TELEKOMÜNİKASYON SEKTÖRÜNDE ÇALIŞAN KAYBI TAHMİNİ İÇİN MAKİNE ÖĞRENMESİ MODELİ SEÇİMİ

Büşra UZAK

T.C.
BURSA ULUDAĞ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

TELEKOMÜNİKASYON SEKTÖRÜNDE ÇALIŞAN KAYBI TAHMİNİ İÇİN MAKİNE ÖĞRENMESİ MODELİ SEÇİMİ

Büşra UZAK
0000-0003-0797-5364

Doç. Dr. Betül YAĞMAHAN
(Danışman)

YÜKSEK LİSANS TEZİ
ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI

BURSA – 2022

TEZ ONAYI

Büşra UZAK tarafından hazırlanan "Telekomünikasyon Sektöründe Çalışan Kaybı Tahmini İçin Makine Öğrenmesi Modeli Seçimi" adlı tez çalışması aşağıdaki jüri tarafından oy birliği ile Bursa Uludağ Üniversitesi Fen Bilimleri Enstitüsü Endüstri Mühendisliği Anabilim Dalı'nda YÜKSEK LİSANS olarak kabul edilmiştir.

Danışman: Doç. Dr. Betül YAĞMAHAN

Başkan :	Doç. Dr. Betül YAĞMAHAN 0000-0003-1744-3062 Uludağ Üniversitesi, Mühendislik Fakültesi, Endüstri Mühendisliği Anabilim Dalı	İmza
Üye :	Doç. Dr. Aslı AKSOY 0000-0002-2971-2701 Uludağ Üniversitesi, Mühendislik Fakültesi, Endüstri Mühendisliği Anabilim Dalı	İmza
Üye :	Doç. Dr. Zehra Kamışlı ÖZTÜRK 0000-0003-3156-6464 Eskişehir Teknik Üniversitesi, Mühendislik Fakültesi, Yöneylem Araştırması Anabilim Dalı	İmza

Yukarıdaki sonucu onaylarım
Prof. Dr. Hüseyin Aksel EREN
Enstitü Müdürü
../../….

B.U.Ü. Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmasında;

- tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- ve bu tezin herhangi bir bölümünü bu üniversite veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

16/05/2022

Büşra UZAK

TEZ YAYINLANMA
FİKRİ MÜLKİYET HAKLARI BEYANI

Enstitü tarafından onaylanan lisansüstü tezin/raporun tamamını veya herhangi bir kısmını, basılı (kâğıt) ve elektronik formatta arşivleme ve aşağıda verilen koşullarla kullanıma açma izni Bursa Uludağ Üniversitesi'ne aittir. Bu izinle Üniversiteye verilen kullanım hakları dışındaki tüm fikri mülkiyet hakları ile tezin tamamının ya da bir bölümünün gelecekteki çalışmalarda (makale, kitap, lisans ve patent vb.) kullanım hakları tarafımıza ait olacaktır. Tezde yer alan telif hakkı bulunan ve sahiplerinden yazılı izin alınarak kullanılması zorunlu metinlerin yazılı izin alınarak kullandığını ve istenildiğinde suretlerini Üniversiteye teslim etmeyi taahhüt ederiz.

Yükseköğretim Kurulu tarafından yayınlanan "**Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına İlişkin Yönerge**" kapsamında, yönerge tarafından belirtilen kısıtlamalar olmadığı takdirde tezin YÖK Ulusal Tez Merkezi / B.U.Ü. Kütüphanesi Açık Erişim Sistemi ve üye olunan diğer veri tabanlarının (Proquest veri tabanı gibi) erişimine açılması uygundur.

Öğrencinin Adı-Soyadı: Büşra UZAK

Tarih: 16/05/2022

İmza

Bu bölüme öğrenci tez teslimi sırasında el yazısı ile okudum anladım yazmalı ve imzalamalıdır.

ÖZET

Yüksek Lisans Tezi

TELEKOMÜNİKASYON SEKTÖRÜNDE ÇALIŞAN KAYBI TAHMİNİ İÇİN MAKİNE ÖĞRENMESİ MODELİ SEÇİMİ

Büşra UZAK

Bursa Uludağ Üniversitesi
Fen Bilimleri Enstitüsü
Endüstri Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Betül YAĞMAHAN

Günümüzde işletmelerin en büyük problemlerinden biri yetiştirdikleri çalışanları kaybetmeleridir. Çalışan kaybının şirketlere birçok maliyeti vardır. Bu nedenle çalışan kaybının tahmin edilmesi ve önlem alınması oldukça önem taşımaktadır. Bu kadar önemli bir konuda yapılan tahminlerin doğruluğu da alınacak aksiyonların hatalı olmaması ve çalışan kaybının azaltılması için oldukça önemlidir. Birçok tahmin yöntemi bulunmaktadır ancak bu çalışmada makine öğrenmesi yöntemlerinden olan sınıflandırma yöntemi kullanılarak telekomünikasyon sektörüne ait bir çalışan veri seti analiz edilmiştir. Çalışmanın amacı sekiz adet sınıflandırma modeli ile veri setinin analiz edilerek bu problem için en uygun sınıflandırma modelinin önerilmesidir. Bu uygulamaya ait modeller Python dili ile kodlanmıştır. Veri kümesinin %70'i modelin eğitilmesinde ve doğrulanmasında, %30'u ise modelin test edilmesinde kullanılmıştır. Uygulanan modeller doğruluk, çapraz doğrulama skoru, kesinlik, duyarlılık, f_1 skoru ve Eğri Altında Kalan Alan (EAKA) metriklerine göre değerlendirilmiştir. Kullanılan modeller arasında en iyi sınıflandırma modeli %92,2 doğruluk değeri ile rastgele orman modeli olarak bulunmuştur. İkinci en iyi model ise %91,4 doğruluk değeri ile gradyan artırma makineleri modeli olarak bulunmuştur. Bu veri setini uygulanan modeller arasında %89,1 doğruluk oranı ile en kötü sınıflandıran model ise k-en yakın komşu olmuştur.
Problem özelinde gelecekte yapılacak sınıflandırma çalışmaları için bu çalışmada uygulanan modeller değerlendirildiğinde en iyi metrik değerlerine ulaşılan yani en iyi sınıflandıran rastgele orman modeli önerilmektedir.

Anahtar Kelimeler: Makine Öğrenmesi, Sınıflandırma, Çalışan Kaybı, Tahminleme, Doğruluk değeri
2022, vii +72 sayfa.

ABSTRACT

MSc Thesis

CHOOSING MACHINE LEARNING MODEL FOR PREDICTING EMPLOYEE CHURN IN THE TELECOMMUNICATION INDUSTRY

Büşra UZAK

Bursa Uludağ University
Graduate School of Natural and Applied Sciences
Department of Industrial Engineering

Supervisor: Assoc. Prof. Dr. Betül YAĞMAHAN

One of the biggest problems of businesses today is losing their employees. Employee churn has many costs to companies. For this reason, it is very important to predict the loss of employees and take precautions. The accuracy of the estimates made on such an important issue is also very important to ensure that the actions to be taken are not erroneous and to reduce the churn of employees. There are many estimation methods, but in this study, an employee data set belonging to the telecommunications sector was analyzed by using the classification method, which is one of the machine learning methods. The aim of the study is to analyze the data set with eight classification models and to propose the most suitable classification model for this problem. These models are coded with Python language. 70% of the dataset was used in training and validation the model and 30% in testing the model. The applied models were evaluated according to accuracy, cross validation score, precision, sensitivity, f_1 score and Area Under the Curve (AUC) metrics. Among the models used, the best classification model was found to be the random forest model with an accuracy of 92.2%. The second best model was found to be the gradient increasing machines model with an accuracy value of 91.4%. k nearest neighbor is the worst classifying model among the applied models with an accuracy rate of 89.1%.
When the models applied in this study are evaluated for the classification studies to be carried out in the future, the random forest model, which has the best metric values, that is, which classifies the best, is recommended.

Key words: Machine Learning, Classification, Employee Churn, Prediction, Accuracy
2022, vii + 72 pages.

ÖNSÖZ ve TEŞEKKÜR

Yüksek lisans eğitimim boyunca ve bitirme tezi çalışma sürecimde yardımını ve desteğini hiç esirgemeyen kıymetli danışman hocam Doç. Dr. Betül YAĞMAHAN' a sonsuz şükranlarımı sunarım.

Tüm eğitim hayatım boyunca bilgi ve donanımlarından yararlandığım ve her türlü zahmetine rağmen pes etmemem için beni sürekli motive eden değerli hocalarıma, beni hayat boyu her türlü kararımda destekleyen, maddi manevi yanımda olan ve ömrümün gizli kahramanları olan anneme, babama, abime ve hep rol model aldığım en iyi arkadaşım ve en iyi destekçim olan ablama ayrıca teşekkür ederim.

Büşra UZAK
16/05/2022

İÇİNDEKİLER

Sayfa

SİMGELER ve KISALTMALAR DİZİNİ

Simgeler	**Açıklama**
ε	Epsilon
t	Düğüm
$g(t)$	Gini Katışıklık İndeksi
i	Bağımsız Değişken Sınıfı
j	Bağımlı Değişken Sınıfı
k	Komşu Sayısı
$t_{sağ}$	Sağ Düğüm
t_{sol}	Sol Düğüm
$p_{sağ}$	Sağ Taraftaki Bağımlı *t* Düğümündeki Durumların Oranı
p_{sol}	Sol Taraftaki Bağımlı *t* Düğümündeki Durumların Oranı
$\Phi(s,t)$	*s* Örneklemindeki *t* Düğümlerinin Ayrılma İndeksi
z	Regresyon Denklemi Çıktısı
x_i	*i.* Bağımsız Değişken
a_i	*i.* Bağımsız Değişkenin Regresyon Katsayısı
$P(z)$	Bir Girdi Noktasının Bir Sınıfa Ait Olma Olasılığı
y_i	Sınıf Etiketi
w	Ağırlık Vektörü
b	Eğilim Değeri
$d1$	Gini İndeksi İlk Veri Noktası
d2	Gini İndeksi İkinci Veri Noktası
N_s	Gizli Katmandaki Nöron Sayısı
N_g	Girdi Katmanındaki Nöron Sayısı
$N_ç$	Çıktı Katmanındaki Nöron Sayısı
N_d	Gözlem Sayısı
N_b	Katman Sayısı

Kısaltmalar	**Açıklama**
ACC	Accuracy (Doğruluk)
AGA	Aşırı Gradyan Artırma
AİK	Alıcı İşlem Karakteristikleri
CSV	Cross Validation (Çapraz Doğrulama)
ÇDS	Çapraz Doğrulama Skoru
DDA	Doğrusal Diskriminant Analiz
DSA	Derin Sinir Ağları
DVM	Destek Vektör Makinesi
EAKA	Eğri Altında Kalan Alan
GAA	Gradyan Artırma Ağaçları
IBM	International Business Machines
İK	İnsan Kaynakları
KA	Karar Ağacı
k-EYK	k-En Yakın Komşu
LR	Lojistik Regresyon

NB	Naive Bayes
OEA	Otomatik Ekileşim Algılama
RO	Rastgele Orman
SRA	Sınıflandırma ve Regresyon Ağacı
SSA	Sığ Sinir Ağları
SSAGH	Stokastik Sinirsel Analog Güçlendirme Hesaplayıcısı
YSA	Yapay Sinir Ağları

ŞEKİLLER DİZİNİ

Sayfa

ÇİZELGELER DİZİNİ

1. GİRİŞ

Günümüzde işletmelerin en büyük problemlerinden biri yetiştirdikleri çalışanları kaybetmeleridir. Çalışan kayıplarının şirketlere birden fazla maliyeti olmaktadır. Kaybedilen çalışanın yerine aynı becerilere sahip yeni bir çalışan arama eforu işe alım ekiplerine düşmektedir. Daha sonra bulunan yeni çalışana işin aktarılması ile bir zaman kaybı oluşmaktadır. Eğer şirkete özgü işleyen bir yapı varsa bu yapı özelinde bir eğitim verilmesi gerekebilir, bu da başka bir maliyeti oluşturur. Tüm bunlara ek olarak hali hazırda yürütülen projelerin bu gibi çalışan kaybı durumlarında termin tarihine yetişmesi için çalışanların yedeklenmesi de ayrıca iş gücü ve zaman kaybıdır. Tüm bu iş gücü, eğitim ve zaman kaybı maliyetleri bu konunun sürekli takip edilmesini gerektirmektedir. Bu nedenle şirketler çalışan kayıplarını ve maliyetleri öngörerek azaltmak ve önlem almak için çeşitli tahmin yöntemleri kullanmaktadır. Birçok tahmin yöntemi bulunmaktadır ancak günümüzde tahmin problemleri en çok makine öğrenmesi yöntemleri ile çözülmektedir.

Makine öğrenmesi yöntemleri ile tahminleme yapılırken kullanılacak yöntem verinin özelliklerine göre belirlenir. Özellikle çalışan veya müşteri kaybı problemlerinde veri setinin büyük çoğunluğu kategorik veriden oluştuğu için genellikle sınıflandırma yöntemi ile tahmin çalışması yapılmaktadır.

Bu çalışmada ise telekomünikasyon sektöründeki çalışan kaybını tahmin etmek için kullanılan makine öğrenmesi modelleri kıyaslanarak en iyi modelin önerilmesi amaçlanmaktadır.

Çalışmanın ikinci bölümünde literatür araştırması yapılmıştır. Ayrıca çalışma kapsamında kullanılan sınıflandırma modelleri açıklanmıştır. Üçüncü bölümünde çalışmanın yapıldığı ortam, kullanılan veri seti ve yöntemlerin parametreleri detaylı açıklanmıştır. Dördüncü bölümde uygulanan model sonuçlarının incelenmesi ve tüm modellerin karşılaştırılması yapılmıştır. Beşinci bölümde ise en iyi sınıflandırma modeli belirlenerek gelecekteki çalışmalar için öneride bulunulmuştur.

2. KURAMSAL TEMELLER ve KAYNAK ARAŞTIRMASI

Bu bölümde tez çalışmasında uygulanması amaçlanan modellerin daha önceki çalışmalarda ne şekilde kullanıldığını öğrenmek ve hangi makine öğrenmesi modelinin hangi amaçlarla kullanıldığını anlamak amacıyla kuramsal temeller ve kaynak araştırmalarından bahsedilmektedir.

2.1. Makine Öğrenmesi

Makine öğrenmesi ifadesi ilk defa 1950 senesinde Turing tarafından makinelerin insanlar gibi düşünebilme özelliğinin mümkün olup olmadığının araştırılması ile ortaya çıkmıştır (Turing, 1950). Bu fikrin sorgulanması ile makinelere farklı yetkinlikler kazandırılarak uygulama kapsamları büyütülmüştür. Aynı zaman diliminde geliştirilen yapay sinir ağları temelli ilk bilgisayar Stokastik Sinirsel Analog Güçlendirme Hesaplayıcısı ve IBM'de geliştirilen satranç oyunu makine öğrenmesi çalışmalarının ilk örneklerindendir (McCarthy ve Feigenbaum, 1990; Crevier, 1993). Makine öğrenmesinin keşfi ve 1950 sonrasında yapay zeka ve derin öğrenme kavramları da ortaya atılarak makine öğrenmesi modelleri desteklenmiştir.

1990 yılında ise kendi kendini organize eden haritalar yöntemi ileri sürülmüştür (Kohonen, 1990). Özellikle 1990 yılı ve sonrasında insanların daha kolay ulaşılabildiği teknoloji ile birlikte analiz edilmeyi bekleyen çok fazla veri toplanabilmiştir. Büyük boyutlu verilerin toplanması ve sorunsuz saklanabilmesi ise ilerleyen yıllarda verilerin anlamlı sonuçlar üretilmek için kullanılması ihtiyacını oluşturmuştur. Veri madenciliği ile analiz edilerek işlenen verilerle anlamlı sonuçlar elde edilmiştir. Üretilen makine öğrenmesi algoritmalarıyla da makinelerin öğrenebileceği bilgiler belirli amaçlara yönelik hizmet vermesi için makineler tarafından kodlanmıştır. E-posta da spam filtre özelliği gibi temel seviye uygulamalarda kullanılan makine öğrenmesi modelleri günümüzde çok farklı alanlarda kullanılmaktadır. Örneğin; sağlık alanında önceki hastaların şikâyetlerine bakılarak yeni hastaların hastalıklarının tahmini, birliktelik analizi ile alışveriş merkezlerinde birbiri ile alakasız ama birlikte en çok satılan ürünlerin yakın konumlandırılması, akıllı saat ya da çeşitli cihazlarla nabız kontrolünün yapılması, adım sayılarının analiz edilerek hareket oranına göre egzersiz önerilmesi gibi örnekler

verilebilir. İnsansız araba kullanımı ise en iyi makine öğrenmesi uygulama alanlarındandır. Yapay zeka sistemi ile entegre edilen arabalarda anlık verilerin anlık analizi ile insansız hareket kabiliyeti kazanması sağlanmıştır (Tekin vd., 2018). Modelleme iki aşamadan oluşmaktadır. İlk aşamada modeli eğitmek için optimizasyon probleminin çözüme kavuşması gerekir. Verinin saklanabilirliği ve en az sürede analiz edilebilirliği değerlendirildiğinde optimum algoritmanın seçilmesi büyük önem taşımaktadır. İkinci aşamada model eğitildikten sonra elde edilen sonuçların anlamlı olması beklenmektedir. Bazı çalışmalarda sonucun anlamlı oluşundan daha çok veri işleme süresi ve verinin boyutu daha önemlidir (Alpaydin ve Bach, 2014).

Veriler eğitim ve test verisi olarak ikiye ayrılır. Özetle makine öğrenmesi modelleri eğitim verisini algoritma ile kullanarak karar verir ve üretilen matematiksel model ile analiz amacında belirlenen konuya yönelik alınacak kararlara destek olur. Üretilen model ile çıkarımlar sağlanır ve ilgili kişilere raporlanır (Bishop, 2006).

2.2. Makine Öğrenmesi Çeşitleri

Makine öğrenmesi, günümüzde çok popüler olan otomotiv, eğlence, fen, tıp ve pazarlama gibi çoğu alanda kullanılan, yapay zekânın bir alt dalıdır. Makine öğrenmesi genellikle tahmine dayalı analitik veya tahmine yönelik modelleme olarak da tanımlanır. Temel olarak, otomatik öğrenme ve geliştirme ilkesine dayanır (Yang,2019).

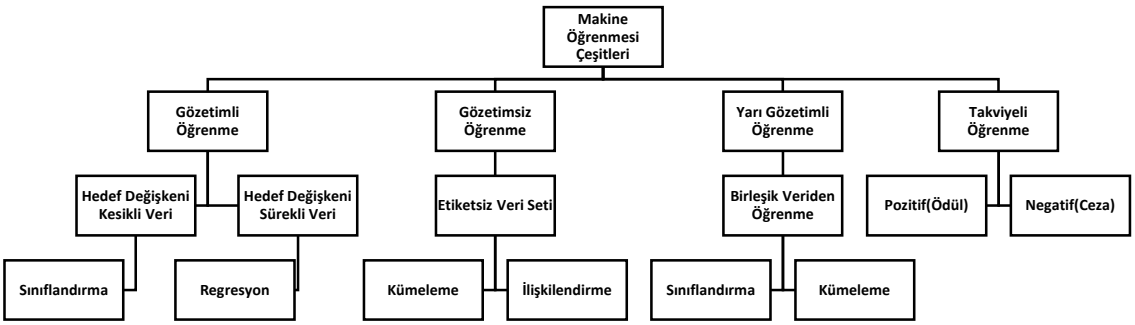
Makine öğrenmesi, *y* değişkenlerini kabul edilebilir bir aralıkta öngörmek için eğitim verilerini sınıflandıran algoritmalar kullanır. Bu algoritmalar modele yeni veri kümeleri iletilirken, performansı en iyilemek ve zaman içinde model zekâsı geliştirmek için operasyon sürecini öğrenir ve optimize ederler (Chen ve Jeng, 2011).

Büyük boyutlu karışık değişken ve veri tiplerinden oluşan bir veri setinin içerisinden anlamlı bir veri çıkarma sürecine veri madenciliği denilmektedir. Veri madenciliğinin bulunması en çok makine öğrenmesinin gelişimine fayda sağlamıştır. Makine öğrenmesi keşfinden bu zamana kadar teknikler sürekli iyileşmeye devam etmektedir. İyileştirilen

bu teknikler sayesinde farklı yöntemlerle bilime katkı sağlanmıştır. Faydalı olan bu yöntemler şu şekilde sıralanabilir (Şenol vd., 2020):

• Gözetimli Öğrenme
• Gözetimsiz Öğrenme
• Takviyeli Öğrenme
• Yarı Gözetimli Öğrenme

Makine öğrenmesi çeşitleri Şekil 2.1'de gösterilmektedir. Çeşitlerin hangi veri setlerine özel kullanılabileceği de şekilde detaylı biçimde belirtilmiştir.



Şekil 2.1. Makine öğrenmesi çeşitleri şeması (Sarker, 2021)

2.2.1. Gözetimli Öğrenme

Gözetimli öğrenme yaklaşımı temel olarak öğrenmedeki denetimden gelir. Yani kullanılan veri seti ile eğitilen modelin veri setindeki bağımlı değişken şeklinde ifade edilen etiket verisi ile modelin denetlenmesi ve doğruluğunun test edilmesi söz konusudur. Örneğin, evlerin renklerinin daha önce ev rengi bağımlı değişkeni ile ifade edilen bir etiket verisi ile modelin eğitilip sınıflandırılarak belirlenmesi gözetimli öğrenmeye örnek verilebilir (Blum ve Mitchell, 1998). Gözetimli öğrenme yöntemleri iki gruba ayrılır:

• Sınıflandırma Yöntemi
• Regresyon Yöntemi

2.2.2. Gözetimsiz Öğrenme

Gözetimsiz öğrenme yaklaşımı eğitim verisi sınıf etiketli olmadığı için öğrenme sürecini gözetmez. Bu yaklaşımda veri içindeki sınıfların bulunması için kümeleme uygulanabilmektedir. Örneğin, denetlenmeyen bir öğrenme metodu eğitim verisi olarak el ile yazılan rakam ifadelerinin görüntüsünü alıp 0'dan 10'a kadar oluşturulan rakam kümeleri olduğunu kabul eder ve her bir rakamı 10 kümeden birine atadığında bu atama işleminin doğruluğunu denetlemez. Çünkü herhangi bir etiketli değişken veri kümesinde bulunmamaktadır (Chapelle vd., 2006). Gözetimsiz öğrenme yöntemleri iki gruba ayrılır:

• Kümeleme
• Boyut Azaltma, İlişkilendirme

2.2.3. Yarı Gözetimli Öğrenme

Bu yaklaşımda kullanılan eğitim verilerinin büyük bir kısmını gözetimsiz öğrenme yönteminde olduğu gibi etiketlenmemiş girdi öznitelikleri oluşturmaktadır. Buna rağmen veri kümesinde az miktarda bağımlı değişken değeri ile kaydedilmiş veri de bulunmaktadır. Etiket verisi bulunan bağımlı değişken değerlerine ulaşmanın maliyetli ve zor olduğu yerlerde bu yöntem kullanılır (Mohri ve Medina, 2012). Yarı gözetimli öğrenme yöntemleri üç gruba ayrılır:

• Sınıflandırma Yöntemi
• Regresyon Yöntemi
• Sıralama Yöntemi

2.2.4. Takviyeli Öğrenme

Takviyeli öğrenme, gözetimli öğrenme ve gözetimsiz öğrenmenin dışında kalan üç ana makine öğrenmesi yöntemlerinden biridir. Takviyeli öğrenme, etiketli giriş ve çıkış verisi çiftlerinin olmasına ihtiyaç duyulmamasından dolayı gözetimli öğrenmeden farklıdır. Gözetimli öğrenmeye kıyasla keşfedilmemiş bölgelerin araştırılması ile mevcut girdi bilgisinin kullanımı arasında bir denge oluşturmayı amaçlamaktadır. Takviyeli öğrenme,

robot kontrolü, telekomünikasyon, asansör sırası, tavla ve satranç gibi çeşitli problem alanlarına uygulanabilir. Uygulama yöntemleri aşağıdaki gibi ikiye ayrılır:

• Sınıflandırma Yöntemi
• Kontrol

2.3. Makine Öğrenmesi Model Metrikleri

Literatür araştırması yapıldığında birçok makine öğrenmesi algoritması bulunmaktadır. Bu algoritmalardan bazıları; yapay sinir ağları (YSA), k-EYK, DVM, KA, LR analizi ve gradyan artırma ağacı (GAA) algoritmasıdır. Bu algoritmalardan bir kısmı kümeleme ve sınıflandırma yaparken bir kısmı bağımlı değişkeni tahmin etmek için kullanılmaktadır. Makine öğrenmesi modelleri parametre ve hiper düzlemler ile kurulur.

2.3.1. Hiper Parametre

Öğrenim modelleri kurulurken kullanılan parametreler, modelin öğrenim süresindeki verilerden çıkarılabilen ve modeli tasarlayan tarafından önceden öngörülebilen tanımlanabilen olmak üzere iki gruba ayrılır. Bunlar model parametresi ve hiper parametrelerdir. Model parametreleri şeklinde ifade edilen parametreler genellikle veri kümesinden tahmin elde edebilen veya öğrenebilen parametrelerdir. Bu parametrelerde modeli tasarlayan kişinin değiştirme yapması beklenmez. Öğrenilebilen modelin bir kısmı olarak kaydedilir. YSA modelindeki sabitler, DVM destekleyici vektörleri, DR veya LR modellerindeki sabitler ve benzeri parametreler model parametrelerine örnek gösterilebilir. Model parametrelerinden farklı olarak hiper parametreler, veri kümesinden öngörülemez ve modeli tasarlayan kişinin değişikliği gerekebilir (Tanyıldızı ve Demirtaş, 2019).

2.3.2. Hiper Düzlem

Bazı makine öğrenmesi algoritmalarında amaç *n* ölçekli bir uzayda verinin noktalarını belirgin bir şekilde ifade eden bir düzlem bulmaktır. Bu düzleme hiper düzlem denir. İki veri sınıfını ayırmak için kullanılabilecek birden fazla mümkün hiper düzlem vardır. Bu

düzlemler arasından marjı maksimum olan, yani her iki sınıfın veri noktaları arasında maksimum mesafeye sahip bir düzlem bulmak amaçlanmaktadır (Kumaş, 2021).

2.3.3.Parametre Optimizasyonu

Birçok hiper parametrenin alabileceği değer aralığı sonsuz veya çok geniştir. Ancak daha önce benzer problemlerden edinilen tecrübelere dayanarak da hiper parametrelerin alabileceği değerler için makul bir aralık belirlemek mümkündür. Bu aralıkta bazı ana noktalar belirlenir ve bir hiper parametre için bir değer listesi oluşturulur. Belirlenen aralıktaki değerlerin kombinasyonlarını hiper parametrelere atayarak modeli eğitip sonuçlarını değerlendiren bir metot kullanılır. Bu metoda grid arama metodu denir. Grid arama metodu en iyi sonucu veren kombinasyonu bulur. Bu metodun optimizasyon için tercih edilmesinin en önemli sebeplerinden biri, paralel işlemler koşturarak zamandan tasarruf sağlamasıdır (Keleş vd., 2020).

2.3.4. Sınıflandırma Modelleri Performans Metrikleri

Bu çalışmada da probleme uygun olarak bazı sınıflandırma yöntemleri kullanılmaktadır. Bu yöntemlerin ölçümlenebilmesi için çeşitli parametreler vardır. Bazı durumlarda pozitif sınıfta yer alan bir örnek, tahminde de pozitif olarak sınıflandırılabilir. Bu durum Doğru Pozitif (DP) ayrılma olarak adlandırılır. Diğer taraftan, pozitif sınıfta olan bir örnek negatif sınıfa aitmiş gibi tahmin edilmiş olabilir. Bu durum Yanlış Negatif (YN) ayrılma olarak adlandırılır. YN ayrılma aslında 2. tip hatadır. Negatif sınıfta yer alan bir örnek negatif sınıfta sınıflandırılırsa Doğru Negatif (DN) ayrılma olarak adlandırılırken yine negatif sınıfta yer alan bir örnek pozitif sınıfa ait gibi tahmin edilirse Yanlış Pozitif (YP) ayrılma olarak isimlendirilir. YP ayrılma 1. tip hataya işaret eder. Sınıflandırıcı için DP ve YP iki önemli değerlendirme ölçütüdür (Doğan vd., 2021).Çizelge 2.1, bu ifadelerle oluşturulan hata matrisini gösterir.

Çizelge 2.1. Hata Matrisi

		Gerçek Sınıf	
		Pozitif(Pasif)	Negatif(Aktif)
Tahmin edilen sınıf	Pozitif(Pasif)	DP	YP
	Negatif(Aktif)	YN	DN

DPO (Doğru Pasif Oranı): Gerçek pasiflerin pasif olarak sınıflandırılanlara oranını verir ve denklem 2.1'de görüldüğü gibi hesaplanır.

$$DPO = \frac{DP}{DP + YN} \tag{2.1}$$

YPO (Yanlış Pasif Oranı): Gerçek aktiflerin pasif olarak sınıflandırılanlara oranını verir ve denklem 2.2'de görüldüğü gibi hesaplanır.

$$YPO = \frac{YP}{YP + DN} \tag{2.2}$$

DNO (Doğru Aktif Oranı): Gerçek aktiflerin aktif olarak sınıflandırılanlara oranını verir ve denklem 2.3'de görüldüğü gibi hesaplanır.

$$DNO = \frac{DN}{DN + YP} \tag{2.3}$$

YNO (Yanlış Aktif Oranı): Gerçek pasiflerin aktif olarak sınıflandırılanlara oranını verir ve denklem 2.4'de görüldüğü gibi hesaplanır.

$$YNO = \frac{YN}{YN + DP} \tag{2.4}$$

Duyarlılık ve Kesinlik: Gerçek pozitif oranı olarak adlandırılan duyarlılık ve gerçek negatif oranı olarak adlandırılan özgüllük oranı sınıflandırıcının pozitif ve negatif sınıfları nasıl ayırdığı hakkında önemli bilgiler sunar. Duyarlılık denklem 2.5'de, kesinlik denklem 2.6'da görüldüğü gibi hesaplanır.

$$Kesinlik = \frac{DP}{DP + YP} \tag{2.5}$$

$$Duyarlılık = \frac{DP}{DP + YN} \tag{2.6}$$

f_1 Ölçütü: Duyarlılık ve kesinliğin harmonik ortalamasıdır. F ölçütü denklem 2.7'de görüldüğü gibi hesaplanır.

$$F = \frac{2(Kesinlik * Duyarlılık)}{Kesinlik + Duyarlılık} \tag{2.7}$$

Eğri Altında Kalan Alan (EAKA): Bir sınıflandırma modelinin tanımsal tatmin ediciliğini kabul edebilmek için kullanılan kolay bir metot, performans ölçütünün ifadesidir (Obuchowski vd., 2004). En yaygın kullanılan ölçüm ise, alıcı işlem karakteristiği (AİK) eğrisinin altında kalan alandır (Obuchowski, 2005). EAKA ne kadar büyük ise istenilen sınıfın tahmin edilmesinde modelin performansı o kadar istenilen şekilde olur. EAKA' nın mümkün değerleri 0,5'ten 1'e kadar değişim gösterir (Grove, 2006). Bu bir modelin ayırt etme kabiliyetini anlatmanın çok etkili bir yoludur. Ayırt etme yeteneğine sahip bir model, AİK eğrisinin sol üst bölgeye eğimli olması modelin başarılı olduğunu gösterir.

Doğruluk Değeri: Hata matrisinde belirtilen ifadelerin kullanılmasıyla sınıflandırıcıların doğruluk değeri denklem 2.8'de görüldüğü gibi hesaplanır (Ecemiş vd., 2019).

$$Doğruluk = \frac{(DP + DN)}{(DP + YP + DN + YN)} \tag{2.8}$$

Çapraz Doğrulama Skoru (ÇDS): Farklı kombinasyonlarla model eğitilerek çalıştırılır ve en yüksek doğruluk değerinin bulunması amaçlanır. Genellikle doğruluk değeri ile benzer sonuçlar elde edilir.

2.4. Sınıflandırma Yöntemleri

Bu kısımda sık kullanılan sınıflandırma yöntemleri detaylı bir şekilde anlatılmıştır.

2.4.1. Sınıflandırma ve Regresyon Ağacı Algoritması

SRA algoritması Morgan ve Sonquist'in Otomatik Ekileşim Algılama (OEA) isimli karar ağaçları modellerinin devamiyeti niteliğinde Breiman ve arkadaşları tarafından 1984 senesinde önerilmiştir (Breiman vd., 1984).

Breiman ve arkadaşları 1984 yılında, SRA yönteminin en popüler makine öğrenmesi yöntemlerinden biri olan parametrik olmayan bir regresyon tekniği olduğunu belirtmiştir. SRA algoritması, sözde bir karar ağacı oluşturmak için geçmiş verileri kullanan bir sınıflandırma yöntemidir. Hem nümerik hem de kategorik veri türlerini, bağımsız ve bağımlı değişken olarak kabul edebilen SRA algoritması, sınıflandırma ve regresyon problemlerinde bir çözüm olarak kullanılabilir. Aykırı değerlere karşı sağlamlık bu algoritmanın bir avantajıdır. Genellikle bölme algoritması, ayrı düğümlerdeki aykırı değerleri izole eder (Breiman vd., 1984).

SRA algoritmasını kullanan sınıflandırma ağacı, öğrenme örneğinin daha küçük parçalara bölünmesini gerçekleştiren bölme kuralına göre oluşturulmuştur. Maksimum homojenlik için her verinin iki parçaya bölünmesi gerekir. İlk hangi öznitelikten bölünebileceği ve bölünme değeri bölünme kriteri değeri incelenerek hesaplanır. Bölünme kriterlerinden biri olarak gini ölçütü değeri veri kümesindeki değişkenlerin oranı olarak tanımlanabilir. İki varlığın gini ölçütü değeri aynı çıkarsa çıktı dağılımları aynı demektir (Adak ve Yurtay, 2013).

SRA algoritmasının parametreleri, varsayılan değerleri ve parametrelerin tanımları Çizelge 2.2'de yer almaktadır.

Çizelge 2.2. Sınıflandırma ve regresyon ağacı parametreleri (Özkan,2012)

Sınıflandırma ve Regresyon Ağacı		
Parametreler	**Varsayılan Değerler**	**Tanım**
Bölünme Kriteri(criterion)	gini	Bir bölünmenin kalitesini ölçme fonksiyonudur.
Bölücü(splitter)	best	Her düğümde bölünmeyi seçmek için kullanılan stratejidir. En iyi bölmeyi seçmek için "best" yapılır.

Çizelge 2.2. Sınıflandırma ve regresyon ağacı parametreleri (devam)

Sınıflandırma ve Regresyon Ağacı		
Parametreler	**Varsayılan Değerler**	**Tanım**
Maksimum Derinlik(max_depth)	Hiçbiri	Ağacın maksimum derinliğini ifade eder. Hiçbiri ise, tüm yapraklar saf olana veya tüm yapraklar minimum örnek sayısından daha azını içerene kadar düğümler genişletilir.
Minimum Örnek Sayısı(min_samples_split)	2	Bir düğümün bölünmeden önce sahip olması gereken minimum örnek sayısıdır.
Minimum Yaprak Örnek Sayısı(min_samples_leaf)	1	Bir yaprağın sahip olması gereken minimum örnek sayısıdır.
Minimum Ağırlıklı Örnek Sayısı (min_weight_fraction_leaf)	0.0	Minimum yaprak örnek sayısıdır. Ağırlıklı örneklerin, toplam örnekler içerisindeki oranıdır. Ağacın dengeli gitmesi için kullanılır.
Maksimum Örnek Sayısı(max_features)	Hiçbiri	Maksimum örnek sayısını ifade eder.
Rastgele Durum(random_state)	Hiçbiri	Hem ağaç oluştururken kullanılan örneklerin önyüklemesinin rasgeleliğini hem de her düğümde en iyi bölünmeyi ararken dikkate alınacak özelliklerin örneklemesini kontrol eder.
Maksimum Yaprak Sayısı(max_leaf_nodes)	Hiçbiri	En iyi şekilde bir ağacın büyümesidir. Varsayılan olarak ayarlandığında sınırsız sayıda yaprak ve düğüm anlamına gelir.
Minimum Düğüm Safsızlığı (min_impurity_decrease)	0.0	Düğüm safsızlığı parametresi ağaçların öz nitelikleri(verileri) nasıl böldüğünü gösterir. İlk olarak ana düğümün düğüm safsızlığını (node impurity) hesaplanır, devamında bölme için belirli bir özellik kullanılacaksa alt düğümlerin düğüm safsızlıkları hesaplanır.
Sınıf Ağırlıkları (class_weight)	Hiçbiri	Formdaki sınıflarla ilişkili ağırlıkları belirlemeye yarayan parametredir.
Karmaşıklık (ccp_alpha)	0.0	Minimum maliyet-karmaşıklık budaması için kullanılan karmaşıklık parametresidir. Varsayılan olaraka ayarlandığında, budama yapılmaz.

2.4.2. Rastgele Orman Algoritması

RO, birden fazla karar ağacı oluşturan bir topluluk öğrenme yöntemidir. RO, daha güçlü bir öğrenici oluşturmak için bir grup zayıf öğreniciyi birleştirerek temel karar ağacı yapısı

üzerinde bir iyileştirme sağlayan bir topluluk yaklaşımı alır (Breiman, 2001). Topluluk yöntemleri, algoritma performansını iyileştirmek için böl ve yönet yaklaşımını kullanır. Topluluk öğrenme metodlarında birden fazla sınıflayıcının oluşturduğu sonuçlar birleşerek, topluluğu temsilen tek bir karara varılır (Breiman, 2001).

Algoritma, ön yükleme yaklaşımı üzerinden gözlemlerin bir alt kümesini uygulayan rastgele ikili ağaçlar çalıştırır, ilk veri kümesinin eğitim verilerinin rastgele bir seçimi seçilir ve modeli oluşturmak için uygulanır, dahil edilmeyen veriler torba dışı olarak tanımlanır (Catani vd., 2013). RO, bir değişkenin önemini, o değişken için torba dışı verilere izin verildiğinde, diğerleri sabit bırakılırken tahmin hatasının ne kadar arttığına bakarak tahmin eder (Liaw ve Wiener 2002; Catani vd., 2013).

Uygulama kısmında, model çalıştırılmadan önce farklı hiper parametrelerin tanımlanması gerekir. Kodlama ile parametre en iyilemeu yapılır. Uygun parametre aralıkları seçilerek, bu aralıklarda en iyi sonuç üreten parametre değerleri iteratif olarak belirlenebilmektedir.

RO algoritmasının parametreleri, varsayılan değerleri ve parametrelerin tanımları Çizelge 2.3'de yer almaktadır.

Çizelge 2.3. Rastgele orman algoritması parametreleri (Probst vd., 2019)

Rastgele Orman		
Parametreler	**Varsayılan Değerler**	**Tanım**
Tahmin Sayısı (n_estimators)	100	Ormandaki ağaç sayısıdır.
Bölünme Kriteri(criterion)	Gini	Bir bölünmenin kalitesini ölçme fonksiyonudur.
Maksimum Derinlik (max_depth)	Hiçbiri	Ağacın maksimum derinliğini ifade eder. Hiçbiri ise, tüm yapraklar saf olana veya tüm yapraklar minimum örnek sayısından daha azını içerene kadar düğümler genişletilir.
Minimum Örnek Sayısı (min_samples_split)	2	Bir düğümün bölünmeden önce sahip olması gereken minimum örnek sayısıdır.
Minimum Yaprak Örnek Sayısı(min_samples_leaf)	1	Bir yaprağın sahip olması gereken minimum örnek sayısıdır.
Minimum Ağırlıklı Örnek Sayısı(min_weight_fraction_leaf)	0.0	Minimum yaprak örnek sayısıdır. Ağırlıklı örneklerin, toplam örnekler içerisindeki oranıdır. Ağacın dengeli gitmesi için kullanılır.

Çizelge 2.3. Rastgele orman algoritması parametreleri (devam)

Rastgele Orman		
Parametreler	**Varsayılan Değerler**	**Tanım**
Tahmin Sayısı (n_estimators)	100	Ormandaki ağaç sayısıdır.
Bölünme Kriteri(criterion)	Gini	Bir bölünmenin kalitesini ölçme fonksiyonudur.
Maksimum Derinlik (max_depth)	Hiçbiri	Ağacın maksimum derinliğini ifade eder. Hiçbiri ise, tüm yapraklar saf olana veya tüm yapraklar minimum örnek sayısından daha azını içerene kadar düğümler genişletilir.
Minimum Örnek Sayısı (min_samples_split)	2	Bir düğümün bölünmeden önce sahip olması gereken minimum örnek sayısıdır.
Minimum Yaprak Örnek Sayısı(min_samples_leaf)	1	Bir yaprağın sahip olması gereken minimum örnek sayısıdır.
Minimum Ağırlıklı Örnek Sayısı(min_weight_fraction_leaf)	0.0	Minimum yaprak örnek sayısıdır. Ağırlıklı örneklerin, toplam örnekler içerisindeki oranıdır. Ağacın dengeli gitmesi için kullanılır.
Maksimum Öznitelik Sayısı(max_features)	sqrt	Maksimum öznitelik sayısını ifade eder.
Maksimum Yaprak Sayısı (max_leaf_nodes)	Hiçbiri	Maksimum yaprak sayısını ifade eder.
Minimum Düğüm Safsızlığı(min_impurity_decrease)	0.0	Düğüm safsızlığı parametresi ağaçların öz nitelikleri(verileri) nasıl böldüğünü gösterir.
Ön Yükleme(bootstrap)	Doğru	Ağaç oluştururken önyükleme örneklerinin kullanılıp kullanılmadığını ifade eder.
oob Skoru(oob_score)	Yanlış	Tahmin hatalarını hesaplamak için kullanılan bir parametredir.
İş Sayısı(n_jobs)	Hiçbiri	Paralel olarak çalıştırılacak iş sayısını ifade eder.
Rastgele Durum(random_state)	Hiçbiri	Hem ağaç oluştururken kullanılan örneklerin önyüklemesinin rasgeleliğini hem de her düğümde en iyi bölünmeyi ararken dikkate alınacak özelliklerin örneklemesini kontrol eder.
Ayrıntı Düzeyi(verbose)	0	Eğitim ve tahmin etme sırasında ayrıntı düzeyini kontrol eder.
Yeni Orman Parametresi (warm_start)	Yanlış	Eski çözümleri unutup yepyeni bir ormana sığdırması için ayarlanır.
Sınıf Ağırlıkları (class_weight)	Hiçbiri	Formdaki sınıflarla ilişkili ağırlıkları belirleyemeye yarayan parametredir.
Karmaşıklık (ccp_alpha)	0.0	Minimum Maliyet-Karmaşıklık Budaması için kullanılan karmaşıklık parametresidir. Varsayılan olarak ayarlandığında, budama yapılmaz.
Maksimum Örneklem Sayısı(max_samples)	Hiçbiri	Herhangi bir ağaca orijinal veri kümesinin hangi bölümünün verileceğini belirler.

2.4.3. Gradyan Artırma Ağaçları Algoritması

GAA, 2001 senesinde Friedman tarafından regresyon ve sınıflama amacıyla önerilen bir toplu makine öğrenme yöntemidir (Friedman, 2001). RO ve GAA arasındaki fark, gradyan destekli ağaç modellerinin sırayla öğrenmesidir. GAA'da, bir dizi ağaç oluşturulur ve her ağaç serideki önceki ağacın hatalarını düzeltmeye çalışır. Ağaçlar, daha fazla iyileştirme elde edilemeyene kadar sırayla eklenir.

GAA algoritmaları başlangıç olarak makine öğrenme topluluğu tarafından sınıflama ihtiyacı için tanıtılmıştır (Freund ve Schapire, 1996). Temel düşünce, gelişmiş öngörü doğruluğuna sahip "güçlü bir öğrenen" bulmak için "zayıf öğrenenler" adı verilen birkaç basit modeli tekrarlı olarak birleştirmektir. Friedman, GAA algoritmasını kayıp fonksiyonları olgusu ile birleştirerek artırmaya ilişkin istatistiksel bir yön vermiştir (Friedman, 2000). GAA, kayıp işlevini en aza indiren ek bir yöntem bulmayı amaçlayan nümerik bir en iyileme algoritması olarak görülebilir. Bu şekilde, GAA algoritması tekrarlı olarak her adımda kayıp işlevini en iyi düşüren yeni bir nihai ağacı yani "zayıf öğrenenler" ekler. Daha net olarak, regresyonda algoritmik modeller bir öngörme ile başlar,bu genellikle kayıp işlevini en üst seviyede azaltan bir nihai ağacıdır.

GAA algoritmasının parametreleri, varsayılan değerleri ve parametrelerin tanımları Çizelge 2.4'de yer almaktadır.

Çizelge 2.4. Gradyan artırma ağaçları algoritması parametreleri

Gradyan Artırma Ağaçları		
Parametreler	**Varsayılan Değerler**	**Tanım**
Tahmin Edici Nesne (init)	Hiçbiri	İlk tahminleri hesaplamak için kullanılan bir nesnedir.
Kayıp(loss)	deviance	En iyilenecek kayıp fonksiyonudur. Varsayılan olasılıklı çıktılarla sınıflama için sapmayı ifade eder.
Öğrenme Oranı (learning_rate)	0.1	Öğrenme oranıyla her sınıflandırıcının katkısını azaltma oranıdır.
Tahmin Sayısı(n_estimators)	100	Ormandaki ağaç sayısıdır.
Örneklerin Oranı(subsample)	1.0	Bireysel temel öğrenicileri tahmin etmek için kullanılacak örneklerin oranıdır.

Çizelge 2.4. Gradyan artırma ağaçları algoritması parametreleri (devam)

Gradyan Artırma Ağaçları		
Parametreler	**Varsayılan Değerler**	**Tanım**
Bölünme Kriteri(criterion)	friedman_mse	Bölünmenin kalitesini ölçme fonksiyonunu belirler.
Minimum Örnek Sayısı(min_samples_split)	2	Bir düğümün bölünmeden önce sahip olması gereken minimum örnek sayısıdır.
Minimum Yaprak Örnek Sayısı(min_samples_leaf)	1	Bir yaprağın sahip olması gereken minimum örnek sayısıdır.
Yaprak Düğüm Minimum Örnek Sayısı (min_weight_fraction_leaf)	0.0	Bir yaprak düğümde olması gereken tüm giriş örneklerinin toplam ağırlıkların minimum ağırlıklı bölümüdür.
Maksimum Derinlik(max_depth)	Hiçbiri	Ağacın maksimum derinliğini ifade eder. Hiçbiri ise, tüm yapraklar saf olana veya tüm yapraklar minimum örnek sayısından daha azını içerene kadar düğümler genişletilir.
Minimum Düğüm Safsızlığı (min_impurity_decrease)	0.0	Düğüm safsızlığı parametresi ağaçların öz nitelikleri(verileri) nasıl böldüğünü gösterir. İlk olarak ana düğümün düğüm safsızlığını (node impurity) hesaplanır, devamında bölme için belirli bir özellik kullanılacaksa alt düğümlerin düğüm safsızlıkları hesaplanır.
Rastgele Durum(random_state)	Hiçbiri	Hem ağaç oluştururken kullanılan örneklerin önyüklemesinin rasgeleliğini hem de her düğümde en iyi bölünmeyi ararken dikkate alınacak özelliklerin örneklemesini kontrol eder.
Maksimum Öznitelik Sayısı(max_features)	Hiçbiri	Maksimum öznitelik sayısını ifade eder.
Ayrıntı Düzeyi(verbose)	0	Eğitim ve tahmin etme sırasında ayrıntı düzeyini kontrol eder.
Maksimum Yaprak Sayısı(max_leaf_nodes)	Hiçbiri	En iyi şekilde bir ağacın büyümesidir. Varsayılan olarak ayarlandığında sınırsız sayıda yaprak ve düğüm anlamına gelir.
Yeni Orman Parametresi(warm_start)	False	Eski çözümleri unutup yepyeni bir ormana sığdırması için ayarlanır.
Doğrulama Seti(validation_fraction)	0.1	Erken durdurma için doğrulama seti olarak ayırılacak eğitim verilerinin oranıdır.
Erken Durdurma Kriteri(n_iter_no_change)	Hiçbiri	Doğrulama puanı iyileşmediğinde eğitimi sonlandırmak için erken durdurmanın kullanılıp kullanılmayacağına karar vermek için kullanılır. Erken durdurmayı devre dışı bırakmak için varsayılan olarak ayarlanmıştır.
Durdurma Kriteri Toleransı(tol)	0.0001	Durdurma kriterleri için toleransı ifade eder.
Karmaşıklık(ccp_alpha)	0.0	Minimum maliyet-karmaşıklık budaması için kullanılan karmaşıklık parametresidir. Varsayılan olaraka ayarlandığında, budama yapılmaz.

2.4.4. Aşırı Gradyan Artırma Algoritması

AGA, 2014 yılında Chen tarafından tanıtılan ağaç tabanlı bir yöntemdir (Chen ve Guestrin, 2016). Gradyan destekli ağaçların ölçeklenebilir ve doğru bir uygulamasıdır, özellikle performans hesaplama hızını ve modelini en iyileme etmek için tasarlanmıştır. GA ile karşılaştırıldığında, AGA fazla uydurma etkisini azaltmak için bir düzenleme terimi kullanır, daha iyi bir öngörü ve çok daha hızlı hesaplama çalıştırma süreleri sağlar (Ajit, 2016). AGA algoritmasının çeşitli parametreleri ve varsayılan değerleri vardır. AGA algoritmasının parametreleri, varsayılan değerleri ve parametrelerin tanımları Çizelge 2.5'da yer almaktadır.

Çizelge 2.5. Aşırı gradyan artırma ağaçları algoritması parametreleri (Wade, 2020)

Aşırı Gradyan Artırma Algoritması		
Parametreler	**Varsayılan Değerler**	**Tanım**
Öğrenme Oranı (learning_rate)	0.1	Öğrenme oranıyla her sınıflandırıcının katkısını azaltma oranıdır.
Örneklerin Oranı(subsample)	1.0	Bireysel temel öğrenicileri tahmin etmek için kullanılacak örneklerin oranıdır.
Tahmin Sayısı(n_estimators)	100	Ormandaki ağaç sayısıdır.
Örneklem Kullanılma Durumu(bootstrap)	Yanlış	Ağaç inşa ederken örneklem kullanılıp kullanılmadığıdır. Varsayılan ise, tüm veri kümesi her ağacı oluşturmak için kullanılır.
Karmaşıklık(ccp_alpha)	0.0	Minimum maliyet-karmaşıklık budaması için kullanılan karmaşıklık parametresidir. Varsayılan olarak ayarlandığında, budama yapılmaz.
Sınıf Ağırlıkları(class_weight)	Hiçbiri	Ağırlıkları giriş verilerindeki sınıf frekanslarıyla ters orantılı otomatik olarak ayarlamak için y değerlerini kullanır.
Bölünme Kriteri(criterion)	Gini	Bir bölünmenin kalitesini ölçme fonksiyonudur.
Maksimum Derinlik(max_depth)	Hiçbiri	Ağacın maksimum derinliğini ifade eder. Hiçbiri ise, tüm yapraklar saf olana veya tüm yapraklar minimum örnek sayısından daha azını içerene kadar düğümler genişletilir.
Maksimum Öznitelik Sayısı(max_features)	Hiçbiri	Maksimum öznitelik sayısını ifade eder.
Maksimum Yaprak Sayısı(max_leaf_nodes)	Hiçbiri	En iyi şekilde bir ağacın büyümesidir. Varsayılan olarak ayarlandığında sınırsız sayıda yaprak ve düğüm anlamına gelir.
Maksimum Örneklem Sayısı(max_samples)	Hiçbiri	Herhangi bir ağaca orijinal veri kümesinin hangi bölümünün verileceğini belirler.

Çizelge 2.5. Aşırı gradyan artırma ağaçları algoritması parametreleri (devam)

Aşırı Gradyan Artırma Algoritması		
Parametreler	**Varsayılan Değerler**	**Tanım**
Minimum Düğüm Safsızlığı (min_impurity_decrease)	0.0	Düğüm safsızlığı parametresi ağaçların öz nitelikleri(verileri) nasıl böldüğünü gösterir. İlk olarak ana düğümün düğüm safsızlığını (node impurity) hesaplanır, devamında bölme için belirli bir özellik kullanılacaksa alt düğümlerin düğüm safsızlıkları hesaplanır.
Rastgele Durum(random_state)	Hiçbiri	İkili koordinat alçalma için verileri karıştırmak için sözde rastgele sayı üretimini kontrol eder.
Ayrıntı Düzeyi(verbose)	Yanlış	İlerleme mesajlarının standart çıkışa yazdırılıp yazdırılmayacağını ifade eder.
Yeni Orman Parametresi(warm_start)	Yanlış	True olarak ayarlandığında, başlatma olarak sığdırmak için önceki çağrının çözümünü yeniden kullanın, aksi takdirde önceki çözümü silmeniz yeterlidir. Sözlüğe bakın.
İş Sayısı(n_jobs)	Hiçbiri	Paralel olarak çalıştırılacak iş sayısını ifade eder.
oob Skoru(oob_score)	Yanlış	Tahmin hatalarını hesaplamak için kullanılan bir parametredir.
Minimum Bölünme Safsızlığı(min_impurity_split)	Hiçbiri	Ağaç büyümesinde erken durma eşiğini ifade eder. Bir düğüm saflığı eşiğin üzerindeyse bölünür, aksi takdirde bir yapraktır.
Minimum Örneklem Yaprak Sayısı(min_samples_leaf)	1	Bir yaprak düğümünde olması gereken tüm giriş örneklerinin toplam ağırlıklarının minimum ağırlıklı bölümüdür.
Minimum Örneklem Bölünme Sayısı(min_samples_split)	2	İç düğümü bölmek için gereken en az örnek sayısıdır.

2.4.5. Lojistik Regresyon Algoritması

LR, orijinal olarak 1958'de Cox tarafından önerildiği gibi doğrusal diskriminantları içeren geleneksel bir sınıflandırma algoritmasıdır (Cox,1958). Birincil çıktı, verilen girdi noktasının belirli bir sınıfa ait olma olasılığıdır. Olasılığın girdi alanını iki bölgeye ayıran doğrusal bir sınır değerine dayanarak model oluşturur. LR'nin uygulanması kolaydır, bu da onu en yaygın kullanılan sınıflandırıcılardan biri yapar (Raschka, 2015).

LR, bir kategori olasılığının bir dizi açıklayıcı değişkenle ilişkili olduğu istatistiksel bir modelleme tekniğidir. Lojistik model aşağıdaki denklem 2.9 ve denklem 2.10'da görüldüğü gibi tanımlanır (Dong vd., 2016).

$$z = a_0 + \sum_{i=1}^{n} a_i x_i \tag{2.9}$$

$$P(z) = \frac{e^z}{1 + e^z} \tag{2.10}$$

Burada kullanılan z sembolü katsayılarla ağırlıklandırılmış bağımsız değişkenlerden oluşan bir regresyon denkleminin çıktısına eşittir. x_i ifadesi ise *i.* bağımsız değişkeni ifade eder. a_i ise *i.* bağımsız değişkenin regresyon katsayısını ifade etmektedir. P(z) olasılığı bir girdi noktasının bir sınıfa ait olma olasılığıdır. LR modelinin parametreleri, analitik olarak elde edilemediğinden, iteratif bir yöntem olan maksimum olabilirlik tekniği ile tahmin edilmektedir (Albayrak, 2009). Tahmin edilen mantıksal bağımlı değişkenin değerleri ile gözlenen değerler arasındaki olabilirliğin maksimum yapılması amaçlanmaktadır (Özdemir vd., 2011).

LR yönteminde bağımsız değişkenlerin kategorik değişkenlerle açıklanırken 0 ya da 1 olma olasılığı hesaplanmaktadır. Bağımlı değişken öncelikle mantıksal değişkene (logaritmalar alınarak) dönüştürülür. Böylece şıklardan herhangi bir tanesinin olma oranının tahmini yapılmaktadır. Üstünlük Oranı değerlerinin doğal logaritması alındığında aşağıdaki denklem 2.11'e ulaşılmaktadır (Sharma ve Arikawa, 1996):

$$\ln \frac{P(z)}{1 - P(z)} = a_0 + a_1 x_1 + a_2 x_2 + \cdots + a_n x_n \tag{2.11}$$

Denklem 2.11, olasılık değeri olarak denklem 2.12'deki gibi ifade edilebilir.

$$P(z) = \frac{1}{1 + e^{-(a_0 + a_1 x_1 + a_2 x_2 + \cdots + a_n x_n)}} \tag{2.12}$$

LR algoritmasının çeşitli parametreleri ve varsayılan değerleri vardır. LR'da en önemli iki parametre çözücü fonksiyon ve rastgele durumdur. Hiper parametrelere kullanıcılar istediği değeri verebilir (Ilhan ve Güdar, 2021).

LR algoritmasının parametreleri, varsayılan değerleri ve parametrelerin tanımları Çizelge 2.6'de yer almaktadır.

Çizelge 2.6. Lojistik regresyon algoritması parametreleri

Lojistik Regresyon		
Parametreler	**Varsayılan Değerler**	**Tanım**
Ceza (penalty)	l2	Kullanılacak ceza normunu tanımlar. None seçilmesi durumunda hiç bir ceza normunu uygulamaz.
İkili Formülasyon(dual)	Yanlış	İkili veya birincil formülasyonu ifade eder. İkili formülasyon sadece liblinear çözücü ile l2 cezası için uygulanmaktadır.
Durdurma Kriteri Toleransı(tol)	0.0001	Durdurma kriterleri için toleransı ifade eder.
Düzenlilik Parametresi(C)	1.0	Düzenlilik parametresidir.
Ceza (penalty)	l2	Kullanılacak ceza normunu tanımlar. None seçilmesi durumunda hiç bir ceza normunu uygulamaz.
İkili Formülasyon(dual)	Yanlış	İkili veya birincil formülasyonu ifade eder. İkili formülasyon sadece liblinear çözücü ile l2 cezası için uygulanmaktadır.
Durdurma Kriteri Toleransı(tol)	0.0001	Durdurma kriterleri için toleransı ifade eder.
Düzenlilik Parametresi(C)	1.0	Düzenlilik parametresidir.
Eğitim Karar Fonksiyonu (fit_intercept)	Doğru	Bu parametre, karar fonksiyonuna bir sabitin (önyargı veya kesme) eklenmesi gerektiğini belirtir.
Durdurma Ölçütü (intercept_scaling)	1	Yalnızca çözücü fonksiyonu liblinear olduğunda ve doğru olarak ayarlandığında kullanışlıdır.
Sınıf Ağırlıkları (class_weight)	Hiçbiri	Formdaki sınıflarla ilişkili ağırlıkları belirleyemeye yarayan parametredir.
Rastgele Durum (random_state)	Hiçbiri	Hem ağaç oluştururken kullanılan örneklerin önyüklemesinin rastgeleliğini hem de her düğümde en iyi bölünmeyi ararken dikkate alınacak özelliklerin örneklemesini kontrol eder.
Çözücü(solver)	lbfgs	Optimizasyon probleminde kullanılacak algoritma fonksiyonudur.
Maksimum İterasyon Sayısı(max_iter)	100	Maksimum iterasyon sayısını ifade eder.
Çoklu Sınıf (multi_class)	Oto	Birden çok grup içerisinde sınıflandırmayı sağlayan parametredir.
Ayrıntı Düzeyi (verbose)	0	Eğitim ve tahmin etme sırasında ayrıntı düzeyini kontrol eder. Varsayılan ayrıntılı çıktının etkinleştirilmemesidir.
Yeni Orman Parametresi (warm_start)	Yanlış	Eski çözümleri unutup yepyeni bir ormana sığdırması için kullanılır.

Çizelge 2.6. Lojistik regresyon algoritması parametreleri(devam)

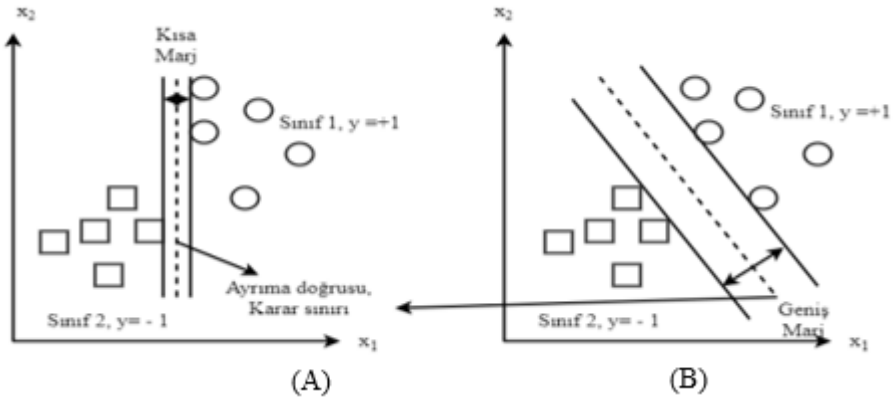
Lojistik Regresyon		
Parametreler	**Varsayılan Değerler**	**Tanım**
İş Sayısı (n_jobs)	Hiçbiri	Paralel olarak çalıştırılacak iş sayısını ifade eder.
Karıştırma Parametresi (l1_ratio)	Hiçbiri	Bir ceza fonksiyonu karıştırma parametresidir. Yalnızca ceza fonksiyonu elasticnet olduğunda kullanılır.

2.4.6. Destek Vektör Makinesi Algoritması

DVM ilk olarak 1995 yılında Vapnik ve Cortes tarafından önerilmiştir (Cortes ve Vapnik, 1995). DVM iki boyutlu uzayda doğrusal, üç boyutlu uzayda düzlemsel ve çok boyutlu uzayda hiper düzlem şeklindeki ayırma mekanizmaları ile veriyi iki ya da daha çok sınıfa ayırma yeteneğine sahiptir. Veri grubunun bir doğru ile ayrılabildiği durum, grubun doğrusal olarak ayrılabildiği durumdur. Burada Vapnik tarafından ileri sürülen bir fikir, iki sınıfı ayıran nesnenin bir doğru yerine bir koridor olması ve bu koridorun genişliğinin bazı veri vektörleri tarafından belirlenerek mümkün olan en büyük genişlikte olmasıdır (Cortes ve Vapnik, 1995). DVM yaygın olarak iki olası kategoriden bir makine öğrenimi ile yeni veri örnekleri atamak için ayırt edici bir sınıflandırıcı olarak kullanılır. *n* boyutlu verileri iki sınıfa ayıran bir hiper düzlem tanımlanır, hiper düzlem burada destek vektörleri olarak adlandırılan en yakın veri noktalarına olan geometrik mesafeyi maksimize eder (Raschka, 2015). DVM algoritmasının amacı, *n* boyutlu bir uzayda (*n* öznitelik sayısı) veri noktalarını belirgin bir şekilde sınıflandıran bir hiper düzlem bulmaktır. İki veri noktası sınıfını ayırmak için seçilebilecek birçok olası hiper düzlem vardır. Hiper düzlem tarafından tanımlanan iki sınıf arasındaki mesafe miktarı marj olarak ifade edilmektedir (Doğan vd., 2021). Amaç maksimum marjı olan, yani her iki sınıfın veri noktaları arasında maksimum mesafeye sahip bir düzlem bulmaktır (Kumaş, 2021).

Şekil 2.2'de görüleceği üzere tanımlanan veri kümesini doğrusal olarak ayıran birçok hiper düzlem vardır. Bu kanonik hiper düzlemler doğrusal ayrılabilir formda olan veriyi, aynı sınıfa ait veri noktalarını hiper düzlemin tamamen aynı tarafında bırakacak şekilde ayırmaktadır. Amaç en iyi genelleme yeteneğine sahip tek bir optimal ayırıcı hiper düzlemi bulmaktır.

Şekil 2.2 (A)'da veriyi bütünüyle ayırabilen nispeten dar marja dolayısıyla daha yüksek beklenen riske sahip bir hiper düzlem; Şekil 2.2 (B)'de ise daha geniş ve daha kabul edilebilir bir hiper düzlem olduğu görülmektedir (Kecman, 2001).



Şekil 2.2. Ayrıcı hiper düzlemler **A)** Minimum marja sahip olan ayırıcı hiper düzlem **B)** Maksimum marja sahip olan ayırıcı hiper düzlem (Kecman, 2001)

DVM, doğrusal sınıflandırma gerçekleştirmeye ek olarak, doğrusal olmayan sınıflandırmayı verimli bir şekilde gerçekleştirmek için bir çekirdek yöntemi fikrini de sunar. Öz nitelikleri, verilerin ayrılabilir olduğu yeni bir öz nitelik alanına aktaran bir özellik eşleme metodolojisidir (Muller vd., 2001).

Şekil 2.3 (A)'da gösterildiği üzere iki sınıflı verileri birbirinden ayırabilen birçok hiper-düzlem çizilebilir. Ancak DVM'nin amacı kendisine en yakın noktalar arasındaki uzaklığı maksimuma çıkaran hiper düzlemi bulabilmektir. Şekil 2.3 (B)'de görüldüğü üzere sınırı maksimuma çıkararak en uygun ayrımı yapan hiper düzleme optimum hiper düzlem ve sınır genişliğini sınırlandıran noktalar ise destek vektörleri olarak adlandırılır. Doğrusal olarak ayrılabilen iki sınıflı bir sınıflandırma probleminde DVM' nin eğitimi için *k* sayıda örnekten oluşan eğitim verisinin $\{ x_i, y_i \}$, $i=1,.....,k$ olduğu kabul edilirse, optimum hiper düzleme ait eşitsizlik denklemleri denklem 2.13 ve denklem 2.14'deki gibi olur.

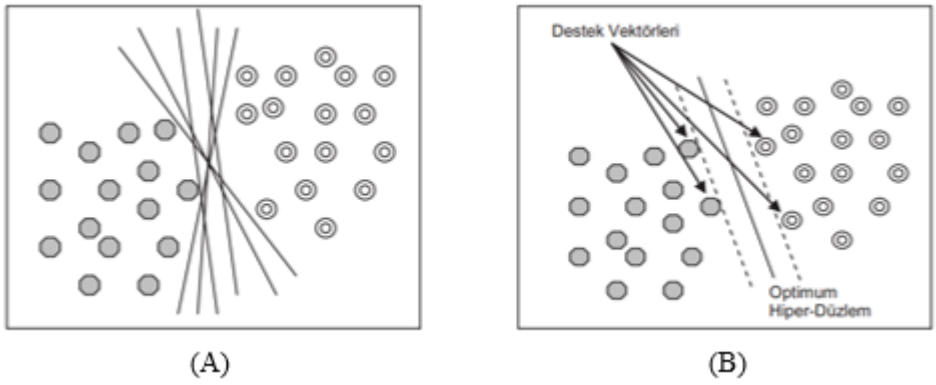
$$w.x_i + b \geq +1 \; her \; y = +1 \; için \tag{2.13}$$

$$w.x_i + b \leq +1 \; her \; y = -1 \; için \tag{2.14}$$

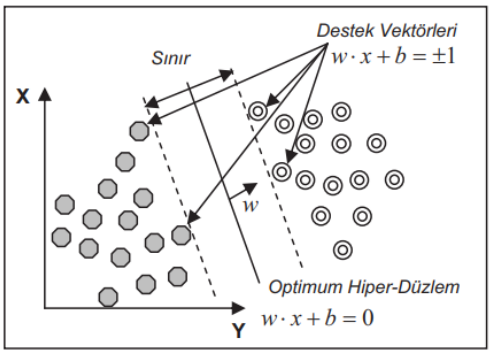
Burada $x \in R^N$ olup N-boyutlu bir uzayı, $y \in \{-1, +1\}$ ise sınıf etiketlerini, w ağırlık vektörünü (hiper-düzlemin normali) ve b eşik değerini göstermektedir (Osuna vd., 1997). Optimum hiper-düzlemin belirlenebilmesi için bu düzleme paralel ve sınırlarını oluşturacak iki hiper düzlemin belirlenmesi gerekir. Bu hiper düzlemleri oluşturan noktalar destek vektörleri olarak adlandırılır ve bu düzlemler denklem 2.15'de görüldüğü gibi ifade edilirler (Kavzoğlu ve Çölkesen, 2010).

Şekil 2.4'de doğrusal ayrılabilen veri setlerinde optimum hiper düzlem görülmektedir.

$$w.x_i + b = \pm 1 \qquad (2.15)$$



Şekil 2.3. Hiper düzlemler **A)** İki sınıflı bir problem için hiper düzlemler **B)**Optimum hiper düzlem ve destek vektörleri(Kavzoğlu ve Çölkesen, 2010).



Şekil 2.4. Doğrusal olarak ayrılabilen veri setleri için hiper düzlemin belirlenmesi (Kavzoğlu ve Çölkesen, 2010).

Optimum hiper düzlemin sınırının maksimuma çıkarılması için $\|w\|$ ifadesinin minimum hale getirilmesi gerekir. Bu durumda en uygun hiper düzlemin belirlenmesi denklem 2.16 sınırlı optimizasyon probleminin çözümünü gerektirir.

$$min\left[\frac{1}{2}\|w\|^2\right] \tag{2.16}$$

Buna bağlı sınırlamalar ise;

$$y_i(w.x_i + b) - 1 \geq 0 \; ve \; y_i \in \{1, -1\} \tag{2.17}$$

şeklinde denklem 2.17'de ifade edilir (Cortes ve Vapnik, 1995). Bu optimizasyon problemi lagrange denklemleri kullanılarak çözülebilir. Bu işlem sonrasında;

$$L(w, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{k} \alpha_i y_i(w.x_i + b) + \sum_{i=1}^{k} \alpha_i \tag{2.18}$$

denklem 2.18 eşitliği elde edilir. Sonuç olarak, doğrusal olarak ayrılabilen iki sınıflı bir problem için karar fonksiyonu denklem 2.19'daki şekilde yazılabilir (Osuna vd., 1997).

$$f(x) = sign\left(\sum_{i=1}^{k} \lambda_i y_i(x.xi) + b\right) \tag{2.19}$$

DVM algoritmasının çeşitli parametreleri ve varsayılan değerleri vardır. DVM algoritmasının parametreleri, varsayılan değerleri ve parametrelerin tanımları Çizelge 2.7'de yer almaktadır.

Çizelge 2.7. Destek vektör makineleri algoritması parametreleri

Destek Vektör Makineleri		
Parametreler	**Varsayılan Değerler**	**Tanım**
Düzenlilik Parametresi(C)	1.0	Düzenlilik parametresidir.
Çekirdek Fonksiyonu(Kernel)	rbf	Algoritmada kullanılacak çekirdek türünü belirtir.
Çekirdek Fonksiyonun Derecesi(degree)	3	Polinom çekirdek fonksiyonunun derecesidir.
Çekirdek Fonksiyon Katsayısı(gamma)	Ölçek	"rbf", "poli" ve "sigmoid" için çekirdek katsayısıdır.

Çizelge 2.7. Destek vektör makineleri algoritması parametreleri (devam)

Destek Vektör Makineleri		
Parametreler	**Varsayılan Değerler**	**Tanım**
Bağımsız terim(coef0)	0.0	Çekirdek fonksiyonundaki bağımsız terimdir. Yalnızca 'poli' ve 'sigmoid' çekirdek fonksiyonunda anlamlıdır.
Küçülen Buluşsal Yöntem Kullanımı(shrinking)	Doğru	Küçülen buluşsal yöntemin kullanılıp kullanılmayacağını ifade eder.
Olasılık(probability)	Yanlış	Olasılık tahminlerinin etkinleştirilip etkinleştirilmeyeceğini ifade eder.
Durdurma Kriteri Toleransı(tol)	0.0001	Optimizasyon için tolerans değerini ifade eder.
Ön Bellek Boyutu(cache_size)	200	Çekirdek önbelleğinin boyutunu ifade eder.
Sınıf Ağırlığı(class_weight)	Hiçbiri	Sınıf ağırlığını ifade eder. Varsayılan olduğunda tüm sınıfların bir ağırlığı eşittir.
Ayrıntı Düzeyi(verbose)	Yanlış	İlerleme mesajlarının standart çıkışa yazdırılıp yazdırılmayacağını ifade eder.
Maksimum İterasyon(max_iter)	200	Maksimum yineleme sayısını ifade eder.
Karar Ağacı Şekli(decision_function_shape)	Ovr	Bu parametre karar fonksiyonunun şekli ile ilgilidir.
Bağ Durumu (break_ties)	Yanlış	Bu parametre bağların koparılıp koparılmayacağını ifade eder.
Rastgele Durum(random_state)	Hiçbiri	İkili koordinat alçalma için verileri karıştırmak için sözde rastgele sayı üretimini kontrol eder.

2.4.7. Yapay Sinir Ağları Algoritması

YSA insanın sinir sisteminden ilham alınarak çalışma prensibine benzetilmesiyle oluşturulmuştur. İlk YSA modeli 1943'de, McCulloch ve Pitts tarafından önerilmiştir. McCulloch ve Pitts, insan beyninin hesaplama yeteneğinden esinlenerek, basit bir sinir ağı modeli geliştirmiştir (Öztemel, 2003). Yapay sinir ağlarının bilinen ağ yapısı girdi katmanı, gizli katman ve çıktı katmanı olmak üzere üç kısımdan meydana gelmektedir. Girdi katmanı, tahmin özniteliklerinin değerlerini alır çıktı katmanı ise tahmin edilen

sonucu elde etmeye yarar. Gizli ve çıktı katmanlarında her bir nöron, nörona verilen bütün verilerin ağırlıklandırılmış toplam değerini alır. Sonrasında, çıktı sonucunun hesaplanmasının mümkün olması ancak aktivasyon fonksiyonunun uygulanmasıyla sağlanır (Silva vd., 2017).

Çok katmanlı algılayıcı olarak da bilinen sinir ağları, insan sinir sisteminin operasyonlarını simüle etmek için tasarlanmıştır. Bir sinir ağının en basit biçimi, tek bir algılayıcıdır. Bir algılayıcı için temel öğeler, girdi değerleri, ilişkili ağırlıklar, tahmin, aktivasyon fonksiyonları ve hesaplanmış bir çıktıdır (Kuyucu, 2012).

Bir sinir ağı, karmaşık problemlerin üstesinden gelmek için giriş ve çıkış arasında birden fazla katman içerebilir. Sinir ağlarının bu karmaşık yapısı, onu, yeterli gizli katman verildiğinde, herhangi bir normal fonksiyonu istenen herhangi bir doğruluk düzeyine göre modelleyebilen evrensel bir yaklaşım aracı yapar. Model, yaygın olarak derin öğrenme ile derinleşecek şekilde genişletilebilir (Öztemel, 2003).

Donanımın hızlı gelişimi ve geri yayılım tekniklerinin sürekli araştırılması nedeniyle, sinir ağları şu anda makine öğreniminde en çok araştırılan konudur (Murphy, 2012). YSA'nın en önemli özelliği, deneyimlerden (tecrübe) yararlanarak öğrenebilmesidir. YSA, öğrenmenin yanı sıra bilgiler arasında ilişkiler oluşturma yeteneğine de sahiptir. Ayrıca veri birleştirme, kavramsallaştırma ve filtreleme için de kullanılabilir. YSA'nın endüstriyel uygulamalar, finans uygulamaları, askeri ve savunma uygulamaları, tıp ve sağlık uygulamaları, mühendislik uygulamaları, robot bilim, görüntü işleme, örüntü tanıma dışında iletişim sanayi, eğlence amaçlı tahmin gibi özel uygulama alanları da bulunmaktadır (Uğur ve Kınacı, 2006).

Genelde verilen bir girdi setine karşılık çıktı değerleri verilerek belirtilen öğrenme kuralına göre ağırlık değerleri otomatik olarak değiştirilmektedir. Eğitim verisinin tamamlanmasından sonra eğitilmiş olan ağ, ağırlık değerlerinin son durumuna göre, verilen herhangi bir veri setinin sonucunu tahminleyebilmektedir. Günümüzde belirli amaçlarla ve değişik alanlarda kullanılmaya uygun birçok yapay sinir ağı modeli (Perceptron, Adaline, MLP, LVQ, Hopfield, Recurrent, SOM, ART ve PCA gibi)

geliştirilmiştir. Öğrenme çeşitlerinden gözetimli öğrenme, gözetimsiz öğrenme, takviyeli öğrenme ve karma stratejiler kullanılmaktadır (Uğur ve Kınacı, 2006).

En önemli noktalardan bir tanesi gizli katmanda kaç nöron olacağına karar vermektir. Gizli katman nöron sayısı, öğrenme sırasında bellekte fazla bilgi barındırmak haricinde öğrenme işleminin daha iyi yapılmasını da sağlamaktadır. Gizili katman nöron sayısı bulunması için çeşitli yöntemler öne sürülmüştür. Bunlar 2n+1, 2n,n, n/2 gibi girdi katmanı nöron sayısına oranlı gizli katman sayılarıdır. Çelebi ve Bayraktar, yaptıkları çalışmalarda gizli katmandaki nöron sayısının belirlenmesinde genel bir kuralın bulunmadığına dikkat çekmişlerdir. Gizli katman, girdi katmanın aldığı ağırlıklandırılmış veriyi probleme uygun bir fonksiyonla işleyerek bir sonraki katmana iletir. Bu katmanda gereğinden az nöron kullanılması girdi verilerine göre daha az hassas çıktı elde edilmesine sebep olur. Gereğinden çok sayıda nöron kullanılması durumunda aynı ağda yeni tip veri gruplarının işlenmesinde zorluklar ortaya çıkar ve aşırı öğrenme olur. Gizli katman sayısının denklem 2.20'de görülen formül ile bulunduğu çalışmalar da mevcuttur (Bayru, 2007).

$$N_s = \frac{(\frac{1}{2}(N_g + N_c) + \sqrt{N_d})}{N_b} \tag{2.20}$$

Denklem 2.20'de ifade edilen N_s gizli katmandaki nöron sayısını, N_g girdi katmanındaki nöron sayısını, $N_ç$ çıktı katmanındaki nöron sayısını, N_d gözlem sayısını, N_b katman sayısını göstermektedir.

Çalışmaların çoğunda gizli katman sayısının deneme yoluyla bulunduğu görülmektedir.

Bir başka parametre olan momentum katsayısı ağın öğrenmesi esnasında, yerel bir optimum noktaya takılıp kalmaması için ağırlık değişim değerinin belirli bir oranda bir sonraki değişime eklenmesini sağlar. Momentum katsayısının kullanılması, ani sıçramaları ortadan kaldırma eğilimi gösterecektir, ancak her zaman işe yaramayabilir ve hatta yakınsamaya zarar verebilir. Önceden de ifade edildiği gibi, momentum katsayısı bir önceki parametre değişiminin belli bir oranının her iterasyonda bir sonraki parametre değişimine eklenmesi ile gerçekleşir (Dalkılıç, 2020).

Öğrenme katsayısı ise yani öğrenme hızı faktörü, eğitim örüntüsü ile ağın çıktı örüntüsünü birbirine yaklaştırmak için ağırlıkların ayarlanmasında kullanılmaktadır. Kullanıcı tarafından seçilen öğrenme hızı çok düşük olduğu durumlarda ağırlık değişimi çok yavaş olur ve dolayısıyla ağ çok yavaş öğrenmektedir. Hızın büyük olması durumunda ağırlıklarda da büyük miktarda değişimler olması demektir, bu durumda öğrenme katsayısının ağın performansını azaltıcı etkide bulunduğu gözlenmektedir (Dalkılıç, 2020).

Hata toleransı için YSA'nın optimum sonuca ulaştıklarına ilişkin kesin bir bilgi yoktur. Bu nedenle YSA kullanan kişiler ağın performansını ölçerken e kadar bir hatayı kabul etmektedirler. Bu hataya hata toleransı adı verilmektedir. Herhangi bir hata toleransının altındaki noktada ağın öğrenmiş olduğu kabul edilir. Bu noktalara lokal çözümler denilmektedir. En iyi çözüm olmamalarına rağmen kabul edilebilir bir hata seviyesinin altında bir hataya sahip olduğundan kabul edilebilir çözümler olarak ele alınabilirler. YSA algoritmasının çeşitli parametreleri ve varsayılan değerleri vardır.

YSA algoritmasının parametreleri, varsayılan değerleri ve parametrelerin tanımları Çizelge 2.8'da yer almaktadır.

Çizelge 2.8. Yapay sinir ağları algoritması parametreler

Yapay Sinir Ağları		
Parametreler	**Varsayılan Değerler**	**Tanım**
Gizli Katman Boyutu (hidden_layer_sizes)	100	En az maliyet-karışıklık budaması için kullanılan karışıklık parametresidir. Varsayılan olarak seçildiğinde, budama yapılmaz.
Aktivasyon Fonksiyonu (activation)	Relu	Gizli katman için aktivasyon fonksiyonunu ifade eder.
Çözücü(solver)	Adam	Ağırlık optimizasyonu için çözücü fonksiyonudur.
Düzenleme(alpha)	0.0001	L2 düzenleme teriminin gücüdür. L2 düzenlileştirme terimi, kayba eklendiğinde numune boyutuna bölünür.
Grup Boyutu(batch_size)	Auto	Stokastik optimize ediciler için mini partilerin boyutudur.
Öğrenme Oranı(learning_rate)	Constant	Öğrenme oranı, bir ağın parametrelerini ne kadar hızlı güncellediğini tanımlar.

Çizelge 2.8. Yapay sinir ağları algoritması parametreler(devam)

Yapay Sinir Ağları		
Parametreler	**Varsayılan Değerler**	**Tanım**
Başlangıç Öğrenme Oranı(learning_rate_init)	0.0001	Kullanılan ilk öğrenme oranıdır. Ağırlıkların güncellenmesinde adım boyutunu kontrol eder. Yalnızca çözücü='sgd' veya 'adam' olduğunda kullanılır.
Öğrenme Oranı Üs Değeri(power_t)	0.5	Ters ölçekleme öğrenme oranı için üssü ifade eder. Yalnızca çözücü='sgd' olduğunda kullanılır.
Maksimum İterasyon(max_iter)	200	Maksimum yineleme sayısıdır.
Karıştırıcı(shuffle)	Doğru	Her yinelemede örneklerin karıştırılıp karıştırılmayacağı. Yalnızca çözücü='sgd' veya 'adam' olduğunda kullanılır.
Rastgele Durum(random_state)	Hiçbiri	Ağırlıklar ve önyargı başlatma için rasgele sayı üretimini, erken durdurma kullanılıyorsa tren testi bölünmesini ve çözücü='sgd' veya 'adam' olduğunda toplu örneklemeyi belirler.
Durdurma Kriteri Toleransı(tol)	0.0001	Optimizasyon için toleransı ifade eder.
Ayrıntı Düzeyi(verbose)	Yanlış	İlerleme mesajlarının çıkışa yazdırılıp yazdırılmayacağını ifade eder.
Yeni Orman Parametresi(warm_start)	Yanlış	Doğru olarak ayarlandığında, başlatma olarak sığdırmak için önceki çağrının çözümünü yeniden kullanın, aksi takdirde önceki çözümü silmeniz yeterlidir.
İtme(Momentum)	0.9	Momentum, önceki adımların bilgisi ile bir sonraki adımın yönünü bilmeye yardımcı olur. Salınımların önlenmesine yardımcı olur.
Nesterovs İtme(nesterovs_momentum)	Doğru	Nesterov'un momentumunun kullanılıp kullanılmayacağını ifade eder. Yalnızca çözücü='sgd' ve momentum > 0 olduğunda kullanılır.
Erken Durdurma (early_stopping)	Yanlış	Doğrulama puanı iyileşmediğinde eğitimi sonlandırmak için erken durdurmanın kullanılıp kullanılmayacağını ifade eder.
beta_1	0.9	Adam'daki ilk moment vektörünün tahminleri için üstel bozulma oranıdır.[0, 1) olmalıdır. Yalnızca çözücü='adam' olduğunda kullanılır.
beta_2	0.999	Adam'daki ikinci moment vektörü tahminleri için üstel bozulma oranıdır. [0, 1) olmalıdır. Yalnızca çözücü='adam' olduğunda kullanılır.
Epsilon	1,00E-08	Adam cinsinden sayısal kararlılık değeridir. Yalnızca çözücü='adam' olduğunda kullanılır.
Erken Durdurma Kriteri (n_iter_no_change)	10	Erken durdurma kriteridir. Erken durdurmayı devre dışı bırakmak için varsayılan olarak ayarlanmıştır.
Erken Durdurma (early_stopping)	Yanlış	Doğrulama puanı iyileşmediğinde eğitimi sonlandırmak için erken durdurmanın kullanılıp kullanılmayacağını ifade eder.

Çizelge 2.8. Yapay sinir ağları algoritması parametreler(devam)

Yapay Sinir Ağları		
Parametreler	**Varsayılan Değerler**	**Tanım**
Maksimum Kayıp Fonksiyonu (max_fun)	15000	Yalnızca çözücü='lbfgs' olduğunda kullanılır. Maksimum kayıp işlevi çağrısı sayısını ifade eder.
Doğrulama Seti (validation_fraction)	0.1	Erken durdurma için doğrulama seti olarak ayırılacak eğitim verilerinin oranıdır.

2.4.8. k-En Yakın Komşu Algoritması

Bu algoritma, sınıflandırma ve regresyon problemlerinde kullanılan parametrik olmayan bir algoritmadır. Bu algoritmanın sınıflandırma problemleri için öngörüsü, eğitim verilerindeki yeni örneğe en yakın olan k veri noktalarını belirlemek ve bu yeni örneği baskın değer ile sınıflandırmaktır. Verilerin birbirine olan mesafesini ölçmek amacıyla Öklid uzaklığı, Manhattan uzaklığı ve Minkowski uzaklığı gibi yöntemler kullanılır. Regresyon problemleri için öngörüsü, k komşularının ortalama değerini bulup yeni örnek değerini hesaplamaktır. k-EYK az öznitelikle iyi çalışabilir, fakat öznitelik boyutları arttığında sorun yaşar (Murphy 2012). Bu algoritmanın avantajlarının bir kaçı; basit ve veriminin yüksek olması, veri dağılımı ile ilgili çıkarım yapmaması ve verinin eğitiminin hızlı olmasıdır. Dezavantajları ise; sınıflandırmanın normalden fazla zaman alması, bilgisayarda fazlaca depolama alanı gereksinimi ve veri kümesindeki eksik değerlerin işlem eforunu artırmasıdır.

k-EYK algoritması, örnek bazlı algoritmaların kümesindedir. k-EYK eğitme sürecini veri kümesinde bulundurulan eğitim kümesi ile gerçekleştirir. Eğitim kümesi en yakın varsayılan k tane veriyi, belirlinen uzaklık ölçütü çerçevesinde benzeşen noktalarının hesaplanması ile yapmaktadır (Dudani, 1976). Bu uzaklık ölçütleri Minkowski, Öklid, Chebyshev ve kosinüs eşitlikleri kullanarak belirlenmektedir. Literatürde ise çoğunlukla Öklid mesafesinin tercih edildiği görülmektedir (Bhatia, 2010).

d1 ve *d2* iki noktalar kümesi olmak üzere; ($d1 = x_1, x_2, \dots, x_n \; ve \; d2 = \; y_1, y_2, \dots, y_n$)

d1 ve *d2* arasındaki mesafe denklem 2.21'de gösterildiği gibi hesaplanır (Bhatia, 2010):

$$d(d1, d2) = d(d2, d1) = \sqrt{\sum_{i=1}^{n} (d1_i - d2_i)^2} \quad (2.21)$$

Yeni bir veri sınıf modelleme amacıyla algoritmaya ulaştığında, öğrenmiş veri kümesi içerisinde bulunan k adet en yakın komşu sınıfının etiketlerine bakılır. Daha sonra sınıfların etiketlerinin çoğunluğuna göre yeni u veri o kümeye dahil edilir (Muja ve Lowe, 2009). k-EYK algoritmasında performans ölçümünün dışardan girilen komşu sayısına duyarlılığı ve belirlenen uzaklık ölçütüne karşı hassasiyeti en temel eksikliklerindendir (Liu ve Zhang, 2012).

Diğer bir şekilde mevcuttaki verilerin birbirleri ile olan uzaklığı kullanılarak sınıflandırma yapıldığı için değişkenlerin devamlı olması gerekse de kategorik değişkenlerin bulunduğu durumlarda mesafe hesaplanması için bazı metotlar tavsiye edilmektedir (Han vd., 2012). Fakat kategorik değişkenlerin çoğunlukta olduğu iş gören seçim modellemelerinde, bu metodun beklenen doğruluk ile sonuçlar veremeyebileceği de göz önünde bulundurulmalıdır.

k-EYK algoritmasının çeşitli parametreleri ve varsayılan değerleri vardır. k-EYK algoritmasının parametreleri, varsayılan değerleri ve parametrelerin tanımları Çizelge 2.9'da yer almaktadır.

Çizelge 2.9. k-EYK algoritması parametreleri

K-En Yakın Komşu		
Parametreler	**Varsayılan Değerler**	**Tanım**
Komşu Sayısı(k)	5	Bir sınıflandırma yapılırken kaç adet komşuya bakılacağını ifade eder.
Ağırlıklar(w)	üniform	Değişken ağırlıklarının nasıl dağılım gösterdiğini tanımlaya yarayan parametredir.
Algoritma(algorithm)	otomatik	En yakın komşuları hesaplamak için kullanılan algoritmayı ifade eder.
Yaprak Boyutu(leaf_size)	30	Ağaçta kullanılacak yaprak boyutunu ifade eder.

Çizelge 2.9. k-EYK algoritması parametreleri (devam)

K-En Yakın Komşu		
Parametreler	**Varsayılan Değerler**	**Tanım**
p	2	Uzaklık ölçüm metriği için güç parametresini ifade eder.
Uzaklık Ölçüm Metriği(distance)	Minkowski	Ağaç için kullanılacak uzaklık metriğidir.
Uzaklık Ölçüm Parametresi	Hiçbiri	Metrik fonksiyonu için ek bağımsız değişkenleri ifade eder.
İş Sayısı(n_job)	Hiçbiri	Komşu araması için çalıştırılacak paralel iş sayısını ifade eder.

2.5. Kaynak Araştırması

1990 ve sonrasında günümüzdeki oyunlar, görüntü ve sinyal işleyiciler, robotik kodlama gibi birçok alanda makine öğrenmesi ve yapay zekâ kullanılmaktadır (Guner vd., 2017). Literatürde bu alan ile ilgili birçok çalışma bulunmaktadır.

Chin Yuan ve ark. (2012) Tayvan'daki teknoloji işletmelerinin hızla yetenekli çalışanları kaybetmeleri ile birlikte oluşan çalışan devir oranı tahmin etme ihtiyacını karşılamak amacıyla bir tahminleme çalışması yapmıştır. Bu nedenle kendi kendini organize eden harita olarak bilinen kümeleme analizi ile yapay sinir ağı modeli kurulmuştur. Ayrıca bu iki modelin birleşimi ile hibrit bir model de kurulmuştur. Bu üç farklı model ile çalışan devir oranları tahmin edilerek önlem alınması amaçlanmıştır. Sonuç olarak en iyi sınıflandıran modelin hibrit model olduğu bulunmuştur.

Ajit (2016), küresel bir şirketin insan kaynakları verileriyle aşırı gradyan artırma (AGA) algoritmasını geçmişte sık kullanılan altı gözetimli sınıflandırıcıyla karşılaştırarak en iyi çalışan devir oranını tahmin etmeyi amaçlamıştır. AGA, lojistik regresyon (LR), naive bayes (NB), rastgele orman (RO), k en yakın komşu (k-EYK), doğrusal diskriminant analiz (DDA) ve destek vektör makineleri (DVM) ile karşılaştırılmıştır. Çalışan devrini tahmin etmek için AGA sınıflandırma modelinin önemli ölçüde daha yüksek doğruluğa ulaştığı görülmüştür.

Ribes ve ark. (2017) ise farklı veri seti örneklemleri ile müşteri kaybını en iyi tahmin eden müşteri kaybı tahmin modelinin bulunmasını amaçlanmıştır. NB, DDA, DVM ve RO modelleri farklı veri seti örneklemleri için kurulmuştur. Algoritmalar açısından ağaç temelli olanların en iyi performansı gösterdiği görülmüştür. Aralarında en iyi performansa sahip sınıflandırıcının ise RO modeli olduğu bulunmuştur. Bulunan sonuçlara göre müşteriyi elde tutma politikaları tasarlanmış ve test etmek için model çıktıları tartışılmıştır.

Sisodia ve ark. (2017) yapmış olduğu çalışmada Kaggle web (www.kaggle.com) sitesinden elde edilen insan kaynakları (İK) veri setine müşteri kayıp oranını tahmin edecek bir model uygulanarak herhangi bir organizasyonda müşteri kaybını optimize eden noktaların bulunmasını amaçlanmıştır. Bu veri setine k-EYK, DVM, NB, RO ve karar ağacı (KA) modelleri uygulanmıştır. RO modeli en yüksek doğruluğu verirken DVM modeli en düşük doğrulukla sınıflandırmıştır. Bu problem için RO algoritması ile sınıflandırma yapılması tavsiye edilmiştir.

Alamsyah ve Salma (2018) Endonezya'nın bir telekomünikasyon şirketinde NB, KA ve RO algoritmalarını kullanarak İK verilerini analiz etmiştir. Bu sınıflandırma modellerini karşılaştırararak en doğru tahmini yapan modeli bulmayı amaçlamışlardır. %96,6 ile NB,%88,7 ile KA ve %97,5 ile RO algoritma doğruluk değerlerine ulaşılmıştır. Müşteri kaybını en doğru tahmin eden ve en güvenilir sınıflandırma modelinin RO olduğu bulunmuştur.

Fang ve ark. (2018) Çin'e ait bir kuruluşun müşteri kayıp oranını tahmin etmek üzere bir vaka çalışması yapmışlardır. Analizde müşteri yaşının ve pozisyonunun müşteri kaybına etkileri incelenerek kayıp oranlarının tahmin edilmesi amaçlanmıştır. Bu çalışmada koşullu bir yarı markov modeli kullanılmıştır. Toplam çalışan sayısının 2015 yılı öncesinde azaldığı ancak 2015'ten 2016 yılına doğru hızla arttığı sonucuna varılmıştır. Her yıl tahmin edilen sayının gerçek değerden daha yüksek olduğu görülmüştür. Çalışma sonucunda markov modeli çalışan yaşına göre tahminde bulunduğunda doğruluk oranı %86,9 olarak gerçekleşirken, pozisyona göre tahmin ettiğinde doğruluk oranı %90,5

olarak bulunmuştur. Bu model ile müşteri kayıp oranı tahmin edilirken pozisyona göre model yazılması önerilmiştir.

Shah ve ark. (2020) Brezilya'da bir kurye şirketinde çalışan kuryelerin devamsızlık ve işten ayrılma oranlarını tahmin etmek amacı ile en iyi tahmini yapan modeli bulmak üzere makine öğrenmesi modelleri kurmuştur. Sığ sinir ağları (SSA), derin sinir ağları (DSA), KA, DVM ve RO sınıflandırma modelleri karşılaştırılmıştır. Bu modeller arasından %84,3 ile en yüksek doğruluğa sahip olan modelin RO olduğu bulunmuştur ve önerilmiştir. Önerilen model, çalışanların işe alım sırasındaki davranışlarını bilmek isteyen kuruluşlara faydalı bir mekanizma sağlayacak ve verimsiz veya sürekli olarak işe gelmeyen çalışanlara ödeme yapma maliyetini azaltabilecektir.

Literatür araştırması Çizelge 2.10'de özetlenmiştir.

Çizelge 2.10: Çalışan Kaybı ile ilgili Çalışmalar

Çalışma İsmi, Yılı, Yazarı	**Amaç**	**Kullanılan Modeller**	**Sonuç**
Teknoloji profesyonelleri için devir oranını tahmin etmek için hibrit veri madenciliği ve makine öğrenimi kümeleme analizini kullanma(Chin-Yuan vd., 2012)	Tayvan'daki teknoloji işletmelerinin 2012 yılında hızla yetenekli çalışanları kaybetmeleri ile birlikte çalışan devir oranını tahmin etme ihtiyacı doğmuştur. Üç farklı model ile çalışan devir oranları tahmin edilerek önlem alınması amaçlanmıştır.	1. K-means SOM 2. Backward Propagation Network 3. Hibrit model(1+2)	Sonuç olarak en iyi sınıflandıran modelin hibrit model olduğu bulunmuştur .
Makine Öğrenimi Algoritmaları Kullanan Kuruluşlarda Çalışan Devir Hızının Tahmini: Aşırı Gradyan Artırma Örneği(Ajit vd., 2016)	Küresel bir şirketin insan kaynakları verileriyle aşırı gradyan artırma (AGA) algoritmasının geçmişte sık kullanılan altı gözetimli sınıflandırıcıyla karşılaştırılarak en iyi çalışan devir oranının tahmin edilmesi amaçlanmıştır.	1.Lojistik Regresyon(LR) 2.Naïve Bayes(NB) 3.Rastgele Orman(RO) 4.K En Yakın Komşu (k-EYK) 5.Doğrusal Diskriminant Analiz(DDA) 6.Destek Vektör Makineleri(DVM) 7.Aşırı Gradyan Artırma Makineleri(AGA)	Çalışan devrini tahmin etmek için AGA sınıflandırma modelinin . 7. ondalığa (dahil) kadar diğer modellerden daha iyi ondalık performansa sahip olduğu ölçülmüştür.

Çizelge 2.10: Çalışan Kaybı ile ilgili Çalışmalar (devam)

Çalışma İsmi, Yılı, Yazarı	Amaç	Kullanılan Modeller	Sonuç
Müşteri devrini tahmin etme ve elde tutma politikaları tasarımı: bir vaka çalışması(Ribes vd., 2017)	Farklı veri seti örneklemleri ile müşteri kaybını en iyi tahmin eden müşteri kaybı tahmin modelinin bulunması amaçlanmıştır.	1.Naive Bayes (NB) 2.Doğrusal Diskriminant Analiz (DDA) 3.Destek Vektör Makineleri (DVM) 4.Rastgele Orman (RO)	Algoritmalar açısından ağaç temelli olanların en iyi performansı gösterdiği görülmüştür. Aralarında en iyi performansa sahip sınıflandırıcının ise RO modeli olduğu bulunmuştur. Bulunan sonuçlara göre müşteriyi elde tutma politikaları tasarlanmış ve test etmek için model çıktıları tartışılmıştır.
Makine Öğrenimi Modellerinin Değerlendirilmesi Müşteri Kaybı Tahmini (Sisodia vd., 2017)	Kaggle web sitesinden elde edilen insan kaynakları (İK) veri setine müşteri kayıp oranını tahmin edecek bir model uygulanarak herhangi bir organizasyonda müşteri kaybını optimize eden noktaların bulunması amaçlanmıştır.	1.K En Yakın Komşu(k-EYK) 2.Destek Vektör Makineleri(DVM) 3.Naïve Bayes(NB) 4.Karar Ağacı (KA) 5.Rastgele Orman (RO)	RO modeli en yüksek doğruluğu verirken DVM modeli en düşük doğrulukla sınıflandırmıştır. Bu problem için RO algoritması ile sınıflandırma yapılması tavsiye edilmiştir.
Müşteri Kaybı Tahmin Modelinin Karşılaştırmalı Çalışması (Alamsyah ve Salma, 2018)	Endonezya'nın bir telekomünikasyon şirketinde NB, KA ve RO algoritmaları kullanılarak İK verilerinin analiz edilip sınıflandırma modellerinden en doğru tahmini yapan modelin bulunması amaçlanmıştır.	1.Naïve Bayes(NB) 2.Karar Ağacı (KA) 3.Rastgele Orman (RO)	Müşteri kaybını en doğru tahmin edenin ve en güvenilir sınıflandırma modelinin RO olduğu bulunmuştur.
Müşteri Devir Hızını Tahmin Etme Modeli: Çinli İşletmelere İlişkin Bir Örnek Olay İncelemesi (Fang vd., 2018)	Analizde müşteri yaşının ve pozisyonunun müşteri kaybına etkileri incelenerek kayıp oranlarının tahmin edilmesi amaçlanmıştır.	Koşullu bir yarı Markov (SMK) modeli	Çalışma sonucunda markov modeli çalışan yaşına göre tahminde bulunduğunda doğruluk oranı %86,9 olarak gerçekleşirken, pozisyona göre tahmin ettiğinde doğruluk oranı %90,5 olarak bulunmuştur. Bu model ile müşteri kayıp oranı tahmin edilirken pozisyona göre model yazılması önerilmiştir.

Çizelge 2.10: Literatür Araştırmasının Özeti(devamı)

Çalışma İsmi, Yılı, Yazarı	Amaç	Kullanılan Modeller	Sonuç
İşyeri Devamsızlığının Öngörülmesi için Gelişmiş Derin Sinir Ağı (Shah vd., 2020)	Brezilya'da bir kurye şirketinde çalışan kuryelerin devamsızlık ve işten ayrılma oranlarını tahmin etmek amacı ile en iyi tahmini yapan modeli bulmak üzere makine öğrenmesi modelleri kurulmuştur.	1.Sığ Sinir Ağları(SSA) 2.Derin Sinir Ağları(DSA) 3.Karar Ağacı (KA) 4.Destek Vektör Makineleri(DVM) 5. Rastgele Orman (RO)	Sığ sinir ağları (SSA), derin sinir ağları (DSA), KA, DVM ve RO sınıflandırma modelleri karşılaştırılmıştır. Bu modeller arasından %84,3 ile en yüksek doğruluğa sahip olan modelin RO olduğu bulunmuştur ve önerilmiştir. Önerilen model, çalışanların işe alım sırasındaki davranışlarını bilmek isteyen kuruluşlara faydalı bir mekanizma sağlayacak ve verimsiz veya sürekli olarak işe gelmeyen çalışanlara ödeme yapma maliyetini azaltabilecektir.

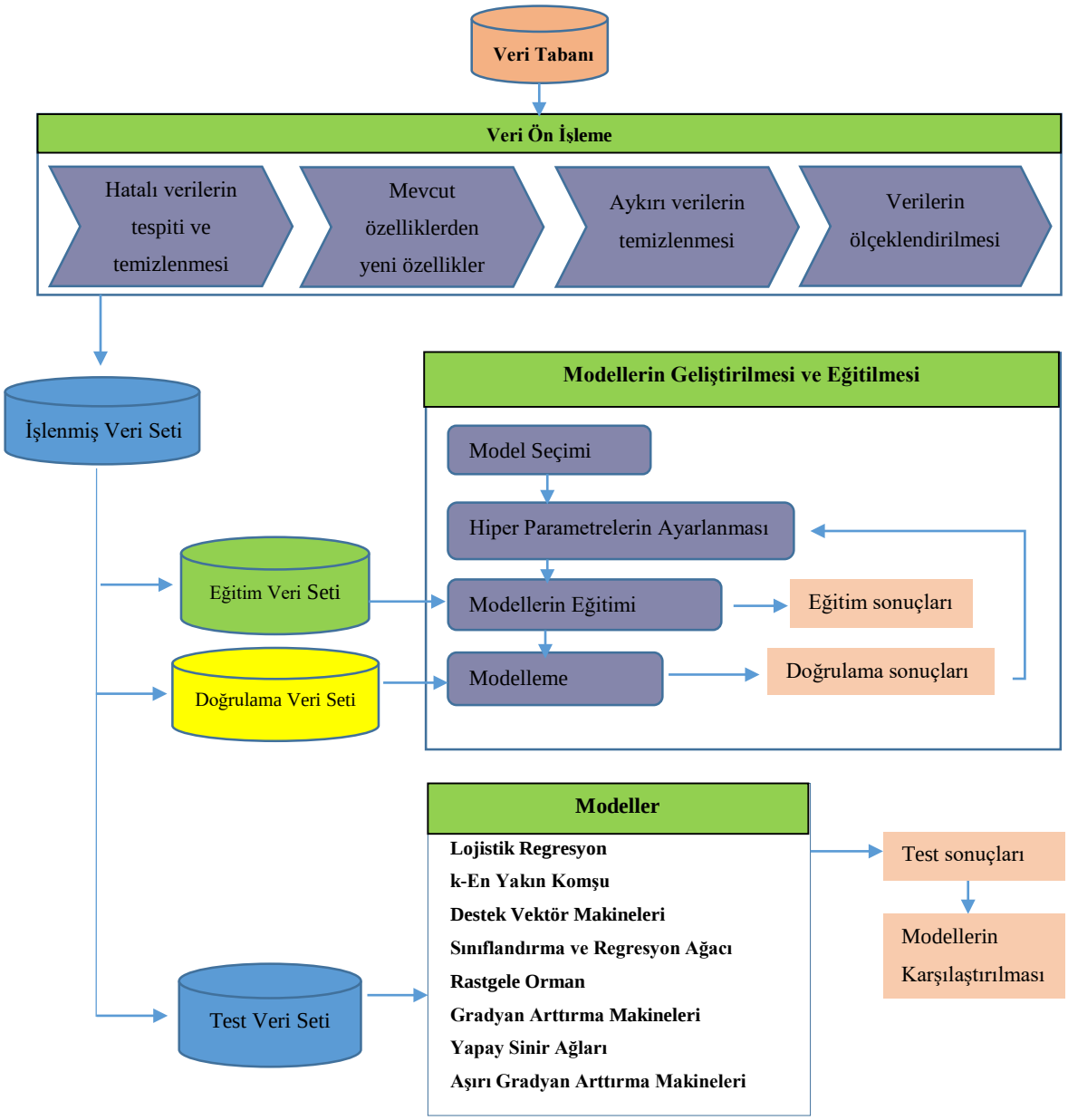
3. MATERYAL ve YÖNTEM

Bu bölümde ayrıntılı şekilde kullanılan veri seti ve algoritmalardan bahsedilmektedir.

3.1. Veri Tanımı ve Veri Ön İşleme

Bu uygulamada kullanılan veri seti gerçek bir telekomünikasyon şirketi verisinin veri gizliliği çerçevesinde maskelenerek türetilmesi ile oluşturulmuştur. Veri setinde telekomünikasyon şirketinde çalışanların istihdam durumu ve demografik verileri mevcuttur. Çalışmanın amacında belirtildiği gibi sınıflandırma modelleri kıyaslanarak problem özelinde çalışan kaybını en iyi tahmin eden sınıflandırma modeli bulunması amaçlanmaktadır.

Bu çalışmada izlenen yol haritası Şekil 3.1.'de gösterilmektedir. Bu yol haritasında bir veri analizi sürecinin uçtan uca tüm detayları bulunmaktadır. Veri tabanından ve farklı kaynaklardan elde edilen veri seti, veri ön işleme adımlarına tabi tutulmuştur. Hatalı veriler temizlenmiştir, mevcut değişkenlerden yeni bir değişken türetme ihtiyacı oluşmadığı için türetilmemiştir. Ancak klasik analiz modelinde olduğundan değerlendirilmesi için yol haritasına eklenmiştir. Yaş ve kıdem değişkenlerinde olan aykırı değerler tespit edilip temizlenmiştir. Daha sonra veri seti içerisinde daha anlamlı bir analiz yapabilmek için değişkenlere akan veri normalize edilerek ölçeklendirilmiştir. Veri ön işleme süreci tamamlandıktan sonra işlenmiş veri seti eğitim verisi, doğrulama verisi ve test verisi olmak üzere üç parçaya ayrılmıştır. Uygun bir makine öğrenmesi modeli seçildikten sonra eğitim verisi ile model eğitilmiştir. Doğrulama veri seti ile modelleme adımı tamamlanmıştır. Modelleme yapıldıktan sonra grid arama fonksiyonu ile modelin en iyi çalışabileceği parametreler belirlenmiştir. Test veri seti ile de sekiz farklı makine öğrenmesi modeli kurulup doğruluk, kesinlik, duyarlılık ve diğer metrikler ile modeller kıyaslanmıştır.



Şekil 3.1. Veri analizi modellerinin geliştirilme süreci

Çalışmada kullanılan veri seti Çizelge 3.1'de özetlenmiştir. Veri seti toplamda 16655 satır eşsiz (unique) çalışan kaydından oluşmaktadır. Veri toplamda on üç öznitelikten oluşmaktadır. Özniteliklerden beş tanesi nümerik, sekiz tanesi ise kategorik veri tipinden oluşmaktadır. Sicil değişkeni çalışanların sicil numaralarını, unvan değişkeni çalışanların şirket içerisindeki unvan bilgilerini, fonksiyon değişkeni çalışanların bağlı olduğu müdürlüğün ismini, kıdem değişkeni yıl bazında çalışanların şirket içerisindeki kıdemini, lokasyon değişkeni çalışanların bağlı olduğu lokasyon bilgisini, işten ayrılma nedeni

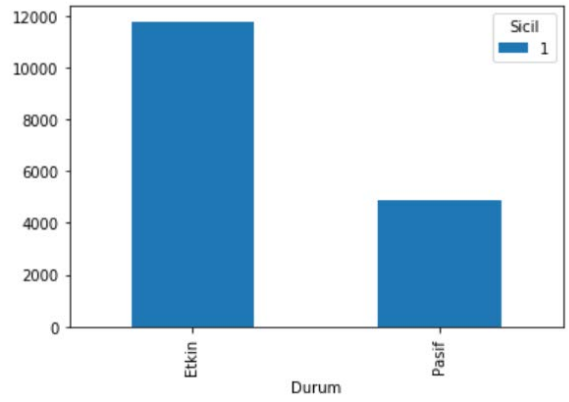
çalışanların neden işten ayrıldığını açıklar. Cinsiyet değişkeni çalışanların cinsiyetini, medeni hal değişkeni çalışanların medeni hallerini, çocuk sayısı değişkeni çalışanların çocuk sayısını, askerlik değişkeni erkek çalışanların askerlik durumunu, okul türü değişkeni çalışanların eğitim seviyesini, yaş değişkeni çalışanların yaşını ifade eder. Tek bağımlı değişken olan durum değişkeni ise çalışan işten ayrıldıysa pasif ifadesi ile çalışanın işten ayrıldığını, aktif ifadesi ile de çalışanın etkin olarak çalıştığını ifade eder.

Çizelge 3.1. Veri seti özeti

Değişken İsmi	**Minimum - Maksimum Değeri**	**Değişken Tipi**	**Değişken İçeriği**
Sicil	1	Nümerik	Örneğin; 50063021
Unvan	0-4	Kategorik	Müşteri Temsilcisi, Uzman, Takım Lideri, Birim Yöneticisi, Müdür
Fonksiyon	0-6	Kategorik	Planlama, IT, İnsan Kaynaklar, Finans, İdari İşler, Hukuk, Operasyon
Kıdem(yıl)	0-22	Nümerik	1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21, 22
Lokasyon	0-13	Kategorik	Trabzon, İzmir, Erzurum, Diyarbakır, Kayseri, İstanbul, Edirne, Rize, Bursa, Konya, Ankara, Şanlıurfa, Gaziantep, Sakarya
İşten Ayrılma Nedeni	0-21	Kategorik	Askerlik, Çalışma Şartlarına Uyumsuzluk, Devamsızlık, Eğitim, Emeklilik, Etik Neden, Evlilik, İşe Uygunluk, İşten Ayrılmadı, Kariyer Beklentisi, KPSS, Kreş, Küçülme, Maaş ve Yan Haklar, Ölüm, Performans, Sağlık, Sözleşme, Taşınma, Vardiya Düzeni, Yoğun Çalışma Saatleri, Yönetici Nedeniyle
Cinsiyet	0-1	Kategorik	Kadın, Erkek
Medeni Hal	0-1	Kategorik	Bekar, Evli
Çocuk Sayısı	0-4	Nümerik	1,2,3,4
Askerlik	0-3	Kategorik	Tamamlandı, Tecilli, Muaf, Yapılmadı
Okul Türü	0-4	Kategorik	Ön Lisans, Lisans, Master, Lise, Doktora
Yaş	25-60	Nümerik	18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,56,57,58,60,61
Durum	0-1	Kategorik	0: Pasif, 1: Aktif

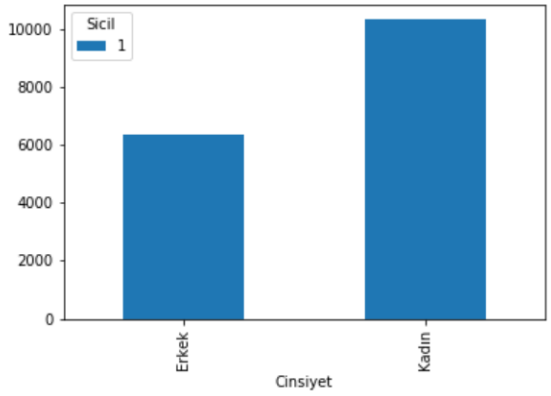
Çalışmanın veri setinde bulunan sicil özniteliğinin tamamı eşsiz nümerik verilerden oluştuğu için ve işten ayrılma nedenleri özniteliği ise *y* bağımlı değişkeninin gerçekleşmesi durumunda oluşan kategorik bir öznitelik olduğu için sonucu manipüle etmemeleri amacıyla veri setinden çıkarılmıştır. Ayrıca 121 yaşında görünen bir çalışanın yaşı model sonucunu manipüle etmemesi adına ortalama çalışan yaşı atanarak güncellenmiştir. Kıdemi yaşından büyük görünen iki çalışanın aykırı (outlier) kıdem değerleri ise ortalama kıdem değeri atanarak güncellenmiştir.

Bağımlı değişken olan durum özniteliğinin 11783 adet aktif (etkin) ve 4872 adet pasif çalışan şeklinde sınıflandırıldığı Şekil 3.2'de görülmektedir.



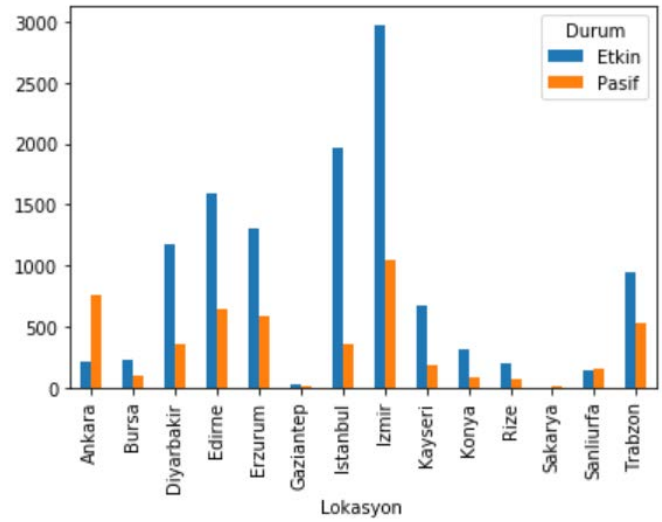
Şekil 3.2. Durum Bazında Çalışan Dağılımı

Cinsiyet bazında dağılım incelendiğinde kadın çalışanların erkek çalışanlardan sayıca daha fazla olduğu Şekil 3.3'de görülmektedir.



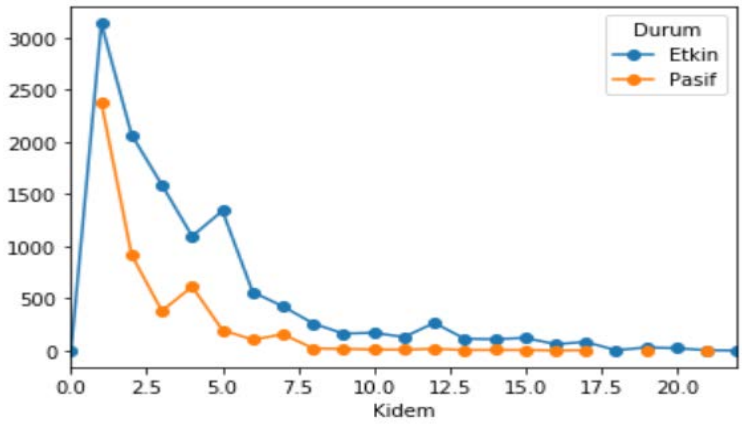
Şekil 3.3. Cinsiyet Bazında Çalışan Dağılımı

Lokasyon bazında istihdam durumu incelendiğinde ise Şekil 3.4'de görüldüğü gibi en yüksek aktif ve pasif çalışan sayısının İzmir lokasyonunda olduğu en düşük aktif çalışan sayısının ise Sakarya lokasyonunda bulunduğu görülmektedir. En düşük pasif çalışan sayısı ise Gaziantep ve Sakarya lokasyonlarında görülmektedir.



Şekil 3.4. Lokasyon Bazında İstihdam Durumu Dağılımı

Kıdem bazında istihdam durumu incelendiğinde Şekil 3.5'de görüldüğü gibi yaklaşık 2 yıl kıdeme sahip çalışanlarda hem aktif çalışan sayısının hem de pasif çalışan sayısının yüksek olduğu görülmektedir. Ayrıca grafiğe bakıldığında şirketin büyük çoğunluğunu 2- 5 yıl kıdem aralığındaki kişilerin oluşturduğu söylenebilir



Şekil 3.5. Kıdem Bazında İstihdam Durumu Dağılımı

Veri özetinden de anlaşılacağı üzere veri setinin büyük çoğunluğu kategorik veriden oluşmaktadır. Bu çalışmada özniteliklerin sınıflandırma modellerinde çalışabilmesi için nümerik hale çevrilmesi gerekmektedir. Kategorik özniteliklerin nümerik özniteliklere dönüştürülmesi ise kodlama (encoding) işlemi ile gerçekleştirilmektedir. Bu nedenle çalışmada veri yapısına göre çeşitli kodlama yöntemleri kullanılmıştır.

3.1.1. Etiket Kodlama

Kategorik verilerin nümerik değerlere dönüştürüldüğü yöntemdir (Zhuang, 2015). Bu çalışmada iki sınıftan oluşan cinsiyet, medeni hal ve bağımlı değişken olan durum öznitelikleri etiket kodlayıcı (label encoder) fonksiyonu ile kategorik veri tipinden nümerik veri tipine dönüştürülmüştür.

3.1.2. Sıralı Kodlama

Kodlama işleminde dikkat edilmesi gereken kısım özniteliklerin sıralı olup olmadığıdır. Örneğin; lokasyon değişkeni kategorik bir özniteliktir. Bu öznitelik nümerik hale çevrilmek istendiğinde sıralı kodlama (label encoding) kullanılmaktadır. Bu işlem İstanbul < Bursa < Trabzon şeklinde lokasyonları nümerik değerin büyüklüğüne göre ağırlıklandırmaktadır. Bu şekilde çalıştırılan modellerde elde edilen sonuçlar ise sağlıklı olmayacaktır. Bu nedenle sıralı kodlama işlemi yalnızca aralarında sıra olan ve birbirine göre üstünlüğü olan veri tiplerinde kullanılabilmektedir (Choong, 2017).

Bu çalışmada kullanılan unvan ve okul türü isimli kategorik öznitelikler sıralı veri barındırdığı için bu yöntem kullanılarak nümerik hale çevrilmiştir.

3.1.3. One Hot Kodlama

Aralarında bir sıra olmayan ve iki ve üzeri sınıftan oluşan veri tiplerinde bu yöntem kullanılmaktadır (Choong, 2017). Bu çalışmada fonksiyon, lokasyon ve askerlik öznitelikleri kategorik veri tipinden nümerik veri tipine bu yöntem ile dönüştürülmüştür.

Kodlama işleminden sonra öznitelikler arası ilişkilerin incelenebilmesi için oluşturulan öznitelik korelasyon matrisi Şekil 3.6'de verilmiştir. Durum bağımlı değişkeni ile en ilişkili olan özniteliğin kıdem, en ilişkisiz özniteliğin ise cinsiyet olduğu bulunmuştur. Ek olarak kıdem değişkeni ile yaş değişkeninin arasında 0,59 oranında, çocuk sayısı değişkeni ile yaş değişkeni arasında 0,5 oranında, cinsiyet değişkeni ile askerlik değişkeni arasında 0,49 oranında doğru orantılı bir korelasyon bulunmuştur. Ters orantılı korelasyon ise 0,12 oranında cinsiyet değişkeni ve unvan değişkeni arasında, 0,023 oranında yaş ile cinsiyet arasında bulunmuştur.

Şekil 3.6: Öznitelik Korelasyon Matrisi

3.2. Uygulama

Bu uygulama i5-8265U Intel Core ortamında ve Python dili ile yazılmıştır. Veri kümesi eğitim, test ve doğrulama verisi olmak üzere üç parçaya ayrılarak kullanılmıştır. Sekiz farklı sınıflandırma algoritması ile model eğitilerek test edilmiştir. Bu çalışmada tüm sınıflandırma modellerinin hiper parametreleri grid arama metodu kullanılarak belirlenmiştir.

3.2.1. Lojistik Regresyon

Lojistik regresyon ile hedef değişkenin pasif olarak tanımlanan değerinin gerçekleşme olasılığı hesaplanmıştır. Hiper parametrelere çeşitli değerler verilerek en yüksek doğruluğu veren değerler araştırılmak istenmektedir. Bu çalışmada kullanılan parametreler ve grid arama tekniği ile bulunan en iyi değerleri Çizelge 3.2'de bulunmaktadır. Model bu en iyi değerlere göre kurulmuştur.

Çizelge 3.2. Lojistik regresyon modeli parametreleri

Lojistik Regresyon		
Parametre	**Grid Arama Metodu İle Test Edilen Değerler**	**En İyi Değer**
Ceza(penalty)	l2,l1	l2
Düzenlilik Parametresi(C)	np.logspace(3,7)	1000.0
Çözücü(solver)	newton-cg, liblinear	newton-cg
Rastgele Durum (random_state)	np.logspace(0,1)	1

3.2.2. k-En Yakın Komşu

Sınıflandırılmak istenilen yeni çalışanların önceki çalışan profillerinden en fazla kaçıncı yakınlıktaki komşusu ile yakınlığına bakılması gerektiğini bulmak için deneme yanılma yöntemi kullanılarak model birçok k değeri ile çalıştırılmıştır. En iyi k değeri Çizelge 3.3' de görüldüğü gibi 11 olarak bulunmuştur.

Çizelge 3.3. k-en yakın komşu sınıflandırma modeli en iyi komşuluk parametresi

Parametre	**Sınıflandırma Doğruluk Performansı(%)**										
	k:1	**k:2**	**k:3**	**k:4**	**k:5**	**k:6**	**k:7**	**k:8**	**k:9**	**k:10**	**k:11**
Doğruluk	88,14	87,51	88,14	87,96	88,92	87,96	88,14	88,26	89,10	88,77	89,13
Çapraz Doğrulama Skoru	84,78	82,92	86,34	84,12	86,67	84,90	86,34	84,93	86,28	84,66	86,01

Sınıflandırılmak istenen yeni çalışanın önceki çalışan profillerinden k tanesine yakınlığına grid arama metoduyla da bakılmıştır. Kıyaslandığında iki şekilde de en iyi doğruluk k değeri 11 olduğunda bulunmuştur. Çizelge 3.4 'de görüldüğü gibi modelin parametresi olan k değeri 1-50 komşu aralığında olacak şekilde model kurulmuştur. Yani yeni çalışan en fazla 50'nci komşusuna kadar bakılarak sınıflandırılmıştır. Uygun parametre seçimi için model değişik parametre değerleri ile çalıştırılmıştır.

Çizelge 3.4. k-en yakın komşu modeli parametreleri

k-En Yakın Komşu		
Parametre	**Grid Arama Metodu İle Test Edilen Değerler**	**En İyi Değer**
Komşu Sayısı	1-50	11
Uzaklık Ölçüm Metriği	Öklid, Minkowski	Öklid

3.2.3. Destek Vektör Makineleri

İki sınıf arasındaki ayrımın optimum olması amaçlanarak hiper düzlemi bulmak üzere çeşitli parametrelerle model çalıştırılmıştır. Çizelge 3.5'de görüldüğü gibi doğrusal olarak ayrılabilen veri setine uygun parametreler kullanılmıştır.

Çizelge 3.5. Destek Vektör Makineleri Modeli Parametreleri

Destek Vektör Makineleri		
Parametre	**Grid Arama Metodu İle Test Edilen Değerler**	**En İyi Değer**
Düzenlilik Parametresi(C)	1-10	8
Çekirdek Fonksiyonu(Kernel)	linear, rbf	Linear(Doğrusal)

Grid arama yöntemi ile hızlı bir şekilde yapılan arama işlemi Çizelge 3.6'da deneme yanılma yöntemiyle de yapılmış ve sonuç yine aynı çıkmıştır.

Çizelge 3.6. Destek vektör makineleri modeli uygun parametre değeri seçimi

Çekirdek Fonksiyonu	**C (Marj) Değeri**	**Doğruluk**	**Çapraz Doğrulama Skoru**
Doğrusal	C=1	%90,324	%90,223
	C=2	%90,384	%90,293
	C=3	%90,444	%90,223
	C=4	%90,384	%90,253
	C=5	%90,484	%90,223
	C=6	%90,384	%90,255
	C=7	%90,484	%90,257
	C=8	%90,693	%90,733

Çizelge 3.6. Destek vektör makineleri modeli uygun parametre değeri seçimi (devam)

Çekirdek Fonksiyonu	C (Marj) Değeri	Doğruluk	Çapraz Doğrulama Skoru
Doğrusal	C=9	%90,484	%90,485
	C=10	%90,484	%90,483
	C=11	%90,484	%90,484
	C=12	%90,484	%90,484
	C=13	%90,584	%90,584
	C=14	%90,584	%90,584
	C=15	%90,554	%90,554
Radyal Tabanlı	C=1	%89,763	%88,652
	C=2	%89,823	%90,073
	C=3	%89,913	%89,913
	C=4	%90,003	%90,003
	C=5	%90,183	%90,183
	C=6	%90,183	%90,153
	C=7	%90,123	%90,163
	C=8	%90,213	%90,233
	C=9	%90,363	%90,203
	C=10	%90,363	%90,293
	C=11	%90,393	%90,123
	C=12	%90,453	%90,223
	C=13	%90,543	%90,163
	C=14	%90,603	%90,163
	C=15	%90,683	%90,316

Ceza parametresi C ve çekirdek fonksiyonu için hangi değerlerinin seçileceği kullanıcıya bırakılmıştır. DVM optimizasyonu içerisinde bu değerler belirlenmez. Kullanıcı tarafından sisteme her defasında bir parametre çiftinin girilip sonuç alınması ve eğer uygun değilse bir diğerinin denenmesi oldukça zahmetli ve zaman alıcı bir iştir. Dahası bu yolla C ve çekirdek fonksiyonu için gereken değerler uzayının çok küçük bir kısmında arama yapılabilir. Bu problemi çözmenin en kolay yolu grid arama tekniğidir (Hsu vd.,2004). Bu teknikte yüksek bir sınıflama doğruluk oranı veren, uygun parametre setinin belirlenmesi, çekirdek fonksiyonu ve C için belirlenen alt ve üst sınır içinde tüm farklı kombinasyonların denenmesi ile elde edilir.

Yerel bir arama tekniği olan grid aramada parametre değerleri için belirlenen aralığın iyi ayarlanması gerekmektedir (Lin vd., 2008). Çok geniş belirlenen aralık boşa geçen hesaplama zamanı anlamına gelirken, dar bir aralığın belirlenmesi ise tatmin edici sonuçların arama uzayının dışında bırakılması dolayısıyla iyi sonuçlardan vazgeçilmesi anlamına gelebilmektedir. DVM için uygun parametrenin belirlenmesi ayrı bir çalışma konusu olarak hala gelişme aşamasındadır (Tolun, 2008).

Burada DVM yönteminde kullanılacak parametrelerin seçimi için de grid arama tekniği kullanılmıştır. Bu veri seti için en iyi çekirdek fonksiyonu doğrusal fonksiyon ve en iyi C parametre değeri sekiz olarak bulunmuştur. Çekirdek fonksiyon tipi doğrusal fonksiyon olarak bulunduğu için orantısal olduğundan gamma operatörüne değer atanmamıştır.

3.2.4. Sınıflandırma ve Regresyon Ağacı

SRA ile veri setleri içerisindeki karmaşık yapıların basit karar yapılarına dönüştürülmesi amaçlanmıştır. Kullanılan bu model ile veri seti bağımlı değişkene göre homojen alt gruplara ayrılmıştır.

Çizelge 3.7 ’de görüldüğü gibi uygun parametre seçimi için model grid arama yöntemi kullanılarak değişik parametrelerle çalıştırılmıştır ve en iyi değerler bulunmuştur. SRA sınıflandırma modeli kurulurken bu parametre değerleri kullanılmıştır. Çizelge 3.7’de belirtilmeyen tüm parametrelerin varsayılan değerleri kullanılmıştır.

Çizelge 3.7. Sınıflandırma ve regresyon ağacı modeli parametreleri

Sınıflandırma ve Regresyon Ağacı		
Parametre	**Grid Arama Metodu İle Test Edilen Değerler**	**En İyi Değer**
Maksimum Derinlik(max_depth)	1,3,5,8,10	8
Minimum Örnek Sayısı(minSamples Split)	2,3,5,10,20,50	20

3.2.5. Rastgele Orman

RO ile karar ağacının her bir düğüm noktasında rastgele değişken seçimi yapılarak en iyi dallara ayrılması için her bir ağacın önceden belirlenen hata oranları göz önüne alınarak en iyi tahmin değerine ulaşılması amaçlanmıştır. Çizelge 3.8'de görüldüğü gibi uygun parametre seçimi için model grid arama yöntemi kullanılarak değişik parametrelerle çalıştırılmıştır. En iyi değerler bulunmuştur. RO sınıflandırma modeli kurulurken bu parametre değerleri kullanılmıştır. Çizelge 3.8'da belirtilmeyen tüm parametrelerin varsayılan değerleri kullanılmıştır.

Çizelge 3.8. Rastgele orman modeli en iyi parametreleri

Rastgele Orman		
Parametre	**Grid Arama Metodu İle Test Edilen Değerler**	**En İyi Değer**
Tahmin Sayısı (n_estimators)	100,200,500,1000	100
Maksimum Öznitelik Sayısı(max_features)	3,5,7,8	3
Minimum Örnek Sayısı(min_samples_split)	2,5,10,20	20

3.2.6. Gradyan Arttırma Makineleri

GA makineleri ile çalışanların en iyi şekilde sınıflandırılması için model birçok parametre değeri ile çalıştırılmıştır. En iyi parametreler Çizelge 3.9'da görüldüğü gibi bulunmuştur. GA sınıflandırma modeli kurulurken bu parametre değerleri kullanılmıştır. Çizelge 3.9'da belirtilmeyen tüm parametrelerin varsayılan değerleri kullanılmıştır.

Çizelge 3.9. Gradyan arttırma makineleri modeli en iyi parametreleri

Gradyan Arttırma Makineleri		
Parametre	**Grid Arama Metodu İle Test Edilen Değerler**	**En İyi Değer**
Öğrenme Oranı(learning_rate)	0.1, 0.01, 0.001, 0.05	0.01
Tahmin Sayısı(n_estimators)	100,200,500,1000	1000
Maksimum Derinlik(max_depth)	2,3,5,8	5

3.2.7.Yapay Sinir Ağları

YSA modeli ile aşamalı bir şekilde model inşa edilerek bir aktivasyon fonksiyonu ile model çalıştırılmaktadır. Çok katmanlı YSA'nın kullanması gereken önemli parametreler vardır. İterasyon, aktivasyon, çözücü ve öğrenme oranı başlıca parametrelerdir.

YSA ile çalışanların en iyi şekilde sınıflandırılması için model birçok parametre değeri ile çalıştırılmıştır. En iyi parametreler Çizelge 3.10'da görüldüğü gibi bulunmuştur. YSA sınıflandırma modeli kurulurken bu parametre değerleri kullanılmıştır. Çizelge 3.10'da belirtilmeyen tüm parametrelerin varsayılan değerleri kullanılmıştır.

Çizelge 3.10.Yapay sinir ağları parametreleri

Yapay Sinir Ağları		
Parametre	**Grid Arama Metodu İle Test Edilen Değerler**	**En İyi Değer**
alpha	1, 5, 0.1, 0.01, 0.03, 0.005, 0.0001	0.01
Gizli Katman Sayısı(hidden_layer_sizes)	(10,10), (100,100,100), (100,100), (3,5)	(10, 10)
Çözücü(solver)	lbfgs,adam	lbfgs
Aktivasyon Fonksiyonu (activation)	relu,sigmoid	relu

3.2.8. Aşırı Gradyan Arttırma Makineleri

AGA makineleri ile aşamalı bir şekilde model inşa edilerek bir kayıp fonksiyonu ile modeller serisi oluşturulmuştur. Seri içerisindeki bir model serideki bir önceki modelin tahmin hataları ile oluşturularak ilerlemiştir.

Bu çalışmada AGA modelinde kullanılan parametrelerin en iyi değerleri grid arama tekniği ile bulunmaktadır. En iyi parametreler Çizelge 3.11'de görüldüğü gibi bulunmuştur. AGA sınıflandırma modeli kurulurken bu parametre değerleri kullanılmıştır. Çizelge 3.11'de belirtilmeyen tüm parametrelerin varsayılan değerleri kullanılmıştır.

Çizelge 3.11. Aşırı gradyan arttırma makineleri modeli en iyi parametreleri

Aşırı Gradyan Artırma Makineleri		
Parametre	**Grid Arama Metodu İle Test Edilen Değerler**	**En İyi Değer**
Öğrenme Oranı (learning_rate)	0.1, 0.01, 0.001	0.1
Örneklerin Oranı(subsample)	0.6, 0.8, 1	0.8
Tahmin Sayısı (n_estimators)	100,500,1000,2000	100
Maksimum Derinlik (max_depth)	3,5,7	5

4. BULGULAR

4.1. Lojistik Regresyon Modeli Bulguları

Belirlenen hiper parametre değerleri ile kurulan lojistik regresyon sınıflandırma modelinin performans ölçüm metriklerinin sonuçları Çizelge 4.1'de gösterilmektedir. %90,8 oranında doğru sınıflandırma yapılmıştır. Bu sonuç %95 oranında kesindir. 10 kez çapraz doğrulama işlemi yapılmıştır ve sonuç bulunan doğruluk değerinden daha yüksek çıkmıştır. Modelin sınıflandırma duyarlılığı %91 oranında çıkmıştır. Kesinlik ve duyarlılık metriklerinin harmonik ortalaması olan f_1 ölçütü ise %93 olarak bulunmuştur. Performansı ifade eden ve bir AİK eğrisinin altında kalan alan ise %90 olarak bulunmuştur.

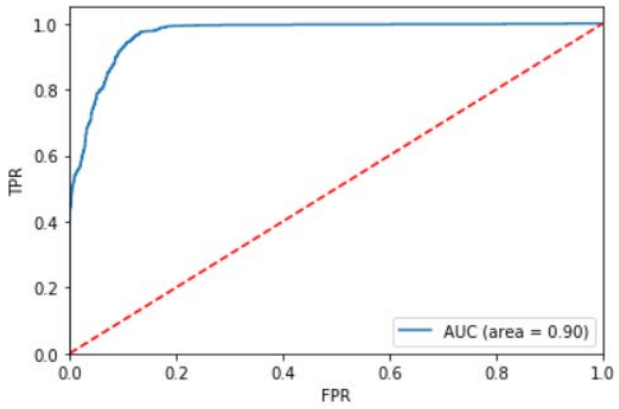
Çizelge 4.1. Lojistik regresyon sınıflandırma modeli sonuçları

Lojistik Regresyon Sınıflandırma Modeli Sonuçları					
Doğruluk	Çapraz Doğrulama Skoru	Kesinlik	Duyarlılık	f_1 Ölçütü	EAKA
0,908	0,911	0,95	0,91	0,93	0,90

Model sonuçlarının hata matrisinde gösterimi ise Şekil 4.1'de görülmektedir. Gerçekte pasif olup modelin aktif olarak sınıflandırdığı çalışan sayısı 204, gerçekte aktif olup modelin pasif olarak sınıflandırdığı çalışan sayısı 103 kişidir. Gerçekte aktif olup modelle aktif olarak sınıflandırılanların sayısı 851 ve gerçekte pasif olup modelle pasif olarak sınıflandırılanların sayısı ise 2173 adettir.

Şekil 4.1. Lojistik Regresyon Sınıflandırma Modeli Hata Matrisi

Şekil 4.2.'de bulunan grafikte FPR ifadesi ile isimlendirilen x ekseni yanlış pasif oranlarını, TPR ifadesi ile isimlendirilen y ekseni ise doğru pasif oranlarını ifade etmektedir. AİK grafiğinde eğrinin altında kalan alan EAKA ile tahmin performansının oranı ifade edilmektedir. EAKA metriği için ideal değer 1'dir. Şekil 4.2'de görüldüğü gibi bu çalışmada kullanılan lojistik regresyon modelinde EAKA değerinin %90 olarak gerçekleştiği görülmüştür.



Şekil 4.2. Lojistik Regresyon AİK eğrisi

4.2. k-En Yakın Komşu Modeli Bulguları

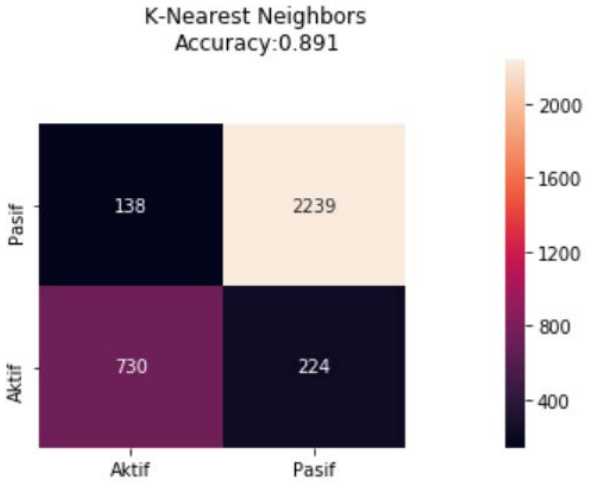
Belirlenen hiper parametre değerleri ile kurulan k-en yakın komşu sınıflandırma modelinin performans ölçüm metriklerinin sonuçları Çizelge 4.2'de gösterilmektedir.

%89,1 oranında doğru sınıflandırma yapılmıştır. Bu sonuç %91 oranında kesindir. 10 kez çapraz doğrulama işlemi yapılmıştır ve sonuç bulunan doğruluk değerinden daha düşük çıkmıştır. Modelin sınıflandırma duyarlılığı %94 oranında çıkmıştır. Kesinlik ve duyarlılık metriklerinin harmonik ortalaması olan f_1 ölçütü ise %93 olarak bulunmuştur. Performansı ifade eden ve bir AİK eğrisinin altında kalan alan ise %85 olarak bulunmuştur.

Çizelge 4.2. k-en yakın komşu sınıflandırma modeli sonuçları

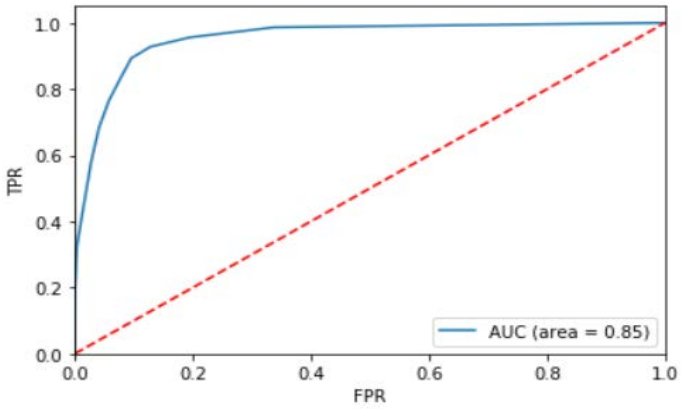
k-En Yakın Komşu Sınıflandırma Modeli Sonuçları					
Doğruluk	Çapraz Doğrulama Skoru	Kesinlik	Duyarlılık	f_1 Ölçütü	EAKA
0,891	0,86	0,91	0,94	0,93	0,85

Model sonuçlarının hata matrisinde gösterimi ise Şekil 4.3'de görülmektedir. Gerçekte pasif olup modelin aktif olarak sınıflandırdığı çalışan sayısı 138, gerçekte aktif olup modelin pasif olarak sınıflandırdığı çalışan sayısı 224'tür.Gerçekte aktif olup modelle aktif olarak sınıflandırılanların sayısı 730 ve gerçekte pasif olup modelle pasif olarak sınıflandırılanların sayısı ise 2239 adettir.



Şekil 4.3. k-En Yakın Komşu Modeli Hata Matrisi

Şekil 4.4'de görüldüğü gibi bu çalışmada kullanılan k-en yakın komşu modelinde modelin performans başarısını ifade eden AİK grafiğinde eğrinin altında kalan alan EAKA değerinin %85 olarak gerçekleştiği görülmüştür.



Şekil 4.4. k-En Yakın Komşu Modeli AİK eğrisi

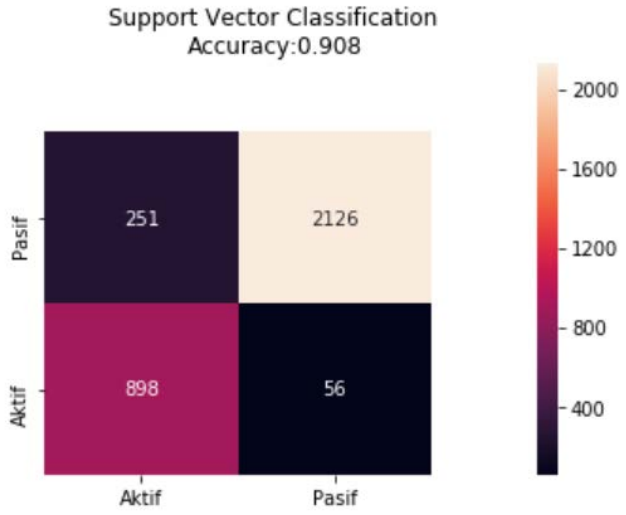
4.3. Destek Vektör Makineleri Modeli Bulguları

Belirlenen hiper parametre değerleri ile kurulan destek vektör makineleri sınıflandırma modelinin performans ölçüm metriklerinin sonuçları Çizelge 4.3'de gösterilmektedir. %90,7 oranında doğru sınıflandırma yapılmıştır. Bu sonuç %97 oranında kesindir. 10 kez çapraz doğrulama işlemi yapılmıştır ve sonuç bulunan doğruluk değeri ile benzer çıkmıştır. Modelin sınıflandırma duyarlılığı %89 oranında çıkmıştır. Kesinlik ve duyarlılık metriklerinin harmonik ortalaması olan f_1 ölçütü ise %93 olarak bulunmuştur. Performansı ifade eden ve bir AİK eğrisinin altında kalan alan ise %92 olarak bulunmuştur.

Çizelge 4.3. Destek vektör makineleri sınıflandırma modeli sonuçları

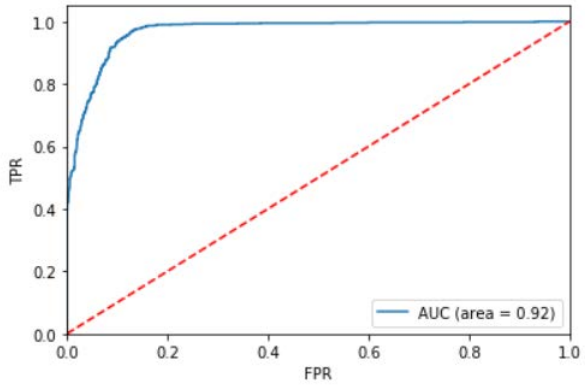
Destek Vektör Makineleri Sınıflandırma Modeli Sonuçları					
Doğruluk	Çapraz Doğrulama Skoru	Kesinlik	Duyarlılık	f_1 Ölçütü	EAKA
0,907	0,906	0,97	0,89	0,93	0,92

Model sonuçlarının hata matrisinde gösterimi ise Şekil 4.5.‘de görülmektedir. Gerçekte pasif olup modelin aktif olarak sınıflandırdığı çalışan sayısı 251, gerçekte aktif olup modelin pasif olarak sınıflandırdığı çalışan sayısı 56’dır. Gerçekte aktif olup modelle aktif olarak sınıflandırılanların sayısı 898 ve gerçekte pasif olup modelle pasif olarak sınıflandırılanların sayısı ise 2126 adettir.



Şekil 4.5. Destek Vektör Makineleri Modeli Hata Matrisi

Şekil 4.6.’de görüldüğü gibi bu çalışmada kullanılan destek vektör makineleri sınıflandırma modelinde modelin performans başarısını ifade eden AİK grafiğinde eğrinin altında kalan alan EAKA değerinin %92 olarak gerçekleştiği görülmüştür.



Şekil 4.6. Destek Vektör Makineleri AİK eğrisi

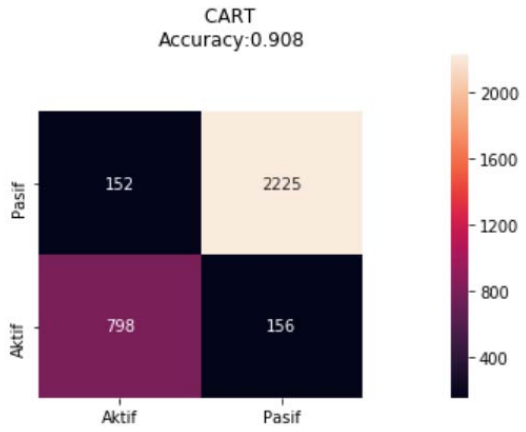
4.4. Sınıflandırma ve Regresyon Ağacı Modeli Bulguları

Belirlenen hiper parametre değerleri ile kurulan sınıflandırma ve regresyon ağacı sınıflandırma modelinin performans ölçüm metriklerinin sonuçları Çizelge 4.4'de gösterilmektedir. %90,7 oranında doğru sınıflandırma yapılmıştır. Bu sonuç %93 oranında kesindir. 10 kez çapraz doğrulama işlemi yapılmıştır ve sonuç bulunan doğruluk değeri ile benzer çıkmıştır. Modelin sınıflandırma duyarlılığı %94 oranında çıkmıştır. Kesinlik ve duyarlılık metriklerinin harmonik ortalaması olan f_1 ölçütü ise %94 olarak bulunmuştur. Performansı ifade eden ve bir AİK eğrisinin altında kalan alan ise %89 olarak bulunmuştur.

Çizelge 4.4. Sınıflandırma ve regresyon ağacı sınıflandırma modeli sonuçları

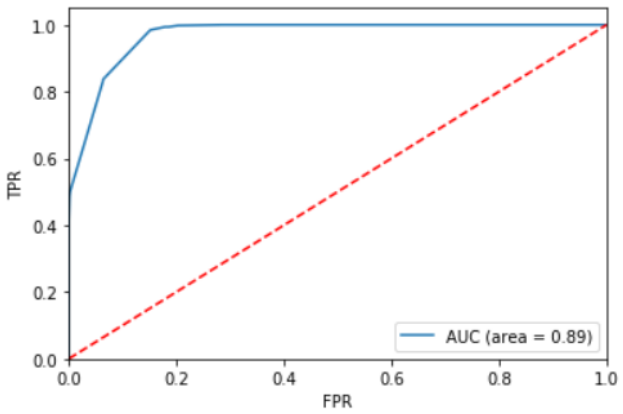
Sınıflandırma ve Regresyon Ağacı Sınıflandırma Modeli Sonuçları					
Doğruluk	Çapraz Doğrulama Skoru	Kesinlik	Duyarlılık	f_1 Ölçütü	EAKA
0,907	0,905	0,93	0,94	0,94	0,89

Model sonuçlarının hata matrisinde gösterimi ise Şekil 4.7'de görülmektedir. Gerçekte pasif olup modelin aktif olarak sınıflandırdığı çalışan sayısı 152, gerçekte aktif olup modelin pasif olarak sınıflandırdığı çalışan sayısı 156'dır. Gerçekte aktif olup modelle aktif olarak sınıflandırılanların sayısı 798 ve gerçekte pasif olup modelle pasif olarak sınıflandırılanların sayısı ise 2225 adettir.



Şekil 4.7 Sınıflandırma ve Regresyon Ağacı Sınıflandırma Modeli Hata Matrisi

Şekil 4.8'de görüldüğü gibi bu çalışmada kullanılan sınıflandırma ve regresyon ağacı sınıflandırma modelinde modelin performans başarısını ifade eden AİK grafiğinde eğrinin altında kalan alan EAKA değerinin %89 olarak gerçekleştiği görülmüştür.



Şekil 4.8. Sınıflandırma ve Regresyon Ağacı AİK eğrisi

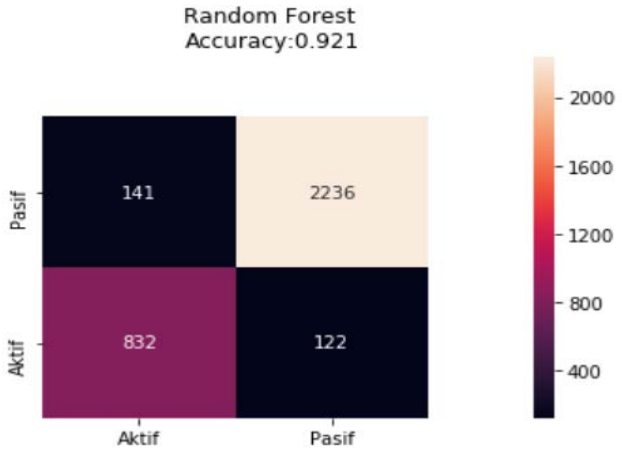
4.5. Rastgele Orman Modeli Bulguları

Belirlenen hiper parametre değerleri ile kurulan rastgele orman sınıflandırma modelinin performans ölçüm metriklerinin sonuçları Çizelge 4.5'de gösterilmektedir. %92,2 oranında doğru sınıflandırma yapılmıştır. Bu sonuç %95 oranında kesindir. 10 kez çapraz doğrulama işlemi yapılmıştır ve sonuç bulunan doğruluk değerinden daha düşük çıkmıştır. Modelin sınıflandırma duyarlılığı %94 oranında çıkmıştır. Kesinlik ve duyarlılık metriklerinin harmonik ortalaması olan f_1 ölçütü ise %94 olarak bulunmuştur. Performansı ifade eden ve bir AİK eğrisinin altında kalan alan ise %91 olarak bulunmuştur.

Çizelge 4.5. Rastgele orman sınıflandırma modeli sonuçları

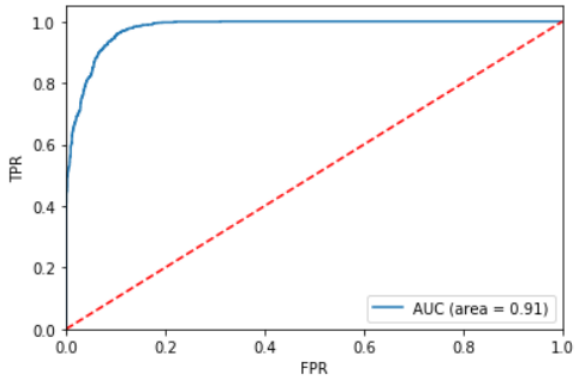
Rastgele Orman Sınıflandırma Modeli Sonuçları					
Doğruluk	Çapraz Doğrulama Skoru	Kesinlik	Duyarlılık	f_1 Ölçütü	EAKA
0,922	0,911	0,95	0,94	0,94	0,91

Model sonuçlarının hata matrisinde gösterimi ise Şekil 4.9'de görülmektedir. Gerçekte pasif olup modelin aktif olarak sınıflandırdığı çalışan sayısı 141, gerçekte aktif olup modelin pasif olarak sınıflandırdığı çalışan sayısı 122 kişidir. Gerçekte aktif olup modelle aktif olarak sınıflandırılanların sayısı 832 ve gerçekte pasif olup modelle pasif olarak sınıflandırılanların sayısı ise 2236 adettir.



Şekil 4.9. Rastgele Orman Modeli Hata Matrisi

Şekil 4.10'da görüldüğü gibi bu çalışmada kullanılan rastgele orman sınıflandırma modelinde modelin performans başarısını ifade eden AİK grafiğinde eğrinin altında kalan alan EAKA değerinin %91 olarak gerçekleştiği görülmüştür.



Şekil 4.10. Rastgele Orman AİK eğrisi

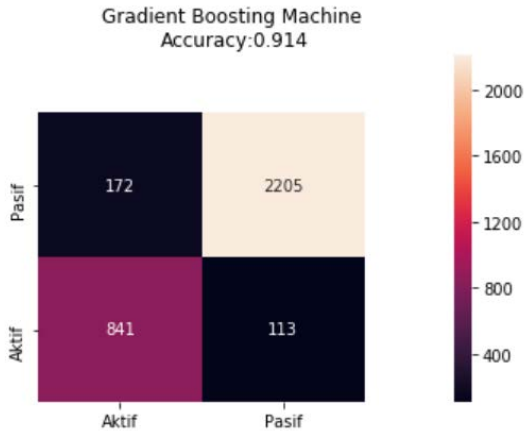
4.6. Gradyan Artırma Makineleri Modeli Bulguları

Belirlenen hiper parametre değerleri ile kurulan gradyan artırma makineleri sınıflandırma modelinin performans ölçüm metriklerinin sonuçları Çizelge 4.6'de gösterilmektedir. %91,4 oranında doğru sınıflandırma yapılmıştır. Bu sonuç %95 oranında kesindir. 10 kez çapraz doğrulama işlemi yapılmıştır ve sonuç bulunan doğruluk değeri ile benzer çıkmıştır. Modelin sınıflandırma duyarlılığı %93 oranında çıkmıştır. Kesinlik ve duyarlılık metriklerinin harmonik ortalaması olan f_1 ölçütü ise %94 olarak bulunmuştur. Performansı ifade eden ve bir AİK eğrisinin altında kalan alan ise %90 olarak bulunmuştur.

Çizelge 4.6. Gradyan artırma makineleri sınıflandırma modeli sonuçları

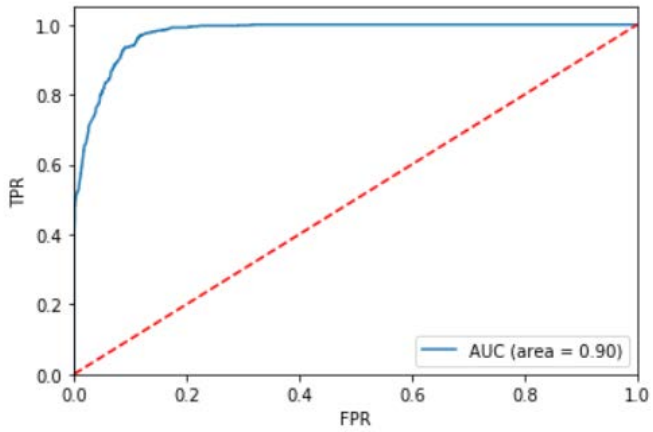
Gradyan Artırma Makineleri Sınıflandırma Modeli Sonuçları					
Doğruluk	Çapraz Doğrulama Skoru	Kesinlik	Duyarlılık	f_1 Ölçütü	EAKA
0,914	0,912	0,95	0,93	0,94	0,90

Model sonuçlarının hata matrisinde gösterimi ise Şekil 4.11' de görülmektedir. Gerçekte pasif olup modelin aktif olarak sınıflandırdığı çalışan sayısı 172, gerçekte aktif olup modelin pasif olarak sınıflandırdığı çalışan sayısı 113 kişidir. Gerçekte aktif olup modelle aktif olarak sınıflandırılanların sayısı 841 ve gerçekte pasif olup modelle pasif olarak sınıflandırılanların sayısı ise 2205 adettir.



Şekil 4.11. Gradyan Arttırma Makineleri Modeli Hata Matrisi

Şekil 4.12'de görüldüğü gibi bu çalışmada kullanılan gradyan arttırma makineleri modelinde modelin performans başarısını ifade eden AİK grafiğinde eğrinin altında kalan alan EAKA değerinin %90 olarak gerçekleştiği görülmüştür.



Şekil 4.12. Gradyan Arttırma Makineleri Modeli AİK eğrisi

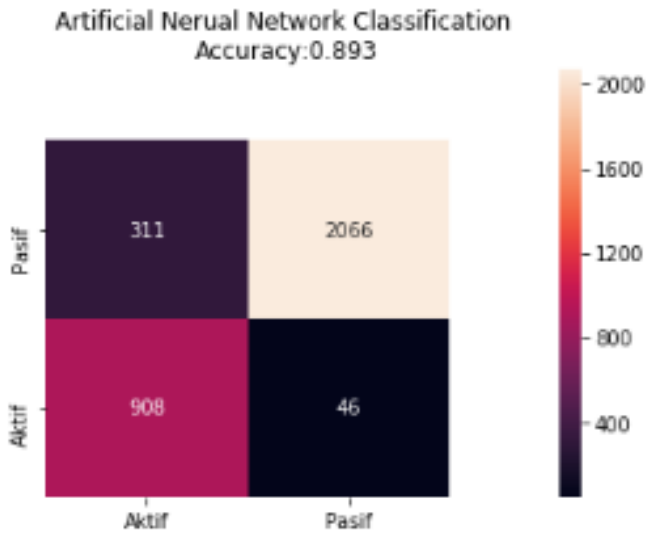
4.7. Yapay Sinir Ağları Modeli Bulguları

Belirlenen hiper parametre değerleri ile kurulan yapay sinir ağları sınıflandırma modelinin performans ölçüm metriklerinin sonuçları Çizelge 4.7.'de gösterilmektedir. %89,3 oranında doğru sınıflandırma yapılmıştır. Bu sonuç %98 oranında kesindir. 10 kez çapraz doğrulama işlemi yapılmıştır ve sonuç bulunan doğruluk değerinden daha yüksek çıkmıştır. Modelin sınıflandırma duyarlılığı %87 oranında çıkmıştır. Kesinlik ve duyarlılık metriklerinin harmonik ortalaması olan f_1 ölçütü ise %92 olarak bulunmuştur. Performansı ifade eden ve bir AİK eğrisinin altında kalan alan ise %91 olarak bulunmuştur.

Çizelge 4.7. Yapay sinir ağları sınıflandırma modeli sonuçları

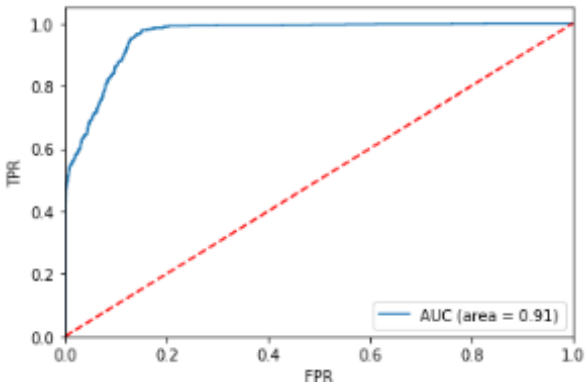
Yapay Sinir Ağları Sınıflandırma Modeli Sonuçları					
Doğruluk	Çapraz Doğrulama Skoru	Kesinlik	Duyarlılık	f_1 Ölçütü	EAKA
0,893	0,904	0,98	0,87	0,92	0,91

Model sonuçlarının hata matrisinde gösterimi ise Şekil 4.13.'de görülmektedir. Gerçekte pasif olup modelin aktif olarak sınıflandırdığı çalışan sayısı 311, gerçekte aktif olup modelin pasif olarak sınıflandırdığı çalışan sayısı 46 kişidir. Gerçekte aktif olup modelle aktif olarak sınıflandırılanların sayısı 908 ve gerçekte pasif olup modelle pasif olarak sınıflandırılanların sayısı ise 2066 adettir.



Şekil 4.13. Yapay Sinir Ağları Modeli Hata Matrisi

Şekil 4.14'de görüldüğü gibi bu çalışmada kullanılan yapay sinir ağları modelinde modelin performans başarısını ifade eden AİK grafiğinde eğrinin altında kalan alan EAKA değerinin %91 olarak gerçekleştiği görülmüştür.



Şekil 4.14. Yapay Sinir Ağları Modeli AİK eğrisi

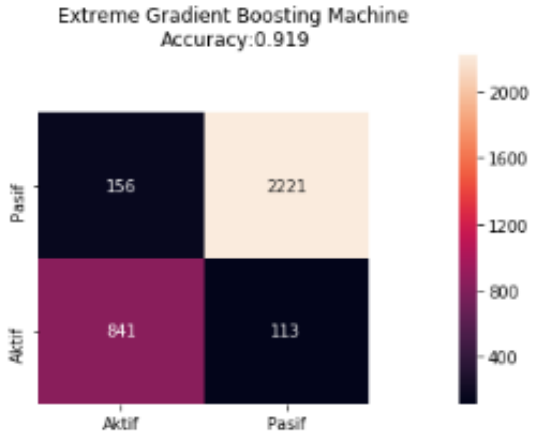
4.8. Aşırı Gradyan Artırma Makineleri Modeli Bulguları

Belirlenen hiper parametre değerleri ile kurulan aşırı gradyan artırma makineleri sınıflandırma modelinin performans ölçüm metriklerinin sonuçları Çizelge 4.8'de gösterilmektedir. %91,9 oranında doğru sınıflandırma yapılmıştır. Bu sonuç %95 oranında kesindir. 10 kez çapraz doğrulama işlemi yapılmıştır ve sonuç bulunan doğruluk değerinden daha düşük çıkmıştır. Modelin sınıflandırma duyarlılığı %93 oranında çıkmıştır. Kesinlik ve duyarlılık metriklerinin harmonik ortalaması olan f_1 ölçütü ise %94 olarak bulunmuştur. Performansı ifade eden ve bir AİK eğrisinin altında kalan alan ise %91 olarak bulunmuştur.

Çizelge 4.8. Aşırı gradyan artırma makineleri sınıflandırma modeli sonuçları

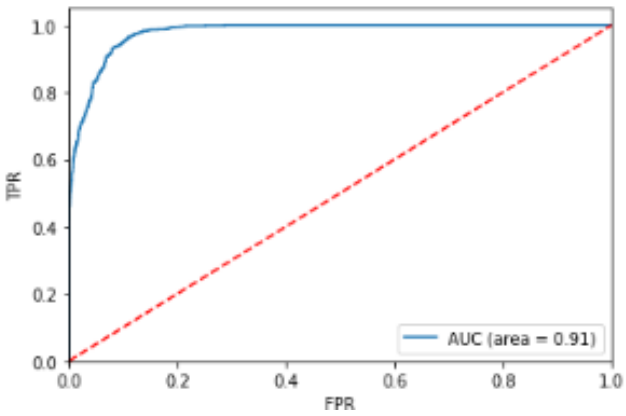
Aşırı Gradyan Artırma Makineleri Sınıflandırma Modeli Sonuçları					
Doğruluk	Çapraz Doğrulama Skoru	Kesinlik	Duyarlılik	f_1 Ölçütü	EAKA
0,919	0,905	0,95	0,93	0,94	0,91

Model sonuçlarının hata matrisinde gösterimi ise Şekil 4.15'de görülmektedir. Gerçekte pasif olup modelin aktif olarak sınıflandırdığı çalışan sayısı 156, gerçekte aktif olup modelin pasif olarak sınıflandırdığı çalışan sayısı 113 kişidir. Gerçekte aktif olup modelle aktif olarak sınıflandırılanların sayısı 841 ve gerçekte pasif olup modelle pasif olarak sınıflandırılanların sayısı ise 2221 adettir.



Şekil 4.15. Aşırı Gradyan Arttırma Makineleri Hata Matrisi

Şekil 4.16'de görüldüğü gibi bu çalışmada kullanılan ekstra gradyan arttırma makineleri modelinde modelin performans başarısını ifade eden AİK grafiğinde eğrinin altında kalan alan EAKA değerinin %91 olarak gerçekleştiği görülmüştür.



Şekil 4.16. Aşırı Gradyan Arttırma Makineleri Modeli AİK eğrisi

4.9. Tüm Modellerin Karşılaştırılma Bulguları

Çizelge 4.9'da görüldüğü gibi her bir metrik yüksek değerden düşük değere doğru renklendirilmiştir. En yüksek değeri yeşil en düşük değeri ise kırmızı rengin ifade edeceği şekilde renklendirme yapılmıştır.

Çizelge 4.9. Kullanılan tüm modellerin karşılaştırılması

TÜM MODELLERİN KARŞILAŞTIRILMASI								
Metrikler	**Lojistik Regresyon**	**K-En Yakın Komşu**	**Destek Vektör Makineleri**	**Sınıflandırma ve Regresyon Ağacı**	**Rastgele Orman**	**Gradyan Attırma Makineleri**	**Yapay Sinir Ağları**	**Aşırı Gradyan Arttırma Makineleri**
Doğruluk	0,908	0,891	0,907	0,907	0,922	0,914	0,893	0,919
Çapraz Doğrulama Skoru	0,911	0,860	0,906	0,905	0,911	0,912	0,904	0,905
Kesinlik	0,950	0,910	0,970	0,930	0,950	0,950	0,980	0,950
Duyarlılık	0,910	0,940	0,890	0,940	0,940	0,930	0,870	0,930
***f1* Ölçütü**	0,930	0,930	0,930	0,940	0,940	0,940	0,920	0,940
EAKA	0,900	0,850	0,920	0,890	0,910	0,900	0,910	0,910

Sonuç olarak yeşil renklerin en yoğun olduğu yani her metrikte en yüksek değere ulaşılan RO sınıflandırma modeli %92,2 doğruluk oranı ile bu veri setini en iyi sınıflandıran modeldir. En iyi ikinci model ise literatür araştırmasında da özellikle son yıllarda öne çıkan GAA makineleri olmuştur. Uygulanan modeller arasında en kötü model ise %89 doğruluk oranı ile k-EYK sınıflandırma modeli olmuştur. Bu model tüm metriklerde en düşük değere sahip olmasına rağmen en yüksek üç duyarlılık metriğinden birine ulaşmıştır.

5. SONUÇ

Bu çalışmada bir telekomünikasyon şirketinin insan kaynakları veri setine uygulanan ve çalışan kaybı tahmin problemi için kullanılan sekiz sınıflandırma modeli hesaplanan metriklere göre değerlendirilmiştir. Her bir modelin doğruluk, çapraz doğrulama skoru, kesinlik, duyarlılık, $f_1$ölçütü ve EAKA metrik değerleri hesaplanmıştır. Bu modeller genellikle müşteri kaybını ölçen modellerin tahmin problemlerinde kullanılırken bu çalışmada çalışan kaybını tahmin etmek ve en iyi tahmini yapan modeli bulmak amacıyla kullanılmıştır. Çalışmayı diğer çalışmalardan farklı kılan en önemli kısmı hedef kitlesidir. Sonuç olarak problem özelinde gelecekte yapılacak sınıflandırma çalışmaları için bu çalışmada uygulanan modeller değerlendirildiğinde en iyi metrik değerlerine ulaşılan yani en iyi sınıflandıran RO modeli önerilmektedir.

Ancak veri gizliliğinden dolayı alınamayan çalışanların maaşı, evinin bağlı olduğu lokasyona uzaklığı gibi verilerin temini ile aynı modeller tekrar kurularak değerlendirilebilir.

Ayrıca şirketlerde yazılım bilmeyen kullanıcıların rahatça bu tahminleme modellerini kullanabilmesi için bir arayüz tasarlanarak en yüksek doğruluğu veren RO algoritması ile tahminleme yapabilmesi mümkün olabilir. Ek olarak çalışmada kullanılmayan diğer sınıflandırma modelleri ile çalışmanın kapsamı genişletilebilir.

KAYNAKLAR

Adak, M. F., & Yurtay, N. (2013). Gini algoritmasını kullanarak karar ağacı oluşturmayı sağlayan bir yazılımın geliştirilmesi. *Bilişim Teknolojileri Dergisi*, *6*(3), 1-6.

Ajit, P. (2016). Prediction of employee turnover in organizations using machine learning algorithms. *algorithms*, *4*(5), C5.

Alamsyah, A., & Salma, N. (2018, August). A comparative study of employee churn prediction model. In *2018 4th International Conference on Science and Technology (ICST)* (pp. 1-4). IEEE.

Albayrak, A. S. (2009). Türkiye'de Yerli Ve Yabancı Ticaret Bankalarının Finansal Etkinliğe Göre Sınıflandırılması: Karar Ağacı, Lojistik Regresyon Ve Diskriminant Analizi Modellerinin Bir Karşılaştırması. *Süleyman Demirel Üniversitesi İktisadi Ve İdari Bilimler Fakültesi Dergisi*, *14*(2), 113-139.

Alpaydin, E. & Bach, F. (2014), Introduction to Machine Learning, MIT Press, Cambridge, United States.

Bayru, P. (2007). Elektronik Basında Tüketici Tercihleri Analizi: Yapay Sinir Ağları İle Lojit Modelin Performans Değerlendirilmesi. *Unpublished Doctoral Dissertation). İstanbul University, Institute of Social Sciences, İstanbul*.

Bhatia, N. (2010). Survey of nearest neighbor techniques. *arXiv preprint arXiv:1007.0085*.

Bishop, D. V. (2006). What causes specific language impairment in children?. Current directions in psychological science, 15(5), 217-221.

Blum, A., & Mitchell, T. (1998, July). Combining labeled and unlabeled data with co-training. In PAİKeedings of the eleventh annual conference on Computational learning theory (pp. 92-100).

Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5-32.

Breiman, L., Friedman, J., Olshen, R., & Stone, C. (1984). Cart. *Classification and Regression Trees*.

Catani F, Lagomarsino D, Segoni S, Tofani V (2013) Landslide susceptibility estimation by random forests technique: sensitivity and scaling issues. Nat Hazards Earth Syst Sci 13:2815–2831.

Chapelle, O., Chi, M., & Zien, A. (2006, June). A continuation method for semi-supervised SVMs. In PAİKeedings of the 23rd international conference on Machine learning (pp. 185-192).

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).

Chen, Y. T., Tu, Y. M., & Jeng, B. (2011). A machine learning approach to policy optimization in system dynamics models. *Systems Research and Behavioral Science*, *28*(4), 369-390.

Chin-Yuan, F., Sheau-Pyng, J., & Ming-Fong, L. (2012, December). Using system thinking to investigate co-opetition analysis for manufacturers in the cloud industry. In *2012 IEEE International Conference on Industrial Engineering and Engineering Management* (pp. 2048-2052). IEEE.

Choong, A. C. H., & Lee, N. K. (2017, November). Evaluation of convolutionary neural networks modeling of DNA sequences using ordinal versus one-hot encoding method. In *2017 International Conference on Computer and Drone Applications (IConDA)* (pp. 60-65). IEEE.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine learning*, *20*(3), 273-297.

Cox, D. R. (1958). The regression analysis of binary sequences. *Journal of the Royal Statistical Society: Series B (Methodological)*, *20*(2), 215-232.

Crevier, D. (1993). Expert systems as design aids for artificial vision systems: a survey. Intelligent Robots and Computer Vision XII: Algorithms and Techniques, 2055, 84-96.

Da Silva, I. N., Spatti, D. H., Flauzino, R. A., Liboni, L. H. B., & dos Reis Alves, S. F. (2017). Artificial neural network architectures and training processes. In *Artificial neural networks* (pp. 21-28). Springer, Cham.

Dalkilic, M. M. Alexander N. Zimmerman, Claudia C. Johnson, Nicholas W. Bussberg &. (2020) Stability and decline in deep-sea coral biodiversity, Gulf of Mexico and US West Atlantic. Springer, 39, 345–359.

Doğan, V., Yüzer, E., Kılıç, V., & Şen, M. (2021). Non-enzymatic colorimetric detection of hydrogen peroxide using a μPAD coupled with a machine learning-based smartphone app. *Analyst*, *146*(23), 7336-7344.

Dong, Y., Zhang, Y., Yue, J., & Hu, Z. (2016). Comparison of random forest, random ferns and support vector machine for eye state classification. *Multimedia Tools and Applications*, *75*(19), 11763-11783.

Dudani, S. A. (1976). The distance-weighted k-nearest-neighbor rule. IEEE Transactions on Systems, Man, and Cybernetics, (4), 325-327.

Ecemiş, A., Dokuz, A. Ş., & Çelik, M. Çeşitli Veri Kümeleri Üzerinde Veri Madenciliği Algoritmalarının Performansının Değerlendirilmesi.

Fang, Y., Qiu, Y., Liu, L., & Huang, C. (2018, March). Detecting webshell based on random forest with fasttext. In *Proceedings of the 2018 International Conference on Computing and Artificial Intelligence* (pp. 52-56).

Freund, Y., & Schapire, R. E. (1996, July). Experiments with a new boosting algorithm. In *icml* (Vol. 96, pp. 148-156).

Friedman, C. P. (2000). The marvelous medical education machine or how medical education can beunstuck'in time. *Medical Teacher*, *22*(5), 496-502.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, 1189-1232.

Grove, E. (2006). Sea Power in the Asia-Pacific Region. In *The Evolving Maritime Balance Of Power In The Asia-Pacific: Maritime Doctrines and Nuclear Weapons at Sea* (pp. 17-33).

Guner, H., Ozgur, E., Kokturk, G., Celik, M., Esen, E., Topal, A. E., ... & Dana, A. (2017). A smartphone based surface plasmon resonance imaging (SPRi) platform for on-site biodetection. Sensors and Actuators B: Chemical, 239, 571-577.

Han, J. W., Breckon, T. P., Randell, D. A., & Landini, G. (2012). The application of support vector machine classification to detect cell nuclei for automated microscopy. *Machine Vision and Applications*, *23*(1), 15-24.

Hsu, C. N., Chung, H. H., & Huang, H. S. (2004). Mining skewed and sparse transaction data for personalized shopping recommendation. *Machine Learning*, *57*(1), 35-59.

Ilhan, R., & Güdar, B. (2021). Yapay Sinir Ağları Kullanarak Kan Testi Sonuçlarının Sınıflandırılması ve Kullanıcı Ara Yüzünün Geliştirmesi. *Avrupa Bilim ve Teknoloji Dergisi*, (29), 1-5.

Kavzoğlu, T., & Çölkesen, İ. (2010). Destek vektör makineleri ile uydu görüntülerinin sınıflandırılmasında kernel fonksiyonlarının etkilerinin incelenmesi. *Harita Dergisi*, *144*(7), 73-82.

Kecman, V. (2001). *Learning and soft computing: support vector machines, neural networks, and fuzzy logic models*. MIT press.

Keleş, M. B., Keleş, A., & Keleş, A. (2020). Yapay zekâ teknolojisi ile uçuş fiyatı tahmin modeli geliştirme. *Turkish Studies*, *15*(4), 511-520.

Kohonen, T. (1990, June). Improved versions of learning vector quantization. In *1990 ijcnn international joint conference on Neural networks* (pp. 545-550). IEEE.

Kumaş, E. (2021). Türkçe Twitter Verilerinden Duygu Analizi Yapılırken Sınıflandırıcıların Karşılaştırılması. Eskişehir Türk Dünyası Uygulama Ve Araştırma Merkezi Bilişim Dergisi, 2(2), 1-5.

Kuyucu, Y. E. (2012). Lojistik regresyon analizi (LRA), yapay sinir ağları (YSA) ve sınıflandırma ve regresyon ağaçları (C&RT) yöntemlerinin karşılaştırılması ve tıp alanında bir uygulama (Master's thesis, Gaziosmanpaşa Üniversitesi, Sağlık Bilimleri Enstitüsü).

Liaw A, Wiener M, (2002) Classification and regression by random forest. R News 2: 18–22.

Lin, S. W., Ying, K. C., Chen, S. C., & Lee, Z. J. (2008). Particle swarm optimization for parameter determination and feature selection of support vector machines. *Expert systems with applications*, *35*(4), 1817-1824.

Liu, H., & Zhang, S. (2012). Noisy data elimination using mutual k-nearest neighbor for classification mining. Journal of Systems and Software, 85(5), 1067-1074.

McCarthy, J., & Feigenbaum, E. A. (1990). In memoriam: Arthur samuel: Pioneer in machine learning. AI Magazine, 11(3), 10-10.

Mohri, M., & Medina, A. M. (2014, January). Learning theory and algorithms for revenue optimization in second price auctions with reserve. In *International conference on machine learning* (pp. 262-270). PMLR.

Muja, M., & Lowe, D. G. (2009). Fast approximate nearest neighbors with automatic algorithm configuration. *VISAPP (1)*, *2*(331-340), 2.

Muller, K. R., Mika, S., Ratsch, G., Tsuda, K., & Scholkopf, B. (2001). An introduction to kernel-based learning algorithms. *IEEE transactions on neural networks*, *12*(2), 181-201.

Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*. MIT press.

Obuchowski, N. A. (2005). ROC analysis. *American Journal of Roentgenology*, *184*(2), 364-372.

Obuchowski, N. A., Lieber, M. L., & Wians Jr, F. H. (2004). ROC curves in clinical chemistry: uses, misuses, and possible solutions. *Clinical chemistry*, *50*(7), 1118-1125.

Osuna, E., Freund, R., & Girosit, F. (1997, June). Training support vector machines: an application to face detection. In *Proceedings of IEEE computer society conference on computer vision and pattern recognition* (pp. 130-136). IEEE.

Özdemir, A.K., Tolun, S. & Demirci, E. (2011). Endeks getirisi yönünün ikili sınıflandırma yöntemiyle tahmin edilmesi: İMKB 100 endeksi örneği. *Niğde Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, *4*(2), 45-59.

Özkan, K. (2012). Sınıflandırma ve regresyon ağacı tekniği (SRAT) ile ekolojik verinin modellenmesi. *Süleyman Demirel Üniversitesi Orman Fakültesi Dergisi*, *13*(1), 1-4.

Öztemel, E. (2003). Yapay sinir ağlari. *PapatyaYayincilik, Istanbul*.

Probst, P., Boulesteix, A. L., & Bischl, B. (2019). Tunability: importance of hyperparameters of machine learning algorithms. *The Journal of Machine Learning Research*, *20*(1), 1934-1965.

Raschka, S. (2015). *Python machine learning*. Packt publishing ltd.

Ribes, E., Touahri, K., & Perthame, B. (2017). Employee turnover prediction and retention policies design: a case study. *arXiv preprint arXiv:1707.01377*.

Sarker, I. H. (2021). Machine learning: Algorithms, real-world applications and research directions. SN Computer Science, 2(3), 1-21.

Shah, P., Mishra, D. K., Shanmugam, M. P., Doshi, B., Jayaraj, H., & Ramanjulu, R. (2020). Validation of Deep Convolutional Neural Network-based algorithm for detection of diabetic retinopathy–Artificial intelligence versus clinician for screening. *Indian Journal of Ophthalmology*, *68*(2), 398.

Sharma, A. K., & Arikawa, S. (1996). *Algorithmic Learning Theory: 7th International Workshop, ALT'96, Sydney, Australia, October 23-25, 1996. Proceedings* (Vol. 1160). Springer Science & Business Media.

Sisodia, D. S., Vishwakarma, S., & Pujahari, A. (2017, November). Evaluation of machine learning models for employee churn prediction. In *2017 international conference on inventive computing and informatics (icici)* (pp. 1016-1020). IEEE.

Şenol, H., Erşan, M., & Görgün, E. (2020). Optimization of temperature and pretreatments for methane yield of hazelnut shells using the response surface methodology. Fuel, 271, 117585.

Tanyıldızı, E., & Demirtaş, F. (2019, November). Hiper Parametre Optimizasyonu Hyper Parameter Optimization. In *2019 1st International Informatics and Software Engineering Conference (UBMYK)* (pp. 1-5). IEEE.

Tekin, E., Sarısoy, F., & Ciğerci, A. E. (2018). Shuttle Run Analysis With Image Processing. *Artificial Intelligence Studies*, *1*(2), 1-12.

Tolun, S. (2008). *Destek vektör makineleri: Banka başarısızlığının tahmini üzerine bir uygulama*. İktisadî Araştırmalar Vakfı.

Turing, A. M. (1950). Mind. *Mind*, *59*(236), 433-460.

Uğur, A., & Kınacı, A. C. (2006). Yapay zeka teknikleri ve yapay sinir ağları kullanılarak web sayfalarının sınıflandırılması. XI. Türkiye'de İnternet Konferansı (inet-tr'06), Ankara, 1(4).

Wade, C. (2020). *Hands-On Gradient Boosting with XGBoost and scikit-learn: Perform accessible machine learning and extreme gradient boosting with Python*. Packt Publishing Ltd.

Yang, X. S. (2019). *Introduction to algorithms for data mining and machine learning*. Academic press.

Zhuang, F., Cheng, X., Luo, P., Pan, S. J., & He, Q. (2015, June). Supervised representation learning: Transfer learning with deep autoencoders. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

ÖZGEÇMİŞ

Adı Soyadı	: Büşra UZAK
Doğum Tarihi	:
Yabancı Dil	: İngilizce
Eğitim Durumu	
Lise	: Bursa Nilüfer Fatih Lisesi
Lisans	: Yalova Üniversitesi-Endüstri Mühendisliği (Ana dal)
	Yalova Üniversitesi-Bilgisayar Mühendisliği (Çift Ana dal)
Yüksek Lisans	: Uludağ Üniversitesi-Endüstri Mühendisliği (Tezli)
Çalıştığı Kurum	: Turkcell Global Bilgi- İK Veri Analitiği Uzmanı (2020-Devam)
İletişim (e-posta)	: