



**HİSSE SENEDİ FİYAT TAHMİNİNDE İSTATİKSEL
VE YAPAY ZEKÂ MODELLERİNİN ETKİSİ:
GERÇEK ZAMANLI VERİ ÜZERİNE BİR
İNCELEME**

Hüseyin KARAKAŞ

Yüksek Lisans Tezi

Endüstri Mühendisliği

Ana Bilim Dalı

Dr. Öğr. Üyesi Çağatay TEKE

2025

(Her Hakkı Saklıdır)

T.C.
BAYBURT ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
ENDÜSTRİ MÜHENDİSLİĞİ ANA BİLİM DALI
ENDÜSTRİ MÜHENDİSLİĞİ TEZLİ YÜKSEK LİSANS PROGRAMI

HİSSE SENEDİ FİYAT TAHMİNİNDE İSTATİKSEL VE YAPAY ZEKÂ
MODELLERİNİN ETKİSİ: GERÇEK ZAMANLI VERİ ÜZERİNE BİR
İNCELEME

YÜKSEK LİSANS TEZİ

Hüseyin KARAKAŞ

Danışman: Dr. Öğr. Üyesi Çağatay TEKE

BAYBURT- 2025

KABUL VE ONAY

Dr. Öğr. Üyesi Çağatay TEKE danışmanlığında, Hüseyin KARAKAŞ tarafından hazırlanan “Hisse Senedi Fiyat Tahmininde İstatiksel ve Yapay Zekâ Modellerinin Etkisi: Gerçek Zamanlı Veri Üzerine Bir İnceleme” başlıklı bu çalışma, 25.06.2025 tarihinde yapılan savunma sınavı sonucunda oy birliği ile başarılı bulunarak jürimiz tarafından Yüksek Lisans Tezi olarak kabul edilmiştir.

Başkan : Doç. Dr. Akın ÖZDEMİR

Üye : Doç. Dr. Didem GÜLERYÜZ ÖZDEN

Üye : Dr. Öğr. Üyesi Çağatay TEKE

Bu tezin kabulü Lisansüstü Eğitim Enstitüsü Yönetim Kurulu’nun/...../20... tarih ve 20...../.....-..... sayılı kararıyla onaylanmıştır.

Doç. Dr. Bünyamin ALİM
Enstitü Müdürü

BEYANNAME

Bayburt Üniversitesi Lisansüstü Eğitim Enstitüsü tez yazım kılavuzuna göre Dr. Öğr. Üyesi Çağatay TEKE danışmanlığında hazırlamış olduğum “Hisse Senedi Fiyat Tahmininde İstatiksel ve Yapay Zekâ Modellerinin Etkisi: Gerçek Zamanlı Veri Üzerine Bir İnceleme” başlıklı yüksek lisans tezimin bilimsel etik değerlere ve kurallara uygun, özgün bir çalışma olduğunu, aksinin tespit edilmesi halinde her türlü yasal yaptırımını kabul edeceğimi beyan ederim.

25.06.2025

Hüseyin KARAKAŞ



ÖN SÖZ

Bu çalışmanın her aşamasında bana yol gösteren ve bilgi ile deneyimlerini cömertçe paylaşan Sayın Dr. Öğr. Üyesi Çağatay TEKE'ye; bu çalışma boyunca sadece danışman değil, aynı zamanda bir mentor olarak sabırlı yaklaşımı ve bilimsel tutumuyla bu tezin kalitesini artırarak akademik gelişimime büyük katkılarda bulunduğu için teşekkür ederim.

Hüseyin KARAKAŞ



ÖZET

YÜKSEK LİSANS TEZİ

HİSSE SENEDİ FİYAT TAHMİNİNDE İSTATİKSEL VE YAPAY ZEKÂ MODELLERİNİN ETKİSİ: GERÇEK ZAMANLI VERİ ÜZERİNE BİR İNCELEME

Hüseyin KARAKAŞ

Bayburt Üniversitesi

Lisansüstü Eğitim Enstitüsü

Endüstri Mühendisliği Ana Bilim Dalı

Tez Danışmanı: Dr. Öğr. Üyesi Çağatay TEKE

Bayburt-2025, Sayfa: 45

Finans piyasalarındaki belirsizlik, yüksek volatilité ve karmaşıklık, fiyat tahminini yatırımcılar için oldukça zor hale getirmektedir. Geleneksel yöntemlerin sınırlılıkları, yatırımcıları daha ileri teknolojilere yönlendirmekte, bu bağlamda makine öğrenmesi ve derin öğrenme yöntemleri önemli araçlar olarak öne çıkmaktadır. Bu tez çalışmasında, makine öğrenmesi ve derin öğrenme yöntemleri kullanılarak farklı tahmin modelleri oluşturulmuş ve hisse senedi fiyatlarının analizi yapılmıştır. Modellerin eğitilmesi ve test edilmesi aşamasında, gerçek hisse senedi verileri kullanılarak kapsamlı bir analiz gerçekleştirilmiştir. Modellerin performansları doğruluk oranları, performans değerlendirme ölçütleri ve işlem süreleri gibi kriterlerle ayrıntılı olarak karşılaştırılmıştır. Elde edilen sonuçlar, LR modelinin hisse senedi fiyat tahmininde daha başarılı olduğunu ortaya koymuştur. Çalışmanın bulguları, finansal piyasalarda daha doğru yatırım kararları alınabilmesi için gelecekteki çalışmalara da ışık tutmaktadır.

Anahtar Kelimeler: Hisse senedi fiyat tahmini, finansal analiz, makine öğrenmesi, derin öğrenme.

ABSTRACT

M. SC. THESIS

THE IMPACT OF STATISTICAL AND ARTIFICIAL INTELLIGENCE MODELS IN STOCK PRICE FORECASTING: A STUDY ON REAL-TIME DATA

Hüseyin KARAKAŞ

Bayburt University

Institute of Graduate Studies

Department of Industrial Engineering

Thesis Advisor: Assist. Prof. Dr. Çağatay TEKE

Bayburt-2025, Pages: 45

Uncertainty, high volatility and complexity in financial markets make price prediction very difficult for investors. The limitations of traditional methods lead investors to more advanced technologies, and in this context, machine learning and deep learning methods stand out as important tools. In this thesis, different forecasting models are created using machine learning and deep learning methods and stock prices are analyzed. During the training and testing phase of the models, a comprehensive analysis was carried out using real stock data. The performance of the models was compared in detail with criteria such as accuracy rates, error metrics and processing times. The results revealed that the LR model is more successful in stock price forecasting. The findings of the study shed light on future studies to make more accurate investment decisions in financial markets.

Keywords: Stock price prediction, financial analysis, machine learning, deep learning.

İÇİNDEKİLER

KABUL VE ONAY.....	II
BEYANNAME	III
ÖN SÖZ	IV
ÖZET	V
ABSTRACT	VI
İÇİNDEKİLER.....	VII
TABLOLAR.....	IX
ŞEKİLLER.....	X
KISALTMALAR.....	XI
GİRİŞ.....	1
Araştırmanın Konusu ve Problemi	4
Araştırmanın Amacı	4
Araştırmanın Önemi ve Gerekçesi	4
Araştırmanın Sınırlılıkları	5
Araştırmanın Varsayımları	5
BİRİNCİ BÖLÜM.....	7
1. LİTERATÜR TARAMASI.....	7
İKİNCİ BÖLÜM	14
2. MATERYAL VE YÖNTEM	14
2.1. VERİ ÖN İŞLEME.....	14
2.1.1. Veri Kalitesini Etkileyen Faktörler	15
2.1.2. Veri Ön İşleme Adımları	15
2.2. PERFORMANS DEĞERLENDİRME ÖLÇÜTLERİ	17
2.2.1. Belirlilik Katsayısı	17
2.2.2. Ortalama Mutlak Hata	17
2.2.3. Hata Kareleri Ortalaması	17
2.2.4. Hata Kareleri Ortalaması Karekökü	18
2.3. MAKİNE ÖĞRENMESİ.....	18
2.3.1. Makine Öğrenmesi Teknikleri.....	19
2.3.2. Doğrusal Regresyon	19
2.3.3. Rastgele Orman	20
2.4. DERİN ÖĞRENME	20
2.4.1. Uzun Kısa Süreli Bellek	21
2.4.2. Yapay Sinir Ağları.....	21
2.5. PYHTON	22
2.5.1. Google Colab	22
2.5.2. Pandas.....	24
2.5.3. Numpy	24
2.5.4. Matplotlib	25
ÜÇÜNCÜ BÖLÜM.....	26
3. UYGULAMA	26
3.1. VERİ SETİNİ HAZIRLAMA	26
3.1.1. Gerekli Kütüphanelerin Yüklenmesi	26

3.1.2. Veri Setinin Eklenmesi	26
3.1.3. Veri Ön İşleme	27
3.1.4. Özellik ve Hedef Değişken Tanımlanması	27
3.1.5. Veri Setinin Eğitim ve Test Kümelerine Ayrılma	28
3.1.6. Verilerin Ölçeklendirilmesi	28
3.2. MODELLERİN OLUŞTURULMASI	29
3.2.1. Lineer Regresyon Modeli	29
3.2.2. Rastgele Orman Modeli	30
3.2.3. LSTM Modeli	31
3.2.4. Yapay Sinir Ağı Modeli	33
3.3. MODELLERİN DEĞERLENDİRİLMESİ	35
3.4. TAHMİN YAPILMASI	38
DÖRDÜNCÜ BÖLÜM.....	39
4. BULGULAR	39
SONUÇ, TARTIŞMA VE ÖNERİLER.....	40
KAYNAKÇA.....	42
ÖZ GEÇMİŞ	45

TABLÖLAR

Tablo 1: LR Modeli Performans Değerlendirme Ölçütleri Sonuçları	29
Tablo 2: RF Modeli Performans Değerlendirme Ölçütleri Sonuçları	30
Tablo 3: LSTM Modeli Performans Değerlendirme Ölçütleri Sonuçları.....	32
Tablo 4: YSA Modeli Performans Değerlendirme Ölçütleri Sonuçları	34
Tablo 5: Modellere Göre Performans Değerlendirme Ölçütleri Sonuçları	40



ŞEKİLLER

Şekil 1: Google Drive ile Colab Açılması	23
Şekil 2: Colab Kod Alanı	23
Şekil 3: Colab Açıklama Alanı	24
Şekil 4: Colab Açıklama Alanı Ekran Çıktısı	24
Şekil 5: Python Kütüphanelerinin Yüklenmesi.....	26
Şekil 6: Gerçek Zamanlı Veri Setinin Yüklenmesi.....	27
Şekil 7: Hisse Senedi Kapanış Fiyatı ile 200, 100 ve 50 Günlük Hareketli Ortalamaları	27
Şekil 8: Özellik ve Hedef Değişken Tanımlaması	28
Şekil 9: Verilerin, Eğitim ve Test Verileri Olarak Ayrılması	28
Şekil 10: LR Modeli ile Tahmin Edilen Değer ve Gerçek Değer	29
Şekil 11: LR Modeli İçin Performans Değerlendirme Ölçütleri Sonuçları	30
Şekil 12: RF Modeli ile Tahmin Edilen Değer ve Gerçek Değer	31
Şekil 13: RF Modeli İçin Performans Değerlendirme Ölçütleri Sonuçları.....	31
Şekil 14: LSTM Model Kodu	32
Şekil 15: LSTM Modeli ile Tahmin Edilen Değer ve Gerçek Değer	33
Şekil 16: LSTM Modeli İçin Performans Değerlendirme Ölçütleri Sonuçları.....	33
Şekil 17: YSA Model Kodu	34
Şekil 18: YSA Modeli ile Tahmin Edilen Değer ve Gerçek Değer	35
Şekil 19: YSA Modeli İçin Performans Değerlendirme Ölçütleri Sonuçları.....	35
Şekil 20: Modellere Göre MAE Performans Değerlendirme Ölçütleri Karşılaştırması	36
Şekil 21: Modellere Göre MSE Performans Değerlendirme Ölçütleri Karşılaştırması.....	36
Şekil 22: Modellere Göre RMSE Performans Değerlendirme Ölçütleri Karşılaştırması	37
Şekil 23: Modellere Göre R^2 Performans Değerlendirme Ölçütleri Karşılaştırması.....	37
Şekil 24: Tahmin Kodu ve Sonucu	38

KISALTMALAR

SVR	: Destek Vektör Regresyonu
KNN	: K-En Yakın Komşular
XGboost	: eXtreme Gradient Boosting
MLP	: Çok Katmanlı Algılayıcılar
CNN	: Convolutional Neural Network
RNN	: Tekrarlayan Sinir Ağları
LSTM	: Uzun Kısa – Süreli Bellek
MAE	: Ortalama Mutlak Hata
MAPE	: Ortalama Mutlak Yüzde Hata
MSE	: Ortalama Kare Hatası
RMSE	: Karekök Ortalama Hata
R²	: Belirlilik Katsayısı
MBE	: Ortalama Önyargı Hatası
RFR	: Rastgele Orman Regresyonu
YSA	: Yapay Sinir Ağı
RF	: Rastgele Orman
ÇKA	: Çok Katmanlı Algılayıcılar
SVM	: Destek Vektör Makineleri
LR	: Doğrusal Regresyon
BIST 100	: Borsa İstanbul Ulusal-100 Endeksi
S & P 500	: Amerikan Borsa Endeksi
FTSE	: Londra Borsa Endeksi
THYAO	: Türk Hava Yolları
NYSE	: New York Borası
Auto ARIMA	: Oto-Regresif Entegre Hareketli Ortalamalar
DAX	: Almanya Birleşik Borsa Endeksi
BISTBANK	: Borsa İstanbul Bankacılık Endeksi
GOLDINDEX	: BIST Altın Endeksi
BISTSINAI	: Borsa İstanbul Sanayi Endeksi
BISTMALI	: İstanbul Mali Endeksi

GİRİŞ

Modern finans dünyasının en önemli hedeflerinden biri hisse senedi fiyatlarını tahmin ederek akıllı yatırımlar yapmak ve bunun sonucunda para kazanmaktır (Demirel, 2019). Bu nedenle hisse senetleri ve fonlar uzun zamandır borsa yatırımcıları, araştırmacılar ve finans ile ilgilenen bilim insanları için çok önemli ve gizemli bir konu olmuştur. Yatırımlarını doğru değerlendirmeyi bilenler borsa sayesinde büyük karlar elde edebilirken, yanlış değerlendirme nedeniyle de büyük kayıplara uğrayabilmektedir.

Türkiye borsası da dahil olmak üzere gelişmekte olan piyasalarda sürdürülebilir büyüme çoğu zaman mümkün olmamaktadır (Altunbaş, 2021). Gelişmiş ülke borsalarındaki işlem hacimleriyle karşılaştırıldığında Borsa İstanbul'daki işlem hacimlerinin daha düşük olması, Borsa İstanbul'un ulusal ve küresel olaylara karşı daha duyarlı hale getirerek daha büyük iniş ve çıkışlar yaşamasına neden olmaktadır. Tüm bu şartlar dikkate alındığında Borsa İstanbul'a işlem yapan yatırımcılar, oluşabilecek bu ani hareketlere daha az maruz kalarak yatırımlarını korumaya ve piyasada yaratılan fırsatları değerlendirmeye çalışmaktadırlar. Bu sebeple yatırımcıların borsada meydana gelebilecek olası iniş ve çıkışları öngörmesi oldukça önemlidir.

Yatırımcılar yatırım kararları alırken hisse senedi için iki çeşit analizden yararlanabilirler. Bunlardan ilki, hisselerin gerçek değerini, ülkedeki siyasi durumu, sanayi ve ekonominin performansını kapsayan temel analizdir. İkincisi ise bir hisse senedinin geçmiş değerleri ve işlem hacimleri gibi piyasa işlemlerine dayalı temel analizdir. Teknik analizin amacı gerçek verileri tahmin etmek değildir. Bunun yerine hisse senedi grafiklerini kullanarak gelecekteki olası kalıpları ve eğilimleri tahmin etmektir. Çünkü hisse senetleri siyasi değişimler, yatırımcı hareketleri, genel ekonomik koşullar gibi pek çok faktörden oldukça etkilenmesidir (Arslankaya ve Toprak, 2021).

Yatırım için alınan kararlar, şirketlerin kar elde etmek amacıyla kullanabilecekleri kaynaklar kullanılarak hangi yatırımların yapılabileceğini belirleyen kararlardır. Bu kararlar firmaların pazar paylarına göre değişmekte, firmanın değerini göstermekte ve ekonomik kalkınmayı etkilemektedir. Hisse senedi fiyatlaması geçmiş fiyatları kullanarak gelecekteki olası fiyatların belirlenmesidir. Bu, zaman serisi tahmini ile alakalı bir konudur ve regresyon temelli bir problemdir. Gelecek fiyatlar belirsiz olduğundan birçok araştırmacının merakını çekmiştir. Değerleme, sabit fiyat hareketleri, doğrusal olmama ve gürültülü veriler gibi nedenlerden dolayı karmaşık bir süreç olmuştur. Fiyatların doğru fiyatlanması ve etkin bir ticaret stratejisiyle birleştirilmesi şirketlerin elde ettiği kârın artmasında ve yatırımcıların sağlıklı yatırım kararları almasına yardımcı olmada etkili olacaktır (Ersayan, 2022).

Matematiksel ve istatistiksel yöntemleri kullanarak tahminde bulunmaya yönelik önceki girişimler, özellikle bilgisayar teknolojisinin ileri seviyelere gelmesiyle birlikte daha başarılı sonuçlar üretmiştir. Son yıllarda geliştirilen yapay zekâ ve algoritmik öğrenme kavramının bu sürece çok büyük katkısı olmuştur (Demirel, 2019). Son yıllarda finansal tahminde makine öğrenmesi yöntemlerinin ve derin öğrenme algoritmalarının kullanımının artması ve elde edilen olumlu sonuçlar araştırmacıların bu alana olan ilgisini artırmış; ancak fiyatların piyasalardaki davranışlarını tam olarak tahmin edebilecek bir yöntem bulunamadığı için bu alandaki araştırmalar aktif olarak yapılmaya devam etmiştir.

Makine öğrenimi, bilgisayarların açıkça programlanmadan verilerden öğrenmelerini ve performanslarını zaman içinde geliştirmelerini sağlayan yapay zekanın bir alt alanıdır (Shaveta, 2023). Makine öğreniminin temel amacı, verilerdeki kalıpları veya düzenlilikleri tanımlayabilen algoritmalar geliştirmek ve daha sonra yeni veriler sunulduğunda tahminler veya kararlar vermek için bu kalıpları kullanmaktır (Scott vd., 2019).

Makine öğrenimi yöntemleri, karmaşık ve büyük veri kümelerini etkili bir şekilde ele alma yetenekleri nedeniyle sağlık, finans ve astronomi gibi çeşitli alanlarda giderek daha fazla kullanılmaktadır (Ali ve Mashwani, 2023). Makine öğrenimi finans alanında risk yönetimi, dolandırıcılık tespiti, kredi skorlama, portföy optimizasyonu ve benzeri amaçlar için kullanılmaktadır. Makine öğrenimi algoritmalarının önemli bir avantajı uyarlanabilirliktir. Veri hacmi arttıkça, algoritmalar insan müdahalesi olmadan öğrenebilir ve gelişebilir, böylece görevlerinde daha verimli hale gelir (Jha vd., 2021). Bununla birlikte, bu tekniklerin uygulanması, kaliteli veri ve önemli hesaplama kaynaklarına duyulan ihtiyaç gibi zorlukları da beraberinde getirmektedir (Jin, 2020).

Derin öğrenme, karmaşık verileri işleme ve geleneksel makine öğrenimi yöntemlerinin zorlandığı görevleri yerine getirme yeteneği nedeniyle önem kazanan bir makine öğrenimi alt alanıdır. İnsan beyninin işleyişini taklit ederek büyük miktarda veriden öğrenmelerini sağlayan yapay sinir ağlarının kullanımını içerir. Derin öğrenmedeki “derin”, bu sinir ağlarındaki katman sayısını ifade eder. Genellikle manuel özellik çıkarımı gerektiren geleneksel makine öğreniminin aksine, derin öğrenme modelleri veri özelliklerini otomatik olarak çıkarabilir ve bunlardan öğrenerek verimliliği ve performansı artırabilir (Cayir vd., 2018).

Derin öğrenmenin etkinliği özellikle bilgisayarla görme, doğal dil işleme ve konuşma tanıma gibi yüksek boyutlu veri işleme ve analizi gerektiren alanlarda belirgin hale gelmiştir. Örneğin Evrişimsel Sinir Ağları (CNN'ler), görüntü ve nesne tanıma görevlerini önemli ölçüde geliştirerek bilgisayarla görmede devrim yaratmıştır. Görüntüleri doğrudan piksel

değerlerinden işleyerek el yapımı özellik çıkarma ihtiyacını ortadan kaldırılabilmektedirler (Iglesias vd., 2021).

Derin öğrenmenin yükselişi, özellikle büyük veri kümelerinin verimli bir şekilde işlenmesine olanak tanıyan GPU'ların ortaya çıkmasıyla birlikte hesaplama gücündeki gelişmelerle kolaylaştırılmıştır. Derin öğrenme araştırmaları ilerledikçe, bilim insanları ve mühendisler yeni modeller ve uygulamalar keşfetmeye devam etmekte, böylece çeşitli gerçek dünya sorunlarını ele almaktadırlar (Lemley vd., 2017).

Bu çalışmada özgün bir yaklaşım olarak iki aşamalı bir model yapısı önerilmiştir. Birinci aşamada, Python programlama dili kullanılarak çevrim içi veri çekme süreci gerçekleştirilmiştir. Kullanıcı tarafından girilen hisse senedi sembolleri (ticker) aracılığıyla, Yahoo Finance üzerinden seçilen herhangi bir hisse senedine ait güncel veriler otomatik olarak elde edilmektedir. Bu yönüyle çalışma, herhangi bir statik veri kümesine bağlı kalmaksızın, gerçek zamanlı ve esnek bir tahmin altyapısı sunmaktadır ve bu özelliği ile literatürdeki birçok geleneksel çalışmadan ayrılmaktadır. İkinci aşamada ise, makine öğrenmesi algoritmaları ile klasik istatistiksel tahmin yöntemleri kullanılarak performans karşılaştırmalı tahmin modelleri geliştirilmiştir. Böylece, her iki aşamanın bir araya getirilmesiyle hem uygulamaya odaklı hem de metodolojik olarak güçlü bir yapı ortaya konmuştur. Çalışmanın en önemli katkılarından biri, gerçek zamanlı veri çekme ve analiz imkânı sunarak, hisse senedi tahmini konusunda dinamik ve güncel bir çözüm önermesidir.

Bu çalışmanın amacı yatırımcılar için alacakları yatırım kararlarına yardımcı olmak için makine öğrenmesi ve derin öğrenme yöntemleri kullanılarak hisse senedi fiyat tahmini yapılmasıdır. Çalışmada hisse senedi fiyatlarını gösteren veri seti Python yazılım dili kullanılarak Yahoo Finance sitesinden çevrimiçi olarak alınacaktır. Alınacak veri setinde günlük hisse senedinin açılış, kapanış, gün içindeki en yüksek değer, gün içindeki en düşük değer ve ayarlanmış kapanış fiyatı (open, high, low, close ve adj close) bulunmaktadır. Çalışmanın birinci bölümünde konu ile ilgili literatür çalışması yapılmıştır. Yapılan tarama sonucunda konu hakkında değişik makine öğrenmesi ve derin öğrenme yöntemlerinin kullanıldığı, her hesaplamada farklı yöntemlerin tahmin için uygun olduğu görülmüştür. İkinci bölümde ise çalışmanın materyal ve yöntemi hakkında bilgi verilmiştir. Çalışmada kullanılan makine öğrenmesi ile derin öğrenme, Python dili ile kullanılacak kütüphaneler hakkında tanımlamalar yapılmış ve kullanılacak algoritmalara değinilmiştir. Üçüncü bölümde uygulama adımları şekiller ile desteklenerek çalışmanın adımları anlatılmıştır. Dördüncü bölümde çalışma hakkında bulunan bulgular ve tartışma konusu ele alınmıştır. Çalışmanın sonunda ise ortaya çıkan sonuçlar hakkında bilgi verilmiştir.

Araştırmanın Konusu ve Problemi

Bu çalışmada, hisse senedi piyasalarında finansal tahmin modellerinin etkinliği incelenmiştir. Makine öğrenmesi ve derin öğrenme yöntemleri kullanılarak, belirli bir hisse senedinin gelecekteki fiyat hareketlerini tahmin etme noktası üzerinde durulmuştur. Finansal piyasalar doğası gereği oldukça dinamik, karmaşık ve öngörülemezdir. Yatırımcılar ve finansal kurumlar, bilinçli kararlar alabilmek için gelecekteki fiyat hareketlerini tahmin etme ihtiyacı duyarlar. Ancak, bu tahminleri yaparken karşılaşılan temel problem, piyasalardaki yüksek volatilité, piyasa katılımcılarının rasyonel olmayan davranışları, makroekonomik faktörler ve beklenmedik olaylar gibi birçok belirsizliğin bulunmasıdır. Geleneksel finansal analiz yöntemleri bu karmaşıklığı tam olarak yakalamakta yetersiz kalabilmektedir. Bu bağlamda, makine öğrenimi ve derin öğrenme modellerinin bu belirsizlik ortamında hisse senedi fiyatlarını ne ölçüde doğru tahmin edebileceği ve bu modellerin geleneksel yöntemlere göre üstünlük sağlayıp sağlayamayacağı önemli bir araştırma problemi olarak öne çıkmaktadır.

Araştırmanın Amacı

Bu çalışmanın amacı, bir hisse senedinin kapanış fiyatlarını tahmin etmek için farklı makine öğrenimi ve derin öğrenme modellerinin (LR, RF, LSTM ve YSA) performansını karşılaştırmaktır. Hisse senedi verilerini kullanarak, regresyon tabanlı (LR ve RF) ve sinir ağı tabanlı (LSTM, YSA) modelleri geliştirmek, geliştirilen modellerin tahmin performanslarını MAE, MSE, RMSE ve R^2 gibi yaygın performans değerlendirme ölçütleri kullanarak nicel olarak değerlendirme, modellerin test seti üzerindeki gerçek değerlere olan yakınlıklarını görselleştirmek ve bu sayede modeller arasındaki performans farklarını somutlaştırmak, performans değerlendirme ölçütleri ve görsel analizler ışığında, hisse senedi fiyat tahmini için en uygun modelin hangisi olduğunu belirlemeye çalışmaktır.

Araştırmanın Önemi ve Gerekçesi

Bu çalışma, yatırımcıların ve portföy yöneticilerinin daha bilinçli ve veri odaklı yatırım kararları almalarına yardımcı olabilecek potansiyel tahmin modeli sunmaktır. Makine öğrenimi ve derin öğrenmenin finans alanındaki uygulamalarının artan önemini vurgulamaktadır. Bu çalışma, yapay zekâ tabanlı algoritmaların piyasa tahminlerindeki pratik kullanımına bir örnek teşkil etmektedir. Hisse senedi piyasalarında farklı makine öğrenimi ve derin öğrenme modellerinin karşılaştırmalı performansına dair literatüre katkı sağlamaktadır. Özellikle Türk piyasası gibi gelişmekte olan piyasalar için bu tür çalışmaların sayısı daha fazla önem kazanmaktadır.

Finansal piyasalar sürekli evrim geçirmekte ve bu evrimle birlikte yeni tahmin yöntemlerine olan ihtiyaç da artmaktadır. Geleneksel istatistiksel yöntemler, piyasalardaki doğrusal olmayan ilişkileri ve karmaşık örüntüleri yakalamakta zorlanabilir. Makine öğrenimi ve derin öğrenme modelleri ise, büyük ve karmaşık veri setlerinden öğrenme kapasiteleri sayesinde bu tür karmaşık ilişkileri tespit etme potansiyeline sahiptirler. Bu çalışma, somut bir hisse senedi çalışması üzerinden ortaya koyarak, geleneksel yaklaşımların ötesine geçme ve tahmin doğruluğunu artırma çabasının bir gerekçesidir. THYAO gibi halka açık ve işlem hacmi yüksek bir hissenin seçilmesi, sonuçların daha genellenebilir olmasına olanak tanımaktadır.

Araştırmanın Sınırlılıkları

Veri Seti Kısıtlılığı: Araştırma, yalnızca geçmiş kapanış fiyatları ve 200, 100 ve 50 günlük hareketli ortalamalar gibi sınırlı sayıda teknik göstergeye dayanmaktadır. Hacim, açılış/yüksek/düşük fiyatlar, diğer teknik göstergeler (RSI, MACD, Bollinger Bantları vb.), temel analiz verileri (şirket bilançoları, gelir tabloları) veya makroekonomik göstergeler (faiz oranları, enflasyon) gibi ek özellikler modele dahil edilmemiştir. Bu eksiklik, tahmin modellerinin piyasayı tam olarak temsil etme yeteneğini kısıtlayabilir.

Model Seçimi: Çalışmada sadece dört spesifik makine öğrenimi ve derin öğrenme modeli (LR, RF, LSTM, YSA) kullanılmıştır. Farklı veya daha gelişmiş modeller (örneğin, Prophet, XGBoost, GRU) daha farklı sonuçlar verebilir.

Dış Faktörlerin Etkisi: Savaşlar, doğal afetler, siyasi istikrarsızlık veya ani ekonomik şoklar gibi beklenmedik dış faktörler, modellerin tahmin yeteneğini önemli ölçüde etkileyebilir ve bu faktörler mevcut modele dahil edilmemiştir.

Tek Hissede Yoğunlaşma: Araştırma sadece bir hisse senedine odaklanmıştır. Elde edilen sonuçlar diğer hisse senetleri veya farklı piyasalar için doğrudan genellenemeyebilir.

Araştırmanın Varsayımları

Verilerin Güvenilirliği: Yahoo Finance'ten çekilen hisse senedi verilerinin doğru ve güvenilir olduğu varsayılmaktadır.

Geçmişin Geleceği Yansıtması: Geçmiş fiyat hareketlerinde bulunan bazı örüntülerin ve ilişkilerin gelecekte de belirli bir ölçüde devam edeceği varsayılmaktadır. Bu, tüm zaman serisi tahmin modellerinin temel varsayımıdır.

Sabit Model Parametreleri: Model eğitiminden sonra elde edilen parametrelerin, tahmin yapıldığı anda geçerli olduğu varsayılmaktadır. Piyasa koşulları değiştikçe bu parametrelerin de değişebileceği göz ardı edilmiştir.

Teknik Analizin Geerlilięi: Kapanıř fiyatı ve hareketli ortalama gibi teknik analiz gstergelerinin, hisse senedi fiyat hareketlerini tahmin etmede anlamlı bilgi ierdięi varsayılmaktadır.



BİRİNCİ BÖLÜM

1. LİTERATÜR TARAMASI

Ersayan (2022) tarafından yapılan çalışmada bir sonraki günün kapanış fiyatlarını tahmin etmeye yönelik bir arayüz geliştirilmiştir. Geliştirilen arayüzde hisse senedi, yıl, algoritma ve metrik seçimi için listeler kullanılmıştır Eğitim ve test verileri seçilen hisse senedinin ve yıllara göre verilerden ön işlemeye tabi tutularak oluşturulmuştur. Seçilen algoritmalar eğitim verileri üzerinden eğitilmiş, algoritmaların yeniden eğitilmesine gerek kalmadan bir sonraki günün kapanış fiyatları tahmin edilmiş ve seçilen kriterlere göre sonuçlar elde edilmiştir. Destek Vektör Regresyonu (SVR), K-En Yakın Komşular (KNN), Karar Ağaçları, Rastgele Orman, eXtreme Gradient Boosting (XGboost), Çok Katmanlı Algılayıcılar (MLP), Convolutional Neural Network (CNN), Tekrarlayan Sinir Ağları (RNN) ve Uzun Kısa – Süreli Bellek (LSTM) algoritmalarını ile Ortalama Mutlak Hata (MAE), Ortalama Mutlak Yüzde Hata (MAPE), Ortalama Kare Hatası (MSE), Karekök Ortalama Hata (RMSE) ve Belirlilik Katsayısı (R^2) metriklerini ölçeklendirmek için kullanmıştır. En iyi sonucu veren algoritma olarak CNN algoritması belirlenmiş, veri sayısının artması sonucu Destek Vektör Makinesi (SVM) algoritmasının tahmin doğruluğunun düşmesi ve veri sayısının artması sonucu LSTM algoritmasının tahmin doğruluğunun yükselmesini sonuçlarını bulmuştur.

Arslan ve Kırıcı (2021) tarafından yapılan çalışmada makine öğrenmesi ile son yıllara ait verileri kullanarak bir sistemi eğitmiş ve eğitilen sistemde gelecekteki Bitcoin verilerini görselleştirip borsa hareketlerine göre kullanıcılara al ya da sat sinyalleri üretmiştir. Yöntem olarak sklearn kütüphanesinde bulunan doğrusal regresyon yöntemini kullanmıştır. Bu yöntem ek olarak borsada sık kullanılan göstergelerden MACD ve RSI indikatörleri eklemiştir. Kullanılan veri setinin %80'nini eğitim verisi, %20'sini ise test verisi olarak kullanılmıştır. Yapılan testler sonucunda doğruluk oranının %95,5 olduğu saptanmıştır.

Arslankaya ve Toprak (2021) tarafından yapılan çalışmada Ereğli Demir ve Çelik Fabrikaları A.Ş.'nin hisse senedi fiyat tahminlerini makine öğrenmesi teknikleri olan Polinom Regresyon ve Rastgele Orman Regresyonu ile derin öğrenme teknikleri olan RNN ve LSTM algoritmalarını kullanarak borsa kapanışında hisse senedi fiyatlarını tahmin etmedeki başarılarını değerlendirmektedir. Algoritmalar Python 3.7 programlama dilinde Spyder editörü (Anaconda3) kullanılarak oluşturulmuştur. Veri tabanı olarak kullanılan Excel 2013 programında yer alan hisse senedi kapanış günleri ve fiyatlarından oluşan veri seti kullanılarak gerçekleştirilmiştir. Toplam veri sayısı 1619'dur. Tahmin sonuçlarının başarısının istatistiksel analizi MSE, MAE ve RMSE performans değerlendirme ölçütleri kullanılarak

gerçekleştirilmiştir. Modellerin başarı durumları hata değerlerine göre sırasıyla Rastgele Orman Regresyon, LSTM, RNN ve Polinom Regresyon modeli olduğu görülmüştür.

Toprak (2023) tarafından yapılan çalışmada Petkim Petrokimya Holding A.Ş.'nin hisse senedi kapanış fiyatı incelenmiştir. Türkiye, Rusya ve Katar'daki önemli petrol endeksleri ve hisse senedi fiyatlarını içeren bir veri seti kullanılarak hesaplanmıştır. Rastgele Orman Regresyon (RFR), Uzun Kısa Süreli Bellek (LSTM) ve Evrişimli Sinir Ağı + Uzun Kısa Süreli Bellek (CNN + LSTM) yöntemleri kullanılarak değerlendirilmiş ve sonuçlar karşılaştırılmıştır. Tahmin başarısını ölçmek için Ortalama Mutlak Hata (MAE), Ortalama Mutlak Yüzde Hata (MAPE), Ortalama Kare Hatası (MSE), Karekök Ortalama Hata (RMSE) ve Belirlilik Katsayısı (R^2) gibi hata ölçüm teknikleri kullanılmıştır. RFR, LSTM ve CNN+LSTM algoritmalarının performansı karşılaştırıldığında, RFR algoritmasının verilerdeki aykırı değerlere karşı sağlamlığının diğer yöntemlere göre daha düşük olduğu görülmüştür. Bu, eğitim veri kümesi boyutunun en yüksek (%80) ve test veri kümesi boyutunun en düşük (%20) olduğu durumda iyi sonuçlar vermiştir. Ancak diğer durumlarda diğer algoritmalarla göre daha az başarılı sonuçlar üretmiştir.

Doğan ve Büyükkör (2022) tarafından yapılan çalışmada gelişmiş ve gelişmekte olan iki borsa olan New York Borsası (NYSE) ve Borsa İstanbul Ulusal-100 Endeksi (BIST 100) endeklerinin yanı sıra BIST 100'ün iki büyük hisse senedi olan Aselsan ve Ereğli Demir Çelik'in son 5 yıllık kapanış verileri kullanılmıştır. Endeksleri tahmin etmede Destek Vektör Regresyonu (SVR), Rastgele Orman (RF) ve Extreme Gradient Boosting (XGBoost) makine öğrenme algoritmaları kullanılmıştır. Ölçüt olarak da MAE, MAPE ve RMSE metrikleri ile değerlendirme yapılmış olup en iyi tahminin Destek Vektör Regresyonu olduğu tespit edilmiştir.

Kalyoncu (2020) tarafından yapılan çalışmada iki farklı veri seti olarak incelenmiştir. Birinci veri setinde BİST 30 da bulunan Türk Hava Yolları (THYAO), Arçelik, Akbank, Aselsan ve Garanti Bank şirketlerinin hisse senetlerinin 2014 – 2019 yılları arasındaki son 5 yıla ait hisse senedi fiyatları yer almaktadır. İkinci veri seti ise Hindistan piyasasında bulunan Tata Global Beverages Limited şirketine ait 2010 – 2019 yılları arasındaki hisse senedi fiyatlarını bulundurmaktadır. Yapay Sinir Ağı (YSA), k-En Yakın Komşular (KNN), Oto-Regresif Entegre Hareketli Ortalamalar (Auto ARIMA) ve Uzun Kısa Süreli Bellek (LSTM) öğrenme yöntemleri kullanılmıştır. Ölçüt olarak RMSE ve MAE değerleri dikkate alınmıştır. TATAGLOBAL veri seti için YSA ve ARIMA modellerinin KNN makine öğrenmesi algoritmasına göre başarı oranının daha iyi olduğu söylenmiştir. BİST 30 veri seti için ise

LSTM derin öğrenme modeli ile ilgili olarak RMSE ve MAE metrik ölçütlerinin başarı olduğu gözlemlenmiştir.

Altunbaş (2021) tarafından yapılan çalışmada XBANK adı ile ifade edilen bankanın endeks verilerinin sonraki gün kapanış değerlerinin tahmin edilmesi için derin öğrenme yöntemleri kullanılması hedeflenmiştir. 2015 – 2019 tarihleri arasındaki bankanın borda endeks değerleri ile farklı teknik analiz göstergeleri, farklı makroekonomik değişkenler ve metin verileri kullanılmıştır. RSI, Momentum, William, Basit Hareketli Ortalama, MACD, Stokastik K ve Stokastik D gibi teknik analiz göstergeleri, euro kuru, dolar kuru, Brent petrol ve gram altın değerleri de makroekonomik göstergeler için kullanılmıştır. Analizde kullanmak için üç çeşit veri seti oluşturulmuştur. Bunlardan birincisi finansal ile metinsel veri setleri ile oluşturulmuştur. İkinci sade finansal verilerin yer aldığı veri seti, üçüncüsünde ise sadece metinsel verilerin olduğu veri setinden oluşturulmuştur. Veri setlerinden modele pozitif anlamda bir faydası olmayan veriler Boruta algoritması ile çıkarılmıştır. Daha sonra model seçimi olarak derin öğrenme modellerinden olan ileri beslemeli derin sinir ağı modeli kullanılmıştır. Sonuç olarak endeks yönünün doğru tahmin etmiş fakat kapanış fiyatları için sapmalar olduğunu gözlemlemiştir.

Demirel (2019) tarafından yapılan çalışmada kullanılan veri seti, Borsa İstanbul Ulusal-100 Endeksi'nde (BIST 100) işlem gören şirketlerin hisselerinin Ocak 2010'dan Ocak 2019'a kadar ikinci seans günlük açılış ve kapanış fiyatlarına ilişkin verilerdir. Çalışma, makine öğrenmesi ve derin öğrenmeyi kullanarak bu hisse senetlerinin açılış ve kapanış fiyatlarını tahmin etmeyi amaçlamaktadır. Bu bağlamda tahmin modeli oluşturmak için Çok Katmanlı Algılayıcılar (ÇKA) ve SVM gibi klasik makine öğrenme yöntemlerinin yanı sıra LSTM gibi derin öğrenme algoritmaları kullanılmıştır. Çalışmada ÇKA ve LSTM ağlarının SVM 'ye göre daha tutarlı tahminler ürettiği ortaya çıkmıştır.

Öngün (2023) tarafından yapılan çalışmada yapay zekanın finans alanındaki uygulamalarından biri olan hisse senedi tahmin algoritmalarına odaklanmaktadır. Çok değişkenli günlük zaman serisi olarak ele alınan problemde, bağımlı değişken Vestel Beyaz Eşya Şirketi hisse senedinin belirli bir zaman aralığındaki kapanış değerleri, bağımsız değişkenler ise o hisse senedini etkileyen diğer değerlerdir. Bu değişkenlerden oluşturulan veri seti kullanılarak doğrusal regresyon, destek vektör makinesi, polinom regresyon, karar ağacı, rastgele orman ve LSTM algoritmaları kullanılarak tahmin modelleri uygulanmıştır. Modellerin hata oranları karşılaştırılarak modeller arasında hangi algoritmanın en uygun sonucu ürettiği tartışılmış ve çok değişkenli finansal zaman serisi analizi için en iyi yöntemin LSTM algoritması olduğu gösterilmiştir.

Pabuçcu (2019) tarafından yapılan çalışmada BIST 100 endeksinin hareket yönü tahmin edilmeye çalışılmıştır. Tahmin için Yapay sinir ağları, destek vektör makineleri ve naive Bayes sınıflandırma algoritmaları olan üç çeşit makine öğrenme algoritması kullanmış ve performanslarını karşılaştırmıştır. Girdi olarak Hareketli Ortalama, Ağırlıklı Hareketli Ortalama, Momentum, Göreceli Güç Endeksi, Hareketli Ortalama Yakınsama İraksama gibi on teknik analiz gösterge aracı modeller için kullanılmıştır. 2009 – 2018 tarihleri arasındaki günlük kapanış fiyatları alınarak veri setini oluşturulmuştur. Sonuç olarak kullandığı üç çeşit modelin de borsa endeksinin hareketliliğini yakalamada kullanılabilir olduğunu, YSA sınıflandırma algoritmasının daha iyi bir model olduğu hakkında bilgi verilmiştir.

Kocaoğlu vd. (2022) tarafından yapılan çalışmada BIST 30 Endeksi'nde işlem gören farklı sektörlerdeki şirketlerin hisse senetlerinin gelecekteki fiyatlarının tahmin edilmesi hedeflenmiştir. Çalışma için holding, beyaz eşya, petrokimya, demir çelik, ulaştırma ve iletişim sektörlerinin her birinden örnek olarak seçilen iki şirketin verileri analiz edilmiştir. Çalışmada sektörlerin hisse senetlerinin analizinin yanı sıra makine öğrenmesi algoritmasının sektörel bazda fiyat tahmin performansları da incelenmiştir. Bu testler için XGBoost, Destek Vektör Makineleri (SVM), K-en yakın komşular (KNN) ve Rastgele Orman (RF) algoritmaları kullanılmıştır. Elde edilen sonuçlar ortalama mutlak hata (MAE), ortalama mutlak yüzde hata (MAPE), ortalama karesel hata (MSE) ve R2 korelasyon metrikleriyle analiz edilmiştir. Sektörel bazda en iyi tahminler Demir Çelik ve Petrol alanındaki firmalar için yapılmıştır. Çalışmadaki en önemli yeniliklerden biri mevcut makro değişikliklerin tahmin modeline etkisinin incelenmesidir. Örnek olarak 5 yıllık dönemde üç kez gerçekleşen Merkez Bankası Başkan değişikliklerinin tahminlere etkisi araştırılmıştır. Sonuç olarak, Başkan değişikliği sonrasında politikalar üzerindeki öngörülemez etkilerin tahmin performansını da olumsuz etkilediği hakkında bilgi sunulmuştur.

Nikou vd. (2019) tarafından yapılan çalışma iShares MSCI Birleşik Krallık Endeksi'nin kapanış fiyatını tahmin etmek amacıyla gerçekleştirilmiştir. Bu amaçla, dört veri madenciliği tekniği (yani YSA, SVR, RF ve LSTM) Python'da kodlama kullanılarak uygulanmış ve işlevleri Ocak 2015'ten Haziran 2018'e kadar iShares MSCI Birleşik Krallık Endeksi'nin günlük kapanış fiyatı ile karşılaştırılmıştır. LSTM bloklu tekrarlayan ağ yönteminin iShares MSCI Birleşik Krallık Endeksi'nin kapanış fiyatının tahmininde diğer yöntemlerden daha iyi işlev gördüğünü ve ardından SVR yönteminin sinir ağı ve RF 'den daha yüksek hassasiyete sahip olduğunu ifade etmiştir. iShares MSCI Birleşik Krallık Endeksi'nin kapanış fiyatını tahmin etmek için derin öğrenme yönteminin test edilen diğer yöntemlerden daha iyi olduğunu; bu nedenle yatırımcılara, yöneticilere ve piyasa analizcilerine bu yöntemi kullanmalarını önermiştir.

Yapılan çalışmanın ve elde edilen sonuçların yatırım şirketlerine ve diğer kullanıcılara hisse senedi alım satımındaki karlılığın değerlendirilmesinde yardımcı olacağını belirtmiştir.

Vijh vd. (2020) tarafından yapılan çalışmada farklı sektörlere ait beş şirketin ertesi gün kapanış fiyatını tahmin etmek için YSA ve RF teknikleri kullanılmıştır. Finansal veriler, hisse senedinin Açılış, Yüksek, Düşük ve Kapanış fiyatları, modele girdi olarak kullanılan yeni değişkenleri oluşturmak için kullanılmıştır. Modeller de standart stratejik göstergeler kullanılarak değerlendirilmiştir. RMSE ve MAPE bu iki göstergenin düşük değerleri, modellerin hisse senedi kapanış fiyatını tahmin etmede etkin olduğunu göstermektedir. YSA, hisse senedinin bir sonraki gün kapanış fiyatını tahmin etmek için kullanılmış ve karşılaştırmalı bir analiz için RF de uygulanmıştır. RMSE, MAPE ve Ortalama Önyargı Hatası (MBE) değerlerine dayalı karşılaştırmalı analiz, YSA'nın RF'ye kıyasla hisse senedi fiyatlarını daha iyi tahmin ettiğini açıkça göstermiştir. Sonuçlar, YSA modeli tarafından elde edilen en iyi değerlerin RMSE (0,42), MAPE (0,77) ve MBE (0,013) olduğunu göstermiştir. Gelecekteki çalışmalarda, muhtemelen daha iyi sonuçlar için kapanış fiyatı, işlem hacmi, kar ve zarar tabloları gibi finansal parametrelerle birlikte finansal haber makalelerini dikkate alan derin öğrenme modelleri geliştirilebileceğini sonucuna varılmıştır.

Patel vd. (2014) tarafından yapılan çalışmada hisse senedi endekslerinin gelecekteki değerlerini tahmin etmek amaçlanmıştır. Deneyler, Hindistan hisse senedi piyasalarından CNX Nifty ve S&P BSE Sensex olmak üzere iki endeksin on yıllık geçmiş verileri üzerinde gerçekleştirmiştir. Tahminler 1-10, 15 ve 30 gün öncesinden yapılmıştır. Belirlenen sorunu ele almak için ilk aşamada SVR ve ikinci aşamada YSA, RF ve SVR içeren iki aşamalı bir füzyon yaklaşımı önerilmiştir. Tek aşamalı ve iki aşamalı füzyon tahmin modelleri ile deneyler yürütülmüştür. Sonuçlar, iki aşamalı hibrit modellerin tek aşamalı tahmin modellerinden daha iyi performansa sahip olduğunu göstermiştir. Performans iyileştirmesi, YSA ve RF'nin SVR ile melezleştirildiği durumda önemlidir. SVR'nin kendisiyle melezleştirildiği durumda performansta orta düzeyde bir iyileştirme gözlemlenmiştir. İki aşamalı tahmin modellerinin tek aşamalı tahmin modellerine göre avantajları, tahminler daha fazla gün önceden yapıldıkça belirginleşmiştir. En iyi genel tahmin performansı SVR-YSA modeli ile elde edilmiştir. Bu, yatırımcılara 1-10 gün veya 15 gün veya 30 gün için 'zarar durdurma emri' yani 'bir ay için zarar durdurma emri' önererek yardımcı olabileceği ön görülmüştür.

Tsai ve Wang 2009 tarafından yapılan çalışmada yatırımcıların hisse senedi yatırımlarında daha doğru kararlar vermelerine yardımcı olmak amacıyla yaygın olarak kullanılan YSA tekniklerini ve DT mükemmel açıklama yeteneğini daha iyi bir karar destek sistemi için birleştirmiştir. YSA ve DT temel modelleri geliştirilerek, DT + YSA ve DT + DT

olmak üzere iki birleşik model karşılaştırılmıştır. Bu modellerin performansını incelemek için kayan pencere yöntemi kullanılmıştır. Deneysel sonuçlara göre, DT + YSA hibrit modeli, elektronik endüstrisi hisselerinde %77 tahmin doğruluğuna sahip ve yatırımcıların yatırımlarını ne zaman gerçekleştireceklerini belirlemelerine yardımcı olmak için güvenilir kurallar sağlamıştır. Özellikle, hisse senedi fiyat artışını ve düşüşünü tahmin etmek için on iki farklı karar kuralı olduğunu belirtmiştir.

Kurucan (2019) tarafından yapılan çalışma İstanbul Menkul Kıymetler Borsası 100 Endeksi üzerinde gerçekleştirilmiştir. Makine öğrenmesi modelleri kullanılarak endeksin günlük yüksek fiyatı tahmin edilmeye çalışılmıştır. Bu verilerle günlük yatırım stratejileri oluşturularak pasif yatırımlara göre daha yüksek getiri elde edilmesi amaçlanmıştır. Çalışmada kullanılan makine öğrenmesi yöntemlerinden YSA ve karar ağaçlarını temel alan bir gradyan artırma modeli olan XGBoost modeli kullanılmıştır. Kullanılan modeller farklı performans kriterleri ile karşılaştırılmıştır. İki farklı zaman diliminde oluşturulan modellerde XGBoost modellerinin YSA modellerine göre daha iyi sonuçlar ürettiği görülmüştür. Seçilen bu modeller kullanılarak iki çeşit yatırım stratejisi oluşturulmuştur. Oluşturulan stratejiler için Borsa İstanbul fiyat oluşum seansının yapısı dikkate alınmıştır. Oluşturulan birçok modele göre XGBoost modelinin borsanın yönünü tahmin etmede yapay sinir ağı modelinden daha iyi performans ortaya çıkardığı gözlemlenmiştir. Çalışma, 5 yıllık bir veri seti kullanılarak 6 aylık bir süre boyunca oluşturulan bir modelde test edilmiştir. Yapılan test zamanında Borsa İstanbul 100 Endeksi'ne al-tut stratejisiyle yatırım uapmanın yüzde 9 kayıpla sonuçlanacağını tespit edilmiştir. Makine öğrenimi ile oluşturulan modeller kullanılarak günlük alım-satım stratejileri oluşturulmuştur. Yatırım stratejisinin ilkinde aynı dönemde yüzde 12'lik dönemsel kazanç elde edilmiştir. İkinci yatırım stratejisinin ise yüzde 31,6 getiri ile dönemsel olarak hepsinden daha yüksek kazanç sağlayabildiği kaydedilmiştir. Test edilen tüm yatırım stratejilerinde pasif yatırıma, yani al-tut stratejisine kıyasla daha iyi performans gözlemlenmiştir. Bu strateji ile Borsa İstanbul'un düşüş eğilimi gösterdiği dönemlerde dahi getiri elde edilebileceği kaydedilmiştir.

Karacan ve Kırdar (2021) tarafından yapılan çalışmada yatırımcıların doğru alım satım kararları almalarına yardımcı olmak için yapay zekâ ve makine öğrenimi alanındaki araştırmalardan bahsedilmiştir. Bu yeni teknolojilerin başarıları vurgulanmıştır. Portföy yönetiminde robotik danışmanlık ve yapay zekanın finasta kullanımı gibi konular yakın gelecekte profesyonel ve akademik çevrelerde gündemde kalmaya devam edecek, önem ve popülerlik kazanacak ve yeni araştırmalarla desteklenmesi gerekeceğini belirtmiştir.

Akşehir ve Kılıç (2019) tarafından yapılan çalışma 1 Ocak 2016 ile 9 Mayıs 2019 tarihler arasındaki verilere bakarak seçilmiş bankacılık hisse senetlerinin ertesi gün kapanış fiyatları tahmin edilmeye çalışılmıştır. Tahmin modelinin geliştirilmesinde stok hareketlerinde bulunan doğrusal modeller nedeniyle karar ağacı, çoklu regresyon ve rastgele orman teknikleri kullanılmıştır. Bu yöntemler kullanılarak oluşturulan modeller için iki farklı girdi değişkeni seti kullanılmıştır. İlk veri setinde 46 teknik ve 4 temel göstergeden oluşan 50 gösterge bulunmaktadır. İkinci set ise teknik göstergelerde yapılan kesintiler sonucunda 29 teknik gösterge ve 4 temel göstergeden oluşan 33 göstergeden oluşmuştur. Her iki modelin bu iki farklı girdi değişkeni seti için tahmin performansı R^2 performans kriteri kullanılarak değerlendirilmiştir. R^2 sonuçlarına bakıldığında teknik göstergelerde yapılan indirimin modellerin tahmin etkisini olumlu yönde etkilediği kaydedilmiştir.

Filiz vd. (2017) tarafından yapılan çalışmada BIST-50 endeksine etki eden faktörler kullanılarak bazı makine öğrenmesi teknikleri ve yapay sinir ağları kullanılarak sınıflandırmalar yapılmıştır. Literatürde çok kez kullanılan BIST-50 endeksinin değişimini etkileyen faktörlerden Almanya Birleşik Borsa Endeksi (DAX), Londra Borsa Endeksi (FTSE), Amerikan Borsa Endeksi (S&P 500), Borsa İstanbul Bankacılık Endeksi (BISTBANK), İstanbul Mali Endeksi (BISTMALI), Borsa İstanbul Sanayi Endeksi (BISTSINAI), BIST Altın Endeksi (GOLDINDEX), Euro/Dolar paritesi, ham petrol fiyatı ve faiz oranları alınmıştır. Sınıflandırma için k-en yakın komşular (k-NN), Naive Bayes sınıflandırıcısı, C4.5 sınıflandırma algoritması ve YSA gibi makine öğrenme yöntemleri kullanılmıştır. Analizler için Weka3.8 programı kullanılmış, bu analizler sonucunda doğru sınıflandırmada en iyi performansı %92,71 oranıyla C4.5 sınıflandırma algoritmasının gösterdiğini ifade etmiştir.

İKİNCİ BÖLÜM

2. MATERYAL VE YÖNTEM

Bu çalışmada veri seti olarak herhangi bir hisse senedi belirtilmemiştir. Sadece yapılan uygulamayı göstermek amacıyla örnek olarak Türk Hava Yolları A.Ş. firmasının hisse senedi alınmıştır. Bunu yapmaktaki amaç yazılan kodlama da etiket olarak belirtilen değişkenin kullanıcı tarafından başka bir hisse senedinin Borsa İstanbul'daki kodu girilmesi ve programın baştan itibaren yeniden çalıştırılması sonucu, girilen hisse senedinin sonuçlarının çıkmasıdır. Yani her hisse senedi için bu programın çalışıyor olmasıdır. Dikkat edilmesi gereken husus, çalışmada 200, 100 ve 50 günlük hareketli ortalama indikatörü kullanılmasıdır. Bundan dolayı örnek alınacak hisse senedinin Borsa İstanbul da tatil günleri hariç 200 günün üzerinde işlem yapması gerekmektedir. Veri setinin başlangıç tarihi kullanıcıya bırakılmış ancak bitiş tarihi bir sonraki günün hisse senedi fiyat tahminini yapmasını istediğimiz için o günün tarihi olması gerekmektedir (Ör: 12.12.2021 – 12.12.2024).

Veri seti Python kodu kullanılarak yahoofinance sitesinden çevrimiçi olarak alınmıştır. Sitede bulunan finance modülünde hisse senedinin borsada işlem gördüğü hisse senedi kodu (Ör: THYAO.IS) yazılarak veri elde edilmiştir.

Alınan veri seti öncelikle veri ön işleme işlemi yapılmış ve daha sonra eğitim ve test verisi olarak iki kısma ayrılmıştır. Yüzde 80 eğitim verisi, yüzde 20 de test verisi olarak alınmıştır.

Veri ön işleme aşaması bittikten sonra makine öğrenmesi ve derin öğrenme yöntemlerinden olan 4 farklı model oluşturulmuştur. Bu 4 model LR, RF, LSTM ve YSA olarak oluşmaktadır. Eğitilen veri seti bu 4 model üzerinden çalıştırılmış ve sonuçlar grafiksel olarak gösterilmiştir.

Model sonuçları performans değerlendirme ölçütleri olarak adlandırılan ölçütler ile değerlendirilmiş ve grafiksel olarak gösterilmiştir. Kullanılan performans değerlendirme ölçütleri MAE, MSE, RMSE ve R- Kare şeklindedir.

2.1. VERİ ÖN İŞLEME

Veri ön işleme, ham verileri kullanışlı ve anlaşılır bir biçime dönüştürme sürecidir. Gerçek veya ham veriler genellikle tutarsız biçimde, insan hatalarına sahip ve eksik olabilir. Veri ön işleme bu tür sorunları çözer ve veri kümelerini veri analizi için daha eksiksiz ve verimli hale getirir. Bu durum veri madenciliği ve makine öğrenimi projelerinin başarısını etkileyebilecek önemli bir süreçtir. Veri kümelerinden bilgi keşfini hızlandırır ve nihayetinde

makine öğrenimi modellerinin performansını etkileyebilir. Başka bir deyişle, veri ön işleme verileri bilgisayarların kolayca çalışabileceği bir biçime dönüştürmektir. Verileri analiz etmeyi veya görselleştirmeyi kolaylaştırır ve veriler üzerinde eğitilen makine öğrenimi ve derin öğrenme algoritmalarının doğruluğunu ve hızını artırır. Bir veri tabanı veri noktalarının bir koleksiyonudur. Veri noktalarına gözlemler, veri örnekleri, olaylar ve kayıtlar da denir. Her örnek, özellikler veya öznitelikler olarak da bilinen çeşitli özellikler kullanılarak tanımlanır. Veri ön işleme, bu özelliklere sahip modelleri etkili bir şekilde oluşturmak için gereklidir. Veri toplanırken ortaya çıkabilecek birçok sorun vardır. Farklı veri kaynaklarından veri toplamanız gerekebilir ve bu da tam sayılar ve float sayıları gibi uyumsuz veri biçimleriyle sonuçlanabilir. Örneğin, iki veya daha fazla bağımsız veri kümesinden veri topluyorsanız, cinsiyet alanı kadınlar ve erkekler için iki farklı değere sahip olabilir. Verilerin önceden işlenmesi, yorumlanması ve kullanılmasını kolaylaştırır. Verilerin önceden işlenmesi, insan hatası veya yanlışlarından kaynaklanan yanlış veya eksik değer olup olmadığı hakkında bilgi verir. Kısacası, veri ön işleme tekniklerini kullanmak veri tabanını daha eksiksiz ve doğru hale getirir.

2.1.1. Veri Kalitesini Etkileyen Faktörler

Veri ön işleme işleminin nasıl yapıldığına bakmadan önce, veri kalitesini etkileyen faktörlerden bazıları aşağıda incelenmiştir.

Doğruluk: Doğruluk, bilginin doğru olduğu anlamına gelir. Güncel olmayan bilgiler, yazım hataları ve tekrarlar bir veri setinin doğruluğunu etkileyebilir.

Tutarlılık: Verilerde tutarsızlık olmamalıdır. Tutarsız veriler aynı soruya farklı yanıtlar verebilir.

Tamlık: Veri setinde eksik veya boş alan olmamalıdır. Bu özellik, veri bilimcilerinin doğru analizler yapmalarını sağlar çünkü veriler tarafından açıklanan durumun tam bir resmine erişebilirler.

Geçerlilik: Veri örnekleri doğru biçimdeyse, belirtilen aralıktaysa ve doğru türdeyse bir veri seti geçerli kabul edilir. Yanlış veri setlerini düzenlemek ve analiz etmek zordur.

Zamanlılık: Veriler, temsil ettiği olay meydana gelir gelmez toplanmalıdır. Zamanla, her veri seti güncel gerçekliği yansıtmadığı için daha az doğru ve kullanışlı hale gelir. Bu nedenle, verilerin zamanında ve alakalı olması veri kalitesinin en önemli özelliğidir.

2.1.2. Veri Ön İşleme Adımları

Veri temizleme: Veri temizleme veya arındırma, eksik değerleri hesaba katarak, aykırı değerleri kaldırarak, tutarsız veri noktalarını düzelterek ve gürültülü verileri yumuşatarak veri kümelerini temizleme işlemidir. Özünde, veri temizlemenin arkasındaki amaç, makine

öğrenimi ve derin öğrenme modelleri için eksiksiz ve doğru örnekler sunmaktır. Veri temizlemede kullanılan teknikler, veri bilimcinin tercihlerine ve çözmeye çalıştığı soruna özgüdür. Eksik değerler ve gürültülü veriler, veri temizleme ve ilgili teknikler sırasında çözülen sorunlardır.

Veri Entegrasyonu: Veriler çeşitli kaynaklardan toplandığından, veri entegrasyonu veri hazırlamanın çok önemli bir parçasıdır. Entegrasyon, çok sayıda tutarsız ve gereksiz veri noktasına yol açarak sonuçta daha düşük doğruluğa sahip modellere yol açabilir. Aşağıda verileri entegre etmek için bazı yaklaşımlar verilmiştir:

Veri konsolidasyonu: Veriler fiziksel olarak bir araya getirilir ve tek bir yerde saklanır. Tüm verilerin tek bir yerde olması verimliliği ve üretkenliği artırır. Bu adım genellikle veri ambarı yazılımının kullanılması içerir.

Veri sanallaştırma: Bu yaklaşımda, bir arayüz birden fazla kaynaktan gelen verilerin birleşik ve gerçek zamanlı görünümünü sağlar. Başka bir deyişle, veriler tek bir bakış açısından görüntülenebilir.

Veri yayılımı: Belirli uygulamalar yardımıyla verilerin bir konumdan diğerine kopyalanmasını içerir. Bu işlem senkron veya asenkron olabilir ve genellikle olay güdümlüdür.

Veri Dönüşümü: Bu, verilerin analiz için uygun bir biçime dönüştürülmesini içerir. Veri dönüşümünde kullanılan yaygın teknikler arasında normalizasyon, standardizasyon ve ayırıklaştırma bulunur. Normalizasyon, verileri ortak bir aralığa ölçeklemek için kullanılırken, standardizasyon, verileri sıfır ortalama ve birim varyansa sahip olacak şekilde dönüştürmek için kullanılır. Ayırıklaştırma, sürekli verileri ayırık kategorilere dönüştürmek için kullanılır.

Veri Azaltma: Bu, önemli bilgileri korurken veri kümesinin boyutunu azaltmayı içerir. Veri azaltma, özellik seçimi ve özellik çıkarma gibi tekniklerle elde edilebilir. Özellik seçimi, veri kümesinden ilgili özelliklerin bir alt kümesini seçmeyi içerirken, özellik çıkarma, önemli bilgileri korurken verileri daha düşük boyutlu bir alana dönüştürmeyi içerir.

Veri Ayırıştırma: Bu, sürekli verileri ayırık kategorilere veya aralıklara bölmeyi içerir. Ayırıştırma, kategorik veriler gerektiren veri madenciliği ve makine öğrenimi algoritmalarında sıklıkla kullanılır. Ayırıştırma, eşit genişlikte bölme, eşit frekanslı bölme ve kümeleme gibi tekniklerle elde edilebilir.

Veri Normalizasyonu: Bu, verileri 0 ile 1 veya -1 ile 1 gibi ortak bir aralığa ölçeklendirmeyi içerir. Normalizasyon genellikle farklı birimlere ve ölçeklere sahip verileri işlemek için kullanılır. Yaygın normalizasyon teknikleri arasında min-maks normalizasyonu, z-puanı normalizasyonu ve ondalık ölçekleme bulunur.

2.2. PERFORMANS DEĞERLENDİRME ÖLÇÜTLERİ

Bu çalışmada derin öğrenme ve makine öğrenmesi tekniklerinin performansını değerlendirmek için aşağıdaki performans değerlendirme ölçütleri kullanılmıştır.

Belirlilik Katsayısı (R^2)

Ortalama Mutlak Hata (MAE)

Ortalama Karesel Hata (MSE)

Ortalama Karesel Hatanın Kökü (RMSE)

2.2.1. Belirlilik Katsayısı

R^2 , model tarafından açıklanan serideki toplam varyasyonun oranının bir tahminidir ve ölçü, seri durağan olduğunda faydalıdır. Durağan R^2 , modelin durağan kısmını basit bir ortalama modelle karşılaştıran bir ölçüdür ve bir eğilim veya mevsimsel desen olduğunda sıradan R^2 'ye tercih edilir. Durağan R^2 , negatif sonsuzluktan 1'e kadar bir aralıkta negatif olabilir. Negatif değerler, incelenen modelin temel modelden daha kötü olduğu anlamına gelir. Pozitif değerler, incelenen modelin temel modelden daha iyi olduğu anlamına gelir.

2.2.2. Ortalama Mutlak Hata

Ortalama mutlak hata (MAE), regresyon analizinde kullanılan ve bir regresyon modelinin tahminlerinin gerçek değerlerden ne kadar saptığını ölçen bir performans ölçüsüdür. MAE, her tahminin mutlak değerinin gerçek değerden alınması ve bu mutlak hataların ortalamasının alınmasıyla hesaplanır. Sıfıra yakın veya sıfıra eşit bir MAE, modelin iyi performans gösterdiğini gösterir. Bu nedenle, MAE genellikle regresyon ve zaman serisi problemlerinde tercih edilen performans ölçüsüdür çünkü sonuçları yorumlanması kolaydır. MAE performans metriğinin denklemi şu şekilde ifade edilebilir: MAE, gerçek değerler ile öngörülen değerler arasındaki farkların mutlak değerlerinin toplamıdır. Bu ölçüt, sürekli iki değişken ile aralarındaki farkın bir ölçüsüdür ve MAE değeri 0 ile ∞ arasında değişebilmektedir.

2.2.3. Hata Kareleri Ortalaması

Hata Kareleri Ortalaması yada diğer adıyla Ortalama karesel hata (MSE), regresyon analizlerinde yaygın olarak kullanılan bir performans metriğidir ve bir regresyon modelinin öngörülen değerlerinin gerçek değerden ne kadar saptığını ölçer. MSE, her tahmin ile gerçek değer arasındaki farkın karesi alınarak ve ardından bu karelerin ortalaması alınarak hesaplanan bir ölçüttür. MSE değeri sıfırı yaklaştıkça, modelin tahminleri gerçek değerlere o kadar yaklaşır

ve daha düşük MSE değeri daha iyi model performansını yansıtır. Ancak, kareler kullandığı için orijinal hata ölçüsünü de kare alır, bu nedenle yorumlanması bazen daha zor olabilir.

2.2.4. Hata Kareleri Ortalaması Karekökü

Hata Kareleri Ortalama Karekökü (RMSE), regresyon analizinde kullanılan ve regresyon modelinin tahminlerinin gerçek değerlere göre ne kadar hata olduğunu ölçen bir performans ölçüsüdür. RMSE, hatanın ortalama karekökü alınarak hesaplanır. Böylece orijinal hata biriminde ifade edilen değeri temsil eder ve modelin tahmin yeteneğinin daha anlaşılır bir biçimde değerlendirilmesine olanak sağlar. RMSE değeri sıfıra yaklaştıkça modelin tahminleri gerçek değerlere yaklaşır ve daha düşük RMSE değeri daha iyi bir model performansını ortaya çıkarır. RMSE, birçok matematiksel olarak hesaplamada istenmeyen bir durum olan mutlak değerlerin kullanılmasını izin vermez.

2.3.MAKİNE ÖĞRENMESİ

Yapay zekanın altında bir dal olan makine öğrenmesi, türlü algoritmalar ve yöntemler ile eldeki veride bazı kalıpları arar ve bunlara karşılık gelen verilere göre önce öğrenir, sonra da eldeki deneyimlerinden yararlanarak ortaya bir çıkarım yapabilen sistemler geliştirilmesine olanak sağlar. Sağladığı bu imkânı, çeşitli istatistiksel ve matematiksel yöntemlerin kullanıldığı birden fazla algoritma ile elde eder. Kullanılan bu algoritmalar, verideki örnekleri inceleyerek, bu örneklerin meydana gelme şekillerini öğrendikten sonra problem üzerinde bir genelleme yapma yeteneği elde eder. Bu elde ettiği yetenek ile daha büyük veriler içerisinde fark edilmesi çok zor olan kalıpları bulup ortaya çıkarır. Bu şekilde yeni bir vakanın meydana gelmeden önce bu vakayı tahmin edebilmektedir. Örnek olarak bir yolcu uçağının ne kadar rötar yapacağı verilebilir. Genel anlamda makineyi eğitmek için 3 temel bileşene ihtiyaç vardır.

Veri Setleri: Veri setleri ismi verilen özel örnek koleksiyonları üzerinde eğitilir. Örnekler metinleri, resimleri, sayıları vb. verileri içerebilir. İyi bir veri seti oluşturmak için çok çaba ve zaman harcamak gerekir.

Özellikleri: Çözümün anahtarı olarak çalışan önemli veri parçalarıdır. Makineye dikkat etmesi gereken şeyleri gösterirler.

Algoritma: İstenilen bir görevi farklı algoritmalar kullanılarak çözüm yapılması mümkündür. Sonuçların hızlı ve farklı olması seçilen algoritmaya bağlı olabilir. İyi bir sonucun çıkması için bazen farklı algoritmaların birleştirilerek aynı anda analiz yapılması gerekmektedir.

2.3.1. Makine Öğrenmesi Teknikleri

Makine öğrenmesi algoritmaları nasıl öğrendiklerine bağlı olarak dört gruba ayrılır.

Denetimli Öğrenme: Veri setinin üzerinde hangi işlemlerin yapılacağını öğretmek için bilgisayara veri seti aktarılır. Bilgisayara girdiler, çıktılar ve özellikler verildikten sonra bilgisayar da verilen bu kalıplardan öğrenme işini yapmaktadır. Aşağıda bazı denetimli makine öğrenmesi algoritmaları verilmiştir.

Naive Bayes

Destek Vektör Makinesi

Karar Ağacı

K-En Yakın Komşu

Lojistik Regrasyon

Denetimsiz Öğrenme: Bu öğrenmede sadece büyük veri setleri mevcuttur. Veri seti ve çıktıların olmadığı bir öğrenme yöntemidir. Denetimsiz öğrenmede kümeleme işlemleri yer almaktadır. Uzaklık, benzerlik, yakınlık gibi ölçütlere göre analiz ederek sınıflara ayırmaktadır.

Yarı Denetimli Öğrenme: Bu denetimli ve denetimsiz öğrenme arasında kalan öğrenmedir. Etiketlenmiş küçük bir veri ile etiketlenmemiş bir verinin birlikte kullanılmasıdır.

Takviyeli Öğrenme: Bir dizi eylemi, sınıf parametrelerini ve veri setinden elde edilen nihai değeri içeren düzenli öğrenme süreçlerine sahip bir öğrenme yöntemidir. Bir makine öğrenimi algoritması, farklı çıktıları ve seçenekleri değerlendirerek sonuç almaya çalışır. Önerilerin makine öğrenimi, hataları öğretir ve hatalardan sonuçlar çıkararak bunların tekrar olmasını önler. En iyi sonucu daha önceki deneyimlerinden sonuçlar çıkararak elde etmeye çalışır (Görgün, 2020).

2.3.2. Doğrusal Regresyon

Doğrusal regresyon, bir öğrenme modeli kurmayı ve girdi verilerine en yararlı şekilde doğrusal bir çizgi oluşturarak iki değişken arasındaki ilişki kurmaya çalışmayı amaçlayan denetimli bir makine öğrenme tekniğidir (Bardak vd., 2024).

Tahmin yapmak için doğrusal regresyon modelini kullanmaya başlamadan önce, tahmin edilen değişken (bağımlı) ile girdi verileri (bağımsız değişkenler) arasında bir ilişki olup olmadığının belirlenmesi gerekir.

Doğrusal regresyon, en uygun düz olan çizgiyi veri giriş noktaları boyunca aramaktadır. En uygun olan çizgiye regresyon çizgisi denmektedir (Bardak vd., 2024). Bir veri seti için bir

regresyon çizgisi modellendikten sonra, çizgiden uzakta bulunan veri değerlerine aykırı değerler denmektedir (Bardak vd., 2024). Aykırı değerler yanlış verileri temsil eder ve uygunsuz bir regresyon çizgisini işaret edebilir. Bu tür yanlış veri değerleriyle tahmin edilmek istenen doğruluk önemli ölçüde azaltılabilir. Doğrusal regresyon her vakit girdi değişkeni ve çıktı değişkeni ile arasında doğrusal bir ilişki olduğunu varsayar ve bu daima doğru değildir. Doğrusal regresyon, doğrusal olmayan veriler için en iyi uyumu sağlamaz. Son olarak, doğrusal regresyon modelleri sıklıkla en küçük kareler yaklaşımı kullanılarak uyarlanmaktadır.

2.3.3. Rastgele Orman

Bu regresyon, RF sınıflandırıcısına dayalı bir tahmin ağaçları koleksiyonudur. Her ağaç rastgele ormandaki diğer ağaçlara benzer bir dağılıma sahiptir ve bağımsız olarak örneklenen rastgele vektörlere bağlıdır. Rastgele orman, davranış analizi ve tahminlerinde kullanılan RF tabanlı bir modelleme tekniğidir. Rastgele orman veri girişinin farklı bir sınıflandırma örneğini temsil eden birkaç karar ağacı içerir. Rastgele orman tekniği örnekleri tek tek ele alır ve en çok oy alan örneği seçilen tahmin olarak alır. Sınıflandırmalardaki her ağaç, girdilerini ilk veri kümesindeki örneklerden alır. Ardından, her düğümde ağacı büyütmek için rastgele seçilen özellikler kullanılır. Ormandaki ağaçların her biri, tahmine kesin olarak ulaşılan kadar alıştırmanın sonuna kadar budanmamalıdır. Bu şekilde rastgele orman, zayıf korelasyonlara sahip herhangi bir sınıflandırıcının güçlü bir sınıflandırıcı olmasını sağlar. Rastgele orman tekniği, binlerce değişken içeren büyük verileri de işleyebilir. Bir sınıf, veriler diğer sınıflardan daha az seyrek olduğunda veri kümelerini otomatik olarak dengeleyebilir (Kocaoğlu vd., 2022).

2.4. DERİN ÖĞRENME

Derin öğrenme, beyin yapısından ve yapay sinir ağları adı verilen özelliklerden ilham alan bir tür makine öğrenme algoritmasıdır. Yapay nöronlar da biyolojik nöronlar gibi giriş sinyallerini alır, bu sinyaller toplanır, işlenir ve çıkışlara iletilir. Özellikle günümüzde internetin ileri seviyelere gitmesiyle birlikte dijital ortamda her saniye büyük miktarlarda milyonlarca veri üretilmekte ve depolanmaktadır. Eğitim için kullanılan veri miktarı arttıkça klasik makine öğrenmesi yöntemlerinin performansı büyük verileri işlemede yetersiz hale gelmiştir. Makine öğrenmesi yöntemleri hızlı olmasına ve çabuk iyileşme sağlamasına rağmen günümüzde internet kullanımının yaygınlaşmasıyla elde edilebilecek verinin boyutu da artmıştır. Hızlı bir sonuç elde etmekten ziyade, elde edilen sonucun performans değerinin daha iyi olması önemli hale gelmiştir. Derin öğrenme sistemleri bu büyük veriyi verimli bir şekilde kullanarak etkili sonuçlar elde edebilir. Diğer alanların aksine derin öğrenme, sınıflandırma işlevlerini ve görevlerini doğrudan verilerden gerçekleştirmeyi öğrenir. Bazı modeller, ilgili özelliklerin

ıkarılmasına gerek kalmadan doęrudan eęitim gerekleřtirebilir. Derin ęrenme modelleri, insan ęrenme seviyelerine ulařmak ve hatta bu seviyeleri ařmak iin eřitli ve ok sayıda ok sinirli aę mimarileri kullanarak eęitilebilir (řıřmanoęlu vd., 2020).

2.4.1. Uzun Kısa Sreli Bellek

Sinir aęı mimarileri 1990'lı yıllardan sonra daha sık kullanılmaya bařlanmıřtır. Bu mimarilerden ikisi dięerlerine gre ne ıkmaktadır. Bunlardan ilki, 1991 yılında Hochreiter tarafından geliřtirilen Tekrarlayan Sinir Aęlarıdır. Bu aęlar bilginin devamını saęlayan dngler ierir ve bu dnglerin standart bir sinir aęından hibir farkı yoktur. RNN'lerin aynı aęın birden fazla kopyası olduęu ve her birinin dięerine mesaj gnderdięi sylenebilir. RNN, doęal dil iřleme, konuřma tanıma, grnt sınıflandırma ve eviri gibi birok alıřmada bařarıyla kullanılmıřtır. Ancak bazı durumlarda bu aęların ęrenme yeteneęinin sınırlı olduęu grlmřtr. Bazen zorlu bir grevin stesinden gelmek iin sadece en son bilgilere bakmak yeterli olmayabilir ve daha fazla ierięe ihtiya duyulan durumlarda da yeterli olmayabilir. Standart RNN sinir aęlarının eęitimindeki bu sorunlar, 1997 yılında Hochreiter ve Schmidhuber tarafından geliřtirilen uzun kısa sreli bellek (LSTM) ile ařılmıřtır. Bunu, aęa bellek hcreleri ve eřitli kapılar ekleyerek bařarmıřtır. LSTM, birok grev iin standart versiyondan ok daha iyi alıřan ve uzun vadeli baęımlılıkları ęrenebilen, ok zel bir tekrarlayan sinir aęı trdr. Uzun sre bilgiyi hatırlama davranıřı sergilerler. Bu mimari, bellek geiř mekanizmasıyla uzun vadeli baęımlılıkları ęrenebildięi iin konuřma tanıma, doęal dil iřleme, mzik retimi, duygu sınıflandırma gibi zamansal dizi modelleme problemlerinde yaygın olarak kullanılmaktadır. Ancak RNN'den farklı olarak temel LSTM mimarisi, bir veya daha fazla bellek hcresine sahip tek bir kapı yerine giriř kapısı, ıkıř kapısı ve unutma kapısı gibi 3 kapıya sahiptir (Hızlısoy, 2020).

2.4.2. Yapay Sinir Aęları

Yapay sinir aęı (YSA), insan beyni gibi biyolojik sinir sistemlerinin modellenmesiyle geliřtirilen bir tekniktir. Byk veri tabanlarına dayalı tahminler yapma konusunda mkemmelen bir yeteneęe sahiptir. Borsa tahminlerinde yaygın olarak kullanılan YSA, geriye yayılım algoritmasına dayanmaktadır. Geri yayılım algoritmasına dayalı olarak geliřtirilen ok katmanlı algılayıcı sinir aęı kullanılmaktadır. Giriř dęmleri olarak bir dizi anlamsal dęm ieren bir giriř katmanından, bir veya daha fazla gizli hesaplama dęm katmanından ve hesaplama dęmlerinden oluřan bir ıktı katmanından oluřur (Tsai & Wang).

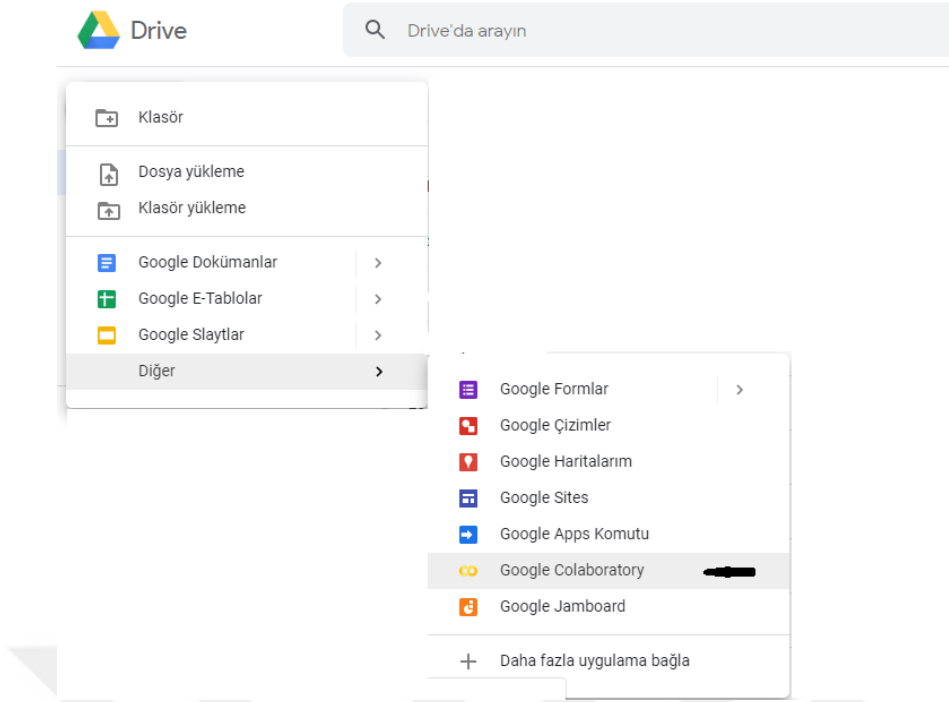
2.5. PYHTON

Python programlama dili, genel amaçlı, üst düzey bir programlama dilidir. 1990'ların sonlarında Guido van Rossum tarafından ortaya çıkmış ve geliştirilmiştir. Python, basit ve açık söz dizimi ile bilinir ve çok çeşitli uygulamalar için kullanılabilir. Python kullanılarak çeşitli programlar geliştirilebilir. Bunlar arasında Web tasarımları, veritabanı sistemleri, bilimsel hesaplama, oyunlar, yapay zekâ ve daha birçok alanda kullanılmaktadır. Python dilini kullanmanın birden fazla avantajı vardır. Basit ve açık söz dizimi nedeniyle okunması ve öğrenilmesi kolaydır. Bunun yanında çok sayıda içerisinde barındırdığı kütüphane ve modüller ile kullanım sürecini hızlandırmaktadır. Python açık kaynak kodlu bir yazılım dili olduğundan, yardımlara ve belgelere ulaşım daha kolaydır. Diğer karmaşık programlama dilleri, geniş uygulama yelpazesi nedeniyle öğrenilmesi uzun zaman alır ve ustalaşması oldukça zordur. Ancak Python program söz dizimi hem okunması kolay hem de umut vericidir. Kararlı programlama dili nedeniyle öğrenilmesi kolaydır. Ayrıca yeni başlayanlar için ideal bir seçim olarak öne çıkar. Sonuç olarak, Python kullanarak program geliştirmede yeni bir adım atan herkes hızlı ve kolay bir şekilde ilerleme kaydedebilmektedir. Diğer karmaşık dillere göre sadeliğin ön planda olduğu bir dildir (Bakdemir, 2024).

2.5.1. Google Colab

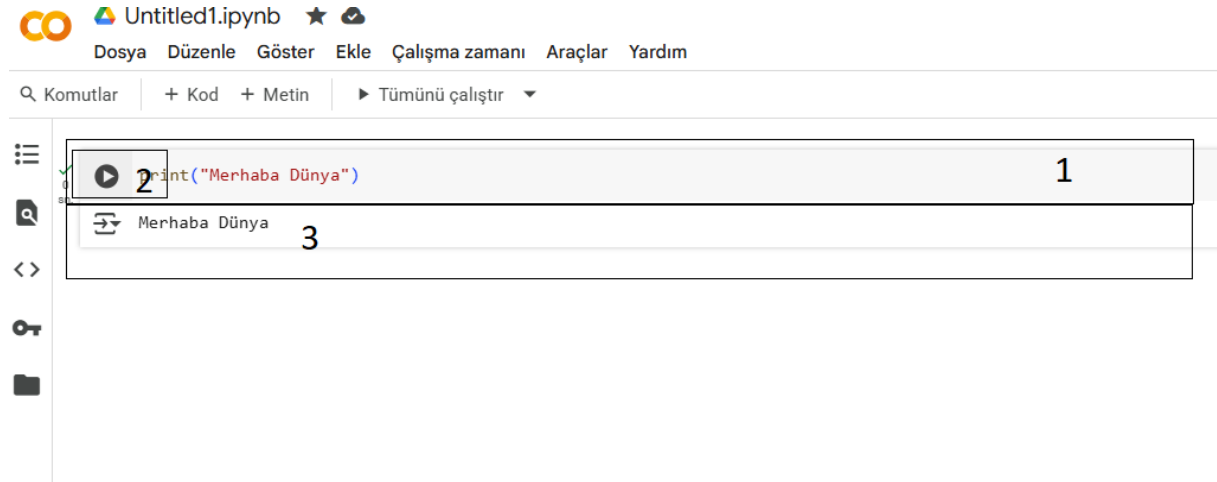
Google Colab, bilgisayarınızda yerel kurulumla gerek kalmadan Jupyter Notebook ortamında Python kodu yazmanıza ve çalıştırmanıza olanak tanıyan, tarayıcı tabanlı ücretsiz bir çevrimiçi platformdur. Python tabanlı projelerde, özellikle makine öğrenimi, derin öğrenme, istatistiksel analiz ve veri biliminde yaygın olarak kullanılmaktadır. Google Colab hem oturum açarak hem de oturum açmadan kullanılabilir. Oturum açıldığı zaman yapılan çalışmalar Google Drive hesabına kaydedilmekte ve daha sonra kullanıcı tarafından kendi hesabından çalışmayı açıp derleyebilir.

Google Drive hesabına giriş yapan kullanıcı Şekil 1.'de gösterildiği gibi colab sayfasını açıp kod yazmaya başlayabilir.



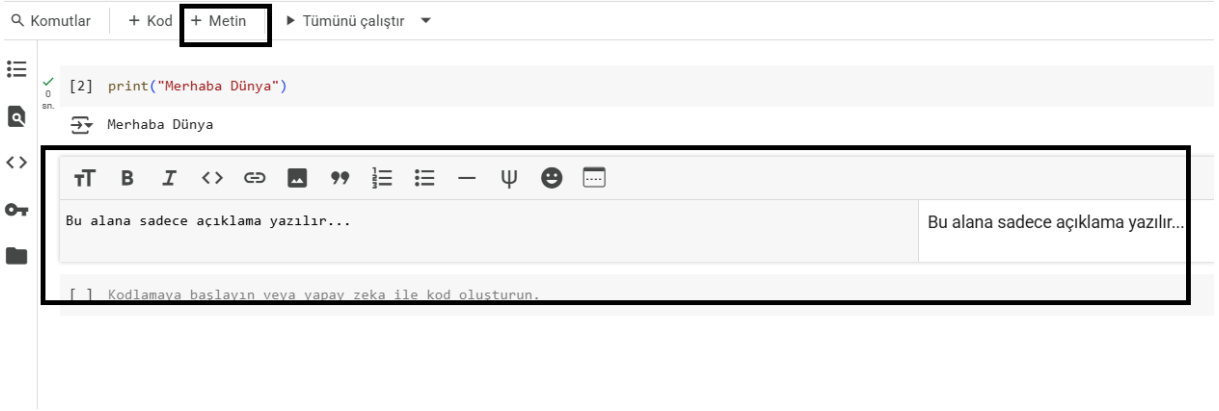
Şekil 1: Google Drive ile Colab Açılması

Colab sayfası açıldıktan sonra aşağıda Şekil 2.'de gibi bir sayfa kullanıcıyı karşılamaktadır. Bu sayfada 1 ile gösterilen alana kod yazmaya başlanılır. Yazılan kodun çalışması için 2 ile gösterilen kısımda bulunan oynatma tuşuna basılarak yazılan kodun çalışması sağlanır. Kodun çalıştıktan sonraki çıktısı 3 ile işaretlenen alanda görünmektedir.

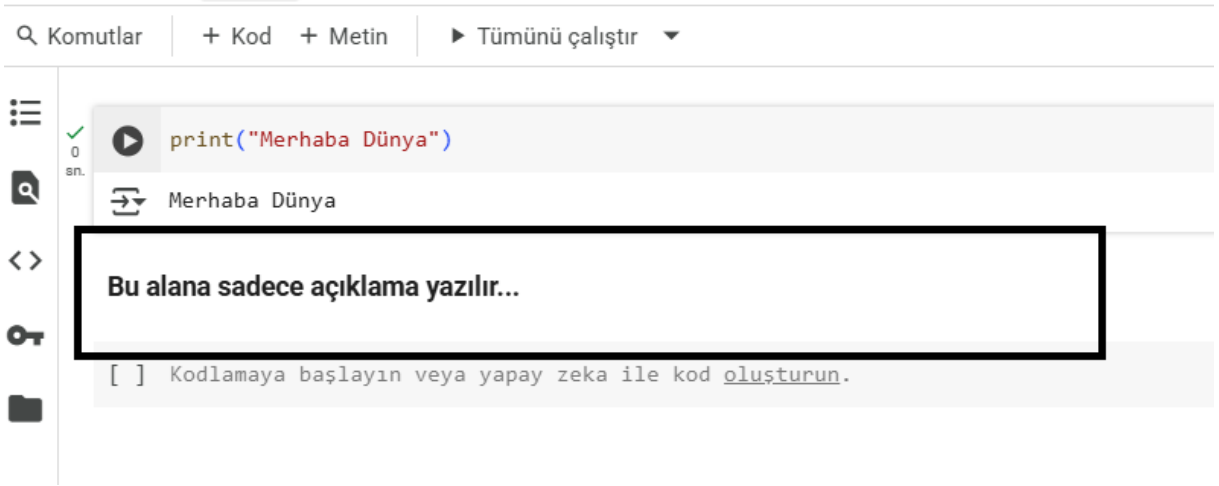


Şekil 2: Colab Kod Alanı

Eğer çalışma alanına kod değil de açıklama yazılacaksa “metin” yazılan yere tıklanır ve aşağıya sadece açıklama yazılacak bir alan açılır. İstenilen açıklamanın font bilgileri değiştirilebilir ve daha sonra açıklama yeniden düzenlemesi yapılabilir. Aşağıda Şekil 3.'de açıklama yazılacak alan gösterilmiştir. Şekil 4.'de ise açıklamanın ekran çıktısı gösterilmiştir.



Şekil 3: Colab Açıklama Alanı



Şekil 4: Colab Açıklama Alanı Ekran Çıktısı

2.5.2. Pandas

Veri analizini ve veri ön işlemeyi basitleştiren açık kaynaklı bir literatür kütüphanesidir. Python dili desteklenir ve pandalar dağıtılmış işleme için uygun değildir. Pandas veri yapıları için çok kullanışlıdır. Tartışılan nesnelerin çoğu veri çerçeveleridir. NumPy'deki veri yapısı sayesinde SQL, Excel gibi ilgili veri yapılarını işleyebilir ve endeksleyebilir (Sevim, 2019).

2.5.3. Numpy

Python kullanarak bilimsel değerlendirmeler yapabilen bir kütüphanedir. NumPy ile Python dili kullanılarak matrislerde ve dizilerde fonksiyonlar kolayca oluşturulabilir. Ek olarak, kütüphane, N boyutlu bir dizinin nesneleriyle işlemleri uygular, kütüphanenin kendisi için benzersiz olan işlevleri içerir (Fourier dönüşümleri, rastgele sayıların düzenli dağılımları,

doğrusal cebir işlemleri, sayıların düzensiz dağılımları, vb.) C/C++ ve fortran kodları yaratma yeteneği gibi beceriler görülebilir (Sevim, 2019).

2.5.4. Matplotlib

Python’da yapılan veri hesaplamalarının görselleştirmesinde kullanılan bir kütüphanedir. Genel olarak 2 boyutlu çizimler için kullanılır. 3 boyutlu ve daha fazla çizimler için başka kütüphaneler kullanılır.



ÜÇÜNCÜ BÖLÜM

3. UYGULAMA

3.1. VERİ SETİNİ HAZIRLAMA

3.1.1. Gerekli Kütüphanelerin Yüklenmesi

Bu aşamada çalışma için gerekli olan Python kütüphaneleri yüklenmiştir. Veri setinin elde edilmesi için yahoofinance, matematiksel hesaplamalar için numpy, veri analizi için pandas, verilerin görselleştirilmesi için matplotlib kütüphaneleri, eğitim ve test verisi için scikit-learn kütüphanesinin train_test_split fonksiyonu, performans değerlendirme ölçütleri için sklearn.metrics ve veriyi normalleştirme için MinMaxScaler kütüphaneleri yüklenmiştir. Aşağıda Şekil 5.'de gösterilmiştir.

```
import os
import random
import numpy as np
import pandas as pd
import yfinance as yf
import matplotlib.pyplot as plt

from sklearn.linear_model import LinearRegression
from sklearn.ensemble import RandomForestRegressor
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Dropout, LSTM
import tensorflow as tf
```

Şekil 5: Python Kütüphanelerinin Yüklenmesi

3.1.2. Veri Setinin Eklenmesi

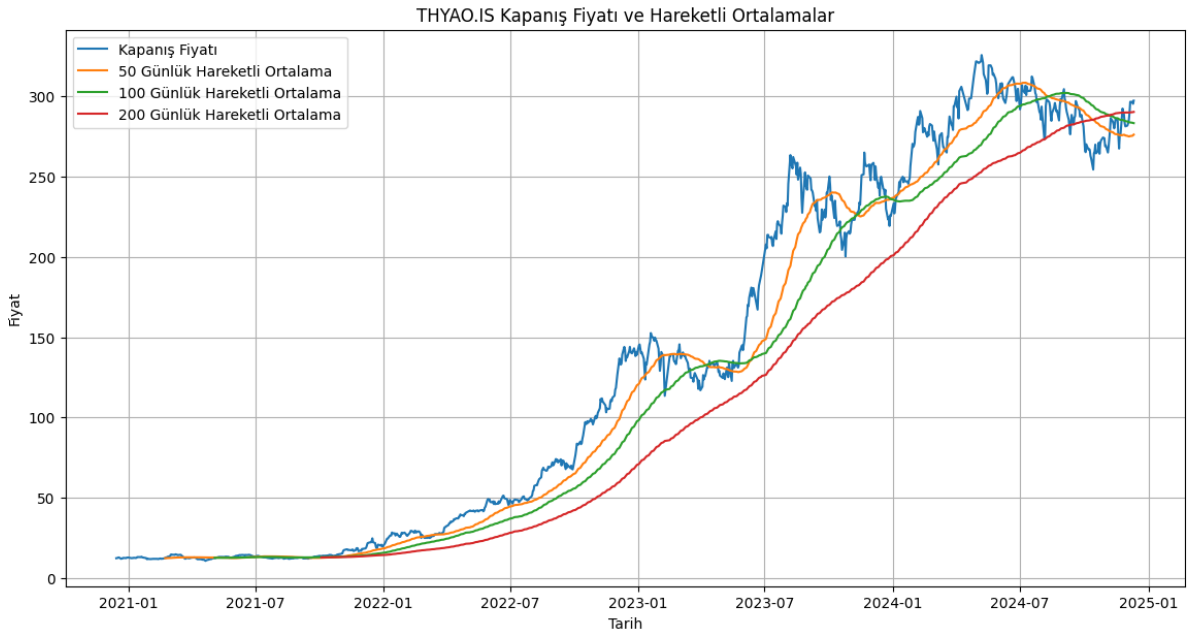
Bu aşamada yahoofinance'dan veri seti çekimi yapılmıştır. Etiket değişkeni ile hisse senedinin borsadaki hisse senedi kodu yazılmış, başlangıç ve bitiş tarihleri yazılmıştır. Veri setinde hisse senedinin açılış, kapanış, en yüksek değeri, en düşük değeri ve işlem hacmi görülmektedir. Veri seti Şekil 6.'da gösterilmiştir.

```
# Veri çekimi
start_date = "2020-12-12"
end_date = "2024-12-12"
df = yf.download("THYAO.IS", start=start_date, end=end_date)
```

Şekil 6: Gerçek Zamanlı Veri Setinin Yüklenmesi

3.1.3. Veri Ön İşleme

Veri setinde hisse senedinin kapanış fiyatlarını üzerinde işlem yapılacaktır. Hisse senedi kapanış fiyatlarının 200, 100 ve 50 günlük hareketli ortalamaları alınmış ve boş değerler veri setinden kaldırılmıştır. Şekil 7.'de hisse senedinin kapanış fiyatı ve 200, 100 ve 50 günlük hareketli ortalamalar arasındaki ilişkiyi gösteren grafik gösterilmiştir.



Şekil 7: Hisse Senedi Kapanış Fiyatı ile 200, 100 ve 50 Günlük Hareketli Ortalamaları

3.1.4. Özellik ve Hedef Değişken Tanımlanması

Bu aşamada özellik değişkeni ve hedef değişkeni olarak adlandırılan iki parametre tanımlanacaktır. Şekil 8.'de gösterildiği $X = df.drop('Close', axis=1)$ satırı veri setindeki kapanış fiyatları özelliklerini seçer. Seçilen bu özellikler modellerin eğitimde kullanılacak girdileri oluşturur. $Y = df['Close']$ satırı ise hedef değişken olarak kullanılacak kapanış fiyatlarını tanımlar.

```
# Özellikler ve hedef
features = ['Close', 'MA50', 'MA100', 'MA200']
X = df[features]
y = df['Target'].values.reshape(-1, 1)

scalerX = MinMaxScaler()
scalerY = MinMaxScaler()
X_scaled = scalerX.fit_transform(X)
y_scaled = scalerY.fit_transform(y)
```

Şekil 8: Özellik ve Hedef Değişken Tanımlaması

3.1.5. Veri Setinin Eğitim ve Test Kümelerine Ayrılma

Veri setindeki verilerin modellerde çalıştırılması için eğitim ve test kümelerine ayırma işlemi yapılmıştır. Bu kod ile kullanılan veri setinin yüzde 80'i eğitim, yüzde 20'si test verisi olacak şekilde ayırma işlemi yapılmıştır. Python kodları olan if ve else şart koşulları ile veri setindeki verinin yetersiz olması durumunda kullanıcıya hata mesajı vermek amacıyla bir döngü içine alınmıştır. Bunun amacı projenin daha sağlıklı çalışması içindir. Aşağıda şekil 9.'da kod çalışması gösterilmiştir.

```
# Veri bölme
split = int(len(X_scaled) * 0.8)
X_train, X_test = X_scaled[:split], X_scaled[split:]
y_train, y_test = y_scaled[:split], y_scaled[split:]
y_test_orig = scalerY.inverse_transform(y_test)
```

Şekil 9: Verilerin, Eğitim ve Test Verileri Olarak Ayrılması

3.1.6. Verilerin Ölçeklendirilmesi

Bu alanda veri setindeki veriler ölçeklendirme işlemi için ele alınacaktır. MinMaxScaler fonksiyonu ile veriler için belirli bir aralığa (genellikle [0,1] aralığına) dönüştürme işlemi yapılmıştır. Ölçeklendirme yapılmasındaki amaç özelliklerin çok farklı aralıklara sahip olması ve bu durumun bazı algoritmalarda (KNN, YSA ve SVM algoritmaları gibi mesafeye sahip olanlar) yanlış sonuçların üretilmesine sebep olabilmesindendir. Ölçeklendirme sayesinde derin öğrenme ve doğrusal model kullanımında özelliklerin benzer aralıklarda olması çalışılan modellerin performansının artmasına ve daha hızlı öğrenmesine yardımcı olacaktır.

3.2. MODELLERİN OLUŞTURULMASI

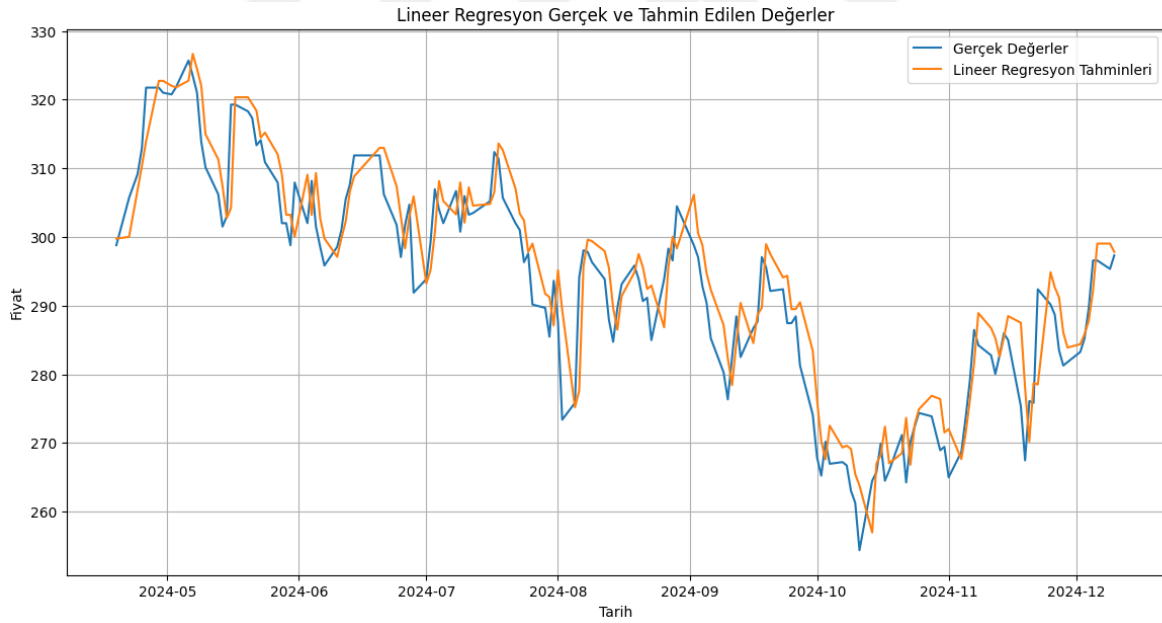
3.2.1. Lineer Regresyon Modeli

Kullanılan modelde öncelikle gerekli olan `sklearn.linear_model` kütüphanesi yüklenmiştir. Modelin performansını değerlendirmek için MEA, MSE, RMSE ve R^2 performans değerlendirme ölçütleri eklenmiştir. Sonrasında model eğitilmiş ve test verisi ile tahmini yapılmıştır. Tablo 1.'de eğitilen ve tahmini yapıldıktan sonra ortaya çıkan LR modeli performans değerlendirme ölçütleri sonuçları gösterilmiştir.

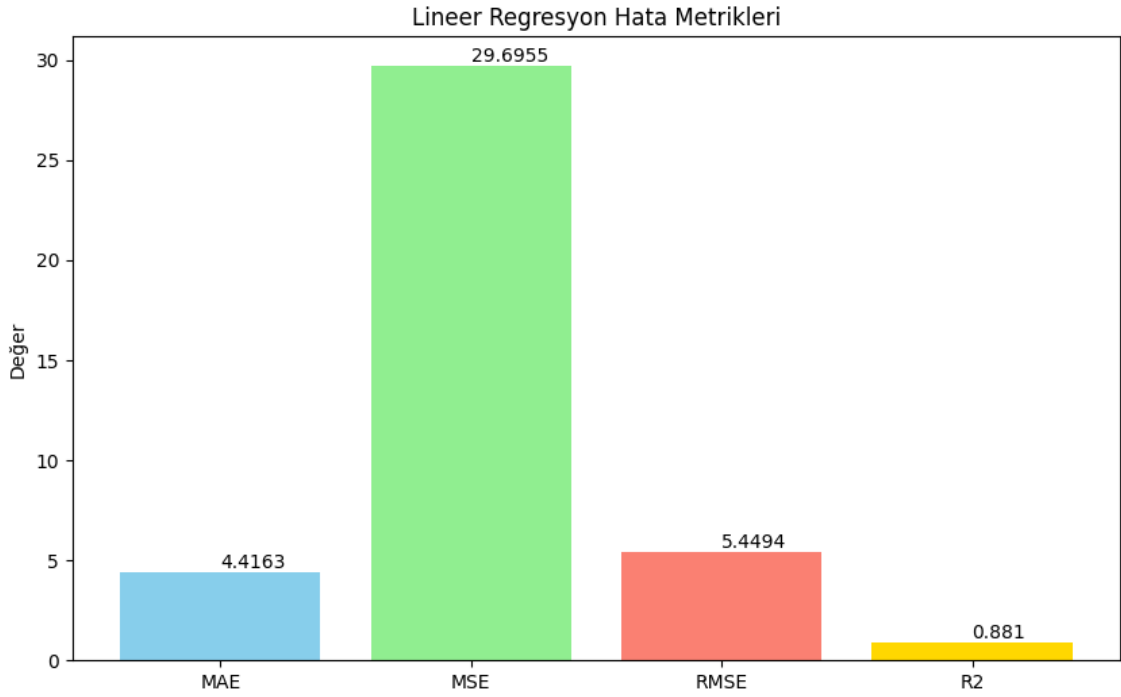
Tablo 1: LR Modeli Performans Değerlendirme Ölçütleri Sonuçları

LR MAE	LR MSE	LR RMSE	LR R^2
4.4163	29.6955	5.4494	0.8810

Şekil 10.'da gerçek değer ve lineer regresyon ile tahmin edilen değeri karşılaştıran çizgi grafiği ve Şekil 11.'de ise performans değerlendirme ölçütleri sonuçları mum grafiği gösterilmiştir.



Şekil 10: LR Modeli ile Tahmin Edilen Değer ve Gerçek Değer



Şekil 11: LR Modeli İçin Performans Değerlendirme Ölçütleri Sonuçları

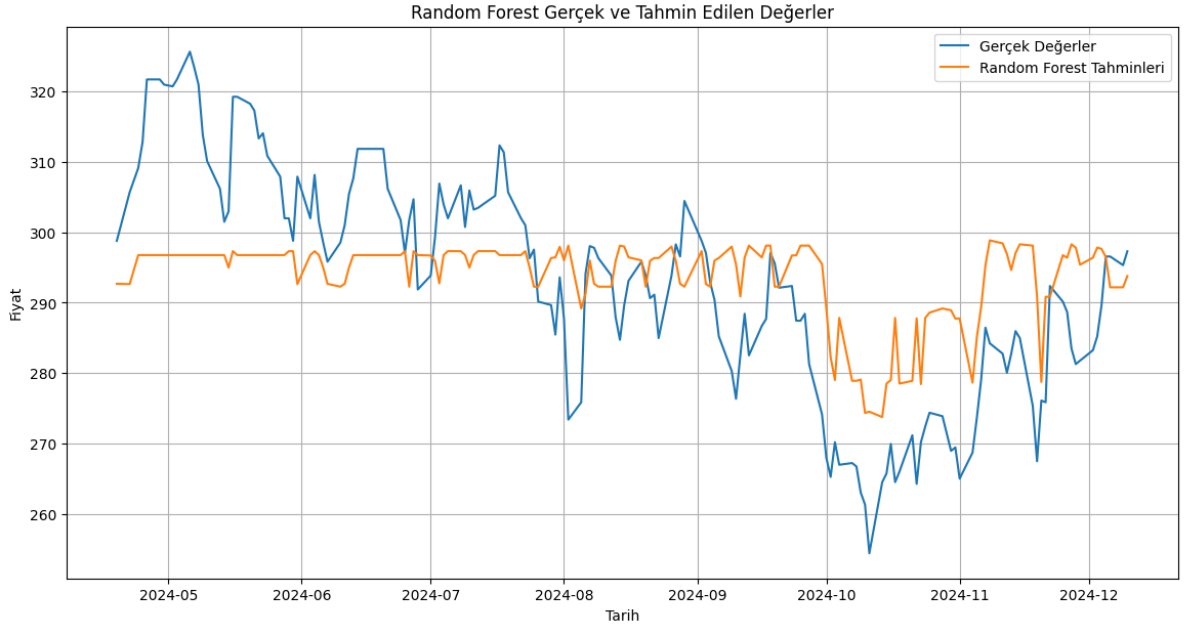
3.2.2. Rastgele Orman Modeli

Kullanılan modelde öncelikle gerekli olan `sklearn.linear_model` kütüphanesi yüklenmiştir. Modelin performansını değerlendirmek için MEA, MSE, RMSE ve R^2 ise performans değerlendirme ölçütleri eklenmiştir. Sonrasında model eğitilmiş ve test verisi ile tahmini yapılmıştır. Tablo 2.'de eğitilen ve tahmini yapıldıktan sonra ortaya çıkan RF modeli performans değerlendirme ölçütleri sonuçları gösterilmiştir.

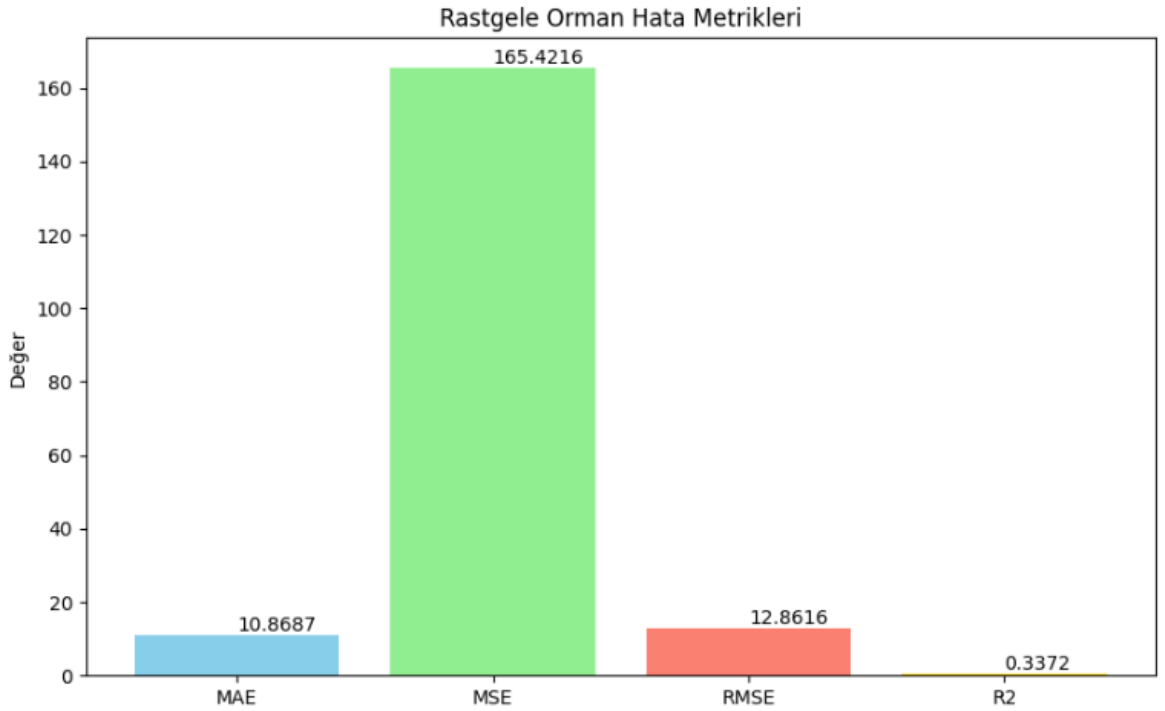
Tablo 2: RF Modeli Performans Değerlendirme Ölçütleri Sonuçları

RF MAE	RF MSE	RF RMSE	RF R^2
10.8687	165.4216	12.8616	0.3372

Şekil 12'de gerçek değer ve lineer regresyon ile tahmin edilen değeri karşılaştıran çizgi grafiği ve Şekil 13.'de ise performans değerlendirme ölçütleri sonuçları mum grafiği gösterilmiştir.



Şekil 12: RF Modeli ile Tahmin Edilen Değer ve Gerçek Değer



Şekil 13: RF Modeli İçin Performans Değerlendirme Ölçütleri Sonuçları

3.2.3. LSTM Modeli

Bu model giriş katmanı, hafıza katmanı ve çıkış katmanı olarak üç temel katmandan oluşmaktadır. Şekil 14.'de gösterilen komutlar açıklanmıştır.

```

# LSTM veri hazırlığı
def make_seq(X, y, w=10):
    Xs, ys = [], []
    for i in range(w, len(X)):
        Xs.append(X[i-w:i])
        ys.append(y[i])
    return np.array(Xs), np.array(ys)

w = 10
X_seq, y_seq = make_seq(X_scaled, y_scaled, w)
Xtr, Xte = X_seq[:split-w], X_seq[split-w:]
ytr, yte = y_seq[:split-w], y_seq[split-w:]

# LSTM
lstm = Sequential([
    LSTM(50, input_shape=(w, X_scaled.shape[1])),
    Dropout(0.2),
    Dense(1)
])
lstm.compile(optimizer='adam', loss='mse')
lstm.fit(Xtr, ytr, epochs=100, batch_size=32, verbose=0, validation_split=0.1)
lstm_res = scalerY.inverse_transform(lstm.predict(Xte))
lstm_res = evaluate(y_test_orig, lstm_res, "LSTM")

```

Şekil 14: LSTM Model Kodu

Öncelikle LSTM modelinin çalışması için gerekli kütüphaneler yüklenmiştir. Daha sonra LSTM modeli oluşturulmuştur. Yukarıda belirttiği gibi çalışması için model üç boyutlu hale getirilmiştir. Sonraki aşamada model eğitime geçilmiştir. Tahmin verileri ile tahmin yapıp performans değerlendirme ölçütleri ile sonuçlar oluşturulmuştur.

Epochs komutu modelin eğitim veri kümesi boyunca kaç defa yeniden eğitileceğini göstermektedir. Her dönem bir tahmin ve hata hesaplama ile optimizasyon içermektedir.

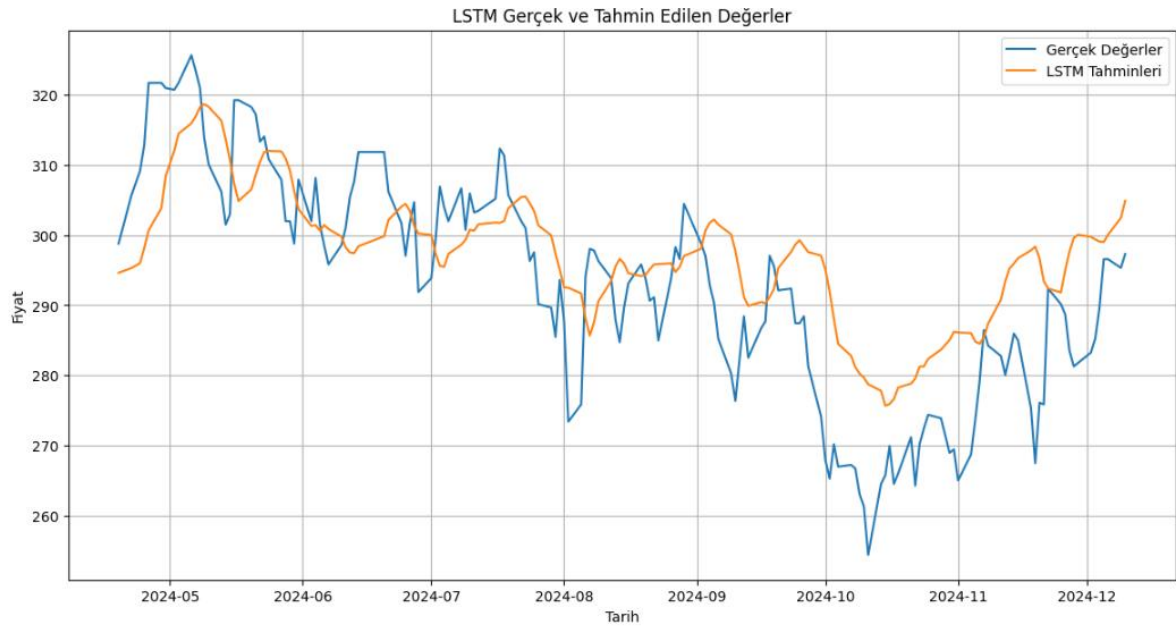
Eğitim sırasında, model ağırlıklarını güncelleme yapmadan önce her partide kaç örneğin aynı anda işlenmesini sağlayan komut batch_size komutudur.

Tablo 3'te eğitilen ve tahmini yapıldıktan sonra ortaya çıkan LSTM modeli performans değerlendirme ölçütleri sonuçları gösterilmiştir.

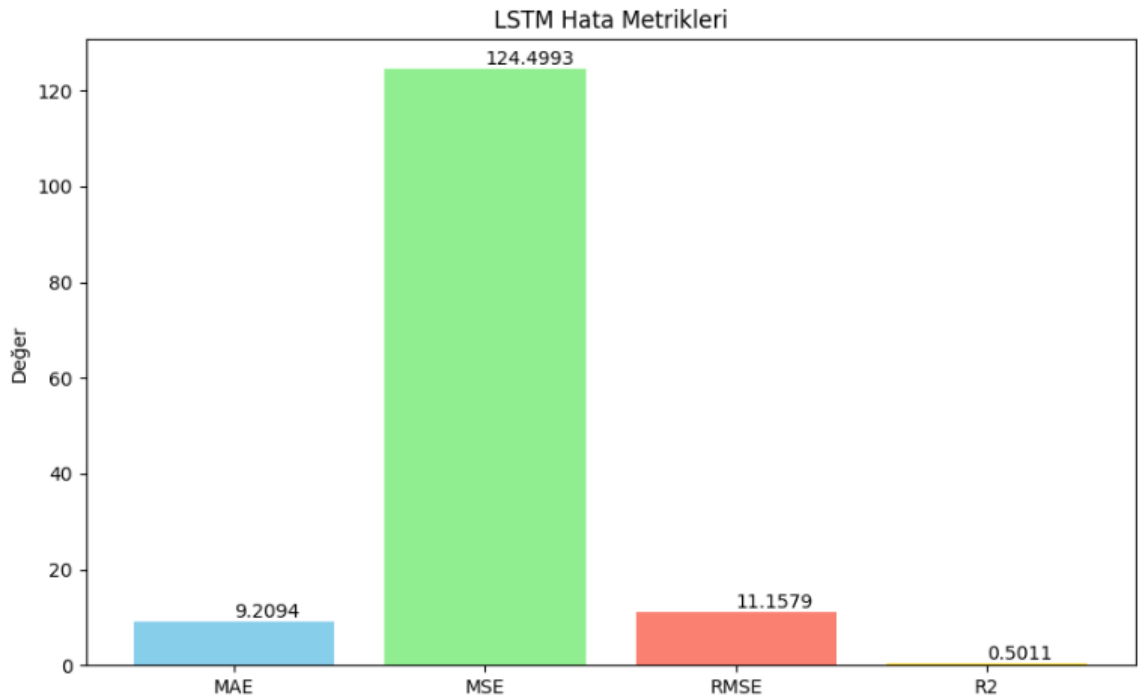
Tablo 3: LSTM Modeli Performans Değerlendirme Ölçütleri Sonuçları

LSTM MAE	LSTM MSE	LSTM RMSE	LSTM R ²
9.2094	124.4993	11.1579	0.5011

Şekil 15.'de eğitimi yapıldıktan sonra tahmin sonuçları açıklanan ve gerçek değer ile LSTM modeline göre ortaya çıkan grafik gösterilmiştir. Şekil 16.'da ise performans değerlendirme ölçütleri gösterilmiştir.



Şekil 15: LSTM Modeli ile Tahmin Edilen Değer ve Gerçek Değer



Şekil 16: LSTM Modeli İçin Performans Değerlendirme Ölçütleri Sonuçları

3.2.4. Yapay Sinir Ağı Modeli

Öncelikle YSA modelinin çalışması için gerekli kütüphaneler yüklenmiştir. Daha sonra YSA modeli oluşturulmuştur. Sonraki aşamada model eğitime geçilmiştir. Tahmin verileri ile tahmin yapıp performans değerlendirme ölçütleri ile sonuçlar oluşturulmuştur. Şekil 17.'de YSA modelinin çalışması için oluşturulan kod satırları gösterilmiştir.

```
# YSA
ann = Sequential([
    Dense(64, activation='relu', input_shape=(X_train.shape[1],)),
    Dropout(0.2),
    Dense(32, activation='relu'),
    Dropout(0.2),
    Dense(1)
])
ann.compile(optimizer='adam', loss='mse')
ann.fit(X_train, y_train, epochs=100, batch_size=32, verbose=0, validation_split=0.1)
ann_pred = scalerY.inverse_transform(ann.predict(X_test))
ann_res = evaluate(y_test_orig, ann_pred, "ANN")
```

Şekil 17: YSA Model Kodu

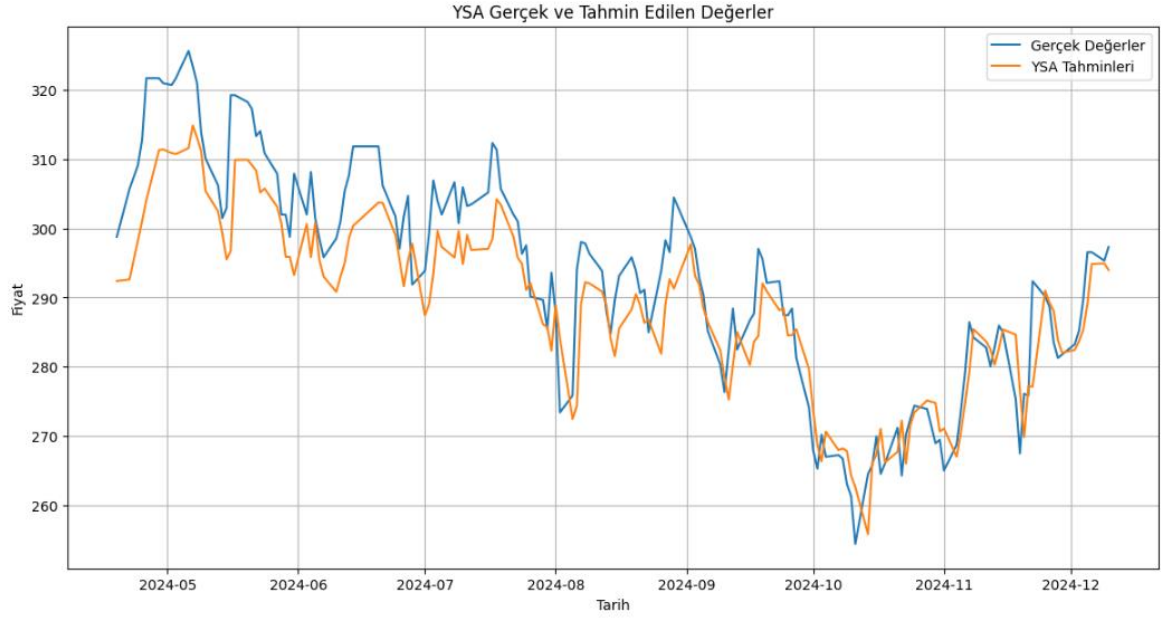
Epochs komutu modelin eğitim veri kümesi boyunca kaç defa yeniden eğitileceğini göstermektedir. Her dönem bir tahmin ve hata hesaplama ile optimizasyon içermektedir. Eğitim sırasında, model ağırlıklarını güncelleme yapmadan önce her partide kaç örneğin aynı anda işlenmesini sağlayan komut batch_size komutudur.

Tablo 4.'de eğitilen ve tahmini yapıldıktan sonra ortaya çıkan YSA modeli performans değerlendirme ölçütleri sonuçları gösterilmiştir.

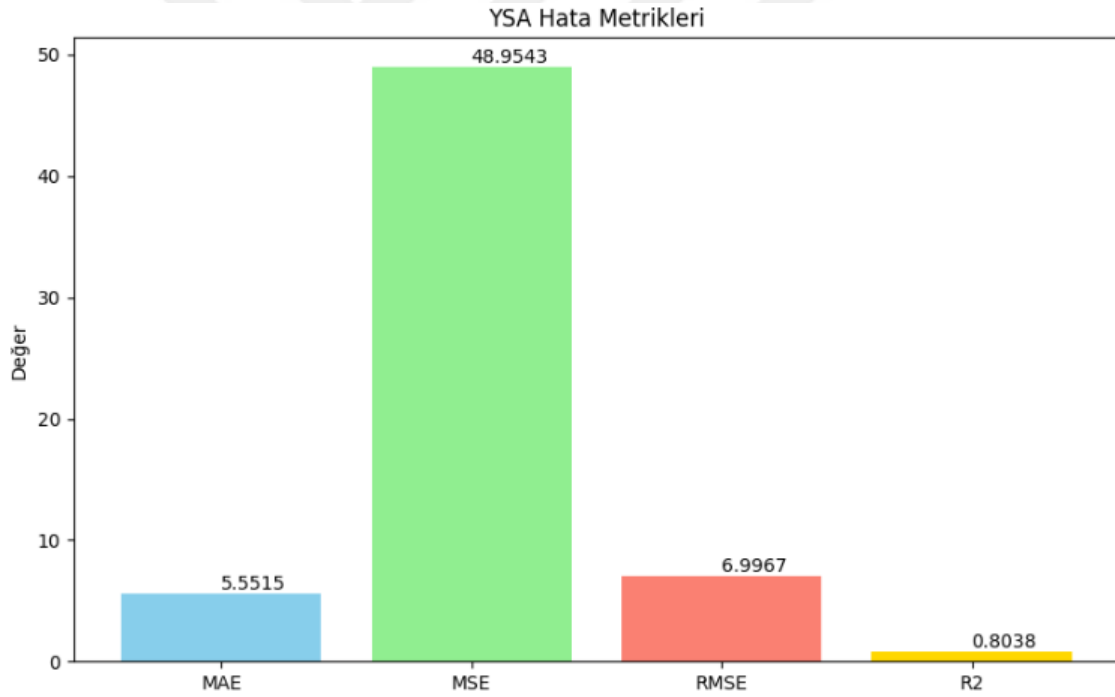
Tablo 4: YSA Modeli Performans Değerlendirme Ölçütleri Sonuçları

YSA MAE	YSA MSE	YSA RMSE	YSA R ²
5.5515	48.9543	6.9967	0.8038

Şekil 18.'de eğitim yapıldıktan sonra tahmin sonuçları açıklanan ve gerçek değer ile YSA modeline göre ortaya çıkan değer grafik olarak gösterilmiştir. Şekil 19.'da ise performans değerlendirme ölçütleri gösterilmiştir.



Şekil 18: YSA Modeli ile Tahmin Edilen Değer ve Gerçek Değer

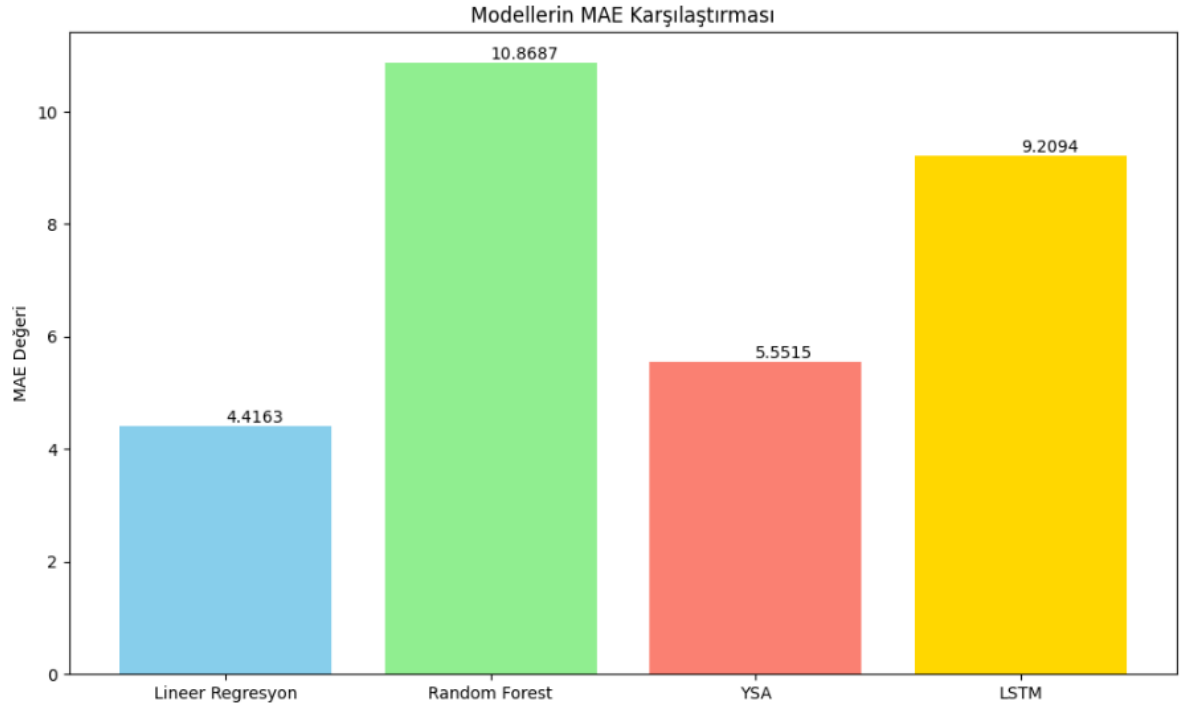


Şekil 19: YSA Modeli İçin Performans Değerlendirme Ölçütleri Sonuçları

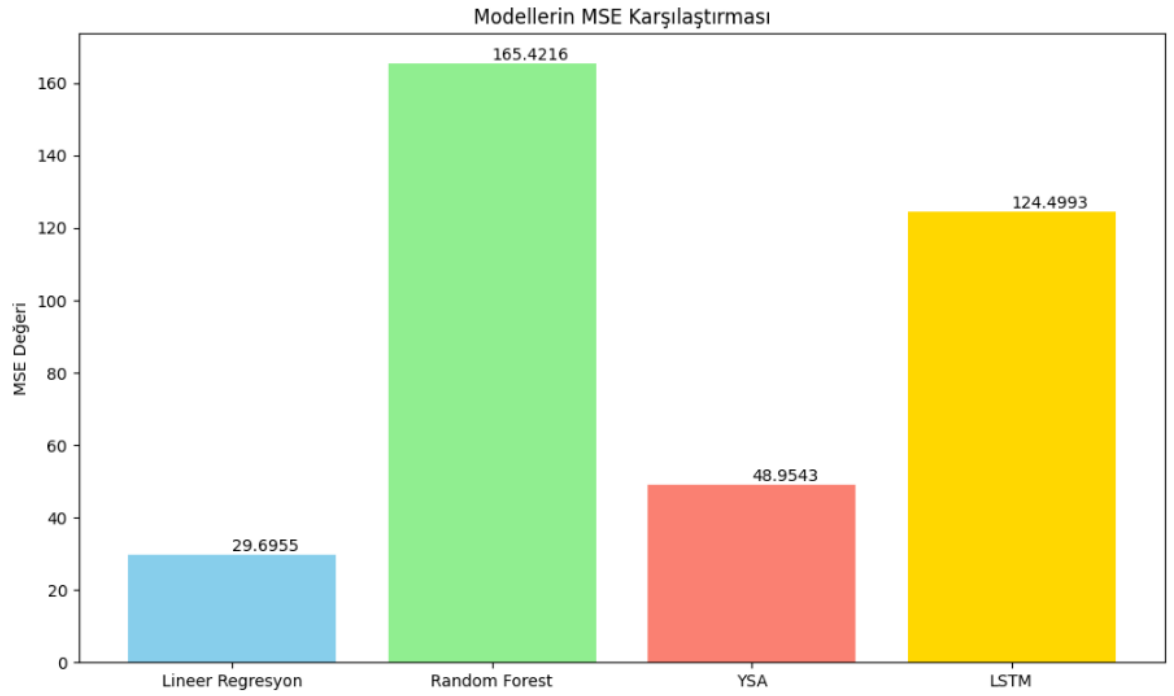
3.3. MODELLERİN DEĞERLENDİRİLMESİ

Çalışmada modeller eğitildikten ve tahmin sonuçları elde edildikten sonra, modellerin değerlendirilmesi için performans değerlendirme ölçütleri kullanılmıştır. Şekil 20.'de MAE performans değerlendirme ölçütlerine göre LR, RF, LSTM ve YSA modellerinin sonuçları, Şekil 21.'de MSE performans değerlendirme ölçütlerine göre LR, RF, LSTM ve YSA modellerinin sonuçları, Şekil 22.'de RMSE performans değerlendirme ölçütlerine göre LR, RF,

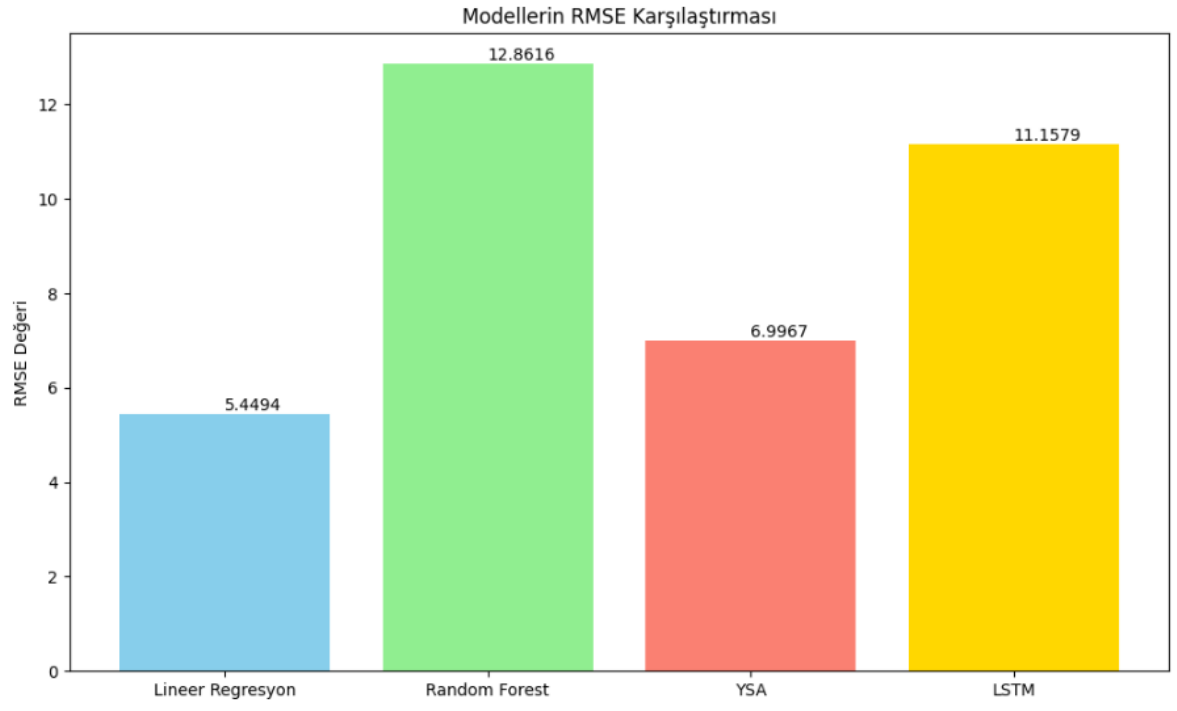
LSTM ve YSA modellerinin sonuçları ve Şekil 23.'de R^2 performans değerlendirme ölçütlerine göre LR, RF, LSTM ve YSA modellerinin sonuçları gösterilmiştir.



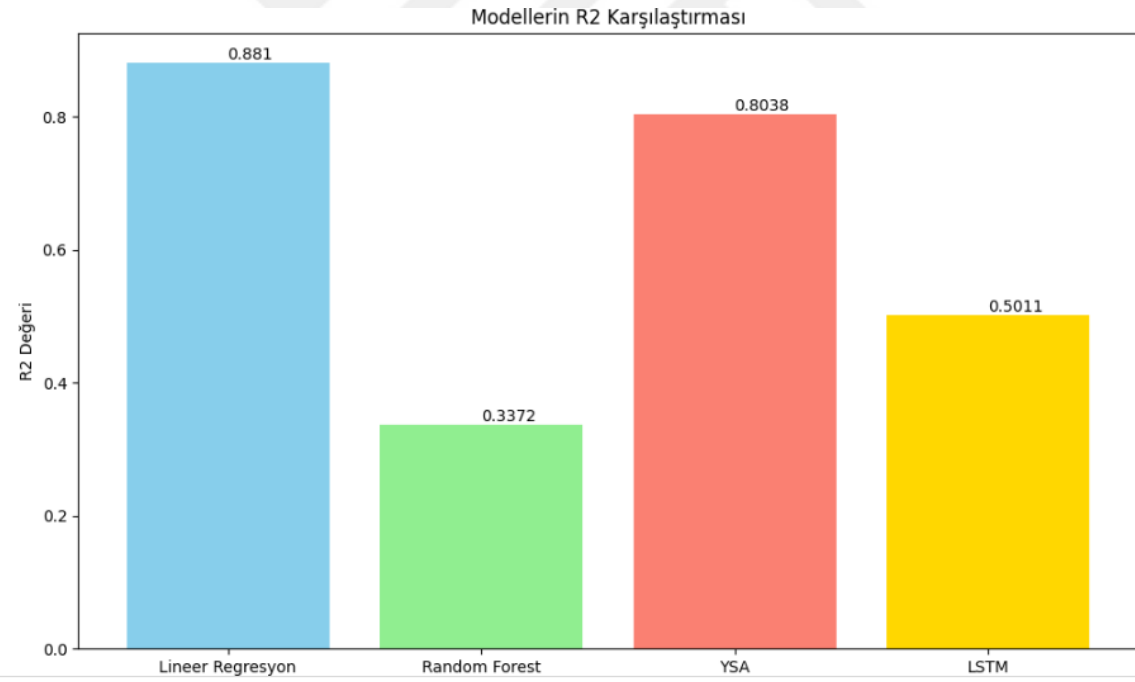
Şekil 20: Modellere Göre MAE Performans Değerlendirme Ölçütleri Karşılaştırması



Şekil 21: Modellere Göre MSE Performans Değerlendirme Ölçütleri Karşılaştırması



Şekil 22: Modellere Göre RMSE Performans Değerlendirme Ölçütleri Karşılaştırması



Şekil 23: Modellere Göre R² Performans Değerlendirme Ölçütleri Karşılaştırması

3.4. TAHMİN YAPILMASI

Çalışmanın sonucunda eğitilen modellerin performans değerlendirme ölçütleri ile değerlendirilmesi neticesinde en iyi modelin LR modeli olduğu ortaya çıkmıştır. Bundan dolayı yapılacak olan tahmin, veri setinin bilinen son tarihinden sonraki gün olan 12-12-2024 tarihindeki hisse senedi fiyatı olacaktır. Aşağıda, Şekil 24.'de tahmin için kullanılan kod bloğu ve sonucunda bulunan tahmin sonucunun ekran görüntüsü verilmiştir.

```
#Bir sonraki günü hisse senedi fiyat tahminini ver

def predict_next_day_price():
    """
    En iyi modeli kullanarak bir sonraki işlem gününün hisse senedi fiyatını tahmin eder.
    """
    # Son günün verilerini al
    last_day_data = X_scaled[-1].reshape(1, -1) # Lineer, RF, ANN için uygun şekil
    last_day_data_seq = X_scaled[-w:].reshape(1, w, X_scaled.shape[1]) # LSTM için uygun şekil

    # En iyi modeli kullanarak tahmin yap
    if best_model['Model'] == "LSTM":
        prediction_scaled = lstm.predict(last_day_data_seq)
    elif best_model['Model'] == "Random Forest":
        prediction_scaled = rf.predict(last_day_data).reshape(-1, 1)
    elif best_model['Model'] == "ANN":
        prediction_scaled = ann.predict(last_day_data)
    elif best_model['Model'] == "Lineer Regresyon":
        prediction_scaled = lr.predict(last_day_data)
    else:
        print("Uygun model bulunamadı.")
        return None

    # Tahmini orijinal ölçeğe geri dönüştür
    predicted_price = scalerY.inverse_transform(prediction_scaled)[0, 0]

    # Bir sonraki işlem gününün tarihini bul
    last_date = df.index[-1]
    next_day = last_date + pd.Timedelta(days=1)
    # Borsa kapalı olduğu günleri (hafta sonu) atla
    while next_day.weekday() > 4: # Pazartesi=0, Pazar=6
        next_day += pd.Timedelta(days=1)

    print(f"Bir sonraki işlem günü ({next_day.strftime('%Y-%m-%d')}) için {best_model['Model']} tahmini: {predicted_price:.2f} ₺")
    return predicted_price

# Bir sonraki günün fiyatını tahmin et
predict_next_day_price()

Bir sonraki işlem günü (2024-12-11) için Lineer Regresyon tahmini: 297.83 ₺
np.float64(297.8262640598796)
```

Şekil 24: Tahmin Kodu ve Sonucu

Çıkan tahmin sonucu, veri seti için alınan hisse senedinin 12-12-2024 tarihli hisse senedi fiyatına bakıldığında, sonuçların birbirine yakın olduğu gözlemlenmiştir.

DÖRDÜNCÜ BÖLÜM

4. BULGULAR

Bu bölümde, analiz sonuçlarını ve çalışmada örnek olarak alınan Türk Hava Yolları A.Ş. hisse senedi fiyatlarını tahmin etmek için kullanılan farklı modellerin performansları anlatılmıştır. Bu çalışmada değerlendirilen modeller arasında Doğrusal Regresyon, Rastgele Orman, LSTM ve YSA bulunmaktadır. Bu modeller, gelecekteki fiyatları tahmin etmek amacıyla Yahoo Finance'ten alınan geçmiş hisse senedi fiyat verileri kullanılarak eğitilmiş ve test edilmiştir. Bu modellerin performansı MAE, MSE, RMSE ve R^2 gibi temel ölçütler kullanılarak değerlendirilmiştir.

Lineer regresyon modeli hata oranları en düşük olan ve en iyi performansı gösteren modeldir. Diğer modellere göre çok daha iyi sonuçlar elde edilmiştir. R^2 değeri en yüksek model olup, bu değer modelin verilere uyumunu göstermektedir. Tahmin hataları da minimum düzeyde kalmıştır. RF, LSTM ve YSA modelleri LR modeline göre R^2 değerleriyle kıyaslandığında daha zayıf sonuçlar üretmiş, bu da modellerin verilere çok iyi uymadığını ve tahminlerinde büyük hatalar yaptığını göstermektedir. LSTM ve RF modelleri bu çalışma için LR modelinin gerisinde kalmış ve veri setinin şartlarını karşılayamamıştır. YSA modeli ise LR modeline en yakın model olduğu gözlemlenmiştir. LR modelinin başarılı olmasının sebebi veri setinin yapısına uygunluğunun yanı sıra eğitim kolaylığı ve hızına bağlıdır. RF modeli bir ağaç modeli olmasına rağmen belirli veri türlerine özel hiperparametre ayarlaması ve model optimizasyonu gerekebilir. LSTM ve YSA modellerinin başarılı olmamasının sebeplerinden bazıları veri setinin zaman sırasına göre sıralı fakat tam olarak bir zaman serisi olmamasından kaynaklanmaktadır. Bu modeller zamanla değişen ver sıralı veriler üzerinde daha verimli çalışabilir. Bu neden standart regresyon problemlerinde etkili olmayabilir.

Model geliştirme ve optimizasyon yapılması, veri setini zenginleştirilmesi ve farklı modeller kullanılarak farklı karşılaştırmalar yapılabilir. LSTM ve YSA için daha büyük veri setleri ele alınabilir. Veri ön işleme adımlarında 200, 100 ve 50 günlük hareketli ortalamaların alınması ve “NaN” değerler kaldırılması modellerin performansını etkileyebilmektedir.

SONUÇ, TARTIŞMA VE ÖNERİLER

Bu çalışmada Python yazılım dili kullanılarak makine öğrenmesi ve derin öğrenme yöntemleri kullanılarak bir hisse senedi fiyat tahmini yapılmıştır. Bunun için LR, RF, LSTM ve YSA'dan oluşan dört model oluşturulmuştur. Oluşturulan modeller MAE, MSE, RMSE ve R^2 olan performans değerlendirme ölçütleri ile model performansları değerlendirilmiştir. LR için MAE tahmin değeri 4,4163, MSE tahmin değeri 29,6955, RMSE tahmin değeri 5,4494 ve R^2 tahmin değeri 0,8810 olarak çıkmış ve en iyi tahminde bulunan model olmuştur. Diğer modellerin değerleri de Tablo 5'te gösterilmiştir.

Tablo 5: Modellere Göre Performans Değerlendirme Ölçütleri Sonuçları

Model	MAE	MSE	RMSE	R^2
LR	4,4163	29,6955	5,4494	0,8810
RF	10,8687	165,4216	12,8616	0,3372
LSTM	9,2094	124,4993	11,1579	0,5011
YSA	5,5515	48,9543	6,9967	0,8038

LR modelinin en iyi tahmin performansı göstermesinin sebebi, modelin basit yapısı ve doğrusal ilişkili verilere uyacak şekilde öğrenebilmesidir. Özellikle R^2 değeri (0,8810), ilgili modelin, veri setindeki varyansın çoğunu başarıyla açıklayabildiğini göstermektedir. Bu durum, mevcut verinin LR modeline uygun olduğunu göstermektedir. RF modeli ise beklenmedik derecede kötü sonuçlar üretmiştir. Bunun ise birkaç nedeni vardır. RF daha karmaşık ilişkileri öğrenme yeteneğine sahip olmasına rağmen hiperparametrelerin aşırı veya yanlış ayarlanması böyle bir modelin doğruluğunu azaltabilir. Negatif bir R^2 değeri, modelin eğitim verilerine gerçekte ne kadar zayıf uyduğunu göstermektedir. LSTM ve YSA modellerinin çalışmada çok kötü bir performans ortaya koymasındaki neden, bu modellerin zaman serisi verilerine veya karmaşık ilişkiler içinde olan veri setlerine gereksinim duymasından dolayıdır. Diğer bir deyişle, kullanılan veri seti, derin öğrenme modeli olan LSTM ve YSA modelleri için gerekli olan koşulları karşılamamaktadır.

Elde edilen sonuçlar, daha basit modellerin, kullanılan veri seti için daha iyi performans sergilediğini göstermektedir. LR modelinin genel olarak yüksek doğruluğu, doğrusal ilişki içeren veri kümelerinde en uygun yaklaşım olabileceğini göstermektedir. RF, LSTM ve YSA gibi daha karmaşık modellerin performansı, model eğitimi sırasında karşılaşılan sorunlar veya veri kümesinin doğası gereği zayıf kalmıştır. Bu durum, model seçerken dikkate alınması gereken önemli bir faktör olduğunu ve doğru modelin veri setinin özelliklerine uymasının kritik olduğunu vurgulamaktadır.

Veri setinin niteliğine göre doğrusal regresyonun tercih edilmesi önerilir. Ancak daha karmaşık veri kümelerinde, RF ve derin öğrenme modelleri (LSTM, YSA) için hiperparametre optimizasyonu ve model ince ayarı ile daha iyi sonuçlar sağlanabilir.



KAYNAKÇA

- Ali, A., & Mashwani, W. (2023). A supervised machine learning algorithms: applications, challenges, and recommendations. *Proceedings of the Pakistan Academy of Sciences: A. Physical and Computational Sciences*, 60(4). [https://doi.org/10.53560/ppasa\(60-4\)831](https://doi.org/10.53560/ppasa(60-4)831)
- Akşehir, Z. D., & Kılıç, E. (2019). Makine öğrenmesi teknikleri ile banka hisse senedi senetlerinin fiyat tahmini. *Türkiye Bilişim Vakfı Bilgisayar Bilimleri ve Mühendislik Dergisi*, 12(2), 30-39.
- Altunbaş, C. (2021). *Derin öğrenme ile hisse senedi piyasası tahmini* (Yüksek lisans tezi). Yükseköğretim Kurulu Ulusal Tez Merkezi'nden edinilmiştir. (Tez No. 675038)
- Arslan, M. E., & Kırıcı, P. (2021). Makine öğrenmesi ile borsa analizi. *Avrupa Bilim ve Teknoloji Dergisi*, 28, 1116-1120. <https://doi.org/10.31590/ejosat.1012785>
- Arslankaya, S., & Toprak, Ş. (2021). Makine öğrenmesi ve derin öğrenme algoritmalarını kullanarak hisse senedi fiyat tahmini. *Uluslararası Mühendislik Araştırma ve Geliştirme Dergisi*, 13(1), 178-192. <https://doi.org/10.29137/umagd.771671>
- Bakdemir, M. (2024). *Python ile makine öğrenmesi algoritmalarını kullanarak rüzgar santralinin elektrik üretim tahmini* (Yüksek lisans tezi). Yükseköğretim Kurulu Ulusal Tez Merkezi'nden edinilmiştir. (Tez No.864984)
- Bardak, S., Ersen, N., Polat, K., & Akyüz, K. C. (2024). Makine öğrenmesi yöntemleri ile hisse senedi fiyat tahmini: kâğıt firması örneği. *Artvin Çoruh Üniversitesi Orman Fakültesi Dergisi*, (25), 47 – 58. <https://doi.org/10.17474/artvinofd.1500569>
- Buchanan, B. G., & Wright, D. (2021). The impact of machine learning on UK financial services. *Oxford Review of Economic Policy*, 37(3), 537–563. <https://doi.org/10.1093/oxrep/grab016>
- Cayir, A., Dag, H., & Yenidogan, I. (2018). Feature Extraction Based on Deep Learning for Some Traditional Machine Learning Methods. *2018 3rd International Conference on Computer Science and Engineering* (pp. 494-497). Saraybosna Bosna Hersek.
- Demirel, U. (2019). *Hisse senedi fiyatlarının makine öğrenmesi yöntemleri ve derin öğrenme algoritmaları ile tahmini* (Yüksek lisans tezi). Yükseköğretim Kurulu Ulusal Tez Merkezi'nden edinilmiştir. (Tez No.547320)
- Doğan, S., & Büyükkör, Y. (2022). Makine öğrenmesi ile finansal zaman serisi tahminleme. *Ankara Hacı Bayram Veli Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 24(3), 1205-1230. <https://doi.org/10.26745/ahbvuibfd.1191080>
- Ersayan, B. E. (2023). *Hisse senedi fiyatlarının tahmini için derin öğrenmesi çerçevesinin geliştirilmesi* (Yüksek lisans tezi). Yükseköğretim Kurulu Ulusal Tez Merkezi'nden edinilmiştir. (Tez No.768994)
- Filiz, E., Karaboga, H. A., & Akogul, S. (2017). BIST – 50 endeksi değişim değerlerinin sınıflandırılmasında makine öğrenmesi yöntemleri ve yapay sinir ağları kullanımı. *Çanakkale Üniversitesi Sosyal Bilimler Enstitüsü Dergisi* 26(1), 231 – 241.

- Gür, Y. E. (2024). Stock price forecasting using machine learning and deep learning algorithms: a case study for the aviation industry. *Fırat Üniversitesi Mühendislik Bilimleri Dergisi*, 36(1), 25 – 34. <https://doi.org/10.35234/fumbd.1357613>
- Hızlısoy, S. (2020). *Music emotion recognition using convolutional long short term memory deep neural networks* (Yüksek Lisans Tezi). Yükseköğretim Kurulu Ulusal Tez Merkezi'nden edinilmiştir. (Tez No. 647909)
- Iglesias, L. L., González, D. R., Blanco, J. A. P., Fernández-Miranda, P. M., Vega, J. A., Del Barrio, A. P., Mandly, A. A. G., & Bellón, P. S. (2021). A primer on deep learning and convolutional neural networks for clinicians. *Insights into Imaging*, 12(1). <https://doi.org/10.1186/s13244-021-01052-z>
- Jha, K. K., Hassan, M. A. M., Jha, R., Mahesh, T., Yadav, S. K., & Jha, A. K. (2021). A brief comparison on machine learning algorithms based on various applications: a comprehensive survey. *2021 IEEE International Conference on Computation System and Information Technology for Sustainable Solutions* (pp. 1-5). Bangalore India.
- Jin, W. (2020). Research on machine learning and its algorithms and development. *Journal of Physics: Conference Series*, 1544(1), 012003. <https://doi.org/10.1088/1742-6596/1544/1/012003>
- Kalyoncu, S. (2020). *Borsa analizi ve tahmini için derin öğrenme ağları* (Yüksek lisans tezi). Yükseköğretim Kurulu Ulusal Tez Merkezi'nden edinilmiştir. (Tez No.640210)
- Karacan, S., & Kırdar, M. (2021). Hisse senedi fiyat tahmininde makine öğrenmesi ve yapay zekâ kullanımı. *Uluslararası Sosyal Araştırmalar Dergisi*, 14(76), 666-674.
- Kocaoğlu, D., Turgut, K., & Konyar, M. Z. (2022). Sector-based stock price prediction with machine learning models. *Sakarya Üniversitesi Bilgisayar ve Bilgi Bilimleri Dergisi*, 5(3), 415–426. <https://doi.org/10.35377/saucis...1200151>
- Kumar, D., & Singh, S. (2024). Analyzing the impact of machine learning algorithms on risk management and fraud detection in financial institution. *International Journal of Research Publication and Reviews*. <https://doi.org/10.55248/gengpi.5.0524.1135>
- Kurucan, F. (2019). *Büyük veri ve finansal piyasalarda istatistikî makine öğrenmesi metodlarının yatırım kararlarında kullanılması* (Yüksek lisans tezi). Yükseköğretim Kurulu Ulusal Tez Merkez'inden edinilmiştir. (Tez No.596000)
- Lemley, J., Bazrafkan, S., & Corcoran, P. (2017). Deep learning for consumer devices and services: pushing the limits for machine learning, artificial intelligence, and computer vision. *IEEE Consumer Electronics Magazine*, 6(2), 48–56. <https://doi.org/10.1109/mce.2016.2640698>
- Mienye, E., Jere, N., Aruleba, K., Obaido, G., & Mienye, I. D. (2024). Deep learning in finance: a survey of applications and techniques. *AI*, 5(4), 2066–2091. <https://doi.org/10.3390/ai5040101>
- Nikou, M., Mansourfar, G., & Bagherzadeh, J. (2019). Stock price prediction using deep learning algorithm and its comparison with machine learning algorithms. *Intelligent*

System in Accounting, Finance and Management, 26(4), 164 – 174. [DOI: 10.1002/isaf.1459](https://doi.org/10.1002/isaf.1459)

- Nwaimo, C., Adegbola, M., & Adegbola, A. (2024). Predictive analytics for financial inclusion: Using machine learning to improve credit access for under banked populations. *Computer Science & IT Research Journal*, 5(6), 1358 – 1373. <https://doi.org/10.51594/csitrj.v5i6.1201>
- Pabuçcu, H. (2019). Borsa endeksi hareketlerinin makine öğrenme algoritmaları ile tahmini. *Uluslararası İktisadi ve İdari İncelemeler Dergisi*, (23), 179-190. <https://doi.org/10.18092/ulikidince.484138>
- Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock market index using fusion of machine learning techniques. *Expert Systems With Applications*, 42(4), 2162 – 2172. <https://doi.org/10.1016/j.eswa.2014.10.031>
- Rizinski, M., Chitkushev, L. T., Mishev, K., Trajanov, D., Vodenska, I., & Peshov, H. (2022). Ethically responsible machine learning in fintech. *IEEE Access*, 10, 97531–97554. <https://doi.org/10.1109/access.2022.3202889>
- Scott, I. A., Coiera, E. W., Cook, D., & Richards, B. (2019). Machine learning in clinical practice: prospects and pitfalls. *Medical Journal of Australia*, 211(5), 203. <https://doi.org/10.5694/mja2.50294>
- Sevim S., (2019). *Makine Öğrenme Yöntemleriyle Hastalıkların Önceden Belirlenmesi ve Diyabet Üzerine Bir Uygulama* (Yüksek lisans tezi). Yükseköğretim Kurulu Ulusal Tez Merkezi'nden edinilmiştir. (Tez No.632856)
- Shaveta, S. (2023). A review on machine learning. *International Journal of Science and Research Archive*, 9(1), 281–285. <https://doi.org/10.30574/ijrsra.2023.9.1.0410>
- Şişmanoğlu, G., Koçer, F., Önde, M. A., & Şahingöz, Ö. K. (2020). Derin öğrenme yöntemleri ile borsa fiyat tahmini. *Bitlis Eren Üniversitesi Fen Bilimleri Dergisi*, 9(1), 434 – 445. <https://doi.org/10.17798/bitlisfen.571386>
- Öngün M. (2023). *Çok değişkenli finansal zaman serisi analizinde yapay zekâ kullanımı*. (Cilt 11, Sayı 1) YBS Ansiklopedi.
- Toprak, Ş. (2022). *Makine öğrenmesi yöntemlerini kullanarak bir petrokimya firmasının hisse senedi fiyat tahmini* (Yüksek lisans tezi). Yükseköğretim Kurulu Ulusal Tez Merkezi'nden edinilmiştir. (Tez No.787766)
- Tsai, C., & Wang, S., (2009). Stock price forecasting by hybrid machine learning techniques. *Procedings of the International MultiConference of Engineers and Computer Scientists*. (Vol. 1, No: 755, p.60).
- Vijh, M., Chandola, D., Tikkiwal, V. A., & Kumar, A. (2020). Stock closing price prediction using machine learning techniques. *Procedia computer science*, 167, 599-606.

ÖZ GEÇMİŞ

Hüseyin KARAKAŞ tarihinde da doğdu. İlköğrenimini Mimar Sinan İlkokulu ve Yakubiye Ortaokulunda tamamladı. Ortaöğrenimini 2003 yılında Şanlıurfa Lisesinde tamamladı. 2022 yılında Bayburt Üniversitesi Endüstri Mühendisliği Bölümünden mezun oldu. 2023 yılında Bayburt Üniversitesi Lisansüstü Eğitim Enstitüsü Endüstri Mühendisliği Ana Bilim Dalı'nda yüksek lisans eğitimine başladı.

