FATİH SULTAN MEHMET VAKIF ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
BİLGİSAYAR MÜHENDİSLİĞİ PROGRAMI

MAKİNE ÖĞRENME METOTLARI KULLANILARAK FONLARIN FİYAT TAHMİNİ İÇİN MODEL GELİŞTİRİLMESİ

YÜKSEK LİSANS TEZİ

SEDRETTİN ÇALIŞKAN

İSTANBUL, 2020

FATİH SULTAN MEHMET VAKIF ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
BİLGİSAYAR MÜHENDİSLİĞİ PROGRAMI

MAKİNE ÖĞRENME METOTLARI KULLANILARAK FONLARIN FİYAT TAHMİNİ İÇİN MODEL GELİŞTİRİLMESİ

YÜKSEK LİSANS TEZİ

Sedrettin ÇALIŞKAN
(170221009)

Danışman
Dr. Öğr. Üyesi Ebubekir KOÇ

İSTANBUL, 2020

BEYAN/ ETİK BİLDİRİM

Bu tezin yazılmasında bilimsel ahlak kurallarına uyulduğunu, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, tezin herhangi bir kısmının bağlı olduğum üniversite veya bir başka üniversitedeki başka bir çalışma olarak sunulmadığını beyan ederim.

Sedrettin Çalışkan

İmza

Aileme,

MAKİNE ÖĞRENME METOTLARI KULLANILARAK FONLARIN FİYAT TAHMİNİ İÇİN MODEL GELİŞTİRİLMESİ

SEDRETTİN ÇALIŞKAN

ÖZET

Bu tez çalışmasında, makine öğrenme algoritmaları kullanılarak fonların fiyat tahmini için model geliştirilmesi yapılmıştır. Fon fiyat tahmin sistemi amacıyla geliştirilmesi yapılan model çalışmasında kullanılan veri setinin gerçek veri değerlerinden alınması, yapılan modelin tahmin başarısında reel piyasadaki fon bilgilerine yakın değerler olmasını, model tutarlığını ve güvenirliliğini sağlamıştır. Veri seti, Takas İstanbul(İstanbul Takas ve Saklama Bankası A.Ş.-Takasbank)'un platform ve veri kaynağı sağlayıcılığı yaptığı Türkiye Elektronik Fon Dağıtım Platformu (TEFAS) web sitesi üzerinden, 02.01.2019 – 31.12.2019 tarihleri arasında ki erişime açık olan fon bilgilerinde elde edilmiştir. Bu veri seti araştırmayı da özgün bir kategoriye yerleştirmiştir.

Model çalışması için kullanılan makine öğrenme teknikleri Kısmi En Küçük Kareler Regresyonu (PLSR), Ridge Regresyonu(RR), Destek Vektör Regresyonu (SVR) ve Yapay Sinir Ağları(YSA) algoritmalarıyla yapılmıştır. Geliştirilen model başarı değerlendirmeleri ise; Hata Karelerinin Ortalaması(MSE), Hata Kare Ortalamasının Karekökü(RMSE), Ortalama Mutlak Hata(MAE) ve R Kare

Oranı(R2) ölçüm metriklerine göre yapılmıştır. Bu ölçüm metriklerine göre PLSR, RR ve SVR algoritmalarında elde edilen sonuçların kabul edilebilirlik oranlarının çok düşük olduğu tespit edilmiştir. YSA ile elde edilen tahmin değerlerinin başarı oranının ise, model önerisi için kabul edilebilir ölçekte olduğu gözlemlenmiştir. Bu değerlendirmelerden hareketle, fonların fiyat tahmini çalışmasında YSA ile geliştirilen model tercih edilmiştir. Sonuç olarak, fonların fiyat tahmin için geliştirilen model çalışmasında, YSA modeli tahmin edilen fon fiyat değerinin birim pay başına düşen hata oranı, RMSE değerine göre (+/-) 6.2 olarak bulunmuştur. Model başarı yüzdesi ise, R2 ölçüm metrik sonucuna bakılarak %90 üzerinde bir tahmin başarısı olarak elde edilmiştir. Ayrıca, ölçüm metrik sonuçları da göz önüne alındığında regresyon problemleri için PLSR, RR ve SVR modellerinin de kullanılabilirliklerini yitirmedikleri de değerlendirilmiştir.

Anahtar Kelimeler; Makine Öğrenme, Fon Fiyat, PLSR, RR, SVR ve YSA

MODEL DEVELOPMENT FOR THE PRICE ESTIMATION OF FUNDS BY USING MACHINE LEARNING METHODS

SEDRETTİN ÇALIŞKAN

ABSTRACT

In this thesis, a model was developed for price estimation of funds by using machine learning algorithms. Taking the data set used in the model study developed for the purpose of the fund price estimation system from the real data values ensured that the model was close to the real market fund information, model consistency and reliability. Data set, swap Istanbul (Istanbul Settlement and Custody Bank -Takasbank) 's platform and data source Agreements on providing his Turkey Electronic Funds Distribution Platform (TEFAS) through the website, 02.01.2019 - 12.31.2019 open access to that between obtained in fund information. This dataset also placed the research in a unique category.

Machine learning techniques used for model study were performed using Partial Least Squares Regression (PLSR), Ridge Regression (RR), Support Vector Regression (SVR) and Artificial Neural Networks (ANN) algorithms. The developed model success evaluations are; Average of Error Squares (MSE), Square Root of Error Square Mean (RMSE), Average Absolute Error (MAE) and R Squared (R2) were made according to measurement metrics. According to these measurement

metrics, the acceptability rates of the results obtained in PLSR, RR and SVR algorithms were found to be very low. It was observed that the success rate of the predicted values obtained with ANN was at an acceptable scale for the model proposal. Based on these evaluations, the model developed with ANN was preferred in the price estimation study of the funds. As a result, in the model study developed for the price estimation of the funds, the error rate per unit share of the estimated fund price value of the ANN model was found as (+/-) 6.2 according to the RMSE value. The model success percentage was obtained as a prediction success over 90% by looking at the R2 measurement metric result. Also, considering the measurement metric results, it was evaluated that PLSR, RR and SVR models did not lose their usability for regression problems.

Keywords; Machine Learning, Fund Price, PLSR, RR, SVR and ANN

ÖNSÖZ

Önsöz, tezin amacı, önemi, kısaca kapsamı, gerekiyorsa tezin hazırlanmasında karşılaşılan güçlükler konusunda bilgilerin yanı sıra, çalışma aşamasında bilimsel katkı sağlayan, maddi ve manevi desteği olan kişi ve/veya kurumlara teşekkür gibi açıklamaları içerir.

İÇİNDEKİLER

Sayfa

KISALTMALAR LİSTESİ

ÇİZELGE LİSTESİ

1. GİRİŞ

Günümüzün dijitalleşen bilgi çağının insanlara sunduğu yeni imkanlarla birlikte bilginin dolaşım hızı artmış ve bilgiye erişim de oldukça kolaylaşmıştır. Bu durum da kendisiyle beraber yeni bir pazar alanı oluşturmuştur. Son yüzyılda bilgiyi geliştiren inovatif teknolojilere yatırım yapan ülkeler, global pazarlarda rekabet üstünlüğü elde etmekte ve ekonomileri için büyük bir katma değer faktörü oluşturmaktadırlar. Artık ülkelerin gelişmişlik düzeylerini geleneksel yöntemlerle elde ettikleri gelir oranları belirlememekte, tam aksine bilgiye, bilgi teknolojisine, yenilikçi teknolojilere ve bilgi merkezli üretilen sistemlere yaptıkları katkılar, yatırımlar şekillendirmektedir. Dolayısıyla modern çağın getirdiği bu perspektifle güç ve gelişimi elinde tutan ülkelerin, asıl odak merkezlerini bilgi ve otomasyon sistemlerine yaptıkları yatırımlar ön plana çıkarmaktadır. Bu kapsamda dijital çağla beraber birçok kavram, kurum ve işleyişler geleneksel kültürü yapı bozumuna uğratacak şekilde yeniden tanımlanmaktadır. Ekonomiden sağlık sektörüne, eğitimden sanayiye ve daha benzeri birçok sektörde yeni ufuklar ve imkanlar doğmaktadır. Bu sektörlerdeki teknolojik yatırımların geldiği nokta da artık insan gücünün üretimi yerine uzman sistemler, akıllı yazılımlar, donanımlar, robotik tasarımlar gibi yapay zekanın ve onun alt dalları olan makine öğrenmesi ve derin öğrenme konularını kapsayan geniş bir yelpazede araştırma ve geliştirme çalışmaları yapılmaktadır. Bu çalışmaların temelinde geleneksel olan yapıların nasıl otomatize edilerek uzman sistemlere dönüştürüleceği üzerinde durulmaktadır. Sektörel bazda yapılan bu otomasyon sistemleri çok sayıda kolaylığı da beraberinde getirmektedir. Bu tez çalışmasında da, finans sektöründe geleneksel yöntemlerle yapılan birçok işlem ve sürecin teknolojik gelişmelerin sunduğu imkanlar kapsamında nasıl daha iyi otomatize edilebileceği incelenmektedir. Finans dünyasında yapılan bir yatırım aracının fiyatının ne olacağı öngörüsünün, finansçıların birden fazla parametreyi geleneksel yöntemlerle hesaplamaya çalışmasıyla mümkün hale gelirken, bu hesaplama süreçleri her ne kadar birçok araştırma sonucunda yapılsa da, öngörü sürecinde hesaba katılamayan ya da parametrelerin hesaplanmasında yaşanan

zorluklar nedeniyle doğruluk payı zayıflamaktadır. Bu sebeple öngörü sürecinde birden fazla parametreyi ve geçmişteki yatırım aracına ait verileri hesaba katarak, daha doğru öngörü sonuçları elde edilmesini sağlayacak teknolojik yöntemler kullanılmaktadır. Bu çalışmayla da bu yöntemlerden makine öğrenme teknikleri kullanılarak yatırım araçlarından fonların fiyatını tahminleyecek bir model uygulama önerisi üzerinde durulacaktır.

Finans sektörüne dair alınan fonların fiyat tahminleri üzerine birçok araştırmacı uzun yıllardır çalışmaktadır. Fon en özet haliyle yatırım yapmayı düşünen yatırımcılara belli bir orandan katılım pay hakkı verilerek toplanan paralar ile katılma pay sahipleri adına, risk çeşitlendirilmesi ve inançlı mülkiyet esaslarına göre, portföy işletme amacıyla kullanılan mal varlığıdır[1]. Fonların fiyat tahmini de alım satımı gerçekleştirilecek olan fonun fiyatını ya da yaklaşık değerini önceden bilinmesini sağlar. Buradaki fonların fiyat tahmin çalışmalarındaki hedeflenen amaç, bir yatırımcının doğru yatırım araçlarına yatırım yapmasını sağlayarak kazanç elde etmesini gerçekleştirmek ve yatırım aracının değerinin önceden öngörülüp öngörülemeyeceğini sağlamaktır. Öngörü, belirli varsayımlar öncülüğünde bir değişkenin gelecekte elde edeceği çıktı değerini önceden yaklaşık olarak belirlenmesi biçiminde tanımlanabilmektedir. Ancak yatırımcıların öngörü de bulunabilmesi için mevcut birçok parametrenin de dahil olduğu sistemleri tahminlemek kolay olmamaktadır. Doğru öngörüde (tahminde) bulunmanın birçok başarılı kararları beraberinde getireceği ve bu ulaşılan başarıların da maksimum düzeye çıkartılabileceği gerçeği, öngörü çalışmalarına olan merakı da artırarak bu alanda sürekli önemli gelişmelerin yaşanmasına ön ayak olmaktadır. Daha önceden geleneksel olarak, matematiksel ve istatistiksel metotlarla yapılan tahmin denemeleri, şimdi özellikle teknolojik gelişmelerin ilerlemesiyle makine öğrenme teknikleriyle beraber daha başarılı sonuçlar vermektedir.

Son yıllarda elde edilen başarılarla, makine öğrenme metotları farklı alanlarda da kullanım yaygın lığı kazanmaya başlamış ve uygulama sahaları da giderek genişlemektedir. Bu genişlemeyle beraber makine öğrenmenin temel uygulama

alanlarından biri de tahmin edilebilirlik olmuş ve makine öğrenme ile doğruya en yakın tahmin değerlerini elde edebilmek için daha optimize tahmin modelleri üzerinde kullanılmaya başlanmıştır. Ayrıca, makine öğrenme teknikleri ile yapılan tahmin çalışmaları birçok sektörde uygulama geliştirme süreçlerinde de kullanılmaktadır. Yapılan bu tahmin çalışmalarında başarılı sonuçların elde edilmesi ile makine öğrenme tekniklerinin tahmin alanında kullanımını artırmakta ve yaygınlaştırmaktadır. Bu kazanımlar göz önünde bulundurularak bu çalışma da makine öğrenme metotları kullanılacak ve finans sektörü için ihtiyaca göre en doğru veya doğruya en yakın sonuçlar elde edebilecek bir fon fiyat tahminleme model önerisi üzerinde durulacaktır. Geliştirilecek olan fonların fiyat tahmin model önerisinin, finans sektöründeki tahmin çalışmalarının yatırımcılar için önemli bir konu olduğu göz önüne alındığında, bu alana dair önemli bir katkı sağlayacağı aşikardır. Yatırımcıların alım satımını yapacağı fonlardan kâr merkezli kazanç elde etmelerini sağlayarak, tahminlenebilir değerler üzerinden çıkarımlar ve analizler yapılması mümkün kılınarak fon yatırımcılarının kazançlı yatırımlar için doğru yönlendirilmesine zemin oluşturulacaktır. Bu kapsamda, yatırımcı fon alırken, makine öğrenme yöntemiyle oluşturulacak tahmin modelinde fon fiyatını belirleyen birden fazla parametre değeri olan; fonun mevcut fiyatı, alan kişi sayısı, menkul dağılım oranları, toplam piyasa değeri ve bunların dışında piyasalarda çok büyük oranda etkisi olan faiz oranları, döviz fiyatları, kıymetli maden fiyatları gibi değişkenlerin tahmin sistemi üzerindeki etkisini değerlendirilmeye katabilecektir. Bu şekilde fon alım satımını yapacak yatırımcıya belli bir sapma değeri oranında o fonun fiyatını öngörülebilir yaparak, yatırımcı tercihlerinde fon fiyat tahmininin çok büyük bir etki yaratacağı kaçınılmazdır. Buraya kadar yapılan bu açıklamalardan da görülmektedir ki, geliştirilen fon fiyat tahmin model sisteminin bireysel yatırımlara ve yatırımcılara sunduğu katkının dışında finans sektörü için de önemli bir katkı payı sunacağı öngörülmektedir. Ayrıca, bu çalışma da geliştirilecek olan fon fiyat tahmin model sistemi için deneysel çalışmalarda kulanılacak veri setinin gerçek verilerden oluşması, elde edilen sonuçların gerçek piyasalardaki fonların fiyat değerleri ile

örtüşür olmasını sağlayarak tahmin modelinin daha doğru ve güvenilir bir alt yapıya sahip olmasını sağlayacaktır. Bu durum, yapılan bu çalışmayı diğer çoğu çalışmadan ayıran temel bir özelliktir. Kullanılan veri kümesi, Türkiye'de fonların tek platform üzerinde toplanmasını sağlayarak, fon piyasası için önemli katkı sağlayan Takas İstanbul(İstanbul Takas ve Saklama Bankası A.Ş.-Takasbank) tarafından platform sahipliği ve sağlayıcılığı yaptığı TEFAS(Türkiye Elektronik Fon Alım Satım Platformu) web sitesi üzerinden alınan fon bilgilerinden oluşturulmuştur. TEFAS platformu ise; tüm fonların tek bir sistem üzerinden karşılaştırılmasını yapan ve tek bir yatırım hesabıyla piyasadaki tüm fonlara erişim imkanı sağlayan elektronik bir fon platformudur[2]. Bu veri setini kullanarak yapılacak model önerisinin, TEFAS platformu üzerinden halka açık olan gerçek fon bilgileri işlenerek yapılacağından bu alanda gerçekleşecek birçok çalışma için araştırmacılara örnek oluşturması ve platform verilerinin benzer araştırma ekosistemlerinde de yaygınlaşmasına katkı sağlayacağı öngörülmektedir.

Genel olarak tez çalışması beş bölümden oluşmaktadır. İlk olarak, giriş bölümünde teknolojik gelişmelerin geldiği son nokta ve tüm sektörlerde kullanım olanaklığı üzerine sunduğu imkanlar araştırılmıştır. Daha sonra teknolojinin finans sektöründe kullanımına ve sağladığı kolaylıklara değinilmiştir. Devamında, finans sektöründe önemli bir yere sahip olan tahmin çalışmalarının öneminden bahsedilmiş ve tez çalışmasının temel önerisi, amacı, deneysel bölümde kullanılan veri setinin önemi ve tezin model çalışmasının sektöre katkısı bağlamında bir değerlendirme yapılmıştır.

İkinci bölüm olan literatür araştırmasında, makine öğrenme çalışmaları ile ilgili ekonomi, sağlık, eğitim ve sanayi gibi birçok sektörde yapılmış olan araştırmalar incelenmiş ve incelenmesi yapılan araştırmalar genel olarak; çalışmanın amacı, kullanılan makine öğrenme algoritmaları, veri seti ve elde edilen başarı sonuçları kapsamında tasniflenerek, makine öğrenme çalışamlarının genel kullanım alanları ve sonuçları değerlendirilmiştir. Daha sonrasında, spesifik olarak makine öğrenme çalışmalarının fiyat tahmini uygulamalarında kullanım sıklıkları, en çok

kullanılan algoritmalar, başarı sonuçları ve hangi sektörlerde yoğun olarak çalışmaların yapıldığı araştırmalar incelenmiş ve sınıflandırılması yapılmıştır. Literatür bölümün son kısmında ise, model öneri çalışmasının geliştirme aşamasında kullanılan algoritmaların, neden tercih edildiği, fiyat tahmin çalışmalarında kullanım sıklıkları ve elde edilen başarı çıktıları incelenerek model çalışmasında kullanılan algoritmaların, tercih edilmesini sağlayan nedenlerin güçlendirilmesine ve kavramsal anlam bütünlüğünün sağlanmasına zemini oluşturulmuştur.

Üçüncü bölüm de ise kullanılan makine öğrenme algoritmaları diğer bir deyişle metot çalışmaları üzerinde durulmuştur. Tahmin problemleri genelde regresyon algoritmalarının konusu olduğu için fonların fiyat tahmininin model geliştirilmesi, doğrusal çoklu regresyon algoritmalarından; Kısmi En Küçük Kareler Regresyonu(PLSR) ve Ridge Regresyonu(RR) kullanılmış ve doğrusal olmayan çoklu regresyon algoritmalarından ise; Destek Vektör Regresyonu(SVR) ve Yapay Sinir Ağları(YSA) kullanılarak detaylı analizleri yapılmıştır. Yapılan analizler, kullanılan her bir algoritmanın bölüm başlığı altında tasniflenerek, algoritmaların teorik yapısı, matematiksel denklem yapıları ve çalışma adımlarından bahsedilmiştir.

Dördüncü bölüm olan model çalışmasında, ilk olarak veri seti hakkında ayrıntılı bilgiler verilmiş ve veri kümesi, veri ön işlem adımları sırasıyla yapılarak deneysel çalışmalarda kullanılmak üzere temizlenmiş bir veri seti haline getirilmiştir. Daha sonra model geliştirme de kullanılacak olan PLSR, RR, SVR ve YSA algoritmalarının her biri için ayrı adımlar halinde, modelin eğitim ve test veri setleriyle model kurulma aşaması ve model sonuçları elde edilmiştir. Daha sonra kurulan model lerden her bir algoritmanın optimizasyon problemine konu olan değişkenleri incelenerek, model optimizasyon çalışmalarının gerçekleştirilmesini sağlanarak nihayi doğrulanmış model sonuçlarına varılmıştır. Ayrıca, her bir model sonuçlarının değerlendirmesi de dört algoritma için ortak ölçüm metrikleri olan; Hata Karelerinin Ortalaması(MSE), Hata Kare Ortalamasının Karekökü(RMSE), Ortalama Mutlak Hata(MAE) ve R Kare Oranı(R2) metotlarıyla ölçülerek, optimize edilmemiş model ile doğrulanmış model sonuçları karşılaştırması yapılmıştır. Bölüm sonunda

ise, dört modelin doğrulanmış sonuçları üzerinden karşılaştırma tabloları çıkarılmış, her bir algoritmayla kurulan model için değerlendirmeler yapılmış ve dört ölçüm metriğine göre en yüksek başarı oranı sağlayan algoritmanın tespiti yapılmıştır.

Tez çalışmasının son bölümünde çalışmanın verilerinin değerlendirilmesinden ve fonların fiyat tahmini için kullanılan PLSR, RR, SVR ve YSA algoritmalarının avantaj ve dezavantajlarından bahsedilmiştir. Devamında bu algoritmaları kullanarak geliştirilen model sonuçlarının değerlendirilmesi yapılmış ve tez çalışmasının temel amacı olan model önerisi kapsamında tercih edilen modelin seçilmesini sağlayan faktörlerin ve kısıtların kapsamlı değerlendirilmesi gerçekleştirilmiştir. Ayrıca, deneysel çalışmalarda karşılaşılan teknik kısıtlar üzerinde de durulmuş ve tez çalışmasının genel değerlendirilmesi yapılarak gelecek çalışmalar için önerilerde bulunulmuştur.

2. LİTERATÜR ARAŞTIRMASI

2.1. MAKİNE ÖĞRENME ÇALIŞMALARI İLE İLGİLİ YAPILAN GENEL ARAŞTIRMALAR

Bu bölümde makine öğrenme çalışmalarının sektör bazında kullanımına dair bir incelemesi sunulmuştur. Bu kapsamda ekonomi, eğitim ve sağlık sektörleri bazındaki alanlara dair yapılan referans çalışmalar incelenmiştir. Makine öğrenmesi, esas olarak 1959 yılında bilgisayar biliminin yapay zekada sayısal öğrenme ve model tanıma çalışmalarından geliştirilmiş bir alt dalıdır. Matematiksel ve istatistiksel yöntemler kullanarak mevcut verilerden çıkarımlar yapar ve bu verileri geçmiş deneyimleri kullanarak modeller ve tahminlerde bulunur. Bu bağlamda makine öğrenmesi, insanların geri beslemeli öğrenim şeklinin bilgisayarda taklit edilmiş halidir[3].

Makine öğrenme Gözetimli ve Gözetimsiz olmak üzere, temelde iki farklı çerçevede gerçekleştirilir. Gözetimli öğrenme, verilerdeki etiketlenmiş gözlemlerden algoritma gözlemlerinin nasıl etiketlenmesini gerektiğinin öğretilmesidir. Uygulama

olarak Sınıflandırma (Classification) ve Regresyon (Regression) problemlerine odaklanmaktadır. Gözetimsiz öğrenme ise, etiketlenmemiş gözlemlerden bilinmeyen yapı ve ilişkilerin keşfini yapmasını ve gözükmeyen örüntüleri öğrenmesini sağlamaktadır. Uygulama olarak olarak ise, Kümeleme (Clustering) ve Boyut Azaltımı (Dimensionality Reduction) çalışmalarına odaklanmaktadır[3,4].

Makine öğrenmesinin bu tanımından hareketle birçok alanda çeşitli uygulamalar geliştirilmiştir. Bu çalışmalardan ekonomi alanına dair 2018 yıllında Kalaycı [5], makine öğrenmesi yöntemleri ile kredi risk analizi konusunu inceleyerek Türkiye'deki KOBİ müşterilerinin ödeme alışkanlıklarına göre ileri bir tarihte müşterinin kredi ödeme durumunun problemli kredi olup olmayacağını tahmin çalışmaları yapılmış ve müşteri risk skoru belirlemiştir. Bu çalışmada kullanılan veri seti Yapı Kredi Bankası tarafından sağlanmıştır. Kullanılan veriler bankanın KOBİ müşterileri için hazırladıkları kredi paketlerinin, 1 Ocak 2015 - 1 Ekim 2016 tarihleri arasında açılmış olan veri kullanılarak, bir deney veri seti hazırlanmıştır. Bu çalışmada, makine öğrenmesi tekniklerinden, lojistik regresyon (LR), karar ağaçları (KA), destek vektör makineleri (DVM), yapay sinir ağları (YSA), rastgele orman algoritması (ROA) ve meyilli hızlandırma (MH) yöntemi kullanılmıştır [5]. Yapılan deneysel çalışmalardan en iyi performans başarımı %83,05 başarı ile meyilli hızlandırma (MH) algoritması vermiştir [5].

Ekonomi alanında yapılan çalışmalara dair bir diğer güncel çalışma ise 2019 yıllında Gülcü ve Çalışkan [6] tarafından, makine öğrenme teknikleri kullanılarak elektrik piyasasında müşteri skorlama için model geliştirme çalışması yapılmıştır. Bu çalışma da kullanılan veri seti, Takasbank platformu kullanılarak günlük elektrik piyasası ile ilgili sözleşmeler yapan müşteri verilerinden oluşturulmuş ve veri seti 1 Ocak 2018 - 31 Aralık 2018 tarihleri arasında Takasbank platformu kullanılarak yapılan sözleşmelerden hazırlanmıştır. Makine öğrenmesi ve veri madenciliği tekniklerinden RFM (Recency, Frequency, Monetar) ve Kümeleme algoritmalarında K-Means, DBSCAN, Agglomerative yöntemleri uygulanmıştır. Çalışmanın sonucunda her bir müşteri için RFM metodu ile müşteri skorları hesaplanılmış ve

hesaplanan değerlerle kümeleme teknikleri yapılarak benzer skorlara sahip müşteri doğru sınıflandırılması yapılmıştır. Başarı ölçütlerini Takasbank kendi skorlama çalışmalarına kıyaslayarak ve kurum işbirliği çerçevesinde iyi bir başarı ile çalışmaları gerçekleştirilmiştir.

Makine öğrenmesinin bir diğer çalışma alanı bulduğu eğitim konusunda ise birden fazla çalışma bulunmaktadır. Eğitim konusu ile ilgili 2016 yılında Livieris ve arkadaşları [7], makine öğrenme tekniklerini kullanarak eğitimde öğrencilerin performansını tahmin etmek için bir karar destek sistemi yapmışlardır. Bu yapılan çalışma ile öğrencilerin bir öğretim yılının final sınavlarına ilişkin performansını tahmin etmek için kullanıcı dostu olan bir karar destek aracı tasarlanmıştır. Kullanılan veri seti, 2007-2010 yılları arasında Yunanistan'da özel bir okuldaki 14-15 yaşlarındaki öğrencilerin sözlü, vize ve final notları gibi öğrenci performansı hakkında bilgi veren verilerinden hazırlanmıştır. Makine öğrenme algoritmalarında Yapay Sinir Ağları (YSA), K En Yakın Komşu (KNN) ve Destek Vektör Makinesi (SVM) metotları kullanılmıştır. Eğitim alanına dair yapılan bir başka çalışma 2018 yılında Karabıyık [8], akademik yayınlar için makine öğrenmesi tabanlı arama motoru tasarlanması ve uygulanması konusunu incelemiştir. Makine öğrenme teknikleri ile akademik yayınların bulunması için kısıtlayıcı bir ağ örümceği tasarlanmıştır. Tasarlanan akademik arama motoru ile bir kullanıcının yaptığı çalışmasının özetinden yola çıkarak, en uygun yayın seçeneklerini sunacak bir çevrim içi sistemin yapılması sağlanmıştır. Kullanılan veri seti, Scimago Journal & Country Rank veri tabanından alınmış, 14 ana konu başlıklı, 450 adet metin özeti kullanılmıştır. Çalışma da, arama sonuçların doğru şekilde değerlendirilmesi için metin sınıflandırma metotları kullanılmıştır. Bu metotlardan Sade Bayes ve Destek Vektör Makinesi (SVM) kullanılmış ve elde edilen sonuçlardan Sade Bayes sınıflandırıcıda %80 başarı, Destek Vektör Makineleri sınıflandırıcısında ise %70 oranında başarı elde edilmiştir.

2015 yılında Kourou ve arkadaşlarının [9] sağlık çalışmaları ile ilgili yaptıkları, kanser prognoz ve tahmininde makine öğrenme teknikleri konusu

incelenmiştir. Kanser tipinin erken teşhisi ve hastalık sürecinin seyri hakkında önceden tahmin yapılmasının, klinik süreçlerin yönetimini kolaylaştıracağı için yapılmış olan kanser tahmin çalışmaları incelenmiştir. Yapılan bu inceleme de makine öğrenme tekniklerin kanser tespitinde kullanımına dair PubMed sonuçlarına göre 7510'dan fazla çalışma yayınlanmıştır. Bu yayınların büyük çoğunluğunun makine öğrenme algoritmaları ile tümörlerin saptanması ve kanser tipinin tahmini/ prognoz ile ilgili olduğu belirtilmiştir. Yaygın olarak kullanılan makine öğrenmesi tekniklerinden, Yapay Sinir Ağları (YSA), Bayes Ağları (BN), Destek Vektör Makineleri (SVM) ve Karar Ağaçları (DT) dahil olmak üzere bu teknikler tahmin modellerin geliştirilmesi için kanser araştırmalarında kullanılmıştır.

Sağlık alanında yapılan bir diğer çalışma da, 2017 yılında Çakmak [10], makine öğrenmesi yöntemleriyle tümör kontrol olasılığının hesaplanması konusunu inceleyerek onkoloji hastalarının aldığı radyoterapi tedavisinin tümör de ne kadar değişiklik göstereceğine dair bir tahmin modeli oluşturmuştur. Bu uygulamada kullandıkları veri seti, 2012-2015 yılları arasında, Karadeniz Teknik Üniversitesi Tıp Fakültesi Farabi Hastanesi, Radyasyon Onkoloji servisine gelen 30 hastanın verilerinden oluşturulmuştur. Kullanılmış olan makine öğrenme teknikleri, Destek Vektör Makineleri (SVM) ve Yapay Sinir Ağları (YSA) ile tahmin ve sınıflandırma çalışmaları yapılmıştır. SVM ve YSA başarı oranları ise, SVM modeli için %90 duyarlık ve YSA için %80 duyarlılık değerleri şeklinde elde edilmiştir. İki model karşılaştırmasında SVM modelinin daha başarılı sonuç verdiği görülmüştür. Sağlık alanına dair son olarak inceleyeceğimiz 2018 yıllında Saçlı [11] tarafından yapılan, makine öğrenmesi yardımıyla böbrek taşlarının elektromanyetik özelliklerinin sınıflandırılması konusu çalışılmıştır. Veri seti ise, 20 hastadan elde edilen 105 böbrek taşının elektromanyetik özelliklerini içeren veri kümesinden oluşmaktadır. Makine öğrenme tekniklerinden Yapay Sinir Ağları (YSA) ve En Yakın Komuş (KNN) yöntemleri kullanılmıştır. Deney sonuçlarında ise, YSA başarı ölçütünde %97 ve KNN metotu ile de %99 oranında başarı elde edilmiştir[11].

Makine öğrenmesin genel uygulama alanlarına dair birden fazla akademik çalışma bulunmaktadır.Bu başlık altında ise literatür taramasında örnek uygulama alanlarında olan ekonomi, eğitim ve sağlık sektörlerinde yapılmış olan akademik çalışmalardan sadece birkaçı üzerinden durulmuştur. Çok geniş bir sektörel bazda makine öğrenme teknikleri kullanılmış ve özellikle Türkiye'de kullanılan akademik çalışmalar ile ilgili 2019 yılında Adar ve Delice [12] tarafından, Türkiye'de makine öğrenmesi ile ilgili yapılan tez çalışmalarına yönelik bir literatür taramasında, Türkiye'de makine öğrenmesi algoritmalarının hangi alanda daha çok uygulandığının tespit edilmesi ve sektör bazında sınıflandırma işlemleri yapılmaya çalışılmıştır. Bu amaç doğrultusunda 146 tez incelenmiş ve en çok sağlık alanına dair çalışmanın bulunduğu belirtilmiştir. Makine öğrenme tekniklerinden ise, En Yakın Komuşu (KNN), Destek Vektör Makineleri (SVM), Karar Ağaçları ve Naive Bayes algoritmalarının sıklıkla kullanıldığı tespit edilmiştir.

2.2. MAKİNE ÖĞRENME METOTLARI İLE FİNANS SEKTÖRÜNDE YAPILAN FİYAT TAHMİN ÇALIŞMALARI

Bu bölümde yapılan literatür taramasında makine öğrenme tekniklerinin finans sektöründeki kullanım alanı ve literatür taramasında spesifik olarak 'fiyat' tahmini yapılmış olan çalışmaların incelenmesi yer almaktadır.

2004 yılında Tektaş ve Karataş [13] makine öğrenme tekniklerinden Yapay Sinir Ağları'nı kullanarak finans alanında Hisse Senetlerinin fiyat tahminlemesi konusunda yaptıkları çalışma da, İstanbul Menkul Kıymetler Borsası'nda (İMKB) işlem yapan şirketlerden yedisinin, hisse fiyatları için tahmin modeli geliştirmeye çalışılmıştır. Model geliştirme için kullanılan veri seti, 2002-2003 yılları arasında İMKB de işlem gören yedi şirketin haftalık ve günlük verilerinden oluşturulmuştur. Yaptıkları bu uygulamada temel amaçlarının Yapay Sinir Ağlarının finans sektöründe kullanımını yaygınlaştırılmasına katkıda bulunmaktadır. Fiyat tahmininde ise günlük verilerden daha başarılı sonuçlar elde etmişlerdir. Finans sektöründe makine öğrenme

teknikleri ile yapılmış olan başka bir çalışma ise, 2005 yılında Karaatlı ve arkadaşları [14], Hisse Senedi Fiyat Hareketlerinin Yapay Sinir Ağları Yönetimi ile Tahmin Edilmesi konusunun incelenmesidir. Uygulamadaki amaçlarının yapay sinir ağları metodunu kullanarak borsa endeksi tahmin modelini geliştirmek ve İstanbul Menkul Kıymetler Borsası (İMKB)'nın verilerini kullanarak uygulama geliştirilmesi olduğunu belirtmişlerdir. Kullanılan veri seti, İMKB 100 endeksin 1990 Ocak ve 2002 Aralık tarihleri arasındaki aylık verilerinden oluşmaktadır. Yaptıkları deneysel çalışmada yapay sinir ağları ile elde ettikleri başarı oranının yüksek olduğunu ve başarı ölçütlerinin ise hata karelerin karekökü (RMSE) olduğu bildirilmiştir. Ayrıca, model olarak kullandıkları yapay sinir ağlarının RMSE değerinin regresyon yöntemlerine göre daha düşük değer de çıktığını belirtmişlerdir.

İstanbul Menkul Kıymetler Borsası (İMKB) verilerini kullanarak finans sektöründe fiyat tahmin çalışmalarını devam ettiren bir diğer çalışma 2009 yılında Gür [15], Hisse Senedi Fiyat Hareketlerinin Tahmini için Bir Yapay Sinir Ağı Modeli Önerisi adıyla bir yüksek lisans çalışması olarak yayınlanmıştır. İMKB 30'da yer alan şirketlerin hisse senedi fiyat değişimlerinin seans bazlı öngörüsü için model geliştirmeleri yapılmıştır. Kullanılmış olan veri seti, 04 Ocak 1999 ve 30 Kasım 2008 yılları arasındaki İMKB 30'da yer alan 9 şirket verisinden oluşmaktadır. Uygulama çalışmalarında, İMKB 30'da yer alan 9 şirket için YSA başarı tahmini 1.seans için %2,84, 2.seans için %3,5 ortalama mutlak hata oranı olarak elde edilmiştir. 2011 yılında Khan ve arkadaşları [16] tarafından, makine öğrenme tekniklerinden Yapay Sinir Ağlarını kullanılarak Bangladeş Borsası'ndaki hisse senedi fiyat tahmin çalışmaları yapılmıştır. Veri seti Borsada işlem gören ACI ilaç firmasının geçmiş verilerinden oluşmuştur. Deneysel kısmında ise, fiyat tahmini için yapay sinir ağları metotlarından ileri beslemeli sinir ağı ile geri yayılım algoritmaları kullanılmıştır. Fiyat tahmin başarı oranı ise 1.simülasyon için %3,71, 2. simülasyon için % 1,53 ortalamasında elde edilmiştir[16].

Bir başka çalışma ise, makine öğrenme tekniklerinden Destek Vektör Makineleri (SVM) ve metin madenciliği metotlarını kullanarak 2013 yılında

Hagenau ve arkadaşları [17] tarafından, finans haberlerinden yola çıkarak hisse senedi fiyat tahmini yapacak otomatik bir haber okuma modeli geliştirilmesi yapılmıştır. Veri seti, Almanya ve İngiltere'den yayınlanan kurumsal finans kuruluşlarının, 1997-2011 yılları arasındaki borsa hisse senedi fiyat bilgisi olan haber kaynaklarından oluşturulmuştur. Duyurular DGAP ve EuroAdhoc haber kaynaklarından elde edilmiştir. Uygulama kısmında metinlerin sınıflandırılması için metin madenciliği teknikleri ve karar destek sistemi (SVM) ile başarı oranı %76'ya varan doğruluk elde edildiğini ifade etmişlerdir. Fiyat tahmin çalışmalarıyla ilgili 2013 yıllında Hegazy ve arkadaşları [18] ise, Borsa'da işlem gören hisselerin fiyat tahmini için bir makine öğrenme modeli önerisinde bulunmuşlardır. Kullanılan veri setinde, S&P 500 borsalarındaki tüm hisse senedi sektörlerini kapsayan birçok şirket verisi işlenmiştir. Bu sektörler, bilgi teknolojisi (Adobe, Hp ve Oracle), finans (American Express ve Bank of New York) ve benzeri verilerden oluşmuştur. Uygulama olarak makine öğrenme algoritmalarından, Parçacık Sürü Optimizasyonu(PSO), En Küçük Kareler Destek Vektör Makineleri (LS-SVM) ve Yapay Sinir Ağları (YSA) metotları kullanılmıştır. Elde edilen sonuçlardan önerilen modelin daha iyi tahmin doğruluğuna, PSO algoritmasının LS-SVM'yi optimize ederek ulaştığını belirtmişlerdir.

Makine öğrenme teknikleriyle fiyat tahmin modeli önerilerinden olan bir diğer çalışma 2014 yılında Leung ve arkadaşları [19] tarafından, hisse senedi fiyat tahmininde makine öğrenmesi yaklaşımıyla bir uygulama önerisi olmuştur. Bu uygulamada kullanılan veri seti, S&P 500'ün Bilgi Teknolojisi sektöründeki şirketlerin verilerinden oluşturulmuştur. Deneysel kısımında, Yapısal Destek Vektör Makinesi (SSVM) ile doğru sınıflandırılmış düğümlerin başarı oranının %78'in üzerinde olmasından dolayı, modelin doğru öğrenildiği ifade edilmiştir. 2015 yılına gelindiğinde Çalışkan ve Deniz [20], makine öğrenme tekniklerinden yapay sinir ağları ile hisse senedi fiyatları ve yönlerin tahmini çalışmalarını yapmışlardır. Veri seti olarak, 14 Aralık 2009 ile 21 Kasım 2014 tarihleri arasında BİST 30'da yer alan 27 şirketin bilgisi kullanılmıştır. Uygulama olarak Yapay Sinir Ağlarıyla kurdukları

tahmin modeli, 27 şirket için 25 Kasım 2014 ile 01 Aralık 2014 tarihleri arasındaki verilerle yapılmıştır. Fiyat tahmini için ortalama mutlak hata değerinin %1,80 ve yönlerin artacak, azalacak şeklindeki tahmin değerindeki başarı oranı ise %58 olduğunu tespit etmişlerdir.

Makine öğrenme teknikleri ile fiyat tahmin çalışmalarında 2016 yılında Yüksel ve Akkoç [21] tarafından, altın fiyatlarının tahmini için yapay sinir ağları kullanılarak bir uygulama model geliştirilmesi yapılmıştır. Uygulamada veri setini, 03 Ocak 2002 ve 31 Ekim 2013 tarihleri arasında yer alan 2885 veri oluşturmuştur. Veri setini hazırlarken altın fiyat tahmininin daha hassas değerlerle yapılması için değişken olarak altın verisini etkileyecek; Gümüş fiyatları, Brent Petrol fiyatları, ABD doları/ EUR paritesi, EuroNext100 endeksi, Amerika Dow Jones Endeksi, 13 Hafta vadeli ABD bonosu faiz oranı ve ABD TÜFE endeksi değerleri de kullanılmıştır. Deneysel kısımda ise yapay sinir ağları ile elde edilen sonuçları R2, RMSE, MAE ve MAPE (%) kriterleri hesaplanarak başarı değerlendirilmesi yapılmıştır. Ayrıca duyarlılık analizi sonuçlarına göre altın fiyatlarının, gümüş ve petrol fiyatlarını önemli derecede belirlediği tespit edilmiştir. 2016 yılında ise Addai [22], finansal tahminler için makine öğrenme tekniklerini kullanarak hisse senedi/ endeks getirilerini tahminleme çalışmaları yapmıştır. Veri seti, Yahoo'dan elde edilmiştir. Veri içerikleri; Hisse senetlerinin açılış fiyatları, düşük fiyatları, yüksek fiyatları, işlem hacmi, kapanış fiyatları ve düzeltilmiş kapanış fiyatları hakkındaki bilgilerden oluşturulmuştur. Elde edilen veriler 1 Ocak 1999 ile 31 Aralık 2008 tarihleri arasını kapsamaktadır. Deneysel kısmında ise, hisse senedi endeksindeki günlük getirilerin hareketini tahmin etmek için makine öğrenme tekniklerinden beş farklı teknik uygulanmıştır. Bu teknikler Yapay Sinir Ağı (YSA), Lojistik Regresyon, Doğrusal Diskriminant Analizi (LDA), Karesel Diskriminant Analizi (QDA) ve K-En Yakın Komşu (KNN) algoritmalarından oluşmaktadır. Uygulama sonuçlarından en iyi performansı YSA'nın gösterdiğini ve hisse senetlerinin/endekslerinin açılış fiyatlarını kullanarak getiri tahmininde yaklaşık başarı değerinin % 61 olduğu ifade edilmiştir. Borsa hisse senedi fiyat tahminiyle ilgili bir başka çalışma 2016 yılında

Özçalıcı [23], yapay sinir ağları ile fiyat tahmin uygulamasın BIST30 senetlerini kullanarak yapmıştır. Veri kümesi, Ocak 2010 ile Kasım 2015 tarihleri arasındaki BIST30'da yer alan hisse senetlerinden oluşturulmuştur. Uygulama olarak daha önce yapılmış olan çalışmalardan farklı olarak 1 gün, 2 gün ve 20 gün sonraki hisse senedi fiyatlarının kapanışının tahmin edildiğini ve hisse senetlerinin fiyat hareketlerini %72.88'e oranda tahmin başarısı gösterdiği belirtilmiştir.

Fiyat tahmin çalışmalarının kripto para piyasasındaki uygulanmasına dair ise 2016 yılında McNally [24], makine öğrenme tekniklerini kullanarak Bitcoin fiyatını tahmin etme modeli önerisinde bulunmuştur. Uygulama amacının, Bitcoin fiyatının USD cinsinden yönünün ne kadar doğru olabileceğinin tahmin edilmesi olduğunu ve fiyat verilerinin Bitcoin fiyat endeksinden elde edildiği belirtilmiştir. Kullanılan veri seti, 19 Ağustos 2013 ile 19 Temmuz 2016 yılları arasında oluşturulmuştur. Deneysel kısmında, Yinelenen Sinir Ağı (RNN) ve Uzun Kısa Süreli Bellek (LSTM) tekniklerin uygulanmasıyla fiyat tahmin sonuçları elde edilmiştir. Tahmini başarı oranı olarak, LSTM algoritması ile %52'lik en yüksek sınıflandırma doğruluğu ve %8'lik bir ortalama hata karesi (RMSE) elde edilmiştir. Kripto para piyasası fiyat tahminiyle ilgili yapılmış olan bir başka çalışma 2018 yıllında Sakız ve Gencer [25] tarafından, makine öğrenme tekniklerinden yapay sinir ağlarını kullanarak Bitcoin fiyatını tahminleme konusunda bir sunum şeklinde gerçekleştirilmiştir. Kullanılan veri setini, Ocak 2015 ile Nisan 2018 tarihleri arasındaki Bitcoin fiyatlarının günlük kapanış fiyatları alınarak oluşturulmuştur. Uygulama kısmında yaptıkları tahmin sonucu, 2018 Mayıs ayındaki Bitcoin fiyatının 80,955 USD olarak bulunmasıdır. Bulunan sonucun tahmin doğruluğunun performans ölçümü ortalama hata karesi(RMSE) değerine göre bulduklarını belirtmişlerdir. Ama gerçekleşen Mayıs 2018 ortalama fiyatı ise 7,487 USD olmuştur. Tahmin edilen değeri ile gerçekleşen değer arasındaki büyük farklılığın sebebinin Bitcoinin fiyat değişkenin çok olması ve kullandıkları verinin azlığından kaynaklandığını belirtilmişlerdir. 2018 yıllında kripto para fiyat tahminiyle ilgili yapılmış olan bir diğer çalışma da ise Aktepe [26], makine öğrenmesi tekniklerinden yararlanarak kripto para piyasasında fiyat tahminleme ve

kar getirebilecek algoritmalar üzerinde çalışmıştır. Veri kümesi, 1 Eylül 2017 ile 1 Mayıs 2018 tarihleri arasında Binance Coin (BNB)'den aldığı kripto paraların, Açılış-Yüksek-Düşük Kapanış fiyat verileriden oluşmaktadır. Uygulama kısmında, makine öğrenme tekniklerinden hem sınıflandırma hem de regresyon kullanılmıştır. Daha sonra geliştirilen her modelin fiyat tahmin hedeflenmesinde farklı eşik ayarları kullanılarak özellik eşlemesini kullanan bir topluluk öğrenmesi önerilmiştir. Devamında seçili olan kripto para birimlerini birlikte göz önünde bulundurarak, modellerin birbirleriyle ve satın alma / tutma stratejisinin karşılaştırılması ile basit portföy alım satım durumu için başarılı model geliştirilmesi yapılmıştır.

Borsa'da işlem gören hisse senedi fiyat verileri üzerine 2018 yılında Kanmaz [27], Borsa İstanbul'da farklı sektörlerden oluşan bir çok şirkete ait fiyat verileriyle bu şirketlerin ekonomi haberlerinde kullanılması arasındaki bağlantıyı inceleme konusu seçmiştir. Çalışmasında, makine öğrenme teknikleri, doğal dil işleme ve metin madenciliği yöntemlerini kullanarak ekonomi haberlerinin hisse senetleri üzerindeki etkisini incelemiştir. Deneysel kısımda geliştirilen model başarı tahmin oranı, ekonomi haberlerindeki sınıflandırmalara göre %70 oranında bir doğruluk payı vermiştir. Ayrıca, yapılan çalışmanın sonucunda ekonomi haberlerinin olumlu/ olumsuz etkisinin ilgili hisse senedi fiyatları üzerinde de olumlu/olumsuz bir etki yarattığının tespit edildiği belirtilmiştir.

2019 yılında Demirel [28], makine öğrenme teknikleri ve derin öğrenme yöntemleri ile hisse senetlerinin açılış ve kapanış fiyatlarının tahmini konusunu incelemiştir. Veri seti, BIST 100'de işlem gören 42 adet şirketin 1 Ocak 2010 ile 1 Ocak 2019 tarihleri arasındaki verilerinden oluşturulmuştur. Uygulama bölümünde, makine öğrenme tekniklerinden Çok Katmanlı Algılayıcılar (ÇKA), Destek Vektör Makineleri (DVM) ve Uzun Kısa Dönemli Hafıza (UKVH) gibi derin öğrenme metotları kullanılmıştır. Çalışmanın sonucu olarak, ÇKA ve UKVH metotların da DVM'e yöntemine göre daha tutarlı tahminler tespit edilmiştir. Ayrıca, ÇKA ve UKVH metotlarından elde edilen tahmin başarı oranı % 95 güven aralığında gerçekleşmiş, gerçek oranlar ile öngörülen oranlar arasında anlamlı bir farkın

olmadığı tespit edilmiştir. Bir başka çalışma ise 2019 yılında Pabuçcu [29] tarafından, makine öğrenme algoritmaları ile borsa endeks hareketinin yönüne dair tahmin çalışmaları yapılmıştır. Veri kümesi, BİST 100 endeksin 2009 ile 2018 tarihleri aradındaki endeks günlük kapanış fiyatlarından oluşturulmuştur. Uygulama bölümünde makine öğrenme tekniklerinden yapay sinir ağları (YSA), destek vektör makineleri (SVM) ve naive Bayes metotları kullanılmıştır. Sonuç olarak ise, her üç modelinde borsa endeks hareketinin pozitif/negatif yön tahmininde başarılı olduğunu ve yapay sinir ağı (YSA) algoritmasından diğer iki modelden daha performanslı sonuçlar elde edildiği ifade edilmiştir. Bu bölümde inceleyeceğimiz son bir diğer çalışma ise 2019 yılında Akşehir ve Kılıç [30]'a ait, makine öğrenme tekniklerini kullanarak banka senetlerinin fiyat tahmin çalışmalarının yapılmasıdır. Uygulama alanında, banka hisse senetlerinde bir sonraki kapanış fiyatlarının tahmininin yapılmasında bir çok değişkenin varolduğunu ve bu durumun zor bir problem teşkil ettiği ifade edilmiştir. Veri seti, 1 Ocak 2016 ve 9 Mayıs 2019 tarihleri arasındaki Borsa İstanbul BİST 100 endeksinde yer alan 5 büyük bankanın, hisse senetlerinin açılış, kapanış, yüksek/düşük işlem hacim bilgilerinden oluşturulmuştur. Deneysel kısmında, karar ağaçları (DT), çoklu regresyon (MLR) ve rassal olmayan orman (RF) yöntemleri kullanılmıştır. Model başarı ölçütünü, R^2 (R Square) yöntemi ile yaptıklarını ve fiyat tahmin için kullanılan makine öğrenme algoritmalarından başarılı sonuçlar elde edildiği belirtilmiştir.

Bu bölümde makine öğrenme teknikleri ile finans sektöründe yapılmış olan fiyat tahmin çalışmaları genel olarak incelendiğinde, sıklıkla borsa hisse senetlerinde fiyat tahmini yapılması ciddi bir çalışma sahası oluşturmuştur. Makine öğrenme algoritmalarından en çok kullanılanların ise, Yapay Sinir Ağları, Karar Ağaçları ve Çoklu Regresyon yöntemleri olduğu görülmüştür.

2.3. METOT BÖLÜMÜNDE KULLANILAN ALGORİTMALAR İLE İLGİLİ YAPILAN ÇALIŞMALAR

Bu bölümde, fonların fiyat tahmini için model geliştirmesinde kullanılacak olan Kısmi En Küçük Kareler Regresyonu(PLSR), Ridge Regresyonu(RR), Destek Vektör Regresyonu(SVR) ve Yapay Sinir Ağları(YSA) algoritmaları üzerine daha kapsamlı bir literatür incelemesi yapılacaktır. Her bir alt başlıkta birden fazla çalışma inceleme konusu olacaktır.

2.3.1.Kısmi En Küçük Kareler Regresyonu(PLSR) İle İlgili Yapılmış Çalışmalar

Literatür araştırmasının bu kısmında, Kısmi En Küçük Kareler Regresyon(PLSR) algoritması kullanılarak tahmin işlemi yapılmış olan çalışmalar üzerinde durulacaktır. Bu bağlamda 2007 yılında Janik ve arkadaşları[31] tarafından PLS algoritması kullanılarak, toprakta mevcut olan organik karbon fraksiyonlarının konsantrasyonunu tahmin etmek için model geliştirmeleri yapılmıştır. Model geliştirme çalışmalarında kullandıkları veri seti, Avustralya'daki tüm eyaletlerden çok çeşitli toprak türlerini ve ana malzemeleri kapsayacak şekilde, 0-0.50 m katman derinliklerinde değişen 177 toprak çeşitinden alınan numuneler ile oluşturulmuştur. Deneysel uygulamada, PLS algoritmasıyla geliştirilen modelin, R kare (R2) ölçüm metriğine göre % 94 oranında başarı elde ettiği belirtilmiştir. Bir diğer çalışma da 2009 yılında Polat ve Günay[32] tarafından, PLS algoritması üzerine ayrıntılı bir değerlendirme araştırması ve örnek bir uygulama ile yapılmıştır. PLS temel çalışma mantığındaki birden fazla bağımsız değişkenin bir bağımlı değişkeni tahmin etme işlemi olduğunu ama kendi çalışmalarında iki bağımlı değişkenle tahmin işlemlerini yaptıklarını ifade etmişlerdir. Bu bağlamda, PLS ile model geliştirildiğinde birden fazla bağımlı değişken olma durumunda, ayrı modeller mi yoksa tek bir model mi kullanılması gerektiği üzerinde durulmuştur. Veri seti, T.C. Çevre ve Orman Bakanlığı'nın 2006 tarihinde ölçülen hava kirliliğini etkileyen meteorolojik parametre değişkenlerinden oluşturulmuştur. Uygulama bölümünde, model tahmin başarısı ölçüm metriklerinden; hata karelerinin ortalaması(MSE), hata karalerinin ortalamasının karekökü(RMSE) ve R kare (R2) metotları kullanılmıştır. PLS modeli

olarak, bağımlı değişkenlerin ayrı ayrı değerlendirilmesi daha uygun bulunmuş ve tahmin başarı derecesinin ise daha iyi olduğu ifade edilmiştir. PLS algoritmasıyla yapılan bir başka çalışma da ise 2012 yılında Taşkın ve arkadaşları[33], yumurtalık kanseri verilerini kullanarak kanser teşhisine dair çalışmalar yapmışlardır. Veri kümesi, FDA-NCI Clinical Proteomics Program Databank veri tabanından alınan 216 örnek yumurtalık kanser verisinden oluşturulmuştur. Bu verilerin, 95 adeti test veri grubu, 121 veri adeti ise yumurtalık kanseri teşhisi konulmuş bilgilerden oluşmaktadır. Deneysel çalışmalarda, PLS algoritması, temel bileşen analizi(TBA) ve diverjans analizi(DA) metotları kullanılarak boyut indirgeme işlemi yapılmıştır. Daha sonra, doğrusal diskriminant analizi(LDA) yöntemi ile sınıflandırma yapılarak metot sonuçları karşılaştırılmıştır. Optimum sonuç olarak, PLS algoritmasıyla diğer yöntemlere göre daha iyi sonuçlar elde edildiği ifade edilmiştir.

Bu bölümde son olarak inceleyeceğimiz bir diğer araştırma ise 2013 yılında Serrano-Cinca ve Gutiérrez-Nieto[34] tarafından, PLS algoritması kullanılarak 2008 yılındaki ABD bankacılık krizinin tahmini için bir çalışma yapılmıştır. Veri seti, Federal Mevduat Sigortası Şirketi'nin (FDIC) veri tabanından alınan ve internette herkese açık olan 2008 yılına ait 8,293 bankanın hesap özeti bilgilerinden oluşturulmuştur. Uygulama olarak ise, PLS ile bankacılık kriz tahmininde yaygın olarak kullanılan 8 algoritmanın performansı ve karşılaştırılması yapılmıştır. Performans ölçüm metriği olarak, doğruluk(accuracy), kesinlik(precision), F-skoru(F-score), Tip I (Type I) ve Tip II (Type II) hata sonuç değerleri incelenerek; hiçbir algoritmanın diğerlerinden daha iyi performans göstermediği ifade edilmiştir. Ayrıca, PLS, doğrusal diskriminant analizi ve destek vektör makineleriyle elde edilen sonuçların birbirine çok yakın olduğu da belirtilmiştir.

2.3.2.Ridge Regresyon(RR) İle İlgili Yapılmış Çalışmalar

Bu bölüm de Ridge Regresyon(RR) algoritması kullanılarak yapılmış çalışmalar incelenecektir. Bu kapsamda 2010 yılında Büyükuysal[35] tarafından, RR

algoritması hakkında ayrıntılı bir araştırma hazırlayarak örnek bir uygulama yapılmıştır. Uygulama veri kümesini, İstanbul Üniversitesi Tıp Fakültesi Hastanesine gelen obezite hastaları arasından rastgele seçilen 20 kişinin; benden ağırlığı, deri alanı, uyluk kemiğinin uzunluğu ve kas çevrelerinin uzunluğu bilgilerinden oluşturulmuştur. Deneysel çalışma da ise, PLS ve RR metotlarını kullanarak bağımlı değişken olan beden ağırlığının tahmin sonuçları karşılaştırılmıştır. Tahmin başarısı olarak, RR ile elde edilen değerlerin daha doğru sonuçlar verdiği ifade edilmiştir. Bir başka çalışma ise 2011 yılında Çekerol ve Nalçakan[36] tarafından, lojistik sektörü içindeki Yurtiçi Kargo şirketinin yük taşıma talebinin RR algoritmasıyla tahmin edilmesidir. Veri seti, TCDD İstatistik Yıllıklarından alınan 1990-2009 tarihleri arasındaki Yurtiçi kargo yük taşıma talebine ait verilerden oluşturulmuştur. Uygulama analiz kısmında, geliştirilen model başarısı R kare (R2) ölçüm metriğiyle 0,64 değeri bulunarak anlamlı bir modelin oluşturulduğu tespit edilmiştir. Model sonuçları olarak, yurtiçi yük taşıma miktarlarına dair yol gösterici anlamlı veriler elde edildiği belirtilmiştir.

RR ile geliştirilen model tahmin uygulamalarından bir diğeri ise 2019 yıllında Küçük[37] tarafından, doğrusal regresyon algoritmalarından RR, LIU ve LASSO ile tahmin yapma üzerine bir inceleme çalışmasıdır. Veri kümesi olarak, Hald veri seti ve Türkiye'de 1998 ile 2006 tarihleri arasındaki istihdam verilerinden oluşturulan iki farklı veri seti ile deneysel çalışmalar yapılmış. Uygulama sonuçları olarak, 3 algoritma için hata kareleri ortalaması ile karşılaştırılması yapılmıştır. Hald veri setine göre, LIU ve LASSO metotları ile daha optimize değerler elde edilmiş ama alternatif olarak RR algoritmasının kullanılabilir olduğu da belirtilmiştir. İstihdam veri seti için de aynı sonuçların geçerli olduğu ifade edilmiştir. Son olarak bu bölümde değerlendirilmesi yapılacak bir başka çalışma ise 2019 yıllında Ayan ve arkadaşları[38] tarafından, Twitter üzerinden atılan twitlerin islamofobik açıdan duygu analizi ile tespit edilmesi hakkında bir inceleme çalışmasıdır. Uygulama veri kümesi, 1 Ağustos 2018 ile 5 eylül 2018 tarihleri arasında, belirli bir anahtar sözcükler setine göre atılan 162,000 farklı twit verilerinden toplanmıştır. Deneysel

bölümde, RR ve Naive Bayes metotları kullanılmış ve model tahmin sonuçlarının karşılaştırılması için doğruluk(accuracy), kesinlik(precision), F-skoru(F1) ölçüm metrikleri ile değerlendirilmeleri yapılmıştır. Model başarısı olarak ise, RR algoritmasıyla F1 değerine göre en yüksek sonuç % 96.9 oranında elde edilmiştir.

2.3.3.Destek Vektör Regresyonu(SVR) İle İlgili Yapılmış Çalışmalar

Bu bölüm de, Destek Vektör Regresyon(SVR) algoritmasıyla ilgili daha önceki bölümlerde incelenenlerden farklı iki çalışma değerlendirilecektir. Bir önceki bölüm de (2.1 makine öğrenme ile yapılan genel çalışmalar) ve bölüm 2.2'deki finans sektöründe fiyat tahminine dair yapılan araştırmalar da sıklıkla kullanılan algoritmalardan biri de destek vektör makinaları (SVM) metodunu içermektedir ve bu kapsamda metot bölümünde kullanmak istediğimiz SVR algoritmasına dair yeterli düzeyde literatür araştırması sağlanmış olmaktadır.

SVR algoritması kullanılarak 2012 yılında Uçak[39], PID kontrolör tasarımı hakkında bir yüksek lisans tez çalışması yapmıştır. Tez çalışmasında, SVM uygulama alanları olan sınıflandırma ve regresyon problemlerinde kullanıma dair detaylı açıklamalar yapılmıştır. Daha sonra, model tanımlama problemi üzerine SVR metodu ile başarılı sonuçlar elde ettiğini ve kurulan tahmin tabanlı model uygulamasında 4'lü tank sistem dinamiklerinin de başarılı bir şekilde tahmin edildiğini belirtmiştir. Son olarak, SVR ile yapılan bir diğer çalışma ise 2017 yıllında Ekinci[40] tarafından, hava kirliliğini tahmin edecek bir model önerisi ile yüksek lisans tez çalışması olarak yapılmıştır. Veri seti, Şubat 2005 ile Mayıs 2015 tarihleri arasında Denizli ili için ABD Ulusal Okyanus ve Atmosfer Dairesi(NOAA) veri tabanından alınan bilgilerden oluşturulmuştur. Deneysel çalışma da, model tahmin sonuçlarının karşılaştırılması için SVR algoritmasına ek olarak yapay sinir ağları(YSA) metoduyla da tahmin işlemleri yapılmıştır. Model sonuçlarının karşılaştırma ölçüm metrikleri olarak, hata kareleri ortalaması ve hata karelerinin ortalamasının karekökü alınarak model değerlendirilmesi yapılmıştır. Model tahmin başarısı ise, SVR ile elde

edilen sonuçların YSA'ya göre daha başarılı olduğu şeklindedir. Ayrıca, model sonuçlarına dair iki modelin avantajları, dezavantajları ve model parametre optimizasyonları hakkında ayrıntılı incelemelerin de yapıldığı belirtilmiştir.

2.3.4.Yapay Sinir Ağları(YSA) İle İlgili Yapılmış Çalışmalar

Literatür araştırmasının son bölümü olan Yapay Sinir Ağları (YSA) ile ilgili yapılmış çalışmalar incelendiğinde, daha önceki bölümlerin (2.1 makine öğrenme tekniklerinin uygulamalarına dair yapılan genel araştırmalar ve bölüm 2.2 finans sektöründe fiyat tahminine dair yapılan araştırmalar) çoğunda, YSA algoritmasıyla yapılan çalışmaların diğer algoritmalara kıyasla daha ağırlıklı olduğu görülmüştür. Bu çalışmaların büyük bir kısmı [11,12,13,14,19,21,23] doğrudan YSA model geliştirme ve tahmin araştırmalarını içermektedir. Bu sebeble tekrara düşülmemesi amacıyla YSA ile ilgili yeterli düzeyde çalışmanın incelendiği görülmektedir. Yapılan araştırmaların sonucuna bakıldığında da, regresyon uygulamaları olmak üzere, YSA algoritmasının makine öğrenme uygulamalarının genelinde sıklıkla kullanıldığı gözlemlenmektedir. Bu kapsamda, çalışma içerisinde fonların fiyat tahmini için geliştirilen modelin, regresyon probleminin olması, uygulamada kullanılacak algoritmaların bu problem türüne uygunluğu önem arz etmektedir. Literatür araştırma sonuçlarına da bakıldığında YSA algoritmasının tercih edilmesi için gerekli argümanları yeterli düzeyde sağladığı incelenmiştir.

Sonuç olarak literatür çalışması genel olarak değerlendirildiğinde, araştırmalarda görülmektedir ki tahmin çalışmalarında YSA, SVR, PLSR ve RR algoritmalarının sıklıkla kullanıldığı incelenmiş ve spesifik olarak da finans sektöründe fiyat tahmin çalışmalarında YSA ve SVM metotlarının kullanıldığı gözlemlenmiştir. Ayrıca, regresyon modellerinin sonuçlarının karşılaştırılması için ise ölçüm metrikleri olarak MSE, RMSE ve R2 metotları kullanılmıştır. Yapılan literatür araştırmasıyla, tez çalışmasının model geliştirme bölümünde kullanılması düşünülen 4 algoritmanın neden tercih edildiği ve model sonuçlarını karşılaştırmak

için kullanacağımız MSE, RMSE ve R2 ölçüm mertiklerin de neden kullanılacağı anlaşılır kılınmıştır.

3. METOT: KULLANILAN MAKİNE ÖĞRENME ALGORİTMALARI

Bu bölümde makine öğrenme yöntemlerinden gözetimli öğrenim metotlarının regresyon algoritmalarını inceleyeceğiz. Regresyon, bir değişkenin bir veya daha fazla değişkenle arasındaki bağlantının matematiksel bir fonksiyonla gösterilmesidir[41]. Yani bağımlı(y), bağımsız(x) değişkenleri düşünüldüğünde y'nin, x'in bir fonksiyonu olarak ifade edilen ilişki biçimi regresyon olarak tanımlanır. Bu durum değişkenler arasında bir sebep-sonuç bağlantısını özetlemektedir. Matematiksel fonksiyon durumuna göre doğrusal ve doğrusal olmayan regresyon çeşitleri bulunmaktadır[42]. Bu çalışma da fonların fiyatlarının nasıl tahmin edileceği sorusu bağlamında değerlendirildiğinde, veri setindeki bağımlı değişken olarak fon fiyatlarının olması ve birden fazla bağımsız değişken değerin bulunması kapsamında regresyon algoritmalarının kullanılması daha uygun bulunmuştur. Regresyon metotlarından doğrusal çoklu regresyon algoritmalarından Kısmi En Küçük Kareler ve Ridge Regresyon ile doğrusal olmayan çoklu regresyon algoritmalarından ise Destek Vektör Regresyonu ve Yapay Sinir Ağları kullanılacaktır.

3.1. DOĞRUSAL ÇOKLU REGRESYON

Birden fazla bağımsız değişken ile bir bağımlı değişken arasındaki doğrusal bağlantıyı incelemektedir. Doğrusal çoklu regresyonun matematiksel olarak gösterimi ise;

$$Y_i = (b_0 + b_1X_1 + b_2X_2 + \ldots + b_iX_i) + e_i \quad \textbf{(3.1)}$$

olarak ifade edilir.

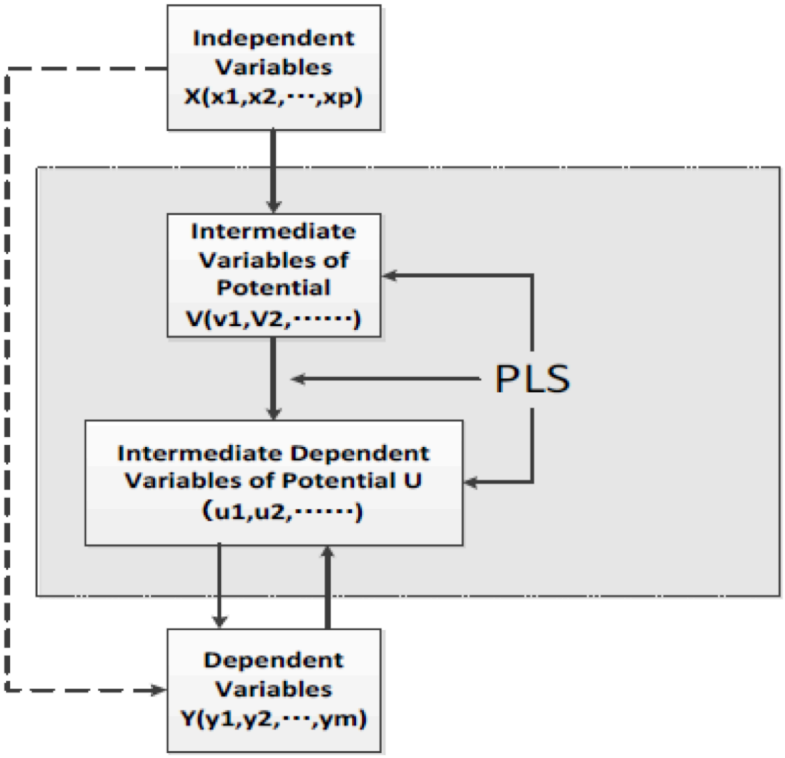
3.1.1. Kısmi En Küçük Kareler Regresyonu (PLSR)

3.1.1.1. Teori

Kısmi En Küçük Kareler Regresyonu, 1960'larda Herman Wold tarafından bir ekonometri tekniği olarak geliştirilmiş ve temel bileşen analizi ile çoklu regresyon özelliklerini genelleştiren ve birleştiren yeni bir teknik olmaktadır. PLS yöntemi, bağımsız(X) değişken grubunun çok fazla olması ve bir dizi bağımlı(Y) değişken tahmin etmemiz gerektiğinde kullanılmaktadır. Teknik olarak ise, X ve Y değişkenleri arasındaki gizli bağlantıdan yararlanarak, veri blokları arasındaki ilişkiyi bulmayı amaçlamaktadır [43]. Ayrıca PLS çalışmaları ilkin, ekonomik ve sosyal durumları modelleme için kullanılmıştır. Fakat yaygın olarak kullanımına oğlu Svante Wold tarafından kemometrik alanındaki çalışmalarla başlanmıştır [44]. Kemometri, istatistik ve matematik tekniklerinin kullanılarak, kimyasal sistemlerden bilgi alma bilimidir. Kimyasal analizlerde, veriden doğru bilginin ya da gizli bilginin açığa çıkarılmasına imkan tanıyan bir araçtır[32]. İstatistik alanındaki ilk kullanımına ise 1988 yılında Höskulddson [45] ve 1989 yılındaki Naes ve Marten [46] tarafından yapılan çalışmalar örnek verilebilir.

3.1.1.2. Matematiksel Model ve Algoritma

Bu kısımda PLS yöntemin temel matematiksel diyagramından bahsedilecektir. PLS için kullanacağım algoritma ise deneysel çalışma bölümünde fonların fiyat tahmin için geliştirilecek olan model algoritması olarak kullanılacaktır.



Şekil 3.1: PLS Matematiksel Diyagramı[47]

PLS matematiksel diyagram şemasından anlaşılacağı üzere, PLS genel olarak potansiyel bağımsız değişkenler ile potansiyel bağımlı değişkenler arasındaki doğrusal ilişki kurmayı amaçlamaktadır. M tane bağımlı değişkenler Y (y1, y2, y3, …, ym) ile p tane bağımsız değişkenler X (x1, x2, x3,…, xp) arasındaki ilişkiyi ise dolaylı olarak yansıtır. Potansiyel bağımsız değişkenler ve potansiyel bağımlı değişkenler, PLS regresyonundaki değişkenlerin doğrusal kombinasyonunu yansıtır ve iki varsayımı mevcuttur:

1. İki potansiyel değişken grubu, bağımsız değişken veya bağımlı değişkenler mutasyon bilgilerini taşımaktadır.

2. Potansiyel değişkenler arasındaki korelasyon değeri maksimize edilmektedir[47].

Bağımlı Değişken Y ve Bağımsız Değişken X için PLS regresyon matematiksel gösterim şeması şu şekildedir;

$$\mathbf{X} = \sum_{j=1}^{A} \mathbf{t}_j \mathbf{p}'_j + \mathbf{E} \text{ ve } \mathbf{Y} = \sum_{j=1}^{A} \mathbf{u}_j \mathbf{q}'_j + \mathbf{F}$$

Şekil 3.2: Bağımlı ve Bağımsız Denklem[48]

Yukarıdaki matematiksel gösterimde $\mathbf{t_j}$, $\mathbf{u_j}$ ler gizli değişkenler olup, $\mathbf{t_j}$ değişkenleri birbirlerine ve aynı bağlamda sonraki $\mathbf{u_j}$ değişken değerine diktir. PLS model için maksimum sayıda gizli değişken sayısı, bağımsız ve bağımlı değişkenler arasındaki kovaryans değeri maksimum olacak şekilde elde edilir[48].

Matematiksel model anlatımından sonra bu çalışmada kullanılacak PLS algoritması klasik ve standart olan NIPALS (Non-Linear Iterative Partial Least Squares; Doğrusal olmayan yinelemeli kısmi en küçük kareler) algoritması olacaktır. NIPALS bağımsız değişken değerleri birden fazla olan ve bağımlı değişken değeri tekil olan veri setleri için güçlü ve sağlam bir algoritma olduğundan dolayı tercih edilmiştir. Temel de tüm PLS algoritmalarının amacı kovaryans matrislerini en fazla sayıya ulaştıracak bileşenlerin elde edilmesidir[49].

NIPALS algoritmasının adımları;

j= 1,2,…J, bileşen sayısını gösterir.
$\mathbf{X_1} = \mathbf{X}$, $\mathbf{Y_1} = \mathbf{Y}$, orijinal matrisler.

1. Veri setimizde bağımlı değişken sayısı tek olduğu için direkt **Y** değişken sütunu $\mathbf{u_j}$ vektörü olarak tanımlanır.

2. **X** ve **Y** bileşenlerinin $\mathbf{u_j}$ üzerinedeki regresyonu **X** ve **u** arasındaki kovaryans değerini en çoklayan **w** ağırlık vektörü $w_j = X'_j u / (u'_j u)$ ile elde edilir.

3. $\mathbf{w_j} / ||\mathbf{w_j}||$ ve $\mathbf{w_j}$ vektörü normuna bölünüp vektör boyu 1 buluncak şekilde ölçeklendirilir.

4. $t_j = X_j w_j$ eşitliği **X'**in bileşeni $\mathbf{t_j}$, $\mathbf{w_j}$ ise ağırlık vektörü ile **X** 'in değerinin bir kombinasyonu olacak formda hesaplanır.

5. $\mathbf{t_j}$ bileşen değerinin **Y** 'yi modelleme değerini açıklayan $\mathbf{c_j}$ ağırlık vektörü ise $c_j = Y'_j t_j / (t'_j t_j)$ ile **Y** 'nin $\mathbf{t_j}$ üzerindeki regresyon değeri bulunur.

6. $\mathbf{c_j}$ vektörü norm değerine bölünerek boyu 1 bulunacak şekilde ölçeklendirilir. $\mathbf{c_j} / ||\mathbf{c_j}||$ şeklinde hesaplanır.

7. **Y** değerinin ilgili bileşeni $\mathbf{u_{j(yeni)}}$, $\mathbf{c_j}$ ağırlık vektörü ve **Y** 'nin doğrusal kombinasyonu ise $\mathbf{Y_j c_j / (c'_j c_j)}$ şeklinde hesaplanır.

8. İkinci adımda kullanılan $\mathbf{u_j}$ değeri ile yedinci adımda kullanılan $\mathbf{u_{j(yeni)}}$ değerileri arasında bir benzerliğin olup olmadığına bakılır. Bu benzerlik, iki vektörün fark normu $\mathbf{10^{-6}}$ gibi çok düşük bir değer olması sağlanır. Bu benzerlik oranı sağlanır ise bir sonraki adıma geçilerek algoritma sonlandırılır, yoksa yedinci adımda bulunan $\mathbf{u_{j(yeni)}}$ değeri ikinci adımdaki yerine yazılarak algoritmaya devam edilir.

9. **X** bileşeni $\mathbf{t_j}$ üzerine regresyonu, bileşen değerinin açıklayıcı değişken üzerindeki etki faktörünü gösteren yük vektörü $\mathbf{p_j}$, $X'_j t_j / (t'_j t_j)$ ile elde edilir.

10. **Y** bileşeni $\mathbf{u_j}$ üzerine regresyonu, bileşen değerinin bağımlı değişken üzerindeki etki faktörünü gösteren yük vektörü $\mathbf{q_j}$, $Y'_j u_j / (u'_j u_j)$ ile elde edilir.

11. Hem **X** hem de **Y** için bileşen değerleri ayrı ayrı hesaplandığında değerler arasında zayıf bir ilişki bulunduğu görülmektedir. Bu zayıf değer ilişkisinin kaldırılması için her bir bileşen değeri için **Y** 'nin bileşeni $\mathbf{u_j}$ 'nın **X** 'in bileşeni $\mathbf{t_j}$ üzerine regresyon değeri bulunan iç bir $\mathbf{b_a}$ katsayısı $b_a = u'_j\, t_j \;/\; (t'_j\, t_j)$ şeklinde hesaplanır.

12. Bulunan bileşen değerleri, yüklerin bağımlı değerleri ve açıklayıcı değişkenleri modellemenin yapılmasında kullanılır. Bileşenler açıklayıcı ve bağımlı değişken $X = TP'$ ve $Y = BTC'$ şeklinde modellenir. Bu aşamdan sonra algoritmanın bir sonraki bileşenlerini elde etmek için kullanılacak olan $\mathbf{X_{j+1}}$ ve $\mathbf{Y_{j+1}}$ artık matrisleri $X_{j+1} \rightarrow X_j - t_j p'_j$ ve $Y_{j+1} \rightarrow Y_j - b t_j\, c'_j$ şekilde hesaplanmaktadır[50,51].

NIPALS algoritmasının 12 adımda olaşan çalışma akışına göre açıklayıcı ve bağımlı değişkenlerdeki değişimin büyük bir kısmı açıklık elde edilene kadar devam etmektedir. Yani X bileşenindeki değer matrisinin sıfır matrisi oluncaya kadar algoritma çalışır. Ayrıca, algoritma hesaplama sürecinde, ihtiyaç duyulacak en az sayıda bileşen değerini vermektedir[50].

3.1.2. Ridge Regresyonu (RR)

3.1.2.1. Teori

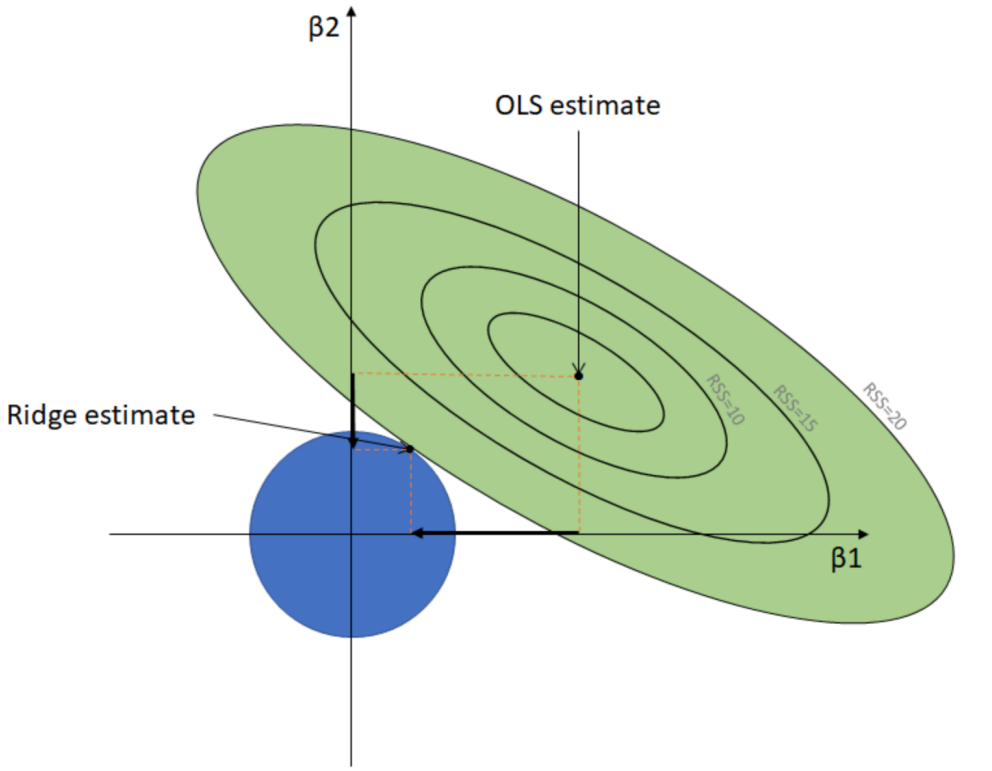
Ridge Regresyonu, 1970 yılında Hoerl ve Kennard tarafından bağımsız değişkenler arasındaki doğrusal yada doğrusala yakın ilişki biçimi olan çoklu doğrusal bağlantı problemini azaltmak için geliştirilen istatistik yöntemidir. Çoklu doğrusal bağlantı problemindeki en önemli olan nokta ise, regresyon katsayılarındaki kovaryans ve varyans sayılarının sonsuza doğru büyümesini sağlayan bir artış göstermesidir. RR yöntemi teknik olarak, X'X gibi bir korelasyon matrisinin köşegen değer elemanlarına, $k > 0$ yanlılık parametre değerlerini ekleyerek ve gerekli olan koşul sayısını azaltarak, çoklu bağlantı sorunundaki kovaryans ve varyans

problemini çözmeyi hedeflemektedir[52]. Kısaca RR'nin amacını tanımlamak istersek, en küçük kareler (EKK) yöntemi ile bulunan kareler toplamını minimize etmek için regresyon katsayılarına bir ceza değeri uygulanmasını sağlayarak sonuç bulmaya çalışmaktadır. Ayrıca, EKK yönteminin yetersiz kalması dolayısıyla RR yöntem geliştirilmesi yapılmıştır[53]. Aşağıda maddeler halinde RR yöntem önerileri sıralanmıştır.

1. Modelin aşırı öğrenme yapısına karşı dirençli olmasını sağlar.
2. Yöntem yapısı olarak yanlıdır fakat varyans değeri düşüktür. (Bazen ise yanlı olan modeller daha çok tercih edilir.)
3. Çok fazla parametre olduğunda EKK'ya göre daha iyi sonuçlar verir.
4. Çok boyutluluk problemine karşı çözüm sunar.
5. En önemlisi ise çoklu doğrusal bağlantı sorununa karşı oldukça etkili bir yöntemdir.
6. Tüm değişkenlerle model kurar. İlgisiz değişkenleri modelden çıkarmaz, sadece katsayılarını sıfıra yakınlaştırır[52-54].

3.1.2.2. Matematiksel Model ve Algoritma

RR yönteminin teorik anlatımından sonra model ve algoritma anlatımlarını yapacağımız bu kısım da, önce RR ile EKK tahmin yapısının iki boyutlu şekil diyagramından ve matematiksel olarak hesaplanma formülünden bahsedilecektir. En son olarak ise RR algoritmasının EKK yönteminden farkılılaştığı ceza terimi katsayısı olan k (yada λ) adımları yazılacaktır.



Şekil 3.3: İki boyutlu uzayda RR ve OLS(EKK) kestiricisinin gösterimi[55]

İki boyutlu görselden de anlaşılacağı üzere, RR ile En Küçük Kareler Yöntemi (Ordinary Least Squares (OLS)) olan EKK arasındaki tahmin değerlerindeki farkın formunu açıklamaktadır. RR , $\hat{\boldsymbol{\beta}} = (\hat{\beta}_1, \ldots, \hat{\beta}_p)$ tahmin değerlerini yaklaşık sıfıra doğru çekerek verideki en iyi tahmin değerlerini elde etmiştir. Yani, $\text{RSS(k)} = \sum_{i=1}^{n} (Y_i - X_i\hat{\beta})^2$ hata kareler toplamı minimum yapılırken, aynı şekilde RR için ceza katsayı terimi olarak hesaplanan $k\sum_{j=1}^{p} \hat{\beta}_j^2$ minimum olacak değerleri yakalayarak modeldeki veriyi oldukça iyi kuracak tahminler bulmaya çalışmaktadır. Şekildeki RR ile EKK tahminleri için örnek verilirse, $\hat{\boldsymbol{\beta}} = (\hat{\beta}_1, \hat{\beta}_2)'$ gibi iki boyutlu uzayda RR tahmini, $RSS(k)$ çemberlerinin $\beta_1^2 + \beta_2^2 = c_2^2(k)$ ile tanımlanan değerleri karşıladığı noktadır. Yukarıdaki şekil, çemberlerin eşit $RSS(k)$'lı β'ların değerler bilgisini göstermektedir. Ayrıca, RR ise lacivert çember içerisinde yer alan en optimum değerleri bulmaya çalışmaktadır[56].

Bu bağlamda RR matematiksel modeli aşağıdaki şekilde özetlenebilir;

$$RR = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 + k \sum_{j=1}^{p} \beta_j^2$$

Şekil 3.4: RR matematiksel gösterimi

Formüldeki ceza terimi katsayısı olan k ayar parametresi başka çalışmalarda ise λ lambda simgesi olarakta gösterilmektedir.

RR algoritmasının ceza parametresi olan k yada λ adımları;

1. Ceza parametresi için bir faz aralığı tanımlanır.

2. Veri seti , {1,…,n} \ i ve {i} içerecek şekilde örnek eğitim ve test setine sırasıyla bölünür.

3. Eğitim seti cross-validation yöntemi kullanarak her bir λ parametresi için RR ile tahminler yapılır. Aşağıdaki şekilde λ için RR tahmin hesaplama işlemleri şu şekilde yapılır;

$$\hat{\beta}_i(\lambda) = (X_i^T X_i + \lambda I_{pp})^{-1} X_i^T Y_i \qquad \textbf{(3.2)}$$

   Ayrıca, hata varyansı tahmini ise: $\hat{\sigma}_i^2(\lambda)$ olarak hesaplanır.

4. Her bir örnek test seti seçilip işlem yapılacak şekilde algoritmanın 1 ile 3.adımları arasında tekrarlanır.

5. Ceza parametresinin cross-validation yöntemi ile bulunan her bir test setinin tahmin performansları ortalaması aşağıdaki yöntemle yapılmaktadır;

$$\frac{1}{n} \sum_{i=1}^{n} log(L[Y_i X_i; \hat{\beta}_i(\lambda) \hat{\sigma}_i(\lambda)]) \qquad \textbf{(3.3)}$$

   Bu hesaplama ile modelin, ceza parametresinin yeni veriler üzerindeki değerine karşılık gelen tahmin sonuçları bulunur.

6. Son olarak cross-validation yönteminin log olasılığını en üst düzeye çıkaran ceza parametresinin değeri, seçim değeri olarak bulunur[57-58].

RR ceza parametresinin seçilmesini sağlayan yukarıdaki adımlardan anlaşılacağı üzere, 1'den 6.adıma kadar çalışma modeli verilen algoritma adımları çalışılarak en optimum k yada λ değeri bulunmaktadır.s

3.2. DOĞRUSAL OLMAYAN ÇOKLU REGRESYON

Birden fazla bağımsız değişken ile bir bağımlı değişken arasındaki ilişkinin doğrusal olmayan formunu incelemektedir. Doğrusal olmayan regresyon değişkenler arasındaki ilişki biçimini bir fonksiyonla modelleştirir. Aşağıda ki doğrusal olmayan çoklu regresyonun matematiksel gösterim örneklerinden biri ise polinom regresyon fonksiyonu şu şekilde ifade edilir;

$$Y_i = \beta_0 + \beta_1 X + \beta_2 X_i^2 + \ldots + \beta_h X_i^h + e_i \qquad \textbf{(3.4)}$$

3.2.1. Destek Vektör Regresyonu (SVR)

3.2.1.1. Teori

Destek Vektör Makineleri(SVM), 1995 yıllında Vladamir Vapnik tarafından büyük boyutlu ve az sayıda eğitim verisinden öğrenebilen yeni bir yaklaşım metodu olarak, sınıflandırma ve regresyon analizleri için kullanılması önerilen bir gözetimli makine öğrenme algoritmasıdır [39].SVM matematiksel model algoritması, başlangıçta sınıflandırma problemleri için geliştirilmiştir. Sınıflandırma probleminde varolan iki sınıf verisinin, birbirinden ayırt edilebileceği en uygun tahmin fonksiyonunu bulmaya çalışmaktadır. Bir diğer ifade şekli ile mevcut verideki iki sınıfı en optimum şekilde ayırabileceği hiper-düzlem tanımlanmasına dayanmaktadır.

Destek Vektör Regresyonu (SVR) ise, sınıflandırma problemlerinin genelleştirilmiş formu olarak tanımlanabilir. SVR'nin temel fikri, veri setindeki eğtim hatasının genelleştirme üst sınırını azaltacak şekilde riski minimize eden bir tahmin fonksiyonu oluşturmaktır. Diğer bir deyişle, SVR modeli sürekli-değerler için çok değişkenli bir fonksiyonu tahmin etmektedir [59]. SVM'nin diğer algoritmalara göre üstünlüğü ise, sınıflandırma ve regresyon problemlerindeki lokal minimuma takılma gibi bazı dezavantajlı durumları bertaraf eden bir yöntem olmasıdır.

SVM algoritmasının avantajları aşağıdaki gibi sıralanabilir;

1. Modelin basit bir yapıya sahip olmasına rağmen güvenilir sonuçlar elde etmesi,
2. Güçlü fonksiyon genelleştirme becerileri ve optimum değerlere ulaşması,
3. Model çalışmalarında yüksek boyutlu uzaylarda etkili olması,
4. Model mimari tasarımında çok az kontrol parametresi kullanılması (gerekli olan sınırlı efor için)

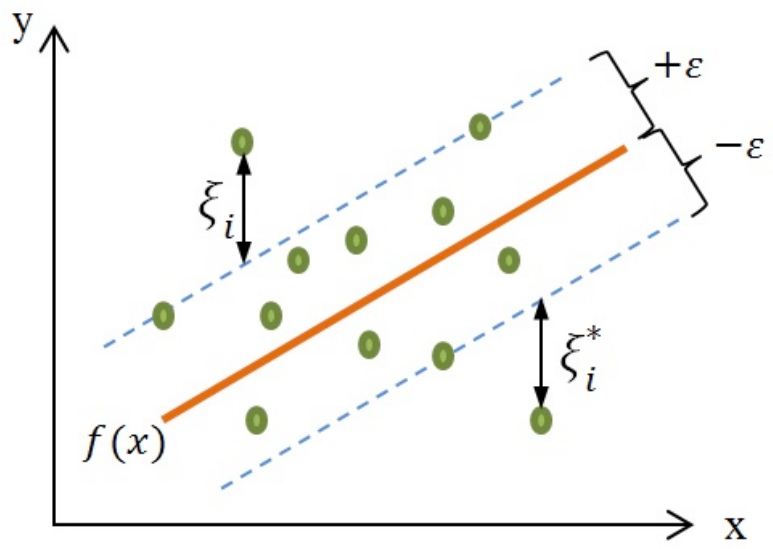
Bu sebeplerden dolayı SVM, makine öğrenme tekniklerinden tahmin etme için kullanılan en popüler algoritmalardan birisi olmuştur [60].

Son olarak SVM kullanımına dair, genelleştirebilme özelliğinin çok yüksek olması, model yapısının güçlü ve uygulamalarda yüksek bir performans gösteriyor olmasından dolayı son yıllarda hava durumu, enerji piyasası, hisse fiyatı, ses ve görüntü tanıma gibi bir çok alanda tahmin çalışmaları için kullanılmakta olduğu belirtilmelidir [61]. Ayrıca, SVM modelinin tahmin edilen değerler için güvenilir bir araç olması ve finansal zaman serilerinin tahmin edilmesinde de başarılı sonuçlar elde ettiği belirtilmiştir [62].

3.2.1.2. Matematiksel Model ve Algoritma

Teorik kısımda anlatıldığı üzere SVM algoritmalarının regresyon problemlerinde başarılı sonuçlar elde etmesi ve tahmin problemlerinde sıkça kullanılması, SVR algoritmalarını tercih etmemizi sağlamaktadır. Bu bölümdeki çalışmamızda SVR model yapısı üzerinde durulacaktır. İlk olarak, SVR matematiksel

modelin temelini oluşturan ve anlaşılması daha basit olan SVR'nin doğrusal matematiksel fonksiyonundan bahsedilecektir. İkinci olarak SVR ağ yapısı ayrıntılı incelenecek ve son olarak ise model çalışmamız için kullanılacak olan algoritmanın yapısı ve adımlarından bahsedilerek bu bölüm sonlandırılacaktır.



Şekil 3.5: Doğrusal SVR örnek gösterimi[63]

Doğrusal SVR'nin matematiksel olarak gösterimi olan örnek bir $f(x) = wx + b + \epsilon$ fonksiyonu ile doğrusal ya da doğrusal olamayan bir eğri ile fonksiyon modeli bulunur iken, aşağıdaki minimizasyon problemi ile en optimum fonksiyon modeli bulunmaya çalışmaktadır.

Minimizasyon denklemi,

$$\frac{1}{2}||w||^2 + C\sum_{i=1}^{m}(\epsilon_i + \epsilon_i^*) \quad \textbf{(3.5)}$$

olacak sekilde aşağıdaki belli kısıtlar çerçevesinde;

$\epsilon_i, \epsilon_i^* \geq 0 \quad i = 1,...,m \quad \epsilon = Epsilon, \epsilon^* = Kisi$

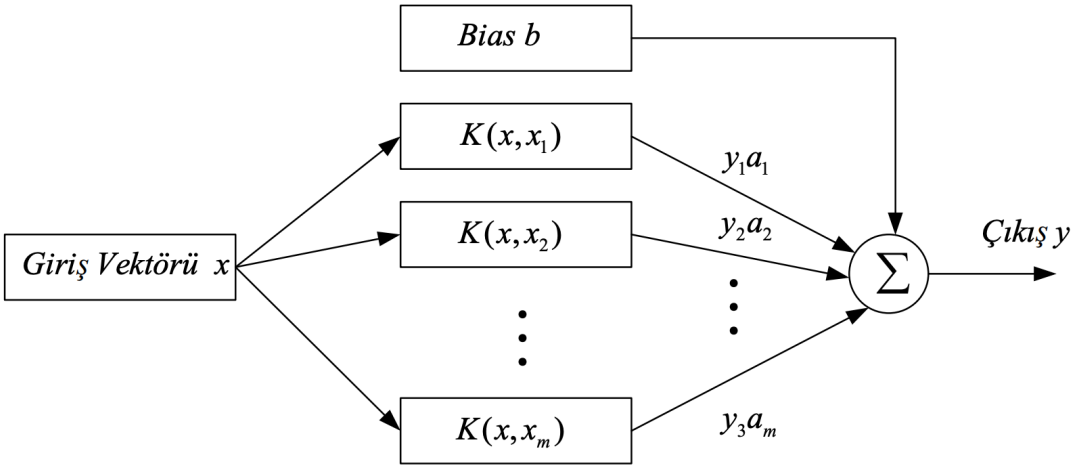
$$y_i - (w * x_i) - b \leq \epsilon + \epsilon_i \quad \textbf{(3.6)}$$

$$(w * x) + b - y_i \leq \epsilon + \epsilon_i^* \quad \textbf{(3.7)}$$

(3.6) ve (3.7) denklem kısıtlarına göre en optimum regresyon denklemi bulmaya çalışırken, gerçek değerler ile tahmin edilen değerler arasındaki farklar, regresyon eğrisinin iki yönünde belirli bir epsilon ve kisi değerlerinden daha uzakta olmayacak şekilde bulunur. Kısıtların denklem üzerindeki etkisini bu şekilde özetleyebiliriz. Ayrıca, (3.5)'te verilen minimizasyon denklemindeki $\boldsymbol{C}$ parametresi, karmaşıklık ya da ceza değerleri olarak ifade edilir. Bu ceza terimi, epsilon ve kisi artıkları üzerinde bir kontrol mekanizması işlevi görmüş olmaktadır.

Genel olarak ifade edilirse, şu ana kadar anlatılan matematiksel model kısıtları ve yukarıdaki şekil 5' te gösterilmekte olan regresyon eğrisini bulmaya çalışırken ki temel amaç, belirli bir marjın aralığına maksimum noktayı en küçük hata ile alabilmesini sağlayacak şekilde bir doğru ya da eğri belirlemektir.

SVR modelin bir diğer önemli kısmı olan ağ yapısını inceleyeceğiz,



Şekil 3.6: SVR model ağ yapısı[40]

Model ağ yapısına göre doğrusal olmayan SVR tahmin fonksiyonu,

$$f(x_i) = w^t \,.\, K(x_{x,x_i}) + b \qquad \textbf{(3.8)}$$

olarak gösterilir. Bu aradaki temel amaç, x_i girişine çıktı y_i'nin bağımlılığını ifade eden (3.8) denklem fonksiyonu tanımlamaktır. Fonksiyonun temel bileşenleri olan w ağırlık vektörü , b eğilimi katsayısı ve $K(x, x_i)$ çekirdek fonksiyon ile çıkış değeri olan y tahmini yapılır. Çekirdek fonksiyonu için seçilecek olan metot ise,

1. Model olarak kullanacağımız yöntem de doğrusal olmayan regresyon denklemi oluşturmak,

2. Fonların fiyat tahmini için veri setimizin birden fazla bağımsız ve bir bağımlı değişken yapısından oluşması,

Yukarıdaki sebeplerden dolayı, SVR modeli uygulanırken Radial Basis Function (RBF) çekirdek fonksiyonu kullanılacaktır.

Son olarak, SVR algoritma akış sürecinden kısaca bahsedilirse;

1. **Adım:** Model için eğitim veri seti seçilir.

2. **Adım:** Bağımsız değişken değerleri normalizasyon yöntemi ile verileri [-1, 1] aralığına getirilecek şekilde hazırlanır.

3. **Adım:** Giriş verileri üzerinden çekirdek fonksiyon hesaplama işlemleri yapılır. Modelimiz için seçtiğimiz RBF fonksiyonu ile hesaplamalar gerçekleştirilir.

4. **Adım:** (3.5)'teki denklem ile optimizasyon probleminin çözümü yapılır.

5. **Adım:** Model tahmin fonksiyonu için regresyon denklemi elde edilir.

6. **Adım:** Girdi verileri ile 5.Adım'da bulunan regresyon denklemi kullanılarak bağımlı değişken tahmin sonuçları elde edilir [64].

Yukarıdaki algoritmanın çalışma akışının özetinden anlaşılacağı üzere, 6 adımda SVR model için tahmin işlemi bu şekilde gerçekleştirilmektedir.

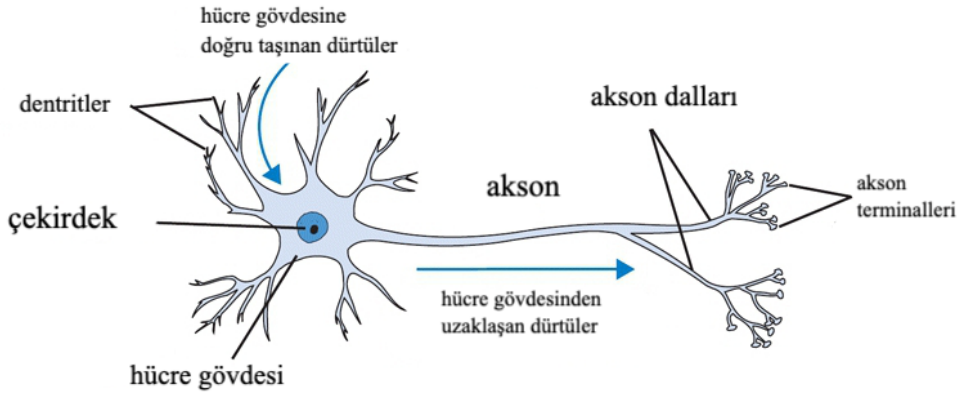
3.2.2. Yapay Sinir Ağları(YSA)

3.2.2.1. Teori

İlk yapay sinir ağ modeli, 1943 yılında Nöropsikiyatrist Warren McCulloch ve bilim adamı Walter Pitts tarafından "A Logical Calculus of The Ideas Immanent In Nervous Activity" adıyla yayınladıkları makale çalışması ile önerilmiştir [65]. Ancak kendi dönemlerinin kısıtlı imkanları nedeniyle, yapay sinir ağ modeli alanında çok fazla ilerleme kaydedilememiştir. Bundan sonraki yıllarda ise çalışmalar artarak devam etmiştir. Fakat sinir ağ modeli için en önemli süreç, 1969'da Marvin Minsky ve Seymore Papert tarafından bu konuda bir kitabın yayınlanmış olmasıdır. Bu kitap, YSA çalışmaları ile ilgili kaygı duyulan etik sorunları ortadan kaldırmış fakat yeni gelişen teknolojilere doğru giden yolu derinleştirerek YSA çalışmalarına olan ilginin azalmasına neden olmuştur. Böylelikle, YSA çalışmalarında 9 yıllık (1960-1969) altın çağı olarak nitelendirilen ilerleme süreci sonlandırılmış, YSA için karanlık çağ olarak belirtilen bir sürecin başlangıcı olmuştur. Daha sonraki süreçte YSA çalışmaları ile ilgili yapılan faaliyetlerde ilk gözle görülür önemli gelişmeler ise 1990'lı yıllara dayanmaktadır [66]. YSA çalışmalarının kısa tarihsel arka planından sonra YSA'nın teorik bağlamından ve çalışma mekanizmasından bahsetmemiz gerekmektedir.

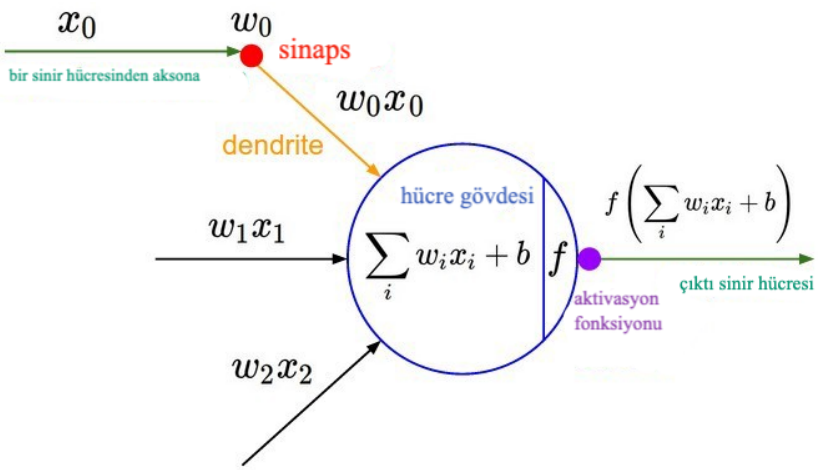
İnsan beyninin çalışma mekanizması üzerine uzun yıllar boyunca bilim adamları tarafından birden fazla araştırma ve çalışma yapılmıştır. Yapılan çalışmaların temel argümanını oluşturan soru ise beynin çalışma mekanizmasının nasıl mevcut gelişen teknoloji altyapısı ile taklit edilebilirliği sağlayabileceği tartışması olmuştur. Beyin sinir hücrelerinin bu kadar mükemmel organize olması ve aynı şekilde çok büyük bir karmaşıklığı kendi içerisinde barındırması, yapay olarak taklit edilebilir mi sorusu bilim insanları için büyük bir merak konusu da olmuştur. Çalışmaların temel felsefesini oluşturan taklit edilebilirlik sorusu, ilk olarak bilim insanlarının sinir hücresinin yapısı üzerinden durmasına ve ilk matematiksel model çalışmalarını bu şekilde yapmasına zemin hazırlamıştır. Bu bağlam bir sinir hücresinin yapısını gösteren şekil 3.7 ve yapay bir sinir hücresinin matematiksel

model gösterimi olan şekil 3.8'deki yapıları karşılaştırarak benzerlik ilişkisi üzerinden irdelenebilir.



Şekil 3.7: Sinir hücresi biyolojik gösterimi[67]

Şekil 3.7'de sinir hücresinin temel yapıları olan sinapslar, akson, soma (hücre çekirdeği) ve dentrite'lerden oluşmaktadır. Sinir hücresinin çalışma yapısı; dentrit'lerden alınan sinyalleri somaya (çekirdeğe) iletir. Soma, iletilen sinyalleri çekirdekte toplar, bu toplama işlemi dentrit'lerden alınan sinyallerin belli bir eşik değeri kapsamında toplanmasıyla gerçekleştirilir. Toplanılan sinyalleri akson'lara iletirler, akson ise aldıklarını diğer hücrelere aktarır. Bu aktarım işleminde önce sinaps'lar bir ön işleme tabi tutularak, gönderilen sinyalleri belirli bir eşik değerine getirerek diğer hücrelere aktarım işlemi yapılır. Bu şekilde bir sinir hücresinde bilgi aktarım süreci tamamlanmış olur [68].



Şekil 3.8: Yapay sinir ağının matematiksel gösterimi [67]

Beyin sinir hücresinin şekil 3.7 deki işleyişinden sonra benzer çalışma akışı olan şekil 3.8'de, yapay sinir hücresinin dışardan gelen *x* girdilerini belli *w* ağırlık değerlerine göre bir toplama fonksiyonu ile işlem yapılarak, bir sonraki adım olan aktivasyon fonksiyonundan geçirip bir çıktı üretilmesini sağlar. Üretilen çıktı ağın diğer bağlantıları üzerinden hücrelere gönderilerek bilgi yani tahmin işlemi gerçekleştirilmiş olur[68].

Aşağıdaki tablo 3.1'deki bir sinir hücresi ile yapay bir sinir hücresinin ortak tanımlarından yola çıkarak iki hücre modelinin ağ yapısını oluşturan terminolojileri özetlenebilir.

Tablo 3.1: Sinir hücresi ile yapay sinir ağı ortak terminolojisi

Sinir Hücresi Sistemi	**Yapay Sinir Ağı**
Nöron	İşlem Elemanı
Dentrit	Toplama Fonskiyonu
Hücre Çekirdeği	Aktivasyon Fonksiyonu
Akson	Çıkış Elemanı
Sinaps	Ağırlıklar

Genel olarak YSA tanımlarsak, beynin çalışma mekanizmasının bir işlevini yerine getirmek için yapılan işlemin modellenmiş formu olarak tanımlanabilir. YSA, sinir hücrelerinin kendi aralarında farklı şekillerde bağlanmasından oluşur ve genel olarak katmanlar biçiminde düzenlenmektedir. Teknolojik alt yapısı olarak ise, elektronik devrelerle ya da bilgisayar ortamında yazılımsal olarak gerçekleştirilir. Ayrıca, beynin bilgiyi işleme metodu olarak YSA, modelin öğrenme sürecinde veriyi işleme, depolama ve genelleştirme özleliklerine sahip paralel dağılmış bir işlemcidir. Son olarak, Turing makineleriyle temeli atılan yapay zeka çalışmalarının en çok araştırma yapılmış konularının başında "Yapay Sinir Ağları" gelmektedir. Temel olarak YSA'yı özetlersek, tamamen insan beynini modelleyerek geliştirilmesi yapılmış bir teknolojidir[69].

YSA'lar bir çok alanda uygulama sahasına sahip olmuştur. Başlıca kullanım alanları aşağıdaki gibi kısaca maddeler halinde gruplandırılabilir:

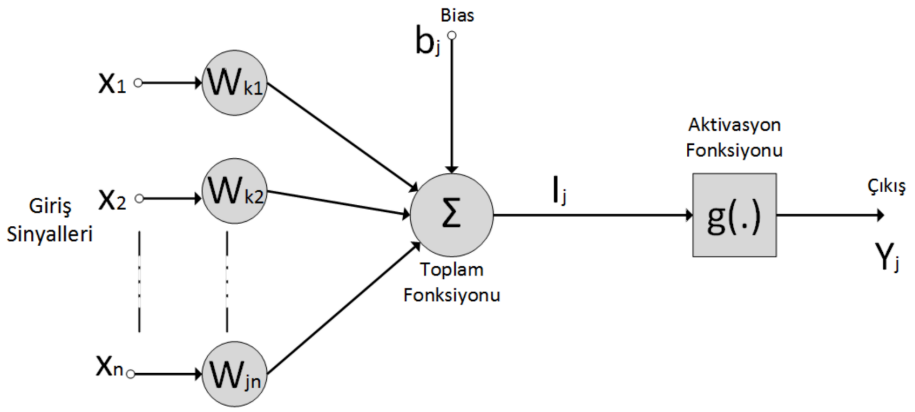
- Bir araya getirme / toplama
- Gruplama / İlişkilendirme
- Sınıflandırma
- Örüntü tanıma
- Regresyon ve genelleme
- Optimizasyon
- Tahmin uygulamaları,

Benzeri alanlar da olmak üzere, pek çok çalışmada kullanılmaktadır. Yapılan çalışmalar incelendiğinde, YSA'ların büyük boyutlu, kompleks, net olmayan, eksik, hatalı olma ihtimali yüksek verilerin olduğu ve probleme dair herhangi bir matematiksel model ya da algoritmanın bulunmadığı durumlarda da oldukça yaygın bir şekilde kullanıldıkları görülmektedir[70].

3.2.2.2. Matematiksel Model ve Algoritma

Bu kısımda ilk olarak YSA'nın, bir yapay sinir hücresinin matematiksel modeli üzerinde durulacak, daha sonra yapay sinir hücrelerinden oluşan ağ yapısı

incelenecektir. Son olarak da, bu tez çalışmasında ana sorumuz olan fonlar için tahmin işleminin nasıl yapılacağı konusu üzerine, birden fazla bağımsız değişkenden bir bağımlı değişken tahmini yapılacak olan model çalışması için ileri beslemeli ağ algoritmalarından çok katmanlı algılayıcı (MLP) ağ yapısından ve öğrenme adımlarından bahsedilecektir. MLP'nin tercih edilme sebebi, YSA regresyon problemlerindeki tahmin çalışmalarında daha çok tercih edilmesi ve araştırmanın model yapısına daha uygun bir algoritma olmasından kaynaklanmaktadır[69,71].



Şekil 3.9: Yapay sinir ağı[66]

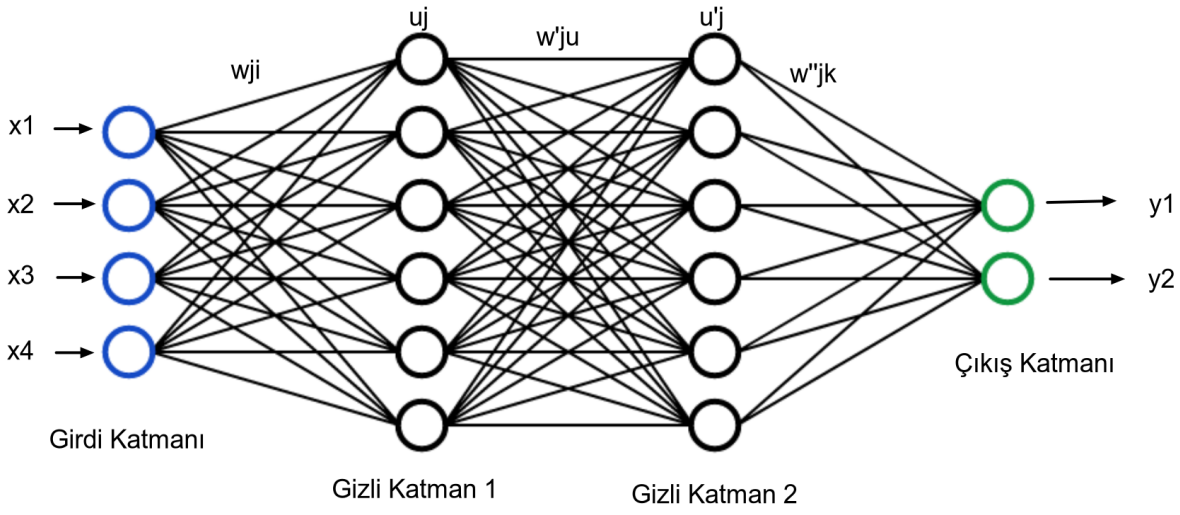
Şekil 3.9'da ki gibi bir yapay sinir hücresindeki tahmin işlemi, giriş sinyalleri $(x_1, x_2 \ldots x_n)$ olan bağımsız değişken değerlerinin, her bir girdi değerine karşılık bir w_k ağırlık ile çarpılır. Bu çarpma işleminde her bir değerin çıktıya olan etkisinin kontrölünü sağlanmış olur. Bir sonraki adımda bulunan bu değerler, toplam fonksiyonu ile işlem yapılarak aktivasyon fonksiyonuna iletilir. Aktivasyon fonksiyonu bir dönüştürme işlemine tabi tutularak bir çıktı değeri (y_j) üretmiş olur. Bu şekilde tahmin işlemini gerçekleştirilmiş olmaktadır. Tahmin çalışmalarında çokça kullanılan aktivasyon fonksiyonları ise;

Sigmoid: Sınıflandırma problemlerinde sıklıkla kullanılır. Giriş verilerini 0 ile 1 aralığındaki değerlere dönüştürür.

Tanjant Hiperbolik: Giriş verilerini -1 ile 1 aralığındaki değerlere dönüştürmektedir. En iyileme problemlerinin daha anlaşılır bir forma gelmesine katkı sağlar.

Doğrultulmuş Dorusal Ünite (ReLU): Girdi verilerini [0 , ∞] aralığındaki değerlere dönüştürür. Derin öğrenme problemlerinde diğer fonksiyonlara göre çok daha hızlı eğitilmektedir.

Sızdırılmış Doğrultulmuş Doğrusal Ünite (Leaky ReLU): ReLU fonksiyonun dezanvatajı olan negatif değerleri sıfıra eşitlemesi problemine çözüm olarak, negatif değerler yerine aktivasyon fonksiyonuna bir eğim eklenmesi önerilmiştir[72].



Şekil 3.10: Çok katmanlı yapay sinir ağı

Şekil 3.9 daki yapay sinir hücreleri bir araya gelerek şekil 3.10 daki gibi yapay sinir ağını oluştururlar. Ağ yapısındaki katman bilgileri için tanımları kısaca şu şekildedir;

Girdi Katmanı: YSA modeline giriş değerleri(x_1, x_2, x_3, x_4) olarak verilen bilgiler bu katmanda bulunur. Her bir değerin bir ağırlık (w_{ji}) bilgisi mevcuttur. Gizli katmanlar arasındaki bağlantıyı sağlar. Ayrıca, ağırlık bilgisi çıkış değeri için bir kontrol etkisi de sağlamış olur.

Gizli Katmanı: Girdi değerlerini ağrılık bilgilerini (w'_{ju}) kullanarak çıktıya dönüştürme işleninin yapıldığı katmandır. Problemin zorluk derecesine göre gizli katman sayısı değişmektedir. Şekil 3.10 da ki gizli katman sayı 2 olup, nöron sayısı ise her bir katmanda 6 adet bulunmaktadır. Nöronlar öğrenilen bilgileri kendi yapısında tutan birimler olarak görülmektedir.

Çıktı Katmanı: Verilen girdi değerlerinin, gizli katmandaki işlemden sonra elde edilen çıktı değerileri (y_1, y_2) olarak üretildiği katmandır. Bu katmanda beklenen değer ile gerçek değer arasındaki fark bulunur ve bulunan değer seçilecek olan hata hesaplama fonksiyonu ile işlem yapılarak ağın performansı değerlendirilir. Bulunan sonucun performansına göre ağrılıkların (w''_{jk}) güncellemesi yapılır. Ağırlık güncellemesi, YSA öğrenme sürecini belirtmektedir. Ağırlık güncelleme için farklı optimizasyon metotları bulunmaktadır. Çalışmamızda kullanacağımız optimizasyon fonksiyonu ise ileri beslemeli ağ algoritmaları olacaktır[71-72].

YSA modelin matematiksel olarak ifade edilmesi ise;

$$h_k(x) = g(\beta_{0k} + \sum_{j=1}^{p} x_j \beta_{jk}) \quad \textbf{(3.9)}$$

$$g(u) = \frac{1}{1 + e^{-u}} \quad \textbf{(3.10)}$$

Şekil 3.9’ daki bir sinir ağ hücresinin, 3.9’da verilen matematiksel denklem formülüne göre çıktı üretme işleminin yapılması, 3.10 denklemine (örnek olarak sigmoid fonksiyonu verilmiş) benzer bir dönüştürme işlemi yapılarak bir sinir ağ hücresinin çıktı ile ilişkilendirilerek bilginin üretilmesini sağlar.

$$f(x) = \gamma_0 + \sum_{k=1}^{H} \gamma_k h_k \quad \textbf{(3.11)}$$

Hücrenin bilgi üretiminden sonraki adımda bilginin diğer katmanla ilişkilendirilmesi 3.11 de verilen doğrusal bir denklem ile çıktıya bağlama işlemini gerçekleştirir. Bir hücrenin bilgi üretimi ve çıktıyla ilişkilendirilme denklemlerinden hareketle, nihai olarak optimize edilmiş YSA problem çözme denklemi 3.12'deki gibi formülüze edilebilir.

$$\sum_{i=1}^{n}(y_i - f_i(x))^2 + \lambda \sum_{k=1}^{H}\sum_{j=0}^{P}\beta_{jk}^2 + \lambda \sum_{k=0}^{H}\gamma_k^2 \qquad \textbf{(3.12)}$$

3.12'deki matematiksel denkleme dair son olarak belirtilmesi gereken nokta, denklemdeki cezalandırma katsayısı olan λ parametreleri genelde 0 ile 1 arasında değerler almaktadır.

Matematiksel model anlatımdan sonra YSA modelinin öğrenme algoritma adımları ise şu şekildedir;

1. Adım: Test veri seti seçiminin, modelin eğitim setinden öğrenme performansı en iyi olacak şekilde gerçekleştirilmesi, ağın eğitim setinden öğrenme süreci tamamlandıktan sonra test veri seti ile ağın öğrenme performası ölçülür. Bu şekilde yeni veriler karşısındaki model başarısı ağın iyi eğitilip eğitilmediğini gösterir.

2. Adım: Modelin topolojik formunun belirlenmesi, ağın öğrenilmesi gereken çıktı değeri için topolojik yapısını netleştirir. Buna göre girdi verileri, gizli katman sayısı, çıktı katman sayısı gibi ağın öğrenme yapısı bu adımda kararlaştırılır.

3. Adım: Ağın parametre yapılarının belirlenmesi, ceza katsayısı, aktivasyon fonksiyonu gibi metot ve değerler belirlenir.

4. Adım: Girdi verileri için ağırlıkların başlangıç değerlerinin belirlenmesi, verilerin bir diğer katman ile ilişkilendirilmesi sürecinde her bir girdi değerinin ağırlıkları başlangıç değerleri için rastgele atanır. Daha sonra öğrenme sürecine göre uygun ağırlık değerlerini ağın kendisi optimize eder.

5. Adım: Öğrenme sürecinde ağırlıkların işlenmesi, ağın öğrenme aşamasında ağırlık değerleri belirli bir kurala göre ağa giriş değerleri ile işleme girer.

6. Adım: Model öğrenmesi için ileri hesaplama işlemlerin olması, verilen girdi ile beklenen çıktı değerleri için hesaplamalar yapılır.

7. Adım: Tahmin edilen çıktı değerleri ile gerçek değerler arasındaki farkın, hata değerlerinin hesaplanması ile bu adımda gerçekleşir.

8. Adım: Ağırlık değerlerinin güncellemesi, bu adımda hata değerlerinin hesaplanmasına göre bulunan hata değerinin optimize edilmesi için ağırlık değerleri güncellenir.

9. Adım: Model öğrenme sürecinin tamamlama aşaması, tahmin edilen ile gerçek değer arasındaki farkın hata değeri kabul edilir bir aşamaya gelinceye kadar ileri beslemeli sinir ağı metodu ile öğrenme sürecine devam eder[71,73].

YSA modelinin eğitilme/öğrenme süreci yukarıdaki algoritmanın öğrenme akışından anlaşılacağı üzere, 9 adımda tahmin işlemini gerçekleştirmiş olacaktır.

4. MODEL ÇALIŞMASI

Tez çalışmasının bu bölümünde, fonların fiyat tahmini için model geliştirme uygulamaları yapılacaktır. Model geliştirme için kullanacağımız makine öğrenme metotlarına dair literatür araştırması bolüm 2'de yapılmıştır. Yapılan literatür araştırmasıyla bu çalışma da kullanılan metotların nasıl tercih edildiği belirlenmiştir. Büyük bir uygulama alanının mevcut olması ve benzer çalışmalarla makine öğrenme metotlarında ciddi başarılar elde edilmiş olması, bölüm 3'de açıklanan algoritmaların kullanılmasını sağlayarak tezin alt yapısını oluşturmuştur. Bu bağlamda bu bölümde uygulama alanı olarak beş ana başlık şeklinde bir akış izlenmiştir. İlk olarak,

uygulama çalışmaların temel yapısını oluşturan ve önemi büyük olan veri seti hakkında detaylı bir bilgilendirme verilmiştir. Sonrasında, model geliştirme bölümüne geçmeden yeni bir başlık açılarak, geliştirilecek olan modellerin kabul doğruluklarının ölçümünün yapılmasını sağlayan metrikler ile ilgi açıklamalar yapılmış ve daha sonra model geliştirme için kullanılacak olan algoritmaların işleyişi ve kodlama süreçleri için detaylı bilgi verilmiştir. Son ana başlık da ise, kullanılan dört algoritma ile uygulaması yapılan model geliştirme için bulunan sonuçların ortak bir tablo da karşılaştırması yapılarak bölüm sonlandırılmıştır. Ayrıca, tüm uygulama geliştirme ve veri işlem süreçlerin de kullanılan platform, programlama dili ve versiyon bilgileri için tablo 4.1'e ve uygulama çalışmalarında kullanılacak olan bilgisayarın teknik özeliklerine dair ise tablo 4.2' ye ayrıntılı olarak bakılabilir.

Tablo 4.1: Uygulama geliştirme platform bilgileri

Uygulama Adı	Açıklama	Version
Anaconda Navigator	Veri bilimi masaüstü portalı	1.9.12
Python3	Nesne yönelimli, yorumlamalı ve etkileşimi yüksek seviyeli bir programlama dilidir.	3.7.7
Spyder	Python programlama geliştirme için açık kaynaklı bir geliştirme ortamıdır.	4.1.2
JupyterLab	Bir çok programlama dilinde etkileşimli bilgi işlemleri için geliştirilmiş açık kaynaklı bir geliştirme ortamıdır.	1.2.6

Tablo 4.2: Bilgisayar teknik özelikleri

İşlemci	CPU @ 2.7 GHz, Dual-Core Intel Core i5
Bellek	8 GB
Grafikler	Intel Iris Graphics 6100 1536 MB
Depolama	SSD 128 GB

4.1. VERİ HAZIRLAMA VE ÖN İŞLEME SÜREÇLERİ

Bu bölüm iki alt başlıktan oluşmaktadır. İlk olarak model geliştirme için kullanılacak olan veri seti hakkında detaylı bir bilgi verilmiştir. İkinci olarak, veri ön işleme süreçleri için veri setinin oluşturulması, temizlenmesi ve verinin görselleştirme adımlarının tümü bu başlık altında incelenmiştir.

4.1.1. Veri Seti Hakkında

Bu çalışmada kullanılan veriler, Takas İstanbul (Takasbank)'un platform ve veri kaynağı sağlayıcılığı yaptığı Türkiye Elektronik Fon Dağıtım Platformu (TEFAS) web sitesi üzerinden genele açık olan bilgilerden elde edilmiştir. Her bir fon için 02.01.2019 – 31.12.2019 tarihleri arasında tipi, türü, toplam değeri, tedavüldeki pay sayısı, pay alan kişi sayısı, fiyatı ve menkul (26 çeşit) oranları ile her bir tarih için faiz bilgisi, altın fiyatı ve dolar fiyatı baz alınarak veri seti hazırlanmıştır. Veri kümesi, toplam 187,438 veri ve 37 kolondan oluşmaktadır. Veri setini oluşturan her bir kolonun tipi ve içeriği hakkında detaylı bilgi ise aşağıda verilmiştir.

- **TARIH**: Fonun işlem gördüğü tarih bilgisini içerir. Veri tipi, date olarak tutulur.
- **FONTIP**: Fonun tip bilgisini içerir ve 3 çeşittir; Borsa Yatırım Fonu, Emeklilik Fonu, Yatırım Fonu. Veri tipi, object olarak tutulmaktadır. Veri setinde object olarak tutulan verilerin, veri ön işlem sürecinde kategorik veri olacak şekilde dönüşümü yapılmaktadır.
- **FONTUR**: Fonların tür bilgisini içermektedir ve 31 çeşit olarak mevcuttur. Tür sayısı çok olduğundan dolayı Tablo 4.3'te bilgileri verilmektedir. Veri tipi, object olarak tutulmaktadır.
- **FON**: Veri setinde fonların uzun isimlendirmeleri mevcut değildir. Bunun yerine, Takasbank tarafından her bir fon'a ait bir kod bilgisi verilmiştir. Toplam fon sayısı 892 ve veri tipi olarakta object bilgisini içermektedir.

- **FONTOPLAMDEGER**: Fonun büyüklük bilgisini içerir ve hesaplama olarak; portföy büyüklüğü + alacaklar - borçlar şeklinde bulunur. Veri tipi, float64 olarak tutulmaktadır.
- **TEDAVPAYSAYISI**: Fonun satılmış olan pay adeti bilgisini içerir. Veri tipi, float64 olarak tutulmaktadır.
- **KISISAYISI**: Fonun yatırımcı sayısı bilgisini barındırır. Veri tipi, int64 olarak bulunmaktadır.
- **FONFIYAT**: Fonun o tarihteki fiyat bilgisini içerir. Veri tipi, float64 olarak tutulmaktadır.
- **FAIZ**: Tarih bazında Merkez Bankasının verdiği faiz bilgisini barındırır. Veri tipi, float64 olarak tutulmaktadır.
- **DOLARFIYAT**: Merkez Bankasının gün sonunda o tarih için en son verdiği dolar fiyatını bulundurmaktadır. Veri tipi, float64 olarak bulunmaktadır.
- **ALTINFIYAT**: Merkez Bankasının gün sonunda o tarih için son verdiği altın fiyat bilgisini bulundurmaktadır. Veri tipi ise, float64'tur.
- **MENKULORAN**: Fonun portföyündeki kıymetin portföy değerine oranının, yüzdelik üzerinden gösterilmesi bilgisini içerir. Veri setindeki menkul oranı, 26 farklı menkul değerini kolon bilgisi olarak bulundurmaktadır. Verideki her bir menkul değerin kodu ve tanımı tablo 4.4'te gösterilmektedir. Veri tipleri ise, tüm menkul değerleri float64 olarak veri kümesinde tutulmaktadır.

Tablo 4.3: Veri setindeki fon türleri

	FONTUR
1	Altın Fonu
2	Gümüş Fonu
3	Hisse Senedi Fonu
4	Başlangıç Fonu
5	Başlangıç Katılım Fonu
6	Borçlanma Araçları Fonu
7	Devlet Katkısı Fonu
8	Değişken Fon
9	Endeks Fon
10	Fon Sepeti Fonu
11	Kamu Borçlanma Araçları Fonu
12	Kamu Kira Sertifikası Fonu
13	Kamu Yabancı Para (Döviz) Cinsinden Borçlanma Araç
14	Karma Fon
15	Katılım Katkı Fonu
16	Katılım Standart Fon
17	Kıymetli Madenler
18	OKS Katılım Standart Fon
19	OKS Standart Fon
20	Para Piyasası Fonu
21	Standart Fon
22	Uluslararası Borçlanma Araçları Fonu
23	Özel Sektör Borçlanma Araçları Fonu
24	Altın ve Diğer Kıymetli Madenler Fonu
25	Hisse Senedi Yoğun
26	Katılım Fonu
27	Kira Sertifikası Fonu
28	Kisa Vadeli Kira Sertifikalari Katilim Fonu
29	Koruma Amaçlı Fon
30	Kısa Vadeli Borçlanma Araçları Fonu
31	Serbest Fon

Tablo 4.4: Takasbank'tan alınan menkul kod tanımları

KOD	ACIKLAMAING	ACIKLAMATR
BB	Bank Bills	Banka Bonosu
DT	Government Bond	Devlet Tahvili
DB	FX Payable Bills	Döviz Ödemeli Bono
DÖT	Foreign Currency Bills	Dövize Ödemeli Tahvil
EUT	Eurobonds	Eurobonds
FB	Commercial Paper	Finansman Bonosu
FKB	Fund Participation Certificate	Fon Katılma Belgesi
GAS	Real Estate Certificate	Gayrı Menkul Sertifikası
HB	Treasury Bill	Hazine Bonosu
HS	Stock	Hisse Senedi
KBA	Government Bonds and Bills (FX)	Kamu Dış Borçlanma Araçları
KKS	Government Lease Certificates	Kamu Kira Sertifikaları
KH	Participation Account	Katılım Hesabı
KM	Precious Metals	Kıymetli Madenler
OSKS	Private Sector Lease Certificates	Özel Sektör Kira Sertifikaları
OST	Private Sector Bond	Özel Sektör Tahvili
TR	Reverse-Repo	Ters-Repo
TPP	TMM	TPP
T	Derivatives	Türev Araçları
VM	Term Deposit	Vadeli Mevduat
VDM	Asset-Backed Securities	Varlığa Dayalı Menkul Kıymetler
YBA	Foreign Debt Instruments	Yabancı Borçlanma Aracı
YHS	Foreign Equity	Yabancı Hisse Senedi
YMK	Foreign Securities	Yabancı Menkul Kıymet
D	Other	Diğer
R	Repo	Repo

Bu bölümde buraya kadar anlatılan, verilerin toplanmasının ve her bir kolon için içerik ve tip bilgilerinin nasıl olduğudur. Ayrıca, verilerin toplanma sürecinde herhangi bir ön işlem süreci uygulanmadığı için mevcut şu an ki format verinin ham halini barındırmaktadır. Bundan dolayı, eksik, yanlış ve gereksiz verilerde bulunmaktadır. Veri işlem bölümünde bu verilerin ayrıntılı incelemesi yapılacaktır.

4.1.2. Veri Ön İşleme Süreçleri

Veri işleme bölümünde, hazırlanmış verinin ham formatı üzerinden ön işleme süreçleri yapılacaktır. Ön işlem süreçleri veri setinin, uygulama bölümünde geliştirilecek olan model için çok önemli bir yere sahiptir. Çünkü ham verinin toplanma sürecinde gereksiz, hatalı, tekrarlı, tutarsız ve boş verilerin olması mümkündür ve verinin bu hali ise geliştireceğimiz model için doğru bir sonuç elde edilmesini engeller. Bu sebepten ötürü, veri seti tutarlı, eksiksiz ve hatasız olmasını sağlayacak daha ideal bir yapıya dönüştürülür. Doğru bir veri seti ise, model geliştirme çalışmalarımız için daha performanslı ve doğru sonuçlar elde edilmesini sağlar. Bu kapsamda, verilerin uygulama geliştirme ve analizler için daha uygun bir yapıya getirilmesini sağlayan sürece veri ön işleme adımı denir[74]. Veri ön işleme adımları aşağıdaki şekilde sıralanabilir.

- Veri temizleme
- Veri bütünleştirme
- Veri indirgeme
- Veri dönüştürme
- Veri madenciliği tekniklerinin uygulanması
- Bulunan sonuçların değerlendirilmesi, uygulanması[74].

Bu adımlar çerçevesinde, veri setimizin adım adım ön işleme süreçlerini yapıp ideal bir veri kümesine dönüşümü sağlanacaktır. Temizlenmiş veri kümesi üzerinden, her bir veri kolonunun ortalaması, maximum, minimum değerleri ve korelasyon matrislerine bakılarak gerekli-gereksiz veri olup olmama konusu

incelenecektir. Son olarak ise, veri analizinin daha iyi anlaşılması için görselleştirme çalışmaları yapılacaktır.

4.1.2.1. Veri Temizleme ve Dönüştürme

Veri setinin düzenlenmemiş formatı şekil 4.1'de verilerek, veri üzerinden temizleme ve dönüştürme işlemleri sırasıyla yapılmaya başlanacaktır.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 187438 entries, 0 to 187437
Data columns (total 38 columns):
 #   Column          Non-Null Count   Dtype
---  ------          --------------   -----
 0   TARIH           187438 non-null  datetime64[ns]
 1   FONTIP          187438 non-null  object
 2   FONTUR          187438 non-null  object
 3   FON             187438 non-null  object
 4   Unnamed: 4      0 non-null       float64
 5   FONTOPLAMDEGER  187438 non-null  float64
 6   TEDAVPAYSAYISI  187438 non-null  float64
 7   KISISAYISI      187438 non-null  int64
 8   FONFIYAT        187438 non-null  float64
 9   FAIZ            187438 non-null  float64
 10  DOLARFIYAT      187438 non-null  float64
 11  ALTINFIYAT      187438 non-null  float64
 12  BB              187438 non-null  float64
 13  DT              185256 non-null  float64
 14  DB              173992 non-null  float64
 15  DÖT             148075 non-null  float64
 16  EUT             113791 non-null  float64
 17  FB              84322 non-null   float64
 18  FKB             55378 non-null   float64
 19  GAS             34803 non-null   float64
 20  HB              21222 non-null   float64
 21  HS              11863 non-null   float64
 22  KBA             6322 non-null    float64
 23  KKS             2693 non-null    float64
 24  KH              1216 non-null    float64
 25  KM              548 non-null     float64
 26  OSKS            211 non-null     float64
 27  OST             11 non-null      float64
 28  TR              0 non-null       float64
 29  TPP             0 non-null       float64
 30  T               0 non-null       float64
 31  VM              0 non-null       float64
 32  VDM             0 non-null       float64
 33  YBA             0 non-null       float64
 34  YHS             0 non-null       float64
 35  YMK             0 non-null       float64
 36  D               0 non-null       float64
 37  R               0 non-null       float64
dtypes: datetime64[ns](1), float64(33), int64(1), object(3)
memory usage: 54.3+ MB
```

Şekil 4.1 : Veri setinin ön işlem yapılmamış formatı

İlk olarak şekil 4.1'de, veri setinin toplam 187,438 veriden ve 38 kolondan oluştuğu bilgisi verilmiştir. Daha sonra dört kolondan oluşan yukarıdaki şekil 4.1'de ki kolon bilgileri sırasıyla, verinin veri kümesindeki kolon sırası, adı, mevcut kolonda bulunan veri sayısı ve tip bilgilerini içermektedir. Son olarak, bulunan verilerin tip sayısı ve ne kadar hafıza kullandığı bilgisi verilmiştir. Şekil 4.1'de anlaşılacağı üzere, dördüncü sıradaki kolonun tanımsız olduğu ve boş veri içerdiği görülmektedir. Ayrıca, 28'inci sıradaki TR'den başlayarak 37'inci kolandaki alana kadar herhangi bir veri içermediği ve gereksiz olduğu anlaşılmaktadır. Bu bilgiler ışığında, veri setimiz hatalı ve gereksiz kolonlardan temizlenecek ve object olan verilerin tiplerin kategorik olarak dönüşüm işlemleri bu adımda yapılarak, şekil 4.2'deki son halini almış olacaktır.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 187438 entries, 0 to 187437
Data columns (total 27 columns):
 #   Column          Non-Null Count   Dtype
---  ------          --------------   -----
 0   TARIH           187438 non-null  datetime64[ns]
 1   FONTIP          187438 non-null  category
 2   FONTUR          187438 non-null  category
 3   FON             187438 non-null  category
 4   FONTOPLAMDEGER  187438 non-null  float64
 5   TEDAVPAYSAYISI  187438 non-null  float64
 6   KISISAYISI      187438 non-null  int64
 7   FONFIYAT        187438 non-null  float64
 8   FAIZ            187438 non-null  float64
 9   DOLARFIYAT      187438 non-null  float64
 10  ALTINFIYAT      187438 non-null  float64
 11  BB              187438 non-null  float64
 12  DT              185256 non-null  float64
 13  DB              173992 non-null  float64
 14  DÖT             148075 non-null  float64
 15  EUT             113791 non-null  float64
 16  FB              84322 non-null   float64
 17  FKB             55378 non-null   float64
 18  GAS             34803 non-null   float64
 19  HB              21222 non-null   float64
 20  HS              11863 non-null   float64
 21  KBA             6322 non-null    float64
 22  KKS             2693 non-null    float64
 23  KH              1216 non-null    float64
 24  KM              548 non-null     float64
 25  OSKS            211 non-null     float64
 26  OST             11 non-null      float64
dtypes: category(3), datetime64[ns](1), float64(22), int64(1)
memory usage: 35.1 MB
```

Şekil 4.2: Veri setinin ön işlem yapılmış formatı

Veri setimiz gereksiz kolon değişkenlerinden temizlenerek, mevcut değişken kolon sayısı 27 olacaktır. Bir sonraki adım da şekil 4.3'de ki veri setinde bulunan null değerleri üzerinde durulacaktır.

	TARIH	FONTIP	FONTUR	FON	FONTOPLAMDEGER	TEDAVPAYSAYISI	KISISAYISI	FONFIYAT	FAIZ	DOLARFIYAT	...	FKB	GAS	HB	HS	KBA	KKS	KH	KM	OSKS	OST
0	2019-01-02	BORSA YATIRIM FONU	Altın Fonu	FGA	8.722176e+07	4.350000e+06	0	20.050979	0.2302	5.2905	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	2019-01-02	BORSA YATIRIM FONU	Gümüş Fonu	FGS	1.238294e+07	7.000000e+05	0	17.689916	0.2302	5.2905	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2	2019-01-02	BORSA YATIRIM FONU	Hisse Senedi Fonu	DJA	1.491130e+07	4.800000e+05	0	31.065216	0.2302	5.2905	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
3	2019-01-02	EMEKLİLİK FONU	Altın Fonu	AEA	1.465104e+09	6.335717e+10	325392	0.023125	0.2302	5.2905	...	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	2019-01-02	EMEKLİLİK FONU	Altın Fonu	AEL	1.784293e+09	7.415738e+10	325460	0.024061	0.2302	5.2905	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

5 rows × 27 columns

Şekil 4.3: Veri setinin ilk 5 satırlık formatı

İlk 5 veri bilgisi verilmiş olan veri seti incelendiğinde, verilerde NaN değerlerin olduğu görülmektedir. Bu bağlamda veri setinin genelinde NaN ya da null değerin var olup olmadığı aşağıdaki şekil 4.4 sorgulanmıştır.

```
TARIH                0
FONTIP               0
FONTUR               0
FON                  0
FONTOPLAMDEGER       0
TEDAVPAYSAYISI       0
KISISAYISI           0
FONFIYAT             0
FAIZ                 0
DOLARFIYAT           0
ALTINFIYAT           0
BB                   0
DT                2182
DB               13446
DÖT              39363
EUT              73647
FB              103116
FKB             132060
GAS             152635
HB              166216
HS              175575
KBA             181116
KKS             184745
KH              186222
KM              186890
OSKS            187227
OST             187427
dtype: int64
```

Şekil 4.4: Veri setinde toplam null değer sayısı

Yukarıdaki şekil 4.4'de verilen bilgi, her değişken karşısında null olan toplam değer sayısıdır. Veri ön işlem sürecinde dikkat edilmesi gereken bir diğer önemli konu ise, veri içinde bütünlüğü bozan null değerlerinin olup olmaması durumudur. Veri setimizde olan null değerleri, şekil 4.4'te görüldüğü üzere sadece menkul oranlar bilgisinde bulunmaktadır. Bunun nedeni veri tabanında menkul oranları tutulurken, bulunan menkul oranlarının yanısıra değeri olmayan menkuller için de değişkenlere null değeri atılmış olmasıdır. Menkul oranlar, hesaplanırken yüzdelik üzerinden 0 ile 100 arasında bir değer aldığından dolayı, null olan değerlerin 0 olacak şekilde dönüştürülmesinin veri yapısını bozmayacağı görülmüştür.

	count	mean	std	min	25%	50%	75%	max
FONTOPLAMDEGER	187438.0	2.394037e+08	5.509213e+08	0.000000	6.929778e+06	3.815812e+07	1.786791e+08	7.207539e+09
TEDAVPAYSAYISI	187438.0	6.301665e+09	1.704090e+10	0.000000	3.456783e+07	5.147614e+08	4.347861e+09	2.577214e+11
KISISAYISI	187438.0	4.325684e+04	1.049055e+05	0.000000	5.100000e+01	1.269000e+03	3.306150e+04	1.064438e+06
FONFIYAT	187438.0	3.982973e+00	2.395512e+01	0.000000	1.523725e-02	3.057850e-02	5.684475e-01	3.657033e+02
FAIZ	187438.0	2.034597e-01	4.610195e-02	0.105100	1.630000e-01	2.295000e-01	2.418000e-01	2.550000e-01
DOLARFIYAT	187438.0	5.683442e+00	2.214480e-01	5.203800	5.532600e+00	5.723800e+00	5.813300e+00	6.213800e+00
ALTINFIYAT	187438.0	2.527867e+05	2.080213e+04	216132.634958	2.323975e+05	2.559016e+05	2.715878e+05	2.868514e+05
BB	187438.0	4.071823e+01	3.596555e+01	-90.330000	7.260000e+00	2.794000e+01	7.994000e+01	2.195700e+02
DT	187438.0	2.295568e+01	2.804406e+01	-119.570000	2.960000e+00	9.650000e+00	3.410000e+01	1.074500e+02
DB	187438.0	1.496090e+01	2.223612e+01	-16.150000	7.100000e-01	5.340000e+00	1.806000e+01	1.561000e+02
DÖT	187438.0	8.487016e+00	1.478798e+01	-23.390000	0.000000e+00	1.760000e+00	9.640000e+00	1.000500e+02
EUT	187438.0	5.873882e+00	1.284448e+01	-83.930000	0.000000e+00	1.000000e-02	6.260000e+00	9.918000e+01
FB	187438.0	3.238396e+00	8.946145e+00	-16.570000	0.000000e+00	0.000000e+00	1.790000e+00	9.824000e+01
FKB	187438.0	1.616070e+00	5.469517e+00	-40.620000	0.000000e+00	0.000000e+00	0.000000e+00	9.756000e+01
GAS	187438.0	1.045527e+00	4.623233e+00	-17.890000	0.000000e+00	0.000000e+00	0.000000e+00	9.670000e+01
HB	187438.0	6.632977e-01	3.542419e+00	-15.650000	0.000000e+00	0.000000e+00	0.000000e+00	5.857000e+01
HS	187438.0	3.054617e-01	2.177117e+00	-16.250000	0.000000e+00	0.000000e+00	0.000000e+00	4.899000e+01
KBA	187438.0	1.008381e-01	9.393350e-01	-16.760000	0.000000e+00	0.000000e+00	0.000000e+00	2.425000e+01
KKS	187438.0	2.969931e-02	5.083447e-01	-12.150000	0.000000e+00	0.000000e+00	0.000000e+00	3.756000e+01
KH	187438.0	5.369669e-03	1.304408e-01	0.000000	0.000000e+00	0.000000e+00	0.000000e+00	6.530000e+00
KM	187438.0	2.637352e-03	1.100750e-01	0.000000	0.000000e+00	0.000000e+00	0.000000e+00	6.940000e+00
OSKS	187438.0	5.484480e-05	7.166541e-03	0.000000	0.000000e+00	0.000000e+00	0.000000e+00	1.000000e+00
OST	187438.0	0.000000e+00	0.000000e+00	0.000000	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00

Şekil 4.5: Sayısal değerlerin ortalama bilgileri

Veri kümesindeki sayısal değişkenlerin ortalama, standart sapma, minimum, maximum değerleri ve verinin dağılım oranlarının bilgisi şekil 4.5'te verilmiştir. Bu bilgilerden de açıkça görülmektedir ki, OST değişkeninin tüm verilerinin 0 olduğu ve geri kalan diğer menkul oran değişkenlerinin de minimum değerlerinin negatif olduğu bilgisi mevcuttur. Gereksiz veri kapsamında OST değişkeninin veri setinden silinme işlemi yapılacaktır. Negatif değerler için ise, menkul değişken oranlarının 0 ile 100 arasında değer alacağı ifade edilmiştir. Lakin, Takasbank'tan alınan bilgiye göre müşteri kendi portföyünü hazırlarken toplam menkul değerinin 100 olması için negatif değerle dengeyi kurmaya çalışmış ve değişken oranlarını bu dengeyi sağlayacağı formatta Takasbank'a bildirmiştir. Veri setinin orijinal yapısının ve bütünlüğünün bozulmaması için negatif değerlere dair bir işlem yapılmamıştır.

4.1.2.2. Veri Korelasyon Matrisi

Veri ön işleminin bu adımında değişkenler arasındaki bağlantı yönüne ve büyüklüğüne dair korelasyon matris ilişkileri incelenecektir. Korelasyon kat sayısı, -1 ile +1 arasında bir değer alır. Buradaki değerinin artı ya da eksi olması ilişkinin büyüklüğü hakkında bilgi vermez, artı değeri alan iki değişkenin, birlikte aynı yönde arttığını ya da azaldığını gösterir. Eksi değeri ise, tam tersi bir korelasyonun iki değişken arasında olduğunu ifade eder. Biri artarken, diğer değişkenin azaldığını veya tam tersinin olması durumudur [75]. İlk olarak tüm sayısal değişkenlerin korelasyon matrisi şekil 4.6'da verilerek, matris grafiği hakkında değerlendirmeler yapılmıştır.

	FONTOPLAMDEGER	TEDAVPAYSAYISI	KISISAYISI	FONFIYAT	FAIZ	DOLARFIYAT	ALTINFIYAT	BB	DT	DB	...	FB	FKB	GAS	HB	HS	KBA	KKS	KH	KM	OSKS
FONTOPLAMDEGER	1.000000	0.623976	0.572384	0.240230	-0.046102	0.028570	0.045445	-0.117865	-0.027456	0.005659	...	0.129416	0.106025	0.128855	0.111821	0.121980	0.096114	0.097218	0.121421	0.027718	0.033211
TEDAVPAYSAYISI	0.623976	1.000000	0.775371	-0.060899	-0.008602	0.005140	0.006192	-0.017060	-0.051967	-0.005084	...	0.064690	0.041446	0.053225	0.049651	0.113149	0.052736	0.048886	0.058034	0.011181	0.015839
KISISAYISI	0.572384	0.775371	1.000000	-0.032681	0.005122	-0.004760	-0.006781	-0.015445	-0.041986	-0.012995	...	0.030175	0.040284	0.068447	0.079492	0.128956	0.112067	0.123169	0.161027	0.038945	0.041491
FONFIYAT	0.240230	-0.060899	-0.032681	1.000000	-0.007930	0.004703	0.008218	-0.048665	-0.015605	-0.016653	...	0.083380	0.091493	0.065059	0.019361	-0.016518	-0.012317	-0.009550	-0.006741	-0.003937	-0.001252
FAIZ	-0.046102	-0.008602	0.005122	-0.007930	1.000000	-0.245448	-0.765568	-0.033752	0.036919	0.018948	...	-0.005592	0.011795	0.001221	0.007406	0.003255	0.006380	0.001681	0.002117	-0.015502	-0.003672
DOLARFIYAT	0.028570	0.005140	-0.004760	0.004703	-0.245448	1.000000	0.637137	0.013332	-0.022635	0.001280	...	0.006523	-0.022790	-0.002092	-0.002664	-0.009928	-0.010439	-0.003190	0.010967	0.006252	-0.002566
ALTINFIYAT	0.045445	0.006192	-0.006781	0.008218	-0.765568	0.637137	1.000000	0.029033	-0.046377	-0.010399	...	0.014475	-0.017570	-0.002947	-0.005810	-0.001857	-0.008426	0.002367	0.004487	0.014997	0.002182
BB	-0.117865	-0.017060	-0.015445	-0.048665	-0.033752	0.013332	0.029033	1.000000	-0.486768	-0.417391	...	-0.230900	-0.159207	-0.134248	-0.126064	-0.097665	-0.082703	-0.050781	-0.043330	-0.024688	-0.008640
DT	-0.027456	-0.051967	-0.041986	-0.015605	0.036919	-0.022635	-0.046377	-0.486768	1.000000	-0.181136	...	-0.137992	-0.119830	-0.080377	-0.057854	-0.053092	-0.039780	-0.005085	0.022366	0.015303	0.011778
DB	0.005659	-0.005084	-0.012995	-0.016653	0.018948	0.001280	-0.010399	-0.417391	-0.181136	1.000000	...	-0.046250	-0.059735	-0.033105	-0.034325	-0.035612	-0.025799	-0.019604	-0.012832	-0.011097	-0.005074
DÖT	0.041470	0.005734	-0.003340	0.045750	-0.011628	0.001866	0.014714	-0.330324	-0.140319	-0.043228	...	0.072753	0.063816	0.019115	0.032271	0.008000	0.036889	0.006667	-0.005970	0.002188	-0.003420
EUT	0.087064	0.042417	0.039501	0.024862	-0.010221	0.017357	0.022112	-0.297759	-0.141279	-0.027267	...	0.090035	0.059019	0.042512	0.049625	0.022105	0.024676	0.016664	0.020945	-0.001334	-0.001435
FB	0.129416	0.064690	0.030175	0.083380	-0.005592	0.006523	0.014475	-0.230900	-0.137992	-0.046250	...	1.000000	0.172609	0.082908	0.085865	0.143654	0.079199	0.009244	-0.003426	0.000199	0.003417
FKB	0.106025	0.041446	0.040284	0.091493	0.011795	-0.022790	-0.017570	-0.159207	-0.119830	-0.059735	...	0.172609	1.000000	0.166006	0.124785	0.141729	0.156235	0.064302	0.007911	0.002714	-0.001466
GAS	0.128855	0.053225	0.068447	0.065059	0.001221	-0.002092	-0.002947	-0.134248	-0.080377	-0.033105	...	0.082908	0.166006	1.000000	0.096039	0.106244	0.090018	0.089072	0.057383	0.067329	-0.000141
HB	0.111821	0.049651	0.079492	0.019361	0.007406	-0.002664	-0.005810	-0.126064	-0.057854	-0.034325	...	0.085865	0.124785	0.096039	1.000000	0.110060	0.090707	0.068995	0.048971	0.009890	-0.000064
HS	0.121980	0.113149	0.128956	-0.016518	0.003255	-0.009928	-0.001857	-0.097665	-0.053092	-0.035612	...	0.143654	0.141729	0.106244	0.110060	1.000000	0.202821	0.130826	0.096938	0.022058	0.014396
KBA	0.096114	0.052736	0.112067	-0.012317	0.006380	-0.010439	-0.008426	-0.082703	-0.039780	-0.025799	...	0.079199	0.156235	0.090018	0.090707	0.202821	1.000000	0.185762	0.123108	0.078569	0.047130
KKS	0.097218	0.048886	0.123169	-0.009550	0.001681	-0.003190	0.002367	-0.050781	-0.005085	-0.019604	...	0.009244	0.064302	0.089072	0.068995	0.130826	0.185762	1.000000	0.174660	0.096591	0.055314
KH	0.121421	0.058034	0.161027	-0.006741	0.002117	0.010967	0.004487	-0.043330	0.022366	-0.012832	...	-0.003426	0.007911	0.057383	0.048971	0.096938	0.123108	0.174660	1.000000	0.344127	0.061897
KM	0.027718	0.011181	0.038945	-0.003937	-0.015502	0.006252	0.014997	-0.024688	0.015303	-0.011097	...	0.000199	0.002714	0.067329	0.009890	0.022058	0.078569	0.096591	0.344127	1.000000	0.196173
OSKS	0.033211	0.015839	0.041491	-0.001252	-0.003672	-0.002566	0.002182	-0.008640	0.011778	-0.005074	...	0.003417	-0.001466	-0.000141	-0.000064	0.014396	0.047130	0.055314	0.061897	0.196173	1.000000

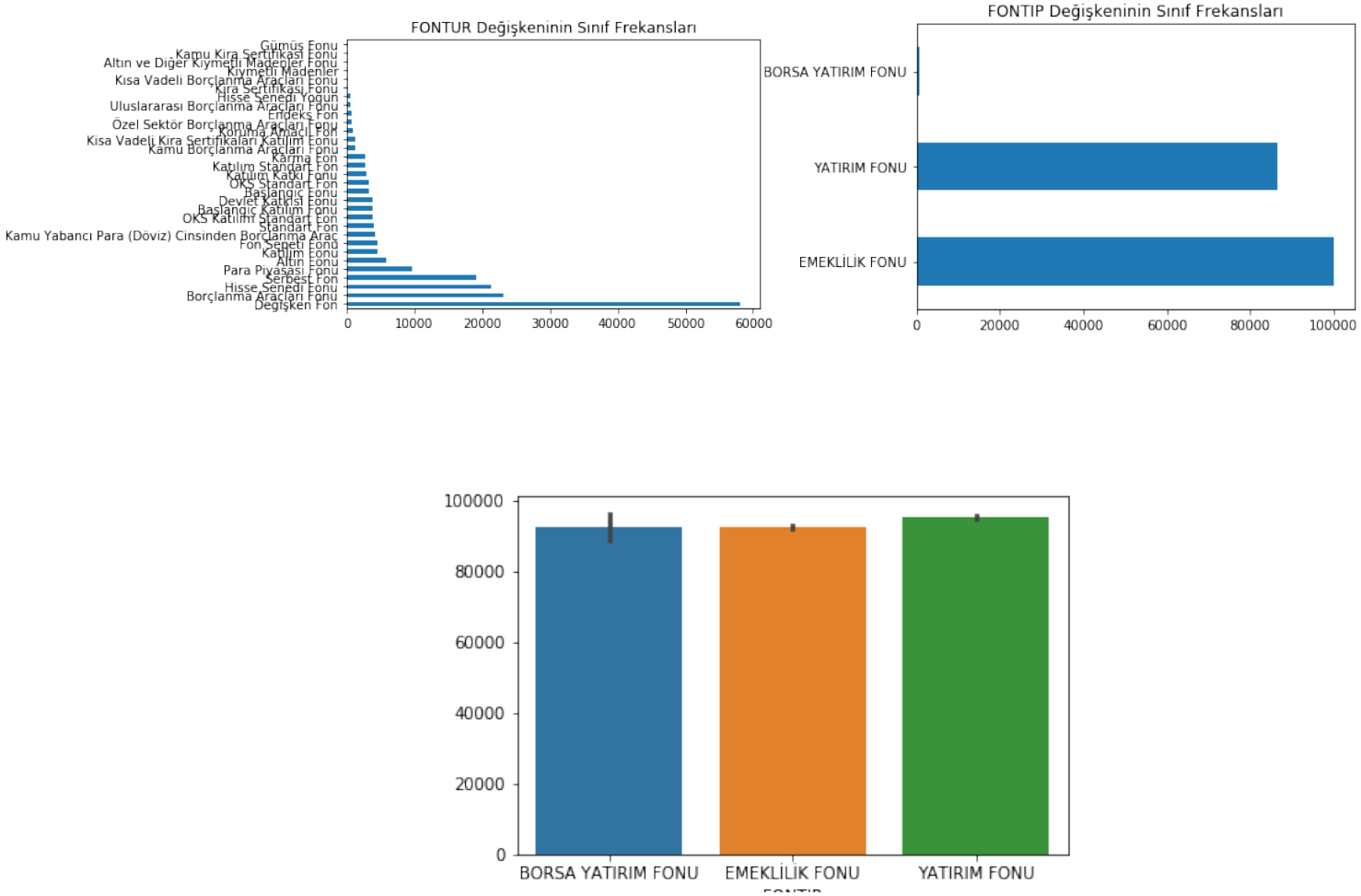
22 rows × 22 columns

Şekil 4.6: Korelasyon matrisi

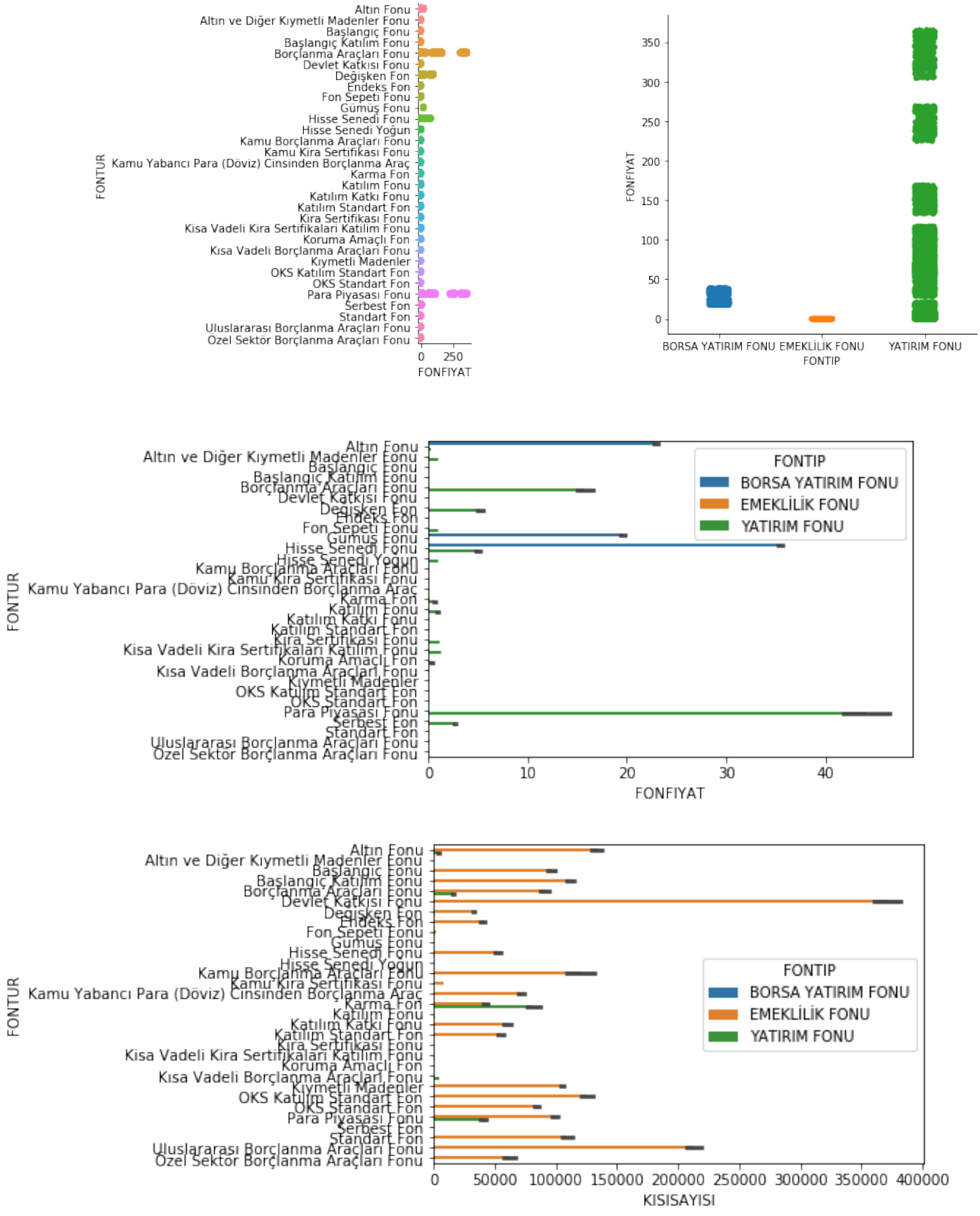
Korelasyon matrisine şekil 4.6'da bakıldığı zaman, değişkenlerin aralarındaki ilişkinin yön ve büyüklük değerlerini görmekteyiz. Genel olarak incelendiğinde, 22 satırda ve 22 kolon matrisinde değerler arasında ciddi bir tutarsızlığın görülmediği ve bazı değişkenler arasında korelasyon katsayısının daha iyi olduğu gözlemlenmektedir. Bu değişkenlerden, FONTOPLAMDEGER - TEDAVPAYSAYISI - KISISAYISI arasındaki ilişkinin diğer değişkenlere oranla daha güçlü olduğu görülmektedir. Veri seti özelikleri bölümünde veriler hakkındaki açıklamalar da, bu güçlü ilişkinin nedenini desteklemektedir. Bu değişkenlerin bağımsız değişken olarak seçilen FONFIYAT değerine olan katkıları dikkate alındığında diğer değişkenlere oranla daha iyi olduğu görülür ve korelasyon matrisinde menkul oran değişkenleri katsayıları da genel olarak normal gözlenmektedir. FONFIYAT değişkeninin FAIZ, DOLARFIYAT ve ALTINFIYAT ile doğrudan korelasyon ilişkisinin zayıf olduğu tespit edilmektedir. Ama ALTINFIYAT değişkeninin, FAIZ ile arasında ters yönde ve DOLARFIYAT ile de aynı yönde güçlü bir ilişkisi olduğu incelenmiştir.

4.1.2.3. Veri Görselleştirme

Bu bölümde veri setindeki kategorik ve sayısal değişkenler için görselleştirme çalışmaları yapılacaktır. Veri ön işlemenin bir önceki adımlarında verinin ideal bir veri setine dönüştürülme süreçleri yapılmış ve bu adımda ise bazı değişkenler üzerinden temel bazı grafikselleştirme işlemleri yapılarak bölüm sonlandırılacaktır.

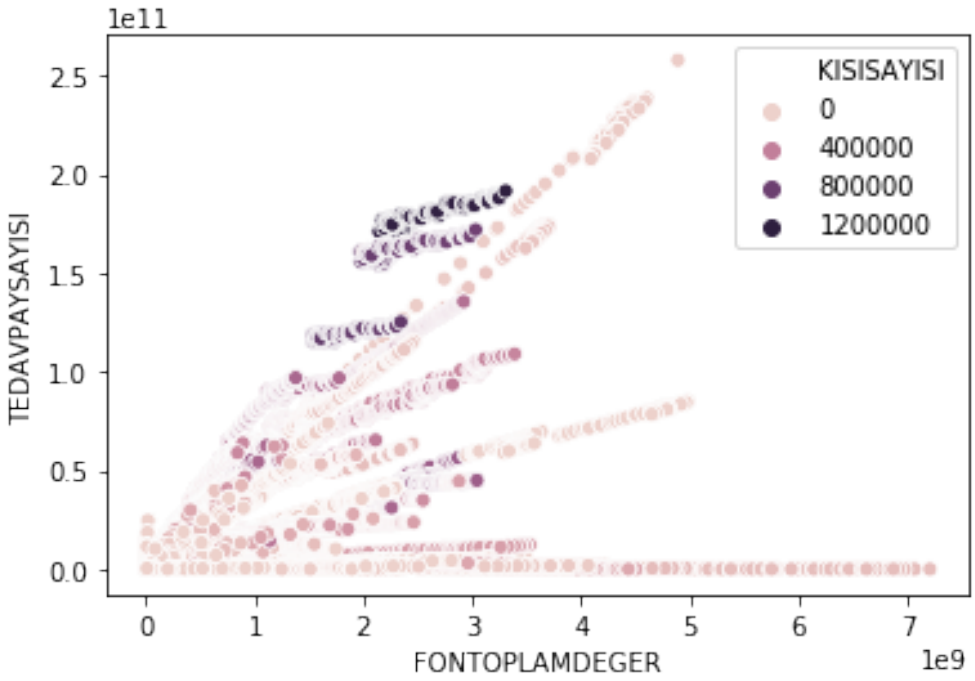


Şekil 4.7: Kategorik değişken dağılım grafikleri

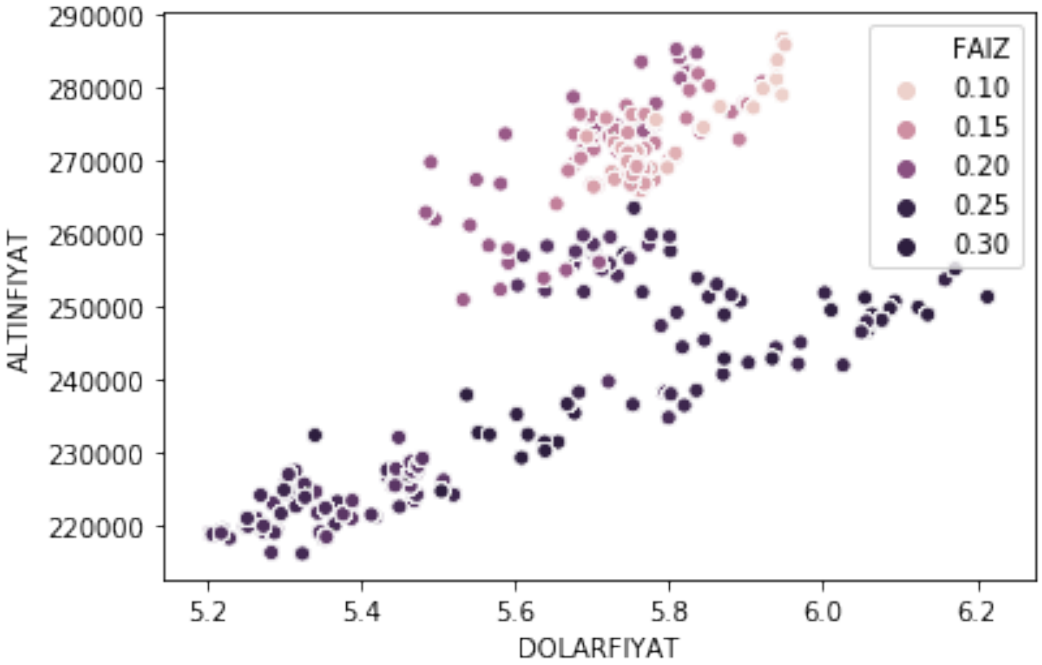


Şekil 4.8: Fon fiyat ve kişi sayısının kategorik değişken grafikleri

Veri görselleştirme amacımız gereği, belli değişkenlerin veri ile olan ilişkilerinin daha anlaşılır olmasını sağlamak için grafikleştirme çalışmaları yapıldı ve bu çalışmalardan, şekil 4.7 ve 4.8'de kategorik değişkenlerin dağılımı ve bağımsız değişken ile olan bağlantılarının görselleştirme işlemleri yapılmıştır. Son olarak yapılan şekil 4.9 ve 4.10'daki grafiklerde, Toplam-Pay-Kisi (FONTOPLAMDEGER-TEDAVPAYSAYISI-KISISAYISI) ve veri setimizde mevcut olan 2019 yılına ait Faiz-Altın-Dolar(FAIZ-ALTINFIYAT-DOLARFIYAT) ilişkisinin korelasyon grafiği verilmiştir. Bir önceki bölüm olan veri korelasyon matris değerleri incelenirken altın değişkeninin, dolar ile pozitif ama faiz değişkeni ile de negatif bir ilişkisinin olduğu ifade edilmişti, bulunan şekil 4.10'daki grafik bu yorumu desteklemektedir.



Şekil 4.9: Toplam-Pay-Kisi sayısı korelasyon grafiği



Şekil 4.10: 2019 Altın-Dolar-Faiz korelasyon grafiği

4.2. MODEL ÖLÇÜM METRİKLERİ

Fon fiyatları tahmini için model geliştirirken, yapılan uygulama sonuçlarının başarı değerlendirmesini ölçecek metriklere ihtiyaç duyulmaktadır. Bu bölümde geliştireceğimiz model çalışmaları regresyon problemlerine dahil olduğundan regresyon model değerlendirme metrikleri kullanılacaktır. Model için kullanılacak 4 algoritmanın sonuç değerlendirmesi aynı ölçüm metriklerine göre değerlendirileceğinden, bu başlık altında genel olarak kullanılacak olan metotlardan bahsedilecektir. Literatür bölümünde, makine öğrenme algoritmalarının değerlendirme sonuçları ile ilgili yapılan çalışmalar incelendiğinde genel olarak çok sık kullanılan ölçüm metrikleri;

1. Hata Karelerinin Ortalaması - Mean Squared Error (MSE)

Hata kare ortalaması, regresyon problemlerinde tahmin eğrisinin gerçek değer noktalarına ne kadar yakın olduğunu belirtir. Matematiksel gösterimi;

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \qquad \textbf{(4.1)}$$

4.1, 4.2 ve 4.3'de ki denklem değişkenleri;

Gözlem sayısı: $\frac{1}{n}$

Gerçek değerler: y_i

Gerçek değerlerin ortalaması: $\bar{y}$

Tahmin edilen değerler: $\hat{y}_i$

MSE değeri, veri setimizdeki bağımsız değişkenlerin tahmin ettiği değer ile gerçek değerin farkının karesinin ortalaması alınarak bulunur. Bulunan değer, birim başına düşen hata payı olarak değerlendirilir. MSE değerinin sıfıra yakın bulunması başarı ölçütü olarak daha iyi bir performans gösterdiği şeklinde ifade edilebilir[76-77].

2. Hata Kare Ortalamasının Karekökü - Root Mean Square Error (RMSE)

Hata karelerinin ortalamasının karekökü, MSE değerinin karekökü alınarak bulunur. RMSE değeri tahmin hatalarının standart sapması olarakta ifade edilebilir. Ayırca RMSE'nin MSE metoduna göre daha avantajlı olan özelliği, bazı durumlarda büyük hataları daha fazla cezalandırma işlevine sahip olduğu belirtilmektedir[76-77]. RMSE matematiksel denklemi;

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad \textbf{(4.2)}$$

3. Ortalama Mutlak Hata - Mean Absolute Error (MAE)

Ortalama mutlak hata, gerçek değerden tahmin edilen değerin farkının mutlak ölçümü alınarak bulunur. Hesaplanan MAE değerleri daha kolay yorumlanabilir oldukları için regresyon ve zaman serisi gibi problemlerde daha çok kullanılmaktadır[76-77]. Matematiksel olarak gösterimi;

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \quad \textbf{(4.3)}$$

4. R Kare Oranı(Belirleme Katsayısı) - R Squared (R^2)

R-kare (R2), bağımsız bir değişken veya bir regresyon modelindeki değişkenler tarafından açıklanan bağımlı bir değişkenin varyans oranını temsil eden istatistiksel bir ölçüdür[76-77]. Yani, bağımsız değişkenin bağımlı değişkendeki, değişikliği açıklama başarısıdır. Matematiksel denklemi;

$$R2 = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2} \quad \textbf{(4.4)}$$

Kısaca 4 madde halinde tanımlamaları ve matematiksel denklemleri verilen yukarıdaki model ölçüm metrikleri, bu tez çalışmasında önerilen model geliştirme sonuçlarını değerlendirme ölçütünde referans alınarak işlemler yapılacaktır.

4.3. DOĞRUSAL ÇOKLU REGRESYON ALGORİTMALARI İLE MODEL GELİŞTİRME

Model geliştirme çalışmalarının ilk adımı olan doğrusal çoklu regresyonlar ile tahmin işleminde; Kısmi En Küçük Kareler Regresyon (PLSR) ve Ridge Regresyon (RR) algoritmalarıyla uygulamalar yapılacaktır. Bölüm 3'de teorik kapsamlarından bahsedilen PLSR ve RR algoritmaların her birinin kendi bölüm başlığı altında, model geliştirme ve optimizasyon adımlarından bahsedilecektir.

4.3.1. Kısmi En Küçük Kareler Regresyonu (PLSR)

Bu kısımda model geliştirme ve model optimizasyon adımları incelenecektir. Model geliştirme alt başlığında, veri setimizin bağımsız değişkenlerinden bağımlı değişken olan fiyat parametresinin tahmini için ilk uygulama adımları incelenecek ve bir sonraki aşama olarak ise, geliştirilmiş olan model üzerinden optimizasyon işlem adımlarından bahsedilerek, PLSR ile model geliştirme bölümü tamamlanmış olacaktır.

4.3.1.1. Model Geliştirme

Kısmi En Küçük Kareler Regresyonu (PLSR) ile model geliştirme bölümünde, model kurma aşamalarını adımlar halinde örnek kod blokları ve elde edilen sonuçları verilerek detaylı bir şekilde anlatımı yapılacaktır. Toplam 3 adımda model geliştirme çalışması gerçekleştirilecektir.

1.Adım : Veri setinde bağımlı ve bağımsız değişken ile eğitim ve test seti ayrımı;

```
X_p = df_pls.drop(['TARIH','FONTIP','FONTUR','FON','FONFIYAT'], axis = 1)
y_p = df_pls["FONFIYAT"]

X_train_p, X_test_p, y_train_p, y_test_p = train_test_split(X_p, y_p, test_size=0.25, random_state=42)

X_p.head()
```

	FONTOPLAMDEGER	TEDAVPAYSAYISI	KISISAYISI	FAIZ	DOLARFIYAT	ALTINFIYAT	BB	DT	DB	DÖT	...	FB	FKB	GAS	HB	HS	KBA	KKS	KH	KM	OSKS
0	8.722176e+07	4.350000e+06	0	0.2302	5.2905	219270.871675	100.00	0.00	0.00	0.00	...	0.00	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1	1.238294e+07	7.000000e+05	0	0.2302	5.2905	219270.871675	100.00	0.00	0.00	0.00	...	0.00	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	1.491130e+07	4.800000e+05	0	0.2302	5.2905	219270.871675	99.83	0.17	0.00	0.00	...	0.00	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	1.465104e+09	6.335717e+10	325392	0.2302	5.2905	219270.871675	1.01	3.44	85.25	3.07	...	7.23	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	1.784293e+09	7.415738e+10	325460	0.2302	5.2905	219270.871675	0.25	99.75	0.00	0.00	...	0.00	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

5 rows × 21 columns

Şekil 4.11: Bağımlı-bağımsız değişken ve eğitim-test seti ayrımı

Veri kümesi üzerinden ilk olarak yapılacak olan bağımlı, bağımsız değişken ve veri setinin eğitim, test olarak ayrımının, 4 algoritmanın model geliştirme aşamasının ortak ilk adımı olduğu için sadece bu başlıkta detaylı anlatımı yapılarak, diğer 3 algoritmanın model geliştirme aşamasında anlatımı yapılmayacaktır. Bu bağlamdan hareketle veri üzerinden, bağımlı değişken fon fiyat bilgisi y ile bağımsız değişkenler ise X parametresine değer ataması yapılarak, tahmin işlemi için girdi ve çıktı verilerinin ayrıştırma süreçleri gerçekleştirilir. Daha sonra ayrımı yapılmış değişkenler üzerinden, veri setinin eğtim ve test olarak bölünmesi gerekir. Eğitim seti ile ilk model kurma aşamasında öğrenme süreci gerçekleştirilir. Öğrenme süreci yapılmış modelin, öğrenme başarı performansını ölçmek için ayrılmış olan test verisi ile öğrenme durumu değerlendirilir. Uygulama bölümü için önemli olan bu ayrıştırma işlemi, veri setimizin genelinde %75 eğitim ve %25 test verisi oluşturacak formatta yapılmıştır. Bölünmüş olan verinin, bağımlı (y_train, y_test) ve bağımsız

(X_train, X_test) değişkenler bazında olacak şekilde eğtim ve test olarak aktarım işlemleri yapılmıştır.

2.Adım: Eğitim seti ile model kurma;

Bu adımda, X_train bağımsız değişkenler ile y_train bağımlı değişken üzerinden ilk kurulan PLSR modeli eğitilir. Bu eğitilen modelin, bağımsız değişken girdilerinin katsayıları ve model parametre yapıları şekil 4.12'de gösterilmektedir.

```
pls_model

PLSRegression(copy=True, max_iter=500, n_components=2, scale=True, tol=1e-06)

pls_model.coef_

array([[ 8.84294439],
       [-4.69831641],
       [-3.03243924],
       [ 0.0337065 ],
       [-0.09314651],
       [-0.02433704],
       [-0.33800515],
       [-0.31991584],
       [-0.81996284],
       [ 0.91092399],
       [-0.09762054],
       [ 1.74953948],
       [ 2.26792273],
       [ 1.19959689],
       [-0.41431445],
       [-1.57986807],
       [-1.26811436],
       [-0.8361369 ],
       [-0.61910158],
       [-0.24914284],
       [-0.09490851]])
```

Şekil 4.12: PLSR Model yapısı ve bağımsız değişken katsayıları

İlk satırda pls_model ile NIPALS algoritmasının parametreleri görülmekte, içerik olarak maximum iterasyon ve bileşen sayıların default değerleri mevcuttur ve bu parametre değişiklikleri optimizasyon bölümünde yapılacaktır. İkinci satırda modelin bağımsız değişken katsayıları verilmiştir. Burada öğrenme modeli için hesaplanan çoklu doğrusal bir fonksiyon denklemin de, birden fazla olan bağımsız değişkenlerin katsayı değerleri bulunmuştur.

3.Adım: Model eğitim ve test seti tahmin bilgileri;

Tablo 4.5: PLSR ölçüm metrik sonuçları

Ölçüm Metrikleri	Eğitim	Test
MSE	497.8090200774829	506.0252001342653
RMSE	22.31163418661849	22.495003892737277
MAE	7.099829131737095	7.119836719007614
R2	0.1273657565331895	0.13348626833437138

Yukarıdaki 4.5'teki tablodan görüleceği üzere, ölçüm metrik sonuçları eğitim ve test verisine göre elde edilmiştir. Ayrıca, modelin mevcut default değerleri üzerinden herhangi bir optimizasyon işlemi yapılmadan bu sonuçlara ulaşılmıştır. Tablodaki sonuçlardan, eğitim ve test değerleri arasında küçük farklılıkların olduğu görülmektedir. Her ne kadar eğitim seti üzerinden alınan bazı sonuçların daha iyi olduğu görülse de, model için daha doğru bir değerlendirme referansı ise test verisinden elde edilen sonuçlar ile olmuştur.

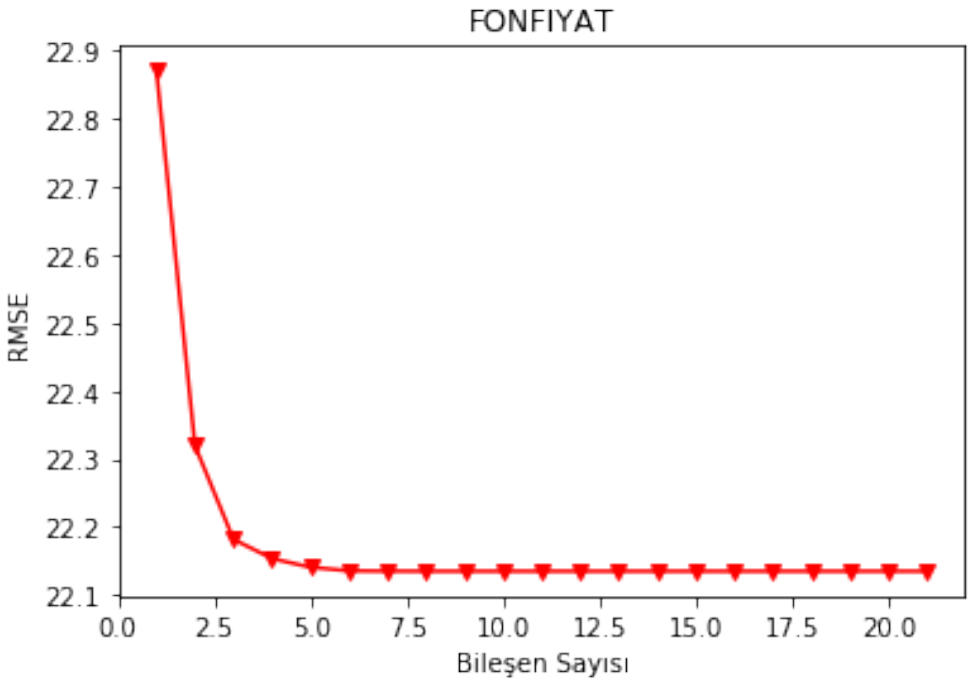
4.3.1.2. Model Optimizasyonu

Kısmi En Küçük Kareler Regresyonu (PLSR) kullanılarak elde edilen tahmin sonuçlarında model optimizasyon parametreleri için yapılan literatür ve örnek çalışmalar incelendiğinde[78], PLSR temel çalışma yapısından hareketle, bağımsız değişkenlerin daha az sayıda ve aralarında çoklu bağlantı problemi olmayan bileşenlere indirgenip model kurma fikrine dayanıyor olması, bileşen sayısını optimizasyon konusu yapmıştır. Model doğrulama ve optimum parametre için Cross-validation metodu kullanılacaktır. Cross-validation, makine öğrenmenin veriler üzerinde doğru ve objektif bir öğrenme süreci için yaygın olarak kullanılan, model seçimi ve performans değerlendirmede tercih edilen basit ve etkili bir yöntemdir[79]. Burada, cross-validation yöntemlerinden K-Fold yöntemi kullanılacak ve optimizasyon çalışması toplam 2 adımdan oluşacaktır.

1.Adım: Optimum bileşen sayısı bulmak;

```
#CV
cv_10 = model_selection.KFold(n_splits=10, shuffle=True, random_state=1)


#Hata hesaplamak için döngü
RMSE = []

for i in np.arange(1, X_train_p.shape[1] + 1):
    pls = PLSRegression(n_components=i)
    score = np.sqrt(-1*cross_val_score(pls, X_train_p, y_train_p, cv=cv_10, scoring='neg_mean_squared_error').mean())
    RMSE.append(score)

#Sonuçların Görselleştirilmesi
plt.plot(np.arange(1, X_train_p.shape[1] + 1), np.array(RMSE), '-v', c = "r")
plt.xlabel('Bileşen Sayısı')
plt.ylabel('RMSE')
plt.title('FONFIYAT');
```

Şekil 4.13: PLSR optimum bileşen sayısı kod bloğu



Şekil 4.14: PLSR optimum bileşen sayısı

Şekil 4.13'teki kod bloğundan görüleceği üzere, k-flod yönteminde veri üzerinde işlem yapılırken veriyi bölme sayısı (n_splits) parametresinin seçimi için kullanılan değer aralığı, 5 ile 10 arasında olmaktadır[79]. Bu çalışma için yapılan deneysel sonuçlarda ise, en optimum değer 10 olarak bulunmuştur. Kod bloğun

çıktısı olan şekil 4.14'te, bileşen sayısı 5 değerinin bir kırılma noktası olduğu ve sonrası değerler için sabit RMSE oranları elde edildiği görülmektedir. Bu bağlamda, tercih edilen en optimum bileşen sayısı 6 olarak seçilmiştir.

2.Adım: Doğrulanmış model tahmin sonuçları;

Tablo 4.6: PLSR ölçüm metrik sonuçları - 2

Ölçüm Metrikleri	
MSE	497.0027614702058
RMSE	22.29355874395575
MAE	6.705839394559914
R2	0.14893622417341656

Tablo 4.6'da optimizasyon işlemleri yapıldıktan sonra doğrulanmış model verilerinin sonuçlarına bakıldığında, fon fiyat tahmini ile gerçek değer arasındaki farkın açıklanma metriği RMSE değerini baz alarak değerlendirdiğimizde, fon fiyat tahmini yaptığımızda birim başına düşen hata payı oranının artı eksi(+/-) 22.29 olarak bir sapma değerinin olduğu görülmektedir. Tahmin başarısının, RMSE sonuç çıktısına göre düşük olduğu değerlendirilmiştir.

4.3.2. Ridge Regresyonu (RR)

Bu bölümde Ridge Regresyonu (RR) ile model geliştirme çalışmaları yapılacak, daha önce bölüm 3'de ayrıntılı olarak RR algoritması hakkında bilgilendirilme yapılmıştır. Burada ise, iki alt başlık olarak; model geliştirme ve optimizasyon konuları incelenecektir.

4.3.2.1. Model Geliştirme

Bu kısımda yapılan model çalışmasının, PLSR model geliştirme aşamalarıyla benzer olması ve tekrar olmaması amacıyla aynı adımlarda ayrıntılı bir anlatım yapılmayacaktır. Ayrıca, uygulamanın birinci adımı olan veri kümesinin test ve

eğitim seti olarak ayrılma aşaması da ortak olduğu için bu adım atlanarak, 2'inci adımdan başlayarak model çalışmamız gerçekleştirilmiş olacaktır.

2.Adım: Eğitim seti ile model kurma;

```
ridge_model
```

```
Ridge(alpha=0.1, copy_X=True, fit_intercept=True, max_iter=None,
      normalize=False, random_state=None, solver='auto', tol=0.001)
```

```
ridge_model.coef_
```

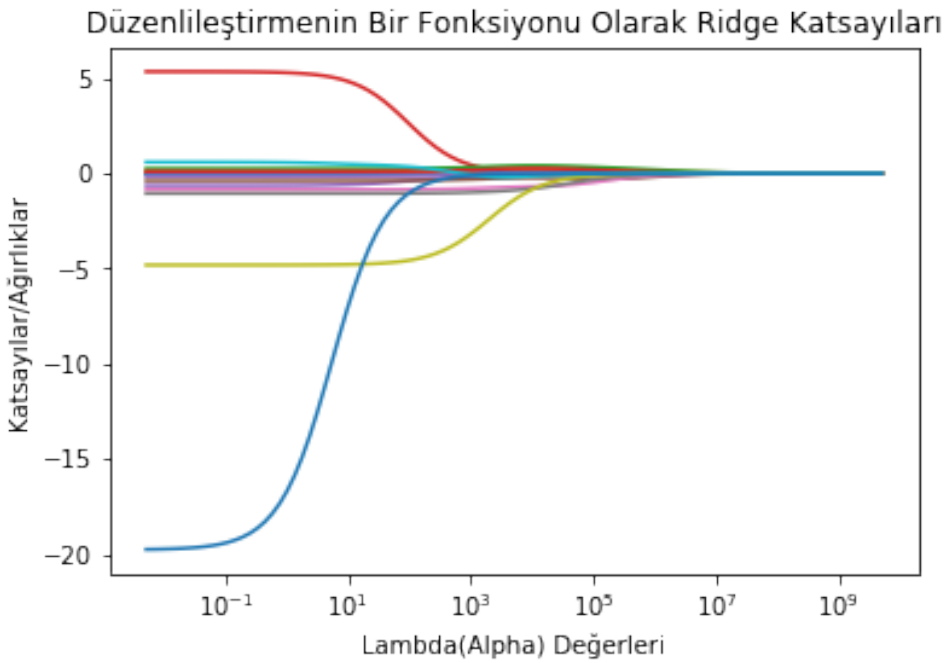
```
array([ 1.99548506e-08, -4.36501444e-10, -1.01864979e-05,  5.33789312e+00,
       -6.33703278e-01,  2.45254214e-06, -1.37536605e-03, -1.26321075e-02,
       -2.43176494e-02,  3.68163153e-02, -1.56943704e-02,  1.05959126e-01,
        2.52256301e-01,  1.04918931e-01, -1.40978035e-01, -3.76084541e-01,
       -8.49985257e-01, -1.04985067e+00, -4.80938207e+00,  5.91301679e-01,
       -1.93941104e+01])
```

Şekil 4.15: RR Model yapısı ve bağımsız değişken katsayıları

Şekil 4.15'teki modelin birinci satırında model yapısı ve parametreleri mevcut iken, ikinci satırda tahmin işlemi yapılacak olan fonksiyon değişken katsayıları bulunmaktadır. Burada, model parametresi olan alpha(lambda) değişkeni, tahmin fonksiyonunda ceza teriminin katsayı değeridir. Deneysel çalışmalar için manuel 0.1 değeri verilmiştir. Diğer model parametreleri default değerlerdir. Alpha(lambda) değişkenin, model katsayıları değerlerine olan etkisini deneysel olarak göstermek için şekil 4.16 ve 4.17'deki çalışmalar yapılmıştır.

```python
lambdalar = 10**np.linspace(10,-2,100)*0.5

ridge_model = Ridge()
katsayilar = []

for i in lambdalar:
    ridge_model.set_params(alpha = i)
    ridge_model.fit(X_train_r, y_train_r)
    katsayilar.append(ridge_model.coef_)



ax = plt.gca()
ax.plot(lambdalar, katsayilar)
ax.set_xscale('log')

plt.xlabel('Lambda(Alpha) Değerleri')
plt.ylabel('Katsayılar/Ağırlıklar')
plt.title('Düzenlileştirmenin Bir Fonksiyonu Olarak Ridge Katsayıları');
```

Şekil 4.16: RR lambda(alpha) örnek kod bloğu



Şekil 4.17: RR lambda(alpha) örnek grafiği

Şekil 4.17'deki grafik çıktısına bakıldığında her bir renk, bir değişkenin katsayı değerini ifade etmekte ve lambda(alpha) değeri değiştikçe katsayı/ağırlık değerlerinde değişim olduğu görülmektedir. Deneysel çalışma için yapılmış olan şekil 4.16'daki kod bloğu ve sonuç grafiğinden anlaşılacağı üzere, alpha parametresinin alacağı değerin katsayılara olan etkisi gözlemlenmektedir.

3. Adım: Model eğitim ve test seti tahmin bilgileri;

Tablo 4.7: RR ölçüm metrik sonuçları - 1

Ölçüm Metrikleri	Eğitim	Test
MSE	489.71681475048916	496.9772456519872
RMSE	22.12954619395728	22.292986467765758
MAE	6.685656583569678	6.706254687410646
R2	0.1415509865886838	0.14897991726786186

Tablo 4.7'de model ölçüm metrik sonuçlarına bakıldığında, PLSR test ve eğitim sonuçlarına göre daha iyi olduğu görülmektedir. Ayrıca, eğitim ve test değerleri arasında da küçük farklılıkların olduğu görülmektedir. Bazı ölçüm metriklerinin eğitim seti üzerinden alınan sonuçlarının, test verisinden daha iyi olduğu görülse de, model için daha doğrulanmış sonuçlar test verisinden elde edilen sonuçlar baz alınarak yapılmaktadır.

4.3.2.2. Model Optimizasyonu

Ridge Regresyonu (RR) model parametre optimizasyon kısmında alpha(lambda) değişkenin optimize edilmesi ile ilgili çalışmalar yapılacaktır. Literatür bölümünde yapılan araştırmalar ve model geliştirme bölümündeki birinci adımda gerçekleştirilen deneysel çalışmalardan anlaşılacağı üzere, model tahmin fonksiyonu üzerinde çok önemli bir etkisi olan ceza teriminin katsayısı alpha değişkeni, optimizasyon konusu olacaktır [79-80]. Optimum değeri bulmak için

Cross-validation metodu kullanılacak ve yapılacak işlem sayısı toplam 2 adımdan oluşacaktır.

1. **Adım:** Optimum alpha(lambda) değerinin bulunması;

```
lambdalar = 10**np.linspace(10,-2,100)*0.5
```

```
lambdalar[0:5]
```

```
array([5.00000000e+09, 3.78231664e+09, 2.86118383e+09, 2.16438064e+09,
       1.63727458e+09])
```

```
from sklearn.linear_model import RidgeCV
ridge_cv = RidgeCV(alphas = lambdalar,
                   scoring = "neg_mean_squared_error",
                   normalize = True)
```

```
ridge_cv.fit(X_train_r, y_train_r)
```

```
RidgeCV(alphas=array([5.00000000e+09, 3.78231664e+09, 2.86118383e+09, 2.16438064e+09,
       1.63727458e+09, 1.23853818e+09, 9.36908711e+08, 7.08737081e+08,
       5.36133611e+08, 4.05565415e+08, 3.06795364e+08, 2.32079442e+08,
       1.75559587e+08, 1.32804389e+08, 1.00461650e+08, 7.59955541e+07,
       5.74878498e+07, 4.34874501e+07, 3.28966612e+07, 2.48851178e+07,
       1.88246790e+07, 1.42401793e+0...
       3.28966612e-01, 2.48851178e-01, 1.88246790e-01, 1.42401793e-01,
       1.07721735e-01, 8.14875417e-02, 6.16423370e-02, 4.66301673e-02,
       3.52740116e-02, 2.66834962e-02, 2.01850863e-02, 1.52692775e-02,
       1.15506485e-02, 8.73764200e-03, 6.60970574e-03, 5.00000000e-03]),
        cv=None, fit_intercept=True, gcv_mode=None, normalize=True,
        scoring='neg_mean_squared_error', store_cv_values=False)
```

```
ridge_cv.alpha_
```

```
0.005
```

Şekil 4.18: RR optimum alpha(lambda) değeri ve kod bloğu

Kod bloğu şekil 4.18'de anlaşılacağı üzere, model geliştirme çalışmasında elde ettiğimiz deneysel sonuç dizisini, eğtim verileri üzerinden cross-validation metodu kullanılarak optimum alpha değeri 0.005 olarak bulunmuştur.

2.**Adım:** Doğrulanmış model tahmin sonuçları;

Tablo 4.8: RR ölçüm metrik sonuçları - 2

Ölçüm Metrikleri	
MSE	497.0262344776716
RMSE	22.294085190419267
MAE	6.69722792342646
R2	0.14889602917267908

Tablo 4.6'daki PLSR model optimizasyon sonuçları ile Tablo 4.8'deki RR doğrulanmış model verilerinin sonuçlarına bakıldığında, birbirine çok yakın değerler olduğu görülmektedir. RR modelin tahmin fonksiyonu, RMSE değerine göre fon fiyat tahmini için birim başına düşen hata payı oranının artı eksi(+/-) 22.29 olarak, bir sapma değerinin olduğu görülmektedir. Tahmin başarısının, RMSE sonuç değerine göre düşük olduğu saptanmaktadır.

4.4. DOĞRUSAL OLMAYAN ÇOKLU REGRESYON ALGORİTMALARI İLE MODEL GELİŞTİRME

Model geliştirme çalışmalarının ikinci adımı olan doğrusal olmayan çoklu regresyonlar ile tahmin işleminde; Destek Vektör Regresyonu (SVR) ve Yapay Sinir Ağları(YSA) algoritmalarıyla model geliştirme süreçlerinin devamı yapılacaktır. Bölüm 3'de teorik kapsamlarından bahsedilen SVR ve YSA algoritmaların her birinin kendi bölüm başlığı altında, model geliştirme ve optimizasyon adımlarından bahsedilecektir.

4.4.1. Destek Vektör Regresyonu (SVR)

Bu bölümde Destek Vektör Regresyonu (SVR) ile model geliştirme çalışmaları yapılacaktır. İki alt başlıkta; model geliştirme ve optimizasyon konuları incelenecektir.

4.4.1.1. Model Geliştirme

Bu kısımda yapılacak model çalışması, toplamda 2 adımdan oluşmaktadır. Daha önceki model çalışmalarında, PLSR model geliştirme adımlarından olan veri kümesinin test ve eğitim seti olarak ayrılma aşaması ortak olduğu için bu adım atlanarak, 2'inci adımdan başlayarak model çalışması gerçekleştirilecektir.

2.Adım: Eğitim seti ile model kurma;

```
scaler = StandardScaler()
scaler.fit(X_train_s)
```

```
StandardScaler(copy=True, with_mean=True, with_std=True)
```

```
X_train_scaled_s = scaler.transform(X_train_s)
```

```
X_test_scaled_s = scaler.transform(X_test_s)
```

```
from sklearn.svm import SVR
```

```
svr_rbf = SVR("rbf").fit(X_train_scaled_s, y_train_s)
```

```
svr_rbf
```

```
SVR(C=1.0, cache_size=200, coef0=0.0, degree=3, epsilon=0.1, gamma='scale',
    kernel='rbf', max_iter=-1, shrinking=True, tol=0.001, verbose=False)
```

Şekil 4.19: SVR model yapısı ve veri standartlaştırma

SVR'nin dikkat edilmesi gereken kritik özelliği, marjinal verilere karşı duyarlı bir yapıya sahip olmasıdır. Bu sebepten ötürü, verilerden daha iyi tahmin değeri sağlamak için, verilerin standartlaştırılması gerekmektedir. Şekil 4.19'da, modelin ilk satırında, StandardScaler metodu kullanılarak, değişkenlerin ortalaması sıfır ve standart sapması bir olacak şekilde standartlaşma işlemi yapılmıştır[62]. Bir sonraki kod satırında, X bağımsız değişkeninin eğtim ve test setleri üzerinde veri standartlaştırma yapılmıştır. Ardından SVR algoritmasının, rbf(Radial Basis Function) çekirdek (kernel) fonksiyonu ile svr_rbf model nesnesi oluşturulmuştur.

3. Adım: Model eğitim ve test seti tahmin bilgileri;

Tablo 4.9: SVR ölçüm metrik sonuçları - 1

Ölçüm Metrikleri	Eğitim	Test
MSE	531.8675777625025	542.8368894117968
RMSE	23.06225439462722	23.298860259931104
MAE	3.677658112113757	3.711111249858252
R2	0.06766281319477463	0.07045020958408388

Tablo 4.9'da model ölçüm metrik değerlerine bakıldığında, eğitim ve test sonuçlarının PLSR ve RR model sonuçlarına göre daha düşük olduğu görülmektedir. Son olarak, SVR model eğitim ve test değerleri arasında da farklılıkların olduğu görülmektedir. Model kabul değerleri, test verisinden elde edilen sonuçlar baz alınarak yapılmaktadır.

4.4.1.2. Model Optimizasyonu

Destek Vektör Regresyonu (SVR) model optimizasyon bölümünde, şekil 4.19'daki model parametreleri üzerinde optimizasyon işlemleri yapılacaktır. Tez çalışmasının daha önceki bölümleri olan literatür incelmesinde SVR üzerine yapılan çalışmalar incelenmiş ve metot bölümünde ayrıntılı olarak SVR teorik ve matematiksel denklem yapısı incelenmişti. Bu çalışmalardan yola çıkılarak SVR modelin çekirdek fonksiyonu rfb için, karmaşıklık ya da ceza değeri olarak ifade edilen ***C*** parametresi optimizasyon problemin konusu olmuştur[62,64]. Model optimizasyon işlemleri toplam 2 adımdan oluşacaktır.

1. **Adım:** Optimum ***C*** parametresi bulunması;

```
svr_rbf

SVR(C=1.0, cache_size=200, coef0=0.0, degree=3, epsilon=0.1, gamma='scale',
    kernel='rbf', max_iter=-1, shrinking=True, tol=0.001, verbose=False)

#'C': [0.1, 1, 10, 100, 1000]
svr_params = {"C": np.arange(0.1,2,0.1)}
svr_cv_model = GridSearchCV(svr_rbf,svr_params, cv = 10)
svr_cv_model.fit(X_train_scaled_s, y_train_s)

pd.Series(svr_cv_model.best_params_)[0]

svr_tuned = SVR("rbf", C = pd.Series(svr_cv_model.best_params_)[0]).fit(X_train,
                                                                        y_train)

y_pred = svr_tuned.predict(X_test)
```

Şekil 4.20: SVR optimum C parametre değeri ve kod bloğu

Şekil 4.20'deki kod bloğunun birinci satırında, *C* parametresi için deneysel çalışmalarda kullanılan örnek veriler verilmiştir. *C* değişken optimizasyonu ile ilgili incelen bir çok çalışmada verilen değer aralıkları uyarlanacak probleme göre belli bir sayı seti verilerek işlemler yapılmıştır. Yapılan çalışmalarda, *C* parametresi (ceza parametre) aralığının çok küçük ya da büyük olması durumunda model öğrenemsi için ciddi minimizasyon problemlerine yol açmaktadır[81-83]. Bu bağlamda, uygulamda kullandığımız *C* parametre aralığını veri setimiz için daha uygun olduğu değerlendirilmiştir. Daha sonra, Cross-validation metodunun hiper parametre optimizasyon tekniği olan GridSearchCV kullanılarak her bir parametre değeri için en doğru model buluncaya kadar, eğtim seti üzerinden çalışarak *C* parametresinin optimum değer bulunur. Son olarak, bulunan optimum değerler ile nihayi doğrulanmış model elde edilir.

2**.Adım:** Doğrulanmış model tahmin sonuçları;

Tablo 4.10: SVR ölçüm metrik sonuçları - 2

Ölçüm Metrikleri	
MSE	497.0262344776716
RMSE	22.294085190419267
MAE	6.69722792342646
R2	0.14889602917267908

Tablo 4.10'daki SVR doğrulanmış model sonuçları, RMSE değerine göre fon fiyat tahmini için birim başına düşen hata payı oranının artı eksi(+/-) 22.29 olarak, bir sapma değerinin olduğu görülmektedir. SVR model sonuçları incelendiğinde, PLSR ve RR model geliştirme çalışmalarının sonuçlarına göre çok daha düşük olduğu gözlemlenmektedir. Tahmin başarısının, RMSE çıktısına göre en düşük performansa sahip model olarak incelenmiştir.

4.4.2. Yapay Sinir Ağları(YSA)

Yapay Sinir Ağları(YSA) ile model geliştirme çalışmaları yapılacak, daha önce bölüm 3'de ayrıntılı olarak YSA algoritması hakkında bilgilendirilme yapılmıştır. Burada ise, iki alt başlıkta; model geliştirme ve optimizasyon konuları incelenerek model geliştirme süreçleri tamamlanmış olacaktır.

4.4.2.1. Model Geliştirme

Bu bölümde yapılacak olan Yapay Sinir Ağları(YSA) ile model geliştirme 2 adımdan oluşacaktır. Bu tez çalışmasında, model geliştirmenin ilk örneği olan PLSR ile model geliştirmede, veri setinin eğitim ve test seti olarak ayrılmasına dair ayrıntılı açıklamalar model çalışmalarının ortak noktası olduğundan önceki bölümlerde yapıldığı dikkate alınarak direk 2. adımdan başlanacaktır.

2.Adım: Eğitim seti ile model kurma;

```
scaler = StandardScaler()
scaler.fit(X_train_y)
```

```
StandardScaler(copy=True, with_mean=True, with_std=True)
```

```
X_train_scaled = scaler.transform(X_train_y)
```

```
X_test_scaled = scaler.transform(X_test_y)
```

```
from sklearn.neural_network import MLPRegressor
```

```
mlp_model = MLPRegressor(hidden_layer_sizes = (100,20)).fit(X_train_scaled, y_train_y)
```

```
mlp_model
```

```
MLPRegressor(activation='relu', alpha=0.0001, batch_size='auto', beta_1=0.9,
             beta_2=0.999, early_stopping=False, epsilon=1e-08,
             hidden_layer_sizes=(100, 20), learning_rate='constant',
             learning_rate_init=0.001, max_fun=15000, max_iter=200,
             momentum=0.9, n_iter_no_change=10, nesterovs_momentum=True,
             power_t=0.5, random_state=None, shuffle=True, solver='adam',
             tol=0.0001, validation_fraction=0.1, verbose=False,
             warm_start=False)
```

Şekil 4.21: YSA model yapısı ve veri standartlaştırma

Yukarıdaki şekil 4.21'de YSA model yapısı verilen kod bloğunun birinci satırında, veri standartlaştırma işlemleri yapılmaktadır. YSA ile model kurma aşamasında birden fazla katman ve hücre yapısının olmasından dolayı, verilerdeki aşırı uç değerlerin, aykırılıkların bulunması veya değişken ölçeklerinin varyans yapılarının birbirinden çok farklı olması, bulunacak sonuçların güvenirliliğini düşürmektedir. Özelikle bu sebepten kaynaklı, YSA model geliştirme aşamasında veri standartlaştırma işlemi yapılmaktadır[68]. Burada kullanılacak olan StandardScaler metodu SVR model geliştirme adımında bahsedilmiştir. Ayrıca, ayrıntılı olarak YSA model parametreleri görülmektedir. Model parametrelerinden, gizli katman sayısı deneysel çalışmalar için manuel olarak (100,20) verilmiştir. Son olarakta, aktivasyon fonksiyonu, alpha ve diğer değişkenlerin default değeri bulunmaktadır.

3. Adım: Model eğitim ve test seti tahmin bilgileri;

Tablo 4.11: YSA ölçüm metrik sonuçları - 1

Ölçüm Metrikleri	Eğitim	Test
MSE	27.809947088094503	39.321943328021554
RMSE	5.273513732616471	6.270721117066326
MAE	1.6469758061055841	1.7579658612687985
R2	0.9512505576249017	0.93266540116882

Tablo 4.11'de model ölçüm metrik değerlerine bakıldığında, eğitim ve test sonuçlarının daha önceki model geliştirme bölümlerindeki tablo 4.5, 4.7 ve 4.9'daki sonuçlarına göre çok daha iyi olduğu gözlemlenmektedir. Ayrıca, eğitim ve test değerleri arasında farklılıklar olduğu görülmektedir. Model değerlendirme için test sonuçları referans alınacaktır.

4.4.2.2. Model Optimizasyonu

Yapay Sinir Ağları(YSA) model optimizasyon kısmında, şekil 4.21'de verilen model parametreleri için optimizasyon çalışmaları yapılacaktır. YSA hakkında, literatür ve metot bölümden yapılan araştırmalarda optimizasyon problemlerine konu olunan parametrelerin; aktivasyon(activation), cezalandırma katsayısı(alpha) ve gizli katman sayısı(hidden_layer_sizes) olduğu tespit edilmiştir[68-70,72]. Deneysel çalışmalar yapılarak, optimizasyon yapılacak olan parametrelerin optimum değerlerinin nasıl elde edildiği, toplam 2 adımdan oluşacak şekilde incelenecektir.

1. **Adım:** Optimum activation, alpha ve hidden_layer_size parametrelerinin bulunması;

```
mlp_params = {'alpha': [0.1, 0.01,0.02,0.005],
              'hidden_layer_sizes': [(20,20),(100,50,150),(300,200,150)],
              'activation': ['relu','logistic']}

mlp_cv_model = GridSearchCV(mlp_model, mlp_params, cv = 10)

mlp_cv_model

mlp_cv_model.fit(X_train_scaled, y_train)

mlp_cv_model.best_params_

mlp_tuned = MLPRegressor(alpha = 0.02, hidden_layer_sizes = (100,50,150))

mlp_tuned.fit(X_train_scaled, y_train)

y_pred = mlp_tuned.predict(X_test_scaled)
```

Şekil 4.22: YSA optimum parametre değerleri ve kod bloğu

Yukarıdaki şekil 4.22'nin birinci satırında, optimizasyonu yapılacak parametreler için deneysel çalışmalarda kullanılan örnek veriler verilmiştir. Burada kullanılan deneysel çalışma örnekleri, alpha(ceza terimi katsayısı) için verilecek değer aralığı çok büyük seçilirse modelin eksik öğrenmesi aynı şekilde çok küçük seçildiğinde de aşırı öğrenme problemlerine yol açacağı ifade edilmiştir [72,84]. Bundan dolayı, model çalışmamız için örnek olarak kullanılan, sayısı dizisi belli bir

aralık olarak verilmiş en optimum değer bulunmaya çalışılmıştır. Aynı konsepte, gizli katman sayısı içinde örnek deneysel veriler kullanılarak model için doğru değerler bulunmuştur. Aktivasyon fonksiyonu ise, en sık kullanılan ve tercih edilen fonksiyonlar parametre setine verilmiştir[85]. Daha sonra, Cross-validation metodunun hiper parametre optimizasyon tekniği olan GridSearchCV kullanılarak örnek parametre seti için en doğru modeli bulacak parametreler, best_params_ yöntemi ile elde edilmiştir. Son olarak, tüm değerler için bulunan optimum değerler ile nihai doğrulanmış model elde edilmektedir.

2**.Adım:** Doğrulanmış model tahmin sonuçları;

Tablo 4.12: YSA ölçüm metrik sonuçları - 2

Ölçüm Metrikleri	
MSE	39.321943328021554
RMSE	6.270721117066326
MAE	1.7579658612687985
R2	0.93266540116882

Tablo 4.12’de YSA doğrulanmış model sonuçları, RMSE değerine göre fon fiyat tahmini için birim başına düşen hata payı oranının artı eksi(+/-) 6.2 olarak, bir sapma değerinin olduğu görülmektedir. YSA model sonuçları incelendiğinde, PLSR, RR ve SVR model geliştirme çalışmalarının sonuçlarına göre çok daha iyi değerler elde ettiği gözlemlenmektedir. Tahmin başarısının, RMSE değerine göre en başarılı model olduğu saptanmaktadır.

4.5. DOĞRULANMIŞ MODEL SONUÇ KARŞILAŞTIRMALARI

Model çalışmasının bu son bölümünde, 4 algoritmanın doğrulanmış model sonuçları karşılaştırma tablosu yapılarak değerlendirilecektir. Ayrıca, her bir model için tahmin edilen değerle gerçek değer arasındaki uzaklık ilişkisini de gösteren grafiklere yer verilerek bölüm sonlandırılacaktır.

Tablo 4.13: Model karşılaştırma ölçüm metrik sonuçları

Ölçüm Metrikleri	PLSR	RR	SVR	YSA
MSE	497.0027614702058	497.0262344776716	542.8368894117968	39.321943328021554
RMSE	22.29355874395575	22.294085190419267	23.298860259931104	6.270721117066326
MAE	6.705839394559914	6.69722792342646	3.711111249858252	1.7579658612687985
R2	0.14893622417341656	0.14889602917267908	0.07045020958408388	0.93266540116882

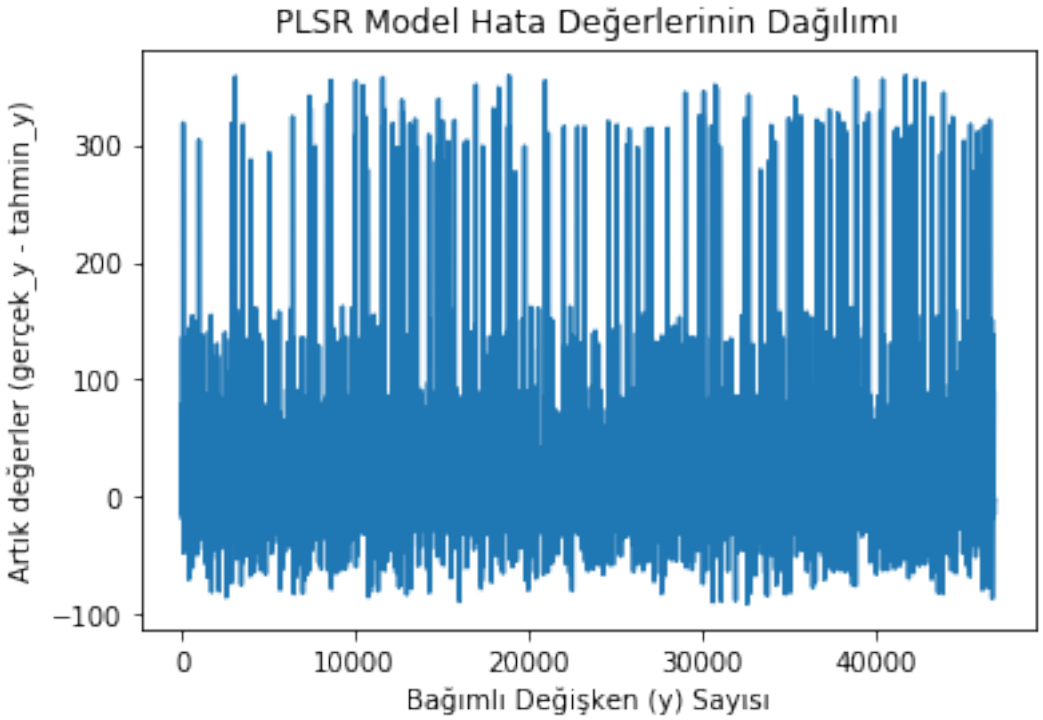
Yukarıdaki tablo 4.13'te model geliştirme çalışmaları yapılmış olan 4 algoritmanın doğrulanmış model ölçüm metrik sonuçları bulunmaktadır. Bu sonuçlar her değerlendirme metriği için ayrı olarak incelenecektir.

- **MSE:** Basitçe, gerçek değer ile tahmin edilen değer arasındaki çizilecek olan regresyon eğrisinin ne kadar gerçek değerlere yakın olduğunu söylemektedir. Kısaca, tahmin fonksiyonun performansını ölçer. Karşılaştırma tablosuna baktığımızda, YSA modelinin en iyi tahmin başarısına sahip olduğu görülmektedir. Daha sonra, PLSR ve RR modellerinin birbirlerine çok yakın değer tahminleri yaptıkları, SVR modeline göre daha iyi oldukları ama genel olarak 3 modelin de çok düşük performansa sahip olduğu tespit edilmiştir.
- **RMSE:** MSE değerinin karekökü alınmış formatıdır. RMSE'nin MSE metoduna göre daha avantajlı olduğu model ölçüm metrikleri bölümünde anlatılmıştır. Kısaca, RMSE tahmin edilen değerlerin, gerçek değere olan uzaklığının (+/-)

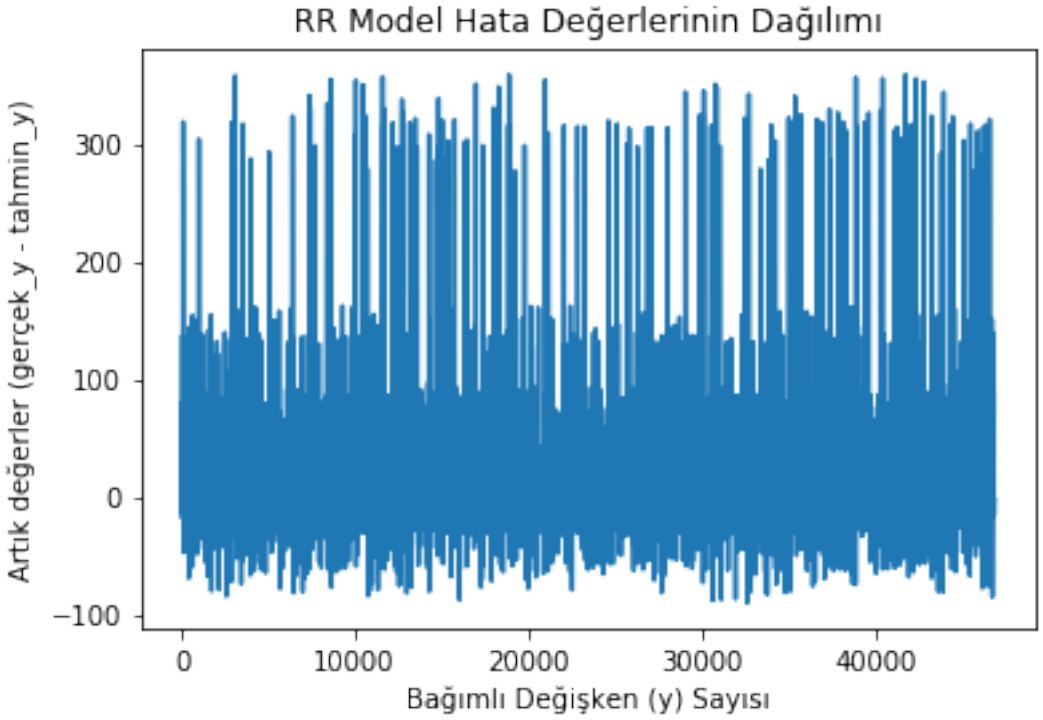
birim pay başına düşen değeridir. Buna göre, YSA modelinin diğer üç modele göre çok başarılı olduğu gözlemlenmektedir. PLSR ve RR model değerlerinin birbirine çok yakın olduğu ama düşük performansa sahip oldukları incelenmiştir. Ayırca, en düşük performansın SVR modeline ait olduğu görülmüştür.

- **MAE:** Kısaca, iki sürekli değişken arasındaki fark değerinin performans ölçüsüdür. Model ölçüm metrikleri bölümünde regresyon problemlerinde sıkça kullanılan ve çıktılarının yorumlanması kolay olan bir yöntem olduğundan bahsedilmiştir. Buradan hareketle model sonuçlarına göre, YSA modelinin yüksek bir başarı performansına sahip olduğu incelenmektedir. SVR de, diğer ölçüm metrik sonuçlarına göre çok iyi bir sonuca ulaşmıştır. Bunun nedeni, negatif yönelimli tahmin puanlarında MAE yönteminin daha iyi performans gösterdiği bilinmekte ve model tahmin değerlerimizin daha çok negatif yönelimli puanlara sahip olmasına bağlı olduğu değerlendirilmektedir. Son olarak, PLSR ve RR modellerinin diğer modellere göre çok düşük performansa sahip oldukları gözlemlenmiştir.
- **R2:** Model için veri uyumluluğu skoru olarak ifade edilebilir. Alacağı en iyi skor değerinin 1 olduğu belirtilmiştir. Model ölçüm metrikleri bölümünde ayrıntılı anlatımı yapılmıştır. Tablo 4.13'teki çıktılara baktığımızda, YSA modelin çok yüksek bir performansa sahip olduğu ve başarı skoru olarak % 90'nın üstünde olduğu görülmektedir. PLSR ve RR'nin birbirine çok yakın sonuçlara sahip olduğu ve başarı skorlarının düşük olduğu belirtilmektedir. Ayrıca, burada SVR modelinin diğer üç modelin sonuçlarına göre en kötü skor değerini elde ettiği incelenmiştir.

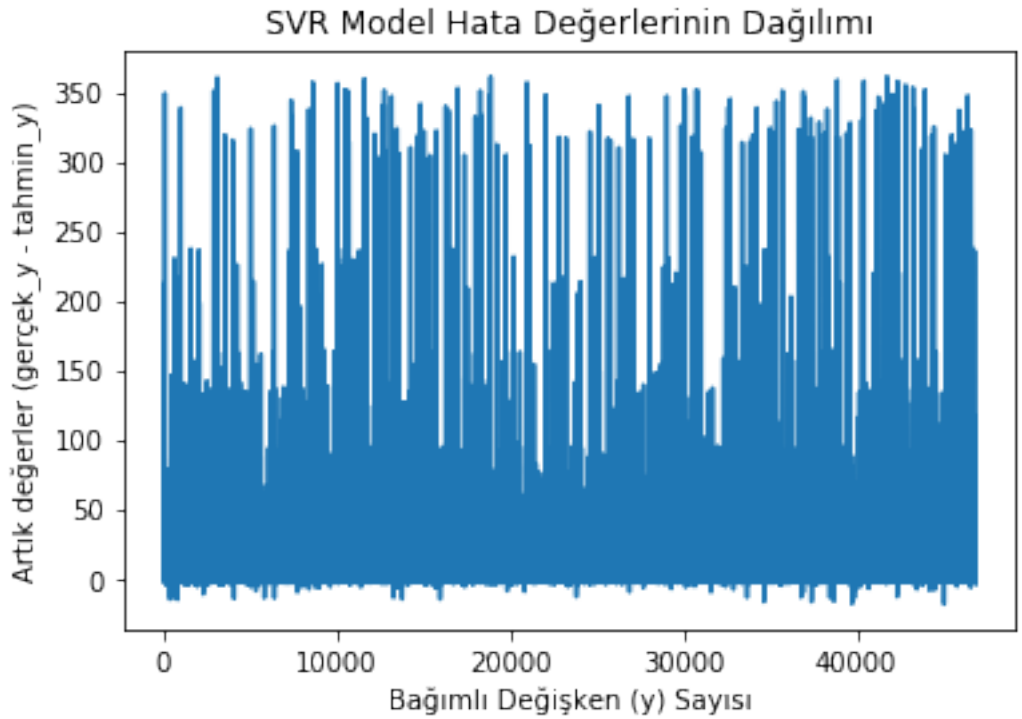
Doğrulanmış model sonuçlarının, ölçüm metriklerine göre değerlendirilme adımları yukarıda yapıldıktan sonra, tahmin edilen değerlerin gerçek değerlerden ne kadar uzakta olduğunu ifade eden artık verileri görselleştirme işlemleri yapılacaktır. Modellerin görselleştirme grafikleri tahmin edilen değer verilerinin dağılımını daha iyi okumamızı sağlayacaktır.



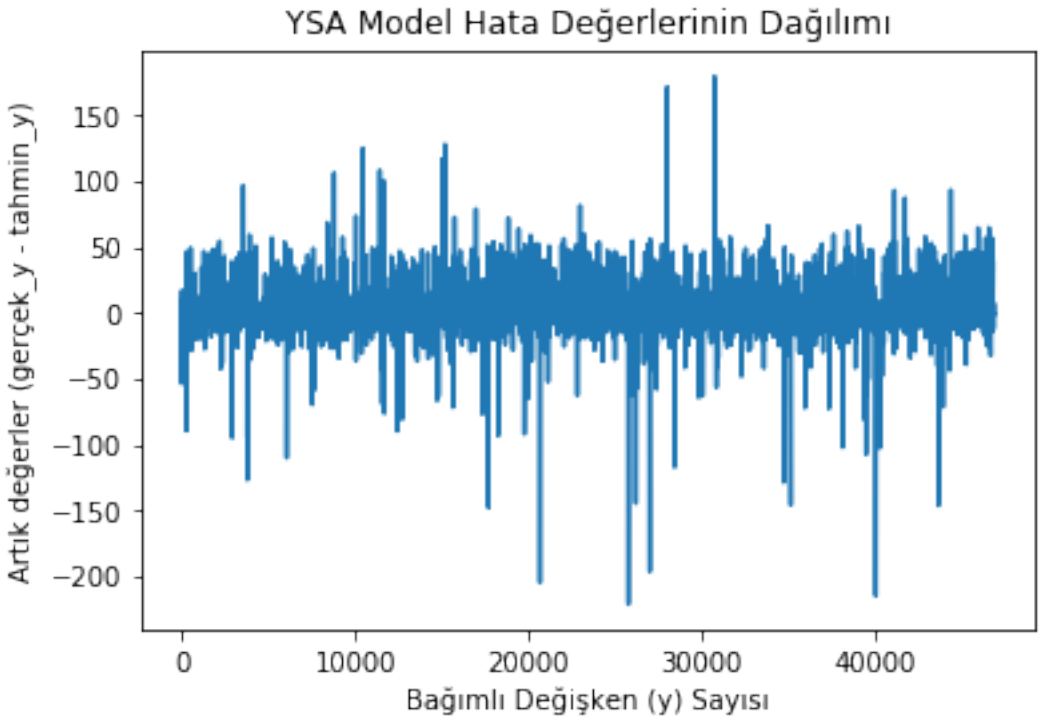
Şekil 4.23: PLSR model artık veri grafiği



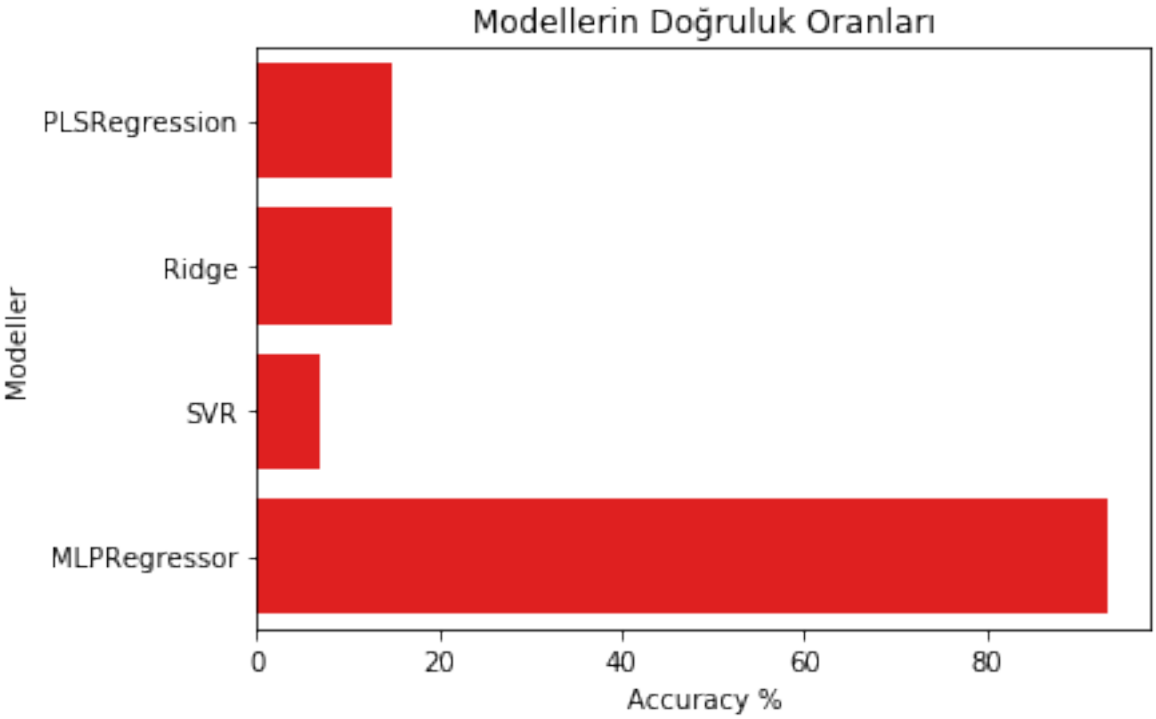
Şekil 4.24: RR model artık veri grafiği



Şekil 4.25: SVR model artık veri grafiği



Şekil 4.26: YSA model artık veri grafiği



Şekil 4.27: 4 modelin doğruluk oran grafiği

Yukarıdaki 4 algoritmayla geliştirilen model hata dağılım grafiklerine baktığımızda, YSA model tahmin veri grafik dağılımında, tahmin edilen değerlerin yayılımının dengeli olduğu görülmektedir. Ayrıca, YSA modelinin ölçüm metrik sonuçlarına göre de en yüksek tahmin başarısına sahip olduğu gözlemlenmiştir. Sonuç olarak, fonların fiyat tahmin için geliştirilen model çalışmasında, YSA modeli tahmin edilen fon fiyat değerinin birim pay başına düşen hata oranı, RMSE değerine göre (+/-) 6.2 olarak bulunmuştur. Model başarı yüzdesi ise, R2 ölçüm metrik sonucuna bakılarak %90 üzerinde bir tahmin başarısı olarak elde edilmiştir. Son olarak, 4 modelin tahmin başarısı R2 ölçüm metriğine göre yapılmış olup, şekil 4.27'de de model karşılaştırması yapılmıştır. Ancak nihai model karşılaştırılmasının yüzdelik sonucu R2 ile değerlendirilemez. Diğer ölçüm metrik sonuçları da göz önüne alındığında regresyon problemleri için PLSR, RR ve SVR modellerinin kullanılabilirliklerini yitirmedikleri tespit edilmiştir. Daha doğru sonuçlar için PLSR, RR ve SVR modellerin optimizasyon bölümlerinde modeller daha geniş deneysel değer aralıkları ile test edilebilir ve veri setlerinden korelasyon ilişkisi zayıf olan değişkenler çıkarılarak modeller tekrar değerlendirilebilir.

5. SONUÇ VE GELECEK ÇALIŞMALAR

Fonlar, finans sektöründe önemli bir yatırım aracıdır. Yatırımcılar için kazançlı yatırımlar yapacakları fonların seçilmesi amacıyla yatırımcı tercihlerinin yönetiminde, yönlendirilmesinde birçok aracı kurum ve kuruluş bulunmaktadır. Temel olarak, yatırımcıların kar sağlayacak kazançlar elde edebilmesi için birden fazla kurum, kuruluş ve finans araştırmacıları fon alım satımında danışmanlık yapmaktadır. Doğru tercihlerin yapılabilmesi içinse piyasa araştırmaları, fonların fiyat dalgalanmaları ve birden fazla parametreyi hesaba katarak öngörüde bulunmaya çalışılmaktadır. Bu süreçlerin hepsi geleneksel anlamda yapılan araştırmalardan elde edilen verilerden hareketle ve fon danışmanlarının tecrübeleri, öngörüleri ölçüsünde yatırımcıya yönelik bir öneriye dönüşmektedir. Fakat, her ne kadar bu öngörüler belli araştırmalar kapsamında yapılsa da, günümüz teknolojik gelişmelerinin geldiği son noktada sunulan imkanlar ölçüsünde çok düşük geçerliliğe sahip öneriler olarak kalmaktadırlar. Artık teknolojik imkanlar ve yapay zeka çalışmaları ile birlikte bu tür manuel yapılan işlemler, akıllı sistemlere dönüştürülerek üst düzey ve çok fonksiyonlu hesaplama işlemleri rahatlıkla yapılabilmektedir. Bu amaçla bu tez çalışmasında, makine öğrenme algoritmaları kullanılarak fonların fiyat tahmininin rahatlıkla, hızlı ve yüksek güvenilirlikte sağlanabilmesi için model geliştirilmesi yapılmıştır.

Fon fiyat tahmin sistemi için geliştirilmesi yapılan ve model çalışmasında kullanılan veri setinin gerçek veri değerlerinden seçilmiş olması, yapılan modelin tahmin başarısında piyasadaki fon bilgilerine yakın değerleri yakalayabilmesi, model tutarlığı ve güvenirliliği sağlaması açısından oldukça önmelidir ve bu durum çalışmayı da benzerlerinden ayırmaktadır. Bundan dolayı veri seti, Takas İstanbul(İstanbul Takas ve Saklama Bankası A.Ş.-Takasbank)'un platform ve veri kaynağı sağlayıcılığı yaptığı Türkiye Elektronik Fon Dağıtım Platformu (TEFAS) web sitesi üzerinden erişime açık olan fon bilgilerinden elde edilmiştir. Her bir fon için 02.01.2019 – 31.12.2019 tarihleri arasında ki tipi, türü, toplam değeri, tedavüldeki pay sayısı, pay alan kişi sayısı, fiyatı ve menkul (26 çeşit) oranları ile

fonların üzerinde önemli bir etkiye sahip olan her bir tarih için faiz bilgisi, altın fiyatı ve dolar fiyatı baz alınarak veri seti hazırlanmıştır. Veri kümesi, toplam 187,438 veri ve 37 kolondan oluşturulmuştur. Veri seti, tutarlı ve temizlenmiş bir veri sağlaması için ön işleme adımlardan geçirilerek deneysel çalışmalarda kullanılmak üzere, eğtim ve test setine ayrılmıştır.

Model geliştirme çalışmaları, Kısmi En Küçük Kareler Regresyonu (PLSR), Ridge Regresyonu (RR), Destek Vektör Regresyonu (SVR) ve Yapay Sinir Ağları(YSA) algoritmalarıyla yapılmıştır. Her bir algoritmanın bölüm başlığı altında ayrıntılı olarak model geliştirme süreçleri anlatılmıştır. Geliştirilen model başarı değerlendirmeleri; Hata Karelerinin Ortalaması (MSE), Hata Kare Ortalamasının Karekökü (RMSE), Ortalama Mutlak Hata (MAE) ve R Kare Oranı (R2) ölçüm metriklerine göre yapılmıştır. Model geliştirme çalışmaları yapılmış olan 4 algoritmanın doğrulanmış model ölçüm metrik değerleri; MSE, RMSE, MAE ve R2 sonuçlarına göre kurulan modellerden PLSR, RR ve SVR algoritmalarının kabul edilebilirlik oranlarının çok düşük olduğu görülmüştür. YSA ile elde edilen tahmin değerlerinin başarı oranının ise, model önerisi için kabul edilebilir ölçekte olduğu gözlemlenmiştir. Ayrıca, 4 algoritmayla tahmin edilen değerlerin gerçek değerlerden ne kadar uzakta olduğunu ifade eden artık verileri görselleştirme işlemleri şekil 4.23, 4.24, 4.25 ve 4.26'da yapılmıştır. Bu veri, artık görsel grafiklerine göre modellerin tahmin değerlerinin tutarlılık yapısını da, daha anlaşılır kılmaktadır. Artık dağılım grafiklerine bakıldığında da, YSA model tahmin veri grafik dağılımının, tahmin edilen değerlerin yayılımının diğer 3 algoritma grafiklerine göre daha dengeli olduğu görülmektedir. Bu değerlendirmelerden hareketle, fonların fiyat tahmini çalışmasında YSA ile geliştirilen model tercih edilmiştir.

Sonuç olarak, fonların fiyat tahmini için geliştirilen model çalışmasında, YSA modelinin tahmin edilen fon fiyat değerinin birim pay başına düşen hata oranı, RMSE değerine göre (+/-) 6.2 olarak bulunmuştur. Model başarı yüzdesi ise, R2 ölçüm metrik sonucuna bakılarak %90 üzerinde bir tahmin başarısı olarak elde edilmiştir. Son olarak, 4 modelin tahmin başarısı R2 ölçüm metriğine göre yapılmış

olup, şekil 4.27'de de model karşılaştırması yapılmıştır. Ancak nihai model karşılaştırılmasının yüzdelik sonucu R2 ile değerlendirilemez. Diğer ölçüm metrik sonuçları da göz önüne alındığında regresyon problemleri için PLSR, RR ve SVR modellerinin kullanılabilirliklerini yitirmedikleri tespit edilmiştir. Daha doğru sonuçlar için PLSR, RR ve SVR modellerinin optimizasyon bölümlerinde daha geniş deneysel değer aralıkları ile test edilebilir ve veri setlerinden korelasyon ilişkisi zayıf olan değişkenler çıkarılarak modeller tekrar değerlendirilebilmelidir.

Çalışmayı diğer araştırmalardan ayıran temel özellik, kullanılan veri setinin Takasbanktan alınan gerçek verilerden oluşması ve elde edilen sonuçların gerçek hayatta karşılaşılan problemlere çözüm olabileceği zemini hazırlamış olmasıdır. Bu bağlamda akademi ve sanayi iş birliğinin ne kadar elzem olduğu görülmekte ve bu çalışmanın benzer çalışmaların oluşturulmasına örnek teşkil etmesi umulmaktadır. Ayrıca, erişime açık olan TEFAS verileri ilk defa doğrudan kullanılarak bir araştırmanın veri setini oluşturmuştur. Bu çalışmayla TEFAS verilerinin kullanımı yaygınlaşarak veri ekosistemine katkısı olacağına inanılmaktadır. Ek olarak, Takasbank'ta gerçekleşecek birçok projeye de örnek kaynak olacağı düşünülmektedir.

TEFAS platformu için fonların fiyat tahminini gerçekleştirmek amacıyla yapılan bu model çalışması birinci adım sayılarak tahmin fonksiyonu oluşturulmuştur. Gelecek çalışmalar da, fon tipleri ve fon türleri bazında kırılımlar gerçekleştirilerek model çalışması daha özel hale getirilebilir. Hatta fon fiyat tahmini üzerinde ki etkisi dikkate alınarak gündelik siyasi olayların belli bir parametre setine göre ayarlanması yapılarak hesaba aktarımı gerçekleştirilebilir. Bu durumların risk getirilerini minimize ederek yatırımcılar için fon kazançlarını nasıl daha iyi optimize edeceği konusunda yeni çalışmalara ilham olacağı umulmaktadır.

Her araştırma da olduğu gibi bu araştırmanın da bazı sınırlılıkları mevcuttur. Bunlardan ilki, Takasbank ve benzeri köklü kuruluşların veri bilgilerinin tahmin edilenden fazla olması ve bu bilgilere erişimin belirli prosedürler kapsamında gerçekleşmesidir. Bu durum teknik anlamda verilere erişimi zorlaştırmış ve

çalışmanın zamansal sınırlılıklarından ötürü bir yıllık veri seti dikkate alınarak çalışma tamamlanmıştır. En ayırt edici ve bu dönem de yazılan tüm tezlerin belirgin kısıtlayıcılıklarından bir diğeri olan pandemi süreci, erişilebilir laboratuvar imkanını yok ettiği için sahip olunan şahsi bilgisayarların yüksek test setleri ile optimizasyona izin verdiği ölçüde çıktılar hazırlanabilmiştir. Yaşadığımız bu yorucu süreç atlatıldıktan sonra, yeni çalışmalarla teknik özelikleri daha kapsamlı bilgisayarlardan modeller tekrar çalıştırılarak, hiper parametre setleri genişletilebilir.

Son olarak bu çalışma da, makine öğrenme teknikleri kullanarak fonların fiyat tahminini gerçekleştirecek model önerisinin amaç ve önemine dair genel bir perspektif sunulmuştur. Temel olarak tahmin çalışmalarında, makine öğrenme teknikilerinin yaygın olarak kullanımlarından bahsedilerek, finans sektöründeki uygulamalarda kullanılmasının önemi vurgulanmıştır. Model geliştirme için kullanılacak olan TEFAS veri setinin, bu tez çalışması için önemi ve diğer çalışmalardan ayrıt edici özelliklerinden bahsedilmiştir. Sonuç olarak araştırmanın en önemli bulgularından biri olan fon fiyat tahmini için model geliştirme de kullanılan dört algoritmadan en optimize sonuç veren algoritmanın YSA olduğu tespit edilmiştir.

KAYNAKÇA

1. https://www.spk.gov.tr/Sayfa/Dosya/922
2. https://www.spk.gov.tr/Duyuru/Dosya/20150107/1, https://www.tefas.gov.tr/
3. Ekonometride Yeni Bir Ufuk: Büyük Veri ve Makine Öğrenmesi (Social Sciences Research Journal, Volume 7, Issue 2, 41-53 (June 2018), ISSN: 2147-5237)
4. **Alpaydın, E.** (2004). Introduction to machine learning (Adaptive Computation and Machine Learning). Cambridge: The MIT press, 11-26.
5. **Kalaycı, S.** (2018). Makine öğrenmesi yöntemleri ile kredi risk analizi, Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
6. Clustering Electricity Market Participants, Turk J Elec Eng & Comp Sci, Tübitak.(Yayın aşamasında)
7. A decision support system for predicting students performance, Themes in Science & Technology Education, 9(1), 43-57, 2016.
8. **Karabıyık, M.A.** (2018). Akademik yayınlar için makine öğrenmesi tabanlı arama motoru tasarlanması ve uygulanması, Yüksek Lisans Tezi, Süleyman Demirel Üniversitesi, Fen Bilimleri Enstitüsü, Isparta.
9. Machine learning applications in cancer prognosis and prediction, Computational and Structural Biotechnology Journal 13(2015) 8-17.
10. **Çakmak, I.** (2017). Makine öğrenmesi yöntemleriyle tümör kontrol olasılığının hesaplanması, Yüksek Lisans Tezi, Karadeniz Teknik Üniversitesi, Sağlık Bilimler Enstitüsü, Trabzon.
11. **Saçlı, B.** (2018). Machine Learning Aided Kidney Stone Classification With Electromagnetic Properties, Istanbul Technical University, Department of Electronics and Communication Engineering, Istanbul.
12. Türkiye'de makine öğrenmesi ile ilgili yapılan tez çalışmalarına yönelik bir literatür taraması, UEMK 2019 Proceedings Book 24/25 October 2019 Gaziantep University, Gaziantep.

13. **Tektaş, A., Karataş, A.** (2004). Yapay Sinir Ağları ve Finans Alanına Uygulanması:Hisse Senedi Fiyat Tahminlemesi, Atatürk Üniversitesi İktisadi ve İdari Bilimler Dergisi..

14. **Karaatlı, M., Güngör, İ., Demir, Y., Kalaycı, Ş.** (2005). Hisse Senedi Fiyat Hareketlerinin Yapay Sinir Ağları Yönetimi ile Tahmin Edilmesi. Yönetim ve Ekonomi Araştırmaları Dergisi; 3(3): 48-38.

15. **Gür, N.** (2009). Hisse Senedi Fiyat Hareketlerinin Tahmini için Bir Yapay Sinir Ağı Modeli Önerisi, Kocaeli Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.

16. Price Prediction of Share Market using Artificial Neural Network (ANN), International Journal of Computer Applications (0975 – 8887) Volume 22– No.2, May 2011.

17. Michael Hagenau Michael Liebmann Dirk Neumann, *Automated news reading: Stock price prediction based on fi nancialnews using context-capturing features*, Decision Support Systems Volume 55, Issue 3, June 2013, Pages 685-697.

18. Osman Hegazy , Omar S. Soliman , Mustafa Abdul Salam., *A Machine Learning Model for Stock Market Prediction*, International Journal of Computer Science and Telecommunications [Volume 4, Issue 12, December 2013].

19. Carson Kai-Sang Leung, Richard Kyle MacKinnon ,Yang Wang, *A Machine Learning Approach for Stock Price Prediction*, IDEAS '14: Proceedings of the 18th International Database Engineering & Applications Symposium,July 2014.

20. Yapay Sinir Ağlarıyla Hisse Senedi Fiyatları ve Yönlerinin Tahmini, Eskişehir Osmangazi Üniversitesi, İİBF Dergisi, Aralık 2015, 10(3),177-194.

21. **Yüksel, R., Akkoç, S.** (2016). Altın Fiyatlarının Yapay Sinir Ağları ile Tahmini ve Bir Uygulama, Doğuş Üniversitesi Dergisi, 17 (1), 39-50.

22. **Addai, S.** (2016). Financial Forecasting Using Machine Learning, Masters Degree, African Institute for Mathematical Science, South Africa.

23. **Özçalıcı, M.** (2016). Yapay Sinir Ağları ile Çok Aşamalı Fiyat Tahmini: BIST30 Senetleri Üzerine Bir Araştırma, Dokuz Eylül Üniversitesi, İktisadi ve İdari Bilimler Fakültesi Dergisi, Cilt:31, Sayı:2, ss. 209-227.

24. **McNally, S.** (2016). Predicting the price of Bitcoin using Machine Learning, School of Computing National College of Ireland, MSc Reseach Project in Data Analytics

25. Sakız, B., Gencer, A.H, Forecasting the Bitcoin Price via Artificial Neural Networks, International Conference of Eurasian Economies 2018, pp.438-444, Tashkent, UZBEKISTAN.

26. **Aktepe, Ç.** (2018). Algorithmic trading on cryptocurrency markets using machine learning techniques, M.Sc. Thesis, Boğaziçi University, Department of Industrial Engineering, İstanbul.

27. **Kanmaz, M.** (2018). The effect of financial news on bist stock prices: A machine learning approach, M.Sc. Thesis, Middle East Technical University, Department of Economics, Ankara.

28. **Demirel, U.** (2019). Hisse senedi fiyatlarının makine öğrenmesi yöntemleri ve derin öğrenme algoritmaları ile tahmini, Yüksek Lisans Tezi, Gümüşhane Üniversitesi, Sosyal Bilimler Enstitüsü, Gümüşhane.

29. Pabuçcu,H., Borsa Endeksi Hareketlerinin Makine Öğrenme Algoritmaları ile Tahmini. Uluslararası İktisadi ve İdari İncelemeler Dergisi. 2019; (23): 190-179.

30. Akşehir, D.Z, Kılıç, E., Makine Öğrenme Teknikleri ile Banka Hisse Senetlerinin Fiyat Tahmini. Türkiye Bilişim Vakfı Bilgisayar Bilimleri ve Mühendisliği Dergisi. 2019; 12(2): 39-30.

31. Janik L. J., Skjemstad J. O., Shepherd K. D., Spouncer L. R. (2007) The prediction of soil carbon fractions using mid-infrared-partial least square analysis. Australian Journal of Soil Research 45, 73-81.

32. **Polat, E., Günay, S.** (2009). Kısmi En Küçük Kareler ve Bir Uygulama. Ondokuz Mayıs Üniversitesi, Fen Edebiyat Fakültesi, İstatistik Bölümü, VI. İSTATİSTİK GÜNLERİ SEMPOZYUMU BİLDİRİLER KİTABI. S,438.

33. **Taşkın, V., Doğan, B., Ölmez, T.** (2013). Prostate Cancer Classification from Mass Spectrometry Data by Using Wavelet Analysis and Kernel Partial Least Squares Algorithm, International Journal of Bioscience, Biochemistry and Bioinformatics, Vol. 3, No. 2.

34. **Serrano-Cinca, C., Guti´errez-Nieto, B.** (2013). Partial Least Square Discriminant Analysis for bankruptcy prediction, Decision Support System,Volume 54, Issue 3,Pages 1245-1255.

35. **Büykuysal, M. Ç.** (2010). Ridge regresyon analizi ve bir uygulama. Yayınlanmamış yüksek lisans tezi. Uludağ Üniversitesi Sağlık Bilimleri Enstitüsü, Bursa.

36. **Çekerol, G. , Nalçakan, M .** (2011). Lojistik Sektörü İçerisinde Türkiye Demiryolu Yurtiçi Yük Taşıma Talebinin Ridge Regresyonla Analizi. Marmara Üniversitesi İktisadi ve İdari Bilimler Dergisi , 31 (2) , 321-344.

37. **Küçük, A.** (2019). Doğrusal Regresyonda Ridge, Liu ve Lasso Tahmin Edicileri Üzerine Bir Çalışma, Yüksek Lisans, Hacettepe Üniversitesi, Fen Bilimleri Enstitüsü, Ankara.

38. **Ayan, B., Kuyumcu, B., Ceylan, B.** (2019). Twitter Üzerindeki İslamofobik Twitlerin Duygu Analizi ile Tespiti. Gazi Üniversitesi Fen Bilimleri Dergisi Part C: Tasarım ve Teknoloji, 7 (2) , 495-502.

39. **Ekinci, E.M.** (2017). Destek Vektör Regresyon ile Hava Kirliliği Tahmini, Yüksek Lisans Tezi, Fen Bilimleri Enstitüsü, Eskişehir Osmangazi Üniversitesi, Eskişehir.

40. **Uçak, K.** (2012). Destek Vektör Regresyonu İle Pıd Kontrolör Tasarımı, Yüksek Lisans Tezi, Fen Bilimleri Enstitüsü, İstanbul Teknik Üniversitesi, İstanbul.

41. **Orhunbilge, N.** (2002), Uygulamalı Regresyon ve Korelasyon Analizi, İstanbul, İ.Ü. İşletme Fakültesi.

42. **GÖK, M.** (2017). MAKİNE ÖĞRENMESİ YÖNTEMLERİ İLE AKADEMİK BAŞARININ TAHMİN EDİLMESİ. Gazi Üniversitesi Fen Bilimleri Dergisi Part C: Tasarım ve Teknoloji. 5(3): 148-139.

43. **Abdi, H.** (2003). Partial Least Squares (PLS) Regression, Lewis-Beck M., Bryman, A., Futing T. (Eds.) Encyclopedia of Social Sciences Research Methods. Thousand Oaks (CA): Sage.

44. **Tobias, R.D.** (1995). An Introduction to Partial Least Squares Regression. UGI Proceedings, Orlando 2-5 April, pp. 1-8.

45. **Höskuldson, A.** (1988). PLS Regression Methods. Journal of Chemometrics, 2: 211-228.

46. **Marten, H. ve Naes, T**. (1989). Multivariate Calibration. John Wiley & Sons.

47. **Wu, J.** (2015). Research on Several Problems in Partial Least Squares Regression Analysis. The Open Electrical & Electronic Engineering Journal, 8, 754-758

48. **Bulut, E., Alma, G., Ö.** (2011). Kısmi En Küçük Kareler Regresyonu Yardımıyla Optimum Bileşen Sayısını Seçmede Model Seçme Kriterlerinin Performans Karşılaştırılması. İstanbul Üniversitesi, İktisat Fakültesi, Ekonometri ve İstatistik Dergisi, Sayı:15 , 38-52.

49. **Rosipal, R., Trejo, L. J.** (2001). Kernel Partial Least Squares Regression in Reproducing Kernel Hilbert Space. Journal of Machine Learning Research 2 , 97-123.

50. **Bulut, E., Alın, A.** (2009). Kısmi En Küçük Kareler Regresyon Yöntemi Algoritmalarından Nipals ve PLS - Kernel Algoritmalarının Karşılaştırılması ve Bir Uygulama. Dokuz Eylül Üniversitesi, İktisadi ve İdari Bilimler Fakültesi Dergisi, Cilt:24, Sayı:2, ss.127-138.

51. **Ümit, Ö.A., Bulut, E.** (2013). TÜRKİYE'DE İŞSİZLİĞİ ETKİLEYEN FAKTÖRLERİN KISMİ EN KÜÇÜK KARELER REGRESYON YÖNTEMİ İLE ANALİZİ: 2005-2010 DÖNEMİ. Dumlupınar Üniversitesi Sosyal Bilimler Dergisi. (37): -.

52. **Demirci, A.M.** (2014). Ridge Regresyonda Sapma Parametresi Olan k'nın Belirlenmesinde Genetik Algoritma Yaklaşımı, Yüksek lisans Tezi, Ondokuz Mayıs Üniversitesi, Fen Bilimleri Enstitüsü, Samsun.

53. **Bağcı, İ.** (2017). Kısmi En Küçük Kareler Yönteminin Simülasyon Verileri ile Diğer Yöntemlerle Karşılaştırılması, Yüksek Lisans Tezi, Muğla Sıtkı Koçman Üniversitesi, Fen Bilimleri Enstitüsü, Muğla.

54. **Polat, E.** (2009). Kısmi En Küçük Kareler Regresyonu, Yüksek Lisans Tezi, Hacettepe Üniversitesi, Fen Bilimleri Enstitüsü, Ankara.

55. **Kim, K.** (2019). Ridge Regression for Better Usage. Erişim Tarihi:…

56. **Şahingöz, T.M.** (2019). Regresyon Analizinde Yanlı Tahmin Yöntemleri, Yüksek Lisans Tezi, Muğla Sıtkı Koçman Üniversitesi, Fen Bilimleri Enstitüsü, Muğla.

57. **Cule, E. &, Lorio, D.M.** (2013). Ridge Regression in Prediction Problems: Automatic Choice of the Ridge Parameter. *Genetic Epidemiology, 37*, 704 - 714.

58. **Wieringen, W.N.** (2015). Lecture notes on ridge regression. *arXiv: Methodology*.

59. **Arat, M.M.** (2014). Destek Vektör Makineleri Üzerine Bir Çalışma, Yüksek Lisans Tezi, Fen Bilimleri Enstitüsü , Hacettepe Üniversitesi, Ankara.

60. **Tolun, S.** (2008). Destek Vektör Makineleri: Banka Başarısızlığının Tahmini Üzerine Bir Uygulama, Doktora Tezi, Sosyal Bilimler Enstitüsü, İstanbul Üniversitesi, İstanbul.

61. **Smola, A.J., Scholkopf, B.** (2004). A tutorial on support vector regression, Statistics and Computing 14, 199–222.

62. **Trafalis, T.B, Ince, H.** (2000). Support Vector Machine for Regression and Applications to Financial Forecasting. In: IJCNN 2000: Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks: Volume 6 edited by Shun-Ichi Amari, et al., içinde 6348, IEEE Computer Society.

63. **Chanklan, R., Kaoungku, N., Suksut, K., Kerdprasop, K., & Kerdprasop, N.** (2018). Runoff Prediction with a Combined Artificial Neural Network and Support Vector Regression. *International Journal of Machine Learning and Computing, 8*, 39-43.

64. **Chen, Y., Tan, H.** (2017). Short-term Prediction of Electric Demand in Building Sector Via Hybrid Support Vector Regression, Applied Energy, Volume 204, Pages 1363-1374.

65. **McCulloch, W. S., Pitts, W.** (1943). A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics, 5(4), 115-133.

66. **Polat, E.** (2019). Türkiye'nin Aylık Elektrik Tüketiminin Yapay Sinir Ağlarıyla Tahmini, Yüksek Lisans Tezi, Fen Bilimleri Enstitüsü, Yıldız Teknik Üniversitesi, İstanbul.

67. **Akpınar, B.** (2019). Görüntü Sınıflandırma için Derin Öğrenme ile Bayesçi Derin Öğrenme Yöntemlerinin Karşılaştırılması, Yüksek Lisans Tezi, Fen Bilimleri Enstitüsü, Afyon Kocatepe Üniversitesi, Afyonkarahisar.

68. **Adıyaman, F.** (2007). Talep Tahmininde Yapay Sinir Ağlarının Kullanılması, Yüksek Lisans Tezi, Fen Bilimleri Enstitüsü, İstanbul Teknik Üniversitesi, İstanbul.

69. **Ataseven, B.** (2013) Yapay Sinir Ağları ile Öngörü Modellemesi. Dergipark, s.101-115, İstanbul.

70. **Akkurt, A.** (2005). Yapay Sinir Ağları Ve Türkiye Elektrik Tüketimi Tahmin Modeli, Yüksek Lisans Tezi, Fen Bilimleri Enstitüsü, İstanbul Teknik Üniversitesi, İstanbul.

71. **Akdoğan, E.** Ders Notları, Mekatronik Mühendisliği Uygulamalarında Yapay Zekâ, Yapay Sinir Ağları,[Alıntı Tarihi: 22.05.2020]. http://ytubiomechatronics.com/wp-content/uploads/2017/10/YSA.pdf

72. **Kuş, Z.** (2019). Mikrokanonikal Optimizasyon Algoritması ile Konvolüsyonel Sinir Ağlarında Hiper Parametrelerin Optimize Edilmesi, Yüksek Lisans Tezi, Fen Bilimleri Enstitüsü, Fatih Sultan Mehmet Vakıf Üniversitesi, İstanbul.

73. **Çayıroğlu, İ.** İleri Algoritma Analizi-5 Yapay Sinir Ağları. Karabük Üniversitesi Mühendislik Fakültesi. [Alıntı Tarihi: 23.05.2020]. http://www.ibrahimcayiroglu.com/dokumanlar/ilerialgoritmaanalizi/ilerialgoritmaanalizi-5.hafta-yapaysiniraglari.pdf

74. **Özkan, Y.** (2008). Veri Madenciliği Yöntemleri, İstanbul: Papatya Yayıncılık. (bak)

75. **Özata, M.** (2014). Regresyon, Korelasyon ve Faktör Analizi, Sosyal Hizmette İleri İstatistik Uygulamaları Dersi.(web erişim adresi)

76. https://scikit-learn.org/stable/modules/classes.html, Regression metrics

77. **Aydemir, E.** (2013). Kusurlu Ürünleri İçeren ekonomik Üretim Miktarı Modelinin Gri Sistem Teorisi Yaklaşımıyla Geliştirilmesi, Doktora Tezi, Fen Bilimleri Enstitüsü, Süleyman Demirel Üniversitesi, Isparta.

78. **Esposito Vinzi, V. and Russolillo, G.** (2013), Partial least squares algorithms and methods. WIREs Comp Stat, 5: 1-19. doi:10.1002/wics.1239

79. **Cawley, C.G., Talbot, C.L.N.** (2010). On Over-fitting in Model Selection and Subsequent Selection Bias in Performance Evaluation, Journal of Machine Learning Research, 11(70):2079−2107.

80. https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.RidgeCV.html, RidgeCV

81. **Lameski, P., Zdravevski, E., Mingov, R., & Kulakov, A.** (2015). SVM Parameter Tuning with Grid Search and Its Impact on Reduction of Model Over-fitting. *RSFDGrC*.

82. **Kor, K.** (2015). Penetration Rate Optimization With Support Vector Regression Method, Yüksek Lisans Tezi, Fen Bilimleri Enstitüsü, İstanbul Teknik Üniversitesi, İstanbul.

83. https://scikit-learn.org/stable/auto_examples/svm/plot_rbf_parameters.html, SVM hiper-parametre

84. **Schilling, N., Wistuba, M., Drumond, L., & Schmidt-Thieme, L.** (2015). Hyperparameter Optimization with Factorized Multilayer Perceptrons. *ECML/PKDD*.

85. **Weissbart,L., Picek, S., Batina,L.** (2019). On the Performance of Multilayer Perceptron in Profiling Side-channel Analysis, IACR Cryptol, ePrint Arch, Report 2019/1476