T.C.
İSTANBUL BEYKENT ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
BİLGİSAYAR MÜHENDİSLİĞİ BİLİM DALI

MAKİNE ÖĞRENMESİ YÖNTEMLERİ İLE DİYABET HASTALIĞI TAHMİNLEME

Yüksek Lisans Tezi

Tezi Hazırlayan

Tuğba KEŞ

İstanbul, 2023

T.C.
İSTANBUL BEYKENT ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
BİLGİSAYAR MÜHENDİSLİĞİ BİLİM DALI

MAKİNE ÖĞRENMESİ YÖNTEMLERİ İLE DİYABET HASTALIĞI TAHMİNLEME

Yüksek Lisans Tezi

Tezi Hazırlayan

Tuğba KEŞ

Öğrenci No.

2020003077

Orcid No:

0009-0008-4252-5047

Danışman:

Dr. Öğr. Üyesi Talat FİRLAR

İstanbul, 2023

YEMİN METNİ

Yüksek Lisans Tezi olarak hazırladığım "**Makine Öğrenmesi Yöntemleri ile Diyabet Hastalığı Tahminleme**" başlıklı bu çalışmanın, bilimsel ahlak ve geleneklere uygun şekilde tarafımdan yazıldığını, bu tezdeki bütün bilgileri akademik ve etik kurallar içinde elde ettiğimi, yararlandığım eserlerin tamamının kaynaklarda gösterildiğini ve çalışmamın içinde kullanıldıkları her yerde bunlara atıf yapıldığını, patent ve telif haklarını ihlal edici bir davranışımın olmadığını belirtir ve bunu onurumla doğrularım. 03.05.2023

Tuğba KEŞ

T.C.
İSTANBUL BEYKENT ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ MÜDÜRLÜĞÜ
TEZLİ YÜKSEK LİSANS SINAV TUTANAĞI

13.../5.../...2023

Enstitümüz ***Bilgisayar Mühendisliği*** Anabilim Dalı ***Bilgisayar Mühendisliği*** Programı yüksek lisans öğrencilerinden ***2020003077*** numaralı ***Tuğba KEŞ***'in *"Beykent Üniversitesi Lisansüstü Eğitim – Öğretim Yönetmeliği"*nin ilgili maddesine göre hazırlayarak, Enstitümüze teslim ettiği ***"Makine Öğrenmesi Yöntemleri ile Diyabet Hastalığı Tahminleme"*** konulu tezini, Yönetim Kurulumuzun 11/04/2023 tarih ve 2023/15 sayılı toplantısında seçilen ve On-Line toplanan biz jüri üyeleri huzurunda, Beykent Üniversitesi Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 29. maddesinin 3. fıkrası gereğince 45 dakika süre ile Zoom programı aracılığıyla on-line olarak aday tarafından savunulmuş ve sonuçta adayın tezi hakkında ***"OYBİRLİĞİ"*** ile ***"KABUL"*** kararı verilmiştir.

İşbu tutanak, 2 nüsha olarak hazırlanmış ve Enstitü Müdürlüğü'ne sunulmak üzere tarafımızdan düzenlenmiştir.

DANIŞMAN
Dr. Öğr. Üyesi Ta*** Fİ***
(İstanbul Beykent Üniversitesi)

ÜYE
Dr. Öğr. Üyesi Ke*** Gö*** NA***
(İstanbul Beykent Üniversitesi)

ÜYE
Dr. Öğr. Üyesi Ke*** ŞA***
(İstanbul Medipol Üniversitesi)

Adı ve Soyadı	:	Tuğba KEŞ
Danışmanı	:	Dr. Öğr. Üyesi Talat FİRLAR
Türü ve Tarihi	:	Yüksek Lisans Tezi, 2023
Alanı	:	Bilgisayar Mühendisliği
Anahtar Kelimeler	:	Yapay Zeka, Lojistik Regresyon, Rasgele Orman, Naive Bayes, K-means, Karar ağacı, Python, Diyabet

ÖZ

MAKİNE ÖĞNRENMESİ YÖNTEMLERİ İLE DİYABET HASTALIĞI TAHMİNLEME

Günümüzde pek çok alanda yaygın olarak kullanılan yapay zeka, insanların karmaşık sorunları daha kolay ve hızlı çözümlemelerine yardımcı oluyor. Yapay zeka, insan zekasını taklit ederek, topladığı veriler aracılığıyla öğrenip, kendini geliştirebiliyor ve yenilik yapabiliyor. Sağlıkta alanında da yaygın olarak kullanılan makine öğrenimi teknikleri, erken teşhis ve yerinde teşhis sağlayarak, kolaylıklar sunmaktadır.

Bu çalışma kapsamında, Spyder aracı üzerinde Python programlama dili kullanılarak, Logistic Regression, XGBoost, Naive Bayes, En Yakın Komşu, Destek Vektör Makineleri, Karar Ağacı gibi yaygın makine öğrenmesi teknikleri Diyabet verisine uygulanmış ve şeker hastalığı tespiti yapılmıştır. Diyabet hastalığı tespiti için makine öğrenimi yöntemleri uygulanıp, karşılaştırılarak, erken teşhis ve gelecekte teşhis öngörüsü için en etkili yöntemler belirlenmiştir.

Bu çalışmanın amacı, toplam 768 hastadan toplanan verilere yapay zeka algoritmalarını uygulayarak, bu tekniklerden elde edilen sonuçlar kıyaslanmış, diyabet teşhisini en yüksek doğruluk oranında sağlayan algoritmanın bulunmasını kapsamaktadır. Bu tezde kullanılan yapay zeka algoritmaları Lojistik Regresyon, KNN, XGBoost, Naive Bayes, Rastgele Orman, Destek Vektör Makineleri, Karar Ağacı'dır. Algoritmalardan elde edilen sonuçlar kıyaslandığında en yüksek doğruluk oranı %96 ile Rastgele Orman ve KNN algoritması ile elde edilmiştir.

Name and Surname	:	Tuğba KEŞ
Supervisor	:	Dr. Faculty Member Talat FIRLAR
Degree and Date	:	Master's Thesis, 2023
Major	:	Computer Engineering
Key Words	:	Artificial Intelligence, Logistic Regression, Random Forest, Naive Bayes, K-means, Decision Tree, Python, Diabetes

ABSTRACT

DIABETES PREDICTION WITH MACHINE LEARNING METHODS

Artificial intelligence, which is widely used in many fields today, helps people to solve complex problems more easily and quickly. Artificial intelligence can learn, improve and innovate through the data it collects by imitating human intelligence. Machine learning techniques, which are also widely used in the field of health, provide convenience by providing early diagnosis and on-site diagnosis.

Within the scope of this study, common machine learning techniques such as Logistic Regression, XGBoost, Naive Bayes, Nearest Neighbor, Support Vector Machines, Decision Tree were applied to Diabetes data by using Python programming language on the Spyder tool and diabetes detection was made. By applying and comparing machine learning methods for the detection of diabetes, the most effective methods for early diagnosis and prediction of future diagnosis are determined.

The aim of this study is to find the algorithm that provides the highest accuracy rate for diabetes diagnosis by applying artificial intelligence algorithms to the data collected from a total of 768 patients, the results obtained from these techniques are compared. Artificial intelligence algorithms used in this thesis are Logistic Regression, KNN, XGBoost, Naive Bayes, Random Forest, Support Vector Machines, Decision Tree. When the results obtained from the algorithms are compared, the highest accuracy rate was obtained by the Random Forest and KNN algorithm with 96%.

İÇİNDEKİLER

Sayfa No.

DÖRDÜNCÜ BÖLÜM

UYGULAMA

TABLOLAR LİSTESİ

Sayfa No.

ŞEKİLLER LİSTESİ

Sayfa No.

KISALTMALAR

ADA	:	Amerikan Diyabet Derneği
AI	:	Yapay Zeka
ANN	:	Yapay Sinir Ağı
CNN	:	Convolutional Neural Network
DKA	:	Diyabetik Ketoasidoz
DL	:	Deep Learning
DÖ	:	Derin Öğrenme
DVM	:	Destek Vektör Makineleri
ELM	:	Aşırı Öğrenme Makinesi
ESA	:	Evrişimsel Sinir Ağı
FGL	:	Açlık Kan Şekeri
GBT	:	Boosted Trees
IDDM	:	İnsüline Bağımlı Diabetes Mellitus
IDF	:	Uluslararası Diyabet Federasyonu
ILPD	:	Indian Liver Patient Dataset
KA	:	Karar Ağacı
KAH	:	Koroner Arter Hastalığı
KNN	:	En Yakın Komşu Algoritması
LM	:	Levenberg-Marquardt Algoritması
LMT	:	Lojistik Model Ağacı
LR	:	Lojistik Regresyon
ML	:	Makine Öğrenimi
MLlib	:	Makine Öğrenimi Kütüphanesi
MLPC	:	Çok Katmanlı Algılayıcıyı
MÖ	:	Makine Öğrenmesi
NB	:	Naive Bayes
OGTT	:	Oral Glikoz Tolerans Testi
PD	:	Parkinson’s Disease
PH	:	Parkinson Hastalığı
PSO	:	Parçacık Sürü Optimizasyonu
REP Tree	:	Karar Ağacı

RO	:	Rastgele Orman
RS	:	Kaba Küme
UCI	:	Makine Öğrenmesi Deposu
YSA	:	Yapay Sinir Ağları

SÖZLÜK

Algoritma: Belirli bir sorun nasıl çözülür veya belirli bir hedefe nasıl ulaşılır. Matematik ve bilgisayar bilimlerinde, bir görevi yerine getirmek için tanımlanan sonlu bir dizi işlem. Bir başlangıç durumu ile başlar ve iyi tanımlanmış bir son durum ile biter.

Anti-diyabetik: Diyabet tedavisinde kullanılan ilaçlardır.

Diabetes Mellitus: Diabetes mellitus, yaygın olarak diabetes mellitus olarak anılır, kandaki glikoz (şeker) seviyesinin normalin üzerinde yükselmesidir ve bu da normalde şeker içermemesi gereken idrarda şeker bulunmasına neden olur.

Genetik: Genetik veya genetik, canlılardaki kalıtımı ve genetik çeşitliliği inceleyen biyoloji dalıdır.

Gestasyonel Diyabet: İlk olarak hamilelik sırasında ortaya çıkan ve doğumdan sonra düzelen glukoz intoleransı, gestasyonel diyabet olarak tanımlanır. Gebelik; bebeğin beslendiği plasentanın salgıladığı hormonların insülin direnci ile karakterize olduğu bir dönemdir.

Glisemik İndeks: Glisemik indeks (GI), farklı gıdaların kan şekerini yükseltme eğrilerini standardize etmek için kullanılan bir terimdir. Glisemik indeks 0 ile 100 orasında bir skalaya sahiptir. Değer 100'e yaklaştıkça, o besinin kan şekerini o derece hızla yükseltebileceği anlamına gelir. Saf glukoz referans nokta olarak belirlenmiştir ve glisemik indeks 100 olarak kabul edilmiştir. Diğer gıdaların glisemik indeksleri aşağıya doğru sıralanır.

Hiperglisemi: Kan şekerinizin yüksek olduğu anlamına gelir. Bu, en az 8 saatlik açlıktan sonra ölçülen 100 mg/dl'den yüksek veya şekerli bir içeceğin alınmasından 2 saat sonra ölçülen 140 mg/dl'den yüksek bir kan şekeri düzeyinin belirlenmesidir.

İnsülin: İnsülin, moleküler ağırlığı 5,8 kilodalton olan polipeptit yapılı bir hormondur ve glukagon ile birlikte vücuttaki karbonhidrat emiliminin düzenlenmesinde rol oynar. hipoglisemik etkiye sahiptir

İterasyon: Tekrar, tekrar, tekrar, tekrar, sıralı işlem anlamına gelir, programlamada döngüler kullanan bir işlem dizisinin açıklamasıdır.

Kardiyovasküler: Kardiyovasküler hastalık, kalp veya kan damarları (arterler ve damarlar) hastalıklarını içeren grup için ortak bir terminolojidir. Kardiyovasküler hastalık, dolaşım sistemini etkileyen tüm hastalıkları içerir.

Kolesterol: Kolesterol, hayvan vücut dokularının hücre zarlarında bulunan ve plazmada taşınan bir steroldür.

Lojistik Regresyon: Sonucu belirlemek için bir veya daha fazla bağımsız değişken kullanarak bir veri setini analiz eden istatistiksel bir tekniktir.

Makine Öğrenimi: Bu, açıkça programlanmaya gerek kalmadan yazılım programlarının sonuçları tahmin etmede daha doğru olmasını sağlayan bir algoritma kategorisidir.

Pre-Diyabetik: Bir kişi, kan şekeri seviyeleri normalden yüksek ancak diyabet teşhisi koyacak kadar yüksek olmadığında prediyabetik olarak tanımlanır. Bir diyabet önleme programına katılan prediyabetiklerde diyabet gelişmiştir.

Yapay Zeka: Aşağıdakiler gibi birçok özelliğe sahip bir dizi yazılım ve donanım sistemidir: İnsan davranışı, sayı duyusu, hareket, dil ve ses algısı.

GİRİŞ

Bir makinenin belirli bir deneyime sahip olması için, bir dizi veriye ihtiyacı vardır. Veri kümelerindeki deneyim ve deneyimlerden yola çıkarak makineye aktarılmalıdır. Veri olmadan, yapay zeka ve makine öğrenimi yöntemleri neredeyse düşünülemez. Yapay zeka, doğadaki her türlü rasyonel davranışı modellemeyi amaçlar. Yapay zeka, problem çözmede insan zekasını taklit etmeyi ve insan beyninin karmaşık yapısını makinelere uyarlamayı hedefler. Makine öğrenimi birçok alanda ve endüstride kullanılabilir.

Diabet veya diabetes mellitus (diabetes mellitus), yüksek kan şekeri ile ilişkili bir hastalıktır. Bunun nedeni pankreasın vücut için yeterli insülin meydana getirememesi veya etkili bir şekilde kullanamamasıdır.

Diyabetik ve diyabetik olmayan popülasyonlardan elde edilen vücut kitle indeksi, yaş, glikoz vb. değerli verilere makine öğrenimi tekniklerinin uygulanmasıyla erken tespit sağlanır. Makine öğrenimi ile ilgili kavramlar açıklanır ve diyabet tespiti için kullanılır. Çalışmanın bu bölümünde çalışma hakkında genel bilgiler verilmektedir. Bölüm 1'de bu konuda literatürdeki benzer çalışmalardan bahsedilmektedir. Bölüm 2, diyabet ve semptomları hakkında genel bilgiler içermektedir. Bölüm 3, makine öğrenimi hakkında genel bilgiler içerir. Bölüm 4, makine öğrenimi tekniklerini kullanan sonuçları sunar. Bu çalışmanın amacı; lojistik regresyon, naive bayes ve karar ağaçları gibi belirli makine öğrenmesi tekniklerinin Python programlarında test edilmesi ve başarılı sonuçlarının gösterilmesidir. Veri kümeleri, tıbbi bilgiler ve Python yazılımının yetenekleri kullanılarak, sistem için genel başarı oranı %96 ile en iyi Rastgele Orman ve KNN algoritması ile elde edilmiştir.

Bu çalışmada 768 veri içeren Kaggle veri seti kullanılmıştır. Şeker hastalığının olup olmadığını belirlemek için veri kümesine makine öğrenimi teknikleri uygulanarak sonuçlar elde edilmiştir. Kan basıncı, yaş, diyabet durumu gibi bireye özgü çeşitli değerleri barındırır.

Bu çalışmada diyabeti olan kişiler 1, diyabeti olmayanlar 0 olarak Boolean veri tipi ile etiketlenmiştir.

Diyabet saptanması için 768 kişi için öznitelik bilgisi kullanıldı, özelliklere makine öğrenmesi algoritmasını uygulandı, algoritmaların sonuçları karşılaştırıldı ve verilen algoritma ile diyabet teşhisi yapılmaya çalışılmıştır.

BİRİNCİ BÖLÜM
LİTARATÜR TARAMASI

1. LİTERATÜR TARAMASI

Bu tez için literatür gözden geçirildiğinde, TIP alanında araştırmaların çeşitli metotlar ve türlü uygulamalar kullanılarak yürütüldüğü tespit edilmiştir. Bu tezde özellikle göze çarpan durum, böylesine ciddi bir hastalığın erken teşhisinde hangi yöntemin en iyi ve en doğru sonuçları verdiğini belirlemek için farklı makine öğrenmesi yöntemleri kullanılması ve karşılaştırılmasıdır. En iyi sonuç Rastgele Orman algoritması kullanılarak elde edilmiştir.

Meme kanserini tahmin etmek için üç makine öğrenmesi yöntemini karşılaştırmışlardır. Büyük bir veri seti kullanarak tahmin modellerini ilerletmek için Lojistik Regresyon (LR- Logistic Regression), YSA ve Karar Ağaçları (KA- Decision Trees) kullanmışlardır. KA %93,6 doğrulukla en iyi neticeye ulaştığını, YSA'nın %91,2 doğrulukla ikinci sırada olduğunu ve LR modelleri ile %89,2 doğruluk oranıyla üç sınıflandırma içinde en düşük netice aldığını bildirmişlerdir. (Delen ve ark., 2005: 113-127)

Levenberg-Marquardt (LM) algoritması ile eğitilmiş çok katmanlı ve olasılıksal bir sinir ağı yapısı kullanarak şeker hastalığı tanısı üzerine karşılaştırmalı bir çalışma yapmışlardır. Yaptıkları çalışmada aynı veri tabanını kullanan önceki çalışmalarla karşılaştırdıklarında Pima Hint şeker hastalığı tanı problemi için iyi neticeler verebileceğini ve Sinir ağı yapılarının şeker hastalığının tanısına yardımcı olmak için başarıyla kullanılabileceği belirtmişlerdir. (Temurtas ve ark., 2009: 8610-8615)

Meme kanseri tanısı ile ilgili bir çalışma yapmışlardır. Yapılan çalışmada veri seti üzerinden çoklu sınıflandırma yapmışlardır. RO sınıflandırma algoritması, veri kümelerini sınıflandırmada %100 doğruluk sonucuna ulaşmışlardır. Meme dokusundan veriler elde edilerek, gelecekte kanser olma riskini tahmin edilebileceğini belirtmişlerdir. (Jacob ve Ramani, 2011: 46-53)

Kalp kapak hastalığının akıllı tanısı için, valvüler kalp hastalığı sınıflandırması bağlamında topluluk metodolojisi (torbalama, güçlendirme ve rastgele alt uzayın) ile karşılaştırmalı bir çalışma gerçekleştirmiştir. DVM'lerin toplulukları ve tek bir DVM sınıflandırıcısı ile karşılaştırılmıştır. Kalp kapak hastalığının tanısı için DVM'nin toplu metotlarının kullanılmasının daha etkili olduğunu belirtmiştir. (Sengur, 2012: 2649-2655)

Hepatit hastalığının tanısı için Kaba Küme (RS- Rough Set) ve aşırı öğrenme makinesine (ELM- Extreme Learning Machines) dayalı yeni bir hibrit tıbbi karar destek sistemini önermiştir. RS yaklaşımı ile veri setinden gereksiz özellikler kaldırılmış ve ELM üzerinden sınıflandırma işlemi gerçekleştirilmiştir. En yüksek %100 sınıflandırma doğruluğunun RS-ELM ile elde edildiğini göstermişlerdir. RS-ELM modelinin literatürdeki diğer metotlara göre daha çok başarılı olduğunu ve bu çalışmada hepatit tespiti için en önemli özelliklerin belirlendiğini belirtmişlerdir. (Kaya ve Uyar, 2013: 3429-3438)

Biyomedikal ses parametreleri kullanılarak MÖ teknikleri ile Parkinson hastalığı (PH-Parkinson's Disease-PD) veri seti üzerinde testler yapmıştır. PH'nın var/yok vaziyetinin sınıflandırılması ve başarı oranlarının saptaması yapılmıştır. Yapılan çalışmada KNN'nin diğer sınıflandırma tekniklerine göre bu veri seti üzerinde en başarılı sınıflandırıcı olduğunu belirtmiştir. (Bizal, 2014)

Akciğer kanserini tespit etmek ve gerçek kötü huylu nodülleri sınıflandırmak için DVM sınıflandırıcısını kullanmışlardır. 56 kanserli veri, 56 kötü huylu nodül, 745 iyi huylu nodül içeren ve 50 normal veri (akciğer enfeksiyonlu) üzerinde çalışma gerçekleştirilmiştir. Belirtilen çalışmada, %100 duyarlılık, %93 özgüllük ve %94 doğruluk sağlandığını bildirmişlerdir. (Manikandan ve ark., 2016: 1-9)

Kalp hastalığı ve kronik böbrek yetmezliğini tanısını elde etmek için bir çalışma yapmışlardır. Bu çalışmada California Irvine Üniversitesi makine öğrenmesi deposu (UCI- UCI Machine Learning Repository) 'dan elde ettikleri veri seti ile KNN

MÖ algoritmasını kullanmışlardır. Hastalığın tanısında, %90'lık bir doğruluk oranına ulaştıklarını belirtmişlerdir. (Tayeb ve ark., 2017: 3897-3903)

Zatürre hastalığının tanısı için Derin Öğrenme (DÖ- Deep Learning-DL) modellerinden Evrişimsel Sinir Ağını (ESA-Convolutional Neural Network-CNN) kullanılmışlardır. Zatürre hastalığını tanısı için elde edilen veriler üzerinden öznitelikler çıkarılmış, sınıflandırma yöntemleri kullanılarak karşılaştırılmışlardır. Yapılan çalışmaların sonucunda DVM algoritması ile %95,8 doğruluk oranı elde ettiklerini ve zatürre hastalığının erken tanısı için önemli olduğunu belirtmişlerdir. (Toğacar ve ark., 2019)

MÖ tabanlı Koroner Arter Hastalığını (KAH) tahmini için 1992 ve 2019 arasındaki bütün ilgili çalışmaları incelemişlerdir. KAH tanısını elde etmede KNN, DVM ve ANN' nin çoğu veri kümesi için en yüksek doğruluğu elde ettiklerini belirtmişlerdir. (Alizadehsani ve ark., 2019)

Karaciğer hastalıklarının tanısına yönelik bir çalışma yapmıştır. Yaptığı çalışmada, WEKA (Waikato Environment for Knowledge Analysis) programını kullanarak RO, REP Tree, J48, Hoeffding Tree, Lojistik Model Ağacı (LMT), IBk, Random Tree ve Decision Stump MÖ algoritmaları ile Hint Karaciğer Hasta Veri Seti (ILPD- Indian Liver Patient Dataset) üzerinde çalışmıştır. Yapılan çalışmada 0,819 en iyi doğruluk oranını RO algoritmasıyla elde edildiğini bildirmiştir. (Karslı, 2019)

MÖ tekniklerini kullanarak böbrek hastalığının tanısı için bir model öngörmüşlerdir. Yapılan çalışmada, DVM algoritması ile 0,97 doğruluk oranı, 0.961 geri çağırma oranı ve 0,986 kesinlik oranlarını hesaplanmışlardır. DVM algoritmasının daha iyi başarım gösterdiğini belirtmişlerdir. Geri çağırma açısından, KA algoritması 0,963 oranı ile en iyi başarıma ulaşmışlar. Hassasiyet açısından, çok katmanlı algılayıcı algoritması ile 0,994 oranı elde edilmiş, bu oran ile veri sınıflandırmada en iyi başarım ulaştıklarını belirtmişlerdir. (Takhti ve ark., 2019: 14-22)

MÖ ile prostat kanseri tanısı üzerinde çalışmışlardır. Yapılan çalışmada prostat kanserinin tanısı için, veri kümesi üzerinde Parçacık Sürü Optimizasyonu (PSO-Particle Swarm Optimization) ve Temel bileşenler analizi kullanılarak öznitelik boyut azaltılması yapılmıştır. Bu şekilde hastalıkları etkileyen genlerin tespit edildiğini belirtmişlerdir. DVM ve KNN algoritmalarını kullanmışlardır. Sınıflandırma başarı sonuçları değerlendirilmiştir. Çalışma neticesinde prostat kanserindeki etkin genler PSO boyut azaltma yöntemi ile tespit edilmiştir ve yaptıkları çalışmada %95,77 başarı elde edildiğini belirtmişlerdir. (Kılıçarslan ve ark., 2019: 769-777)

Karaciğer hastalığının erken tanısı, daha doğru ve sağlam bir yapı oluşturmak için otomatik bir programa ihtiyaç olduğunu bildirmişlerdir. UCI MÖ kitaplığından karaciğer hastalığı ile ilgili veriler alınmış, KA, RO ve DVM algoritmaları ile hastalığı tahmin etmek için kullanmışlardır. Yapılan çalışmada, en yüksek doğruluk oranını 0,9515 ile DVM algoritmasıyla, en iyi hassasiyet oranını 1,00 ile RO algoritmasıyla elde ettiklerini bildirmişlerdir. (Sivasangari ve ark., 2020: 627-630)

COVID-19 hastalığını tespit etmek için üç boyutlu göğüs BT görüntülerini kullanarak üç boyutlu bir DÖ modeli oluşturmuşlar ve yaptıkları bu çalışmayı COVNet olarak isimlendirmişlerdir. Yapılan çalışma sonucunda 0,96 AUC oranı elde ettiklerini ve COVID-19 hastalığını zatürree ve diğer akciğer hastalıklarından doğru bir şekilde ayırabildiklerini belirtmişlerdir. (Li ve ark., 2020)

Diyabet hastalığını erken teşhis etmek için AdaBoost, RO ve KNN algoritmaları kullanarak bir çalışma yapmışlardır. Çalışmada, diyabet hastalığını RO ve KNN algoritmaları %92,30'lik başarı oranı sağlandığını belirtmişlerdir. Çalışmada alınan bir başka sonuçta da diyabet hastalığının yanlış tespit edilmesi açısından RO ve KNN algoritmaları ile en düşük hata oranı sağlandığını belirtmişlerdir. (Veranyurt ve ark., 2020: 275-286)

PH teşhis etmek için bir çalışma yapmışlardır. 252 ayrı kişiden toplanan ses verileriyle PH'nın tespit edilebilmesi hedeflenmiştir. KA, DVM ve KNN sınıflandırma algoritmaları kullanılmıştır. Yapılan çalışmada en iyi performans değerlerini DVM

algoritmaları ile elde etmişlerdir. Bu çalışma ile PH'ın tespit işlemi aşamasındaki zorlu ve maliyetli işlemlerini kolaylaştırarak doktorlara karar noktasında yardımcı ve destek olunabileceğini bildirmişlerdir. (Esmer ve ark., 2020: 1877-1893)

Tiroid hastalığının teşhisinde farklı MÖ algoritmalarının performansını araştıran kapsamlı bir çalışma sunmuşlardır. KNN, DVM, AdaBoost, XGBoost, Gaussian Process Classifier, Gradient Boosting Classifier, Çok Katmanlı Algılayıcıyı (MLPC-Multiplier Perceptron Classifier) gibi makine öğrenme algoritmaları uygulamışlardır. %99,70'lik en yüksek doğruluğu veren MLPC'yi önermişlerdir. Tıp uzmanlarına tiroit hastalığının erken teşhisinde yardımcı olabileceğini belirtmişlerdir. (Asif ve ark., 2020: 222-225)

Cilt kanseri tahmini için gelişmiş MÖ'ye dayalı bir yaklaşım önermişlerdir. Bu çalışmada, MÖ ile yönetilen algoritmaların farklı varyantlarını kullanarak hastalık tahmin yöntemlerinin analizine odaklanmışlardır. Hastalığın tahmininde KNN ve ESA kullanmışlar. ESA'nın, genel bir hastalığı tahmin etmek için KNN algoritmasından %84,5 daha güvenilir olduğunu bildirmişlerdir. (Malladi ve ark., 2021)

Hastalardan alınan beyin omurilik sıvısını kullanarak Alzheimer hastalığı teşhisi çalışmasını yapmışlardır. Araştırmacılar, Alzheimer hastalığı teşhisi için, YSA ve DVM algoritmaları kullanılmış ve %84 duyarlılık ve özgüllük elde ettiklerini bildirmişlerdir. (Ryzhikova ve ark., 2021)

Alzheimer hastalığını daha önceden tespit etmek için RO, YSA, LR, DVM ve NB algoritmalarını kullanarak bir tahmin modeli oluşturmuştur. Yapılan çalışmada, RO algoritması uygulanarak 0,91 doğruluk oranına ulaşmış ve RO algoritmasının diğer tekniklere göre daha iyi performans sağladığını ortaya koymuştur. (Buyrukoğlu, 2021: 50-61)

Tip 2 diyabet hastalarında yaşam riskini tahmin etmek için makine öğrenimi yöntemlerini kullanmaktır. Deney, UK Longitudinal Study of Aging'den elde edilen

veriler kullanılarak oluşturulmuştur. Ağırlıklı oylama algoritmasının makine öğrenimi modeli en iyi sonuçları vermiştir ve başarılı olmuştur. (Fazakis ve ark., 2021)

Makine öğrenimi yöntemlerini kullanan bir çalışma, kronik böbrek hastalığında performans değerlendirmesini analiz etti. UCI tarafından bir kronik böbrek yetmezliği veri seti kullanılmıştır (UCI, 2015). İlk olarak, temel bileşen analizi yöntemini kullanarak veri kümesi üzerinde özellik çıkarımı yaptık ve ardından makine öğrenme algoritmalarını uyguladık. En güçlü başarı oranı rastgele orman algoritması kullanılarak elde edilmiştir. (Emon ve ark., 2021: 713-719)

2021 yılında kalp hastası olan bireylerin tespit edilebilmesi için MÖ modellerini kullanarak, bu algoritmalar değişkenler ve sınıflandırıcı üzerindeki ilişkileri ortaya koyarak bir çalışma gerçekleştirmişlerdir. LR, KNN ve RO yöntemleri kullanarak sırası ile RO ile %88, KNN ile %70 ve LR yöntemi ile de %85 oranında doğruluk oranları bulunmuştur. Bu çalışmanın literatüre olan yardımı ise problemde denenen yöntemler arasında en başarılı neticenin LR yöntemi ile gerçekleştirmiş olmalarıdır. (Coşar ve ark., 2021: 1112-1116)

Bulanık denklik bağıntısı ve yapay sinir ağları algoritması ile sınıflandırma metotuyla kişilerin kalp verilerini kullanarak hastalık tanısı üzerine çalışmışlardır. Bu çalışmanın literatüre olan katkısı ise çalışma neticesinde bulanık denklik bağıntısında %90 ve YSA algoritmasında %85 başarı oranı sağlamışlardır. (Archarya ve ark., 2003: 61-68)

İKİNCİ BÖLÜM
DİYABET HASTALIĞI

1. DİYABET HASTALIĞI TANIMI

Vücudun enerji ihtiyacı; besinlerdeki karbonhidrat, protein ve yağlardan elde edilir. Bu besinler sindirildiğinde, glikoz adı verilen basit bir şeker açığa çıkarırlar. Glikoz, tüm vücut organları için birincil beslenme kaynağıdır. Hücrelerin enerji için glikoz kullanabilmesi için hücrelere sağlanması gerekir. İnsülin, pankreas tarafından salgılanan ve glikozu hücrelere alarak glikojen olarak depolayan bir hormondur. Diyabet, yüksek kan şekeri ile ilişkili bir hastalıktır. Bunun nedeni pankreasın vücut için yeterli insülin üretememesi veya ürettiği insülini etkin kullanamamasıdır. Diyabet, vücutta pankreas adı verilen bir bezin yeterli insülin hormonunu üretememesi veya ürettiği insülin hormonunu etkin bir şekilde kullanamaması sonucu ortaya çıkan, yaşam boyu süren bir rahatsızlıktır. Bu durumun sonucunda yenilen besinlerden kana geçen şeker kullanılamaz ve kan şekeri yükselir (hiperglisemi). (Türk Diyabet Vakfı, 2023)

Yediğimiz besinlerin çoğu, özellikle karbonhidrat içerenler, vücutta enerji olarak kullanılmak üzere glikoza dönüştürülür. Midenin arkasındaki bir organ olan pankreas, "insülin" adı verilen bir hormon üretir. Bu, kasların ve diğer dokuların kandan glikoz almasını ve enerji için kullanmasını sağlar. Kana besinlerle birlikte giren glikoz, insülin hormonu vasıtasıyla hücrelere dağılır. Hücreler yakıt olarak glikoz kullanır. Glikoz miktarı bedenin enerji ihtiyacını aştığında karaciğerde (şeker deposu = glikojen) ve yağ dokusunda depolanır. (Türk Diyabet Vakfı, 2023)

Kan şekeri seviyesi açken 120 mg/dl'yi, tokken (yemeğe başladıktan 2 saat sonra) 140 mg/dl'yi geçmeyen diyabetik olmayanlar. Açlık veya tokluk kan şekeri değerlerinin belirtilen değerlerin üzerinde olması diyabetin varlığına işaret eder. (Türk Diyabet Vakfı, 2023)

Diyabet, açlık kan şekeri (FGL) veya oral glikoz tolerans testi (OGTT) ölçülerek belirlenir. 100-125 mg/dl'lik bir FBC okuması, gizli bir şeker sinyalidir (diyabet öncesi). 126 mg/dl veya daha yüksek bir FBC okuması, diyabetin varlığını gösterir. (Türk Diyabet Vakfı, 2023)

OGTT' de glikozdan zengin sıvı alımından 2 saat sonraki kan şekeri seviyesi önemlidir. 2 saatte 140 ila 199 mg/dl arasında bir kan şekeri okuması gizli glikoz olarak teşhis edilir ve 200 mg/dl veya üstü diyabet olarak teşhis edilir. (Türk Diyabet Vakfı, 2023)

1.1. Diyabet Hastalığı Tipleri

Birkaç diyabet türü vardır, ancak üç ana türü vardır. Bu; tip 1 diyabet, tip 2 diyabet, gebelik diyabeti. Diyabette, vücut ihtiyaç duyduğu insülini üretemediği veya kullanamadığı için kan şekeri seviyeleri yükselir. Tip 1 diyabette vücudun kendi insülin üretimi azalır. Tip 2 diyabet ve gestasyonel diyabette, vücut insülinin etkisine karşı direnç geliştirir. Bu durumların her ikisi de hiperglisemiye (hiperglisemi) neden olur. Henüz tam olarak ortaya çıkmamış bir diyabet şekline gizli diyabet veya prediyabet denir. Tip 2 diyabet, yüksek kan şekeri seviyelerine neden olan çok yaygın bir durumdur. Bu, vücut hücrelerinin normalde ürettiği insüline dirençli hale geldiği ve kan şekeri seviyelerinden yararlanamadığı bir durumdur. Obezite, hareketsiz bir yaşam tarzı, stres, ailede diyabet öyküsü ve yaşlanma, tip 2 diyabete katkıda bulunur. Ancak tip 2 diyabetin semptomları her zaman kendinizi kötü hissetmenize neden olmadığından, onları fark etmek her zaman kolay değildir. Ciddi kalp ve sinir sorunları ve aşırı susama, sık idrara çıkma ve yorgunluk gibi belirtiler daha olasıdır. Tip 2 diyabet, insanları hayatlarının geri kalanında etkileyen bir hastalıktır. Bunu kontrol etmek için diyet değişiklikleri, ilaçlar ve düzenli kontroller gerekebilir. Tip 1 diyabet, vücudun kan şekerini kontrol etmek için yeterli insülin üretemediği bir durumdur. Bunun sonucunda kan şekeri (glikoz) çok yüksek değerlere ulaşır. Çok yüksek kan şekeri seviyelerini kontrol etmek için günlük insülin enjeksiyonları gereklidir. Tip 1 diyabet genellikle erken yaşta başlar. Vücudun bağışıklık sistemi pankreastaki insülin üreten hücrelere saldırdığında ortaya çıkar. Tip 1 diyabet, diyabetik ketoasidoz veya DKA'ya yol açabilir. DKA, vücutta ciddi bir insülin eksikliği olduğunda ortaya çıkar.

Şekeri enerji için kullanamayan vücut, bunun yerine vücutta depolanan yağları kullanmaya başlayacaktır. Vücut depolanmış yağları kullanırken geride keton adı verilen kimyasallar bırakır. Bu durum kontrol altına alınmazsa kanda ketonlar birikerek kan asitliğini artırır. Tip 1 diyabetin farkında olmayan kişiler, özellikle çocuklar, DKA ile şiddetlenene kadar teşhis edilemeyebilir. Bu nedenle, derhal tedavi edilebilmesi için DKA'nın belirti ve semptomlarını tanımak önemlidir. Beklenmeyen kilo kaybı, tip 1 diyabetin semptomlarından biridir. Vücut yiyeceklerden enerji alamayınca, enerji için sahip olduğu kas ve yağları yakmaya başlar. Böylece diyetinizi veya egzersiz tarzınızı değiştirmeden kilo vermeye başlarsınız. Vücudunuzun yağ yakarken ürettiği keton cisimleri mide bulantısı ve kusmaya neden olabilir. Ketonlar kanda tehlikeli seviyelere çıkabilir ve yaşamı tehdit edebilir. Gebelik diyabeti nedir? Gestasyonel diyabet veya gestasyonel diabetes mellitus (gestasyonel diabetes mellitus), hamilelikten önce yeterli insülin yapabilen pankreas hücrelerinin hamilelik ilerledikçe yeterli insülini üretemez hale gelmesiyle ortaya çıkar. Gestasyonel diyabetle ilişkili daha önce diyabet semptomlarınız olmasa bile, hamilelik sırasında kan şekeri düzeyleriniz yükselebilir. Bu durum genellikle hamileliğin sonunda kendi kendine düzelir. Ailesinde diyabet öyküsü olan kişiler, 30 yaşın üzerindeki kişiler ve aşırı kilolu kişiler hamilelik sırasında diyabet geliştirme riski altındadır. Normal şartlar altında, bir kişinin kan şekeri seviyeleri normalden yüksek olduğunda ancak diyabet teşhisi için yeterli olmadığında, durum subklinik diyabet veya prediyabet olarak tanımlanır. Çalışmalar, subklinik diyabetli kişilerin genellikle 10 yıl içinde tip 2 diyabet geliştirdiğini göstermiştir. Sağlıklı bir diyet ve daha aktif bir yaşam tarzı, subklinik diyabetin tip 2 diyabete ilerlemesini önlemenin ve yavaşlatmanın etkili yolları olarak kabul edilir. (Türk Diyabet Vakfı, 2023)

Vücudumuzun enerji ihtiyacı besinlerde bulunan temel besinler, karbonhidratlar, proteinler ve yağlardan karşılanır. Emilmesi için en küçük bileşenlerine ayrılan en önemli besin "glikoz" adı verilen basit bir şekerdir. Glikoz başta beyin olmak üzere vücudun tüm organları için önemli bir besindir. Hücreler ihtiyaç duydukları glikozu midenin arkasında yer alan pankreasın salgıladığı hormonlar yardımıyla kullanırlar. Vücudunuz artık insülin olarak bilinen bu hormonu üretmediğinde, yediğiniz yiyecekler enerji için kullanılmaz. Tip 1 diyabet, insülin

hormonu eksikliğinden kaynaklanır ve sıklıkla çocukluk ve ergenlik döneminde geliştiği için juvenil diyabet olarak adlandırılır. (Türk Diyabet Vakfı, 2023)

Tip 1 diyabet, pankreastaki insülin üreten beta hücrelerinin bir otoimmün süreç tarafından hasar görmesi sonucu ortaya çıkar. Mutlak veya bağıl insülin eksikliği nedeniyle, hastalar hayatlarının geri kalanında insülin hormonunu eksojen olarak (enjeksiyon yoluyla) almak zorundadır. Bu nedenle tip 1 diyabet, insüline bağımlı diabetes mellitus (IDDM) olarak da adlandırılır. Genel olarak, tip 1 diyabet toplumdaki diyabet vakalarının %1'ini oluşturur. Çocukluk çağı tip 1 diyabetin prevalansı ülkeye (bölgeye) göre değişir ve diyabet her yıl 15 yaşın altındaki 100.000 çocukta 1 ila 42 yaş arasında ortaya çıkar. Tip 1 diyabet genellikle kuzey ülkelerinde daha yaygındır. (Türk Diyabet Vakfı, 2023)

Sağlıklı bir insan, vücudu dış etkenlerden koruyan bir bağışıklık sistemine sahiptir. Virüs, aşı, ilaç, fiziksel ya da psikolojik stres gibi herhangi bir nedenle bu sistemin normal durumundan saptığı, vücudun kendi hücrelerini yabancı madde olarak algılayıp onlara saldırdığı ve yok ettiği bir hastalıktır. Tip 1 diyabet ayrıca bir otoimmün hastalık olarak sınıflandırılır. Bilinmeyen nedenlerle bağışıklık sistemi, insülin üretimini devralan pankreas beta hücrelerine müdahale eder ve onları yok eder. Bu hasar alfayı aşarsa, hastalığın belirtileri ortaya çıkar. (Türk Diyabet Vakfı, 2023)

Tip 1 diyabet tedavisinde değişmez bir kural insülin enjeksiyonlarıdır. Bu diyabet tipinde insülin kullanımı gereklidir ve bir cankurtarandır. Tedavinin diğer temel taşları sağlıklı beslenme, düzenli egzersiz ve eğitimdir. Gün boyunca ideal kan şekeri seviyelerini korumak, büyük özen ve günlük bakım gerektirir. Kişinin kendini iyi hissetmesi ve sağlıklı bir yaşam sürmesi için ihtiyaç duyduğu özen, bir yaşam biçimi olmalıdır. (Türk Diyabet Vakfı, 2023)

Tıbbi beslenme tedavisi, yani diyet ayarlamaları, yaşam tarzı değişiklikleri ve egzersiz programının uygulanması birincil tedavi planına dahildir. Bu tedavi rejimi kan şekerini normal sınırlarda tutmuyorsa, tablet şeklinde alınan oral hipoglisemik ilaçlarla tedavi desteklenir. Bununla birlikte, tip 2 diyabetli bazı kişiler kan şekerini

normal sınırlar içinde tutmak için insüline ihtiyaç duyabilir. Bu durumlarda uygun dozda insülin enjeksiyonları ile tedavi desteklenir. Oral hipoglisemik ilaçlar veya insülin tedavisi alan tip 2 diyabet hastaları için haftanın belirli günlerinde kan şekeri düzeylerinin ölçülmesi çok önemlidir. (Türk Diyabet Vakfı, 2023)

1.2. Diyabet Tanısı

Şeker hastalığını teşhis etmek için kullanılan en önemli iki test, açlık kan şekeri testi ve oral glikoz tolerans testidir (OGTT) (glikoz tolerans testi olarak da bilinir). Sağlıklı insanların ortalama açlık kan şekeri seviyesi 70 ila 100 mg/dl'dir. Açlık kan şekerinin 126 mg/dl veya daha yüksek olması diyabet tanısı için yeterlidir. Bu değer 100-126 mg/dl arasında ise OGTT verilerek kişinin tokluk kan şekerine bakılır. Yemeğe başladıktan 2 saat sonra kan şekeri ölçümü sonucunda 200 mg/dl'yi aşan kan şekeri diyabet belirtisi, 10-199 mg/dl aralığı ise prediyabetik evredir. Bu sözde gizli şekerdir. Ayrıca son 3 aydaki kan şekerini tanımlayan HbA1C testleri de %7'nin üzerinde çıkarak diyabet teşhisi konmaktadır.(Medikal Park, 2023)

1.3. Diyabet Tedavisi

Diyabet kesin tedavisi olmayan kronik bir hastalıktır. Hastalığı tedavi etmenin amacı, olumsuz etkilerini önlemek ve hastanın yaşam kalitesini düşürmektir. Hastalığın etkilerini en aza indirmek için kan şekeri düzeylerini normal sınırlar içinde tutmak önemlidir. Uzun vadede komplikasyon riskini azaltmak için hastaların diyabet konusunda eğitilmeleri, kan şekerlerini kontrol etmeleri, doğru beslenmeleri ve yeterli egzersiz yapmaları çok önemlidir. Tip 2 diyabette antidiyabetik ilaçların kullanımı ve tip 1 diyabette insülin tedavisi farmakolojik diyabet kontrol yöntemleridir. Obez diyabetik hastalarda gastrik bypass ameliyatından sonra kan şekeri seviyelerinin sıfıra inebildiği gözlemlenmiştir ancak bu yaygın olarak kullanılan bir çözüm değildir. Diyabet tedavisinde, kan şekeri seviyelerini kontrol etmek için sağlıklı bir diyetin benimsenmesi önemlidir. Doğru ve dengeli beslenmeyi öğrenmek ve öğrendiklerinizi günlük yaşamınıza uygulamak, tıpkı diyabeti olmayan insanlar için olduğu gibi diyabet hastaları için de sağlıklı bir yaşamın temelidir.(Acıbadem, 2023)

Diyabet tedavisinin amacı, kan şekerini normal sınırlar içinde tutarak, kısa veya uzun vadeli sağlık sorunlarını önlemek veya geciktirmektir. (Türk Diyabet Vakfı, 2023)

Sanılanın aksine, insülin enjeksiyonları tütün veya alkol kadar bağımlılık yapmaz veya bağımlılık yapmaz. İnsülinin hayat kurtaran bir ilaç olduğunu hatırlamak ve kendinize enjekte etmekle aslında sağlıklı bir hayat yaşamak için yapmanız gerekenleri yapmış olmak kavramına alışmanıza yardımcı olacaktır. (Türk Diyabet Vakfı, 2023)

İnsülinin rolünü anlamak için öncelikle vücudumuzun işlevlerini yerine getirmek için ihtiyaç duyduğu enerjiyi nasıl sağladığını kısaca anlamamız gerekir. Yediğimiz besinler sindirildikten sonra vücudumuzdaki enzimler tarafından parçalanarak şekere dönüştürülür. Şeker (glikoz) kan dolaşımında vücutta dolaşır. Vücudumuzun ana besin kaynağı olan şeker, kandan vücut hücrelerine (kas, yağ, karaciğer hücreleri) enerji sağlamak zorundadır. İnsülin vücudumuzda midemizin altında ve arkasında yer alan pankreasın beta hücrelerinin salgıladığı bir hormondur. Bu, kandaki şekerin kanı terk etmesini ve hücrelere girmesini sağlar. (Türk Diyabet Vakfı, 2023)

Diyabette glisemik kontrol için beslenme tedavisi, ilaç tedavisi ve insülin tedavisi en önemlileridir ancak egzersiz ihmal edilmektedir. Ancak egzersiz, diyabet tedavisinin en az beslenme tedavisi ve ilaç tedavisi kadar önemli bir parçasıdır. (Türk Diyabet Vakfı, 2023)

Diyabetle yaşamak, karşılaştığınız bazı problemlerin nasıl çözüleceğini öğrenmeyi ve tedaviye uyum sağlamayı gerektirir. Şeker hastalarının, özellikle insülin kullananların karşılaşabileceği bir diğer sorun da hipoglisemidir. Hipogliseminin önlenmesi ve tedavisi glisemik kontrolü sağlamak için çok önemlidir. (Türk Diyabet Vakfı, 2023)

Hipoglisemi, kan şekeri seviyesinin 50 mg/dl veya daha az olması olarak tanımlanır. Hipogliseminin nedeni ortadan kalktıktan sonra hipoglisemi riski de

ortadan kalkar. Aksi halde insülin veya oral antidiyabetik ilaç kullanan kişilerde hipoglisemi oluşabilir. (Türk Diyabet Vakfı, 2023)

1.4. Diyabet Riski

Basit bir test diyabet riskinizi belirleyebilir. Tedavi edilmeyen diyabet, çeşitli sağlık sorunlarına yol açabilir. Erken teşhis size sağlıklı bir yaşam sürme şansı verir. 1997 yılında Türkiye'de yapılan bir araştırmaya göre ülkemizde yaklaşık 3,6 milyon kişi diyabet hastası olup, bunların 1,2 milyonu henüz teşhis edilememiştir. Tip 2 diyabetin belirtileri genellikle 40 yaşın üzerindeki kişilerde görülür, ancak normal yaşlanma sırasında fark edilmeyebilirler. Şeker hastalığı çoğu zaman tesadüfen teşhis edilirken, kişiler başka sağlık sorunları için başvurduklarında veya zayıfladıklarında teşhis edilir. Bu semptomları olan kişiler, bunların aşırı çalışma veya stresten kaynaklandığına inanarak genellikle doktora gitmekten kaçınırlar. Başka bir diyabet türü olan tip 1 diyabet, nadiren uzun süre fark edilmez. Tedavi edilmeyen tip 1 diyabet hızla komaya veya ölüme yol açabilir. Tip 1 diyabet genellikle çocukları ve ergenleri etkilerken, tip 2 diyabet genellikle orta yaşlı ve yaşlıları etkiler. Tip 2 diyabet, diyabet vakalarının %95'inden sorumludur. Erken teşhis size normal, sağlıklı bir yaşam sürme şansı verir. Tip 2 diyabet tedavisi, bir miktar kilo kaybı, dengeli beslenme ve artan fiziksel aktiviteden oluşur. Bu önlemler yeterli olmazsa, doktorunuz ilaç veya insülin tedavisi verebilir. (Türk Diyabet Vakfı, 2023)

Tip 2 diyabet dikkat gerektiren bir hastalıktır. Tedavi edilmediği takdirde sinir hasarı, böbrek hastalığı, kardiyovasküler hastalık ve inme gibi hayatı tehdit eden ciddi komplikasyonlara yol açabilir. Bazı doktorlar da dahil olmak üzere birçok kişi, tip 2 diyabetin tip 1 diyabetten daha az ciddi olduğuna inanır ve tip 2 diyabeti hatalı bir şekilde "sınırda diyabet" veya "biraz şeker" olarak tanımlar. (Türk Diyabet Vakfı, 2023)

Diyabet teşhisi kolaydır ve sağlığınızın objektif bir değerlendirmesi ile başlar. Araştırmalar, yakın akrabalarda obezite ve tip 2 diyabetin diyabet gelişme riskini artırdığını göstermiştir. Ek olarak, bazı etnik grupların diyabet riski daha yüksektir. (Türk Diyabet Vakfı, 2023)

ÜÇÜNCÜ BÖLÜM
YAPAY ZEKA VE MAKİNE ÖĞRENMESİ

1.YAPAY ZEKA

Yapay zeka bağımsız makineler -bu makineler insan olmaksızın kompleks işler yapabilir- inşa etmek için araştırma yapan bilişsel bilim dalıdır. Bu hedef makinelerin düşünmesini ve anlamasını gerektirir. (Pirim, 2006: 81-93)

1.1.Yapay Zeka Tarihçesi

Dünya tarihinde ilk kez "yapay zeka" terimi 1950 yılında Hanover, New Hampshire'daki Darthmouth College'da düzenlenen bir konferansta kullanıldı ve bu fikrin babası Alan Matheson Turing'di. 1943 yılında, 2. Dünya Savaşı sırasında, Alan Turing, kriptografik bir algoritma olan Alman Nazi Enigma Makinesi'ni geliştiren ve çözen en ünlü matematikçidir. (Halis, 2022)

Yapay zekanın varlığını ya da algılanmasını sağlamak kolay olmamıştır. Pek çok insan hayatını, henüz aktif olarak kullanamayan ve günümüze kadar gelen yapay zekayı yaratmaya adadı. Yapay zeka terimi, yaklaşık üç yıl sonra, J. McCarty ve Marvin Minsky'nin Massachusetts Institute of Technology'de ilk Yapay Zeka Laboratuvarını kurmasıyla, 1956'da icat edildi ve ilk kez konuşuldu. (Halis, 2022)

1974-1980'de pek çok kişide yankı uyandıran yapay zeka olgusu kısa sürede çökmüştür. Bu ciddi gerilemenin en büyük nedeni, yapay zekanın gelişim sürecini eleştiren birçok raporun hazırlanması ve yayınlanmasıydı. Son birkaç yıl yapay zeka kışı olarak adlandırıldı. 1987'den 1993'e kadar çok başarılı yedi yılın ardından, yapay zeka bütçesi, azalan hükümet desteğiyle bir kez daha kısıtlandı. Bunun nedeni, bilgisayarların o zamanlar bugünkü kadar popüler olmamasıydı. (Halis, 2022)

1997'den 2020'nin sonuna kadar çalışmalar hız kazandı ve yapay zekanın canlanmasına yardımcı olan IBM, "Satranç bir insan oyunudur" dedi. (Halis, 2022)

Şekil 1. Yapay Zekâ Tarihçesi

Kaynak: Görgün, M. *Makine öğrenmesi yöntemleriyle kalp hastalıklarının tahmin edilmesi* (Master's thesis, Lisansüstü Eğitim Enstitüsü).

İlerleyen zamanlarda yapay zeka daha da gelişecek, bütün dünyayı kuşatacak ve insanlığa katkı sağlayacaktır. Okullarda, hastanelerde, sokaklarda ve hatta evlerde daha çok yer bulacağız. İnsanlığa en fazla zarar tam anlamıyla tembelliktir. (Halis, 2022)

Dünya tarihindeki üç önemli olaydan bahsedebiliriz. Evrenin yaratılışı ve ikinci hayatın başlangıcı. Bir başka büyük olay da yapay zekanın yükselişi. Yapay zeka nedir? Soruyu sormadan önce zekanın ne olduğunu tanımlarsak, zekayı rasyonel düşünme olarak kolayca açıklayabiliriz. AI'da buna modelleme zekası diyebiliriz. Rasyonel düşünme (doğru eylem), mevcut bilgilere dayalı olarak bir hedefe ulaşmada en büyük faydayı sağlayan eylemdir. Makinelerin yapay ve doğal zekası, insan zekası ve hayvan düşüncesinin kanıtladığı gibi bilinç ve duygu içerir. Doğal zekada, özellikle yapay zekada veri iletimi, yapay zekaya göre daha karmaşık, zaman alıcı ve çalıştırılması zordur. Ancak bu verilerin dokümantasyonu, raporlaması,

sınıflandırılması ve diğer analiz sonuçları makine ortamında saklandığından daha stabildir. Aslında yapay zekayı temel alan birçok bilim dalı var ve bunlar disiplinler arası bilimler. Bilgisayarların yaygınlaşması tarafsız kararlar almayı mümkün kılmıştır. (Halis, 2022)

Hatasız işlemler gerçekleştirin ve uzun süren görevleri veya görevleri hızlı bir şekilde tamamlayan sürekli çalışan makine. Bir bilgisayarın belirli bir işlemi gerçekleştirme yeteneği Bunun için program arka planda çalışır ve gerekli fonksiyonlar ona insan eliyle uygulanır. (Halis, 2022)

Eğitim, AI alanının ana hedeflerinden biridir. 1950'lerde Alan Turing "Kendi kendine öğrenen makine" ve "düşünen makine" tanımlarını netleştirerek Günümüzün AI kavramlarının temelini attı. En basit haliyle AI bir makinedir. İnsan zekasını kolayca taklit edebilir ve basitten karmaşığa kadar sorunları çözebilir. Sorunları çözebilecek makineler ve bilgisayar programları geliştirmek için oluşturuldu. Teknoloji ve bir tür bilim olarak tanımlanmaktadır.(Alpaydin, 2010)

AI'nın başka bir tanımı, bir insan bilgisayarı veya bilgisayar tabanlı bir cihazdır. İnsanların olayları ve durumları anlamaları Anlam türetme, çözüm arama, genelleme yapma ve geçmiş deneyimlerden öğrenme. Bir bilgisayar veya bir bilgisayarın belirli bir işlem aracılığıyla bir görevi gerçekleştirme yeteneği Benzer şekilde makine yapımı şeyler de yapay zeka olarak tanımlanmaktadır.(Öztürk ve Şahin, 2018: 25-36)

AI, zekanın özelliklerini sergileyen sistemleri anlama ve tasarlamayı amaçlayan geniş bir araç yelpazesidir.

1.2. Yapay Zeka Uygulama Alanları

Ses tanıma, Görüntü işleme, Doğal dil (lisan) işleme ve kıyaslama alanları yapay zekanın günümüzde en çok kullanılan alanlarıdır. (Halis, 2022)

Aşağıda yapay zekanın günümüzde en çok kullanılan alanları

- Savunma sanayi ve siber güvenlik
- Ses tanıma ve anlama
- Dil çevirileri
- Navigasyon sistemleri
- Öneri sistemleri
- E-ticaret
- Yardımcı robot uygulama sistemleri
- Sağlık hizmetleri sistemleri

2. MAKİNE ÖĞRENİMİ

Makine öğrenimi, öğrenilenleri test etmek ve sonuçlar çıkarmak için bilgisayar sistemleri tarafından sağlanan bilgileri öğrenmek ve değiştirmek için algoritmaların ve istatistiksel tekniklerin kullanılmasıdır. Yapay zekanın bir alt dalı olan makine öğrenimi, eğitim verilerini algoritmalarına göndererek kararlar verir ve bu kararlar matematiksel olarak oluşturulmuş modellerle desteklenir. (Bishop, 2006: 738)

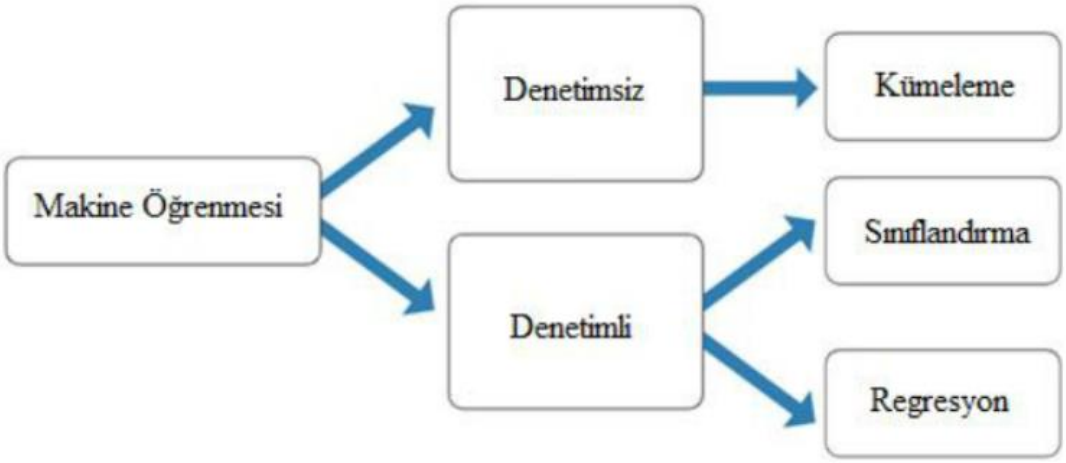
Değerlendirme için tasarlanan bu model, ilerleyen zamanı tahmin etme görevini üstlenmektedir. Neticede, bilgiler eğitim verileri ve test verileri olarak ayrılır. Sonuçlar, algoritmanın içerdiği bilgilerin yüzdesi olarak verilir. Makine öğrenimi tarihinde bilim insanı Alan Turing, yapay zeka araştırmalarının başlangıcı olarak görülebilir. Adı Turing testinde adımı aldı. Bu, yapay zeka çalışmalarının başlangıcıdır. Alan Turing, 2. Dünya Savaşı sırasında Alman ordusu tarafından kullanılan iletişim aracıydı ve savaşı kısaltmak için kullanılan bir Enigma koduna sahipti. 1956'da Stanford Üniversitesi'nden McCarthy, yapay zeka terimini ilk olarak Darmount College yaz okulunda kullandı. Yapay zeka terimi icat edilmeden önce Turing, makine zekası terimini kullanıyordu. Yapay zekanın en yaygın alanlarından biri olan makine öğrenimi adı ilk olarak 1959 yılında Arthur Samuel tarafından geliştirilen Dama programı ile kullanılmıştır. O zamandan beri makine öğrenimi üzerine araştırmalar devam etti, ancak "Farklı olan ne?" Teknoloji oyunlarına

odaklandı. Görüntü işlemcileri, ses işlemcileri, veri madenciliği, robot kodlama gibi birçok alanda yapay zeka ve makine öğrenimi kullanılmaktadır. Turing ve McCarthy her zaman insan beyninin düşünen bir makine olma ihtimalinden bahsederdi. Turing'e göre amaç, insan düşüncesini incelemek ve bir düşünen makine inşa etmek, insan düşünce sistemini çözmeye yardımcı olabilir. Yani "Bilgisayar Mekanizmaları ve Zeka" adlı makalesinde dediği gibi, bilgisayarların akıllı hareket etmesinin mümkün olduğunu söylemiştir. Turing'e göre zayıf ve güçlü yapay zeka arasında net bir ayrım var. İnsan düşüncesini taklit eden yapay zeka, zeka bakımından zayıftır. Fakat, iyi programlanmış yapay zeka çoğunlukla zihinsel ve güçlüdür. (Topal, 2017: 1340-1364)

Derin öğrenme, makine öğreniminin bir alt dalıdır ve makine öğreniminden daha az dokunuşla daha fazla veri gerektirir. Derin öğrenmenin kullanımı 2010'larda popüler hale geldi ve büyük veride rol oynadı. Derin öğrenmede makine öğrenmesi, birden çok katmanda işlenen verileri eş zamanlı olarak işleyerek, makine öğrenimi kitaplıklarını yeniden kullanarak, bunlarda tanımlanan parametreleri bularak ve hangi parametrelerin en iyi değerleri ve oranları verdiğini belirleyerek çalışır. Tekdüze ve etkili bir yöntem hayatımıza girdi. (Halis, 2022)

2.1. Çalışma Teknikleri

Karmaşık, büyük veri setlerinden anlamlı, temiz veri çıkarma işlemine veri madenciliği denir. Veri madenciliği keşifleri, 2017 yılına kadar makine öğrenimini etkiledi. Derin öğrenme araştırması bu şekilde başlamıştır. Makine öğrenimi icat edildiğinden beri yöntemler sürekli olarak gelişti. Çeşitli yöntemler bu geliştirme yöntemleri aracılığıyla bilimi etkilemiştir. Algoritmayı dikkate alarak bu yöntemleri şu şekilde sıralayabiliriz: Sınıflandırma, kümeleme, regresyon ve veri ilişkilerinin belirlenmesi olarak bilinen bu yöntemler, daha sağlıklı sonuçlar üretmek için veri kümelerini etkiler.



Şekil 2. Makine Öğrenmesi ve Alt Dalları

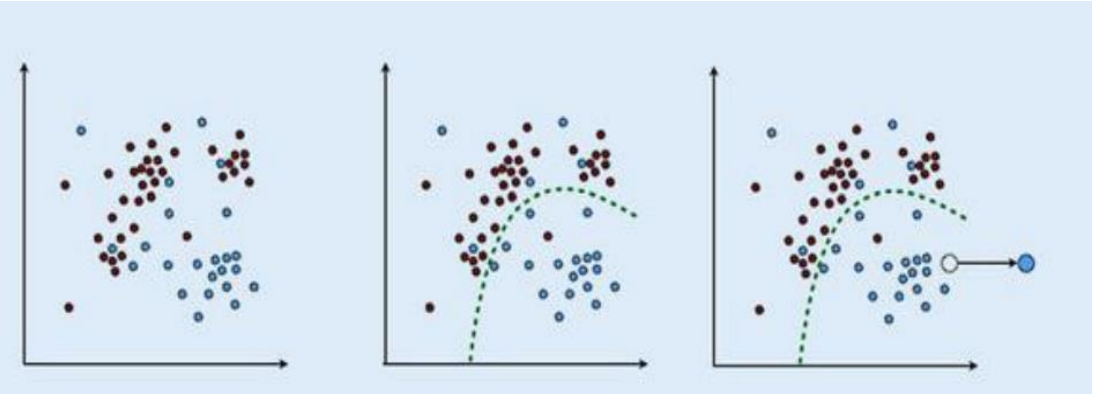
Kaynak: Sakal, (2023), http://muratsakal.com/?p=230, Erişim Tarihi:02/04/2023

2.1.1. Denetimli öğrenme

Denetimli öğrenmede, sistem etiketli verileri kullanan bir veri seti, kullanıcıların belirlediği girdileri ve çıktıları içerir. Bir algoritmanın bu girdileri ve çıktıları nasıl elde ettiğini açıklar. Verilerin bölündüğü sınıf olarak bilinir ve bu sınıflar için çalışır. Bu öğrenme yönteminde, bir algoritma tahminler yapar ve makine hızlanana kadar bunları ayarlar. Denetimli öğrenme tekniklerinin iki tanımlayıcı görevi vardır:

Sınıflandırma, bir veri kümesinden özelliklerine göre bilgi çıkarmaktır. Gözlem verilerinin sonucu makine öğrenmesi ile sınıflandırılır.

Regresyon, makine öğrenimi algoritmalarının değişkenler arasındaki bağlantıları anlaması ve tahmin etmesi gerekir. Bir bağımlı değişkene ve başka bir bağımsız değişken grubuna odaklanır. Eğilimleri tahmin etmek ve çözümlemek için kullanılır.



Şekil 3. Denetimli Öğrenme

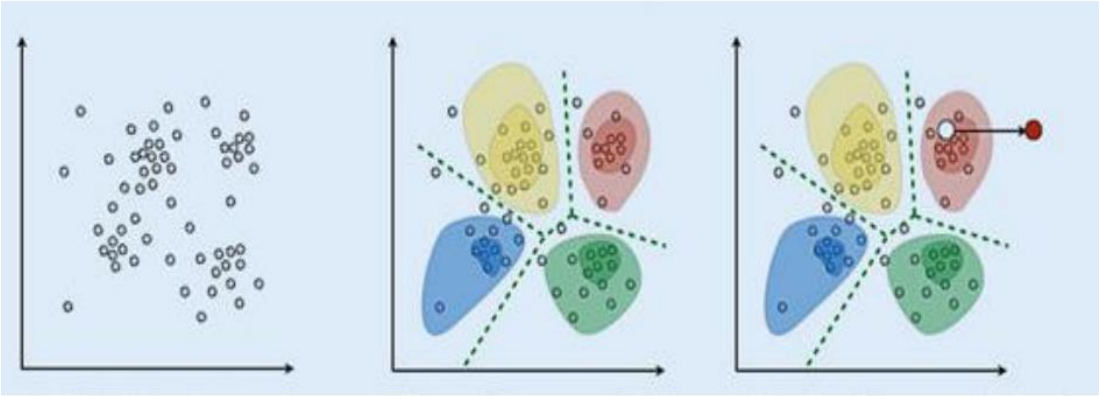
Kaynak: Langs, G., Röhrich, S., Hofmanninger, J., Prayer, F., Pan, J., Herold, C., & Prosch, H. (2018). Machine learning: from radiomics to discovery and routine. *Der Radiologe*, *58*(Suppl 1), 1.

2.1.2. Yarı Denetimli öğrenme

Genel olarak, bir kayıttaki etiketlenmiş veri niceliği, etiketlenmemiş veri niceliğinden daha azdır. Bu etiketlenmemiş veriler diğer tekniklerde nadiren kullanılır. Yarı denetimli öğrenme, etiketlenmemiş öğrenmeye izin veren denetimli öğrenmedir. Araştırmalar, kullanılabilir veri etiketleri olmayan verilerin, az miktarda etiketli veri kullanmaktan daha sorunsuz ve sağlam değerler sağladığını göstermiş ve kanıtlamıştır. Yarı denetimli öğrenme, denetimli ve denetimsiz öğrenme arasında yer alır.

2.1.3. Denetimsiz öğrenme

Denetimsiz öğrenmede, talimat veren bir insan operatör yoktur. Yalnızca makineler, bağlantıları ve korelasyonları belirlemek için veri kümelerindeki verileri analiz eden karar verme makineleridir. Sonuçların yorumlanması tamamen makine öğrenimi algoritmalarına bırakılmıştır. Ne kadar çok veri işlenirse, karar motoru o kadar güçlü olur.



Şekil 4. Denetimsiz Öğrenme

Kaynak: Langs, G., Röhrich, S., Hofmanninger, J., Prayer, F., Pan, J., Herold, C., & Prosch, H. (2018). Machine learning: from radiomics to discovery and routine. *Der Radiologe*, *58*(Suppl 1), 1.

2.1.4. Takviyeli Öğrenme

Bu, bir veri kümesinden bir seri işlevi, sınıf parametresini ve nihai değerleri içeren klasik bir öğrenme sürecini kullanan bir öğrenme metotudur. Makine öğrenimi algoritmaları, çeşitli özeti ve seçenekleri değerlendirerek sonuçlar üretmeye çalışır. Önerilen makine öğrenimi, bize nasıl hata yapacağımızı ve bunların tekrar olmasını önlemek için onlardan nasıl sonuç çıkaracağımızı öğretir. Geçmiş deneyimlerden sonuçlar çıkararak en iyi sonuçları almaya çalışır.

2.1.5. Aşırı öğrenme

Algoritma eğitim verilerini hatırladığından ve bu notları test verileriyle birlikte kullanmaya çalıştığından, eğitim ve test verileri arasındaki büyük bir doğruluk oranına aşırı öğrenme denir.

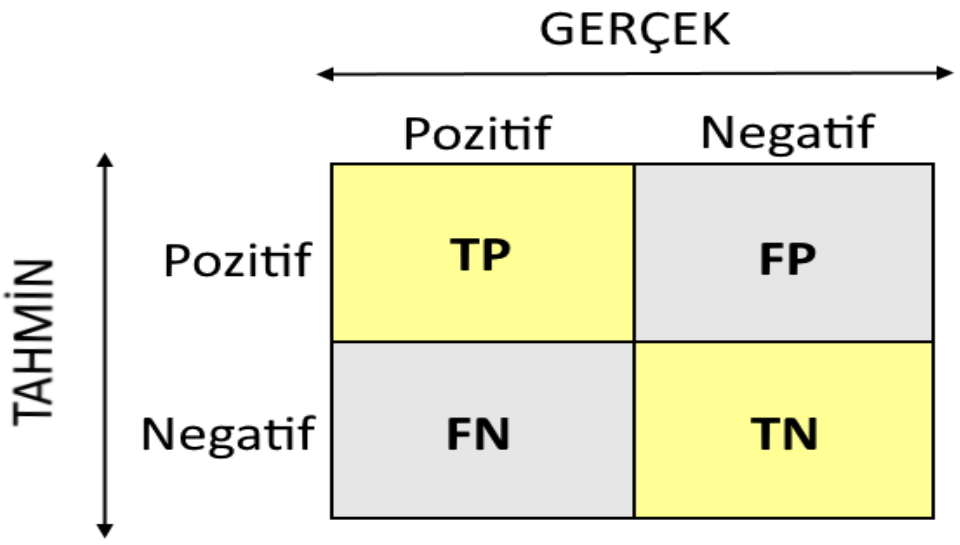
2.1.6. Test Prosedürü

Tipik olarak, bir makine öğrenimi algoritmasıyla eğitim tamamlandığında, veri setinin ayrılan üçte biri test edilir ve sonuçlar bir başarı oranını gösterir. Bu adımda

eğitim algoritması, algoritmik modelin başarı oranını ölçmek için daha önce hiç karşılaşmadığı verilerin üçte ikisine bakar. İsabetleri gerçek pozitifler, gerçek negatifler, yanlış pozitifler ve yanlış negatif olarak ayırmaktadır. Bu değerlerden karışıklık matrisi oluşturulur.

2.1.7. Karışıklık matrisi

Bu, doğruluk kontrolü için kullanılan hedeflerin sayısıyla birlikte büyüyen bir kare matristir. Algoritmaların çıktılarından oluşan bu çalışma, makine öğrenmesindeki algoritmaları incelemektedir. Öğrendikten sonra, optimum algoritmaları ve parametreleri belirlemek için sonuçlar kullanıcılar tarafından test edilir ve değerlendirilir. Karışıklık matrisi, algoritma tarafından tahmin edilen sonuç için doğru ve yanlış tahminlerin oranını verir.



Şekil 5. Karışıklık Matrisi

Kaynak: Bilen, B., https://burhanbilen.medium.com/, Erişim Tarihi: 02/04/2023

TN, FP, FN ve TP, durum ve kestirim arasındaki bağlantıdan asıl bir açıklama vermek için kullanılır.

Doğru & Doğru => Doğru (True Positive-TP)

Doğru & Yanlış => Yanlış (False Positive-FP)

Yanlış & Doğru => Yanlış (False Negative-FN)

Yanlış & Yanlış => Doğru (True Negative-TN)

Gerçek Pozitif Değerlerin Oranı (True Positive Rate): Recall, hassasiyet şeklinde de isimlendirilir. Gerçek pozitif değerlerin ne kadarının doğru tahmin edildiğinin ölçüsüdür. Yüksek olması tercih edilir. Bu hesaplama işlemi aşağıda gösterildiği gibidir.

TPR=TP / (TP + FN)

Gerçek Negatif Değerlerin Oranı (True Negative Rate): Özgüllük veya Seçicilik şeklinde de isimlendirilir. Gerçek negatif değerlerin ne kadarının doğru tahmin edildiğinin ölçüsüdür. Bu hesaplama işlemi aşağıda gösterildiği gibidir.

TNR= TN / (TN + FP)

Yanlış Pozitif Değerlerin Oranı (False Positive Rate): Gerçek pozitif değerlerin ne kadarının yanlış tahmin edildiğinin ölçüsüdür. Bu hesaplama işlemi aşağıda gösterildiği gibidir.

FPR= FP / (FP + TN)

Yanlış Negatif Değerlerin Oranı (False Negative Rate): Gerçek negatif değerlerin ne kadarının yanlış tahmin edildiğinin ölçüsüdür. Bu hesaplama işlemi aşağıda gösterildiği gibidir.

FNR= FN / (FN + TP)

2.1.8. Tahmin hatası

Karışıklık matrisinin doğruluğunu ve başarısını belirlemek için kullanılan verilerden hatalı tahmin ettiği verilerdir. Formül aşağıda yer almaktadır.

$$\frac{FP + FN}{TP + TN + FP + FN}$$

2.1.9. Doğruluk oranı

Toplam olasılık 1 iken tahmin hatası 1'den çıkarıldığında elde edilir. Doğruluk oranı algoritmanın doğru tahmin ettiği sonuçların oranıdır. Formül aşağıda yer almaktadır.

$$\frac{TP + TN}{TP + TN + FP + FN}$$

2.1.10. Geri Çağırma (Recall)

Pek çok çalışmada söz edilme ve isabet oranı gibi adları bulunan geri çağırma, çalışmadaki pozitif sınıf örneklerinin yaptığı doğru tahmin sayısını verir. Formül aşağıda yer almaktadır.

$$\frac{TP}{TP + FN}$$

2.1.11. Hassasiyet (Precision)

Algoritmada bulunan toplam verinin ne kadarının doğru sonuç verdiği bu metrikte gösterilir ve karışıklık matrisine eklenir. Formül aşağıda yer almaktadır.

$$\frac{TP}{TP + FP}$$

2.1.12. F Skoru

F skoru, kesinlik ve geri çağırma değerlerini içeren istatistiksel hesaplamada veri başarısının ve kesinliğinin bir ölçüsüdür. Harmonik ortalama matematiği olarak kullanılır ve tek sonuç verdiği için kullanışlıdır. Formül aşağıda yer almaktadır.

$$2\frac{Hassasiyet\ .\ \text{İ}sabet\ Oranı}{Hassasiyet + \text{İ}sabet\ Oranı}$$

2.1.13. ROC Eğrisi (ROC Curve)

Sınıflandırıcıdan elde edilen sonuçların performansı özetlenerek bir grafik elde edilir. Öngörülen değerin ne kadar iyi olduğunu hesaplanır. Bu, oluşturulan herhangi bir algoritmanın performansını değerlendirmek için yaygın olarak kullanılan bir yöntemdir. ROC eğrisinin altındaki alan AUC olarak adlandırılır. Başarı oranı, ROC eğrisi altındaki alan arttıkça artar. En yüksek AUC değeri 1'dir. Oluşturulan grafiğin x ekseninde yanlış pozitif oranı ve y ekseninde gerçek pozitif oranı vardır.

2.2. Kullanılan Makine Öğrenmesi Yöntemleri

Makine öğrenimi, çeşitli kitaplıklarda bulunan birçok algoritmaya sahiptir. Yapılan çalışmada kullanılan makine öğrenmesi algoritmaları, Lojistik Regresyon, K-En Yakın Komşu, Destek Vektör Makineleri, Naive Bayes, Rastgele Orman ve XGBoost Model algoritmalarıdır.

2.2.1. Veri setini eğitim ve test verilerine ayırım

Makine öğreniminde yapılan eğitimlerin performansını ölçmek, veri seti eğitim verisi ve test verisi olarak ikiye ayrılmalıdır.

Bu tezde, öğrenme performansı test edilmiştir. Kullanılan veri kümesi, test ve eğitim verileri olarak işlenmiştir. Uygulamada %70 eğitim veri seti kullanılırken %30 test veri seti kullanılmıştır.

2.2.2. K-En Yakın Komşu (KNN)

KNN makine öğrenimi algoritmaları; Evelyn Fix ve Joseph L. Hodges Jr. 1950'lerin başında geliştirildi. 1965, Nilsson, New Jersey'de En kısa mesafe sınıflandırması üzerine yaptığı çalışma YSA algoritmasının geliştirilmesine katkıda bulunmuştur. 1967 yılında bu çalışmalar T.M. Cover ve P.E. Heart'ı içerecek şekilde genişletildi istatistiksel analiz için öneriler uygulanmıştır.

K-en yakın komşu algoritması en çok işletilen algoritmalardan biridir ve kullanılması en basit algoritmalardandır. Bu aynı zamanda tembel bir algoritmadır. Eğitim verilerini yararlanmak yerine ezberlediği için eğitim verilerini öğrenmez. Araştırma sonunda bir tahminde bulunulmalıdır. Bu tahmini bulmak için algoritma, ezberlenmiş komşu verileri arayarak tüm veri setini tarar. K değeri anket tarafından belirlenir ve algoritmanın veri setinde kaç öğeye baktığını gösterir. İstatistiksel bir yöntemdir. Algoritmaya bir veri girildiğinde, bu veriye en yakın K öğe alınır ve işlenir. Bu mesafe Öklid fonksiyonu kullanılarak hesaplanabileceği gibi Manhattan ve Minkowski fonksiyonları da kullanılır.

KNN algoritması K en yakın komşu algoritması olarak da bilinir. Algoritmalar, ML algoritmalarını uygulamak için en yaygın kullanılan ve en kolay olanlardır. Denetimli öğrenme algoritmalarından biridir. KNN algoritmaları, regresyon ve sınıflandırma problemlerinde kullanılır ancak daha çok sınıflandırma problemlerinde kullanılır. KNN algoritması, seçilen veriler noktanın yerleştirildiği sınıf ve en yakın komşu k değerine göre belirlenir. KNN, sınıflandırılan veri ile komşuları arasındaki farktır. Mesafeler hesaplanır, en yakın komşuları bulunur ve sınıfa göre veri alınır. KNN algoritmaları, gürültülü, eski ve naif veri kümelerinden daha verimlidir. Esnektir ve diğer algoritmalardan daha sık kullanılır. Bu algoritmanın dezavantajları da vardır. Örneğin, büyük veri kümeleri için mesafe hesaplamasında çeşitli durumları saklamak için kullanılır. (Kilinc ve ark., 2016: 89-94)

KNN algoritmasındaki ilk adım, optimal k-sınıfı değerlerinin belirlenmesidir. K Parametre değerlerini belirtin. k değeri belirlenirken, verilerin boyutu: Yapı dikkate alındığında. k değeri 1 ile n arasında seçilebilir. k-değeri 5'tir. Bu seçeneği seçmek, bu veri noktasının en yakın 5 komşusunu bulacaktır. Algoritma için k değerini kullan Gerekenden daha büyük bir değer seçmek, birbirine yakın olmayan verileri aynı gruba koyar. Sınıflandırma doğruluğu düşüktür. Sınıf değerleri küçük k değerleridir Bu durumda birbirinden çok uzak olmayan sınıflar ayrı gruplara ayrılır. Bu durumda, sınıflandırma oranı hala düşüktür. KNN algoritmasında Temel amaç, birbirine yakın olan verilerin aynı sınıfa ait olup farklı sınıflara ait olduğunun tanınmasıdır. Yeni verilerin dizideki diğer verilerden uzaklığı. Mesafe, örnekler arasındaki benzerliği

ölçmek için kullanılır. İki örnekleme mesafesini hesaplamanın en yaygın olarak kullanılan Öklid hesabıdır. (Hu ve ark., 2016)

$$dist_{euclindean}(x_1x_2) = \sqrt{\sum_{i=1}^{n}(x_{1i} - x_{2i})^2}$$

Öklid uzaklık formülü dışında hesaplama yöntemleri vardır. Bu yöntemler Manhattan mesafe ölçümü, Minkowski mesafe ölçümü, Chebyshev mesafe ölçümü gibi çeşitli hesaplama formülleri mevcuttur. Minkowski, manhattan ve chebyshev uzaklık hesaplama formülleri aşağıdaki eşitliklerde gösterilmiştir. (Prasath ve ark., 2019: 221-248)

$$dist_{minkowsi}(x_1x_2) = \sqrt{\sum_{i=1}^{n}|x_{1i} - x_{2i}|^2}$$
$$dist_{manhattan}(x_1x_2) = \sum_{i=1}^{n}|x_{1i} - x_{2i}|^2$$
$$dist_{chebyshev}(x_1x_2) = max(|x_{1i} - x_{2i}|)$$

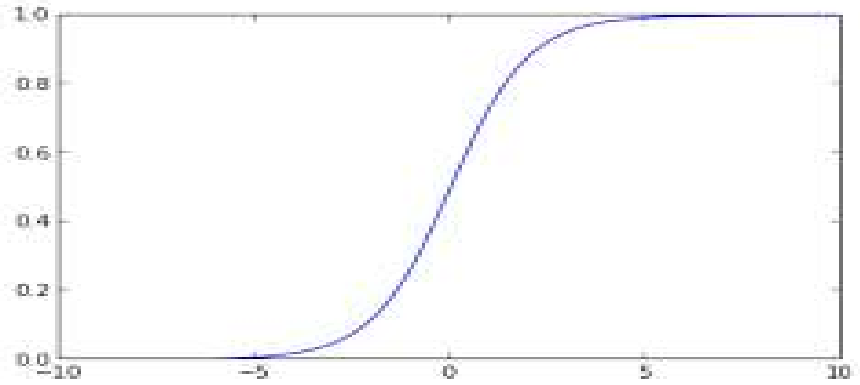
2.2.3. Lojistik Regresyon

Lojistik fonksiyon, nüfus artışının ve oto-katalitik kimyasal reaksiyonların tarihini açıklamak için 19. yüzyılda bulundu. Her iki durumda da w(t) niceliğinin zaman akışı ve büyüme hızı dikkate alınır. Formül aşağıda yer almaktadır. (Cramer, 2002)

$$W(t) = \Omega\frac{\exp(\alpha + \beta t)}{1 + \exp(\alpha + \beta t)}$$

Ω, W'nin üst doygunluk seviyesidir, α, eğrinin x ekseni üzerindeki konumu ve β, eğrinin eğimidir. Lojistik regresyon istatistiksel bir tekniktir ve kategorilerin

sınıflandırılmasında kullanılır. Bu yöntem, ihtimalleri baz alarak bağımsız değişkenler ile regresyonun sonuç değişkenleri arasındaki ilişkiyi inceler ve hesaplar.



Şekil 6. Lojistik Regresyon Eğrisi

Kaynak: Bayramlı, B., (2023), https://burakbayramli.github.io/dersblog/stat/**,** Erişim Tarihi: 02/04/2023

2.2.4. Naive Bayes

Naive Bayes, adını Thomas Bayes adlı bir bilim insanından alan bir sınıflandırma algoritmasıdır. Bu sınıflandırma algoritmasının esası tamamen olasılıksaldır. Olasılıklar kullanılarak gerçekleştirilen hesaplamaları kullanarak bir veri kümesindeki verilerin sınıflandırılmasını temin eder. Naive Bayes tembel bir algoritmadır. Fakat dengesiz ve düzensiz veri setleriyle de çalışır. Veri setindeki her bir öğe için tüm olasılıkları ayrı ayrı hesaplar ve en yüksek olasılık değeriyle sınıflandırır. Ne kadar çok eğitim verisi kullanılırsa o kadar doğru sonuçlar verir ancak daha az veri ile yüksek başarı oranı ile çalışmak mümkündür. Bu algoritma teoreme dayanmaktadır. Formül aşağıda yer almaktadır. (Bayes, 1763: 370-418)

$$P\left(\frac{C_k}{x}\right) = \frac{P\left(\frac{x}{C_k}\right).P(C_k)}{P(x)}$$

$P(x)$:Veri kümesindeki tüm veriler x

$P(Ck)$:k sınıfının olasılığı

$P(Ckx)$:x örneği, k sınıfındandır

$P(xCk)$:k sınıfı olayın x örneğinin oluşumu

Test seti verileri eğitim seti verileriyle eşleşmezse veya test setinde eğitim seti verileriyle eşleşen veri yoksa tahmini bir olasılık hesaplanır ve sonuç olasılık "0" olarak döndürülür. Öneri Sistemleri, Araştırma Analitiği, E-posta Spam Tespiti ve Metin Sınıflandırma. Özellikle gerçek zamanlı tahmin uygulamaları başarı oranı açısından çok daha yüksek sonuçlar sunmaktadır. (Bayes, 1763)

Naive Bayes algoritması Thomas Bayes adlı bilim insanından adını almış bir algoritmadır. (Tapkan, 2011: 247-262)

Olasılıklar kullanılarak gerçekleştirilen hesaplamalar yoluyla bir veri kümesindeki verilerin sınıflandırılmasını sağlar. Bu algoritma, veri setindeki her bir öğe için tüm olasılıkları ayrı ayrı hesaplar ve bunları en yüksek olasılık değerine göre sıralar. NB' de ne kadar çok eğitim veriniz varsa, sonuçlarınız o kadar doğru olur. Veri setindeki veriler küçük olsa bile yüksek bir başarı değeri elde edilebilir. (Datasciencearth, 2022)

Naive Bayes algoritması çeşitli alanlarda kullanılmaktadır. Tahmin, analiz ve sınıflandırma alanlarında kullanılmaktadır. NB' de sınıflandırmanın nasıl yapıldığına bağlıdır. Test setindeki problemin çözümünde kullanılan bilgi (verim) eğitim setinde ise veya tam tersi ise, eşdeğeri yoksa olasılık hesaplamaları ve tahminleri yapılamaz ve sonuç "0" olur. " . (Karakoyun ve Hacıbeyoğlu, 2014: 30-42)

Naive Bayes algoritmasının dezavantajları aşağıda gibi sıralanmıştır;

- Özelliklerin birbirinden tamamen bağımsız olduğu varsayılır, bu nedenle değişkenler arasındaki ilişkiler modellenemeyebilir.

- Bu algoritmayı kullanarak sıfır olasılık problemiyle karşılaşabilirsiniz. Bu durumda örneğin veri setinde hiçbir şeyin olmaması durumu ortaya çıkar. Başka bir deyişle, herhangi bir işlemin sonucu sıfırdır. Bu durumu önlemenin en kolay yolu, bu olasılığı ortadan kaldırmak için tüm verilere bir minimum değer (genellikle 1 değeri) atamaktır. Laplace adı verilen bu yöntem genellikle işe yarar. (Datasciencearth, 2022)

2.2.5. Destek Vektör Makinesi

Destek vektör makineleri, Alexey Chervonenkis ve Vladimir Vapnik tarafından 1963 yılında ortaya çıkarılan istatistiksel öğrenme teorisine dayalı algoritmadır. Vapnik ve ekibi tarafından 1995 yılında daha da geliştirilerek bugünkü haline getirildi. Uygulaması LINE kullanarak yapılır. Gerekirse verileri iki parçaya bölmek için kullanılır. Diğer makine öğrenimi algoritmalarına göre birçok avantajı vardır. (Canbay, 2013)

Çalışmadaki örnek sayısı boyut sayısından az olduğunda kullanışlıdır. Karar mekanizmasında farklı çekirdek fonksiyonlarını kullanabilir. (Canbay, 2013)

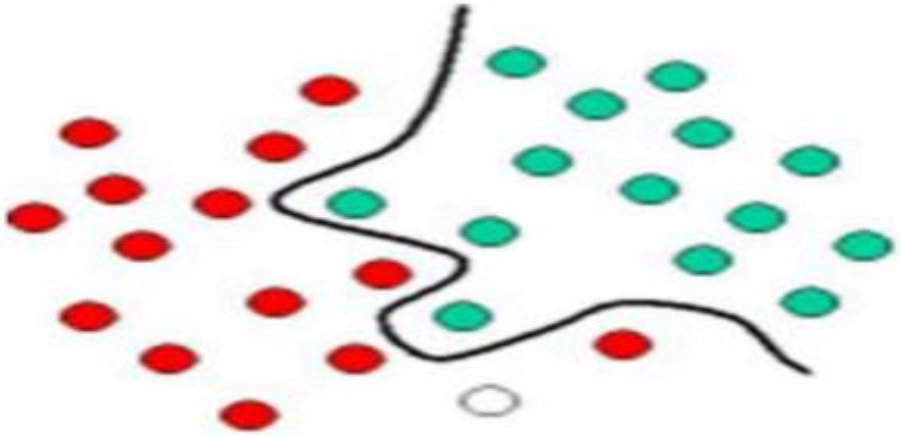
Daha etkili ve başarılı sonuçlar elde etmek için büyük veriyi kullanabilir. Çok sayıda bağımsız değişkeni işleyebilir. (Canbay, 2013)

Veri seti içindeki verilerin doğrusallığına bağlıdır ve iki grupta incelenir. (Canbay, 2013)

2.2.5.1. Doğrusal Olmayan Destek Vektör Makinesi

Doğrusal olmayan DVM, veri kümesi doğrusal olarak ayrılamaz olduğunda ve hata belirli bir hata olmadan belirsiz olduğunda ortaya çıkan bir öğrenme algoritmasıdır. Günlük yaşamdaki olaylarla günlük yaşam arasına düz bir çizgi çekmek imkansızdır. Ayırma eğrileri bu nedenle en önemli sınıflandırma özelliğidir. Arayüz soyulma eğrisi aşağıda gösterilmiştir. İşlevler verilere uygulanır ve temel işlevler adı verilen özelliklerine göre sınıflandırılır. Uygulanan çekirdek işlevinden bir doğrusal çekirdek işlevi aşağıda gösterilmiştir. (Canbay, 2013)

Temel işlevi, ayrılmaz bilgileri izole etmenize ve verilerinizi daha iyi temsil etmenize olanak tanır. (Canbay, 2013)



Şekil 7. Doğrusal Olmayan Regresyon

Kaynak: Ayhan, S., ve Erdoğmuş, Ş. (2014). Destek vektör makineleriyle sınıflandırma problemlerinin çözümü için çekirdek fonksiyonu seçimi. Eskişehir Osmangazi Üniversitesi İİBF Dergisi, 9(1), 175-201.

Lineer çekirdek fonksiyonu:

$$K(x_i, x_j) = (x_i{}^T . x_j)$$

Polinomsal Çekirdek Fonksiyonu:

$$K(x_i, x_j) = (x_i{}^T . x_j)^d$$

Gaussian Radyal Çekirdek Fonksiyonu:

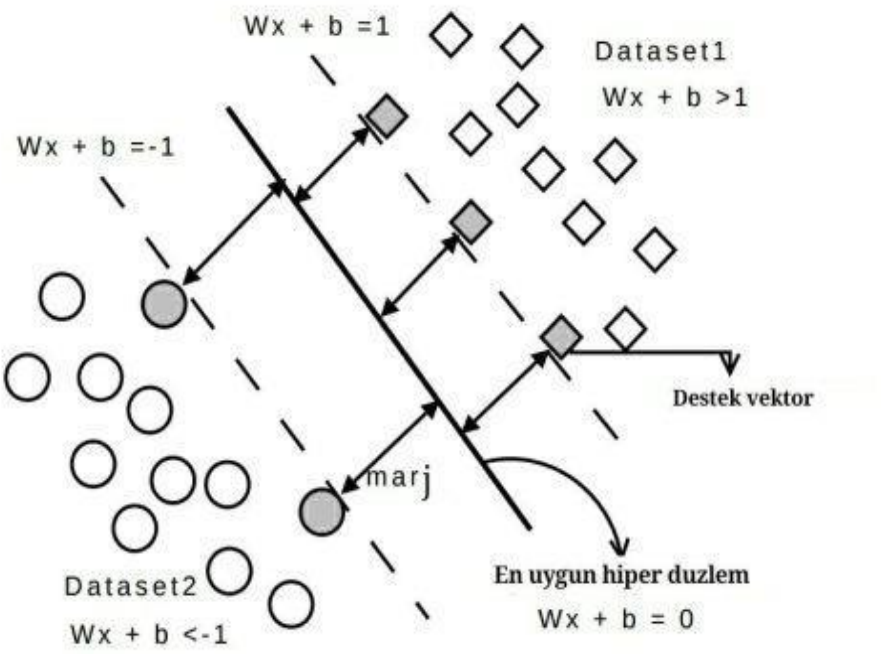
$$K(x_i, x_j) = \exp(\frac{-||x_i - x_j||^2}{2\sigma^2})$$

2.2.5.2. Doğrusal Destek Vektör Makineleri

Doğrusal olarak ayrılabilen veriler sayesinde iki sınıfın birbirinden ayrıştırılabildiği sınır düzleminin doğrusal olduğu makine öğrenmesi algoritması Doğrusal DVM ‘dir. (Canbay, 2013)

$$F(x) = w^T.x + b = \sum_{i=1}^{n} w_i.x_i + b$$

Doğrusal DVM’ de ayrılabilme durumunda veri setlerinin karşılaştırıldığı, marj aralığının belirtildiği destek vektörleri ve oluşan en uygun hiper düzlemi aşağıda verilmiştir. (Canbay, 2013)



Şekil 8. Doğrusal Destek Vektör Makineleri

Kaynak: Görgün, M., (2020), *Makine öğrenmesi yöntemleriyle kalp hastalıklarının tahmin edilmesi* (Master's thesis, Lisansüstü Eğitim Enstitüsü).

2.2.6. Karar ağacı

Karar ağaçları, veri madenciliğinde en yaygın değerlendirilen metotlardan biridir. Diğer sınıflandırma algoritmalarında olduğu gibi karar ağaçlarının birincil

amacı sınıflandırmadır ve karar ağaçları rastgele orman algoritması gibi algoritmalara evrilmiştir. Kayıtların öznitelikleri düğümlerdir ve bu düğümler doğru-yanlış-evet-hayır sorularını yanıtlar. Bu şekilde, her düğüm için veriler iki bölüme ayrılır. Bölünmüş özellik verilerini etkileyen özellik vektörleri incelenir, muvaffakiyet oranı ve bilgisi en etkili olan düğümler dallanma algoritmasına girer. Dallanmadan sonra, algoritma durmaz ve bütün veriler sınıflandırılana kadar devam eder. Karar ağaçları, olasılık ve istatistiğe dayalı algoritmalardır. Karmaşıklık olarak da adlandırılan entropi, istatistikte çok önemlidir. Entropi, beklenmeyen bir olayın meydana gelme olasılığıdır veya algoritmanın olası sapmasının tahmin edilmesi gerekir. Entropinin formülü:

$$H(x) = -\sum S(x) log S(x)$$

Entropi formülünde, S(x) belirli bir sınıfa ait grupların oranını tanımlar ve H(x), logaritmik çarpımlarının integrali alınarak bulunur. Burada H(x), bu grubun kendisinin entropisidir.

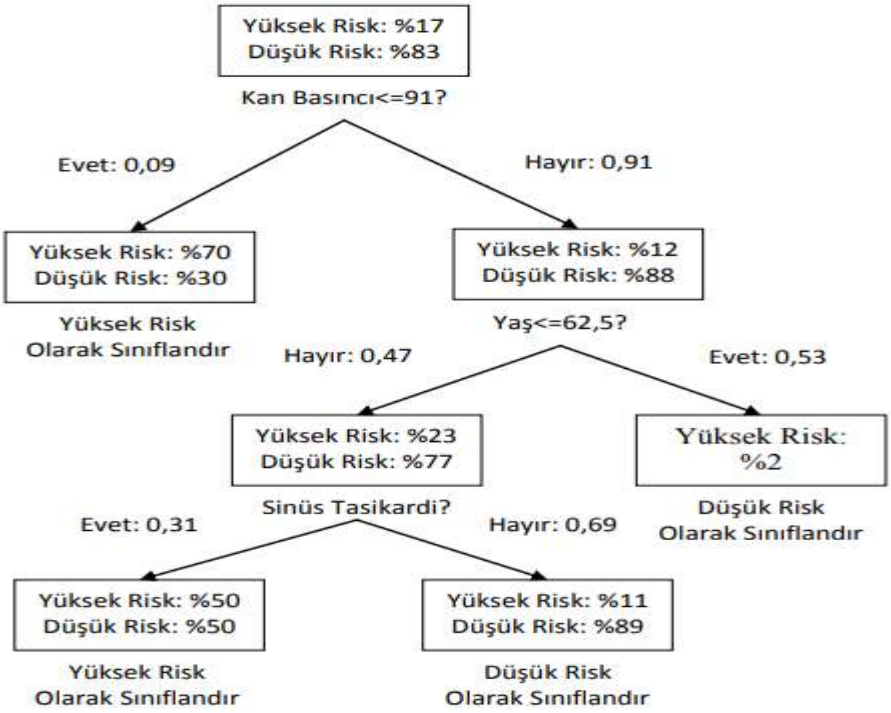
$$Kazanc(S, D) = H(S) - \sum_{V \in D} \frac{|V|}{|D|} \mathrm{H(V)}$$

Karar ağaçlarında önemli bir nokta da belirleyici bilgi edinimidir. Bilgi kazancı, entropi değeri, alt grubun değeri ile bütün veri setinin değeri arasındaki farktır. Bir yukarıdaki formülde açıklanmıştır. Burada S, tüm veri kümesidir ve D, S'nin parçalanmış bir alt bölümüdür. V, D'ye karşılık gelen karar mekanizmasıdır.

Karar ağacı algoritması, veri madenciliğinde, regresyon ve sınıflandırma problemlerine uygulanabilen, hesaplama açısından yoğun ve bilgi teorisi ilkelerine dayanan sınıflandırma algoritması tekniklerinden biridir. Bu algoritmanın çalışma mantığı (karar ağacı), çok büyük bir veri setini daha büyük veri setinin koyduğu kurallar çerçevesinde uygulayarak daha küçük veri setlerine bölmek üzere tasarlanmış bir yapıya sahiptir. (Tapkan, 2011: 240).

Karar ağacı algoritmaları, bir kümeleme algoritması kullanarak girdi verilerini yinelemeli olarak kümeler. Yani gruplara ayrılırlar. Gruplandırma işlemi, yeni kümelenmiş grubun tüm öğeleri aynı sınıf etiketine sahip olana kadar devam eder. Entropi ile ilgili sınıflandırma ağaçları (ID3, C4.5) ve regresyon ağaçları en sık kullanılan algoritmalardır. Karar ağacı yapısını oluşturan algoritma, ağaç yapısını oluşturur ve karar ağacını en üstteki kökten okumaya başlar. Bu ilk başlangıç düğümüne kök kök denir. Farklı algoritmalar, farklı ağaç yapıları ve sınıflandırma sonuçları verir. Karar ağaçlarına dayalı bir dizi algoritma geliştirilmiştir ve bu algoritmalar birçok yönden farklılık göstermektedir. Bunlar kök düğüm, düğüm ve bölme kriterleri seçiminde izlenen yollar olarak sayılabilir. (Tapkan, 2011: 189)

Karar ağacının en üst kök düğümü olarak, öznitelik değerleri kökte ayrılmayı ve aşağı doğru dallanmayı öğrenir. Her dal bir karar kuralını temsil eder ve her yaprak düğüm bir sonucu temsil eder. Bu şekilde, karar ağacı akış şemaları karar vermede faydalıdır. Karar ağaçları, çeşitli yapılarda oluştukları için anlaşılması ve yorumlanması kolaydır. Karar ağacının amacı, veri kümesindeki düğüm sayısını en aza indirgemek, sonuca ulaşmak için en kısa yolu kullanmak ve yüksek başarı elde etmektir.



Şekil 9. Karar Ağacı Örneği

Kaynak: Halis, Y., (2022).Makine Öğrenmesi Yöntemleri İle Kalp Krizi Tahminleme Beykent Üniversitesi Yüksek Lisans Projesi

2.2.7. Extreme Gradient Boosting

XGBoost, Friedman (2001) tarafından geliştirilen GB tabanlı bir algoritmadır. Hem XGBoost hem de GB, gradyan artırma teknikleri kullanır, ancak XGBoost performansı artırmak için daha düzenli bir model şekli kullanır.(Carmona, Climent ve Momparler, 2019)

Bu prosedürler artık spam tespiti, reklam eşleştirme sistemleri, dolandırıcılık tespit sistemleri ve anormal olay tespit sistemleri gibi fizikte kullanılmaktadır.(Chen ve Guestrin, 2016: 785-794)

Bu yöntemlerin önemli avantajı, tahminin veri odaklı olmasıdır. Regresyon ve sınıflandırma ağacı genişletme algoritmaları kullanılarak etkili sonuçlar elde edilir.

XGBoost yalnızca başarılı tahminler sağlamakla kalmaz, aynı zamanda birçok başka tahmin de sağlar.

XGBoost, kaynakları iyi kullanmak ve önceki gradyan yükseltmelerinin sınırlamalarının üstesinden gelmek için tasarlanmış, yüksek düzeyde ölçeklenebilir, esnek ve çok yönlü bir araçtır. XGBoost ile LightGBM ve CATBoost arasındaki temel fark, fazla uydurmayı kontrol etmek için yeni bir düzeltme yönteminin kullanılmasıdır. Bu nedenle model uydurma sırasında daha hızlı ve daha güçlüdür. (Al Daoud, 2019: 6-10)

XGBoosting Architecture Extreme Gradient Boosting Algorithm, yaygın olarak kullanılan, oldukça verimli, esnek bir şekilde tasarlanmış ve optimize edilmiş bir gradyan artırma kitaplığıdır. Makine uygulama algoritmalarının uygulanmasında gradyan geliştirme kitaplığına dayanır.

Hesaplama, makine öğreniminden daha kolaydır. Bu özellik sayesinde çok boyutlu büyük veri analizi için etkin bir şekilde kullanılabilir. XGBoost'a benzer algoritmalar yarışmayı kazandı. XGBoost yöntemini kullanma Rekabet sorunlarına örnek olarak mağaza satış tahminleri, yüksek enerji verilebilir. Fiziksel olay sınıflandırması, web metni sınıflandırması, müşteri davranışı tahmini, hareket kapsamlı çevrimiçi kurs tespiti, reklam tıklama oranı tahmini, kötü amaçlı yazılım sınıflandırması, ürün sınıflandırması, tehdit riski tahmini, terk tahmini olarak belirtilir.(Chen ve Guestrin, 2016)

Bu başarı, bu yöntemlerin bilimsel açıdan ilgi çekici olmasından kaynaklanmaktadır. Bu yöntem, girdinin bilgi kazancını temsil eden bir değişken alır. Değişkenlerin çıktı değişkenleri üzerindeki etkisi belirlenebilmektedir.

XGBoost, Chen tarafından 2016 yılında önerildi. Sınıflandırma ve Regresyon Ağaçlarına (CART) dayalıdır, bölüm özniteliklerini yeniden tanımlar ve bölüm özniteliklerini belirlemek için kayıp işlevinin en aza indirilmesini kullanır. Bu yöntem, güç yüklerini tahmin etmek, petrol fiyatlarını tahmin etmek ve kazaları tahmin etmek

için başarıyla kullanılmıştır. Doğruluğu diğer modellere göre daha yüksek ve eğitim maliyeti diğer modellere göre daha düşüktür. XGBoost, kayıp fonksiyonunu en aza indirmek için en iyi fonksiyonu belirler, kaybı en aza indirir ve şu şekilde tanımlanır:

$$Obj(\theta) = \sum_{i=1}^{n} L(y_i, \hat{y}_i) + \sum_{k=1}^{k} \Omega\left(f_k\right)$$

K'inci ağaç eklendiğinde, önceki ağaç eğitilmiştir. Eğitim hatası ve önceki ağacın normal süresi sabit hale gelir. Kayıp işlevi aşağıdaki gibi yazılır:

$$Obj(\theta) = \sum_{i=1}^{n} L(y_i, \hat{y}_i^{t-1} + f_t(x_i)) + \gamma^T + \frac{1}{2}\lambda \sum_{j=1}^{T} L({w_j}^2 + c)$$

Kayıp fonksiyonunun Ortalama Kare Hata fonksiyonu olduğunu ve ikinci dereceden Taylor açılımı olduğunu varsayarsak sonuç, aşağıdaki denklem ile ifade edilebilir.

$$Obj^t(\theta) \approx \sum_{i=1}^{n} [g_i f_t(x_i) + \frac{1}{2} h_i f_t(x_i)^2] + \gamma^T + \frac{1}{2}\lambda \sum_{j=1}^{T} {w_j}^2$$

2.2.8. LigbtGBM

Light Gradient Boosting Machine algoritması, bir karar ağacı altyapısı kullanan hızlı, yüksek başarımlı bir gradyan artırma çerçevesi olarak tanımlanabilir. Karar ağacı altyapısına sahip diğer algoritmaların aksine, ağaç LightGBM algoritmasında dikey, ancak diğer tüm algoritmalarda yatay olarak büyür. Bilhassa sınıflandırma ve sıralama için kullanılabilen bir yapıdır. Aslında altında yatan ve geliştirilen fikir XGBoost yöntemidir. LightGBM, XGBoost algoritmasının eğitim süresi başarımını iyileştirmek için Microsoft tarafından geliştirilmiş bir gradyan artırma altyapı yöntemidir. LightGBM Mimarisi Son yıllarda veri miktarı arttıkça, LightGBM hız açısından geçmişteki ve günümüzdeki veri bilimi algoritmalarından daha iyi performans göstermekte ve sınıflandırma amacıyla kullanılmaktadır. Büyük veri setleriyle kullanıma uygundur ve daha küçük veri setleriyle kullanılır, fakat etkili sonuçlar vermeyebilir. Uygulanması kolay olan LightGBM'nin parametreleştirilmesi

zordur. Çünkü toplamda 100'e yakın parametre var. Parametreler arasında en yüksek başarı oranını kazanmak için tahmin yapılmalıdır.

2.2.9. Bagging

Bagging yöntemi ayrıca gradyan artırma yöntemini kullanır. Bagging yönteminde, modele giren veri setinden çıkarılan eğitim verileri işleterek yeni eğitim verileri üretilir ve eğitim birçok kez tekrarlanır. Yeni eğitim verileri, rastgele seçilen yinelemeli bir eğitim setinden elde edilir. Eğitim veri setinden alınan örnekler, eğitim veri setine geri gönderilir. Bagging yöntemi, aşırı eğitime ve model performansını geliştirmeye yönelik yeni bir yaklaşımdır. Torbalama, önyükleme toplama anlamına gelir. Ağaç yapısına sahip olduğu için topluluk öğrenme yöntemine aittir. Bagging yöntemi, bakımı kolay bir algoritmadır. Bagging yöntemi, sepet algoritması bağlandığında ortaya çıkar. (Breiman, 1996: 123-140)

Bagging ve boosting arasındaki en büyük fark, ağaçların birbirine bağımlı olup olmadığı ve optimizasyon hatalarının önceki ağaçlardan toplanıp toplanmadığıdır.

2.2.10. Rastgele Orman

Random Forest (RO) da bir KA topluluğudur. Ağaçların yinelemeli olarak oluşturulduğu Gradient Enhanced Trees'in (GBT) aksine, RO birden çok ağacı paralel olarak eğitir. Her ağaç, eğitim verilerinden önyükleme örnekleri üretilerek oluşturulan farklı bir eğitim seti kullanır. Ayrıca, her düğümde en iyi özniteliği seçerken, KA'daki gibi tüm öznitelikler değil, düğümde bulunan tüm özniteliklerin yalnızca bir alt kümesi dikkate alınır. Bu nedenle, RO hem rastgele örneklemeyi hem de özellik seçimini kullanır. Yeni bir örnek tahmini oluşturmak için, RO her ağaçtan tahminlerin ortalamasını alır. Özetle, hem GBT hem de RO, temel model olarak karar ağaçlarını kullanır, ancak eğitim süreci farklıdır ve her yöntemin güçlü ve zayıf yönleri vardır. MLlib hem GBT'yi hem de RO'yu destekler. Makine öğrenimi (ML) teknikleri, özellikle rüzgar hızı/gücü tahmini olmak üzere rüzgar endüstrisinde popülerlik kazanıyor.

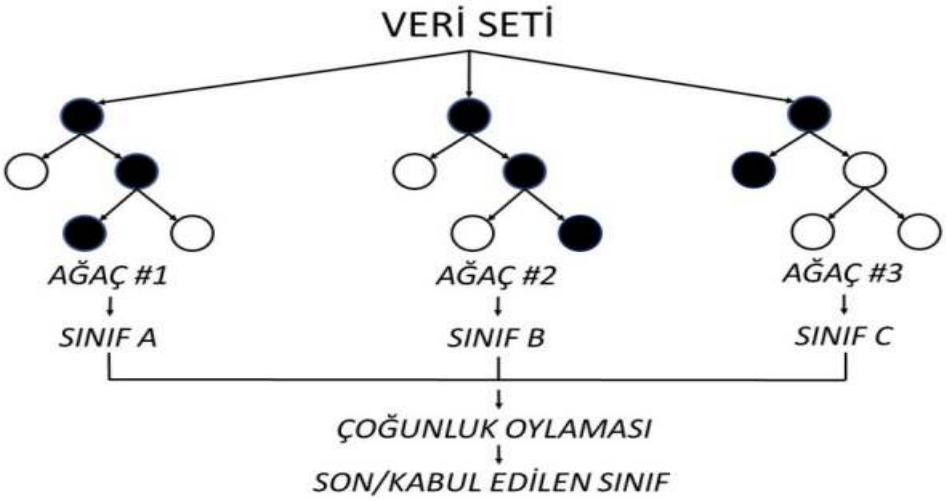
Genel olarak tahmin ve istatistiksel modelleme için makine öğreniminin artan kullanımı, diğer geleneksel tekniklerin kullanılmasındaki zaman boşluğundan kaynaklanmaktadır. Random Forest (RO), rüzgar tahmini amaçları için son derece umut verici bir makine öğrenimi modelleme tekniğidir.

18'e kadar girdi değişkeni ile 1 saatlik ileri rüzgar kuvvetini tahmin ettik ve RO modellemenin gürültülü ve ilgisiz girdilere karşı dayanıklı olduğunu gösterdik. (Lahouar ve Ben Hadj Slama, 2017: 1040-1051)

Kapsamlı özellik seçimi gerçekleştirmek ve saatlik rüzgar gücü tahminlerinin doğruluğunu tek bir algoritmik modelde sıfıra çıkarmak için bir topluluk modelleme çerçevesinin bir bileşeni olarak RO'yu kullandı. (Feng, 2017)

Ultra kısa vadeli rüzgar enerjisi üretimini tahmin etmek için Niche Immunity Lion algoritması tarafından optimize edilmiş hibrit dalgacık ayrışımı ağırlıklı RO modeli kullandık. Geliştirilen model, tipik RO modellerinden, sinir ağlarından ve destek vektör makinelerinden daha iyi performans gösterdi. Burada, ızgarayı (rüzgar hızını tahmin etmek için) ve rüzgar gücünü tahmin etmek için optimize edilmiş RO'yu birleştiren hibrit bir model kullandı. (Niu, Pu ve Dai, 2018)

Rüzgar hızını tahmin etmek için RO modellerinde kullanılan çeşitli atmosferik girdi özelliklerinin kullanışlılığını test etti. Rastgele Orman Algoritması Rastgele Orman Algoritması. Her karar ağacını çoklu karar ağaçlarında gözlenen farklı örneklerle eğiterek farklı modeller oluşturmak ve sınıflandırma yapmak için kullanılabilir. Kullanım kolaylığı ve esnekliği, hem sınıflandırma hem de regresyon problemlerini çözmesini sağlayarak benimsenmesini ve yaygınlaşmasını hızlandırmaktadır. (Vassallo ve ark., 2021: 295-309)



Şekil 10. Rastgele Orman Veri Seti Örneği

Kaynak: Ozturk, M., (2023), https://miracozturk.com/python-ile-siniflandirma-analizleri-rastgele-orman-random-forest-algoritmasi/, Erişim Tarihi: 02/04/2023

Gerekli Algoritma:

• Veri setini analiz için hazırlanması gerekir. (analiz için veri setini oluşturulur ve gerekirse veriler temizlenir)

• Algoritma, her örnek için bir karar ağacı oluşturur ve her karar ağacı için bir tahmin elde eder.

• Tahmin sonucunun her değeri için düzeltmeler yapılır (sınıflandırma problem modu, regresyon problemi ortalaması).

• Son olarak, algoritma sonucu üretmek için nihai tahmin için en değerli değeri seçer.

DÖRDÜNCÜ BÖLÜM

UYGULAMA

1. VERİ SETİ VE ÖZNİTELİK

Bu çalışmada 768 kişinin çeşitli özelliklerini içeren Kaggle Veri Seti kullanılmıştır. Uygulama Python'da yazılmıştır. Çalışma 16 GB RAM, 4 GB ekran kartı 7 8. nesil işlemciye sahip bir laptop üzerinde yapılmıştır. Veri görselleştirme için Matplotlib ve Seaborn kütüphaneleri kullanıldı. Ayrıca çalışmada kullanılan algoritmalar için scikit-learn kütüphanesinden yararlanılmıştır.

Makine öğrenimi algoritmaları kullanılarak bağımsız olarak işlendi. 768 veriden %70 verisi eğitim için girilmiştir ve geri kalan %30 veri test için kullanılmıştı. Python'da makine öğrenimini kullanan diyabet datası üzerine Rasgele Orman ve KNN algoritması kullanılarak en yüksek başarı oranı elde edilmiştir. Mevcut araştırma, farklı algoritmalar kullanarak bu değeri artırmayı ve doğruluğu iyileştirmeyi amaçlamaktadır.

1.1. Veri Seti

Veri setinde toplam 9 özellik bulunmaktadır ve bu özelliklerin ana kullanımlarının sonuçları ve nerede kullanıldıkları açıklanmıştır. Nitelikler aşağıda listelenmiştir.

Tablo 1. Öznitelikler Tablosu

Öznitelik	Öznitelik Bilgisi
Age	Kişinin Yaşı
Pregnancies	Kişinin Hamilelik Bilgisi
Glucose	Kişinin Glikoz Bilgisi
BloodPressure	Kişinin Kan Basıncı Bilgisi
SkinThickness	Kişinin Cilt Kalınlığı Bilgisi
BMI	Kişinin Vücut Kitle Endeksi
İnsülin	Kişinin İnsülin Bilgisi
DiabetesPedigreeFunction	Kişinin Genetik Bilgisi
Outcome	Çıkış-Sonuç Outcome:0 (Diyabet Hastası Değil) Outcome:1 (Diyabet Hastası)

1.2. Veri Setlerine Uygulanan Ön İşlemler

Bir veri setinde veri hazırlama ve ön işleme adımları, veriyi en doğru şekilde kullanmak ve veriden en iyi şekilde yararlanmak için kullanılan yöntemlerdir. Veri kümelerinde toplanan veriler eksik, tutarsız veya güncelliğini yitirmiş olabilir. Herhangi bir makine öğrenimi algoritmasına başlamadan önce, veri ön işleme adımlarını uygulamamız ve verileri optimize etmemiz gerekir. Yüksek başarı oranları gerektiren yapay zeka sistemleri için veri kalitesi oldukça önemlidir. Bu, doğru ve güvenilir bir veri ön işleme adımından sonra mümkündür.

1.3. Eksik Verilerin Bulunması

Özellikle büyük veri setlerinde eksik olan veriyi bulmak ve eksik veriyi eklemek ya da veri setinden eksik veriyi çıkarmak önemlidir. Bu bağlamda kullanılan veri setlerinin incelenmesi, istatistiksel günlük ve diyabet hastalığı veri setlerinde eksik veri mevcuttur. Sıfır sayısal veriler ise Nan olarak tanımlanmıştır

1.4. Standardizasyon

Makine öğrenimi algoritmaları, veriler arasındaki mesafeyi ölçmek için mesafe ölçümlerini kullanır. Verilerin sonuçlara eşit olarak katkıda bulunabilmesi için, onları aynı birime getirmemiz gerekir. Veriler farklı boyutlardayken mesafe hesabı yaparsanız düşük değerlere sahip veriler varyansı artıracak ve aynı etkiyi yaratmayacaktır. Standardizasyon, uygun sütundaki veri seti özelliklerinin ortalaması çıkarılarak ve standart sapmaya bölünerek belirlenir.

1.5. Normalizasyon

Sayısal bir değişkenin değerlerini daha dar bir alana sıkıştırma işlemine normalleştirme işlemi denir. Normalleştirme aykırı değerleri önler. Normalleştirme işlemi, veri kümesindeki değerleri 0 ile 1 arasında yeniden ölçeklendirir. Minimum-maksimum ölçeklendirme olarak da adlandırılır.

1.6. Öznitelik Kullanımı ve Korelasyon Grafiği

1.6.1. Yaş

İnsanlar ihtiyarladıkça bedende çeşitli problemler ile karşılaşılmaktadırlar. Bu nedenle, diyabet sorunlarıyla karşılaşılabildikleri için yaş öznitelik olarak belirlenmiştir.

1.6.2. Hamilelik

Diyabetik gebe ile diyabetik olmayan gebe arasındaki en mühim ayrım, diyabette ketoasidoz riskinin fazlalaşmasıdır ayrıca mevcut insülin direnci de tabloyu daha ağırlaştırabilmektedir. Gebelerde insülin ihtiyacı, gebe olmayanlara oranla daha fazladır. Diyabetik gebelerde gebelik öncesine göre bilhassa ilk 3 aydan sonra insülin ihtiyacı artmıştır. Diyabetik gebelerde organizmanın artmış glikoz ile insülin ihtiyacının dengelenmesi hem anne hem de çocuk açısından önemlidir. Aksi taktirde bebeklerde kalp iskelet sistemi anomalileri, gebe kadında ise erken doğum, düşük, iri doğum veya ölü doğum riski artmaktadır.(Türk Diyabet Vakfı, 2023)

1.6.3. Glikoz

Karbonhidratlar, yağlar ve proteinler beslenmemizin esasını oluşturan üç asıl besin grubudur. Glikoz, karbonhidratların yapı taşıdır. Kelime Yunanca tatlıdır ve vücudumuz için ana enerji kaynağıdır. Yaygın olarak 'dekstroz' olarak adlandırılan üzümler başta olmak üzere meyveler açısından zengindir. Fizyolojik aktivitemizin sağlıklı bir şekilde sürdürebilmesi için yeterli miktarda karbonhidrat tüketmemiz gerekir. Un, patates, pirinç, fındık ve meyve içeren yiyecekler karbonhidrat bakımından yüksektir. Karbonhidrat içeren besinler, sindirim sisteminden geçerken çeşitli enzimatik çalışmalarla parçalanır ve bağırsaktan emilip kana karışan glikoza dönüştürülür. Bu nedenle “kan şekeri” olarak da adlandırılır. Kanın taşıdığı glikoz, tüm doku ve organlarca enerji kaynağı olarak kullanılır. Ortamda ihtiyaç duyduğundan daha fazla glikoz bulunduğunda, ihtiyaç duyulduğunda kullanılan ve karaciğer ve kaslarda depolanan daha kompleks karbonhidratlara dönüştürülür. Sağlıklı kişilerde pankreas tarafından salgılanan insülin ve glukagon hormonları sayesinde kandaki glikoz miktarı belirli bir aralıkta tutulur. Besinlerle alınan ve kana karışan glikoz kan şekerini yükseltir. İnsülin sayesinde kandaki glikoz hücreler tarafından alınır ve kanda yüksek glikoz birikmesini engeller. Kan şekeriniz düşer. Kan şekerini normal sınırlar içinde tutmak için glukagon hormonu salgılanır. Bu şekilde karbonhidratlar glikoza dönüştürülür, depolardan salınır ve kana karışır. İnsülin ve glukagon hormonları kan şekerinin fazla yükselmesini veya düşmesini engelleyerek glikoz seviyelerini düzenler. (Medikal Park, 2023)

Kan şekeri seviyesinin belirli bir aralıkta olmasına “normal kan şekeri”, daha düşük olmasına “hipoglisemi”, yüksek olmasına ise “hiperglisemi” denir. Kan şekeri düzeylerini belirleyen aralıklar, Uluslararası Diyabet Federasyonu (IDF) ve Amerikan Diyabet Derneği (ADA) gibi kuruluşların tanı ve tedavi standartlarına dayanmaktadır. Uluslararası tanı kriterlerine göre sağlıklı bir insanda açlık kan şekerinin 70-100 mg/dl'nin, tokluk kan şekerinin ise 140 mg/dl'nin altında olması gerekir. (Medikal Park,2023)

1.6.4. Kan Basıncı

Kan basıncınızı sağlıklı bir aralıkta tutmak önemlidir. Bu, vücudun daha fazla Komplikasyon gelişme şansınızı azaltır. Diyabetli kişilerde kan basıncı 140/80 mm Hg’nin altında, böbrek, göz hastalığınız, kan damarlarını ve beyne kan akışını etkileyen herhangi bir durumunuz varsa 130/80 mm Hg’nin altında olmalıdır. Bunun nedeni kalbinize, gözlerinize, böbreklerinize ve diğer organlarınıza uygulanan artan baskıdır. (Çelik, 2023)

1.6.5. Cilt Kalınlığı

Yapılan araştırmalar sonucunda; kan şekeri seviyesi oldukça yüksek olan kişilerin kan şekeri normal seviyede olanlara göre daha sık ve şiddetli deri değişikliği yaşadıkları sonucuna varılmıştır. Kan şekerinin yükselmesi ile beraber deri hücrelerinde ve derinin bağ dokusunda karmaşık bozukluklar oluştuğu için deri rahatsızlıklarının ortaya çıkması da muhtemeldir. Dolayısı ile kan şekerinin yüksek olması, kontrol altına alınmaması gibi durumlar deri rahatsızlıklarını tetikler. (Çelik, 2023)

1.6.6. Vücut Kitle Endeksi

Şeker hastalığı da dahil olmakla birlikte türlü hastalıklarla obezite ilişkisi, bilimsel olarak ispatlanmış bir sorundur. Günümüzde küresel bir salgın haline dönüşen şeker hastalığı en yaygın olanıdır. Vücut Kitle İndeksi (BKİ) 30 kg / m2'den yüksek olan insanlar Tip 2 diyabetlilerde görülmektedir. (Vural, 2023)

1.6.7. İnsülin

Besinler sindirime uğradıktan sonra bedenimizde yer alan enzimler sayesinde şekere parçalanır. Şeker (glikoz) kan akışı ile bedenin tüm alanlara taşınır. Vücudumuzun ana besin kaynağı olan şeker, enerji elde edebilmek için kandan vücut hücrelerine (kas hücreleri, yağ hücreleri ve karaciğer hücreleri)dağılmaktadır. İnsülin, bedenimizde midenin altında ve arka tarafında bulunan pankreas adındaki organın, beta hücrelerinden salgılanan hormondur. Kandaki şekerin kandan ayrılarak hücreye dahil olmasını sağlar. Böylece, kandaki şeker düzeyi de artmamış olacaktır. Diyabetli olmayan bir bireyde her gıda alımı sonrası, pankreas alınan besinlerin enerji haline dönüşmesini sağlamak için insülin üretir. Bu demektir ki tüm bireyler insüline tabidir. Diyabetlilerde ise, pankreas yeterli düzeyde insülini üretmez veya üretilen insülin hedef hücreler (kas, yağ ve karaciğer hücreleri) tarafından kullanılmaz. Bu durumda bedenimiz için hayati değere sahip olan insülini dışarıdan vücudumuza sağlamamız gerekmektedir. (Türk Diyabet Vakfı, 2023)

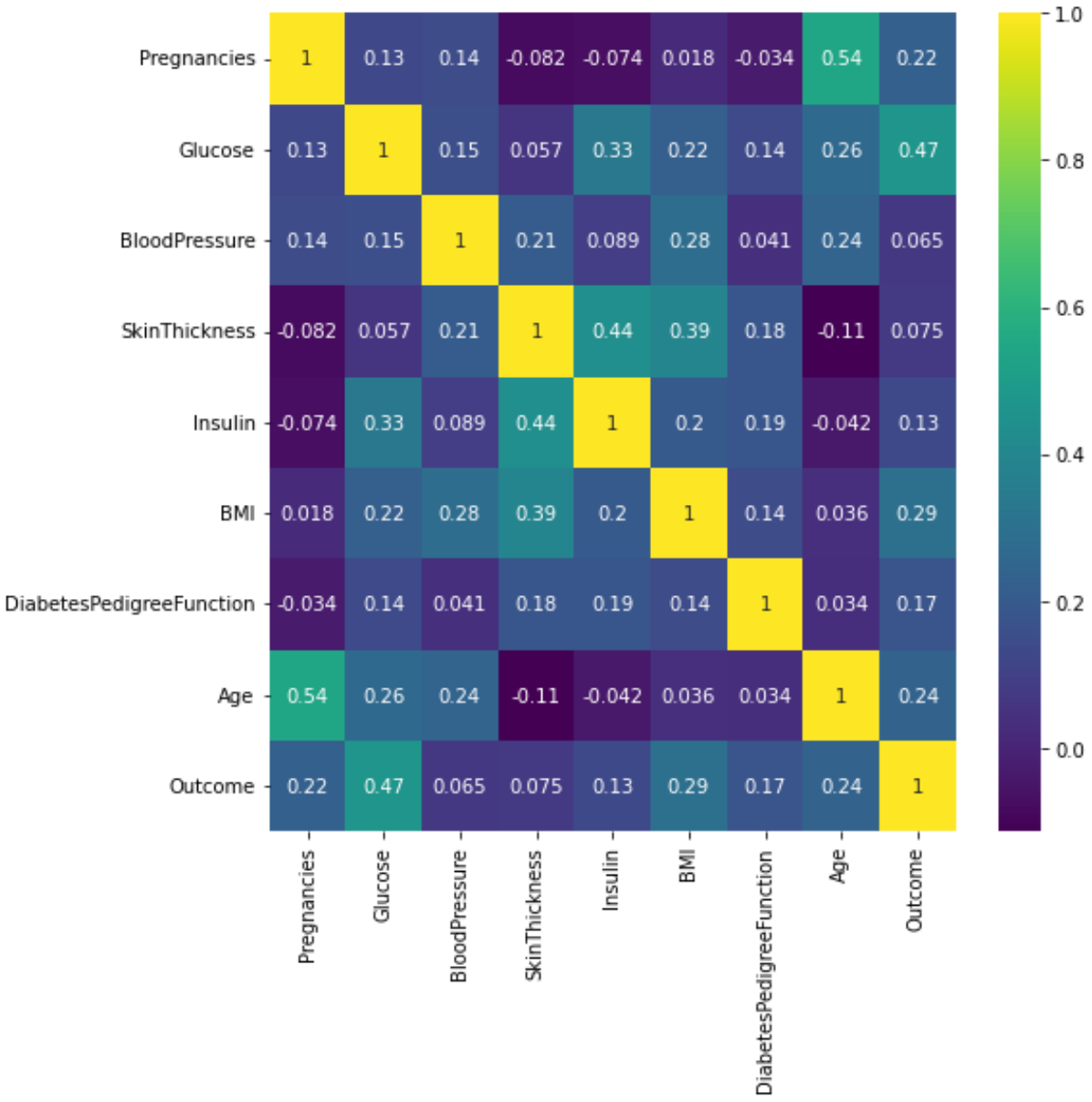
1.6.8. Kalıtım

Şeker hastalığında genetik eğilim önemli bir unsurdur. Eğer anne ve babadan biri 50 yaşından sonra Şeker hastalığına yakalanmışsa risk yüzde 7, daha erken yaşta Şeker hastalığına yakalanmışsa risk yüzde 15'i buluyor. Hem anne hem baba diyabetik ise risk yüzde 50'ye çıkmaktadır. Ancak sağlıklı yaşam önlemleri ile bu riski aşağı çekmenin olası olduğu görülmektedir. (Acıbadem, 2023)

1.6.9. Korelasyon Grafiği

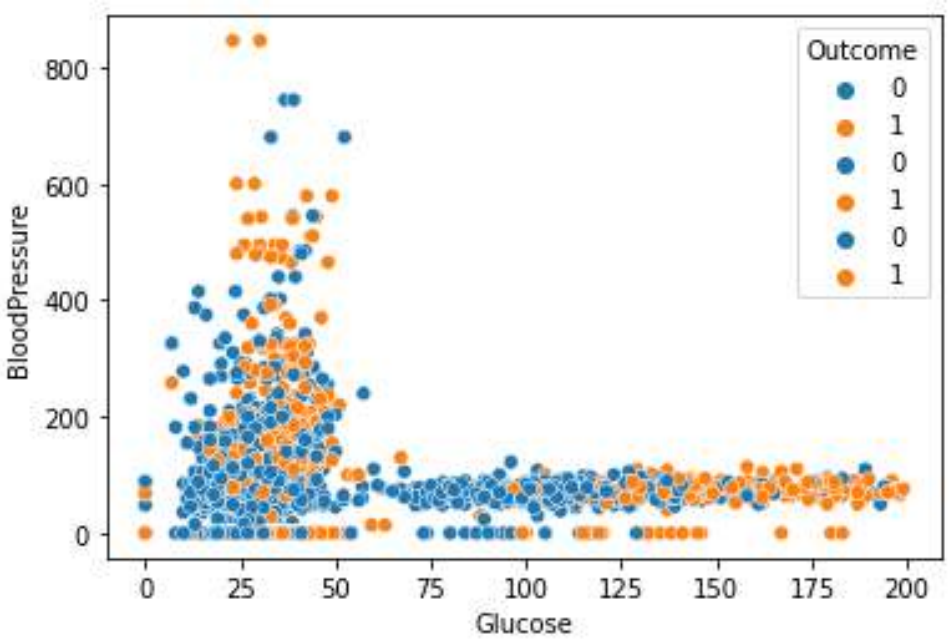
Korelasyon Nedir? Korelasyon -1 ve 1 arasında değişen oranlardır. 1'e yaklaştıkça pozitif doğrusal ilişki olduğu görülür(Doğru Orantı).1'den 0 a yaklaştıkça ilişki pozitif doğrusal ilişki bozulmaktadır. Değerler -1 ile 0 arasındayken ters ilişki mevcut olduğu anlamına gelmektedir(ters orantılı). Aşağıdaki ekran görüntüsünde outcome kolonu şeker hastalığı olup olmadığını belirtmektedir. Örneğin; Glikoz 0.45 iken 1'e yakın olduğundan pozitif doğrusal ilişki mevcut anlamına gelmektedir. Bu durum da glikoz arttıkça şeker hastası olma ihtimalinin arttığını ortaya koymaktadır.

Tüm veri setindeki bilgilerin birbiriyle olan korelasyon bağlantıları aşağıdaki görselde yer almaktadır.



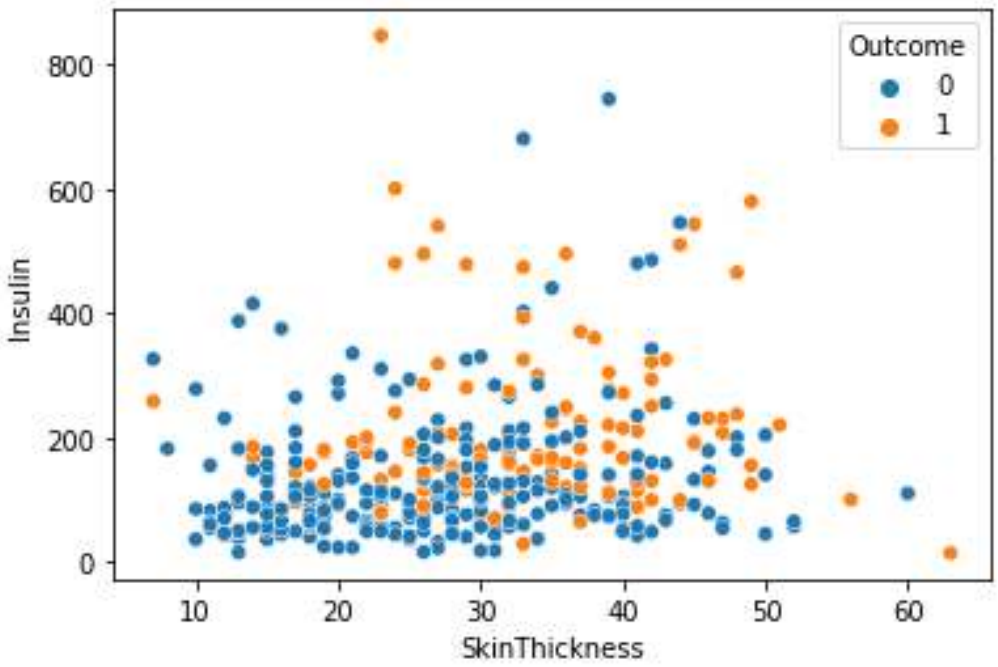
Şekil 11. Tüm Veri Setindeki Bilgilerin Birbiriyle Olan Korelasyon Bağlantıları

Glikoz ve kan basıncı arasındaki ilişki: Glikoz 0 ile 70 seviyelerindeyken ve kan basıncı 0 ile 600 seviyelerindeyken şeker hastalığı olma riski artmaktadır. Glikoz 100 ile 200 seviyelerindeyken ve kan basıncı 0 ile 100 seviyelerindeyken şeker hastalığı olma riski artmaktadır. Aşağıda glikoz ve kan basıncı arasındaki ilişkiyi gösteren grafik yer almaktadır.



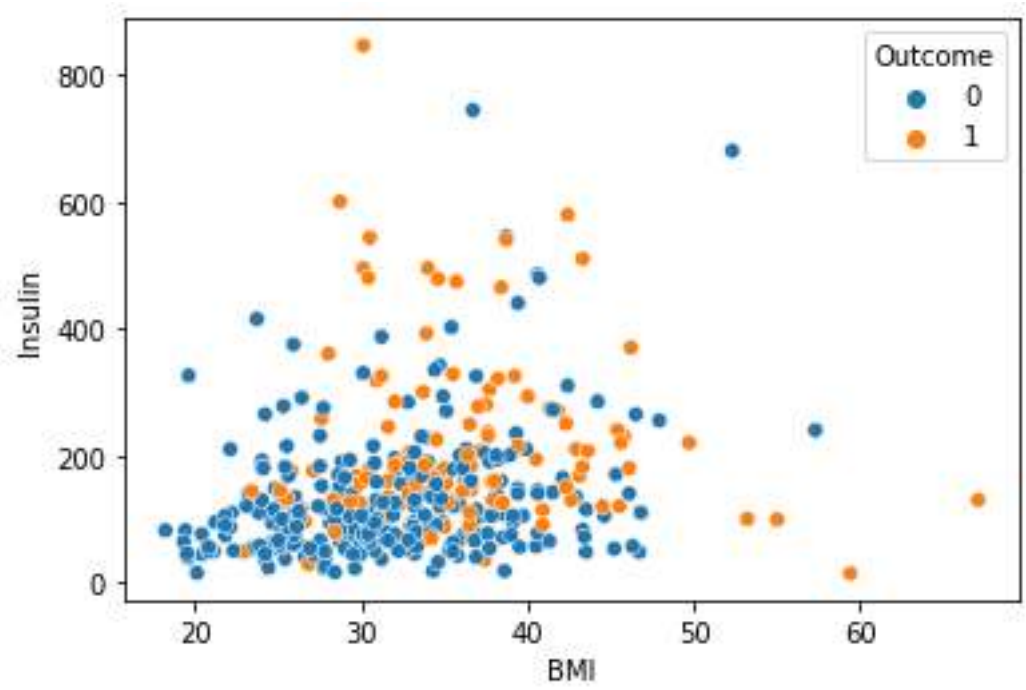
Şekil 12. Glikoz ve Kan Basıncı Arasındaki İlişki Grafiği

İnsülin ile cilt kalınlığı arasındaki ilişki: insülin 0 ile 400 seviyelerindeyken ve cilt kalınlığı 0 ile 50 seviyelerindeyken şeker hastalığı riski artmaktadır. Aşağıda insülin ve cilt kalınlığı arasındaki ilişkiyi gösteren grafik yer almaktadır.



Şekil 13. İnsülin ve Cilt Kalınlığı Arasındaki İlişki Grafiği

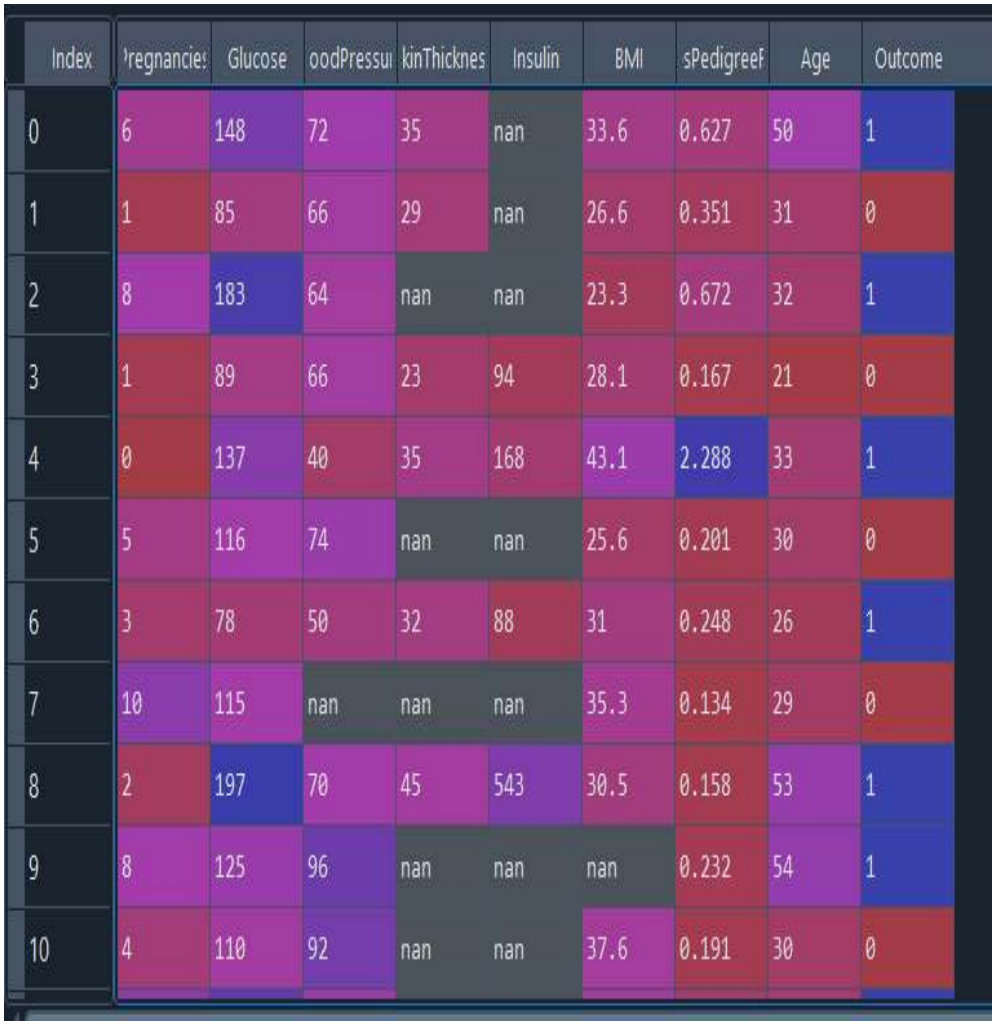
İnsülin ile Vücut Kitle Endeksi arasındaki ilişki: insülin 0 ile 600 seviyelerindeyken ve vücut kitle endeksi 25 ile 50 seviyelerindeyken şeker hastalığı riski artmaktadır.



Şekil 14. İnsülin ve Vücut Kitle Endeksi Arasındaki İlişki Grafiği

2. UYGULAMA

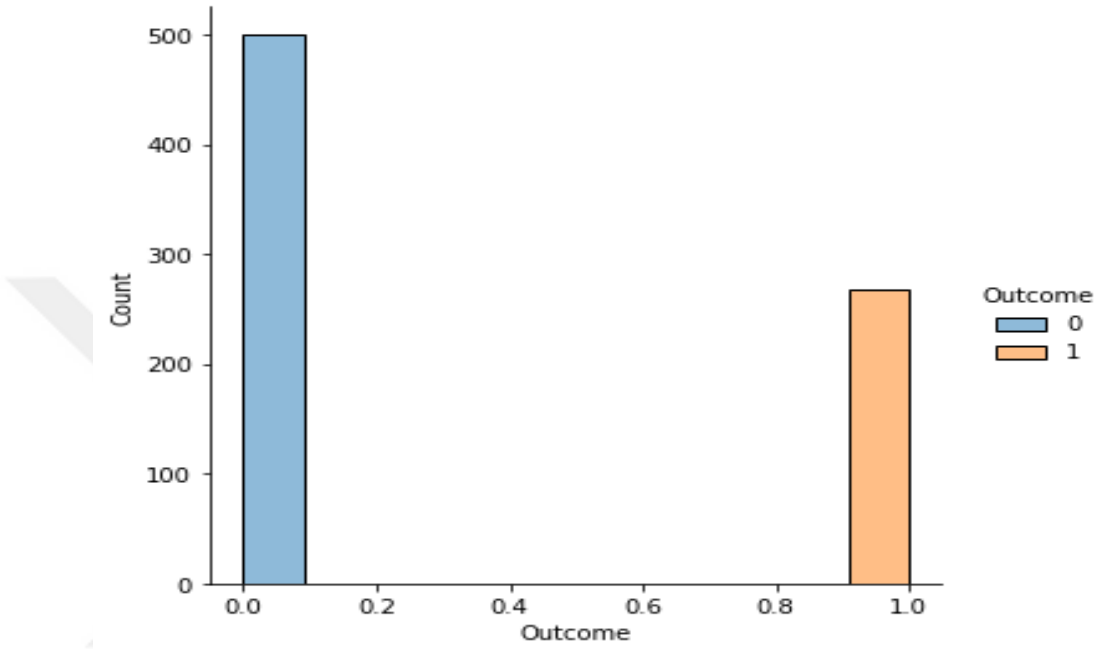
Makine öğrenimi tekniklerini kullanan birden fazla araç vardır, ancak bu çalışma için Python programlama dili seçilmiştir. Python programlama dilinin kullanımı çok daha kolay, daha verimli ve daha analitiktir. Diyabet hastalığı Python programlama dili kullanılarak farklı şekillerde modellenmiştir. Kaggle'dan alınan veri setleri .csv uzantısına sahip olduğundan, veri setlerini okumak ve işlemek için Python programlama dilinde CSV kullanılmıştır. Veri setinin içeriği aşağıda detaylandırılmıştır.



Index	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	nan	33.6	0.627	50	1
1	1	85	66	29	nan	26.6	0.351	31	0
2	8	183	64	nan	nan	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1
5	5	116	74	nan	nan	25.6	0.201	30	0
6	3	78	50	32	88	31	0.248	26	1
7	10	115	nan	nan	nan	35.3	0.134	29	0
8	2	197	70	45	543	30.5	0.158	53	1
9	8	125	96	nan	nan	nan	0.232	54	1
10	4	110	92	nan	nan	37.6	0.191	30	0

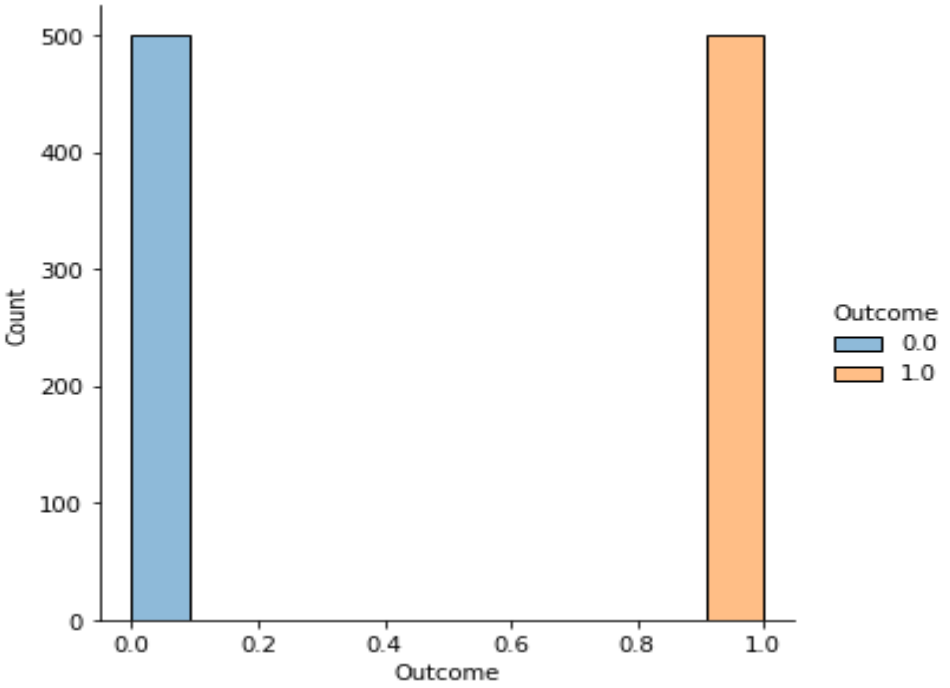
Şekil 15. Tüm Veri Setindeki Bilgilerin Çıktısı

Veri setine herhangi bir işlem yapılmadan outcome bilgisinde diyabet olan hastalar 1 olmayan hastalar ise 0 ile ifade edilmektedir. 768 veriden 500 veri diyabet hastası değil iken 268 tanesi diyabet hastasıdır. Train edilirken daha doğru sonuç elde edebilmek için dengede olması beklenir. Bu nedenle imbalance'lık (dengesizlik) giderilmiştir. Aşağıda görsel yer almaktadır.



Şekil 16. Outcome Değerlerinin İmbalance Edilmeden Önceki Grafiği

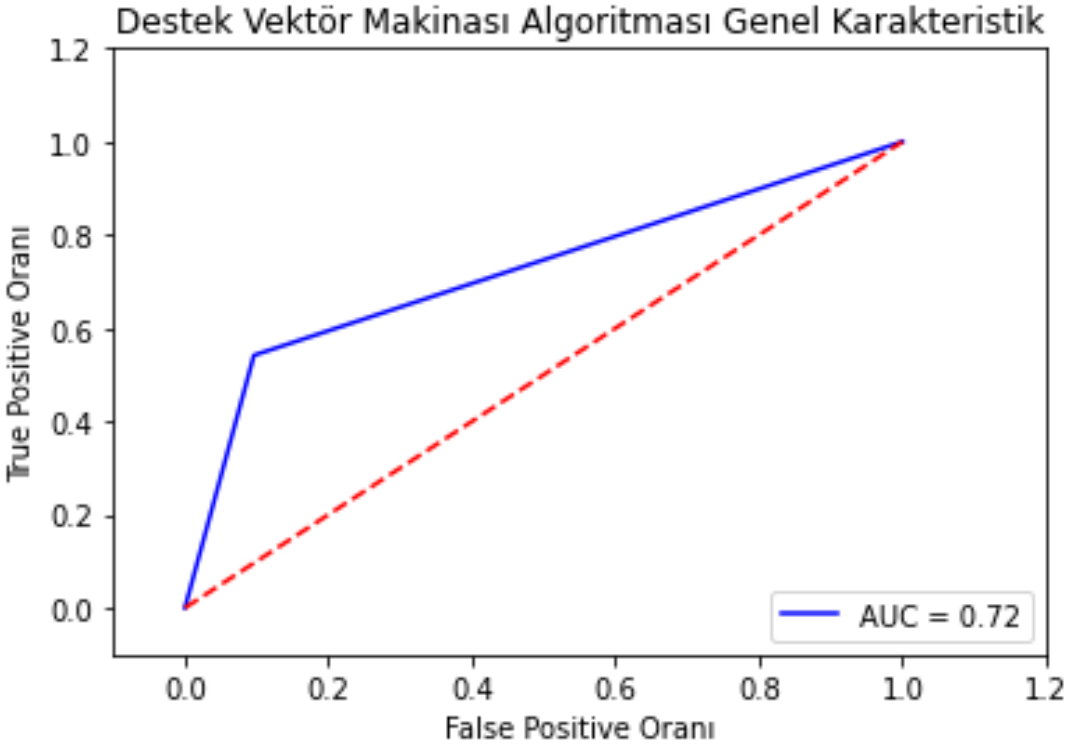
Veri seti imbalance edildikten sonra elde edilen grafik aşağıda yer almaktadır. Verilerin dengede olması beklenir. Train edilirken daha doğru sonuç elde etmek için gereklidir. Verisi az olan etiketin verilerini rastgele bir şekilde çoklayarak etiketlerdeki veri sayısı eşitlenmiş ve imbalance'lık giderilmiştir.



Şekil 17. Outcome Değerlerinin İmbalance Edildikten Sonraki Grafiği

2.1. Destek Vektör Makinesi Algoritmasının Değerlendirilmesi

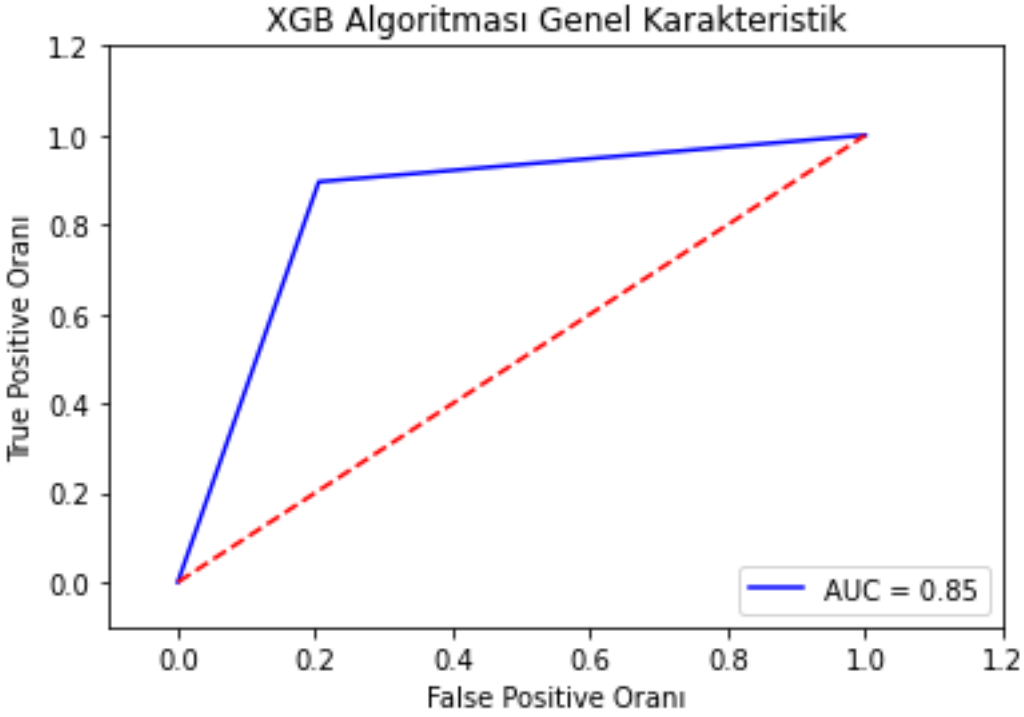
DVM modelini değerlendirmek için, %70 train işlemi %30 test işlemi gerçekleştirilmiştir. DVM modelini değerlendirmek için test işlemi sonucunda %72 AUC oranı elde edilmiştir. Sistemin genel başarı (doğruluk) oranı olarak %76 'lık oran çıkmıştır.



Şekil 18. DVM AUC Grafiği

2.2. XBoost Algoritmasının Değerlendirilmesi

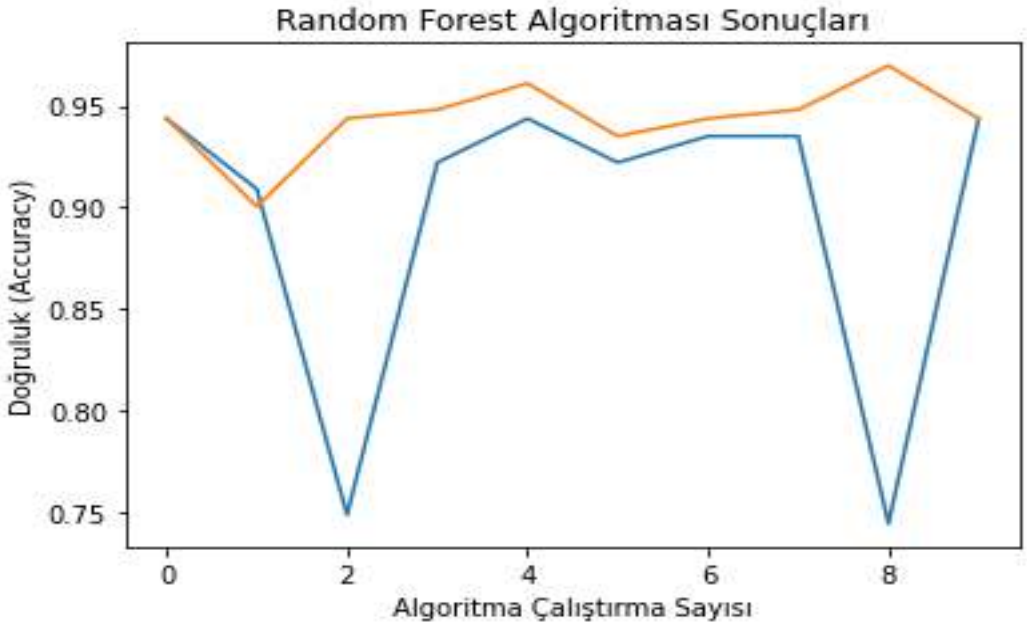
XGBoost modelini değerlendirmek için, %70 train işlemi %30 test işlemi gerçekleştirilmiştir. Test işlemi sonucunda %85 AUC oranı elde edilmiştir. Sistemin genel başarı oranı olarak %82'lık oran çıkmıştır.



Şekil 19. XGBoost AUC Grafiği

2.3. Rastgele Orman Algoritmasının Değerlendirilmesi

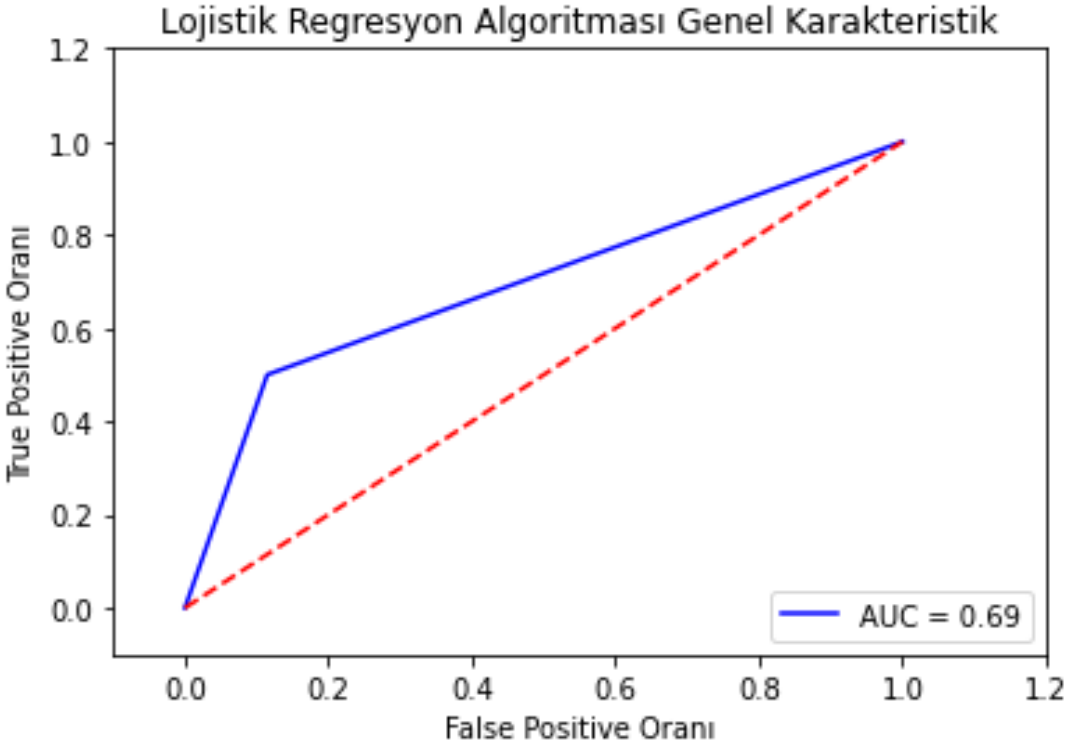
RF modelini değerlendirmek için, %70 train işlemi %30 test işlemi gerçekleştirilmiştir. Model 10 kere çalıştırılmış ve sistemin genel başarı oranı olarak en yüksek %96' lık oran çıkmıştır.



Şekil 20. Random Forest Doğruluk Grafiği

2.4. Lojistik Regresyon Algoritmasının Değerlendirilmesi

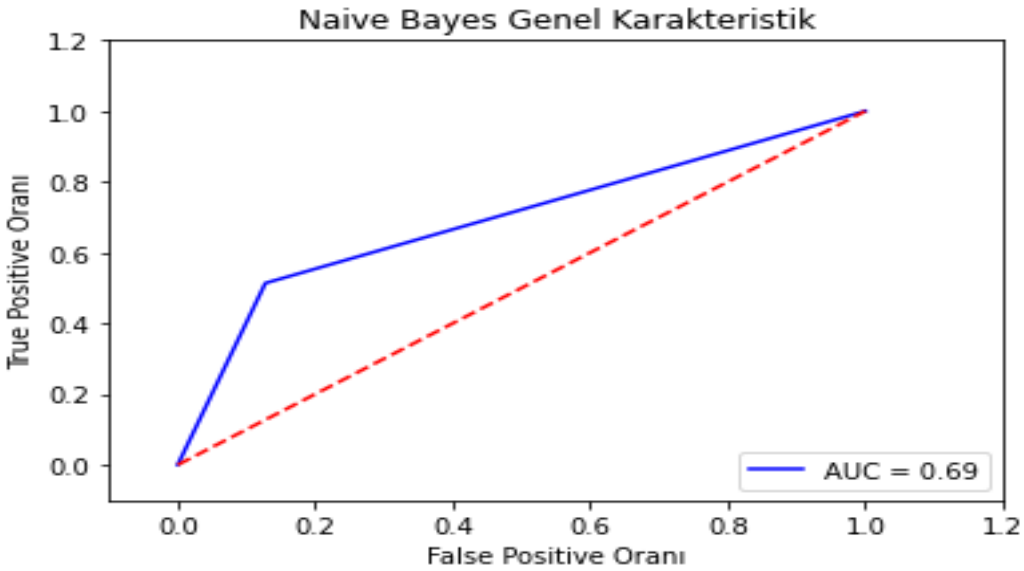
LR modelini değerlendirmek için, %70 train işlemi %30 test işlemi gerçekleştirilmiştir. Lojistik Regresyon değerlendirmek için test işlemi sonucunda %69 AUC oranı elde edilmiştir. Modelin genel başarı oranı (doğruluk) olarak en yüksek %78' lik oran çıkmıştır.



Şekil 21. Lojistik Regresyon AUC Grafiği

2.5. Naive Bayes Algoritmasının Değerlendirilmesi

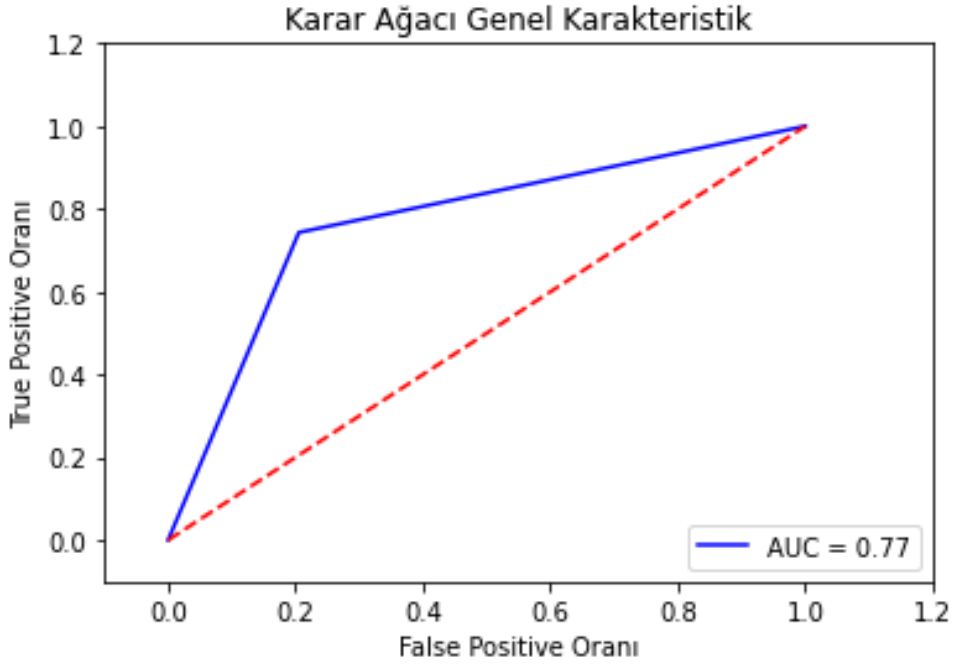
NB modelini değerlendirmek için, %70 train işlemi %30 test işlemi gerçekleştirilmiştir. Test işlemi sonucunda %69 AUC oranı elde edilmiştir. Modelin genel başarı oranı (doğruluk) olarak en yüksek %76' lık oran çıkmıştır.



Şekil 22. Naive Bayes AUC Grafiği

2.6. Karar Ağacı Algoritmasının Değerlendirilmesi

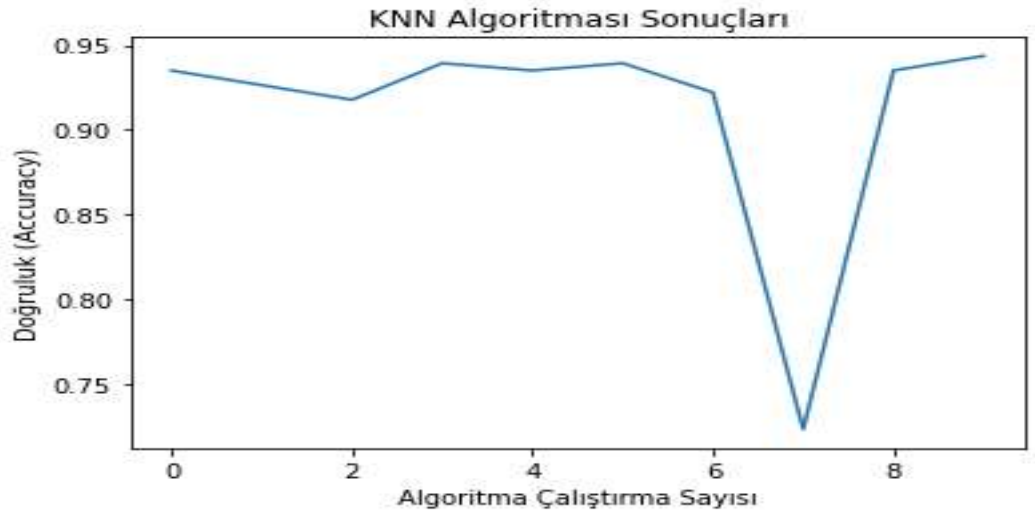
DT modelini değerlendirmek için, %70 train işlemi %30 test işlemi gerçekleştirilmiştir. Test işlemi sonucunda %77 AUC oranı elde edilmiştir. Modelin genel başarı oranı (doğruluk) olarak en yüksek %79' luk oran çıkmıştır.



Şekil 23. Karar Ağacı AUC Grafiği

2.7. KNN Algoritmasının Değerlendirilmesi

KNN modelini değerlendirmek için, %70 train işlemi %30 test işlemi gerçekleştirilmiştir. Model 10 kere çalıştırılmış ve sistemin genel başarı oranı olarak en yüksek %96'lık oran çıkmıştır.



Şekil 24. KNN Doğruluk Grafiği

2.8. Algoritmaların Özet Tablosu

Tablo 2. Algoritmaların Özet Tablosu

Model	Doğruluk	F1-Score
Destek Vektör Makineleri	%75.6	%75
XGBoost	%81.6	%83
Rastgele Orman	%96	%96
Lojistik Regresyon	%78	%76
Naive Bayes	%76	%75
Karar Ağacı	%79	%79
KNN	%96	%96

3. UYGULAMAYA AİT YAZILIM

Tez çalışmasında gerçekleştirilen kodlar açıklamalarıyla birlikte aşağıda yer almaktadır.

```
"""
Created on Mon Jan 16 21:42:50 2023

@author: tuğba
"""
# Kullanılan kütüphanelere ait kodlar aşağıda yer almaktadır. Numpy, Pandas, Seaborn, Matplotlib, Sklearn, Imblearn, Xgbboost kütüphaleri kullanılmıştır.
import numpy as nmp # linear algebra
import pandas as pnd
import seaborn as sbn
import matplotlib.pyplot as plt

from sklearn.impute import KNNImputer
from sklearn.model_selection import train_test_split, GridSearchCV
from imblearn.over_sampling import RandomOverSampler
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn import tree
from sklearn.naive_bayes import GaussianNB
from sklearn import svm
import xgboost as xgb
```

```
from sklearn.metrics import f1_score
from sklearn import metrics
from sklearn.metrics import accuracy_score
from sklearn.metrics import classification_report

# Aşağıda csv veri dosyası okunmaktadır. Veri dosyası içeriğinde diyabetli ve diyabetli olmayan veriler yer almaktadır.
bdf = pnd.read_csv('diabetes.csv')

# Aşağıda baştan 5 elemanı gösteren kod yer almaktadır.
bdf.head()

# Aşağıda yer alan kod ile hangi verilerin kullanıldığına, veri tiplerine null olmayan kayıtların sayısına bakıyoruz.
bdf.info()

#Aşağıda yer alan kod ile istatiksel bilgilere bakıyoruz.
bdf.describe()

# null değerleri bulunur ve yazdırılır.
null_vals = bdf.isnull().sum()
print(null_vals)

# Aşağıda yer alan kod ile outcome sütununun grafiksel olarak gösterilmesi sağlanır.
sbn.displot(bdf, x='Outcome' , kind='hist' , hue='Outcome')
```

```
plt.show()


# glikoz ile kan basıncı arasındaki grafiği gösteren koddur.
glu = sbn.scatterplot(x= "Glucose" ,y= "BloodPressure",
            hue="Outcome",
            data=bdf);


# vücut kitle endeksi ile insülin grafiği arasındaki grafiği gösteren koddur.
Bmi = sbn.scatterplot(x= "BMI" ,y= "Insulin", hue="Outcome", data=bdf);


# Cilt kalınlığı ile insülin arasındaki grafiği gösteren koddur.
Skin = sbn.scatterplot(x= "SkinThickness", y= "Insulin", hue="Outcome",
            data=bdf);


sbn.pairplot(bdf)
plt.title('Değikenlerin Saçıntı Grafiği')


# korelasyon grafiği
plt.subplots(figsize=(8,8))
sbn.heatmap(bdf.corr(), annot=True,cmap='viridis')



# Aşağıda yer alan kod ile veri temizleme işlemi yapılmaktadır.
bdf['BMI'].replace(0, nmp.nan , inplace = True)
bdf['BloodPressure'].replace(0, nmp.nan , inplace = True)
```

```
bdf['Insulin'].replace(0, nmp.nan , inplace = True)

bdf['Glucose'].replace(0, nmp.nan , inplace = True)

bdf['SkinThickness'].replace(0, nmp.nan , inplace = True)


# toplam null sayısını buluyoruz.

bdf.isnull().sum()


# Sıfır yerine null koymuştuk null değerlerine uygun değer atıyoruz.

imputer = KNNImputer(n_neighbors=7)

imputed_bdf = pnd.DataFrame(imputer.fit_transform(bdf))

imputed_bdf.columns = bdf.columns


imputed_bdf.isnull().sum()


imputed_bdf.info()


# veri seti içindeki çiftlenen kayıtlar bulunur.

imputed_bdf.duplicated().sum()


# test ve eğitim amaçlı X ve y değişkenleri oluşturulmuştur. %70 eğitim - %30 test veri seti kullanılmıştır.

X = imputed_bdf[["Glucose","BloodPressure","BMI","Age"]]

X_n = imputed_bdf[["Pregnancies","Glucose","BloodPressure",


"SkinThickness","Insulin","BMI","DiabetesPedigreeFunction","Age"]]
```

```
y = imputed_bdf[['Outcome']]
X

X_trn, X_tst, Y_trn, Y_tst = train_test_split(X, y, train_size=0.7,
                                              test_size=0.3,
                                              random_state=0)

# Datadan bias değerlerini atıyoruz.
orneklem = RandomOverSampler(sampling_strategy='minority')
X_orn,Y_orn = orneklem.fit_resample(X,y)
Sampled_Dataset = X_orn.merge(Y_orn,left_index=True,right_index=True)

sbn.displot(Sampled_Dataset , x='Outcome' , kind='hist' , hue='Outcome')
plt.show()

X_trn2, X_tst2, Y_trn2, Y_tst2 = train_test_split(X_orn, Y_orn,
                                              train_size=0.7,
                                              test_size=0.3,
                                              random_state=0)

# Lojistik regresyon algoritması
logreg = LogisticRegression(random_state=16)
```

#Lojistik regresyon algoritmasında Y_tst , y_coz değerleri için doğruluk oranı tespit edilmiştir.

```
logreg.fit(X_trn, Y_trn)
y_coz = logreg.predict(X_tst)
cnf_matrix = metrics.confusion_matrix(Y_tst, y_coz)
cnf_matrix
t3=accuracy_score(Y_tst , y_coz)
print(t3, "log reg accurancy")
t5=f1_score(Y_tst, y_coz, average='weighted')
print(t5, "log reg f1-score")
```

Lojistik regresyon algoritmasında Y_tst2 , y_coz2 değerleri için doğruluk oranı tespit edilmiştir.

```
logreg.fit(X_trn2, Y_trn2)
y_coz2 = logreg.predict(X_tst2)
t4=accuracy_score(Y_tst2 , y_coz2)
print(t4, "log reg accurancy1")
t6=f1_score(Y_tst2, y_coz2, average='weighted')
print(t6, "log reg1 f1-score")

target_names = ['Şeker Hastalığı Yok', 'Şeker Hastası']
print(classification_report(Y_tst2, y_coz2, target_names=target_names))
print(classification_report(Y_tst, y_coz, target_names=target_names))
```

SVM Algoritması

Destek Vektör Makineleri algoritmasında lineer ve lineer olmayan değerler için çalışma yapılmıştır.

```
poly2 = svm.SVC(kernel='poly', degree=4, C=1, decision_function_shape='ovo').fit(X_trn2, Y_trn2)

poly = svm.SVC(kernel='poly', degree=3, C=1, decision_function_shape='ovo').fit(X_trn, Y_trn)

lin = svm.SVC(kernel='linear', C=1, decision_function_shape='ovo').fit(X_trn, Y_trn)

lin2 = svm.SVC(kernel='linear', C=1, decision_function_shape='ovo').fit(X_trn2, Y_trn2)


poly_coz = poly.predict(X_tst)

lin_coz = lin.predict(X_tst)

poly2_coz = poly.predict(X_tst2)

lin2_coz = lin.predict(X_tst2)
```

Destek vektör makinelerinde lineer ve lineer olmayan değerler için doğruluk ve f1 skor değerleri tespit edilmiştir.

```
poly2_accuracy = accuracy_score(Y_tst2, poly2_coz)

lin2_accuracy = accuracy_score(Y_tst2, lin2_coz)

poly_accuracy = accuracy_score(Y_tst, poly_coz)

poly_f1 = f1_score(Y_tst, poly_coz, average='weighted')

print('Accuracy (Polynomial Kernel): ', "%.2f" % (poly_accuracy*100))

print('F1 (Polynomial Kernel): ', "%.2f" % (poly_f1*100))
```

Destek vektör makinelerinde lineer değerler için doğruluk ve f1 skor değerleri tespit edilmiştir.

```
lin_accuracy = accuracy_score(Y_tst, lin_coz)
lin_f1 = f1_score(Y_tst, lin_coz, average='weighted')
print('Accuracy (Linear Kernel): ', "%.2f" % (lin_accuracy*100))
print('F1 (Linear Kernel): ', "%.2f" % (lin_f1*100))
```

Destek vektör makinelerinde lineer ve lineer olmayan değerler için doğruluk ve f1 skor değerleri tespit edilmiştir.

```
accuracy_poly = poly.score(X_tst, Y_tst)
accuracy_lin = lin.score(X_tst, Y_tst)
print("Doğruluk (Accuracy) Polynom:", accuracy_poly)
print("Doğruluk (Accuracy) Lineer:", accuracy_lin)
```

Destek vektör makinelerinde lineer ve lineer olmayan değerler için karışıklık matrisi tespit edilmiştir.

```
poly = metrics.confusion_matrix(Y_tst, poly_coz)
rbf = metrics.confusion_matrix(Y_tst, lin_coz)
print(poly)
print(rbf)
```

```
        ##################################################################
#
        #ROC Eğrisi Fonksiyonuna ait kodlar aşağıda yer almaktadır.
        def ROC_Ciz(y_tst, y_coz, ne):
            f, t, sinir = metrics.roc_curve(y_tst, y_coz)
            roc_auc = metrics.auc(f,t)
            plt.title(ne + ' Genel Karakteristik')
            plt.plot(f, t, 'b',label='AUC = %0.2f'% roc_auc)
            plt.legend(loc='lower right')
            plt.plot([0,1],[0,1],'r--')
            plt.xlim([-0.1,1.2])
            plt.ylim([-0.1,1.2])
            plt.ylabel('True Positive Oranı')
            plt.xlabel('False Positive Oranı')
            plt.show()

        # ###################### XGB Algoritması ######################
        #XGB algoritmasına ait kodlar aşağıda yer almaktadır. Eğitim ve Test değerleri
belirlenmiştir. Y_tst, y_coz1 değerleri için doğruluk oranı tahmin edilmiştir. Y_tst2,
y_coz2 değerleri için doğruluk oranı tahmin edilmiştir. Ayrıca bu değerler için ROC
eğrisine ait grafiği çizdiren kodlar da aşağıda yer almaktadır.

        xgb_algoritma = xgb.XGBClassifier()
        xgb_algoritma.fit(X_trn , Y_trn)
        y_coz1 = xgb_algoritma.predict(X_tst)
```

```
t1=accuracy_score(Y_tst, y_coz1)
print(t1 , "XGB accurancy")
f2 = f1_score(Y_tst, y_coz1, average='weighted')
print(f2 , "XGB f1-score1")

xgb_algoritma.fit(X_trn2 , Y_trn2)
y_coz2 = xgb_algoritma.predict(X_tst2)
t2=accuracy_score(Y_tst2, y_coz2)
print(t2 , "XGB accurancy2")
f1 = f1_score(Y_tst2, y_coz2, average='weighted')
print(f1 , "XGB f1-score2")

ROC_Ciz(Y_tst, y_coz1, "XGB Algoritması")
ROC_Ciz(Y_tst2, y_coz2, "XGB Algoritması")

# ############## KARAR AĞACI Algoritması ########################
```

#Karar Ağacı algoritmasına ait kodlar aşağıda yer almaktadır. Eğitim ve Test değerleri belirlenmiştir. X_tst, Y_tst değerleri için skor tahmin edilmiştir. X_tst2, Y_tst2 değerleri için skor tahmin edilmiştir. Ayrıca bu değerler için ROC eğrisine ait grafiği çizdiren kodlar da aşağıda yer almaktadır.

```
clf = tree.DecisionTreeClassifier(criterion="entropy", random_state=0)
clf1 = clf.fit(X_trn,Y_trn)
y_coz1 = clf1.predict(X_tst)
clf1.score(X_tst, Y_tst)
```

```
f2 = f1_score(Y_tst, y_coz1, average='weighted')
print(f2 , "karar ağacı f1-score1")

mat = metrics.confusion_matrix(Y_tst, y_coz1)
print('Karar Ağacı Veri 1')
print(mat)

clf2 = clf.fit(X_trn2,Y_trn2)
y_coz2 = clf2.predict(X_tst2)
clf2.score(X_tst2, Y_tst2)
f1 = f1_score(Y_tst2, y_coz2, average='weighted')
print(f1 , "karar ağacı f1-score2")

mat = metrics.confusion_matrix(Y_tst2, y_coz2)
print('Karar Ağacı Veri 2')
print(mat)

ROC_Ciz(Y_tst, y_coz1, "Karar Ağacı")
ROC_Ciz(Y_tst2, y_coz2, "Karar Ağacı")

################## NAIVE BAYES Algoritması ############################
```

#Naive Bayes algoritmasına ait kodlar aşağıda yer almaktadır. Eğitim ve Test değerleri belirlenmiştir. X_tst, Y_tst değerleri için skor tahmin edilmiştir. X_tst2, Y_tst2 değerleri için skor tahmin edilmiştir. Ayrıca bu değerler için ROC eğrisine ait grafiği çizdiren kodlar da aşağıda yer almaktadır.

```
gnb = GaussianNB()
gnb1 = gnb.fit(X_trn,Y_trn)
y_coz1 = gnb1.predict(X_tst)
gnb1.score(X_tst, Y_tst)
f1 = f1_score(Y_tst, y_coz1, average='weighted')
print(f1 , "naive bayes f1-score2")

gnb2 = gnb.fit(X_trn2,Y_trn2)
y_coz2 = gnb2.predict(X_tst2)
gnb2.score(X_tst2, Y_tst2)
f2 = f1_score(Y_tst2, y_coz2, average='weighted')
print(f2 , "naive bayes f1-score2")

ROC_Ciz(Y_tst, y_coz1, "Naive Bayes")
ROC_Ciz(Y_tst2, y_coz2, "Naive Bayes")

################################
```

#KNN algoritmasına ait kodlar aşağıda yer almaktadır. KNN algoritmasını tahmin etmek amacıyla fonksiyon tanımlanmıştır. k tahmin değeri rastgele belirlenmiştir. İki ayrı fonksiyon sonucunda doğruluk oranını ve f1-score tahmin eden değer dönmektedir.

```
def KNN_Tahmin(Xtr, Ytr, Xt, Yt):
    k_values = [k for k in range(1,100,2)]

    model = KNeighborsClassifier()
```

```
    param_grid = dict(n_neighbors=k_values)

    # parametre aralıklarını belirliyoruz
    grid = GridSearchCV(model, param_grid, cv=10, scoring='accuracy',
                        return_train_score=False,verbose=1,n_jobs=-1)

    # model for grid arama algoritması
    grid_search=grid.fit(Xtr, Ytr)

    print(grid_search.best_params_)
    tahmin = grid.predict(Xt)
    print("Doğruluk (Accuracy) KNN: ",accuracy_score(tahmin,Yt))
    return accuracy_score(tahmin,Yt)

################################

def KNN_Tahmin1(Xtr, Ytr, Xt, Yt):
    k_values = [k for k in range(1,100,2)]

    model = KNeighborsClassifier()
    param_grid = dict(n_neighbors=k_values)

    # parametre aralıklarını belirliyoruz
    grid = GridSearchCV(model, param_grid, cv=10, scoring='accuracy',
                        return_train_score=False,verbose=1,n_jobs=-1)
```

```
    # model for grid arama algoritması
    grid_search=grid.fit(Xtr, Ytr)


    print(grid_search.best_params_)
    tahmin = grid.predict(Xt)
    print("F1-Score KNN: ",f1_score(tahmin, Yt, average='weighted'))
    return f1_score(tahmin, Yt, average='weighted')
```

#Rastgele Orman algoritmasına ait kodlar aşağıda yer almaktadır. Rastgele Orman algoritmasını tahmin etmek amacıyla fonksiyon tanımlanmıştır. İki ayrı fonksiyon sonucunda doğruluk oranını ve f1-score tahmin eden değer dönmektedir.

```
def RandomForest_Tahmin(Xtr, Ytr, Xt, Yt):
    model = RandomForestClassifier()
    model.fit(Xtr,Ytr)
    tahmin = model.predict(Xt)
    print("Doğruluk (Accuracy) Random Forest: ",accuracy_score(tahmin,Yt))
    return accuracy_score(tahmin,Yt)
##################################################
def RandomForest_Tahmin1(Xtr, Ytr, Xt, Yt):
    model = RandomForestClassifier()
    model.fit(Xtr,Ytr)
    tahmin = model.predict(Xt)
    print("F1-Score Random Forest: ",f1_score(tahmin, Yt,
average='weighted'))
```

```
    return f1_score(tahmin, Yt, average='weighted')


X_yeni=X_n.drop(["DiabetesPedigreeFunction","SkinThickness","Insulin","
Pregnancies"],1)

X_yeni.columns

orneklem = RandomOverSampler(sampling_strategy='minority')

X_orn,Y_orn = orneklem.fit_resample(X_yeni, y)

Sampled_Dataset = X_orn.merge(Y_orn,left_index=True,right_index=True)
```

#KNN ve RF algoritmasına ait kodlar aşağıda yer almaktadır. KNN ve RF algoritmasını tahmin etmek amacıyla fonksiyon çağrılmıştır. Fonksiyon sonucunda dönen doğruluk oranını bir liste içine alınmıştır. Bu listeler hem KNN hem de RF için grafiksel olarak gösterilmiştir.

```
accuracy_listKNN = []

score_listKNN=[]

accuracy_listRF = []

score_listRF=[]



for i in range(10):

    X_trn3, X_tst3, Y_trn3, Y_tst3 = train_test_split(X_orn, Y_orn,
train_size=0.7, test_size=0.3)

    accuracy_listKNN.append(KNN_Tahmin(X_trn3,Y_trn3,X_tst,Y_tst))

score_listKNN.append(KNN_Tahmin1(X_trn3,Y_trn3,X_tst,Y_tst))
```

```
accuracy_listRF.append(RandomForest_Tahmin(X_trn3,Y_trn3,X_tst,Y_tst))
score_listRF.append(RandomForest_Tahmin1(X_trn3,Y_trn3,X_tst,Y_tst))
```

```
plt.title('KNN Algoritması Sonuçları')
plt.plot(accuracy_listKNN)
plt.ylabel('Doğruluk (Accuracy)')
plt.xlabel('Algoritma Çalıştırma Sayısı')

plt.title('KNN Algoritması Sonuçları')
plt.plot(score_listKNN)
plt.ylabel('F1-Score')
plt.xlabel('Algoritma Çalıştırma Sayısı')

plt.title('Random Forest Algoritması Sonuçları')
plt.plot(accuracy_listRF)
plt.ylabel('Doğruluk (Accuracy)')
plt.xlabel('Algoritma Çalıştırma Sayısı')

plt.title('Random Forest Algoritması Sonuçları')
plt.plot(score_listRF)
plt.ylabel('F1-Score')
plt.xlabel('Algoritma Çalıştırma Sayısı')
```

SONUÇ

Bu tezde diyabet gibi önemli hastalıkların proaktif olarak nasıl teşhis edileceği ve hangi makine öğrenimi tekniklerinin daha başarılı ve doğru sonuçlar verdiği araştırıldı. Günümüzde her alanda bulunabilen yapay zeka sistemleri, sağlık alanında da yaygın olarak kullanılmaktadır. Dahil olduğu her alanda hayatı kolaylaştıran yapay zeka, sağlık sektöründe de hastalıkların erken teşhisinde yardımcı olacaktır. Özellikle sağlık alanında hastalığın türü ne olursa olsun erken teşhis çok önemlidir.

Makine öğrenmesi olarak adlandırılan yapı verilerden elde ettiği bilgilerle algoritmalar üzerinden tahminler yürütürken insanların öğrenme şeklini taklit eder ve elde edilen sonuçların gerçekliğini hesaplar. Makine öğrenmesi; yüz tanıma, veri sınıflama, verileri kümeleme, ürün önerisi, hastalık tanısı, trafik tahmini, karakter analizi gibi pek çok alanda kullanılmakta ve hızla gelişmektedir. İnsan sağlığı tüm evrende önemli bir yere sahiptir. Sağlık alanında birçok tanı ve hastalık üzerinde bilimsel çalışmalar yapılmaktadır. Makine öğrenmesi her alanda olduğu gibi sağlık alanında da birçok çalışmada yüksek etkiye sahip sonuçlar elde edilmesine katkı sunmuştur. Makine öğrenmesi ile göğüs kanseri, diyabet hastalığı, siğil tedavisi, karaciğer hastalığı, KOAH teşhisi, tiroid hastalığı, siroz, parkinson, psikiyatri, korenor arter hastalığı ve covid 19 teşhisi gibi birçok sağlık alanında çalışmalar yapılmıştır.

Python yazılımı ile öznitelikler arasındaki ilişkiler grafiksel olarak belirlenmiştir. İnsülin ile vücut kitle endeksi arasındaki ilişki, insülin ile cilt kalınlığı arasındaki ilişki ve glikoz ile kan basıncı arasındaki ilişki grafiksel olarak gösterilmiştir. Öznitelikler arasındaki tüm ilişki Python yazılım dili aracılığıyla grafiksel olarak gösterilip şeker hastalığının hangi aralıklarda yoğunlaştığı tespit edilerek şeker hastalığı riski üzerine yorum yapılabilir.

Python yazılımı ile KNN, Karar Ağaçları, Naive Bayes, Destek Vektör Makineleri, XGBoost, Rastgele Orman ve Lojistik Regresyon algoritmaları kullanılarak Diyabet Hastalığı, Kaggle veri seti üzerinde çalışılmıştır. Veri setinde eksik veri kontrolü sağlanmıştır. Algoritmalar uygulanırken oluşturulan modelin güvenilirliğini artırmak için veri setinde çapraz doğrulama işlemi yapılmıştır. Kullanılan bu algoritmaların sınıflama performansları ve değerlendirme ölçütleri

hesaplanarak karşılaştırılmıştır. Yapılan çalışmalarda uygulanan modellerin performansı Doğruluk (Accuracy), Duyarlılık (Recall), Kesinlik (Precision), F1-Score, ROC/AUC ile hesaplanmıştır.

Bu tezde diyabet rahatsızlığı erken teşhis edilmesi açısından sorunun farklı makine öğrenim algoritmalarıyla Spyder uygulaması üzerinden örneklemesi yapılmıştır. KNN yöntemi ile %96, Naive Bayes yöntemi ile %69, C.4.5 Karar Ağacı yöntemi ile de %77, Destek Vektör Makineleri ile (doğrusal ve doğrusal olmayan model için) %76, Lojistik Regresyon ile %78, XGBoost modeli ile %82 ve Rastgele Orman yöntemi ile %96 başarılı tahmin oranları elde edilmiştir. Bu tezde kullanılan veri seti (yaş, kan basıncı, glikoz, insülin vb.) özelliklerini diyabet hastalığı teşhisi ile inceleyerek diyabet hastalığı teşhis değerini (0,1) en yüksek oranda doğru tahmin etmektir. Test verisi ile eğitim verisi performansı test edilmiş ve istatiksel olarak performans oranı diğer yöntemlere göre mukayese edilmiştir.

Bu çalışmada veri setlerinde kullanılan algoritmalardan sonra oluşturulan modellere göre diğer kullanılan algoritmalara kıyasla Rastgele Orman ve KNN algoritması ile daha iyi sonuç elde edildiği görülmüştür. Bu tez çalışmasında elde edilen sonuçlara göre şu önerilerde bulunulabilir:

- Sistem başarısını yükseltmek için farklı özelliklerin eklendiği veri seti ile değerlendirilebilir.
- Yeni geliştirilen makine öğrenmesi algoritmaları ile kullanılabilir.
- Farklı veri işleme teknikleri ve farklı algoritmalar kullanılarak sistemin kazancı değerlendirilebilir.
- Diyabet ile ilgili farklı veri setleri kullanılabilir.

Netice olarak bu araştırma sonuçları şeker hastalığı tanısını elde etmede değil yalnızca hekimlere değerlendirmeleri ve ileri tetkikler çalışmaları için fikir verme amaçlı kullanıldığında şeker hastası olan bireylerin hastalığının hayati tehlike kapsamına gelmeden tedbir alınabilmesi için hekimlere uygulanabilir bir kaynak sunmaktadır.

Bu çalışmada gerçekleştirilen tezin literatüre katkısı diyabet gibi mühim bir rahatsızlığın tedavisinde erken tanısının konulmasının sağlanmasıdır. Bu erken teşhisi

en doğru oranda ve yöntemde makine öğrenmesi ile saptanması çok önemlidir. Bu sebeple, bu çalışmada yer alan mevcut veri setine farklı makine öğrenmesi yöntemi ile en yüksek oranda netice elde etmek çok önemlidir. Erken teşhis en yüksek oranda rastgele orman ve KNN yöntemi ile tespit edilmiş ve olumsuz sonuçları en aza indirgenmesi amacıyla çalışmalar yürütülmüştür. Diyabet hastalığı tanısı için makine öğrenmesi yöntemleri arasında en yüksek oranın rastgele orman ve KNN yönteminden elde edildiği ve bu yöntemlerin kullanılabileceği gösterilmektedir.

KAYNAKÇA

Acharya, Bhat, P. S., Iyengar, S. S., Rao, A., & Dua, S. (2003). Classification of heart rate data using artificial neural network and fuzzy equivalence relation. Pattern recognition, 36(1), 61-68.

Al Daoud, E. (2019). Comparison between XGBoost, LightGBM and CatBoost using a home credit dataset. *International Journal of Computer and Information Engineering*, *13*(1), 6-10.

Alfeilat A.,Hassanat, Lasassmeh, Tarawneh, Alhasanat, Salman E., Prasath, Effects of distance measure choice on k-nearest neighbor classifier performance: a review. *Big data*, *7*(4), (2019), 221-248.

Alizadehsani, R., Abdar, M., Roshanzamir, M., Khosravi, A., Kebria, P. M., Khozeimeh, F., ... & Acharya, U. R. (2019). Machine learning-based coronary artery disease diagnosis: A comprehensive review. *Computers in biology and medicine*, *111*, 103346.

Alpaydin, E. (2010). Introduction to machine learning, 2nd edn. Adaptive computation and machine learning.

Ayhan, S., ve Erdoğmuş, Ş. (2014). Destek vektör makineleriyle sınıflandırma problemlerinin çözümü için çekirdek fonksiyonu seçimi. Eskişehir Osmangazi Üniversitesi İİBF Dergisi, 9(1), 175-201.

Bayes, T. (1763). LII. An essay towards solving a problem in the doctrine of chances. By the late Rev. Mr. Bayes, FRS communicated by Mr. Price, in a letter to John Canton, AMFR S. *Philosophical transactions of the Royal Society of London*, (53), 370-418.

Bishop, C. M., & Nasrabadi, N. M. (2006). *Pattern recognition and machine learning* (Vol. 4, No. 4, p. 738). New York: springer.

Bizal, Ö. (2014). Makine Öğrenmesi Teknikleriyle Parkinson Hastalığının Belirlenmesi.

Breiman, L. (1996). Bagging predictors. *Machine learning*, *24*, 123-140.

Buyrukoğlu, S. (2021). Early detection of Alzheimer's disease using data mining: comparison of ensemble feature selection approaches. *Konya Mühendislik Bilimleri Dergisi*, *9*(1), 50-61.

Canbay, Y., (2013). Diyabet verilerinin destek vektör makineleri kullanılarak sınıflandırılması (Master's thesis, Erciyes Üniversitesi, Fen Bilimleri Enstitüsü).

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

Coşar, M., & Deniz, E. (2021). Makine Öğrenimi Algoritmaları Kullanarak Kalp Hastalıklarının Tespit Edilmesi. *Avrupa Bilim ve Teknoloji Dergisi*, (28), 1112-1116.

Delen, D., Walker, G., & Kadam, A. Predicting breast cancer survivability: a comparison of three data mining methods. *Artificial intelligence in medicine*, *34*(2), 2005, 113-127.

Emon, M. U., Islam, R., Keya, M. S., & Zannat, R. (2021, January). Performance Analysis of Chronic Kidney Disease through Machine Learning Approaches. In *2021 6th International Conference on Inventive Computation Technologies (ICICT)* (pp. 713-719). IEEE.

Esmer, S., Uçar, M. K., İbrahim, Ç. İ. L., & Bozkurt, M. R. (2020). Parkinson hastalığı teşhisi için makine öğrenmesi tabanlı yeni bir yöntem. *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, *8*(3), 1877-1893.

Fazakis, N., Kocsis, O., Dritsas, E., Alexiou, S., Fakotakis, N., & Moustakas, K. (2021). Machine learning tools for long-term type 2 diabetes risk prediction. *IEEE Access*, *9*, 103737-103757.

Görgün, M., (2020), *Makine öğrenmesi yöntemleriyle kalp hastalıklarının tahmin edilmesi* (Master's thesis, İstanbul Aydın Üniversitesi Lisansüstü Eğitim Enstitüsü).

Halis, Y., (2022).Makine Öğrenmesi Yöntemleri İle Kalp Krizi Tahminleme Beykent Üniversitesi Yüksek Lisans Projesi

Jacob, S. G., & Ramani, R. G. (2011). Discovery of knowledge patterns in clinical data through data mining algorithms: Multi-class categorization of breast tissue data. *International Journal of Computer Applications*, *32*(7), 46-53.

Karakoyun, M., & Hacibeyoğlu, M. (2014). Biyomedikal Veri Kümeleri İle Makine Öğrenmesi Siniflandirma Algoritmalarinin İstatistiksel Olarak Karşilaştirilmasi. *Dokuz Eylül Üniversitesi Mühendislik Fakültesi Fen ve Mühendislik Dergisi*, *16*(48), 30-42.

Karslı, Ö. B. (2019). *Makine öğrenme yöntemleri ile karaciğer hastalığının teşhisi* (Master's thesis, Ağrı İbrahim Çeçen Üniversitesi, Fen Bilimleri Enstitüsü).

Kaya, Y., & Uyar, M. (2013). A hybrid decision support system based on rough set and extreme learning machine for diagnosis of hepatitis disease. *Applied Soft Computing*, *13*(8), 3429-3438.

Kılıçarslan, S., Kemal, A. D. E. M., & Cömert, O. (2019). Parçacık sürü optimizasyonu kullanılarak boyutu azaltılmış mikrodizi verileri üzerinde makine öğrenmesi yöntemleri ile prostat kanseri teşhisi. *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, *7*(1), 769-777.

Kilinc, D., Borandag, E., Yucalar, F., Tunali, V., Simsek, M., & Ozcift, A. (2016). Classification of scientific articles using text mining with kNN algorithm and R language. *Marmara Journal of Pure and Applied Sciences*, *3*, 89-94.

Lahouar, A., & Slama, J. B. H. (2015). Day-ahead load forecast using random forest and expert input selection. *Energy Conversion and Management*, *103*, 1040-1051.

Langs, G., Röhrich, S., Hofmanninger, J., Prayer, F., Pan, J., Herold, C., & Prosch, H. (2018). Machine learning: from radiomics to discovery and routine. *Der Radiologe*, *58*(Suppl 1), 1.

Li, L., Qin, L., Xu, Z., Yin, Y., Wang, X., Kong, B., ... & Xia, J. (2020). Using artificial intelligence to detect COVID-19 and community-acquired pneumonia based on pulmonary CT: evaluation of the diagnostic accuracy. *Radiology*, *296*(2), E65-E71.

Malladi, R., Vempaty, P., & Pogaku, V. (2021). WITHDRAWN: Advanced machine learning based approach for prediction of skin cancer.

Manikandan, T., & Bharathi, N. (2016). Lung cancer detection using fuzzy auto-seed cluster means morphological segmentation and SVM classifier. *Journal of medical systems*, *40*, 1-9.

Niu, D., Pu, D., & Dai, S. (2018). Ultra-short-term wind-power forecasting based on the weighted random forest optimized by the niche immune lion algorithm. *Energies*, *11*(5), 1098.

Öztürk, K., & Şahin, M. E. (2018). Yapay sinir ağları ve yapay zekâ'ya genel bir bakış. *Takvim-i Vekayi*, *6*(2), 25-36.

Pirim, A. G. H. (2006). Yapay zeka. *Yaşar Üniversitesi E-Dergisi*, *1*(1), 81-93.

Ryzhikova, E., Ralbovsky, N. M., Sikirzhytski, V., Kazakov, O., Halamkova, L., Quinn, J., ... & Lednev, I. K. (2021). Raman spectroscopy and machine learning for biomedical applications: Alzheimer's disease diagnosis based on the analysis of cerebrospinal fluid. *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy*, *248*, 119188.

Sengur, A. (2012). Support vector machine ensembles for intelligent diagnosis of valvular heart disease. *Journal of medical systems*, *36*, 2649-2655.

Sivasangari, A., Reddy, B. J. K., Kiran, A., & Ajitha, P. (2020, October). Diagnosis of liver disease using machine learning models. In *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)* (pp. 627-630). IEEE.

Takhti, S. B., & Jahantigh, F. F. (2019). A model for diagnosis of kidney disease using machine learning techniques. *Razi Journal of Medical Sciences*, *26*(8), 14-22.

Tapkan, P., Özbakır, L., & Baykasoğlu, A. (2011). Weka ile veri madenciliği süreci ve örnek uygulama. *Endüstri Mühendisliği Yazılımları ve Uygulamaları Kongresi*, *30*, 247-262.

Tayeb, S., Pirouz, M., Sun, J., Hall, K., Chang, A., Li, J., ... & Latifi, S. (2017, December). Toward predicting medical conditions using k-nearest neighbors.

In *2017 IEEE International Conference on Big Data (Big Data)* (pp. 3897-3903). IEEE.

Temurtas, H., Yumusak, N., & Temurtas, F. (2009). A comparative study on diabetes disease diagnosis using neural networks. *Expert Systems with applications*, *36*(4), 8610-8615.

Toğacar, M., Ergen, B., & Sertkaya, M. E. (2019). Zatürre Hastalığının Derin Öğrenme Modeli ile Tespiti. *Firat University Journal of Engineering*, *31*(1).

Topal, Ç. (2017). Alan Turing'in toplum bilimsel düşünü: toplumsal bir düş olarak yapay zekâ. *Ankara Üniversitesi Dil ve Tarih-Coğrafya Fakültesi Dergisi*, *57*(2), 1340-1364.

Vassallo, D., Krishnamurthy, R., & Fernando, H. J. (2021). Utilizing physics-based input features within a machine learning model to predict wind speed forecasting error. *Wind Energy Science*, *6*(1), 295-309.

Veranyurt, Ü., Deveci, A., Esen, M. F., & Veranyurt, O. (2020). Makine Öğrenmesi Teknikleriyle Hastalık Sınıflandırması: Random Forest, K-Nearsest Neighbour ve Adaboost Algoritmaları Uygulaması. *Uluslararası Sağlık Yönetimi ve Stratejileri Araştırma Dergisi*, *6*(2), 275-286.

İNTERNET KAYNAKLARI

Acıbadem, (2023), https://www.acibadem.com.tr/acibadem-tr/diyabet-tedavi/#genel-rapor,Erişim Tarihi: 05/02/2023

Bayramlı, B., (2023), https://burakbayramli.github.io/dersblog/stat/**,** Erişim Tarihi: 02/04/2023

Bilen, B.,(2023), https://burhanbilen.medium.com/, Erişim Tarihi: 02/04/2023

Celik, (2023), https://www.dralpercelik.com/diyabette-deri-hastaliklari-ve-cilt-sorunlari/, Erişim Tarihi: 05/02/2023

Celik, (2023), https://www.dralpercelik.com/diyabet-ve-hipertansiyon/, Erişim Tarihi: 05/02/2023

Datasciencearth, (2022), https://www.datasciencearth.com/algorithm-naive-bayes-classifier/, Erişim Tarihi: 14/02/2022

Kaggle, (2022), https://www.kaggle.com/fmendes/diabetes-from-dat263x-lab01,ErişimTarihi: 20/01/2022

Medikal Park, (2023), https://www.medicalpark.com.tr/seker-hastaligi-diyabet-nedir/hg-1703#:~:text=Sa%C4%9Fl%C4%B1kl%C4%B1%20bireylerde%20a%C3%A7l%C4%B1k%20kan%20%C5%9Fekeri,uygulanarak%20tokluk%20kan%20%C5%9Fekeri%20ara%C5%9Ft%C4%B1r%C4%B1l%C4%B1r, Erişim Tarihi: 05/02/2023

Medium, (2023), https://medium.com/deep-learning-turkiye/ , Erişim Tarihi: 02/04/2023

Ozturk, M., (2023), https://miracozturk.com/python-ile-siniflandirma-analizleri-rastgele-orman-random-forest-algoritmasi/, Erişim Tarihi: 02/04/2023

Sakal, (2023), http://muratsakal.com/?p=230, Erişim Tarihi:02/04/2023

Türk Diyabet Vakfı, (2023), https://www.turkdiab.org/diyabet-hakkinda-hersey.asp?lang=TR&id=46 , Erişim Tarihi: 05/02/2023

Veri Bilimi Okulu, (2023), https://www.veribilimiokulu.com/yapay-sinir-agiartificial-neural-network-nedir/ , Erişim Tarihi: 20/03/2023

Vural, (2023), https://www.drerolvural.com/obezite-ve-diyabet-iliskisi/, Erişim Tarihi: 05/02/2023