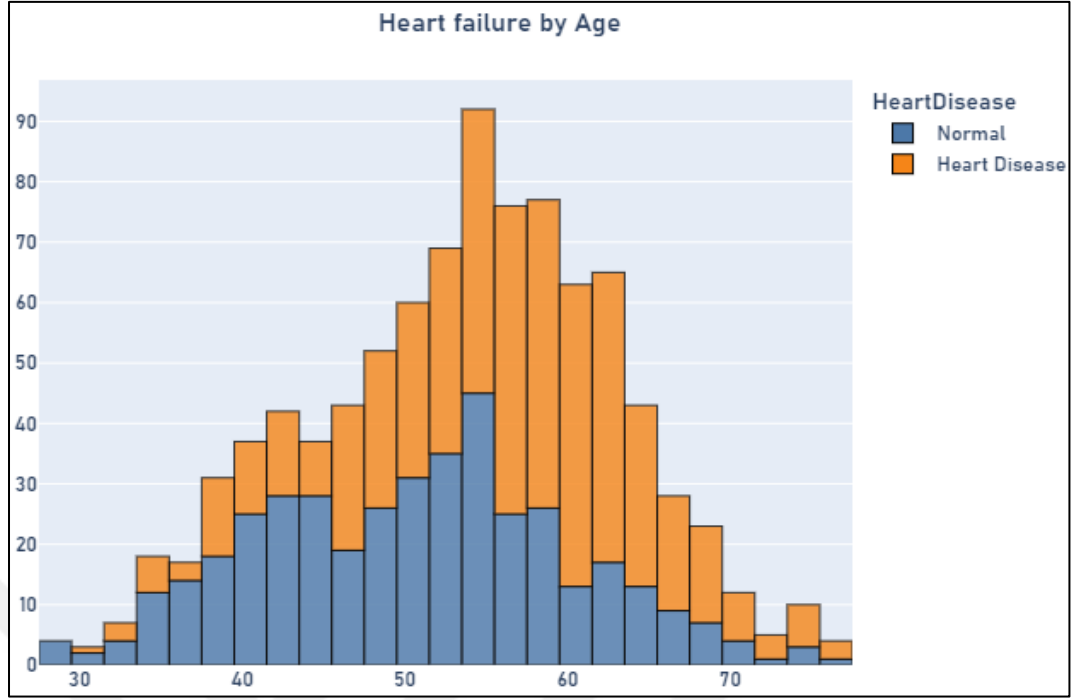
Şekil-4.20 ve Şekil-4.21’ de eğitim ve test veri setleri için 50 tur boyunca çalıştırılıp erken sonlandırma ile elde edilen kayıp ve doğruluk değerleri görülmektedir. Model başarılı bir şekilde eğitilmekte ve test verisinde başarılı doğruluk sonuçları vermektedir. Hastalarının dinlenme halindeki kan basınçları Şekil-4.22’ de kolesterol değerleri Şekil-4.23’ de, yaşları Şekil-4.24’ de, ST eğrileri Şekil-4.25’ de, EKG değerleri Şekil-4.26’ da, göğüs ağrı tipleri Şekil-4.27’ de, cinsiyetleri Şekil-4.28’ de ve egzersiz anjin değerleri Şekil-4.29’ da grafiklerde görülmektedir. Bu değerler kalp krizinin teşhisi için hayati öneme sahiptir.



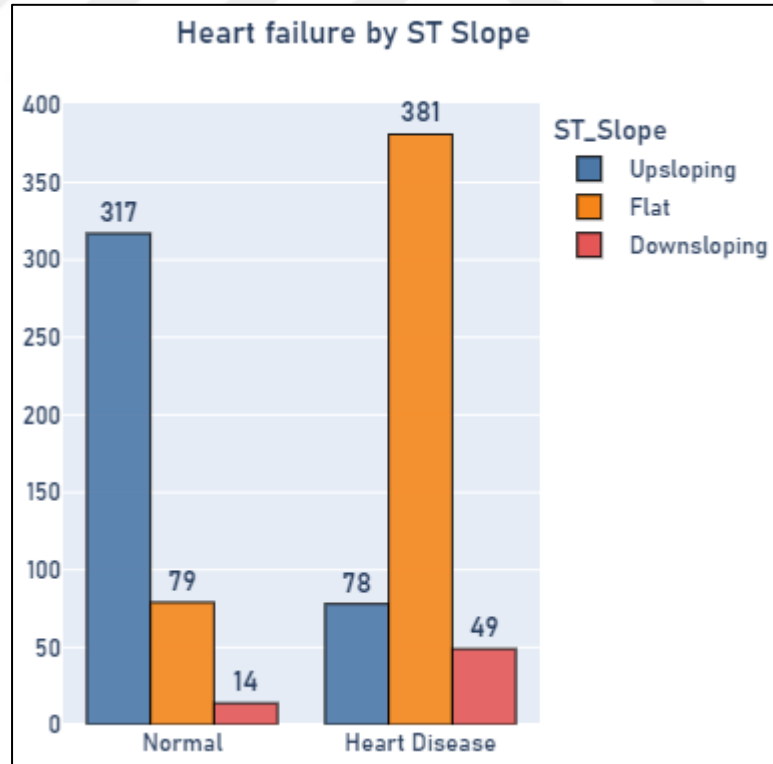
Şekil 4.22 Dinlenme kan basıncına göre kalp krizi riski dağılımı



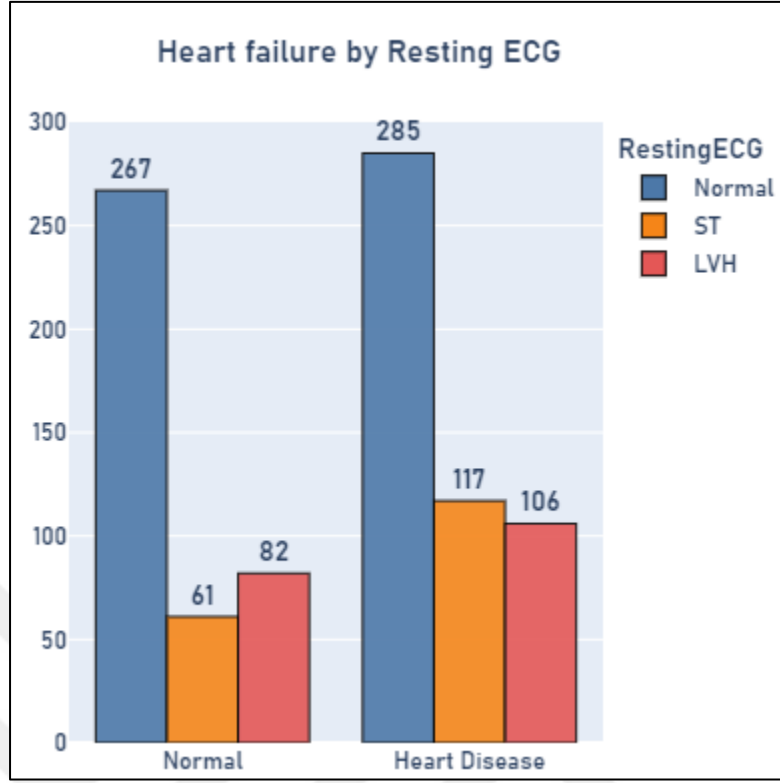
Şekil 4.23 Kolesterol değerlerine göre kalp krizi riski dağılımı



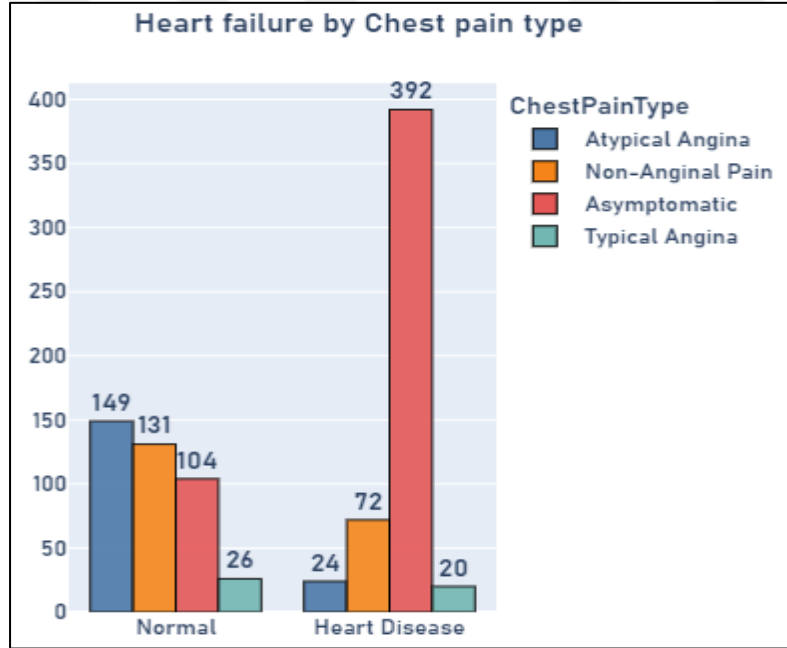
Şekil 4.24 Hastanın yaşına göre kalp krizi riski dağılımı



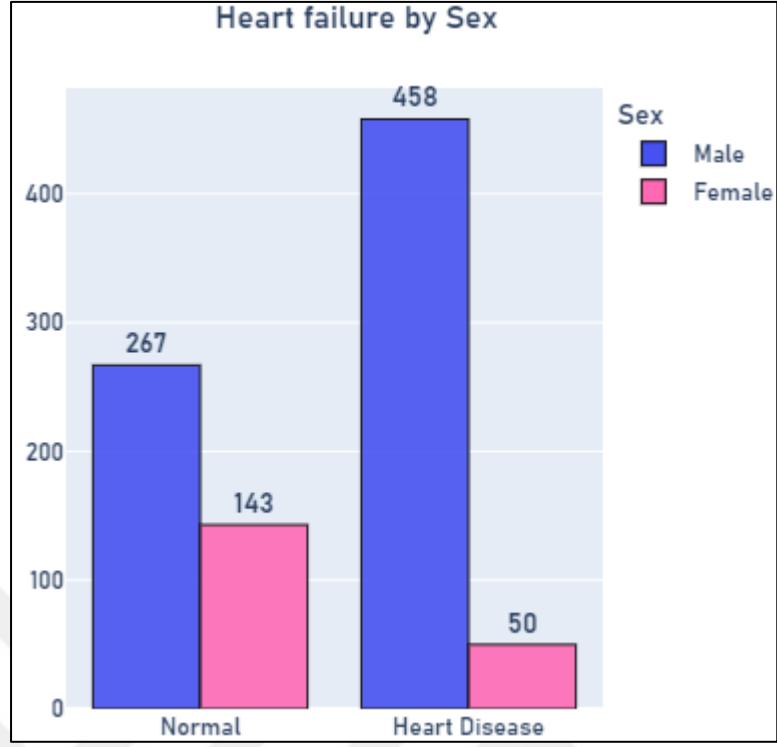
Şekil 4. 25 Hastanın ST eğrisi değerine göre kalp krizi riski dağılımı



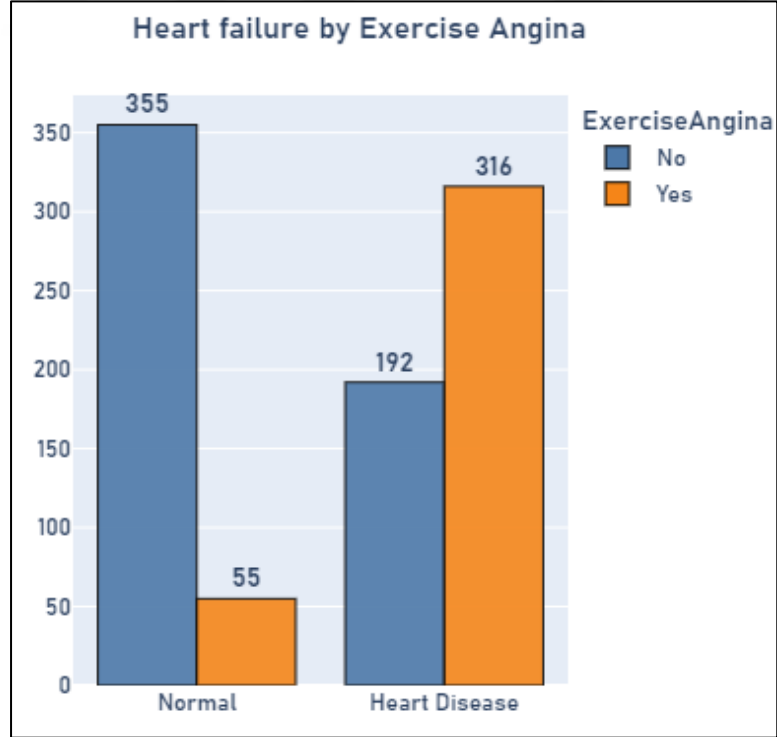
Şekil 4.26 Hastanın EKG değerine göre Kalp Krizi Riski dağılımı



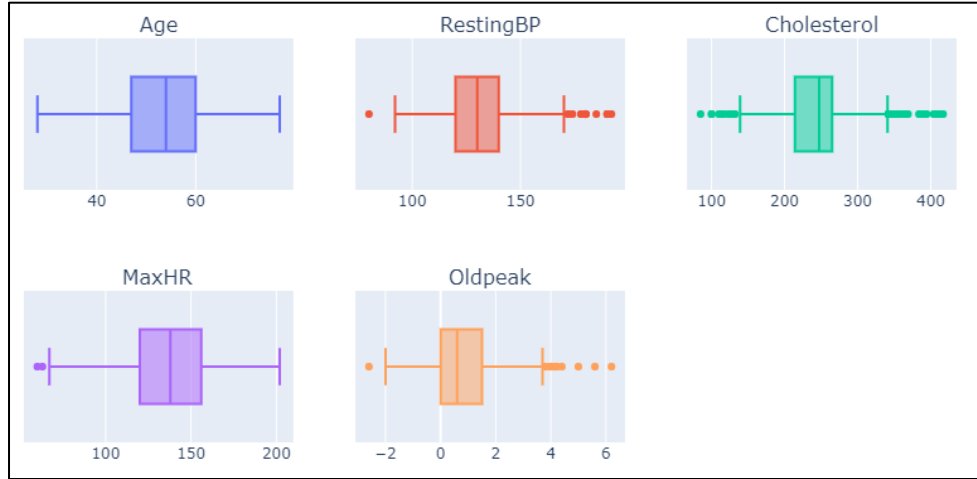
Şekil 4.27 Hastanın göğüs ağrısı tipine göre kalp krizi riski dağılımı



Şekil 4.28 Hastanın cinsiyetine göre kalp krizi riski dağılımı

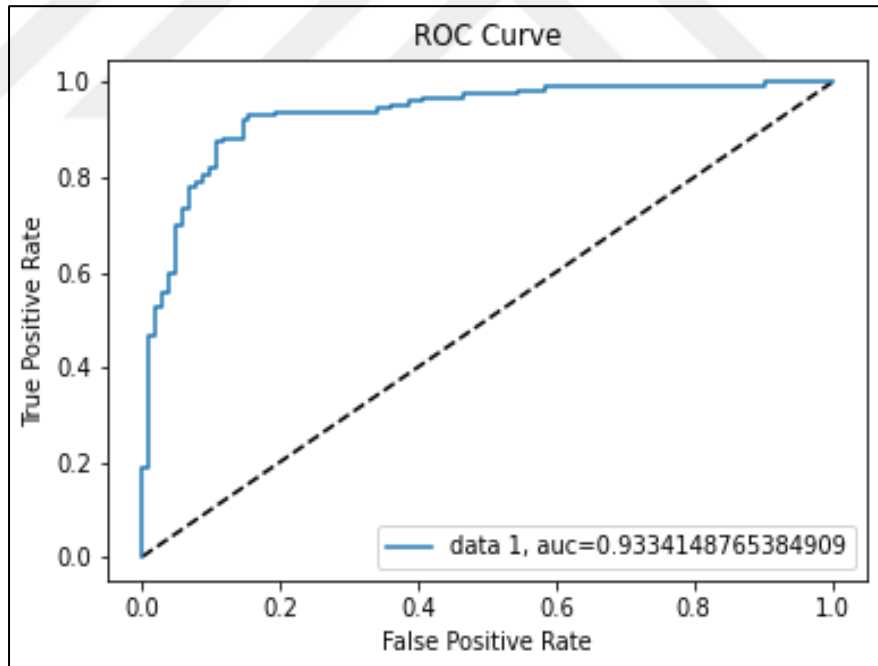


Şekil 4.29 Hastanın egzersiz anjin tipine göre kalp krizi riski dağılımı



Şekil 4.30 Hastanın kalp krizi riski değerleri dağılımı

Şekil-4.30 de hastaların kalp krizi riskinde değerlendirilen değerlerinin dağılımları gösterilmiştir.



Şekil 4.31 Kalp krizi riski modeli ROC eğrisi

Şekil 4.31 de ROC eğrisi ile doğru pozitiflerin, yanlış pozitiflere olan kesri grafikte gösterilmiştir. Tez kapsamında derin öğrenme modeli olan evrişimli sinir ağı kalp krizi risk bulgularını içeren 303 hastanın metin veri seti üzerinde eğitilmiş ve test edilmiştir.

Model tasarlanırken veri seti eğitim ve test için %80'e %20 olarak ayrılmış, sınıflandırıcı için Sofmax tercih edilmiştir. Düşürme değeri (dropout) farklı değerler test edilerek 0.2 belirlenmiştir. Optimizasyon algoritması Adam öğrenme oranı ise 0.001 olarak düzenlenmiştir. Ölçüt olarak doğruluk ölçüt alınmış ve erken öğrenme parametresi ile öğrenme bittiğinde durması sağlanmış, tur sayısı 50 olarak belirlenmiştir. Model başarılı şekilde çalışmış ve kalp krizi riski olan hastaları %89 doğrulukla tespit etmiştir. Burada daha önce medikal görüntüleri sınıflandırdığımız modeller üzerinde değişiklik yaparak medikal metin veri setine uyarladık. Dolayısıyla güncellerek aynı derin öğrenme modeli üzerinden hem görüntü hem metin veri seti eğitilmiş ve yüksek doğrulukla sınıflandırma yapılmıştır.

5. SONUÇ

Tez çalışmamızda sağlık alanında derin öğrenme tekniklerinin görüntü veri setlerine uygulanması ile bu alanda önemli katkılar sağlayarak teşhis ve sınıflandırma işlemlerinin yapay zekâ çözümleri ile daha hızlı ve doğru yapılabilmesi hedeflenmiştir. Tez boyunca daha çok üzerinde durulan çalışma bir derin öğrenme modeli olan evrişimli sinir ağıdır. Evrişimli sinir ağılarını temel alan model hem beyaz kan hücreleri sınıflandırılmasında hem de göğüs röntgen görüntülerinden COVID-19 ve tüberküloz teşhisinde başarılı sonuçlar vermiştir. Ayrıca hastaların kalp krizi bulgularından oluşan metin veri seti üzerinde çalıştırılarak yüksek doğruluk oranıyla risk tahminleri yapan derin öğrenme modeli geliştirilmiştir. Hibrit modeller tasarlanıp uygulanırken veri setine ve probleme uygun iyileştirmeler ve değişiklikler yapılmıştır. Örneğin beyaz kan hücreleri sınıflandırılmasında kendi oluşturduğumuz evrişimli ağ modelinin hiper parametre seçimlerini genetik algoritma ile yapması sağlanırken bir diğer uygulamamızda öğrenme transferi yöntemleri çalıştırılmış otomatik kodlayıcı ve SVM sınıflandırıcı ile iyileştirilmiş ResNet-50 ile oldukça başarılı sonuçlar elde edilmiştir. Aynı zamanda optimize edilmiş transfer öğrenme modelimiz göğüs röntgenlerinde de çalıştırılmış ve COVID-19 teşhisinde %99 doğrulukla sınıflandırma başarısı elde edilmiştir. Tez konumuzun temel aldığı diğer bir alan olan metin tabanlı çıkarım için yine evrişimli sinir ağıları modeli kullanılmış ve kalp krizi veri seti üzerinde çalıştırılmıştır. Geliştirilen model, metin veri seti üzerinde çalıştırılmadan önce probleme uygun şekilde model düzenlenmiş ve %89 doğruluk oranı ile kalp krizi riski tahmini yapılabilmesi sağlanmıştır. Bütün bu uygulamalar ve başarılı test sonuçları şunu göstermiştir ki, derin öğrenme; hızla artan sağlık verilerinin de katkısı ile, bu verileri analiz edecek derin öğrenme modelleri geliştirilerek erken teşhis, tanı ve mevcut durumdan ileriye yönelik tahminler üretilerek sağlık alanında devrim niteliğinde gelişmelere katkı sağlanmış olacaktır. Tez süresince yaptığımız çalışmalarda birçok model ve bu modellere ait farklı parametreler değişiklikler yapıp geliştirilerek örnek çalışmalar ile test edilmiştir. Öğrenme transferi ile önceden eğitilmiş model ağırlıkları kullanılmış ve bunun üzerine kendi kullandığımız veri setleri de eğitilerek eğitim süresini kısaltarak doğruluğu artırılmıştır. Yine bu modelde düzleştirme katmanından sonraki ve tam bağlı katmandan önceki bölümde yoğunlaştırma katmanları eklenmiştir.

Bu katmanlar ile otomatik kolayıcının kodlama kısmı uygulanarak parametre boyutu indirgenmiş ve sınıflandırıcı olarak SVM ve Softmax ile ayrıca çalıştırılmıştır. Kendi tasarladığımız derin öğrenme modellerinde can alıcı nokta hiper parametrelerin doğru ayarlanması iken öğrenme transferinde önemli olan modeli kendi veri setimize uyarlayarak en iyi sonucu elde etmektir. Bunun için öğrenme transferi yönteminde son katmandan önce eğitim dondurularak üzerine farklı katmanlar eklenebilmekte kendi veri setlerimiz ile eğitilebilmekte ve farklı sınıflandırıcılar uygulanabilmektedir. Nitekim bu çalışmada günümüzde en çok tercih edilen sezgisel parametre iyileştirme yöntemlerinden genetik algoritma ve en çok kullanılan öğrenme transferi modelleri geliştirilerek iyileştirilerek uygulanmıştır. Veri setleri seçiminde, bütün araştırmacılara açık olarak sunulan ve üzerlerinde birçok derin öğrenme modelinin geliştirildiği en çok kullanılan ve çalışılan veri setleri tercih edilmiştir. Kaggle web sitesi üzerinde birçok medikal görüntü veri seti paylaşılmakta, en iyi sınıflandırma ve tahminleme modellerinin geliştirilebilmesi için yarışmalar düzenlenmektedir. Bu tez çalışmasında da Kaggle üzerinden elde edilen medikal görüntü veri setleri üzerinde evrişimli sinir ağı modelleri geliştirilmiş, farklı sınıflandırıcılar ve sezgisel iyileştirme yöntemleri uygulanarak iyileştirilmiştir. Bu bağlamda gerek genetik algorithmadan gerek otomatik kodlayıcı, SVM ve öğrenme transferi modellerinden faydalanılmış, çalıştırılan modeller ve elde edilen sonuçlar karşılaştırmalı olarak değerlendirilmiştir.

Önerdiğimiz model ve Resnet50-SVM modeli beyaz kan hücreleri sınıflandırmasında %97, COVID-19 sınıflandırılmasında %99 doğruluk sağlarken, iyileştirilmiş evrişimli ağ modelimiz kalp krizi riski tahmininde %89 doğruluk oranı ile mevcut birçok modelden daha yüksek oranda sınıflandırma ve tahmin başarısı göstermiştir.

5.1 Gelecek Çalışmalar

Yaptığımız çalışmanın tıp alanında yapay zekâ destekli teşhis, analiz ve tahminleme sistemleri oluşturulması için isteklendirici olacağını ve bu alanda daha birçok geliştirmenin önünü açacağını düşünüyoruz.

Bununla birlikte geliřtirdiđimiz modeller gri kurt, paracık sr iyileřtirmesi gibi farklı sezgisel iyileřtirme algoritmaları kullanılarak test edilebilir. Yine đrenme transferinde Xception, VGG-19, MobileNet gibi farklı modeller ile de test edilerek sonuları deđerlendirilebilir.

Ayrıca geliřtirdiđimiz modeller ve model ađrılıkları (eđitilmiş model) dosyalarda kaydedilerek daha sonra da kullanılabilir. Benzerlik gsteren bařka veri setlerinde kullanılacak eđitim zamanından kazanılarak daha yksek dođruluklu sonular elde edilebilecektir. Dahası kaydedilen model bir web uygulamasına ya da mobil uygulamaya dâhil edilerek arayz zerinden grnt ya da hasta verisinin girilerek tahmin sonucunun uygulama zerinden grntlemesi sađlanabilir. Bylece derin đrenmenin sađladıđı avantajlar bir mobil uygulama zerinden eriřilip kullanılacak kadar kolaylařacaktır.

KAYNAKLAR

- Agarap, A. F. 2017. An Architecture Combining Convolutional Neural Network (CNN) and Support Vector Machine (SVM) for Image Classification. arXiv preprint arXiv:1712.03541.
- Akca, M.F. 2020. Nedir Bu Destek Vektör Makineleri? (Makine Öğrenmesi Serisi-2), Web Sitesi: <https://medium.com/deep-learning-turkiye/nedir-bu-destek-vekt%C3%B6r-makinaları-makina-%C3%B6%C4%9Frenmesi-serisi-2-94e576e4223e> Erişim Tarihi: 18.4.2022.
- Akramifard, H., Firouzmand, M. and Moghadam, R. A. 2012. Extracting, Recognizing, and Counting White Blood Cells From Microscopic Images By Using Complex-Valued Neural Networks. *Journal of Medical Signals and Sensors*, 2(3), 169.
- Albawi, S., Mohammed, T. A. and Al-Zawi, S. 2017. Understanding of a Convolutional Neural Network. In *2017 International Conference on Engineering and Technology*, 1-6.
- Andersen, K. G., Rambaut, A., Lipkin, W. I., Holmes, E. C., and Garry, R. F. 2020. The Proximal Origin of SARS-CoV-2. *Nature Medicine*, 26(4), 450-452.
- Banik, P. P., Saha, R., and Kim, K. D. (2020). An Automatic Nucleus Segmentation and CNN Model Based Classification Method of White Blood Cell. *Expert Systems With Applications*, 149, 113211.
- Ben Atitallah, S., Driss, M., Boulila, W. and Ben Ghezala, H. 2022. Randomly Initialized Convolutional Neural Network for the Recognition of COVID-19 Using X-ray Images. *International Journal of Imaging Systems and Technology*, 32(1), 55-73.
- Bergstra, J. and Bengio, Y. 2012. Random Search for Hyper-Parameter Optimization. *Journal of Machine Learning Research*, 13(02), 281-305.
- Boser, B. E., Guyon, I. M. and Vapnik, V. N. 1992. A Training Algorithm for Optimal Margin Classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, 144-152.
- Byvatov, E. and Schneider, G. 2003. Support Vector Machine Applications in Bioinformatics. *Applied Bioinformatics*, 2(2), 67-77.
- Chaitanya, C. R. A., Kaplanyan, A. S., Schied, C., Salvi, M., Lefohn, A., Nowrouzezahrai, D. and Aila, T. 2017. Interactive Reconstruction of Monte Carlo Image Sequences Using a Recurrent Denoising Autoencoder. *ACM Transactions on Graphics (TOG)*, 36(4), 1-12.

- Che, Z. G., Chiang, T. A. and Che, Z. H. 2011. Feed-Forward Neural Networks Training: A Comparison Between Genetic Algorithm and Back-Propagation Learning Algorithm. *International Journal of Innovative Computing, Information and Control*, 7(10), 5839–5850.
- Chen, S., Kalantidis, Y., Li, L. J., Shamma, D. A., Bernstein, M. and Fei, L. 2016. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations.
- Cheng, Z., Sun, H., Takeuchi, M. and Katto, J. 2018. Deep Convolutional Autoencoder-Based Lossy Image Compression. In *2018 Picture Coding Symposium*, 253-257.
- Chollet, F. 2016. Building Autoencoders in Keras. *The Keras Blog*, 14.
- Cui, J., Li, F. and Shi, Z. L. 2019. Origin and Evolution of Pathogenic Coronaviruses. *Nature Reviews Microbiology*, 17(3), 181-192.
- De Moura, L. V., Mattjie, C., Dartora, C. M., Barros, R. C., and da Silva, A. M. M. 2021. Explainable Machine Learning for COVID-19 Pneumonia Classification With Texture-Based Features Extraction in Chest Radiography. *Frontiers in Digital Health*, 3.
- Duchi, J., Hazan, E. and Singer, Y. 2011. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research*, 12(7), 2121–2159.
- Erdoğan, V. and Metin, M. 2013. Parapnömonik Plevral Efüzyon ve Ampiyem. *Derleme*, 69-76.
- Fang, H., Gupta, S., Iandola, F., Srivastava, R. K., Deng, L., Doll'ar, P., Gao, J., He, X., Mitchell, M., Platt, J. C., Zitnick, L. and Zweig, G. 2015. From Captions to Visual Concepts and Back. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1473–1482.
- Gal, Y. and Ghahramani, Z. 2016. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. In *International Conference on Machine Learning*, 1050-1059.
- Glorot, X. and Bengio, Y. 2010. Understanding the Difficulty of Training Deep Feedforward Neural Networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, 249-256.
- Gomathi, K. and Priyaa, D. D. S. 2016. Multi Disease Prediction Using Data Mining Techniques. *International Journal of System and Software Engineering*, 4(2), 12-14.
- Gondara, L. 2016. Medical Image Denoising Using Convolutional Denoising Autoencoders. In *2016 IEEE 16th International Conference on Data Mining Workshops*, 241-246.

- Goodfellow, I., Bengio, Y. and Courville, A. 2016. Deep learning. Book in preparation for MIT Press.
- Gozes, O. and Greenspan, H. 2019. Deep Feature Learning From a Hospital-Scale Chest X-ray Dataset With Application to TB Detection on a Small-Scale Dataset. In 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 4076-4079.
- Gregor, K., Danihelka, I., Graves, A., Rezende, D. and Wierstra, D. 2015. Draw: A Recurrent Neural Network for Image Generation. In International Conference on Machine Learning, 1462-1471.
- Habibzadeh, M., Krzyżak, A. and Fevens, T. 2013. White Blood Cell Differential Counts Using Convolutional Neural Networks for Low Resolution Images. In International Conference on Artificial Intelligence and Soft Computing, 263-274. Springer, Berlin, Heidelberg.
- Hall, M. A. 1999. Correlation-based feature selection for machine learning.
- Hamidian J.A., Mazloom, S. and Ballard, D. 2020. What the European and American Health Care Systems Can Learn From China COVID-19 Epidemic; Action Planning Using Purpose Designed Medical Telecommunication, Courier Services, Home-Based Quarantine, and COVID-19 Walk-In Centers. *Immunopathol Persa*, 6(2), 17.
- He, K., Zhang, X., Ren, S. and Sun, J. 2015. Delving Deep Into Rectifiers: Surpassing Human-Level Performance on Imagenet Classification. In Proceedings of the IEEE International Conference on Computer Vision, 1026-1034.
- Henry, H. J. 1992. Adaptation in Natural and Artificial Systems: an Introductory Analysis With Applications to Biology, Control, and Artificial Intelligence.
- Hochreiter, S. and Schmidhuber, J. 1997. Long Short-Term Memory. *Neural Computation*, 9(8), 1735-1780.
- Hodosh, M., Young, P. and Hockenmaier, J. 2013. Framing Image Description As a Ranking Task: Data, Models and Evaluation Metrics. *Journal of Artificial Intelligence Research*, 47, 853-899.
- Horne, B. D., Anderson, J. L., John, J. M., Weaver, A., Bair, T. L., Jensen, K. R. and Muhlestein, J. B. 2005. Which White Blood Cell Subtypes Predict Increased Cardiovascular Risk?. *Journal of the American College of Cardiology*, 45(10), 1638-1643.
- Hsu, T. M. H., Weng, W. H., Boag, W., McDermott, M. and Szolovits, P. 2018. Unsupervised Multimodal Representation Learning Across Medical Images and Reports. arXiv preprint arXiv, 1811.08615.

- Huang, C., Wang, Y., Li, X., Ren, L., Zhao, J., Hu, Y. and Cao, B. 2020. Clinical Features of Patients Infected With 2019 Novel Coronavirus in Wuhan, China. *The Lancet*, 395(10223), 497-506.
- Islam, M. Z., Islam, M. M. and Asraf, A. 2020. A Combined Deep CNN-LSTM Network for the Detection of Novel Coronavirus (COVID-19) Using X-Ray Images. *Informatics in Medicine Unlocked*, 20, 10041.
- J., Lee, J. and Lee, Y. 2021. A Transfer Learning Architecture Based on a Support Vector Machine for Histopathology Image Classification. *Applied Sciences*, 11(14), 6380.
- Joachims, T. 1999. Svmight: Support Vector Machine. SVM-Light Support Vector Machine <http://svmlight.joachims.org/>, University of Dortmund, 19(4), 25.
- Joshi, M. D., Karode, A. H. and Suralkar, S. R. 2013. White Blood Cells Segmentation and Classification to Detect Acute Leukemia. *International Journal of Emerging Trends & Technology in Computer Science*, 2(3), 147-151.
- Kaiming, H., Xiangyu, Z., Shaoqing, R. and Jian, S. 2014. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. In *Proceedings of the European Conference on Computer Vision*, 346–361.
- Kansara, D., Sompura, S., Momin, S. and D'silva, M. 2018. Classification of WBC for Blood Cancer Diagnosis using Deep Convolutional Neural Networks. *International Journal of Research in Advent Technology*, 6, 12.
- Karim, A. M., Güzel, M. S., Tolun, M. R., Kaya, H. and Çelebi, F. V. 2019. A New Framework Using Deep Auto-Encoder and Energy Spectral Density for Medical Waveform Data Classification and Processing. *Biocybernetics and Biomedical Engineering*, 39(1), 148-159.
- Karpathy, A., Johnson, J. and Fei-Fei, L. 2015. Visualizing and Understanding Recurrent Networks. *arXiv preprint arXiv:1506.02078*.
- Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R. and Fei-Fei, L. 2014. Large-Scale Video Classification With Convolutional Neural Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1725-1732.
- Kattan, M., Abdullah, R. and Salam, R. A. 2010. Training Feed-Forward Neural Networks Using a Parallel Genetic Algorithm With the Best Must Survive Strategy. In *2010 International Conference on Intelligent Systems, Modelling and Simulation*, 96–99.
- Khozeimeh, F., Sharifrazi, D., Izadi, N. H., Joloudari, J. H., Shoeibi, A., Alizadehsani, R. and Islam, S. M. S. 2021. Combining a Convolutional Neural Network With Autoencoders to Predict the Survival Chance of COVID-19 Patients. *Scientific Reports*, 11(1), 1-18.

- Kickler, T.S. 1999. Clinical Analyzers. *Advances in Automated Cell Counting*. Anal Chem, 71, 363-365.
- Kim, H. C., Pang, S., Je, H. M., Kim, D. and Bang, S. Y. 2003. Constructing support Vector Machine Ensemble. *Pattern Recognition*, 36(12), 2757-2767.
- Kingma, D. P. and Ba, J. 2014. Adam: A Method for Stochastic Optimization. Amsterdam Machine Learning Lab. In *Proceedings of International Conference on Learning Representations*, <https://doi.org/10.48550/arXiv.1412.6980>.
- Kiros, R., Salakhutdinov, R. and Zemel, R.S. 2015. Unifying Visual-Semantic Embeddings With Multimodal Neural Language Models. *Transactions of the Association for Computational Linguistics*.
- Kong, C., Lin, D., Bansal, M., Urtasun, R., & Fidler, S. (2014). What are you talking about? text-to-image coreference. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3558-3565).
- Krishna, R. A., Hata, K., Chen, S., Kravitz, J., Shamma, D. A., Fei-Fei, L., & Bernstein, M. S. (2016, May). Embracing error to enable rapid crowdsourcing. In *Proceedings of the 2016 CHI conference on human factors in computing systems* (pp. 3167-3179).
- Krizhevsky, A., Sutskever, I. and Hinton, G. E. 2012. Imagenet Classification With Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems*, 25.
- Kuan, K., Ravaut, M., Manek, G., Chen, H., Lin, J., Nazir, B. and Chandrasekhar, V. 2017. Deep Learning for Lung Cancer Detection: Tackling the Kaggle Data Science Bowl 2017 Challenge. *arXiv preprint arXiv, 1705.09435*.
- Lakhani, P. and Sundaram, B. 2017. Deep Learning at Chest Radiography: Automated Classification of Pulmonary Tuberculosis by Using Convolutional Neural Networks. *Radiology*, 284(2), 574-582.
- LeCun, Y., Kanter, I., & Solla, S. (1990). Second order properties of error surfaces: Learning time and generalization. *Advances in neural information processing systems*, 3.
- LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278-2324.
- LeCun, Y., Bengio, Y. and Hinton, G. 2015. Deep Learning. *Nature*, 521(7553), 436.
- Li, Q., Guan, X., Wu, P., Wang, X., Zhou, L., Tong, Y. and Feng, Z. (2020). Early Transmission Dynamics in Wuhan, China, of Novel Coronavirus–Infected Pneumonia. *New England Journal of Medicine*, 382, 1199-1207.

- Liang, G., Hong, H., Xie, W. and Zheng, L. 2018. Combining Convolutional Neural Network With Recursive Neural Network for Blood Cell Image Classification. *IEEE Access*, 6, 36188-36197.
- Lin, D., Fidler, S., Kong, C. and Urtasun, R. 2014. Visual Semantic Search: Retrieving Videos Via Complex Textual Queries. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2657-2664.
- Liu, C. T., Lin, T. W., Wu, Y. H., Lin, Y. S., Lee, H., Tsao, Y. and Chien, S. Y. 2018. Computation-Performance Optimization of Convolutional Neural Networks With Redundant Filter Removal. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 66(5), 1908-1921.
- Lopez-Garnier, S., Sheen, P. and Zimic, M. 2019. Automatic Diagnostics of Tuberculosis Using Convolutional Neural Networks Analysis of MODS Digital Images. *PloS one*, 14(2), e0212094.
- Loudon, J. S. 2018. Detecting and Localizing Cell Nuclei in Medical Images Master's Thesis, Norwegian University of science and Technology.
- Lu, R., Zhao, X., Li, J., Niu, P., Yang, B., Wu, H. and Tan, W. 2020. Genomic Characterisation and Epidemiology of 2019 Novel Coronavirus: Implications for Virus Origins and Receptor Binding. *The Lancet*, 395(10224), 565-574.
- Mao, J., Xu, W., Yang, Y., Wang, J. and Yuille, A. L. 2014. Explain Images With Multimodal Recurrent Neural Networks. *arXiv preprint arXiv:1410.1090*.
- McCarthy, J. (1956). Measures of the value of information. *Proceedings of the National Academy of Sciences*, 42(9), 654-655.
- Meng, Q., Catchpoole, D., Skillicom, D. and Kennedy, P. J. 2017. Relational Autoencoder for Feature Extraction. In *2017 International Joint Conference on Neural Networks*, 364-371.
- Meyer, D., Leisch, F. and Hornik, K. 2003. The Support Vector Machine Under Test. *Neurocomputing*, 55(1-2), 169-186.
- Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S. and Dean, J. 2013. Distributed Representations of Words and Phrases and Their Compositionality. In *Advances in Neural Information Processing Systems*.
- Miller, G. F., Todd, P. M. and Hegde, S. U. 1989. Designing Neural Networks Using Genetic Algorithms. In *ICGA*, 89, 379-384.
- Montana, D. J. 1995. Neural Network Weight Selection Using Genetic Algorithms. *Intelligent Hybrid Systems*, 8(6), 12-19.
- Mooney, P. 2018. Identify Blood Cell Subtypes From Images, Kaggle, Web Sites: <https://www.kaggle.com/paultimothymooney/identify-blood-cell-subtypes-from-images> Erişim Tarihi: 09.10.2018.

- Noble, W. S. 2006. What is a Support Vector Machine?. *Nature Biotechnology*, 24(12), 1565-1567.
- Ohlsson, S. 2011. *Deep Learning: How the Mind Overrides Experience*. Cambridge University Press.
- Omid, E. D. and Iddo, G. 2014. Genetic Algorithms for Evolving Deep Neural Networks. In *Proceedings of the Companion Publication of the 2014 Annual Conference on Genetic and Evolutionary Computation*, 1451–1452.
- Oullette, R., Browne, M. and Hirasawa, K. 2004. Genetic Algorithm Optimization of a Convolutional Neural Network for Autonomous Crack Detection. In *Proceedings of the 2004 Congress on Evolutionary Computation*. IEEE Cat. No. 04TH8753, 1, 516–521.
- Palmes, P. P., Hayasaka, T., & Usui, S. (2005). Mutation-based genetic neural network. *IEEE Transactions on Neural Networks*, 16(3), 587-600.
- Pasa, F., Golkov, V., Pfeiffer, F., Cremers, D. and Pfeiffer, D. 2019. Efficient Deep Network Architectures for Fast Chest X-ray Tuberculosis Screening and Visualization. *Scientific Reports*, 9(1), 6268.
- Pascanu, R., Mikolov, T. and Bengio, Y. 2013. On the Difficulty of Training Recurrent Neural Networks. In *International Conference on Machine Learning*, 1310-1318.
- Petmezas, G., Haris, K., Stefanopoulos, L., Kilintzis, V., Tzavelis, A., Rogers, J. A. and Maglaveras, N. 2021. Automated Atrial Fibrillation Detection Using a Hybrid CNN-LSTM Network on Imbalanced ECG Datasets. *Biomedical Signal Processing and Control*, 63, 102194.
- Pisner, D. A. and Schnyer, D. M. (2020). Support Vector Machine. In *Machine Learning*, 101-121. Academic Press.
- Pradhan, A. 2012. Support Vector Machine-A Survey. *International Journal of Emerging Technology and Advanced Engineering*, 2(8), 82-85.
- Pu, Y., Gan, Z., Henao, R., Yuan, X., Li, C., Stevens, A. and Carin, L. 2016. Variational Autoencoder for Deep Learning of Images, Labels and Captions. *Advances in Neural Information Processing Systems*, 29, 2352-2360.
- Quant Dare, 2019. Web Sites: https://quantdare.com/what-is-the-difference-between-deep-learning-and-machine-learning/deep_learning/ Erişim Tarihi: 15.04.2022.
- Rabi, F. A., Al Zoubi, M. S., Kasasbeh, G. A., Salameh, D. M. and Al-Nasser, A. D. 2020. SARS-CoV-2 and Coronavirus Disease 2019: What we Know so Far. *Pathogens*, 9(3), 231.
- Rosenblatt, F. 1958. The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain. *Psychological Review*, 65(6), 386.

- Rumelhart, D. E., Hinton, G. E. and Williams, R. J. 1986. Learning representations by back-propagating errors. *Nature*, 323(6088), 533-536.
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S. and Fei-Fei, L. 2015. Imagenet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3), 211-252.
- Salehi, S., Abedi, A., Balakrishnan, S. and Gholamrezanezhad, A. 2020. Coronavirus Disease 2019 (COVID-19): A Systematic Review of Imaging Findings in 919 Patients. *Ajr Am Journal Roentgenol*, 215(1), 87-93.
- Sathitratanacheewin, S. and Pongpirul, K. 2018. Deep Learning for Automated Classification of Tuberculosis-Related Chest X-Ray: Dataset Specificity Limits Diagnostic Performance Generalizability. *arXiv preprint arXiv:1811.07985*.
- Sayad, A.T. and Halkarnikar, P.P. 2014. Diagnosis of heart disease using neural network approach. *Int Journal Adv Sci Eng Technol*, 2, 88–92.
- Schmidhuber, J. 2015. Deep Learning in Neural Networks: An Overview. *Neural Networks*, 61, 85-117.
- Sermanet P, Eigen D, Zhang X, Mathieu M, Fergus R, and Le Cun. Y. 2014. Overfeat: Integrated Recognition, Localization and Detection Using Convolutional Networks. In *Proceedings of the International Conference on Learning Representations*.
- Shankar, V., Kumar, V., Devagade, U., Karanth, V. and Rohitaksha, K. 2020. Heart Disease Prediction Using CNN Algorithm. *SN Computer Science*, 1(3), 1-8.
- Sharma, S. and Mehra, R. 2020) Conventional Machine Learning and Deep Learning Approach for Multi-Classification of Breast Cancer Histopathology Images—a Comparative Insight. *Journal of Digital Imaging*, 33(3), 632-654.
- Sharon, N. and Therse, P. 2018. White Blood Cell Differential Counting in Blood Smears via Tiny yolo. *CS230: Deep Learning*, Spring, Stanford University, CA. 1-6.
- Shrock, E., Fujimura, E., Kula, T., Timms, R. T., Lee, I. H., Leng, Y. and Elledge, S. J. 2020. Viral Epitope Profiling of COVID-19 Patients Reveals Cross-Reactivity and Correlates of Severity. *Science*, 370(6520), eabd4250.
- Simon, M., Rodner, E. and Denzler, J. 2016. Imagenet Pre-Trained Models With Batch Normalization. *arXiv preprint arXiv:1612.01452*.
- Simonyan, K., & Zisserman, A. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv preprint arXiv:1409.1556*.
- Singh, S. S. B. and Granitzer, J. M. P. D. M. 2021. Comparison of Transfer Learning Models for Legal Document Classification.

- Sivaramakrishnan, R., Antani, S., Candemir, S., Xue, Z., Abuya, J., Kohli, M. and Thoma, G. 2018. Comparing Deep Learning Models for Population Screening Using Chest Radiography. In Medical Imaging 2018: Computer-Aided Diagnosis, 10575, 105751E. International Society for Optics and Photonics.
- Sivari, E., Güzel, M. S., Bostanci, E. and Mishra, A. 2022. A Novel Hybrid Machine Learning Based System to Classify Shoulder Implant Manufacturers. In Healthcare, 10(3), 580.
- Socher, R., Pennington, J., Huang, E. H., Ng, A. Y., & Manning, C. D. (2011, July). Semi-supervised recursive autoencoders for predicting sentiment distributions. In Proceedings of the 2011 conference on empirical methods in natural language processing (pp. 151-161).
- Socher, R., Karpathy, A., Le, Q. V., Manning, C. D. and Ng, A. Y. 2014. Grounded Compositional Semantics for Finding and Describing Images With Sentences. Transactions of the Association for Computational Linguistics, 2, 207-218.
- Stirenko, S., Kochura, Y., Alienin, O., Rokovyi, O., Gordienko, Y., Gang, P. and Zeng, W. 2018. Chest X-Ray Analysis of Tuberculosis by Deep Learning With Segmentation and Augmentation. In 2018 IEEE 38th International Conference on Electronics and Nanotechnology, 422-428.
- Su, Y., Li, D. and Chen, X. 2021. Lung Nodule Detection Based on Faster R-CNN Framework. Computer Methods and Programs in Biomedicine, 200, 105866.
- Sun, Y., Xue, B., Zhang, M. and Yen, G.G. 2019. Evolving Deep Convolutional Neural Networks for Image Classification. IEEE Transactions on Evolutionary Computation.
- Suthaharan, S. 2016. Support Vector Machine. In Machine Learning Models and Algorithms for Big Data Classification, 207-235. Springer, Boston, MA.
- Sutskever, I., Martens, J. and Hinton, G. E. 2011. Generating Text With Recurrent Neural Networks. In ICML.
- Szegedy, C., Ioffe, S., Vanhoucke, V. and Alemi, A. A. 2017. Inception-v4, Inception-Resnet and the Impact of Residual Connections on Learning. In AAAI, 4, 12.
- Tieleman, T. and Hinton, G. 2012. Lecture 6.5-Rmsprop: Divide the Gradient by a Running Average of its Recent Magnitude. COURSE: Neural Networks for Machine Learning, 4(2), 26-31.
- Tschannen, M., Bachem, O. and Lucic, M. 2018. Recent Advances in Autoencoder-Based Representation Learning. arXiv preprint arXiv:1812.05069.
- Turing, Alan Mathison. "Mind." Mind 59, no. 236 (1950): 433-460.

- Ucar, F. and Korkmaz, D. 2020. COVIDiagnosis-Net: Deep Bayes-SqueezeNet Based Diagnosis of the Coronavirus Disease 2019 (COVID-19) From X-ray Images. *Medical Hypotheses*, 140, 109761.
- Ukil, A. 2007. Support Vector Machine. In *Intelligent Systems and Signal Processing in Power Engineering*, 161-226. Springer, Berlin, Heidelberg.
- Van Laarhoven, T. 2017. L2 Regularization Versus Batch and Weight Normalization. *arXiv preprint arXiv:1706.05350*.
- Vedantam, R., Lawrence Zitnick, C. and Parikh, D. 2015. Cider: Consensus-Based Image Description Evaluation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4566-4575.
- Velavan, T. P. and Meyer, C. G. 2020. The COVID-19 Epidemic. *Tropical Medicine & International Health*, 25(3), 278.
- Wang, J., Zhang, J., Bao, W., Zhu, X., Cao, B. and Yu, P. S. 2018. Not Just Privacy: Improving Performance of Private Deep Learning in Mobile Cloud. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2407-2416.
- Wang, W., Huang, Y., Wang, Y. and Wang, L. 2014. Generalized Autoencoder: A Neural Network Framework for Dimensionality Reduction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshop*, 490-497.
- Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M. and Summers, R. M. 2017. Chestx-Ray8: Hospital-Scale Chest X-Ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2097-2106.
- Waring, J., Lindvall, C. and Umeton, R. 2020. Automated Machine Learning: Review of the State-of-the-art and Opportunities for Healthcare. *Artificial Intelligence in Medicine*, 104, 101822.
- Weiss, K., Khoshgoftaar, T. M. and Wang, D. 2016. A Survey of Transfer Learning. *Journal of Big Data*, 3(1), 1.
- Widodo, A. and Yang, B. S. 2007. Support Vector Machine in Machine Condition Monitoring and Fault Diagnosis. *Mechanical Systems and Signal Processing*, 21(6), 2560-2574.
- Wu, Y., Burda, Y., Salakhutdinov, R., & Grosse, R. (2016). On the quantitative analysis of decoder-based generative models. *arXiv preprint arXiv:1611.04273*.
- Zeiler, M. D. and Fergus, R. 2014. Visualizing and Understanding Convolutional Networks. In *European Conference on Computer Vision*, 818-833. Springer, Cham.

- Zhang, C. and Woodland, P. C. 2015. Parameterised Sigmoid and ReLU Hidden Activation Functions for DNN Acoustic Modelling. In Sixteenth Annual Conference of the International Speech Communication Association.
- Zhang, G., Liu, Y. and Jin, X. 2020. A Survey of Autoencoder-Based Recommender Systems. *Frontiers of Computer Science*, 14(2), 430-450.
- Zhao, J., Zhang, M., Zhou, Z., Chu, J. and Cao, F. 2017. Automatic Detection and Classification of Leukocytes Using Convolutional Neural Networks. *Medical & Biological Engineering & Computing*, 55(8), 1287-1301.

