

T.C.
BAŞKENT ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
İŞLETME ANABİLİM DALI
İŞLETME DOKTORA PROGRAMI

YAPAY ZEKA YÖNTEMLERİ İLE FİNANSAL ZAMAN SERİLERİ
ÖNGÖRÜLERİ

DOKTORA TEZİ

HAZIRLAYAN

EFE ARDA

ANKARA – 2020

T.C.
BAŐKENT ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
İŐLETME ANABİLİM DALI
İŐLETME DOKTORA PROGRAMI

YAPAY ZEKA YÖNTEMLERİ İLE FİNANSAL ZAMAN SERİLERİ
ÖNGÖRÜLERİ

DOKTORA TEZİ

HAZIRLAYAN

EFE ARDA

TEZ DANIŐMANI

Prof. Dr. GÜRAY KÜÇÜKKOCAOĐLU

ANKARA – 2020

BAŞKENT ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
DOKTORA TEZ ÇALIŞMASI ORJİNALLİK RAPORU

Tarih: 07 / 09 / 2020

Öğrencinin Adı, Soyadı: Efe ARDA

Öğrencinin Numarası: 21510157

Anabilim Dalı: İşletme

Programı: İşletme Doktora

Danışmanın Unvanı/Adı, Soyadı: Prof. Dr. Güray Küçükkocaoğlu

Tez Başlığı: Yapay Zeka Yöntemleri ile Finansal Zaman Serileri Analizi

Yukarıda başlığı belirtilen Doktora tez çalışmamın; Giriş, Ana Bölümler ve Sonuç Bölümünden oluşan, toplam 102 sayfalık kısmına ilişkin, 19 / 07 / 2020 tarihinde şahsım/tez danışmanım tarafından Turnitin adlı intihal tespit programından aşağıda belirtilen filtrelemeler uygulanarak alınmış olan orijinallik raporuna göre, tezimin benzerlik oranı % 9'dur. Uygulanan filtrelemeler:

1. Kaynakça hariç
2. Alıntılar hariç
3. Beş (5) kelimedenden daha az örtüşme içeren metin kısımları hariç

“Başkent Üniversitesi Enstitüleri Tez Çalışması Orijinallik Raporu Alınması ve Kullanılması Usul ve Esaslarını” inceledim ve bu uygulama esaslarında belirtilen azami benzerlik oranlarına tez çalışmamın herhangi bir intihal içermediğini; aksinin tespit edileceği muhtemel durumda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve yukarıda vermiş olduğum bilgilerin doğru olduğunu beyan ederim.

Öğrenci İmzası:

ONAY

Tarih: 07 / 09 / 2020

Prof. Dr. Güray Küçükkocaoğlu

İmza:

TEŐEKKÜR

Doktora programında yardım ve desteklerini esirgemeyen, alıőmada kullanılan veri setini saęlayan Sayın Prof. Dr. Gray KKKOCAOęLU'na, tm eęitim hayatım boyunca arkamda olan annem İnci ARDA ve babam Servet Arda'ya sonsuz teőekkrlerimi sunarım.

ÖZET

Bilgisayar bilimlerinin sunduğu yöntemler, veriye erişme, veri işleme ve depolama yönünden finans alanındaki gereksinimleri karşılamada insanlara kıyasla kuvvetli avantajlara sahiptir. Bilgisayar bilimlerinin bir alt dalı olan yapay zeka sunduğu nöral ağ, genetik algoritmalar ve makine öğrenmesi gibi yöntemlerle finans alanında ortaya çıkan trend analizi, zaman serisi analizi, portföy yönetimi, dolandırıcılık tespiti, risk yönetimi ve hisse değeri öngörütleri gibi problemlere etkin çözümler sunmaktadır. Bu çok disiplinli tezin amacı, finansal zaman serilerini belirli yapay zeka programlarına öğreterek gelecek tahminleri üretmek ve elde edilen bu öngörütlerin performansını gerçekleştiren değerlerle kıyaslayarak incelemektir.

Tez çalışması kapsamında yedi farklı yapay zeka yöntemi programlanmıştır. Veri seti olarak 2014-2016 senelerini kapsayan 775 iş günü boyunca Borsa İstanbul'da işlem gören, işlem hacmi ve piyasa değeri en yüksek 30 şirketin hisse senedi kapanış fiyatları kullanılmıştır. Algoritmaların performanslarının öğrenme süreleri ile değişiklik göstereceği beklentisi sebebi ile öncelikle verilerin %80'ine tekabül eden 603 günlük veri öğrenme için kullanılmış, kalan %20'si olan 152 günlük veri yöntemlerin test edilmesinde kullanılmak üzere algoritmaların tahmininde kullanılmıştır. "Uzun vade tahminleri" olarak yapılan bu çalışmada "Hızlı Orman Yüzdeleri Dağılımı" algoritmasının en düşük hata oranına sahip olduğu tespit edilmiştir. Tezin bir sonraki adımında verilerin %90'ına tekabül eden 680 günlük veri öğrenme için kullanılmış, kalan %10'u olan 75 günlük kısmı test amacı ile kullanılmıştır. "Orta vade tahminleri" olarak yapılan bu çalışmada "Destekli Karar Ağacı Regresyonu" algoritmasının en düşük hata oranına sahip olduğu tespit edilmiştir. Son olarak verilerin %99'una tekabül eden 747 günlük veri öğrenme için kullanılmış, kalan %1'i olan 8 gün test amacı ile kullanılmıştır. "Kısa vade tahminleri" olarak yapılan bu çalışmada "Nöral Ağ Regresyonu" ve "Poisson Regresyonu" algoritmalarının en düşük hata oranlarına sahip olduğu tespit edilmiştir. Sonuç olarak yapay zeka yöntemlerinin finansal zaman serileri öngörütleri için etkili yöntemler sunduğu ortaya konmuştur.

Anahtar Kelimeler: Finansal Zaman Serileri, Yapay Zeka, Makine Öğrenmesi, Borsa Analizi, Regresyon

ABSTRACT

The methods provided by the field of computer science have considerable advantages in domains of reaching, storing and processing data to satisfy the requirements of finance field compared to humans. A subdomain of computer science; artificial intelligence provides methods such as neural networks, genetic algorithms and machine learning to find effective solutions to financial problems / goals such as trend prediction, portfolio management, fraud detection, risk management and stock prediction. This multidisciplinary thesis aims to teach certain artificial intelligence algorithms the financial time series data, provide future forecasts and compare these forecasts to original values to examine their performance.

Seven different artificial intelligence algorithms have been programmed for this thesis. A dataset of 775 business days between 2014-2016 consisting of closing prices of companies that have İstanbul Stock Exchanges 30 highest trading volume and market value are used. Due to the expectation of different algorithms to provide different performance depending on the number of learning / forecasting days, firstly %80 of the data equaling 603 days have been used for training, and the remaining %20 of the data equaling 152 days have been forecasted by the algorithms. Simulating “Long Term Forecasts”, “Fast Forest Quantile Regression” algorithm has shown the least error percentage. Secondly %90 of the data equaling 680 days have been used for training, and the remaining %10 of the data equaling 75 days have been forecasted by the algorithms. Simulating “Medium Term Forecasts”, “Boosted Decision Forest” algorithm has shown the least error percentage. Lastly %99 of the data equaling 747 days have been used for training, and the remaining %1 of the data equaling 8 days have been forecasted by the algorithms. Simulating “Short Term Forecasts”, “Poisson Regression” and “Neural Network Regression” algorithms has shown the least error percentage. The thesis concludes by confirming artificial intelligence algorithms can be used as effective tools for financial time series forecasting.

Keywords: Financial Time Series, Artificial Intelligence, Machine Learning, Market Analysis, Regression

ÖNSÖZ

Mayıs 11, 1997 tarihinde IBM tarafından tasarlanan “Deep Blue” isimli yapay zeka programının dünya satranç şampiyonu Garry Kasparov’u 6 el üzerinden oynanan bir satranç müsabakasında mağlup etmesi ile ilk defa bir yapay zekanın insandan üstün bir performans gösterebileceğini ispatlamıştır. (Campbell, Hoane, & Hsu, 2001) Ortalama bir insan beyni yaklaşık 50 olası hamleyi hesaplama kapasitesinde iken, Deep Blue 200 milyon pozisyon analiz edip 8-12 hamle ötesini hesaplama becerisi göstermekteydi. (Smith, McGuire, & Huang, The History of Artificial Intelligence, 2006) Belirtilen tarih sonrasında insan beyni kayda değer bir evrim geçirmezken yapay zeka evrimine devam ederek 2016 yılında AlphaGo isimli bir algoritmayla dünyanın en eski ve karmaşık strateji oyununda dünya şampiyonu Lee Sedol’u 4-1 yenerek GO oyununda en yüksek mertebe olan 9-dan sertifikasını almıştır. İnsanlık tarihinin en eski ve zeka ile iliştilen iki strateji oyununda yapay zeka algoritmalarının ortaya koyduğu tartışmasız üstünlük bu tezin ana motivasyon kaynağıdır.

Bir diğer ilham kaynağı da Jay R. Galbraith “Organization Design” kitabında karşımıza çıkan “Job Shop Scheduling” problemi olarak geçen bir üretim maksimum verimlilik hesaplama problemidir. Bu problemde bir ürün üretmek için gerekli birçok cihazın gereken sıra ile kullanılmasını optimize etmeye çalışan bir üretim yöneticisi incelenir. Belirtilen problemin karmaşıklığı şöyle açıklanmaktadır: “beş farklı makinede işi olan beş farklı parça için yapılabilecek toplan kombinasyon sayısı 25 milyarın üzerindedir ve her artan her parça ve makine sayısı üstel bir büyüme sağlar.” Bu durumda 2 haftalık üretimin optimal çözümünü bulmak aylar alır. Sonuç olarak optimal olmayan fakat planlayabildiğimiz en iyi çözüm kabul edilir ve üretim bu çözüm ile planlanır. Kitapta bu problem için bilinen en etkili çözümlerin satranç için tasarlanan yapay zeka algoritmaları tarafından üretildiği açıklanır. (Galbraith, 1977) Satranç oyunu için tasarlanan programların üretim ve işlemler yöntemi gibi işletme problemlerine çözüm sunabilmesi, yapay zeka algoritmalarının farklı alanların problemleri için kullanılabilmesine işaretler.

“Job Shop Scheduling” problemi yapısal olarak piyasa tahmin etme problemine benzerlik göstermektedir. Bu problemde belirli bir süre zarfında maksimum verimlilikle üretim elde edilmeye çalışılır. Piyasa tahmini probleminde ise belirli bir süre zarfı için maksimum dönüş

getiren bir portföy tasarımı söz konusudur. Üretim probleminde üretim yapan makinelerden biri kırılabilir, aşırı ısınma sebebi ile performansı düşebilir. Aynı şekilde piyasadaki bir firmanın fiyatı artabilir, düşebilir. İki problemde karmaşıklık problemi olarak nitelendirilebilir ve etkili öngörüler yapabilme hedefindedir.

Tez çalışması araştırma, algoritma seçme ve programlama ile başlamıştır. 7 farklı algoritma yazılmış, ev ortamı için kuvvetli bir bilgisayarda programlar denenmiş, fakat donanımsal kısıtlamalar sebebi ile sonuç alınamamıştır. Sonrasında Microsoft Azure isimli servis ile çok daha kuvvetli bir bulut sunucusu kiralanmıştır. Çalışmalar belirtilen sunucu üzerinde tamamlanmıştır. Donanımsal sınırlamalar ile ilgili detaylı bilgiler tezde sunulmuştur.

Tezin birinci bölümünde çalışma için gerekli finansal bilgi ve teoremler açıklanmıştır. Yapılan çalışmaya karşı hipotezler olan etkin piyasalar hipotezi ve rassal yürüyüş hipotezi incelenmiş, belirtilen hipotezlerin piyasa etkinliği ve rassallığın tanımı gibi yapıtaşları irdelenmiştir. Arbitraj yaratma kavramı incelenmiş ve zaman serileri tür ve tanımlarına değinilmiştir.

Tezin ikinci bölümünde çalışma için gerekli bilgisayar bilimleri konularına değinilmiştir. Yapay zeka tarafından kullanılan mantık ve olasılık teoremleri açıklanmış, eldeki verinin yapısına göre kullanılacak normal dağılım, koşullu olasılık ve Poisson dağılımı gibi kullanılacak regresyon yönteminin temelini oluşturacak konulara değinilmiştir. Son olarak makine öğrenme yöntemlerinin temel prensipleri gösterilmiştir.

Tezin üçüncü bölümünde yazılan 7 yapay zeka algoritması matematiksel mantıkları ile birlikte açıklanmıştır. Algoritmaların seçilme sebeplerine ve tarihsel gelişimlerine değinilmiştir.

Tezin dördüncü bölümünde bir önceki bölümde açıklanan yapay zeka algoritmalarının ürettiği tahmin değerleri gerçek verilerle kıyaslanmış ve performansları incelenmiştir. Uzun vade, orta vade ve kısa vade tahminler olmak üzere 3 farklı öğrenme ve tahmin süresi kullanılmıştır.

Tezin beşinci ve son bölümünde sonuçlar incelenmiş, yapılan çalışmada görülen performansı artırmak ve arbitraj yaratma hedefi için yapılabilecek diğer finans / yapay zeka alanlarını içeren çalışmalar için önerilerde bulunulmuştur.

Sonuç olarak kısa vade tahminlerinde “Hızlı Orman Yüzdelik Dağılımı” algoritmasının, orta vade tahminlerinde “Destekli Karar Ormanı” algoritmasının, kısa vade tahminlerde ise “Poisson” ve “Nöral Ağ” algoritmalarının en etkili tahminleri ürettiği, yöntemlerin finansal analiz amacıyla kullanılabileceği gösterilmiştir.

İÇİNDEKİLER

TEŞEKKÜR.....	I
ÖZET.....	II
ASBTRACT	III
ÖNSÖZ.....	IV
İÇİNDEKİLER.....	VII
TABLOLAR LİSTESİ.....	X
DENKLEMLER LİSTESİ.....	XII
ŞEKİLLER LİSTESİ.....	XIII
SİMGELER VE KISALTMALAR LİSTESİ.....	XIV
GİRİŞ	1
BÖLÜM I. FİNANSAL TEOREMLER.....	6
1.1 Piyasa Etkinliği Kavramı.....	6
1.2 Etkin Piyasalar Hipotezi	7
1.1.1 Zayıf Form Piyasa Etkinliği.....	8
1.1.2 Yarı Güçlü Form Piyasa Etkinliği	8
1.1.3 Güçlü Form Piyasa Etkinliği	9
1.2 Rassal Yürüyüş Hipotezi.....	10
1.3 Zaman Serileri.....	11
BÖLÜM II. MATEMATİK VE BİLGİSAYAR BİLİMLERİ TEOREMLERİ.....	13
2.1 Yapay Zeka Tarihi.....	13
2.1.2 Turing Testi.....	14
2.2 Makine Öğrenmesi.....	15
2.3 Mantıksal Cebir.....	18
2.3.1 Önermeli Mantık (Propositional Logic).....	18
2.3.2 Birinci Derece Mantık (First Order Logic).....	19
2.3.3 Bulanık Mantık (Fuzzy Logic).....	20
2.4 Olasılık Kuramı	21
2.4.1 Normal Dağılım	23
2.4.2 Koşullu Olasılık (Bayes Teoremi)	24
2.4.3 Poisson Dağılımı	27
BÖLÜM III. YAPAY ZEKA REGRESYON ALGORİTMALARI.....	29
3.1 Regresyon Analizi	29

3.1.2 Bayes Doğrusal Regresyon (Bayes Linear Regression)	32
3.1.3 Karar Ağacı Regresyonu (Decision Tree Regression)	34
3.1.4 Destekli Karar Ormanı Regresyonu (Boosted Decision Forest Regression).....	36
3.1.5 Hızlı Orman Yüzdelik Dağılım Regresyonu (Fast Forest Quantile Regression)	38
3.1.6 Nöral Ağ Regresyonu (Neural Network Regression)	40
3.1.7 Poisson Regresyonu (Poisson Regression).....	43
BÖLÜM IV. ALGORİTMALARIN BIST’TE TEST EDİLMESİ	45
3.1 Uzun Vade Öngörülleri.....	50
4.1.1 Doğrusal Regresyon (Linear Regression)	51
4.1.2 Bayes Doğrusal Regresyon (Bayes Linear Regression)	53
4.1.3 Karar Ağacı Regresyonu (Decision Tree Regression)	55
4.1.4 Destekli Karar Ormanı Regresyonu (Boosted Decision Forest Regression).....	57
4.1.5 Hızlı Orman Yüzdelik Dağılım Regresyonu (Fast Forest Quantile Regression)	59
4.1.6 Nöral Ağ Regresyonu (Neural Network Regression)	61
4.1.7 Poisson Regresyonu (Poisson Regression).....	63
4.1.8 Değerlendirme	65
4.2 Orta Vade Öngörülleri	67
4.2.1 Doğrusal Regresyon (Linear Regression)	68
4.2.2 Bayes Doğrusal Regresyon (Bayes Linear Regression)	70
4.2.3 Karar Ağacı Regresyonu (Decision Tree Regression)	72
4.2.4 Destekli Karar Ormanı Regresyonu (Boosted Decision Forest Regression).....	74
4.2.5 Hızlı Orman Yüzdelik Dağılım Regresyonu (Fast Forest Quantile Regression)	76
4.2.6 Nöral Ağ Regresyonu (Neural Network Regression)	78
4.2.7 Poisson Regresyonu (Poisson Regression).....	80
4.2.8 Değerlendirme	82
4.3 Kısa Vade Öngörülleri.....	84
4.3.1 Doğrusal Regresyon (Linear Regression)	85
4.3.2 Bayes Doğrusal Regresyon (Bayes Linear Regression)	87
4.3.3 Karar Ağacı Regresyonu (Decision Tree Regression)	89
4.3.4 Destekli Karar Ormanı Regresyonu (Boosted Decision Forest Regression).....	91
4.3.5 Hızlı Orman Yüzdelik Dağılım Regresyonu (Fast Forest Quantile Regression)	93
4.3.6 Nöral Ağ Regresyonu (Neural Network Regression)	95
4.3.7 Poisson Regresyonu (Poisson Regression).....	97

4.3.8 Deęerlendirme	99
BÖLÜM V. SONUÇ, ÖNERİLER ve GENEL DEęERLENDİRME	101
KAYNAKLAR.....	103

TABLolar LİSTESİ

Tablo 1: Örnek Çıktı	48
Tablo 2: Uzun Vade Doğrusal Regresyon / AKBNK	51
Tablo 3: Uzun Vade Doğrusal Regresyon Ortalama Hata Yüzdellik Dağılımı	52
Tablo 4: Uzun Vade Bayes Doğrusal Regresyon / AKBNK	53
Tablo 5: Uzun Vade Bayes Doğrusal Regresyon Ortalama Hata Yüzdellik Dağılımı	54
Tablo 6: Uzun Vade Karar Ağacı Regresyonu / AKBNK	55
Tablo 7: Uzun Vade Karar Ağacı Regresyonu Ortalama Hata Yüzdellik Dağılımı	56
Tablo 8: Uzun Vade Destekli Karar Ormanı Regresyonu / AKBNK	57
Tablo 9: Uzun Vade Destekli Karar Ormanı Regresyonu Ortalama Hata Yüzdellik Dağılımı	58
Tablo 10: Uzun Vade Hızlı Orman Yüzdellik Dağılım Regresyonu / AKBNK	59
Tablo 11: Uzun Vade Hızlı Orman Yüzdellik Regresyonu Ortalama Hata Yüzdellik Dağılımı	60
Tablo 12: Uzun Vade Nöral Ağ Regresyonu / AKBNK	61
Tablo 13: Uzun Vade Nöral Ağ Regresyonu Ortalama Hata Yüzdellik Dağılımı	62
Tablo 14: Tablo 17: Uzun Vade Poisson Regresyonu / AKBNK	63
Tablo 15: Uzun Vade Poisson Regresyonu Ortalama Hata Yüzdellik Dağılımı	64
Tablo 16: Uzun Vade Sonuç Değerlendirme	65
Tablo 17: Uzun Vade Tüm Algoritma Tahminleri / AKBNK	66
Tablo 18: Orta Vade Doğrusal Regresyon / AKBNK	68
Tablo 19: Orta Vade Doğrusal Regresyon Ortalama Hata Yüzdellik Dağılımı	69
Tablo 20: Orta Vade Bayes Doğrusal Regresyon / AKBNK	70
Tablo 21: Orta Vade Bayes Doğrusal Regresyon Ortalama Hata Yüzdellik Dağılımı	71
Tablo 22: Orta Vade Karar Ağacı Regresyonu / AKBNK	72
Tablo 23: Orta Vade Karar Ağacı Regresyonu Ortalama Hata Yüzdellik Dağılımı	73
Tablo 24: Orta Vade Destekli Karar Ormanı Regresyonu / AKBNK	74
Tablo 25: Orta Vade Destekli Karar Ormanı Regresyonu Ortalama Hata Yüzdellik Dağılımı	75
Tablo 26: Orta Vade Hızlı Orman Dağılım Regresyonu / AKBNK	76
Tablo 27: Orta Vade Hızlı Orman Yüzdellik Regresyonu Ortalama Hata Yüzdellik Dağılımı	77
Tablo 28: Orta Vade Nöral Ağ Regresyonu / AKBNK	78
Tablo 29: Orta Vade Nöral Ağ Regresyonu Ortalama Hata Yüzdellik Dağılımı	79
Tablo 30: Orta Vade Poisson Regresyonu / AKBNK	80
Tablo 31: Orta Vade Poisson Regresyonu Ortalama Hata Yüzdellik Dağılımı	81
Tablo 32: Orta Vade Sonuç Değerlendirme	82
Tablo 33: Orta Vade Tüm Algoritma Tahminleri / AKBNK	83
Tablo 34: Kısa Vade Linear Regresyon / AKBNK	85
Tablo 35: Kısa Vade Doğrusal Regresyon Ortalama Hata Yüzdellik Dağılımı	86
Tablo 36: Kısa Vade Bayes Linear Regresyon / AKBNK	87
Tablo 37: Kısa Vade Bayes Doğrusal Regresyon Ortalama Hata Yüzdellik Dağılımı	88
Tablo 38: Kısa Vade Karar Ağacı Regresyonu / AKBNK	89
Tablo 39: Kısa Vade Karar Ağacı Regresyonu Ortalama Hata Yüzdellik Dağılımı	90
Tablo 40: Kısa Vade Destekli Karar Ormanı Regresyonu / AKBNK	91
Tablo 41: Kısa Vade Destekli Karar Ormanı Regresyonu Ortalama Hata Yüzdellik Dağılımı	92
Tablo 42: Kısa Hızlı Orman Yüzdellik Regresyonu / AKBNK	93
Tablo 43: Kısa Vade Hızlı Orman Yüzdellik Regresyonu Ortalama Hata Yüzdellik Dağılımı	94
Tablo 44: Kısa Vade Nöral Ağ Regresyonu / AKBNK	95

Tablo 45: Kısa Vade Nöral Ağ Ortalama Hata Yüzdeler Dağılımı.....	96
Tablo 46: Kısa Vade Poisson Regresyonu / AKBNK.....	97
Tablo 47: Kısa Vade Poisson Ortalama Hata Yüzdeler Dağılımı	98
Tablo 48: Kısa Vade Sonuç Değerlendirme	99
Tablo 49: Kısa Vade Tüm Algoritma Tahminleri / AKBNK.....	100

DENKLEMLER LİSTESİ

Denklem 1: Olasılık Formülü	23
Denklem 2: Bayes Teoremi / Koşullu Olasılık	25
Denklem 3: Poisson Olasılık Kütle Denklemi	28
Denklem 4: 2 Bağımlı Değişken Doğrusal Regresyon	31
Denklem 5: Doğrusal Regresyon Matris	31
Denklem 6: Bayes Doğrusal Regresyon Matrisi	32
Denklem 7: Bayes Doğrusal Regresyon Olasılık Dağılımı	32
Denklem 8: Destekli Karar Ormanı Regresyon Modeli	36
Denklem 9: Gradyant Artırma Modeli	36
Denklem 10: Destekli Karar Ormanı Modeli	37
Denklem 11: Dağılım Regresyonu Modeli	38
Denklem 12: Dağılım Regresyonu Yüzdellik Model	39
Denklem 13: Olasılık Kütle Fonksiyonu	44

ŞEKİLLER LİSTESİ

Şekil 1: 2 Zar Örnek Uzayı.....	22
Şekil 2: Sınıflandırıcı Karar Ağacı.....	34
Şekil 3: Regresyon Karar Ağacı	35
Şekil 4: Nöron Yapısı	40
Şekil 5: Tek Katmanlı Yapay Nöron.....	41
Şekil 6: Çok Katmanlı Yapay Nöron	42
Şekil 7: Kod Şeması.....	46

SİMGELER VE KISALTMALAR LİSTESİ

AKBNK	Akbank
AKSA	Aksa
ARCLK	Arçelik
ASYAB	Asya Katılım Bankası
BIMAS	Bim Mağazalar
DOHOL	Doğan Holding
EKGYO	Emlak Konut Gmyo
ENKAI	Enka İnşaat
EREGL	Ereğli Demir Çelik
GARAN	Garanti Bankası
HALKB	T. Halk Bankası
IHLAS	İhlas Holding
ISCTR	İş Bankası (C)
KCHOL	Koç Holding
KOZAL	Koza Altın
KRDMD	Kardemir (D)
MGROS	Migros Ticaret
PETKM	Petkim
SAHOL	Sabancı Holding
SISE	Şişe Cam
SNGYO	Sinpaş Gmyo
TCELL	Turkcell
THYAO	Türk Hava Yolları
TKFEN	Tekfen Holding
TOASO	Tofaş Oto. Fab.
TTKOM	Türk Telekom
TTRAK	Türk Traktör

TUPRS	Tüpraş
VAKBN	Vakıflar Bankası
YKBNK	Yapı Ve Kredi Bank
BIST	Borsa İstanbul
Q	Reel Sayılar
Q+	Pozitif Reel Sayılar
N	Doğal Sayılar
Z	Tam Sayılar
Z+	Pozitif Tam Sayılar
Z-	Negatif Tam Sayılar
\Rightarrow	“İse” koşul önermesi
\Leftrightarrow	“Ancak ve ancak” koşul önermesi
V	“Veya” koşul önermesi
\wedge	“Ve” koşul önermesi
\sim	Değilleme önerme eklemi
T	Totoloji: Kesin Doğruluk
F	Kontradiksyon: Kesin Yanlışıklık
\forall	Evrensel niceleyici
\exists	Özel niceleyici

GİRİŞ

Finans alanında arařtırmacı ve yöneticilerin karřılařtıđı en zorlu problem belirsizliktir. Belirsizlik finansal teörinin ayrılmaz bir parçası olan risk faktörünü de yanında getirir. Riskin ortaya çıkması sadece finansal karar almayı zorlařtırmakla kalmaz, aynı zamanda riski etkin ve etkili bir şekilde yönetebilen ve analiz edebilen yatırımcılar için karlı fırsatlar da yaratır. Piyasadaki durumun gelecekte nasıl olacađı hakkında bilgi sahibi olmak finans alanında çalışan birçok yatırımcı ve arařtırmacının hedefidir. Hiçbir analist veya yatırımcı gelecekte piyasanın nasıl şekilleneceđini kesin olarak öngöremez. Menkul kıymet fiyatları, Louis Bachelier'in rassal yürüyüş hipotezince rasgele bir yürüyüş izler ve bu sebeple tahmin edilemez.

Rassal yürüyüş hipotezinin üzerine kurulu olduđu “rasgelelik” kavramı ise binlerce yıldır tartıřılan felsefi bir sorudur. Rasgelelik nedir? Hangi durumların sonucu önceden belirlenebilir? Bu soruya kargařa kuramı, oyun teorisi, istatistik, olasılık teorisi gibi farklı alanlardan farklı cevaplar bulmak mümkün olsa da tüm tanımlarda ortak olarak rasgelelik kavramı sonucun önceden bilinmediđi ve sonuçların patern içermediđi olaylar için kullanılır. Sonucu belirlenemeyen (indeterminability) olaylar tarihsel olarak incelendiđinde, milattan 2 bin yıl önce Mısır, Hint ve Çin uygarlıklarında olasılık içeren oyunlar görölmektedir. Yunan filozofların çalışmaları incelendiđinde, rasgelelik epistemik erişim yetersizliđi ve ontolojik bir özellik olarak iki farklı yaklařımla ele alınmaktadır. Epistemik erişim yetersizliđi yaklařımınca gerçeklik deterministiktir fakat insanođlu gerçekliđi algılamak için gerekli bilişsel araçlara sahip deđildir. Ontolojik yaklařımca ise gerçeklik deterministik deđildir (Rosenberg, 2011). Milattan önce 4. Yüzyılda Democritus'un çalışlarında rasgelelik kavramı görölmekle birlikte rasgelelik durumunun epistemik erişim yetersizliđi kaynaklı olması durumu ifade edilmiřtir. Democritus'un örneđince “Bir efendi iki farklı uřađı aynı anda su almaya gönderirse iki uřak yolda karřılařacaktır. Bu karřılařma uřaklar için rasgele gerçekleřmiř olsa da sonuç efendi tarafından önceden bilinmektedir.” (Hromkovic, 2005). Bu tez felsefi olarak belirtilen yaklařımı desteklemektedir. Hilesiz bir zar atıldıđında zarın yüzeyinde çıkacak sonuç rasgele kabul edilir. Olasılık teoremi 6 yüzeyli bir zar için her yüzeyin gelme řansının eşit ve 1/6 olduđunu söylese de kořullu olasılık teoremince önceki zarların sonuçları hakkında bilgi sahibi olmak mevcut olasılıđa etki eder. Daha da detaylı olarak incelendiđinde zar, zarın sekme kat sayısı, elden çıkma açısı, elden çıkma hızı, zarın sektiđi yüzeyin sekme kat sayısı, rüzgar ve yerçekimi faktörlerince belirlenen fizik

yasalarınca hareket etmektedir. Bu durumda zar atıldığı anda bu bilgi ve bilgiyi işleme hızına sahip olan bir kimse, zar durmadan önce hangi yüzeyin üstte geleceğini bilebilir. “Raslantısal” olarak tanımladığımız birçok durum, aslında eldeki bilgi ve bilgi işleme hızının yetersiz kaldığı durumlardır.

Bilgisayar bilimleri tarafından incelendiğinde ise rassallık kavramı daha karmaşık bir hal alır. Bir insan için rasgele bir sayı tutmak basit bir işlem iken, bilgisayar bağlamında rasgele diye bir durum yoktur, bilgisayarlar rasgele bir sayı seçemezler. Bunun yerine insanların rassal olarak değerlendirdiği sonuçları taklit eden komplike matematiksel algoritmalar vardır. Belirtilen algoritmalar “gerçek rassal algoritmalar” ve “rassal varsayılan (psuedo-random) algoritmalar” olarak ikiye ayrılır. Gerçek rassal algoritmalar bilgisayarın dışında gerçekleşen bir fiziksel fenomenden girdi alır. Örnek olarak rasgele bir sayı üreten bir algoritma, kullanıcının klavyede bastığı son 1000 tuş için her tuş basma arasında geçen zamanın aritmetik algoritmasını alıp size rasgele bir sayı olarak sunabilir. Rassal sayılan algoritmalar ise bilgisayarın içindeki kaynaklar ile rassal görünen fakat olasılık teoremlerince aslında tahmin edilebilir bir sayı üretir.

Bu algoritmalar üç özelliğe sahip olmalıdırlar. Birinci özellik Martin Löff tarafından öne sürülen “tipik” olma özelliğidir. Özel sonuçlardansa tipik sonuçlar rasgelelik algısı yaratır. Örnek olarak iki tabanında rasgele üretilen “10001011101111010000” veya “01111011001101110001” sayıları insan gözü için rassal sayılabilirken “00000000000000000000” sonucu belirli bir patern izleyen özel bir durumdur ve rassal olarak algılanmaz. İkinci özellik Kolmogorov tarafından öne sürülen “kaotik” olma özelliğidir. Her rassal seri kendisini açıklayan basit bir kural olmaması yönünde kaotik olmalıdır. Az önceki örnekte 1. Ve 2. örnek kaotik iken, son örnek bu özelliği taşımamaktadır. Son olarak Von Mises tarafından öne sürülen “frekans dağılımı tutarlılığı” özelliği sağlanmalıdır. 6 yüzeyle bir zarın 6 defa atılması durumunda 5 defa aynı yüzeyin geleceği sonucunu veren bir algoritma, istatistiksel frekans dağılımı ile tutarsız olduğu için rassal kabul edilmeyecektir (Kolmogorov & Uspenskii, 1987). Bir şans oyunu programı için rassal sayılan algoritmalar daha faydalı iken, kriptografi algoritması gibi algoritmalarda bir saldırganın tahmin edemeyeceği gerçek rassal sayılar faydalıdır. İnsan algısı için rassal bir değer yaratma durumu için tasarlanan bu algoritmalar, bilgisayar algısı için tam olarak rassal olmayabilir. Mevcut finans dünyasında bilgisayar kullanımının ne kadar önemli olduğu da göz önünde bulundurulursa, rassal yürüyüş hipotezinde belirtilen rasgelelik durumu insan

algısının epistemik erişim yetersizliği sebebi ile ortaya çıkıyor olabilir. Bu durumda veriye erişme, işleme ve saklama yönünden insana kıyasla üstün becerileri olan bilgisayarlar, rassallık illüzyonunun altında yatan etkenleri ortaya koyabilirler.

Finans alanına dönülürse rassal etkenler içeren geleceği öngörmek mümkün olmasa da öngörü performansını artıran teknikler mevcuttur. Geçmişte yaşanan benzer durumlarla karşılaştırma, mevsimsellik analizi, mevcut eğilimlerin gözlenmesi, toplum psikolojisi analizi ve istatistiksel analizler bu tekniklere örnektir. Aynı hava durumu analizlerinin nem, basınç, rüzgar gibi faktörler hakkında sahip olduğu veriler ile öngörü yapabilmesi ve kesinlik içermese de büyük ölçüde doğru kabul edilmesi gibi bu tez finans alanında da piyasaya açık olan veriler analiz edilerek öngörü çıkarımları için kullanılabilceği düşüncesi üzerine kuruludur.

Piyasaya açık ve hatta açık olmayan verilerin arbitraj yaratma amacı ile kullanılması konusu, Eugene Fama'nın etkin piyasalar hipotezince mümkün değildir. Etkin piyasalar hipotezi ekonomi alanında ampirik veriler ile en fazla desteklenen hipotezlerdendir. Hipotez, geçmiş çalışmalarca New York ve Amerika Menkul Kıymetler Borsası, Avustralya, İngiltere ve Almanya gibi dünyanın en büyük borsalarında test edilmiş ve büyük çoğunlukla onaylanmıştır. Fakat belirtilen hipotez etkinlik seviyesinin maksimum olduğu teorik bir piyasa için geçerlidir. Gerçek piyasalarda etkinlik seviyesince bilginin fiyata yansımaları, spekülasyon ve benzeri sebeplerden ötürü gecikir ve kısa süreli de olsa bir arbitraj yaratılmış olur. Unutulmamalıdır ki, etkin piyasalar hipotezi 1970'te öne sürülmüştür ve geçen yarım asırlık süre içinde hem finansal enstrümanlar hem de telekomünikasyon ve bilgisayar alanındaki gelişmeler ile yatırım yapma işlemi birçok değişiklikten geçmiş, bu süre zarfında hipotezin öne sürdüğü koşulların sağlanmadığı durumlar görülmüştür. Hipotezin öne sürdüğü gibi tüm firmalar hakkındaki bilgilerin şeffaf ve verimli bir şekilde halka açık olması durumunda Enron, MCI WorldCOM ve Lehman Brothers gibi bir gecede 60 milyar dolar üzerinde değer kayıplarının yaşanması mümkün olmazdı. Hipotez aynı zamanda yatırımcıların mantıklı olduğu ve riske uyarlanmış dönüşler beklediğini varsaymaktadır ancak firma ismi ve büyüklüğü gibi sebeplerle duyguları ile hareket eden "mantıksız" yatırımcılar piyasada aktiftir. Son olarak Warren Buffet, George Soros, Peter Lynch gibi örnekler incelediğinde devamlı olarak piyasa ortalamasının üzerinde getiri sağlayan yatırımlar yapmanın mümkün olduğu görülmektedir. Bu motivasyon ile bu tez piyasa geleceği ekin olarak tahmin edebilecek bir sistem tasarlama ve test etme amacındadır.

Alım – satım, riskten korunma, yatırım gibi alanlarda karar almak için matematiksel finans, sayısal yöntemler ve bilgisayar simülasyonlarına dayanan ve bu kararların risk yönetimini kolaylaştıran disipline kompütasyonel finans (computational finance) veya mali mühendislik (financial engineering) denmektedir. Çok sayıda değişken, girdi ve çıktılarının olduğu bu mali sistemlerde bilgisayarların hesaplama hızı, hafıza genişliği ve matematiksel keskinlikleri uzun zamandır finans alanındaki araştırmacılar tarafından belirli finansal araçların yarattığı riski etkin olarak hesaplamak için kullanılmaktadır. Kompütasyonel finans yöntemleri; yatırım bankacılığı, kurumsal stratejik planlama, menkul kıymet ticareti ve finansal risk yönetimi gibi alanlarda sıkça görülür.

Son yıllarda ön plana çıkan yapay zeka alanı kompütasyonel finans için uygun yöntemler sunmaktadır. Yapay zeka alanı ve sunduğu yöntemler finans alanına kıyasla yeni yöntemler oldukları için uygulamaları henüz birçok disiplinde kullanılmaya başlanmamış veya yüksek performans evresine geçmemiştir. Finans alanında yapay zeka öngörülerini üzerine yapılan çalışmalar incelendiğinde Lee, C. M., & Ko, C. N. (2009) “Time Series Prediction using RBF Neural Networks with nonlinear time-varying evolution PSO algorithm”, Yan, X. B., Wang, Z., Yu, S. H., & Li, Y. J. (2005). “Time Series Forecasting with RBF Neural Network”, Marcek, D., Marcek, M., & Babel, J. (2009). “Granular RBF NN Approach and statistical Methods Applied to Modelling and Forecasting High Frequency Data”, Cao, L. J., & Tay, F. E. H. (2003). “Support Vector Machine with Adaptive Parameters in Financial Time Series Forecasting. IEEE Transactions on Neural Networks”, Enke, D., & Thawornwong, S. (2005). “The Use of Old Data Mining and Neural Networks for Forecasting Stock Markets Returns.”, Theofilatos, K., et al. (2012). “Modelling and Trading the DJIA Financial Index Using Neural Networks Optimized with Adaptive Evolutionary Algorithms” gibi öne çıkan çalışmaların son 20 yılda yapılmış oldukları görülmektedir.

Taylor’cu bir yaklaşım izleyen bu tez, “en uygun yöntem” ve “en uygun çalışan” seçme arayışı için farklı bilgisayar donanımları ve farklı yapay zeka algoritmaları denemiştir. Kullanılan algoritmalar ilerleyen bölümlerde detaylı olarak açıklanmıştır. Bilgisayar donanımı bağlamında öncelikle ev bilgisayarı ve kuvvetli bir grafik işlemcisi ile programlar denenmiş fakat donanımsal yetersizlikler sebebi ile efektif sonuç alınamamıştır. Sonrasında Microsoft Azure servisi ile bir sunucu kiralanmış, çalışma bu sunucu üzerinde tamamlanmıştır.

Belirtilen algoritma analizlerinin etkili olabilmeleri için ne kadar sürede tamamlanabildikleri önemli bir kriterdir. Bir gün sonraki piyasa durumunu %80 doğruluk payıyla tahmin etmek için iki gün harcayan bir analistin sonuçları, analizler doğru olsa bile kullanılamaz. Bu durumda %60 doğruluk payı ile bir saatte sonuçları çıkaran bir yöntem çok daha kullanışlı olacaktır. Piyasaların devamlı veri üretmesi, fiyatları etkileyen etkenlerin sayıca fazlalığı, bu öngörüler için kullanılacak veri setlerinin karmaşık ve dinamik olmasını sağlar. Bu etkenler sonsuz farklı analiz yöntemi ortaya çıkabileceği anlamını taşır ve bu sebeple tek bir yöntemin piyasada devamlı en yüksek performansı göstermesi olası değildir. Bu durumda piyasa tahminleri bir karmaşıklık problemi olarak tanımlanabilir, karlı fırsatlar yakalama yönünde %100 doğru bir cevap arayışında değil etkin olabilecek bir sürede en verimli çözümü üretmek hedefindedir. Çalışmada kullanılan tüm algoritmalar bir saatin altında sonuç vermektedir.

Finans sektöründe yapay zeka kullanımı ile ilgili yabancı kaynaklarda birçok çalışma olmasına rağmen konu hakkında Türkiye’de yapılmış akademik çalışma sayısı noksanlığı göze çarpmaktadır. Bu eksiğin doldurulmasına katkı yapmak amacıyla olan bu tez, İstanbul Menkul Kıymetler borsası verileri üzerine farklı makine öğrenmesi teknikleri kullanarak öngörü yapan sistemlerin işleme şekli, tasarımı ve performans analizlerini içerir. Bu tez belirtilen sorunun karmaşık doğasını ele almak ve yapay zeka alanının sunduğu öngörü yöntemlerinin finans alanında göstereceği performansı incelemek için finans ve bilgisayar mühendisliği bilimlerini içeren disiplinler arası bir yaklaşım ele almaktadır.

Tezin birinci bölümü, yapılan çalışma ile ilgili olan finans teoremlerini içerir. İkinci bölümde belirtilen yapay zeka algoritmalarının çalışması için gerekli olan matematik ve bilgisayar mühendisliği teoremleri açıklanmıştır. Üçüncü bölümde yazılan algoritmalar, algoritmaların seçilme sebepleri ve matematiksel formülleri ile açıklanmıştır. Dördüncü bölümde bu yedi algoritma BIST30 verileri ile eğitilmiş, tüm algoritmalarca belirli sayılarda ileri tarih günlük verileri tahmin edilmiş ve bu sonuçlar gerçek veriler ile kıyaslanarak sonuçlar sunulmuştur. Beşinci ve son bölümde ise sonuçlar yorumlanmış ve bu çalışmanın ötesinde gerçekleştirilebilecek uygulamalar için bulgu ve öneriler belirtilmiştir.

BÖLÜM I. FİNANSAL TEOREMLER

Bu bölümde çalışmayla ilişkili finansal teoremler sunulmuştur. Öncelikle halka açık verilerin kar etme amacı ile kullanılması konusunda Eugene Fama'nın "Etkin Piyasalar Hipotezi" incelenmiştir. Sonrasında piyasadaki hareketleri öngörme hedefinin başarısız olacağını savunan Louis Bachelier'in "Rassal Yürüyüş Hipotezi" sunulmuştur. Son olarak çalışma için kullanılan veri seti bir finansal zaman serisi olduğu için zaman serileri açıklanmıştır.

1.1 Piyasa Etkinliği Kavramı

Piyasa etkinliği bir piyasada yer alan varlıkların fiyatlarının elde edilebilir tüm bilgileri yansıtması olarak tanımlanır. Piyasa etkinliği; faaliyet etkinliği, kaynak dağılımı etkinliği ve bilgi etkinliği olarak üç farklı türe ayrılır.

Faaliyet etkinliği piyasadaki fon arz ve taleplerinin minimum maliyetle işleme alınmasını içerir. Maliyet ne kadar düşük ise faaliyet etkinliği o kadar yüksektir. Kaynak dağılımı etkinliği piyasadaki kaynakların ideal dağıtımınıdır. Kaynakların talebin en yüksek olduğu mal ve hizmetlerin üretiminde kullanılma etkinliğidir. Kaynaklar talep edilen mal ve hizmetler için ne kadar fazla kullanılıyorsa etkinlik kaynak dağılımı etkinliği o kadar yüksektir. Son olarak bilgi etkinliği, piyasadaki fiyatların piyasada bulunan bilginin ne kadarını yansıttığı durumudur. Fiyatlar piyasadaki bilgileri ne kadar yansıtıyorsa bilgi etkinliği o kadar yüksektir. Etkin piyasalar hipotezindeki etkin kavramı bilgi etkinliğini ifade etmektedir. Etkinlik seviyesi yüksek bir markette fiyatlar piyasadaki bilgiyi içermiş olduğundan, herhangi bir yatırımcı herhangi bir bilgiyi kullanarak üstün bir getiri sağlayamazdır. (Fama , 1970)

Bir menkul kıymet piyasasının etkinliği, aşağıda belirtilen koşullara bağlıdır:

- Fiyatlar mevcut tüm bilgilerin kullanılmasıyla oluşan piyasa dengesini yansıtmalıdır
- Fiyatlar hemen ya da çok az gecikme ile, tarafsız olarak yeni bilgi girişine verilen tepkiyi yansıtmalıdır.

Fiyatların bilgi girişini doğru olarak yansıtması, varlıkları borsada işlem gören firmalara ait bilgileri toplayıp analiz edecek ve yorumlayacak profesyonel bir yatırımcı gurubunun varlığına bağlıdır. Bu profesyoneller piyasayı takip ederek bilgi girişinin fiyatlara tam ve anında yansıtması sağlamaktadırlar. (Fama , 1970) (Karan, 2001) (Bayraktar, 2012)

1.2 Etkin Piyasalar Hipotezi

Etkin piyasalar hipotezi Nobel ödüllü Amerikalı ekonomist Eugene Fama tarafından 1970 yılında doktora tezi olarak geliştirilmiştir. Piyasalarda gerçekleşen fiyat hareketlerinin rastlantısal olduğu, tesadüfi bir dağılım gösterdiklerini ve bu sebeple önceden tahmin edilemeyecekleri düşüncesi bu hipotez ile birlikte literatüre girmiş olsa da hipotezin altındaki felsefe çok daha eskiye dayanır. 1565 yılında İtalyan Matematikçi Girolamo Cardano, Şans Oyunları Kitabı (Liber de Ludo Aleae” isimli çalışmasında “Kumar oyunlarının temel kuralı en basit anlamda eşit koşullardır (Rakip, seyirci, kullanılan para, zar, zar kutusu vs). Bu eşit koşullardan şaşılması durumunda rakibin lehine bir durumsa aptallık, kendi lehinize bir durumsa adaletsizlik yapılmaktadır.” demiştir. 1900 yılında Fransız matematikçi Louis Bachelier’in çalışmalarında etkin piyasalar bir martingale sistemi olarak tanımlanmıştır. 1949 yılında Holbrook Working ideal bir piyasada hiç kimsenin başarılı olarak gelecek fiyatları tahmin edemeyeceğini göstermiştir. (Gun & Bernard, 2011)

Eugene Fama’ya göre etkin piyasa, “geniş ve rasyonel bir yatırımcı kitlesinin kar maksimizasyonu için birbirleri ile rekabet halinde olduğu, bilginin herkes tarafından kolay ulaşılabilir olduğu piyasa koşulları” olarak tanımlanmaktadır. (Fama , 1970) Burada önemli noktalar bilginin herkese açık olması ve bu sebeple fiyatların tamamen bu bilgiyi içerdiği önermesidir. Etkin piyasalar hipotezi piyasadaki tüm bilginin hisse senetlerinin fiyatları içine dahil edildiğini, hisse senetlerinin gerçek değerlerini yansıttığını, yani değerinden az ya da fazla fiyattan işlem görmediklerini, bu koşullarda yatırımcıların piyasa ortalamasının üzerinde getiri elde edemeyeceklerini savunmaktadır. Bu tanımın bir sonucu olarak piyasada faaliyet gösteren hiçbir yatırımcının bu bilgileri kullanarak ortalamanın üzerinde bir getiri elde etme şansı yoktur. Hisse senedi tercihi noktasında, herhangi bir tercihin ortalamadan daha iyi bir performans gösterebilme olanağı yoktur. Etkin bir pazarda oluşan fiyat, menkul kıymetin gerçek değerinin en iyi tahmini olacağından piyasada düşük veya aşırı değerlendirilmiş menkul kıymet bulmak mümkün değildir. (Tunçel, 2007) Bu durumda temel analiz veya teknik analiz gibi tekniklerin hiçbiri kullanışlı değildir. Piyasada oluşan fiyatları piyasaya

giren bilgi akışının bir fonksiyonu olarak tanımlayan bu hipotez üç temel varsayımı doğru kabul etmektedir:

- Yatırımcılar rasyoneldir
- Yatırımcılar eksiksiz bilgiye sahiptir
- Yatırımcılar karlarını maksimize etmeyi beklemektedir.

Borsaların etkinlik seviyesini ölçme işlemlerinde hisse senedi fiyatları ve getirilerin davranışı temel gösterge olarak ele alınmaktadır. Hisse fiyatlarının belirtilen bilgilerin ne kadarını yansıttığı piyasa etkinlik seviyesi olarak nitelendirilir. Etkin piyasalar hipotezince piyasa etkinliği zayıf form, yarı güçlü form ve güçlü form olmak üzere üç kategoride sınıflandırılır.

1.1.1 Zayıf Form Piyasa Etkinliği

Zayıf etkinlik formunda piyasa fiyatları tüm ilgili geçmiş fiyat bilgileri yansıtmaktadır. Yani güncel varlık fiyatları piyasadaki tüm geçmiş fiyat bilgilerini içerir. Bu tür piyasalarda yatırımcılar yeni bilgilere eş zamanlı olarak ulaşamazlar. Ayrıca kamuya açıklanmamış bilgilerin belirli piyasa katılımcıları tarafından önceden bilindiği kabul edilir. Bilgiye erken ulaşan yatırımcılar avantajlı konumdadır. (Fama , 1970)

Hipotezin en düşük derecesi olan zayıf etkinlik derecesinde geçmiş bilgiler mevcut fiyatların içinde var olduğu için yatırımcılar geçmiş fiyat hareketlerini kullanarak normalin üstünde getiri sağlayamazlar. Geçmiş fiyat verileri ile analiz yaparak gelecek fiyatları tahmin etmek mümkün değildir. Dolayısıyla teknik analiz, zaman serisi ve benzeri analizler kullanışsızdır. Rastgele davranışlar söz konusu olduğu için belirtilen analizlerin hiçbiri uzun vadede çalışmaz. (Karan, 2001) (Fama , 1970)

1.1.2 Yarı Güçlü Form Piyasa Etkinliği

Yarı güçlü etkinlik formunda piyasa fiyatları halka açıklanmış tüm bilgileri içerdiği kabul edilir. Bir piyasanın yarı güçlü formda etkin olduğunu söyleyebilmek için hisse senedi fiyatları, gazete makaleleri, yıllık mali raporlar, şirket öngörülerini gibi kamuoyuna sunulan, tüm halka açık bilgiler anında fiyatlara tarafsız olarak yansımalıdır (Fama , 1970). Bu formda

bilanço bilgileri, halka açık bilgiler olduğu için önemli rol onarlar. Yarı güçlü piyasa etkinliğini test etmek için kamuoyuna sunulan bilgiler ile hisse senedi getirileri izlenir. Eğer yatırımcılar sürekli olarak aşırı getiri sağlıyorsa, bu piyasa kamuoyuna duyurulan bilgiye göre etkin değildir. Örnek olarak bir piyasadaki hisse senetleri temettü dağıtımını duyurusundan sonra aşırı getiri sağlıyorsa o piyasa temettü dağıtımını duyurusuna göre yarı güçlü formda etkin değildir. (Karan, 2001)

Yarı etkin piyasalarda iç bilgiye sahip olan yatırımcıların öncelikli olarak bilgiye ulaşabilmeleri söz konusudur (inside information). Bilgiye öncelikli olarak ulaşan kimselerin, kamunun sahip olmadığı bilgileri kullanarak ortalama piyasa getirisi üzerinde kazanç elde etmeleri mümkündür. (Fama , 1970)

1.1.3 Güçlü Form Piyasa Etkinliği

Son etkin piyasa formu olan güçlü form piyasa etkinliğinde ise fiyatların gerçek değeri belirlemede kullanılacak tüm bilgileri yansıttığı kabul edilir. Bu durumda profesyonel portföy yöneticileri, iç bilgi sahipleri ve bunun gibi bilgi sahibi kimseler normalin üzerinde getiri sağlayamazlar. (Fama , 1970)

Güçlü form piyasa etkinliğinde fiyatlar yalnızca kamuya açıklanmış bilgileri değil aynı zamanda makro ve mikro ekonomiye ait tüm bilgileri de içermektedir. İçeriden edinilen bilgi eş zamanlı olarak tüm yatırımcılara ulaşmaktadır. Bu piyasa koşullarında yatırımcı fiyatları kalan iki forma göre daha etkili tahmin edebilir. Dolayısıyla bu etkinlik seviyesine sahip piyasalarda temel veya teknik analiz yöntemleri ile getiri sağlanamaz. Aşırı getiri sağlanamamasının nedeni, kamuya açıklanmış veya açıklanmamış tüm bilgilerin çok hızlı şekilde, tüm yatırımcılara ulaşması dolayısıyla bu bilgilerin çok hızlı olarak fiyatlara yansımalarıdır. (Laffont & Maskin, 1990)

1.2 Rassal Yürüyüş Hipotezi

Rassal yürüyüş hipotezi Louis Bachelier tarafından “Spekülasyon Teorisi “(Le Theorie de la Speculation”) isimli doktora tez çalışmasında öne sürülmüştür. Bu çalışma aynı zamanda birçok kaynakta matematiksel finansın doğuşu olarak gösterilir. Bugün “stokastik analiz” olarak geçen birçok yöntem bu çalışmaya dayanır. Çalışma, Brown hareketini matematiksel bir model haline getirip hisse değerlemesi için kullanır. Çalışma finans alanında ileri düzey matematik kullanan ilk örnektir. (Bachelier, 2007)

Rassal yürüyüş hipotezi piyasa fiyatlarının rasgele bir yürüyüş izlediğini ve bu sebeple tahmin edilemeyeceğini öne sürer. Herhangi bir anda piyasadaki herhangi bir varlığın fiyatının artma olasılığı ve düşme olasılığı tam olarak eşittir. Burton Marliel’in sözleri ile “gözü bağlanmış bir maymunun bir gazetenin finans sayfalarına rasgele atacağı dart oklarıyla oluşturulan bir portföy, eğitilmiş uzmanların oluşturacağı portföy kadar etkili olacaktır.” (Malkiel, 1973). Rassal yürüyüş hipotezi etkin piyasalar hipotezi ile tutarlıdır. İki hipotez arasındaki temel farklılık, rassal yürüyüş hipotezi fiyatlar rasgele oluştuğu için tahmin edilemeyeceğini öne sürerken, etkin piyasalar hipotezinin fiyatı belirleyen bilgilerin halihazırda fiyatın içinde bulunması sebebi ile tahmin için işe yaramaz olmasıdır.

Rassal yürüyüş finansal teorinin dışında birçok farklı bilim dalındaki çalışmalarda da görülmektedir. 1880 yılında Lord Rayleigh ses dalgalarının heterojen materyaller içinde gösterdikleri dağılımı incelerken kullanmıştır. 1905 yılında Karl Pearson’un bir ormanı işgal eden sivrisinekleri modelleyen çalışmasında her seferinde rasgele seçilen bir yönde “n” birim ilerleyen sineklerin “n” adım sonrasında ne kadar uzağa gidebileceğini rassal yürüyüş ile göstermiştir. Albert Einstein gaz moleküllerinin çarpışması sonucu izlediği yollar ve Brown hareketi çalışmasında yine rassal yürüyüş modelini kullanmıştır. (Rycroft & Bazant, 2005)

1.3 Zaman Serileri

Zaman serileri, kronolojik olarak sıralanan veri dizileridir. Veri satırları periyodik bir döngü ile sıralanır. Belirtilen periyod saat, gün, ay, yıl gibi farklı değerler olabilir. Önemli olan eşit aralıklar kullanılmasıdır. Zaman serileri sayısal olarak ifade edilebilecek olaylar ve işlemlerin bir zaman etiketi ile ilişkilendirildiği veri setleri olarak da tanımlanabilir. (George E. P. Box, 2015)

Genel olarak zaman serisi, T örneklem büyüklüğü olmak üzere z_t , $t = 1, 2, \dots, T$ biçiminde gösterilir. Buna göre ilk gözlemlenen veri Z_1 ; ikinci gözlemlenen veri Z_2 ; son gözlemlenen veri Z_T ile ifade edilir. Matematiksel olarak bakıldığında bir zaman serisi “ t ” zamanına bağlı bir veri seti olarak $t: \{y(t_0), y(t_1), \dots, y(t_k), \dots\}$ şeklinde tanımlanabilir. Bu serinin herhangi bir elemanı bağımsız bir değişkene bağlı bir fonksiyon ve zaman değeri olarak belirtilebilir: $y(t) = f(a, b, c, \dots, n, t)$. (Taylor, 2008)

Zaman serileri, gözlem kümesince temsil edilen gerçeğin anlaşılması ve zaman serisindeki değişkenlerin gelecekteki değerlerinin doğru bir şekilde tahmin edilmesi amacı ile kullanılır. Zaman serileri dört farklı bileşenden oluşur:

- Trend Bileşeni: Zaman serilerinin uzun süre sonucunda gösterdikleri kararlı durumdur. Zaman serileri uzun dönem açısından kararlı alçalma, yükselme veya sabit kalma eğilimine sahiptir.
- Mevsimsel Bileşen: Zaman serileri açısından kullanılan veriler dönemler arası farklılık gösterir. Zaman serilerinde görülen mevsimsel değişimleri ifade eder.
- Konjonktürel Bileşen: Bir yıldan fazla süren ve uzunluğu sabit kalmayan ve genellikle konjonktürel dalgalanmaların etkisinden dolayı birçok değişkende görülen dalgalanmalardır.
- Rassal Bileşen: Kalan bileşenler tarafından açıklanamayan oynamaların oluşturduğu bileşendir.

Zaman içinde sürekli olarak kaydedilebilen verilere sahip serilere sürekli zaman serileri, sadece belli aralıklarda elde edilebilen verilere sahip serilere de kesikli zaman serileri adı verilmektedir. Elektrik sinyalleri, voltaj, ses titreşimleri gibi mühendislik alanlarına ait seriler sürekli zaman serileri iken; faiz oranı, satış hacmi, üretim miktarı gibi iktisadi seriler kesikli zaman serileridir. Finans alanında incelediğimiz seriler kesikli zaman serileridir.

BÖLÜM II. MATEMATİK VE BİLGİSAYAR BİLİMLERİ TEOREMLERİ

Bu bölümde çalışmayla ilgili matematik ve bilgisayar bilimleri teoremleri sunulmuştur. Öncelikle bilgisayar bilimleri içinde yapay zeka konusunun çıkışını incelemek için yapay zeka tarihi incelenmiştir. Sonrasında çalışmada kullanılan, yapay zekanın bir alt dalı olan “makine öğrenmesi” konusu açıklanmış, son olarak makine öğrenmesi gerçekleştirebilmek için bilgisayarın kullandığı temel mantık ve olasılık teoremleri gösterilmiştir.

2.1 Yapay Zeka Tarihi

Yapay zeka 1950’lerden beri çalışılan bilgisayar bilimleri alanının en karmaşık konularından biridir. Yapay zeka, bilgisayarların insan dilini anlamasından satranç gibi oyunlarda hamle hesaplamalarına, patern yakalama tekniklerinden el yazısı okuma sistemlerine çok geniş bir kapsama sahiptir.

“Yapay Zeka” terimi ilk kez 1956 yılında John McCarthy tarafından konu hakkındaki bir akademik konferansta kullanılmıştır. Fakat makinelerin düşünmesi konsepti bundan da önceki çalışmalara dayanır. Vannevar Bush’un 1945’te yayımladığı “As We May Think” isimli çalışması insan zeka ve algısını kuvvetlendiren yardımcı sistemleri barındırır. 5 yıl sonrasında yapay zeka’nın önemli isimlerinden Alan Turing insanları taklit eden ve satranç gibi insan zekası gerektiren oyunlarda karar alabilen makinelerden bahsetmiştir. (Smith, McGuire, & Huang, The History of Artificial Intelligence, 2006)

Yapay zeka son 60 yıl içinde arama algoritmaları, makine öğrenmesi algoritmaları ve dil analizi gibi onlarca alt dal çıkarmıştır. Öte yandan yapay zeka alanında ortaya çıkan birçok gelişme çoğu insan tarafından fark edilmemektedir. Yapay zeka insanların beklediğinin aksine konuşan makineler veya Mars’a uzay gemisi göndermekten çok; kişiye özel pazarlama, arama motorları gibi fark edilmeyen işlevlerde kullanılır. (Adams, 2004)

2.1.2 Turing Testi

Yapay zeka arařtırmalarının merkezinde, Alan Turing tarafından ortaya sürülen Turing Testi bulunur. 1950 yılında İngiliz matematisyen Alan Turing'in yayımladıđı "Computing Machinery and Intelligence" makalesi insandan ayırt edilemeyecek bir yapay zeka konseptini ortaya sürmüřtür. Bu konseptin "Yapay zeka" ismini alması ise John McCarthy'nin çalışmalarına dayanır. Makalelerin arařtırma konusu makinelerin düşünüp düşünemeyeceđidir. Turing, bu soruya bir cevap bulabilmek için basit bir test öne sürer. "İmitasyon oyunu" olarak ta bilinen bu test pragmatik bir yaklaşımda bulunur: Bir işin makine tarafından mı insan tarafından mı yapıldıđı bilinemediđi noktada makine testi geçmiř sayılır. (Smith, Ting Huang, Yang, & McGuire, 2006)

Turing'in testi aslında bir erkek, bir kadın ve sorgulama yapan bir üçüncü kiřiyi barındıran bir oyuna dayanır. Sorgulama yapan kiřinin amacı hangi kiřinin bay, hangi kiřinin bayan olduđunu bulabilmektir. Sorgulama yapan kiři, karřısındaki kiřinin cinsiyetini onu görmeden, fakat ses tonu veya el yazısı gibi çıktıları kullanarak tahmin edebilir. Turing testinde ise bu testte kullanılan bay veya bayan bir makine ile deđiřtirilir. Sorgulama yapan kiři ise bay / bayan tahmini yerine insan / makine tahmini yapar. Testin detayları řu şekildedir. Öncelikle sorgulama yapan kiři karřısında bir insan ve bir makine olduđunu bilir. Test, bilgisayarın genel dünyada insan olarak kabul edilecek bir zeka göstermesini gerektirmez, testin amacı yalnızca řüpheli bir insanı kandırabilmektir. İkincil olarak test fiziksel özelliklerden bađımsızdır. Sorgulayıcı ve karřısındaki kiřiler arasındaki iletiřim tamamen yazılı olarak yapılır ve insan veya makine tarafından gelen yazı aynıdır (El yazısı vs. gibi bir yöntem izlendiđi için insandır denemez, yazı şekli tamamen aynıdır. Yazının içeriđi önemlidir.) Test bir problem çözme ya da sanat eseri üretme gibi komplike bir hedef içermez. Basitçe sorgulayıcı iki taraf ile de günlük basit bir konuřma yapar. Eđer bir insan, konuřtuđu bir makinayı bařka bir insan zannediyor ise Turing için bu makine düşünme yetkisi gösterecek kadar zekidir. (Saygin, Cicekli, & Akman, 2000)

Turing Testinin yapısı aynı zamanda birçok korkutucu senaryoya ev sahipliđi yapar. "Zeki Makine" testi geçen makine olarak tanımlanırken, insanlar tarafından sınılandıđını fark eden ve bilerek testi geçememeyi tercih eden bir yapay zeka da olasıdır. Bu durumda karar ařamasında tip 2 hata (True Negative) yapıldıđı söylenebilir ve testi geçemeyen makinenin zekâsının testin ötesinde olması durumu felsefi olarak sorgulanabilir.

2.2 Makine Öğrenmesi

Makine öğrenmesi nedir? Ya da bir diğer deyiş ile bilgisayarlar nasıl öğrenir? Her kavramda olduğu gibi makine öğrenmesinin tanımı hakkında da birçok görüş bulunmaktadır. Makine öğrenmesi, bu iş için kullanılan donanım ürünlerini üreten NVIDIA firması tarafından “Verileri ayrıştırmak, onlardan öğrenmek ve ardından dünyadaki bir durum hakkında bir belirleme veya tahmin yapmak için algoritmalar kullanma işlemi” olarak, Stanford Üniversitesi tarafından “Bilgisayarlara sabit kurallar ile programlanmadan karar alma yetkisi kazandırma bilimi”, Amerikan danışmanlık firması Mckinsey & Co tarafından “Kural tabanlı programlama olmadan veriden öğrenme sağlayan algoritmalar.”, Carnegie Mellon Üniversitesi tarafından ise “Tüm öğrenme işlemine etki eden yasaları araştıran ve kazandığı tecrübe ile otomatik olarak gelişen algoritmaları inceleyen bilgisayar bilimi” olarak tanımlanır. Bu tanımlardaki önemli noktaları birleştirerek “Makine Öğrenmesi, bilgisayarların insan gibi öğrenme ve davranmaları için, gözlem ve gerçek dünya etkileşimi verileri ile besleyerek ve özerk bir şekilde performanslarını zaman içinde geliştirmelerini sağlama bilimidir” denilebilir.

Makine öğrenmesi yedi adımdan oluşan bir işlem olarak açıklanabilir. Aslında kullanılan yöntem insanların araba kullanmayı öğrenmesine benzer. Öncelikle kontrollü bir ortamda (gerekirse hata yaparak) bir süre kişiye araba kullanma eğitimi verilir. Yeterli tecrübe kazanıldığında yeterlilik için sınav yapılır. Sınavı geçen öğrenci ehliyet almaya hak kazanır ve sonrasında daha fazla tecrübe kazandıkça araba kullanma yetisi artar. Bilgisayarlar için kullanılan yedi adım, bu çalışmada uygulandığı şekilde aşağıda açıklanmıştır:

1. Veri Toplama: Bilgisayarın öğrenebileceği verileri toplama işlemidir. Çalışma için Prof. Dr. Güray Küçükkocaoğlu tarafından sağlanan 2006 – 2016 arası yaşanmış BIST30 verileri kullanılmıştır.
2. Veri Hazırlama: Güncellik ve verilerin temizliği anlamında en uygun aralığın 2014-2016 aralığı olması sebebi ile bu aralık seçilmiştir. Finans verilerinde hazırlık aşaması karmaşık değildir. Fakat sınıflandırma amaçlı kullanılan algoritmalar için bu bölüm çok önemlidir.

Örnekleme özellikleri: Örneklemin adil seçilmiş olması önemlidir. Önündeki resimdeki kişinin kadın mı erkek mi olduğunu öğrenen bir algoritma verdiğiniz veri kümesi %90 kadın %10 erkek verilerinden oluşur ise doğal olarak bu kişinin yüksek ihtimalle kadın olduğunu düşünecektir. Bu bağlamda algoritmaya verilen iki örnekleme kümesini eşit tutmak önemlidir. Sıradaki adımda elimizdeki veri ikiye ayrılır:

Öğrenme Verisi: Öğrenme işlemi için bilgisayara verilen veri kümesi. Tez kapsamında öğrenme verisi yüzdeleri %80, %90, ve %99 olarak kullanılmıştır.

Değerlendirme Verisi: Öğrenme performansını ölçmek için bilgisayara daha önce verilmemiş değerlerden oluşan veri kümesi. Tez kapsamında değerlendirme verisi yüzdeleri %20, %10 ve %1 olarak kullanılmıştır.

3. Model Seçme: Elimizdeki örneklerden hangi verileri alıp kullanacağımıza karar verilen bölümdür. Erkek / kadın sınıflandırması örneğinde adam elması, sakal, bıyık varlığı gibi özellikler model için önemli olabilir. Finansal zaman serilerinde ise kullanılan model regresyon algoritmasıdır. Bu çalışmada kullanılan modeller doğrusal regresyon, Bayes doğrusal regresyon, karar ağacı regresyonu, destekli karar ormanı regresyonu, hızlı orman yüzdellik regresyonu, nöral ağ regresyonu ve Poisson regresyonudur.
4. Öğretme: Bilgisayarın önündeki örneğin ne olduğunu öğrendiği adımdır. Erkek / kadın örneğinde cinsiyet, finansal zaman serilerinde ise fiyatlar öğretilir. Bilgisayar öğrenebilmek için elindeki verileri kullanarak bir karar doğrusu / eğrisi (modele göre) yaratır. Her örnek için bu doğru / eğriyi günceller. Sonrasında bu test verileri matris şeklinde kullanılarak ağırlık (weight) ve kayma (bias) değerleri üretilir. Erkek / kadın örneğinde saç uzunluğu verisinin etki ağırlığı %10, sakal varlığı verisinin ağırlığı %90 olabilir. Finansal zaman serilerinde ise spesifik bir hisse tahminlerimiz için kalan hisselerden daha önemli olabilir. Belirtilen ağırlık ve kayma değerleri başlangıçta rasgele seçilir. Her yeni örnek ile ağırlık ve kayma değerleri güncellenir. Bu işleme “öğrenme döngüsü” denir. Bu döngü ne kadar fazla yapılırsa o kadar fazla veride öğrenme işlemi gerçekleştirilir ve performans o kadar yükselir. Bu durum bir doyum noktasına ulaşıncaya kadar sürer. Doyum noktasına ulaştıktan sonra ise aşırı öğrenme “over-fitting” denilen durum ortaya çıkar, ve algoritma olmayan kalıp ve özellikler bulmaya zorlanacağı için performans düşer.

5. Deęerlendirme

Öęretme iřlemi bittikten sonra algoritmanın performansı deęerlendirme verileri ile test edilir. Deęerlendirme bölümünde kullanılan veriler öęretme bölümünde kullanılmamıř olmalıdır. 2. Adımda ayrılan veriler bu bölümde kullanılır.

6. Parametre Ayarlaması

Deęerlendirme bölümünde elde ettięimiz sonu, tamamen model semede kullandıęımız parametrelere baęlıdır. Deęerlendirme evresinden sonra istenilen performans seviyelerine ulařmak için parametrelerde ayarlama yapmak gerekebilir.

7. Tahmin (Prediction)

Tüm eęitim adımları tamamlandıktan sonra bilgisayar önüne gelen gerek dünya örnekleri üzerinde tahmin yapmaya hazırdır.

Bilgisayarlarda öęrenme, gözetimli öęrenme (supervised learning) ve gözetimsiz öęrenme (unsupervised learning) olarak ikiye ayrılır. Gözetimli öęrenmede elinizdeki verilerin hangi ıktıyı verdięini bilgisi belirtilir. Örnek olarak fotoęrafta araba olan resimleri bulmaya alıřan bir algoritmaya verdięiniz öęrenme verisinde hangi fotoęraflarda araba var, hangilerinde yok bilgisi iřlemiř ise bu gözetimli öęrenme olur. Gözetimsiz öęrenmede ise eldeki verilerin özelliklerine göre demetleme (clustering) iřlemi yapılır ve belirli gruplar oluřturulur. Bu yöntem ile aynı örnek düşünülürse bilgisayar hangi grupta araba fotoęrafı olduęunu bilmeyecektir, fakat siz ona bir resim verdięinizde ona benzeyen (aynı kümede bulunan) bir fotoęraf sunabilir. (Alpaydin, 2020) Finans alanının sunduęu veri seti ve hedeflenen yatırım kriterleri gözetimsiz öęrenme yöntemleri için daha uygundur. Veri seti analizlerinde zaman serileri algoritmalara verilecek ve ıkarımların otomatik yapılması beklenecektir.

Makine öęrenmesi için kullanılan modeller, belirli mantık ve olasılık teoremleri üzerine kuruludur. Bu sebeple sıradaki bölümde gerekli olan mantık ve olasılık teoremleri incelenmiřtir.

2.3 Mantıksal Cebir

Yapısı alçak gerilim ve yüksek gerilim sinyallerine dayanan bilgisayarlarda mantık, sinyalin varlığı ve yokluğuna dayanır. Mantıksal cebir ya da boolean matematiği, sayısal devrelerin çıkış ifadelerinin giriş değişkenleri cinsinden ifade edilmesi ve elde edilen ifadenin en basit haline ulaşması için kullanılır.

Mantıksal cebirde kullanılan değişkenler veya fonksiyonlar sayısal iki farklı değer alabilir. Bu değerler 1 veya 0 olacaktır. Değişkenlerin veya fonksiyonların aldığı bu değerler sayısal devrelerde eğer “1” ise yüksek gerilim seviyesi, “0” ise alçak gerilim seviyesini gösterecektir. Bilgisayarların temel çalışma mantığı bu “0” ve “1” ler temeline dayanır. Yapay zekanın temeli de mantık ve olasılık hesaplarına dayandığı için farklı mantık türleri incelenmiştir.

Mantıksal cebir 3 kanun üzerine inşa edilmiştir. Bu kanunlar; yer değiştirme kanunu, birleşme kanunu ve dağılma kanunudur. Bu özellikler “De Morgan” kanunları olarak ta bilinir. (Cornish & Fowler, 1977)

2.3.1 Önermeli Mantık (Propositional Logic)

Önermeli mantık, matematik ve matematiksel mantığın ilk konusu sayılabilir. Önermeli mantık olmadan ileri düzey matematiksel mantık mümkün olamaz. Felsefi mantık çalışmalarında da karşılaşılan önermeli mantığın tarihi, M.Ö. 384 yılında Aristotle’nin çalışmalarına dayanır. Aristotle’nin meta fizik çalışmalarında bugün yasa kabul edilen “üçüncü halin imkansızlığı yasası” ve “çelişki yasası” görülmektedir. Bu iki yasa “her önerme doğru ya da yanlış olmak zorundadır” ve “hiçbir önerme aynı anda hem doğru hem yanlış olamaz” olarak tanımlanabilir. (Rescher, 1964) Önermeli mantık ile ilgili en önemli çalışmalar ise Augustus DeMorgan ve George Boole’nin 19. Yüzyıldaki çalışmalarıdır. Bilgisayarlar için çok önemli olan, matematiksel olarak mantıkta doğruluğun “1” yanlışlığın ise “0” olarak tanımlanması ve konuya aynı zamanda “Boolean Matematiği” denmesinin George Boole’nin çalışmalarından kaynaklanırken, yine matematikte sıkça karşımıza çıkan ve mantıkta da uygulanan birleşme ve dağılma kanunu gibi kanunlar DeMorgan’ın ürünüdür. (DeMorgan, 1847) (Burris, 2019)

Önerme, belli bir durumda doğru veya yanlış denebilen cümledir. Matematik, mantık ve felsefede sıkça kullanılır. Örneğin, ‘Her sayının tersi vardır’ cümlesi, bir önermedir ve bu önerme,

- Pozitif doğal sayılar (N^+) yapısında doğru,
- Tam sayılar (Z) yapısında yanlış,
- Doğal sayılar (N) yapısında yanlış,
- Negatif irrasyonel sayılar (Q^-) yapısında doğrudur.

Doğru ve yanlış, doğruluk değerleridir. Doğru doğruluk değeri 1; yanlış doğruluk değeri de 0 olarak gösterilir.

2.3.2 Birinci Derece Mantık (First Order Logic)

Birincil derece mantık, bilgisayar bilimlerinin de aralarında bulunduğu, başta felsefe olmak üzere, matematik ve dilbilim gibi farklı alanlarda kullanılan bir mantık modelidir. Birinci derece mantık, mantıksal olmayan nesnelere üzerinde niceliksel değişkenler kullanır. Birinci derece mantık ve önermeli mantık arasındaki temel fark birinci derece mantığın niceleyici ve ilişkiler kullanmasıdır. Buradaki fikir, daha kompakt bilgi sunumları sağlamak için değişkenleri kullanmaktır. (Sowa, 2000)

Birinci derece mantığın, mantık alanında kendi başına bir alan olması, Pierce’nin 1885’teki çalışmalarına dayanır. Fakat Hilbert’in 1917 yılındaki dersleri, birinci derece mantığı yüksek dereceli mantıklardan ayırması sebebi ile kılavuz kabul edilir. (Ewald, 2019) Birinci dereceden mantık yalnızca bireyler arasında değişen değişkenleri kullanırken; ikinci ve daha üst dereceden mantık bu değişkenlerin yanı sıra bireyler arasında değişen ek değişkenlere de sahiptir.

Birinci derece mantık, skolastik mantıktan geliştiği için aynı dört sesli harf ile gösterilen dört basit önerme türünü içerir:

- A: Evrensel Doğruluk: “Her x bir y’dir.”
- I: Özel Doğrulayıcı: “En az bir x, y’dir.”
- E: Evrensel Yanlışlık: “Hiçbir x y değildir.”
- O: Özel Yanlışlayıcı: “En az bir x y değildir.”

Bu özellikleri kullanarak Aristotle’nin örneklerinden bir çıkarım:

- Tüm geniş yapraklı bitkiler yaprak döker
- Tüm asma bitkileri geniş yapraklıdır
- Sonuç olarak tüm asma bitkileri yaprak döker

Kökene binlerce yıl öncesine dayanan bu skolastik mantığın günümüz matematik ve mantığında kullanılan modern sembollere dönüşmesi ise İtalyan matematisyen Giuseppe Peano’nun çalışmalarına dayanır. (Sowa, 2000) Belirtilen semboller kısaltmalar bölümünde açıklanmıştır.

2.3.3 Bulanık Mantık (Fuzzy Logic)

Bulanık mantık 1961 yılında Dr Lofti Zadeh tarafından California üniversitesinde öne sürülmüştür. Dillerin bilgisayar tarafından işlenmesini üzerinde çalışan Zadeh, doğadaki birçok şey gibi dil mantığının da bilgisayarın kolaylıkla anlayabileceği 0 ve 1’lere çevrilemeyeceğini görerek önermeli mantığa karşı bulanık mantığı öne sürmüştür.

Önermeli mantık her mantıksal hesaplamayı 0 veya 1 olarak sonuçlandırır. Bulanık mantık ise 0 ve 1’i uç nokta doğruluk olarak görür ve aradaki değerleri de kabul eder. Klasik mantık “Su soğuk mu” sorusuna 0 veya 1 ile cevap verdiğinde suyun ılık olma olasılığı yoktur. Donmak üzere olan su ile hafif soğuk olan su arasında da fark yoktur. Bulanık mantık ise suyun 0.38 değerinde soğuk olduğunu öne sürebilir. (Zadeh, 1996)

Bulanık mantık kullanımı bu tezin kapsamı için önemlidir. İncelediğimiz hisse senetleri arasında korelasyon var ya da yok demek faydalı olmayacaktır çünkü varlığın seviyesi önemlidir.

2.4 Olasılık Kuramı

Mantıklı karar alma yönündeki çalışmalar başarılı olma olasılığını maksimize etme işlemine dayanır. Bu sebep ile yapay zeka tüm öngörülerini için olasılık hesaplamaları kullanır ve aynı bir insan gibi gerçekleşme olasılığı en yüksek olan hamleleri seçme eğilimindedir.

Olasılık teoreminin çıkışı aslında birçok farklı bilim dalının gözlem verilerini işleme üzerine çalışmalarına dayanır. Veri işleme ihtiyacı gündelik problemler ve kaynak planlaması öngörülerinde sıkça karşımıza çıkar. Bu ihtiyaç milat öncesi çalışmalarda bile görülmektedir. Roma imparatorluğunda Augustus fethettiği her yeni bölgedeki buğday, su gibi ihtiyaçları öngörebilmek için yaptığı istatistiksel çalışmalar istatistiksel verinin ilk örnekleri sayılmaktadır. 15. Yüzyıla kadar farklı imparatorlukların ürün yetiştirme ve vergi çalışmalarında istatistiksel veriler kullanılmıştır. 16. Yüzyılda Hollanda, İspanya, Fransa, İngiltere ve Almanya ülkeleri, her hastalık için hastalıktan ölen ve iyileşen kişi verilerini saklamaya ve bu verileri kullanarak hastalıkların “ölüm yüzdelerini” hesaplamaya başlaması ile “hastalıktan ölme olasılığı” hesaplaması, istatistikten olasılığa geçiş olarak betimlenir. (Masitrov, 2014) Bu bağlamda olasılık ve istatistik kavramlarının birbirlerine çok benzediği görülmektedir. İstatistik alanı veri ve veri ile ilgili çıkarımlar ile ilgilenirken olasılık, veri veya sonuçların arkasında yatan süreçle ilgilidir.

Matematiksel bir teorem olarak olasılık teoreminin çıkışı 16. Yüzyılda Gerolamo Carano'nun “Şans Oyunları Üzerine Bir Kitap” isimli çalışmasına dayanır. Çalışmada olasılık, olumlu sonuç sayısının olası sonuç sayısına bölümü olarak tanımlanmıştır. Bağımsız olasılıklı gözlemler durumunda olasılıkların çarpımı gibi özelliklerden de bahsetmiştir. 20. Yüzyılın başlarında Kolmogorov'un çalışmalarında “örnek uzay” kavramını tanımlaması ile modern olasılık kuramları bugün bilinen halini almıştır. (Stordahl, 2007)

Olasılık teorisinin amacı matematiksel olarak belirsizliğin ölçümünü yapabilmektir. Fiziksel ve sosyal bir olgunun kesin olarak belirlenmesi olanaksız da olsa, bu tür olgular yeterince gözlemlendiklerinde belirli bir düzenleri oldukları saptanabilir. Matematiksel olarak bakıldığında deneme sayısı limiti sonsuza gittiğinde bu düzen açıkça görülür. Basitçe ifade edersek olasılık, rastlantısal bir olguya ilişkin bir önermenin kesine ya da olanaksıza ne kadar

yakın olduğunu gösteren bir sayıdır. Olasılık 0 ile 1 arasında bir değer alır ve ‘‘0’’ olanaksızlığı ‘‘1’’ ise kesinliği simgeler.

Olasılık teoremini incelemek için birkaç temel kavramı açıklamak gerekir. Öncelikle Rastlantısal deney ya da kısaca deney, sonucu kesin olarak bilinmeyen olgulara ilişkin gözlem yapma ya da veri toplama süreci olarak tanımlanabilir. Örneğin hilesiz bir zar 2 kez atılırsa kaç kez 6 geleceğini veya bir üreticiden sipariş edilen ürünlerin hata yüzdesini bulmak için 50 ürün sipariş ettiğimizde kaç tanesinin defolu geleceğini bilemeyiz. Öyleyse zar 2 kez atılıp, kaç kez 6 geldiği sayıldığında ya da 50 ürün kontrol edildiğinde birer rastlantısal deney yapılmış olur. Rastlantısal deney rastlantısal denemelerin bütünüdür.

Bir rastlantısal deneyde gerçekleşebilecek tüm mümkün farklı sonuçların oluşturduğu küme örnek uzay olarak adlandırılır. Örneğin rastlantısal deney hilesiz bir zarın bir kez atılması ise, deney 6 farklı biçimde sonuçlanabileceği için örnek uzay $S = \{1,2,3,4,5,6\}$ olacaktır. Zar iki kez atılırsa, bu deney 36 farklı şekilde sonuçlanabilir:

	1	2	3	4	5	6
1	(1,1)	(1,2)	(1,3)	(1,4)	(1,5)	(1,6)
2	(2,1)	(2,2)	(2,3)	(2,4)	(2,5)	(2,6)
3	(3,1)	(3,2)	(3,3)	(3,4)	(3,5)	(3,6)
4	(4,1)	(4,2)	(4,3)	(4,4)	(4,5)	(4,6)
5	(5,1)	(5,2)	(5,3)	(5,4)	(5,5)	(5,6)
6	(6,1)	(6,2)	(6,3)	(6,4)	(6,5)	(6,6)

Şekil 1: 2 Zar Örnek Uzayı

Bu 36 sonuç iki zarlı deneyin örnek uzayıdır. Matematiksel olarak bir olayın gerçekleşme olasılığı aşağıdaki şekilde gösterilebilir:

$$P(A) = \frac{n(A)}{n}$$

Denklemde;

$P(A)$ = “A” Olayının gerçekleşme olasılığı

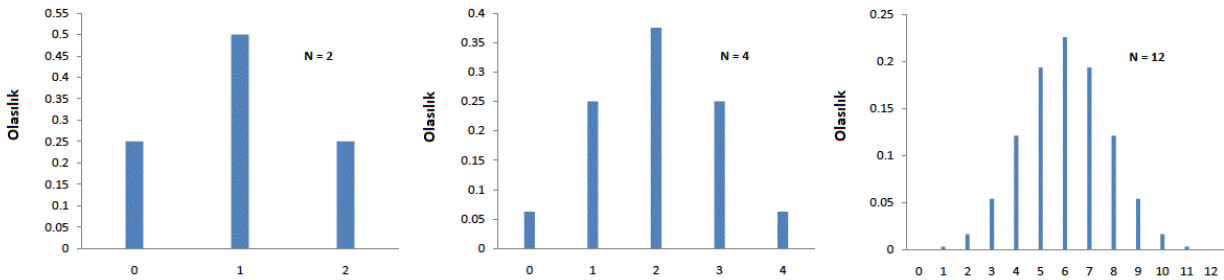
$n(A)$ = İstenen sonuç sayısı

n = Örnek Uzay’da bulunan sonuç sayısı olarak tanımlanır.

Finansal alanından bakıldığında örnek uzay zarlarda gördüğümüz gibi yüzey sayısı ile kısıtlanabilir bir yapıda değildir. Piyasa fiyatları küsuratlı değerler alabilir ve teorik olarak piyasada işlem gören menkul kıymetlerin taban ve tavan fiyatları yoktur, arz – talep dengesine göre fiyat belirlenir. Bu sebeple kesin bir örnek uzaydan bahsetmek mümkün olmasa da örnek uzay olasılık dağılımlarınca indirgenebilir.

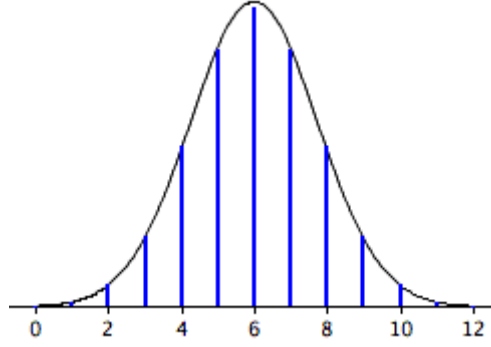
2.4.1 Normal Dağılım

İstatistik ve olasılığın en önemli dağılımlarından biri olan normal dağılım, olarak 1733'te Abraham de Moivre tarafından “Hilesiz bir para 100 defa atılırsa 60 veya daha fazla yazı gelme olasılığı nedir?” probleminin çözümünde ortaya çıkmıştır. 2,4, ve 12 zar için yapılan binom dağılımları aşağıda gösterilmiştir:



Grafik 1: 2, 4, 12 Zar Binom Dağılımları

12 Zar için yapılan binom dağılımları tepe noktaları üzerine çizilen eğri, bugün “çan eğrisi” olarak da bilinen normal dağılımı gösterir.



Grafik 2: Normal Dağılım

Aynı dağılım, “merkezi limit teoremi isimli” çalışmasında Laplace tarafından da ortaya konmuştur. Bu sebeple Moivre-Laplace teoremi olarak da isimlendirilir. (Patel & Read, 1996)

2.4.2 Koşullu Olasılık (Bayes Teoremi)

Aslında 1740 yılında rahip Thomas Bayes tarafından bulunan bu teorem, Laplace tarafından matematiksel bir model haline getirildikten sonra ünlenmiştir. Hilesiz bir zar atıldığında 6 gelme olasılığı Gauss mantığı ve normal dağılım ile bakıldığında $1/6$ 'dır. Zar ne zaman ne koşulda nerede atılırsa atılsın bu olasılık değişmeyecektir. Öte yandan bu olasılığın anlamı ‘ortalama 6 zar atışından birinde zarın 6 numaralı yüzeyi yukarıda kalmalıdır’ demektir. Bu ortalama değer doğruluğu limit sonsuza gittiğinde kabul edilmektedir. Aynı soru şu ek bilgi ile verildiğinde ise Gauss mantığı bir farklılık görmez iken bu statik mantığın hatası ortaya çıkar:

‘Bundan önce atılan 5 zar 6 geldiği bilindiğine göre sırada atılacak olan zarın 6 gelme olasılığı nedir? Şüphesiz ki arka arkaya 6 kere 6 atma ve bir kere 6 atma olasılıkları aynı değildir. Eğer 6 gelme olasılığı $1/6$ ise 6 zarda ortalama 1 kere 6 gelmesini bekleriz. Eğer önceki zarların tümü 6 geldi ise normal dağılımın uç noktasında bir bölgedeyiz ve bu sebeple 6 haricindeki herhangi bir zarın gelme olasılığı çok daha yüksektir. Önceki olayların sonuçları (ortalama değerimiz sabit kaldığı için) gelecekteki olayların olasılıklarını etkiler. Matematiksel olarak teorem aşağıdaki şekilde gösterilebilir, az önce anlatılan örnekteki

veriler formüle uygulandığında belirtilen durumdaki olasılığın 1/6 değil 1/6⁷ olduğu görülmektedir.

Denklem 2: Bayes Teoremi / Koşullu Olasılık

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

Denklemde:

P(A|B): “A” Olayı gerçekleştiği bilindiği durumda “B” Olayı gerçekleşme olasılığı

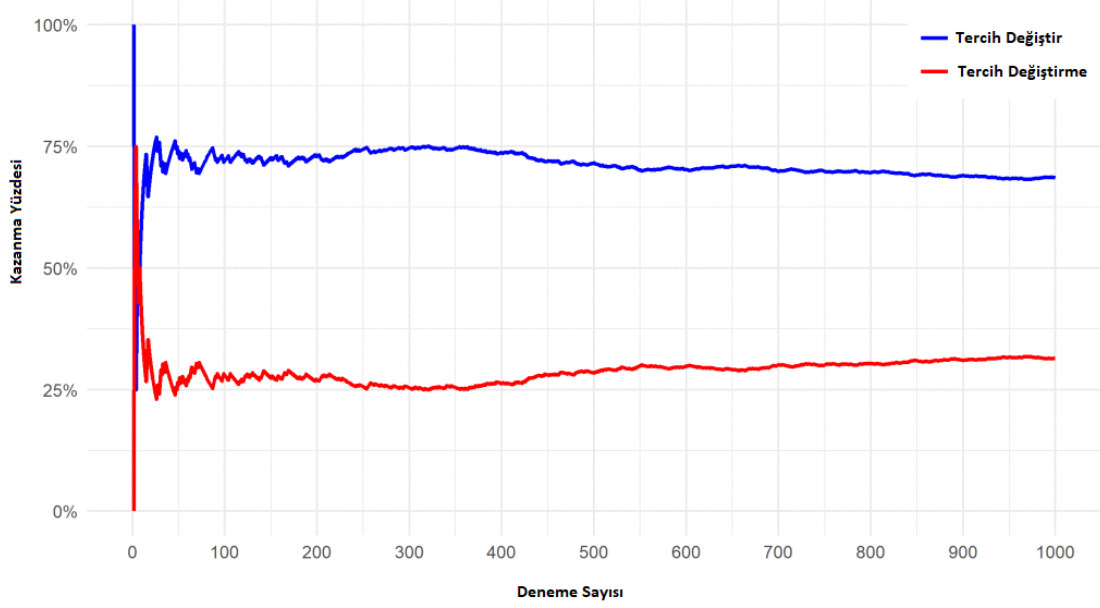
P(A∩B): “A” ve “B” olaylarının birlikte gerçekleşme olasılığı

P(A): “A” olayı bağımsız gerçekleşme olasılığı

P(B): “B” olayı bağımsız gerçekleşme olasılığı olarak tanımlanır.

Bu durum “Monty Hall” problemi ile ünlenmiştir. Monty Hall probleminde ikisinin arkası boş, birinin arkasında bir ödül olan 3 kapının bulunduğu bir yarışma programı incelenir. Yarışmacı A, B veya C kapılarından birini arkasında ödül olduğu inancı ile seçer. Hangi kapının ardında ödül olduğunu bilen sunucu, kapılardan birini açar ve arkasının boş olduğunu yarışmacıya gösterir. Sonrasında yarışmacıya bir seçenek sunar, yarışmacı tercihini değiştirebilir veya tercihini sabit tutabilir. Gauss mantığı ile iki kapıdan birinde ödül var olarak düşünüldüğünde bu seçeneğin sağladığı olasılıklar iki taraf için de %50 görünmektedir. Başlangıçta üç kapı olduğu düşünülürse bu mantık doğru değildir.

Belirtilen problem, iki seçeneği de 1000 defa deneyen bir bilgisayar simülasyonu tasarlanıp eşit sayıda tercih değiştirme ve tercihini sabit tutma seçimi yaptığında ise ortaya çıkan sonuç olasılığın iki taraf için de %50 olmadığını, 1000 denemenin 685’inde tercih değiştiren, 315’inde ise tercihi sabit tutan tarafın kazandığı görülmektedir. Bu sonuçlar Bayes teoreminin bu koşullar için ortaya koyduğu %66-%33 dağılım beklentisini desteklemektedir. (Delgrande & John, 1986)



Grafik 3: Monty Hall Problemi Bayes Simülasyonu

Tezin amacı olan finansal zaman serisi öngörülerinde Bayes teoremi büyük önem taşımaktadır. Finansal zaman serileri devamlılık gösteren değerler içermektedir, yani “dün bu fiyat olduğu bilindiği üzere yarın ne fiyat olacaktır” şeklinde bir durum olduğu için birçok regresyon yönteminde Bayes olasılık teoremi kullanılır.

2.4.3 Poisson Dağılımı

Bu dağılım ilk defa Siméon-Denis Poisson (1781–1840) tarafından diğer olasılık hakkındaki yazıları ile birlikte 1838'de yayınlanan *Recherches sur la probabilité des jugements en matières criminelles et matière civile* ("Ceza hukuku ve medeni hukuk alanlarındaki hükümlerin olasılığı üzerinde araştırmalar") adındaki eserinde ortaya atılmıştır.

Bağımlı değişkenin 0, 1, 2, 3, ... gibi kesikli bir değer aldığı fakat kategorik olmadığı durumlar vardır. Bu tür değişkenlere, doğalgaz boruları üzerinde kazaların sayısı, verilen patentlerin sayısı, yazlıklarda çıkan yangınların sayısı gibi örnekler gösterilebilir. Kesikli ve kategorik olmayan, nadir olaylarla ilişkili bağımlı değişkenli model, bazı varsayımlar altında Poisson regresyon modeli olarak adlandırılır. Poisson regresyon modeli daha çok sayma verilerini analiz etmek için kullanılmaktadır. (Akın, 2002)

Poisson regresyon modelinde regresyon sürecindeki genel kestirimler en çok olabilirlik yöntemi ile gerçekleştirilmektedir. Poisson en çok olabilirlik kestirimi için;

1) Koşullu ortalamanın doğru tanımlanmasında bağımlılık şartı sağlanmalıdır. Ayrıca bağımlı değişken y 'nin Poisson dağılması gereklidir.

2) En çok olabilirlik standart hataları ve t istatistikleri kullanarak hesaplanan istatistiksel sonuçlar hem koşullu ortalama hem de varyansın doğru tanımlanmasını gerektirmektedir. Burada istenen koşul, koşullu varyans ve ortalamanın eşit olmasıdır.

3) Veriler için koşullu varyans ve koşullu ortalamanın eşit olmaması durumunda, en çok olabilirlik yönteminin uygulanması ile elde edilmiş istatistiksel sonuçlar, koşullu ortalamanın doğru tanımlandığının ispat edildiği durumlarda geçerli ve doğrudur.

4) Veriler için koşullu varyans ve ortalamanın eşit olmaması durumunda, Poisson en çok olabilirlik tahmin edicisinden daha etkin tahmin ediciler kullanılabilir. (Deniz, 2005)

Poisson dağılımının genel odaklandığı rassal değişken, sayılabilen bir olaydır; bu olay belli bir sabit uzunlukta olan (genellikle zaman) aralıkta ayrık olarak ortaya çıkar ve bu aralıkta gözlenen olayların sayısı Poisson dağılım için rassal değişkendir. Bu sabit aralıkta

ortaya çıkan olaylar sayısının beklenen değeri (ortaya çıkmanın ortalama sayısı) λ olarak sabittir ve bu ortalama değer aralık uzunluğuna orantılıdır. Örnek olarak bir restorana her yarım saatte 10 müşteri geliyorsa bir saate 20 müşteri gelmesi beklenir. “k” sayıda (k = 0, 1, 2, 3...) olay ortaya çıkma olasılığı aşağıdaki denklem ile ifade edilir:

Denklem 3: Poisson Olasılık Kütle Denklemi

$$f(k, \lambda) = \frac{\lambda^k e^{-\lambda}}{k!}$$

Olasılık kütle fonksiyonu olarak bilinen bu denklemde:

e: doğal logaritmanın tabanı (e = 2.71828...)

k: olasılığı fonksiyon ile verilmekte olan olayın ortaya çıkma sayısı;

k!: k faktöriyel

λ : verilen sabit aralıkta ortaya çıkma sayısının beklenen değeri olarak tanımlanır.

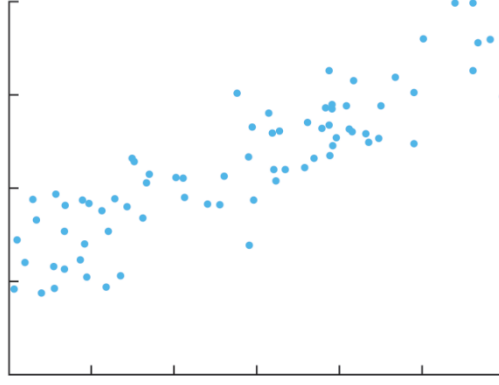
BÖLÜM III. YAPAY ZEKA REGRESYON ALGORİTMALARI

Bu bölümde yapılan çalışmada uygulanan farklı regresyon algoritmaları açıklanmıştır. Öncelikle regresyon analizinin ne olduğu açıklanmış, sonrasında ise seçilen algoritmalar kullandıkları mantık ve matematiksel formüller sunulmuştur. Bir önceki bölümde açıklanan olasılık kuramları hangi algoritma için kullanılıyor belirtilmiştir.

3.1 Regresyon Analizi

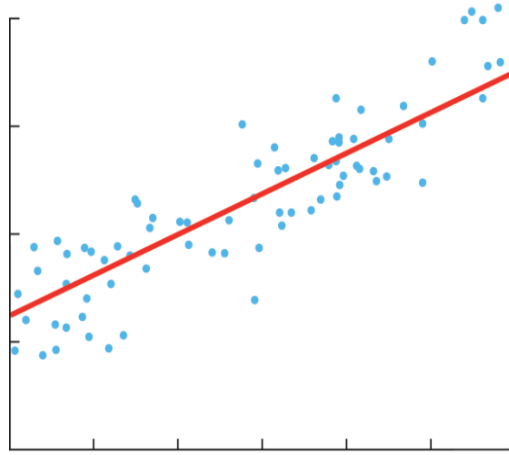
İstatiksel regresyon analizi kavramı terminolojiye matematisyen Adrien-Marie Legendre tarafından kazandırılmış olsa da resmi ismini kazanmadan önce Carl Friedrich Gauss tarafından regresyon yöntemini kullanan çalışmalar görülmektedir. İki matematisyen tarafından da regresyon yöntemlerinde bağımlı değişkenin yüzde ne kadarının bağımsız değişken tarafından açıklandığını göstermek için kullanılan determinasyon katsayısı, “en küçük kareler” yöntemi ismiyle farklı alanlarda öne sürmüştür. Gauss bu yöntemi o kadar mantıklı ve basit görmüştür ki daha önce kesinlikle birisi tarafından bulunduğu düşüncesi ile bulunduğu bu metodu resmi olarak açıklamamıştır. Gauss için önemsiz sayılabilecek bu yöntem modern istatistik ve veri bilimi çalışmalarının merkezindedir. Bugün bildiğimiz hali ile regresyon ise genel olarak R.A. Fisher’in çalışmalarına dayanır. Lojistik regresyon, bayes regresyonu ve parametrik olmayan regresyon gibi yöntemler Fisher’in ürünüdür. (Freund, Wilson, & Sa, 2006)

Regresyon analizi çok bilinmeyenli verilerin analizinde sıkça kullanılan kuvvetli bir analiz yöntemidir. Matematiksel olarak sonuca etki eden faktörleri öne çıkarmak için kullanılır. Hangi faktörler önemlidir, hangi faktörler göz ardı edilebilir gibi sorulara cevap sunar. Regresyon analizinde bu faktörlere değişkenler denir. Anlamaya çalışılan veya tahmin edilen değişkene bağımlı değişken, bu faktöre etki ettiği düşündüğümüz değişkene ise bağımsız değişken denir. Bağımlı ve bağımsız değişkenler Kartezyen sistemin x ve y bileşenleri olarak gösterildiğinde aşağıdaki gibi dağılım grafikleri ortaya çıkar:



Grafik 4: Örnek Dağılım Grafiği

Regresyon modelleri bu veriler arasındaki ilişkiyi açıklayan bir matematiksel formül üretir. Örnek olarak belirtilen iki değişken arasındaki ilişkiyi açıklayan doğrusal bir model aşağıda gösterilmiştir. (Draper & Smith, 1998)



Grafik 5: Örnek Regresyon Grafiği

Yalnızca bir bağımsız değişkeni inceleyen modellere basit regresyon, çok sayıda bağımsız değişken içeren modellere ise çoklu regresyon denmektedir. Tez çalışması için BIST 30’da bulunan 30 firma verileri kullanılmıştır. AKBNK bağımlı değişken olarak seçilmiş; AKSA, ARCLK, ASYAB, BIMAS, DOHOL, EKGYO, ENKAI, EREGL, GARAN, HALKB, IHLAS, ISCTR, KCHOL, KOZAL, KRDM, MGROS, PETKM, SAHOL, SISE, SNGYO, TCELL, THYAO, TKFEN, TOASO, TTKOM, TTRAK, TUPRS, VAKBN, YKBNK hisseleri ise bağımsız değişkenler olarak kullanılmıştır. Kullanılan yedi farklı regresyon algoritması bu bölümde açıklanmıştır.

3.1.1 Doğrusal Regresyon (Linear Regression)

Doğrusal regresyon modeli, bağımlı değişkeninin (y), bir dizi yordayıcı değişken ile (x) çarpılan ağırlıkların (β) doğrusal bir birleşimi olduğunu varsayar. Formül aynı zamanda tesadüfi örnekleme gürültüsünü hesaba katan bir hata terimi (ε) içerir. Örnek olarak iki bağımlı değişken içeren bir sistem için model aşağıdaki gibidir:

Denklem 4: 2 Bağımlı Değişken Doğrusal Regresyon

$$y = \beta_0 + \beta_1 * x_1 + \beta_2 * x_2 + \varepsilon$$

Doğrusal regresyon modeli matris denklemlerini kullanarak herhangi bir sayıda bağımsız değişken için ifade edilebilir. (Draper & Smith, 1998)

Denklem 5: Doğrusal Regresyon Matris

$$y = \beta^T X + \varepsilon$$

Doğrusal regresyon anlaşılabilirliği ve finans alanında sıkça kullanılması sebebi ile kullanılacak ilk algoritma olarak seçilmiştir. Normal dağılımlı veriler için uygun olan doğrusal regresyonun ortalama değerler üzerine kurulu olduğu için yüksek ve alçak değerlerin birbirini dengelediği uzun dönem tahminlerinde etkinliği referans gösterilen çalışmalarda gözlemlenmiştir. (Heshmaty & Kandel, 1985) (Goia, May, & Fusai, 2009) (Altay & Satman, 2005)

3.1.2 Bayes Doğrusal Regresyon (Bayes Linear Regression)

Bayes doğrusal regresyon, olasılık teoremleri bölümünde açıklanan koşullu olasılık teoreminin doğrusal regresyon modeline uygulanması ile elde edilen bir regresyon modelidir. İncelediğimiz zaman serileri, dinamik veriler olduğu için koşullu olasılık teoremini içeren bu modelin doğrusal regresyondan üstün performans göstermesi beklenmektedir. İncelediğimiz durumu “yarın piyasa fiyatları ne olacak?” olarak değil, “bugüne kadar ki fiyat bilgileri bunlar olduğuna göre yarın piyasa fiyatları ne olacak?” düşünmek daha doğru olacağı için bu beklenti oluşmuştur.

Bayes bakış açısında az önceki modelde ($y=$) şeklinde tanımlanan nokta tahminlerinden ziyade, olasılık dağılımlarını kullanarak doğrusal regresyon formüle edilir. Yani y 'nin, tek bir değer olarak tahmin edilmediği, ancak olasılık dağılımından alındığı varsayılır.

Denklem 6: Bayes Doğrusal Regresyon Matrisi

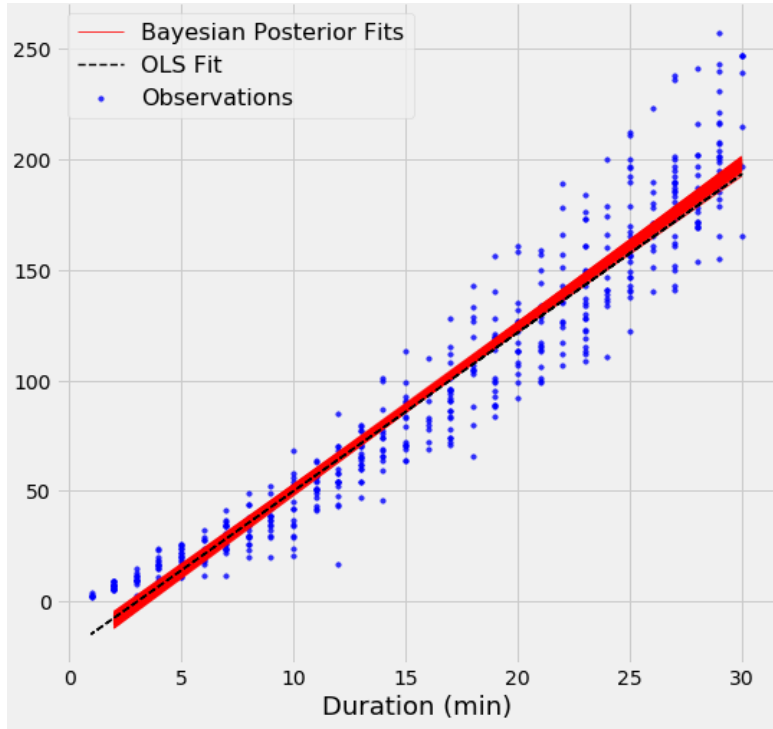
$$y \sim N(\beta^T X, \sigma^2 I)$$

Bayes Doğrusal Regresyonun amacı, bağımlı değişken için tek bir “en iyi” değer bulmak değil, bağımlı değişkeni değerinin olasılık dağılımını belirlemektir. Sonuç bir olasılık dağılımı belirtmekle birlikte model parametreleri de bir dağılım olarak girdi alınır. Bu dağılımlı model aşağıdaki şekilde gösterilebilir: (Yan & Gang Su, 2009)

Denklem 7: Bayes Doğrusal Regresyon Olasılık Dağılımı

$$P(\beta|y, X) = \frac{P(y|\beta, X) * P(\beta|X)}{P(y|X)}$$

Kullanılan model bir doğru değil dağılım olduğu için, çıktısı da bir dağılım grafiği olur:



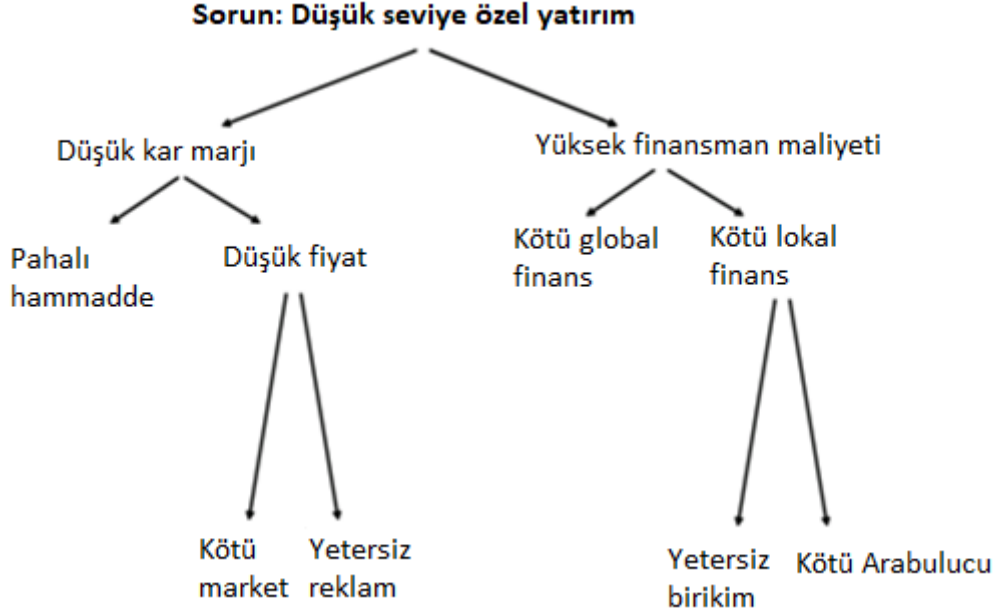
Grafik 6: Bayes Doğrusal Regresyon Örnek Çıktı

Yabancı kaynaklarda “Bayesian Multivariate Time Series Methods for Empirical Macroeconomics” (Koop & Korobilis, 2010), “Applied Bayesian Forecasting and Time Series Analysis” (Pole, West, & Harrison, 1994), “Bayesian Econometrics” (Koop G. , 2003) gibi detaylı kitaplar ve “Bayesian Threshold Autoregressive Models for Nonlinear Time Series” gibi aydınlatıcı uygulamalı makaleler bulunmasına karşılık, Bayes doğrusal regresyon modelinin zaman serileri üzerinde kullanımı hakkında Türkçe kaynak noksanlığı göze çarpmaktadır. “Regresyon Analizinde Bayesçi Yaklaşım” (Şıklar, 2009) gibi çalışmalar Bayes yaklaşımının regresyon modeli için kullanımını anlatsa da uygulamalı örnek yoktur ve bu çalışmanın da tüm referans kaynakları yabancı kaynaklardır.

3.1.3 Karar Ağacı Regresyonu (Decision Tree Regression)

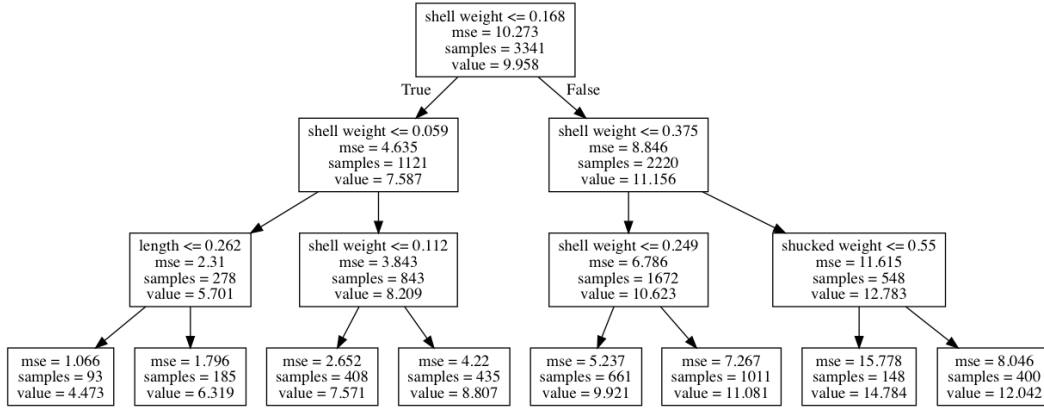
Tezin motivasyon bölümünde belirtilen ve bu teze ilham sağlayan satranç algoritmaları, satrançtaki her hamle sonrası oluşan tahtayı bir boğum olarak ifade ederek strateji üretme amaçlı karar ağaçları üretir. Oyunda gerçekleşebilecek olası hamle sayısı Shannon serisi'nce üstsel olarak arttığı için tüm olası hamleleri birer boğum olarak ifade etmek mümkün değildir. Stratejik olarak mantıklı boğumlar seçilir ve bu boğumların alt dalları olan boğumlar mantıklı hamleler olacağı için işleme gücü ve hafıza gibi kaynaklar yüksek yoğunlukla bu alt boğumlar üzerinde kullanılır. Aynı mantık finans alanında regresyona etki eden faktörlerin önemli boğumlar olarak seçildiği bir sistemle uygulanabilir düşüncesi ile karar ağacı regresyon algoritmaları bu çalışmada test edilmiştir.

Karar ağacı, tahmin yapmak için belirli özelliklerden karar verdirici çıkarımları yapan gözetimsiz bir makine öğrenmesi yöntemidir. Modelin adından da anlaşıldığı gibi bu model veriyi parçalayıp bu parçaların özelliklerini kullanarak öngörü yapar. Tümevarımsal bir öğrenme yöntemidir.



Şekil 2: Sınıflandırıcı Karar Ağacı

Eđitim setimizdeki zelliklere dayanarak, karar ađacı modeli rneklerin sınıf etiketlerini anlamak iin bir dizi soru đrenir. Yorumlanabilirliđin nemli olduđu durumlarda karar ađacı kullanışlı bir modeldir. Yukarıdaki rnek kategorik hedefler ile kurulan bir karar ađacı olduđu iin sınıflandırıcı bir karar ađacıdır. Kategorik hedefler yerine reel sayılar kullanılan karar ađaçları ise regresyon iin kullanılır. (Xu, Watanachaturaporn, Varshney, & Arora, 2005)



Şekil 3: Regresyon Karar Ađacı

Karar ađacı konusunu ieren alıřmalardan “Autoregressive Tree Models for Time-Series Analysis” (Meek, Chickering, & Heckerman, 2002), “Hidden Markov Decision Trees” (Jordan, Ghahramani, & Saul, 1997) ve “Contributions to Decision Tree Induction: Bias / Variance Tradeoff and Time Series Classification” gibi yabancı kaynaklar yazılan algoritma iin kullanılmıřtır. Trk kaynaklar incelendiđinde “Finans Sektrnde Veri Madenciliđi Uygulaması” (Dondurmacı & ınar, 2014) alıřmasında IMKB 30 1990-2010 verileri zerinde sınıflandırma ve regresyon ađaçları modeli uygulanmıřtır fakat bu alıřmanın amacı fiyat ngrleri yapmak deđil algoritmanın sınıflandırıcı zelliklerini kullanarak regresyon iin nemli nitelikleri ne ıkarmaktır.

3.1.4 Destekli Karar Ormanı Regresyonu (Boosted Decision Forest Regression)

Destekli karar ormanı regresyonu; Stokastik gradyan artırmalı hesaplama yaklaşımı, TreeNet ve MART (Multiple Additive Regression Trees) isimleriyle de bilinmektedir. Geçtiğimiz birkaç yıl boyunca, bu teknik, öngörülü veri madenciliği için en güçlü yöntemlerden biri olarak ortaya çıkmıştır. Aslında sınıflandırma problemleri için yazılmış olan bu algoritma aynı zamanda regresyon için de kullanılır.

Destekli karar ormanı regresyonu baz alınan modeldeki alt sekanslarda bulunan kararları birleştirerek tahmin yürüten ilaveli bir modeldir. Genel fikir, ardışık her bir ağacın, önceki ağacın tahmini kalıntılarından yaratıldığı, basit ağaçların kombinasyonu olan bir dizi hesaplamaaktır. Bu model matematiksel olarak bir “g” fonksiyonunun daha basit “f” fonksiyonları toplamı olarak yazılması şeklinde gösterilebilir:

Denklem 8: Destekli Karar Ormanı Regresyon Modeli

$$g(x) = f_0(x) + f_1(x) + f_2(x) + \dots$$

Rastlantısal karar ağacı algoritmasından farklı olarak, her biri bir alt veri örneği kullanarak Destekli Karar Ormanı Regresyonu’nda gradyan artırma adı verilen belirli bir model oluşturma tekniği kullanır.

Örnek olarak $F(x)$ fonksiyonunu optimize etmek istediğimizi var sayalım. Bu durumda gradyan artırma algoritması yinelemeli olarak “ η ” adım için aşağıda belirtilen modeli hesaplayacaktır:

Denklem 9: Gradyant Artırma Modeli

$$x_{t+1} = x_t - \eta \left. \frac{\partial f}{\partial x} \right|_{x = x_t}$$

Bu gradyant artırımını “g” fonksiyonuna uygulayıp her adımda yaşanan empirik kaybı $L(y_i, g(x_i))$ olarak ifade edildiğinde modelin tamamı aşağıdaki şekilde gösterilebilir:

$$f_t = \arg \min_f \sum_{i=1}^N \left[\frac{\partial L(y_i, g(x_i))}{\partial g(x_i)} \Big|_{g = g_t} - f(x_i) \right]^2$$

Ağaçların bu tür "ilave ağırlıklı genişlemeleri", yordayıcı değişkenler ve ilgilenilen bağımlı değişken arasındaki ilişkilerin spesifik niteliği çok karmaşık olsa bile, nihayetinde öngörülen değerlerin gözlenen değerlere iyi bir şekilde uyduğu gösterilebilir. Bu nedenle, gradyant artırma yöntemi (basit ağaçların ağırlıklı bir ek genişlemesinin takılması) oldukça genel ve güçlü bir makine öğrenme algoritmasıdır. (Poyarkov, Drutsa, Khalyavin, Gusev, & Serdyukov, 2016)

İncelenen akademik çalışmalarda destekli karar ormanı modellerinin özellikle doğrusal olmayan fonksiyonlarda başarılı olduğu gözlemlenmiştir. Algoritmanın bu çalışmada kullanılması, doğrusal ve doğrusal olmayan regresyon yöntemlerinin kıyaslanması amacı ile önemlidir. (Rokach, 2016) (Johnson & Zhang, 2014) (Siringnano & Spiliopoulos, 2017). Türkçe kaynaklar incelendiğinde “Zaman Serisi Analizi ve Tahmini: Derin Öğrenme Yaklaşımı” (Güdelek, 2019) çalışmasında “al-sat” kararlarının %60 oranında doğruluğu görülmektedir. Konu hakkında güvenilir Türkçe kaynak noksanlığı göze çarpmaktadır.

3.1.5 Hızlı Orman Yüzdellik Dağılım Regresyonu (Fast Forest Quantile Regression)

Aynı Bayes doğrusal regresyon gibi dağılım regresyonu (Quantile Regression) tek bir ortalama tahmin değeri yerine tahmin edilen değerlerin dağılımı hakkında bilgi edinmek için yararlı bir regresyon yöntemidir. Bu yöntem fiyat tahmini, öğrenci performans tahminleri gibi birçok alanda kullanılır. Özellikle değişkenler arasında zayıf etkileşim olan durumlarda yordayıcı ilişkiler bulma konusunda iyidir. İncelenen çalışmalarda 2001 yılından beri kullanılan bu algoritmanın parametrik olmayan ve çok sayıda yordayıcı değişken içeren durumlarda etkin olduğu görülmüştür. (Meinshausen, 2006) (Yu & Moyeed, 2001) (Hao & Naiman, 2007) Konu hakkında Türkçe kaynak bulunamamıştır.

Bu regresyon algoritması, denetlenen bir öğrenme yöntemidir; bu, etiket sütunu içeren etiketli bir veri kümesi gerektirdiği anlamına gelir. Bir regresyon algoritması olduğundan, etiket sütunu yalnızca sayısal değerler içermelidir. Kuantilin en basit tanımı, bir veri kümesini eşit boyutlu gruplara ayıran bir değerdir; Böylece, nicel değerler gruplar arasındaki sınırları işaretler. İstatistiksel olarak konuşursak, nicelikler, rastgele bir değişkenin kümülatif dağılım fonksiyonunun (Cumulative Distribution Function) tersinden düzenli aralıklarla alınan değerlerdir.

Doğrusal regresyon modelleri, tek bir tahmin kullanarak bir sayısal değişkenin değerini tahmin etmeye çalışırken, ortalama, bazen hedef değişkenin aralığını veya tüm dağılımını tahmin etmeniz gerekir. Bayes regresyonu ve kuantil regresyon gibi teknikler bu amaç için geliştirilmiştir. Dağılım regresyonu, öngörülen değerlerin dağılımını anlamaya yardımcı olur. Bu modülde kullanılanlar gibi ağaç temelli dağılım regresyon modelleri, parametrik olmayan dağılımları tahmin etmede kullanılabilecekleri ek avantajlara sahiptir.

Kullanılan matematiksel model aşağıdaki şekildedir:

Denklem 11: Dağılım Regresyonu Modeli

$$Y_m^p = \sum_{j=1}^n \omega_j(x_m) Y_j$$

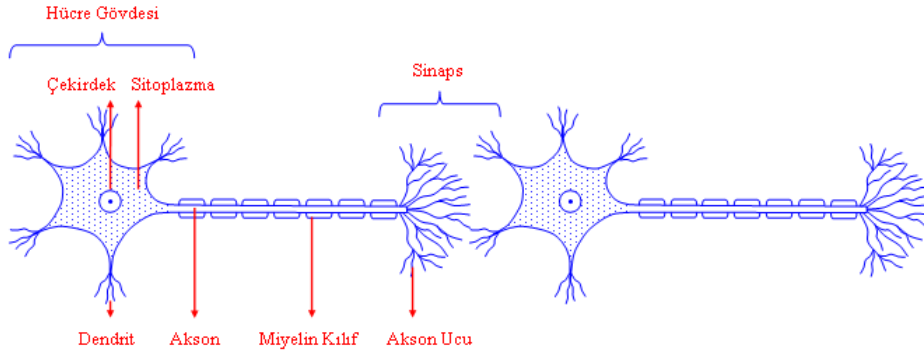
Buna baęlı “ τ ” derece daęılım formülü: Programlanan yapay zeka için 75. Yüzdelik deęer kullanılmıřtır.

Denklem 12: Daęılım Regresyonu Yüzdelik Model

$$\hat{F}(y|X = x) = \sum_{j=1}^n \omega_j(x) I(Y_j < y)$$

3.1.6 Nöral Ağ Regresyonu (Neural Network Regression)

Nöral ağ regresyonu, insan beyni ve sinir sisteminde bulunan nöronları model alan bir regresyon yöntemidir. Sinir sistemimiz “nöron” (neuron) denilen hücrelerden oluşur. Yeni doğan bir insanda ortalama 80 milyon nöron bulunur. Bu nöronlar kendilerinden önceki ve sonraki nöronlara (presinaptik nöron) bağlı bir ağ oluşturur. Sinirbilimde, nöronlar üzerinde taşınan sinyallere Aksiyon Potansiyeli adı verilir. Dendritlerle alınan sinyaller, hücre gövdesinde aksiyon potansiyelinin şiddetine göre bazı biyokimyasal değişimlere sebep olur. Bu değişimler, akson tepeciği (axon hillock) adı verilen, aksonu gövdeye bağlayan irice bölgede meydana gelen değişimleri tetikler ve böylece yeni bir aksiyon potansiyeli oluşturulur. Bu aksiyon potansiyeli, akson boyunca akarak ilerler ve akson ucunda, telodendritler adı verilen dallı bölgeye ulaşır. Burada bulunan sinaps adı verilen boşluğa salınan nörotransmitterler aracılığıyla sinaps sonrası, yani postsinaptik nöron yapısına iletilir. Böylece tek bir aksiyon potansiyeli, nörondan nörona sadece elektro biyokimyasal süreçlerle iletilmiş olur.

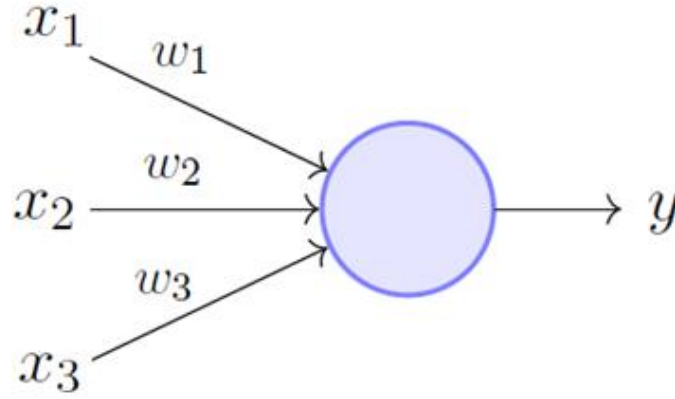


Şekil 4: Nöron Yapısı

Bilgisayar bilimleri bu sistemi basit bir şekilde model alır. Nöronlarda, belirli bir eşik değerin üzerinde bir aksiyon potansiyeli gelir ise, akson ucundaki nörotransmitter kesecikler patlar ve ortaya çıkan kimyasalların bir sonraki hücre sinapsı tarafından emilmesi ile elektrik sinyali iletilir. Eğer aksiyon potansiyeli belirli bir eşik değerini geçemez ise bu nörotransmitter kesecikler patlamaz ve aksiyon potansiyeli iletilmez. Aslında algoritmaların replike ettiği kısım bu kısımdır.

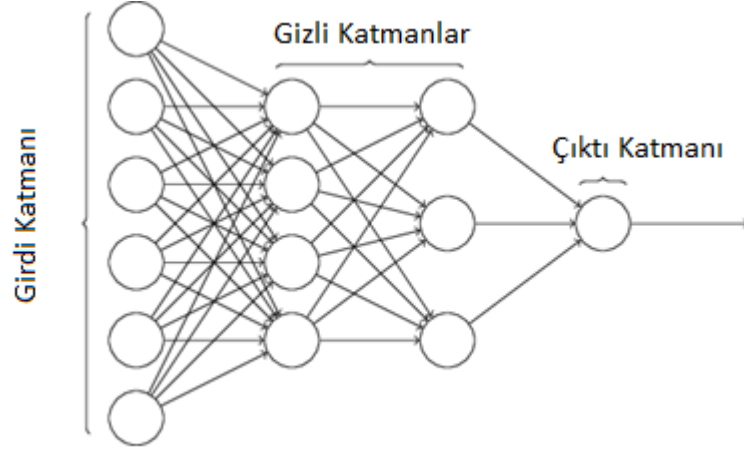
Bilgisayar bilimlerinde nörondan ilham alınan bu yapay nöronlara “perceptron” adı verilir. Aynı sinapsların birden fazla nörondan nörotransmitter madde alması gibi, bir veya

daha fazla girdi bir noda alınır. Şekildeki x_1 , x_2 ve x_3 değerleri bu girdilere örnektir. Belirtilen girdilerin ağırlık vektörleri (w_1 , w_2 , w_3) ile skalar çarpımı, belirlenen eşik değerinden büyük ise çıktı 1, küçük ise çıktı 0 olur. Bu durum beyindeki aksiyon potansiyelinin iletilmemesi veya iletilmemesini sembolize eder. Yukarıdaki örnekte görüldüğü gibi bu model tek bir nöronu sembolize eder ve bu sebeple “tek katmanlı perceptron (single layer perceptron)” olarak bilinir.



Şekil 5: Tek Katmanlı Yapay Nöron

Tahmin edileceği gibi beyin tek bir nörondan oluşmaz. Tek bir nöronu olan bir insanın zeki olmasını beklemek doğru olmaz. Aynı şekilde tek katmanlı nöronlar basit işlemlerin ötesinde kullanılamaz, yalnızca doğrusal fonksiyonlar üzerinde karar verebilirler. Bu belirtilen perceptronların bir ağ oluşturduğu kümeye ise çok katmanlı perceptron denir. Kullanılan nöral ağ regresyon algoritması, 29 bağımsız değişken sebebi ile çok katmanlı bir perceptron modeli kullanmaktadır.



Şekil 6: Çok Katmanlı Yapay Nöron

Nöral ağ en umut verici makine öğrenmesi yöntemlerinden biri olması sebebi ile literatürde en fazla yer bulan yöntemlerdendir. Özellikle patern yakalama özelliği ile öne çıkar. Bu özellik otomatik spam e-posta yakalama, bilgisayar ağ atakları güvenlik sistemleri, kişiye özel pazarlama, yüz tanıma, ses tanıma, fotoğraf içindeki objeleri ayırma gibi birçok uygulamada gerekli olduğu için konu hakkında yapılan çalışma sayısı oldukça fazladır. Finans alanında yapılan nöral ağ kullanan çalışmalar etkileyici sonuçlar sergilemektedir. (Pradeepkumar & Ravi, 2017) (Chen & Leung, 2004) (Man-Chung, Chi-Cheong, & Chi-Chung, 2000) (Leung, Chen, & Daouk, 2000) (Specht, 1991)

3.1.7 Poisson Regresyonu (Poisson Regression)

Olasılık teoremleri kısmında normal olmayan dağılımlar için incelenen Poisson dağılımı heterojen veri içeren veri kümeleri için idealdir. Finansal yatırım için kullanılan birçok veri normal dağılmaz. Bu sebeple Poisson dağılımı bazlı bir regresyon algoritmasının da tez kapsamında kullanılmasına karar verilmiştir.

Poisson regresyonu, sayısal değerleri tahmin etmek için kullanılan regresyon modellerinde, tipik olarak sayıya yöneliktir. Dağılım örnek olarak bir uçustaki hasta insan sayısı, belirli zaman aralığında beklenen sipariş sayısı, ihtimal tablosu üretimi gibi amaçlar için kullanılabilir. Burada dikkat edilmesi gereken nokta, dağılımın yoğunlaştığı ve seyrekleştiği noktalar. Örnek olarak bir restorana gelen sipariş verileri incelendiğinde kahvaltı, öğlen yemeği ve akşam yemeği saatlerinde yoğunluk, ara saatlerde seyreklik olarak göze çarpacaktır. Benzer paternler borsa açılış kapanış saatleri arasında da görülebilir. Finans verilerinin normal dağılıma uymadığı durumu ile ilgili bulgular bir daha önce birçok çalışmada öne sürülmüştür. (Schoutens, 2002) (Nannavecchia, 2015) (Mazzola & Muliere, 2011)

Poisson regresyon yalnızca tahmin edilen değerler aşağıdaki koşullara uygunsa kullanılmalıdır:

- Bağımlı değişken Poisson dağılımına sahipse
- Tüm sayımlar pozitif değerler ise (negatif değerlerde model çökmektedir)
- Veriler tam sayı ise. Kesirli sayılarda normalizasyona baş vurulmalıdır.

Kullanılan veri son özellik hariç kalan iki özelliği desteklemektedir. Tam sayı kullanılma zorunluluğu sebebi ile kesirli TL değerleri yerine tam sayı olan kuruş değerleri kullanılması durumunda son özellikte sağlanmaktadır.

Poisson dağılımı aşağıda belirtilen olasılık kütle fonksiyonunu kullanır:

Denklem 13: Olasılık Kütle Fonksiyonu

$$P_x(k) = \frac{e^{-(\lambda t)} * (\lambda t)^k}{k!} = \text{Poisson } (\lambda t)$$

Gösterilen fonksiyonda:

$P_x(k)$ = “t” zamanda “k” olayının görülme olasılığı

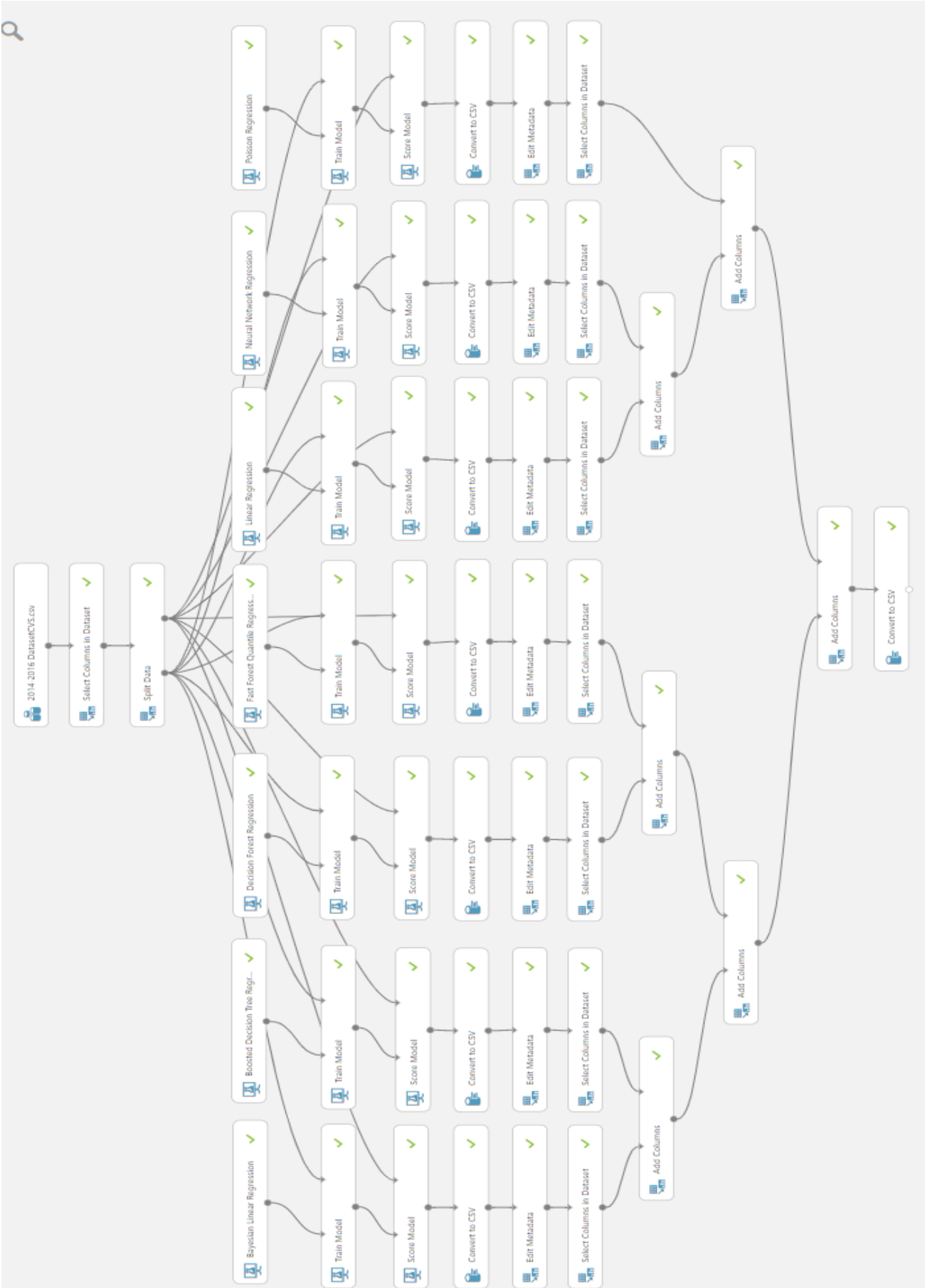
λt = birim zamanda gerçekleşen olay

k = olay sayısı'nı göstermektedir.

İncelenen çalışmalarda Poisson ve Poisson bazlı algoritmaların finans alanındaki farklı problemlerde etkili olduğu görülmüştür. Faria & Goncalves Portekiz bankaları için yaptığı müşteri kredi notu çalışmasında Poisson regresyonu ile etkili sonuçlar göstermektedir. (Faria & Goncalves) Andreas Heinen'in Autoregressive Conditional Poisson algoritması New York Stock Exchange piyasası IBM firması verileri tahminleri çalışmada kullanılan veri seti eski olsa da etkileyicidir. (Heinen, 2000) Yine ABD piyasasında FedEX firması piyasa fiyatları üzerine farklı Poisson regresyonunu test eden çalışma başarılı sonuçlar elde etmiştir. (Jayasekare, Gill, & Lee, 2016)

BÖLÜM IV. ALGORİTMALARIN BİST'TE TEST EDİLMESİ

BİST 30 da bulunan hisseler için belirtilen 7 yapay zeka programı yazılmış, 2014-2016 yılları günlük kapanış değeri verileri ile eğitilmiş ve gelecek tahminleri gerçek verilerle kıyaslanmıştır. Programın çalışma süresi ve karmaşıklık düzeyini makul tutmak için Akbank verileri tahmin edilmiştir. Veri seti Güray Küçükkoçaoğlu tarafından sağlanmış, verilerin en temiz olduğu aralık 2014-2016 olması sebebi ile bu aralık kullanılmıştır. Kod şeması ve çıktı örneği aşağıda gösterilmiştir. Veri seti 02/01/2014 tarihinden 30/12/2016 tarihleri arasında işlem yapılan 755 iş gününü kapsar. Öğrenme sürelerinin performansa etkilerini görmek için öğrenme / tahmin oranları %80-20, %90-10, %99-1 olarak belirlenen 3 farklı çalışma yapılmıştır. Yani birinci deneyde 755 günlük verinin 603 günü ile programlar eğitilmiş, kalan 152 gün tahmin edilmiştir. İkinci deneyde 755 günlük verinin 680 günü ile programlar eğitilmiş, kalan 75 gün tahmin edilmiştir. Son deneyde ise 755 günlük verinin 747 günü ile programlar eğitilmiş, kalan 8 gün tahmin edilmiştir. Bu deneyler kısa orta ve uzun vadeli tahminleri simüle etmektedir. Kod şeması aşağıda gösterilmiştir.



Şekil 7: Kod Şeması

Algoritmaların BIST’te test edilmesi için kullanılan bilgisayar donanımı da özellikle sonuçların kullanışlı sürelerde elde edilebilmesi açısından önemlidir. Derin öğrenme içeren programlar çok fazla sayıda matematiksel hesap gerektirir. Modern işlemcilerde bir veya daha fazla kuvvetli çekirdek bulunur. Normal programlar için bilgisayarın ana işlemcisi (CPU) bu hesaplamalar için yeterlidir fakat öğrenilmesi gereken veri arttıkça bu işlemci yavaş kalmaya başlar. Yapılacak hesaplamalar zor hesaplamalar değildir, yalnızca sayıca çok fazladır. Grafik işlemcileri (GPU) ise binlerce basit çekirdek içerebilirler ve bu sebeple yapay zeka işlemleri için çok daha uygundur. CPU üzerinde günler ve hatta aylar alan hesaplamalar GPU üzerinde birkaç saatte tamamlanabilir. CPU ve GPU tüm modern bilgisayarlarda, hatta cep telefonlarında bile bulunmaktadır. Bu hesaplamaların en hızlı şekilde yapılması için ASICs ve FPGAs denilen iki farklı endüstriyel işlemci şekli daha bulunmaktadır. Bu işlemciler günlük hayatta normal cihazlarda kullandığımız işlemciler değildir. Belirtilen işlemciler ve özellikleri aşağıda açıklanmıştır. (Microsoft, 2020)

<u><i>İşlemci</i></u>	<u><i>Kısaltma</i></u>	<u><i>Açıklama</i></u>
Central Processing Unit	CPU	Çok amaçlı işlemcilerdir, performanslı grafik ve yapay zeka işlemleri için yetersizdir
Graphical Processing Unit	GPU	Yapay zeka hesaplamaları için popüler bir işlemcidir. Çok sayıda çekirdeğe sahip olması paralel hesaplamaları hızlandırır
Application Specific Integrated Circuits	ASICs	Google tarafından üretilen TensorFlow Processor Unit gibi özel çipler yapay zeka hesaplamalarında en yüksek performansı gösterir. Gereksinimler değiştiğinde yeniden programlanamazlar
Field Programmable Gate Arrays	FPGAs	ASIC çiplere performans olarak yakın özellikler gösterirler, yeniden programlanabilen esnek çiplerdir.

Program öncelikle bir ev bilgisayar sisteminde denenmiştir. Kullanılan GeForce RTX 2060 model GPU'nun teknik özellikleri aşağıda belirtilmiştir:

GPU Motoru Özellikleri

NVIDIA CUDA® Çekirdekleri	2176
RTX-OPS	41 T
Giga Ray/sn	6
Artırılmış Saat Hızı (MHz)	1650
Standart Saat Hızı (MHz)	1470

Bellek Özellikleri:

Bellek Hızı	14 Gbps
Standart Bellek Yapılandırması	8 GB GDDR6
Bellek Arayüzü Genişliği	256 bit
Bellek Bant Genişliği (GB/sn)	448

(Nvidia, 2020)

Belirtilen GPU ile yapılan testlerde hesaplamaların haftaları bulan uzun süreler gerektirdiği görülmüştür. Bu sebeple Microsoft'tan üzerinde çok sayıda FPGA bulunan bir sunucu kiralanmış, çalışmalara sunucu üzerinde devam edilmiştir. Belirtilen sunucu üzerinde programlar çalıştırıldıktan sonra elde edilen örnek çıktı aşağıda gösterilmiştir. Tüm programlar 1 saatten az sürede tamamlanmaktadır.

TARİH	AKBNK	Bayesian Linear Forecast	Boosted Decision Tree Forecast	Decision Forest Forecast	Fast Forest 75th Quantile Forecast	Linear Forecast	Neural Network	Poisson Forecast
15-06-16	7.85	7.725740514	7.663112164	7.655416667	7.820000172	7.719312499	7.711872578	7.732929319
16-06-16	7.69	7.589016315	7.60192728	7.592916667	7.75	7.580084811	7.582484722	7.575622737
17-06-16	7.93	7.720732219	7.612642288	7.629166667	7.75	7.711050106	7.655127525	7.66790241
20-06-16	8.2	7.928197709	7.81237936	7.78125	7.980000019	7.919005787	7.813565254	7.867126346
21-06-16	8.23	7.858331093	7.732547283	7.710625	7.949999809	7.848763692	7.791465282	7.837074071
22-06-16	8.18	7.865612626	7.782143593	7.710625	7.940000057	7.853240672	7.774453163	7.824005723
23-06-16	8.38	7.958812175	7.894820213	7.813020833	8.06000042	7.94294634	7.836740494	7.939827561
24-06-16	8.17	7.56896377	7.662322521	7.608541667	7.769999981	7.553014977	7.614908695	7.630827576

Tablo 1: Örnek Çıktı

Öncelikle tüm algoritmalar için algoritma öngörülerini ve gerçek değerler çizgi grafikleri olarak gösterilmiştir.

Sonrasında tüm algoritmaların ürettiği öngörüler için maksimum hata, minimum hata, ortalama hata, maksimum hata yüzdesi, minimum hata yüzdesi ve ortalama hata yüzdesi verileri incelenmiştir. Böylelikle algoritmaların gerçek değerlere TL olarak ne kadar yaklaşabildiği ve belirtilen bu farkların yüzdeleri gösterilmiştir.

- Ortalama hata: Algoritmanın bulduğu tüm sonuçlar ve gerçek değerler arasındaki TL cinsinden farkın aritmetik ortalamasıdır.
- Maksimum hata: Algoritmanın bulduğu tüm sonuçlar ve gerçek değerler arasındaki TL cinsinden en büyük farktır.
- Minimum hata: Algoritmanın bulduğu tüm sonuçlar ve gerçek değerler arasındaki TL cinsinden en küçük farktır.
- Ortalama hata yüzdesi: Tüm tahmin ve gerçek değer için aradaki farkın gerçek değere bölümünün aritmetik ortalamasıdır.
- Maksimum hata yüzdesi: En büyük hatanın gerçek değere bölümüdür.
- Minimum hata yüzdesi: En küçük hatanın gerçek değere bölümüdür.

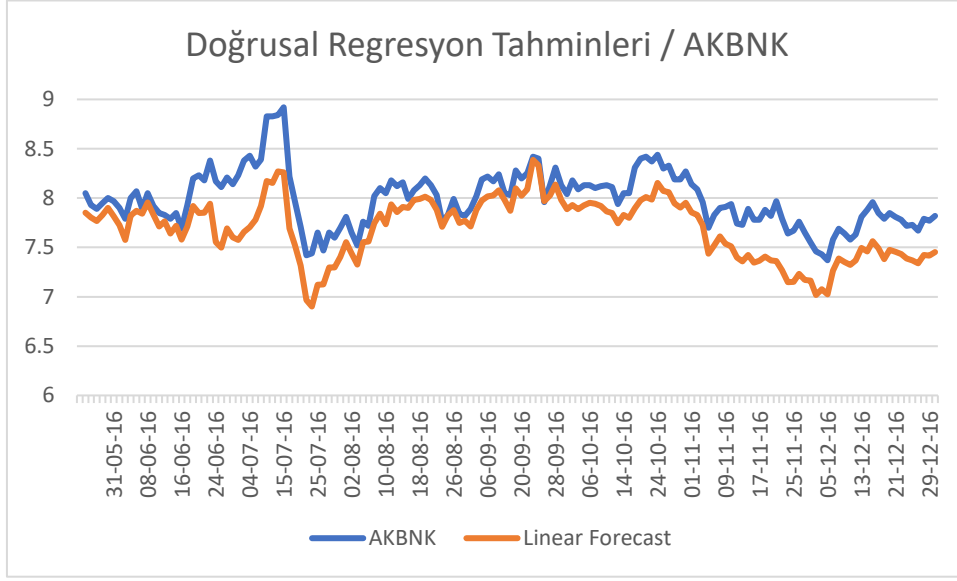
Sonrasında regresyon algoritmasının değişkeni açıklamadaki başarısını göstermek için determinasyon katsayısı (R^2) değerleri sunulmuştur. Determinasyon katsayısının yüksek olması regresyonun başarılı olduğu anlamına gelmez, fakat fiyat iniş çıkışları için kullanılabilir ve bu sebeple alım satım kararları için önemli bir veridir.

Son olarak hata oranı yüzdelerle dağılımları gösterilmiştir. Bu dağılımları incelemenin amacı algoritmaların yaptığı hatalar çoğunlukla küçük hatalar mı büyük hatalar mı göstermek, tahminler gerçek değerden yüksek ya da alçak olma eğiliminde mi incelemek ve yapılan hataların normal dağılıp dağılmadıklarını göstermektir.

3.1 Uzun Vade Öngörüleri

Makine öğrenmesi yöntemlerinde önerilen en düşük öğrenme oranı %80'dir. Bu sebeple başlangıç noktası olarak bu değer seçilmiştir. En az öğrenme bu çalışmada gerçekleşeceği için tüm çalışmada görülecek en düşük performansın bu çalışmada görülmesi beklenmektedir. İlk yapılan çalışmada verilerin %80'i öğrenme için kullanılmış, kalan %20 tahmin edilmiş ve gerçek veriler ile karşılaştırılmıştır. 755 günlük verinin 603 günü ile programlar eğitilmiş, kalan 152 gün tahmin edilmiştir. Tüm algoritmalar için tahmine karşılık gerçek değerlerin grafikleri, hata oranları, hata yüzdeleri, determinasyon katsayıları ve hata yüzdesi grafikleri gösterilmiştir. Çalışma Microsoft Azure servisi üzerinde kiralanan bir sunucuda yapılmıştır. Tüm algoritmalar 1 saatten kısa bir sürede tamamlanmaktadır.

4.1.1 Doğrusal Regresyon (Linear Regression)



Tablo 2: Uzun Vade Doğrusal Regresyon / AKBNK

Maksimum Hata: 0,73

Minimum Hata: 0

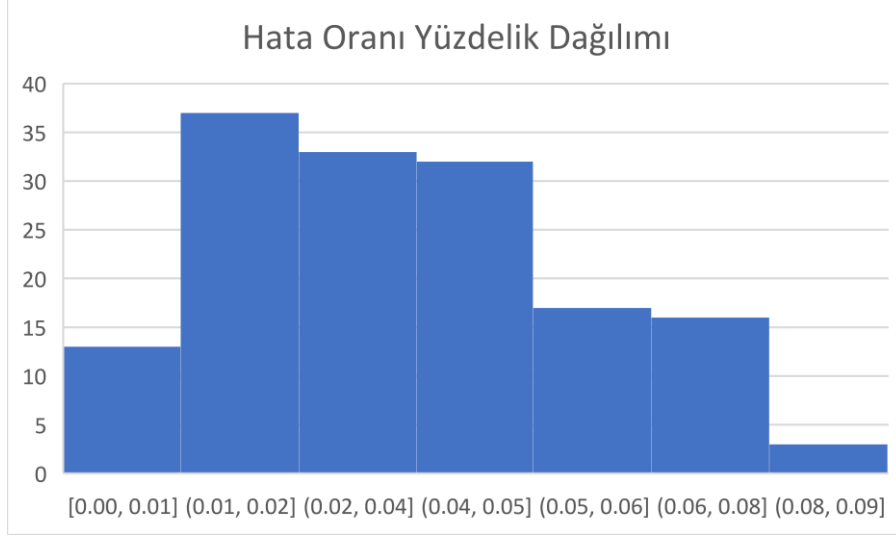
Ortalama Hata: 0,30

Maksimum Hata Yüzdesi: %8,6096

Minimum Hata Yüzdesi: %0,0488

Ortalama Hata Yüzdesi: %3,7006

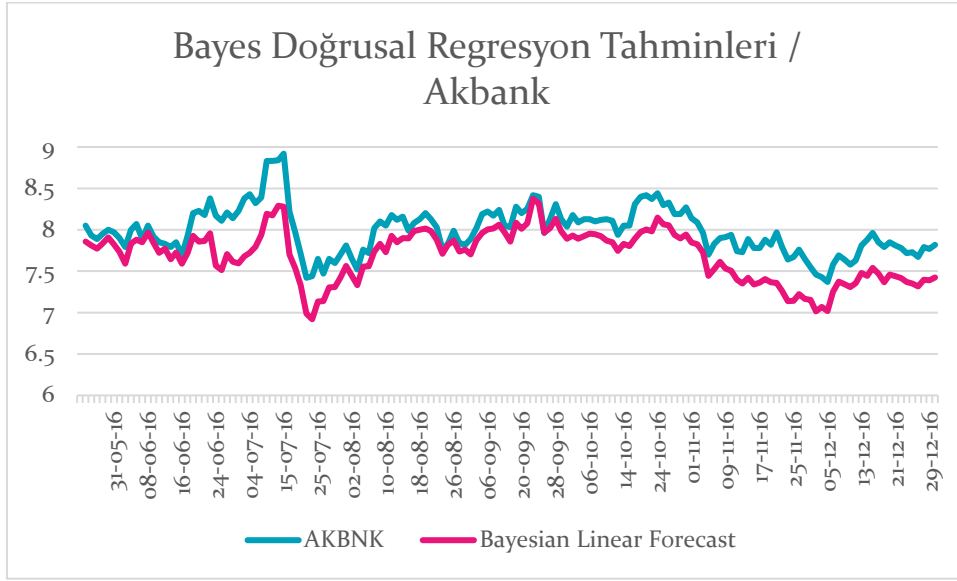
R^2 : 0.730512



Tablo 3: Uzun Vade Doğrusal Regresyon Ortalama Hata Yüzdelerik Dağılımı

Sonuçlar doğrusal regresyon algoritmasının gerçek değerleri ortalama 30 kuruş hata ile tahmin ettiğini göstermektedir. Yapılan en yüksek hata 73 kuruş, en düşük hata durumunda ise değerler tamamen doğru tahmin edilmiştir. Determinasyon katsayısı değeri 0,73'tür. Grafikler incelendiğinde doğrusal regresyon algoritmasının devamlı olarak gerçek değerlerin altında tahminler ürettiği, hiçbir tahmininde gerçek değer üzerinde tahmin üretmediği görülmektedir. Bu sonuç hata oranı yüzdelerik dağılımı grafiğindeki tüm değerlerin pozitif olmasından da görülebilir.

4.1.2 Bayes Doğrusal Regresyon (Bayes Linear Regression)



Tablo 4: Uzun Vade Bayes Doğrusal Regresyon / AKBNK

Maksimum Hata: 0,71

Minimum Hata: 0

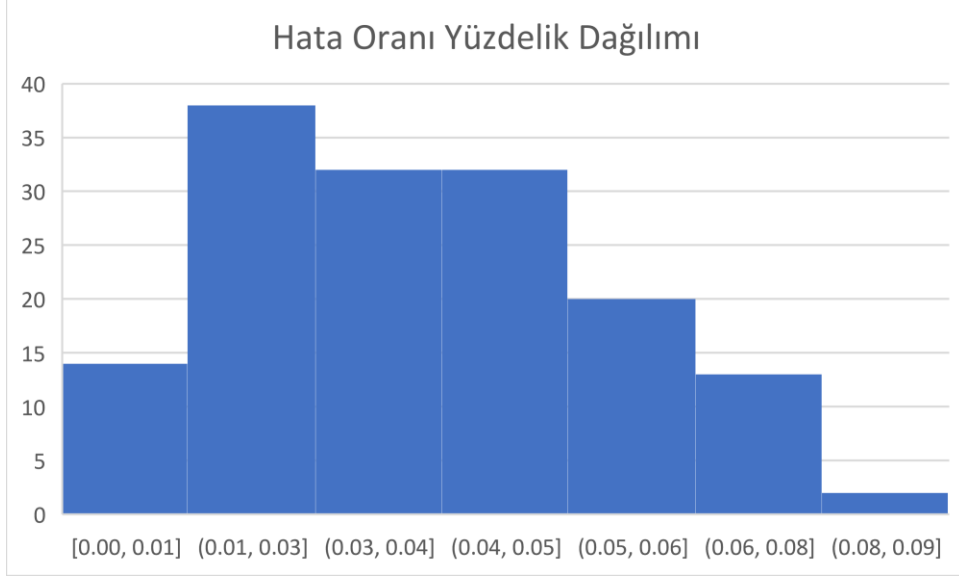
Ortalama Hata: 0,30

Maksimum Hata Yüzdesi: %8,3776

Minimum Hata Yüzdesi: %0,0529

Ortalama Hata Yüzdesi: %3,7102

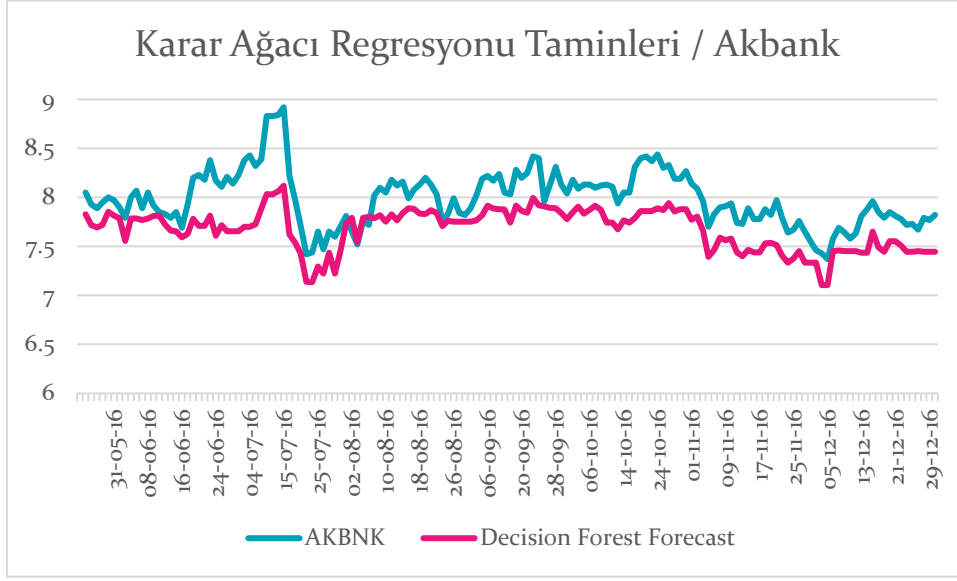
R^2 : 0.739606



Tablo 5: Uzun Vade Bayes Doğrusal Regresyon Ortalama Hata Yüzdelerik Dağılımı

Sonuçlar Bayes doğrusal regresyon algoritmasının gerçek değerleri ortalama 30 kuruş hata ile tahmin ettiğini göstermektedir. Yapılan en yüksek hata 71 kuruş, en düşük hata durumunda ise değerler tamamen doğru tahmin edilmiştir. Determinasyon katsayısı değeri 0,74'tür Grafikler incelendiğinde doğrusal regresyon algoritmasının devamlı olarak gerçek değerlerin altında tahminler ürettiği, hiçbir tahmininde gerçek değer üzerinde tahmin üretmediği görülmektedir. Bu sonuç hata oranı yüzdelerik dağılımı grafiğindeki tüm değerlerin pozitif olmasından da görülebilir. Sonuçlar doğrusal regresyon algoritmasına yakın olsa da ufak aynı algoritmaya kıyasla bir performans artışı görülmektedir.

4.1.3 Karar Ağacı Regresyonu (Decision Tree Regression)



Tablo 6: Uzun Vade Karar Ağacı Regresyonu / AKBNK

Maksimum Hata: 0,80

Minimum Hata: 0,02

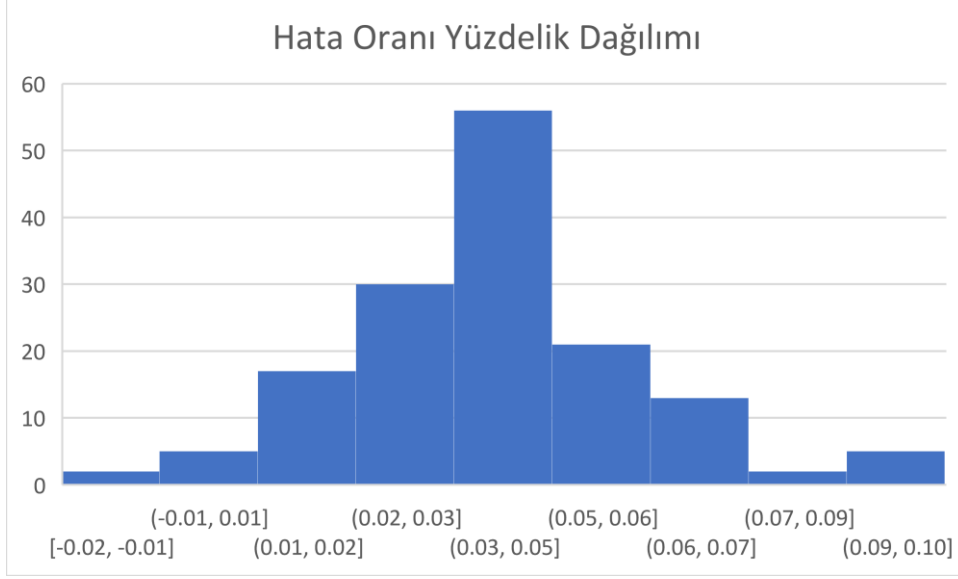
Ortalama Hata: 0,31

Maksimum Hata Yüzdesi: %9,0530

Minimum Hata Yüzdesi: %0,3075

Ortalama Hata Yüzdesi: %3,8993

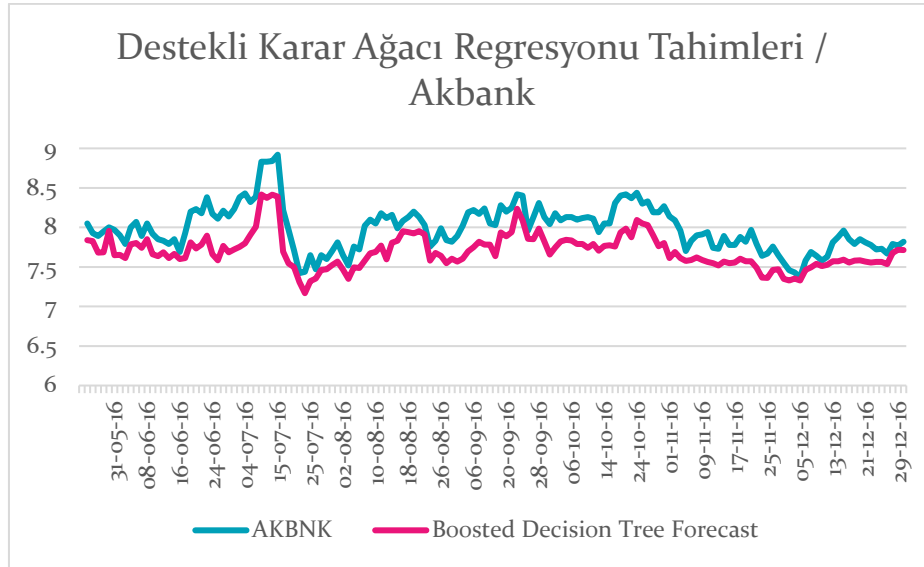
R^2 : 0.674318277



Tablo 7: Uzun Vade Karar Ağacı Regresyonu Ortalama Hata Yüzdelerik Dağılımı

Sonuçlar karar ağacı regresyonu algoritmasının gerçek değerleri ortalama 31 kuruş hata ile tahmin ettiğini göstermektedir. Yapılan en yüksek hata 80 kuruş, en düşük hata durumunda ise 2 kuruş hata ile değerler tahmin edilmiştir. Determinasyon katsayısı değeri 0,67'dir. Grafikler incelendiğinde karar ağacı regresyonu algoritmasının büyük çoğunlukla gerçek değerlerin altında tahminler ürettiği görülmektedir fakat sayıca az olsa da gerçek değerlerin üstünde tahminler de vardır. Bu sonuç hata oranı yüzdelerik dağılımı grafiğindeki negatif bölümün ve pozitif bölümün karşılaştırılarak görülebilir.

4.1.4 Destekli Karar Ormanı Regresyonu (Boosted Decision Forest Regression)



Tablo 8: Uzun Vade Destekli Karar Ormanı Regresyonu / AKBNK

Maksimum Hata: 0,63

Minimum Hata: 0,04

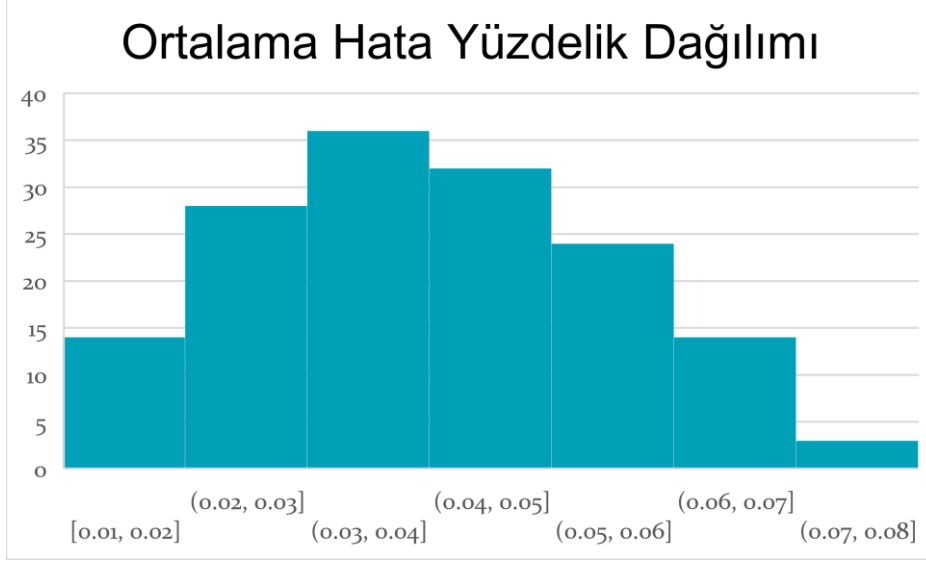
Ortalama Hata: 0,29

Maksimum Hata Yüzdesi: %7,4843

Minimum Hata Yüzdesi: %0,5499

Ortalama Hata Yüzdesi: %3,5879

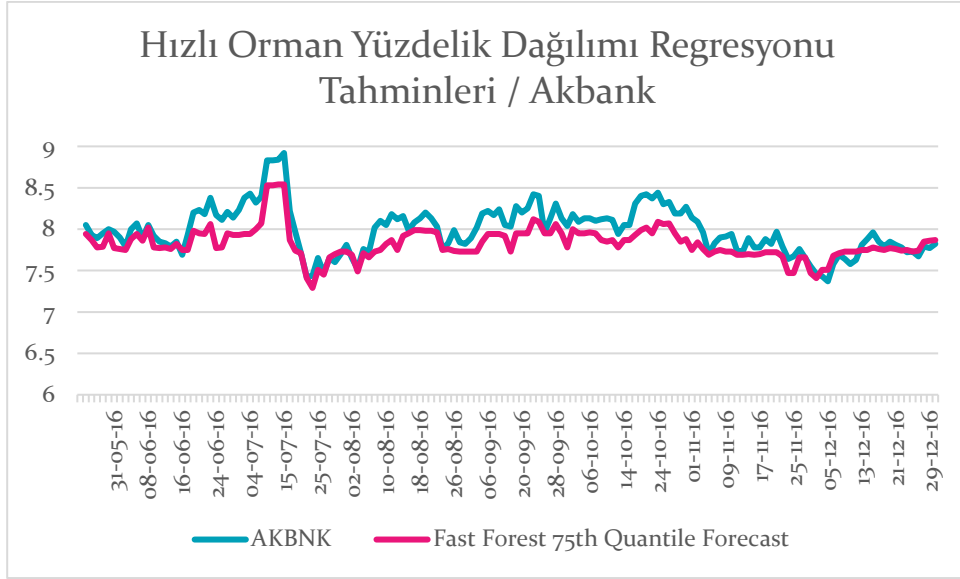
R^2 : 0.730511742



Tablo 9: Uzun Vade Destekli Karar Ormanı Regresyonu Ortalama Hata Yüzdelik Dağılımı

Sonuçlar destekli karar ormanı regresyonu algoritmasının gerçek değerleri ortalama 29 kuruş hata ile tahmin ettiğini göstermektedir. Yapılan en yüksek hata 63 kuruş, en düşük hata durumunda ise 4 kuruş hata ile değerler tahmin edilmiştir. Determinasyon katsayısı değeri 0,73'tür. Bu değer karar ağacı regresyonu ile kıyaslandığında bu algoritmanın değişkenler arasındaki ilişkiyi daha iyi açıkladığı görülmektedir. Grafikler incelendiğinde destekli karar ormanı regresyonu algoritmasının devamlı olarak gerçek değerlerin altında tahminler ürettiği görülmektedir. Bu sonuç hata oranı yüzdelik dağılımı grafiğindeki tüm değerlerin pozitif olmasından görülebilir. Karar ağacı regresyonu algoritması ile kıyaslandığında maksimum hata değerinde ciddi bir fark görülmektedir. Minimum hata değerinin karar ağacı regresyonu algoritmasından yüksek olması ve tüm tahminlerin gerçek değerlerin altında olması dikkat çekicidir.

4.1.5 Hızlı Orman Yüzelik Dağılım Regresyonu (Fast Forest Quantile Regression)



Tablo 10: Uzun Vade Hızlı Orman Yüzelik Dağılım Regresyonu / AKBNK

Maksimum Hata: 0,49

Minimum Hata: 0,00

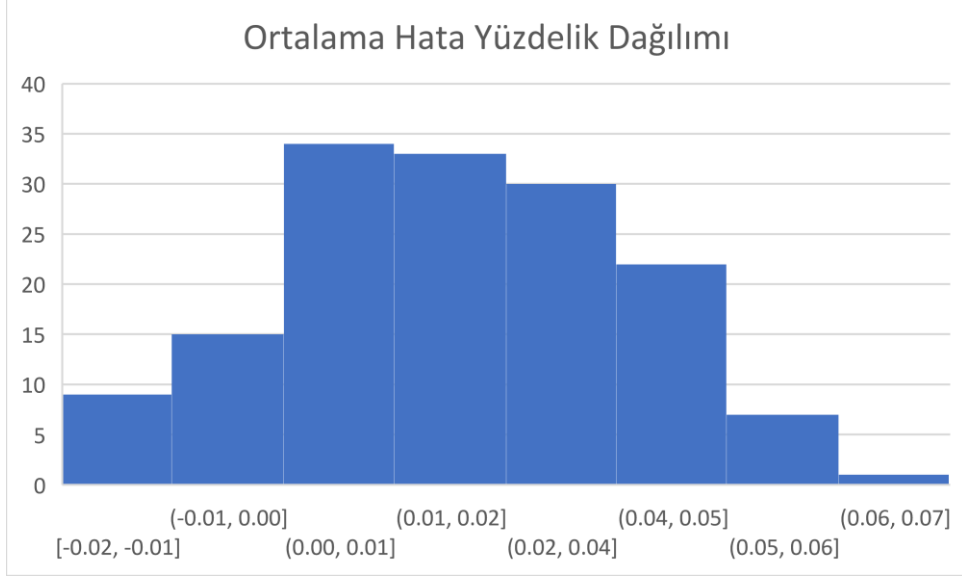
Ortalama Hata: 0,17

Maksimum Hata Yüzdesi: %5,8125

Minimum Hata Yüzdesi: %0

Ortalama Hata Yüzdesi: %2,14

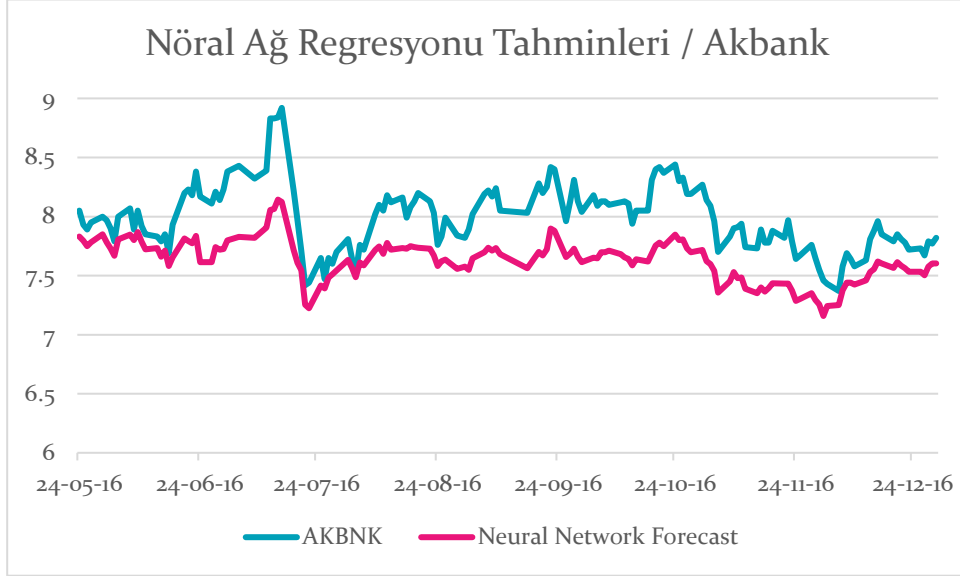
R²: 0.816486082



Tablo 11: Uzun Vade Hızlı Orman Yüzdeleri Regresyonu Ortalama Hata Yüzdeleri Dağılımı

Sonuçlar hızlı orman yüzdeleri dağılım regresyonu algoritmasının gerçek değerleri ortalama 17 kuruş hata ile tahmin ettiğini göstermektedir. Yapılan en yüksek hata 49 kuruş, en düşük hata durumunda ise değerler tamamen doğru tahmin edilmiştir. Determinasyon katsayısı değeri 0,82'dir. Bu değer önceki iki karar ağacı regresyonu algoritmaları ile kıyaslandığında bu algoritmanın değişkenler arasındaki ilişkiyi daha iyi açıkladığı görülmektedir. Grafikler incelendiğinde destekli karar ormanı regresyonu algoritmasının normal dağılıma yakın fakat gerçek değerlerin altında tahmin etme eğiliminde olduğu görülmektedir. Bu sonuç hata oranı yüzdeleri dağılımı grafiğindeki değerlerin pozitif tarafa eğimli olmasından görülebilir. Önceki algoritmalarla kıyasla daha etkili ortalama hata, maksimum hata ve minimum hata değerleri görülmektedir.

4.1.6 Nöral Ağ Regresyonu (Neural Network Regression)



Tablo 12: Uzun Vade Nöral Ağ Regresyonu / AKBNK

Maksimum Hata: 0,80

Minimum Hata: 0,03

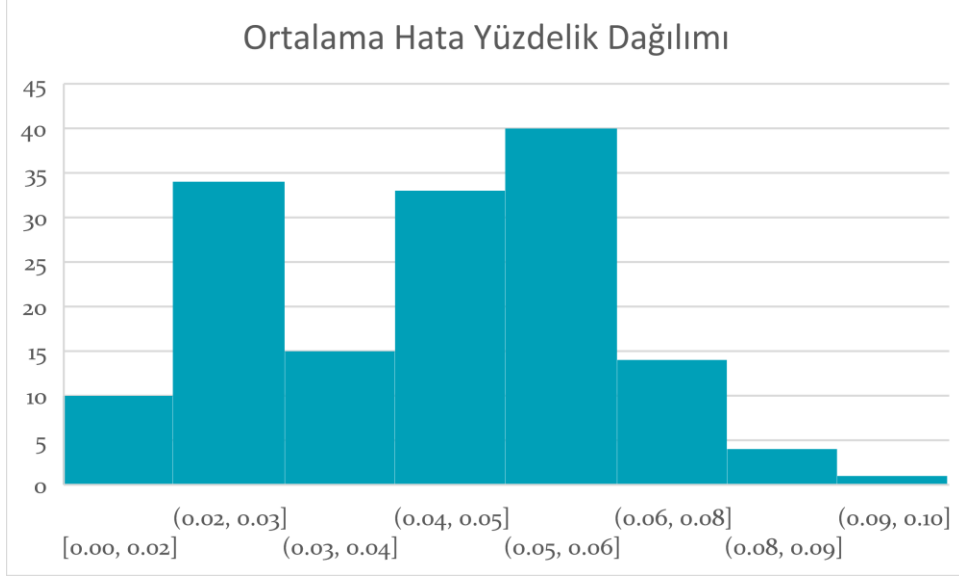
Ortalama Hata: 0,35

Maksimum Hata Yüzdesi: %8,9294

Minimum Hata Yüzdesi: %0

Ortalama Hata Yüzdesi: %4,3867

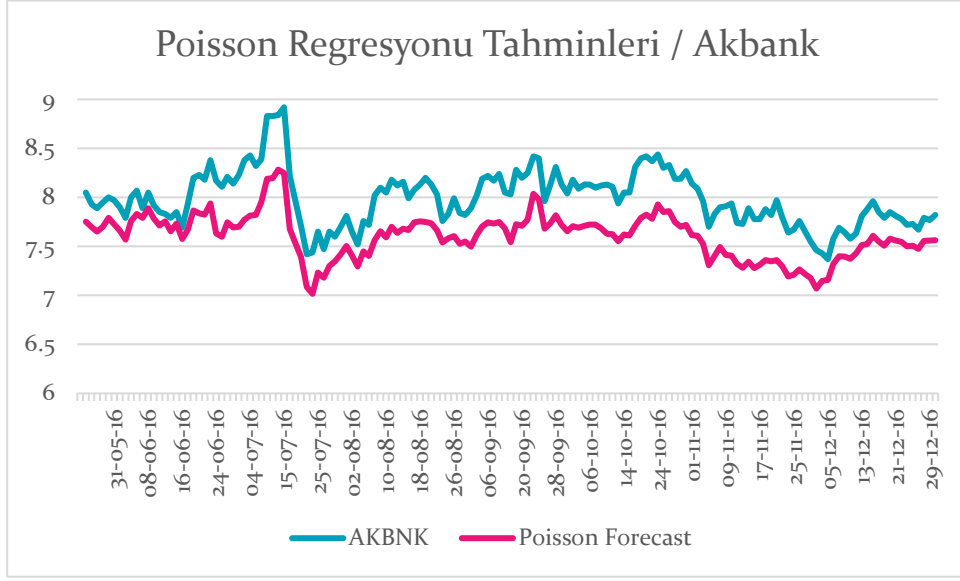
R^2 : 0.724042775



Tablo 13: Uzun Vade Nöral Ağ Regresyonu Ortalama Hata Yüzdeleri Dağılımı

Sonuçlar nöral ağ regresyonu algoritmasının gerçek değerleri ortalama 35 kuruş hata ile tahmin ettiğini göstermektedir. Yapılan en yüksek hata 80 kuruş, en düşük hata durumunda ise 3 kuruş hata ile değerler tahmin edilmiştir. Determinasyon katsayısı değeri 0,72'dir. Grafikler incelendiğinde nöral ağ regresyonu devamlı olarak gerçek değerlerin altında tahminler ürettiği görülmektedir. Bu sonuç hata oranı yüzdeleri dağılımı grafiğindeki tüm değerlerin pozitif olmasından görülebilir. İncelenen makaleler göz önünde bulundurulduğunda nöral ağ kalan regresyon yöntemlerinden kötü sonuçlar çıkartması şaşırtıcıdır.

4.1.7 Poisson Regresyonu (Poisson Regression)



Tablo 14: Tablo 17: Uzun Vade Poisson Regresyonu / AKBNK

Maksimum Hata: 0,67

Minimum Hata: 0,07

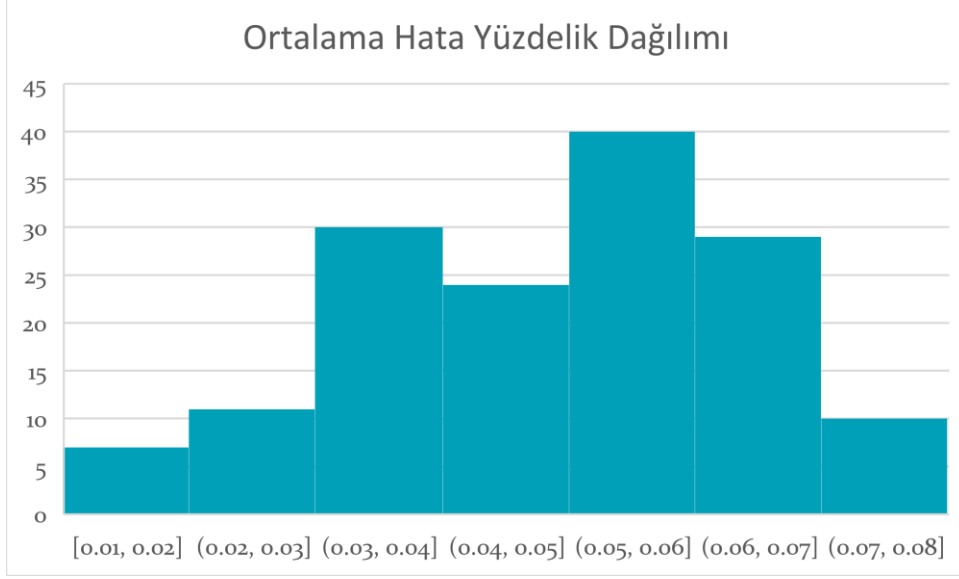
Ortalama Hata: 0,38

Maksimum Hata Yüzdesi: %7,6379

Minimum Hata Yüzdesi: %0,9353

Ortalama Hata Yüzdesi: %4,7690

R^2 : 0.803203465



Tablo 15: Uzun Vade Poisson Regresyonu Ortalama Hata Yüzdeleri Dağılımı

Sonuçlar Poisson regresyonu algoritmasının gerçek değerleri ortalama 38 kuruş hata ile tahmin ettiğini göstermektedir. Yapılan en yüksek hata 67 kuruş, en düşük hata durumunda ise 7 kuruş hata ile değerler tahmin edilmiştir. Determinasyon katsayısı değeri 0,80'dir. Grafikler incelendiğinde nöral ağ regresyonu devamlı olarak gerçek değerlerin altında tahminler ürettiği görülmektedir. Bu sonuç hata oranı yüzdeleri dağılımı grafiğindeki tüm değerlerin pozitif olmasından görülebilir. 38 kuruşluk ortalama hata bu çalışmada görülen en yüksek ortalama hata olması şaşırtıcıdır.

4.1.8 Değerlendirme

Sonuçlar karşılaştırıldığında aşağıdaki tablo ortaya çıkmaktadır:

	Doğrusal	Bayes Doğrusal	Karar Ağacı	Destekli Karar Ormanı	Hızlı Orman Yuzdelik	Nöral Ağ	Poisson
Max Hata	0.725084	0.705935774	0.8012083	0.630928688	0.489999943	0.796503	0.669986
Min Hata	0.003826	0.004242464	0.023125	0.041821117	0	0.032615	0.073238
Ortalama Hata	0.295765	0.296440934	0.3146676	0.289329583	0.173962466	0.354538	0.383143
Max Hata %	8.609664	8.37767679	9.0530119	7.484326074	5.812573464	8.929406	7.637946
Min Hata %	0.048861	0.053297286	0.3075133	0.549972063	0	0.433713	0.935352
Ortalama Hata %	3.700646	3.7102227	3.8993939	3.587942226	2.141477077	4.38677	4.769098
R2	0.730512	0.739605978	0.6743183	0.730511742	0.816486082	0.724043	0.803203

Tablo 16: Uzun Vade Sonuç Değerlendirme

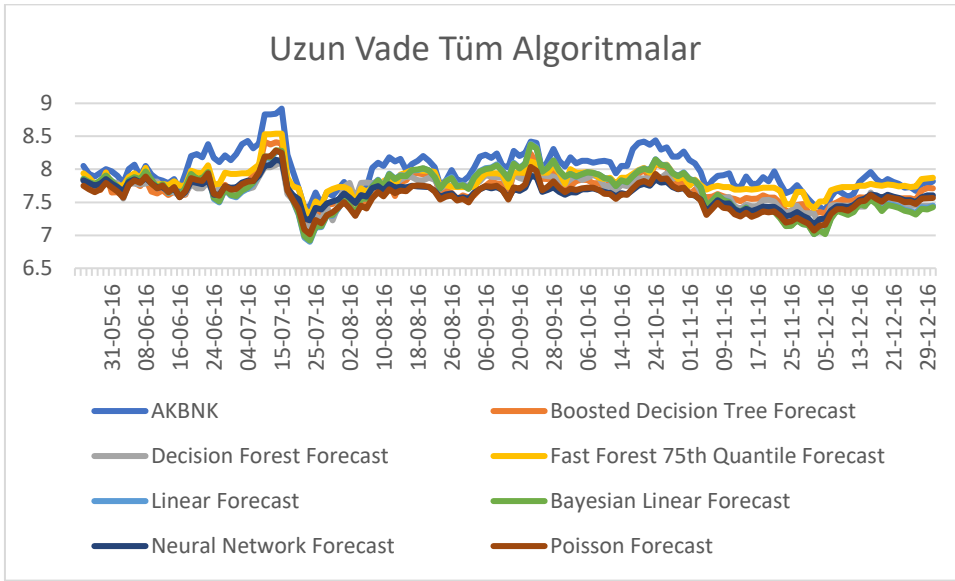
Ortalama hata kolunu karşılaştırıldığında bu deney için en etkin tahminleri 17 kuruş ortalama hata ve %2,14 ortalama hata yüzdesi ile Hızlı Orman Yüzdeler Regresyon algoritmasının ürettiği görülmektedir.

En iyi ikinci algoritmanın ise 29 kuruş ortalama hata ve %3,58 ortalama hata yüzdesi ile Destekli Karar Ormanı Regresyonu algoritması olduğu görülmektedir. Determinasyon katsayısı satırı incelendiğinde yine bu algoritmanın değişkenler arasındaki ilişkiyi en iyi açıklayan algoritmadır.

Kalan algoritmaların sonuçları birbirine yakın olmakla birlikte maksimum hata yüzdelerindeki farklılıklar sebebi ile Bayes Doğrusal Regresyon üçüncü, Doğrusal Regresyon algoritması dördüncü, Karar Ağacı algoritması ise beşinci sıradadır. Belirtilen algoritmalar birbirinin geliştirilmiş versiyonları kabul edilebileceği için bu sonuçların beklenen sonuçlar olduğu söylenebilir.

Poisson ve Nöral Ağ algoritmalarının bu çalışmadaki en kötü sonuçları üretmesi bu çalışmanın en şaşırtıcı yanıdır. Bu iki algoritmanın referans gösterilen makalelerde gösterdikleri yüksek performans sebebi ile daha yüksek performans göstermeleri gerektiği beklentisi vardır.

Tüm algoritmaların tahminleri aşağıdaki grafikte gösterilmiştir:

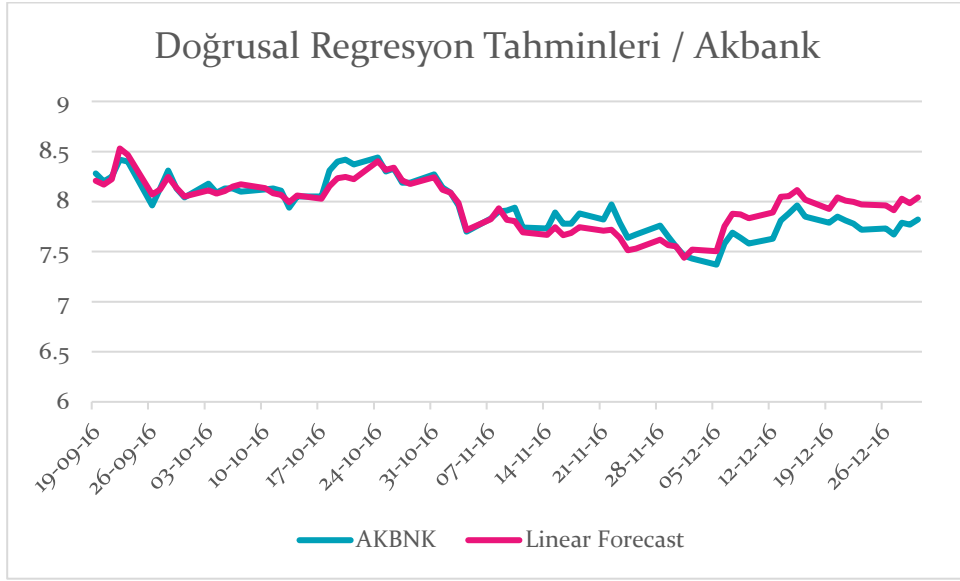


Tablo 17: Uzun Vade Tüm Algoritma Tahminleri / AKBNK

4.2 Orta Vade Öngörüleri

İkinci çalışmada öğrenme oranı arttırılarak verilerin %90'i öğrenme için kullanılmış, kalan %10 tahmin edilmiş ve gerçek veriler ile karşılaştırılmıştır. Performansın birinci çalışmaya kıyasla daha yüksek olması beklenir. 755 günlük verinin 680 günü ile programlar eğitilmiş, kalan 75 gün tahmin edilmiştir. Tüm algoritmalar için tahmine karşılık gerçek değerlerin grafikleri, hata oranları, hata yüzdeleri, determinasyon katsayıları ve hata yüzdesi grafikleri gösterilmiştir. Çalışma Microsoft Azure servisi üzerinde kiralanan bir sunucuda yapılmıştır. Tüm algoritmalar 1 saatten kısa bir sürede tamamlanmaktadır.

4.2.1 Doğrusal Regresyon (Linear Regression)



Tablo 18: Orta Vade Doğrusal Regresyon / AKBNK

Maksimum Hata: 0,26

Minimum Hata: 0

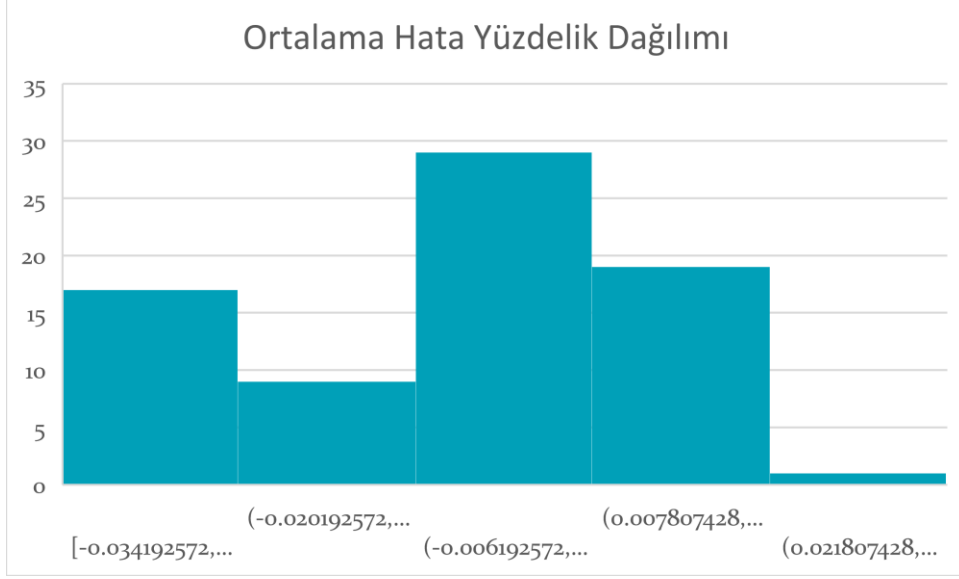
Ortalama Hata: 0,1

Maksimum Hata Yüzdesi: %3,4192

Minimum Hata Yüzdesi: 0

Ortalama Hata Yüzdesi: %1,3274

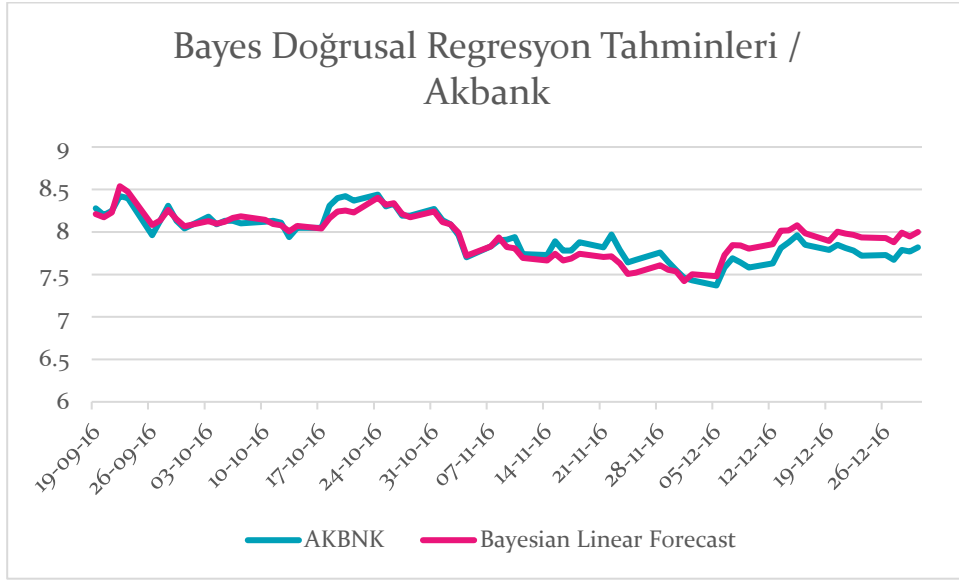
R^2 : 0.758842068



Tablo 19: Orta Vade Doğrusal Regresyon Ortalama Hata Yüzdeleri Dağılımı

Sonuçlar doğrusal regresyon algoritmasının gerçek değerleri ortalama 10 kuruş hata ile tahmin ettiğini göstermektedir. Yapılan en yüksek hata 26 kuruş, en düşük hata durumunda ise değerler tamamen doğru tahmin edilmiştir. Determinasyon katsayısı değeri 0,76'dır. Grafikler incelendiğinde doğrusal regresyon algoritmasının gerçek değerlerin üzerinde ve altında değerler ürettiği görülmektedir. Bu sonuç hata oranı yüzdeleri dağılımı grafiğindeki değerlerin pozitif ve negatif olmasından görülebilir. Negatif değerlerin fazlalıkta olması algoritmanın gerçek değerlerin üzerinde daha fazla sonuç ürettiğini göstermektedir. Doğru grafikleri 05-12-16 tarihi ve sonrasında üretilen tüm değerlerin gerçek değerlerin üstünde olduğunu göstermektedir. Bir önceki çalışma ile kıyaslandığında tüm performans metrikleri daha iyi sonuçlar göstermektedir.

4.2.2 Bayes Doğrusal Regresyon (Bayes Linear Regression)



Tablo 20: Orta Vade Bayes Doğrusal Regresyon / AKBNK

Maksimum Hata: 0,26

Minimum Hata: 0

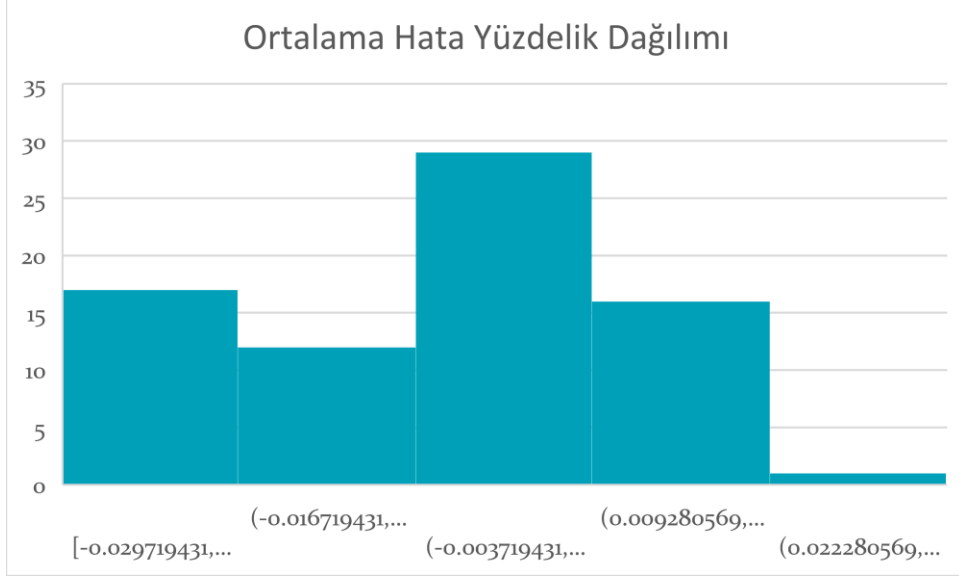
Ortalama Hata: 0,1

Maksimum Hata Yüzdesi: %3,2004

Minimum Hata Yüzdesi: 0

Ortalama Hata Yüzdesi: %1,2251

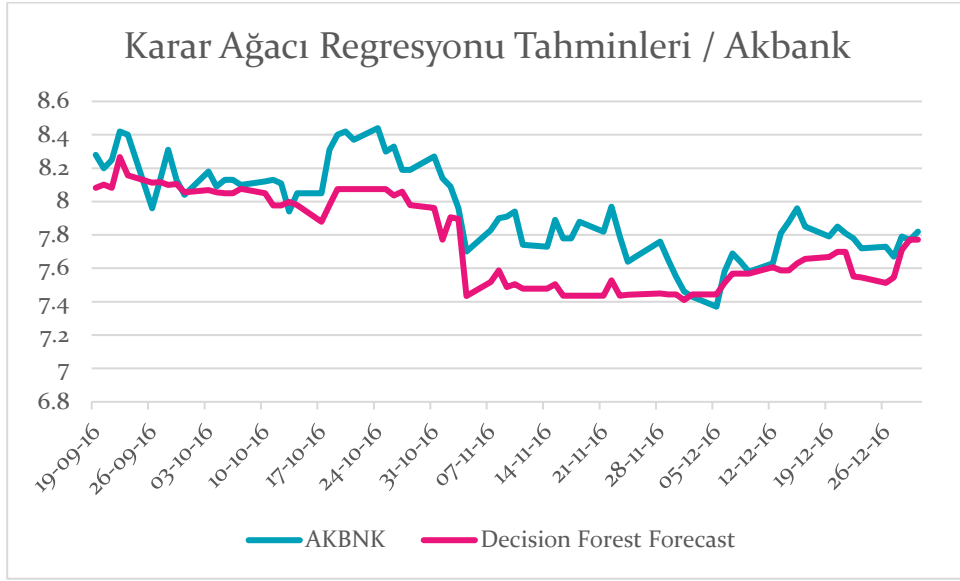
R^2 : 0.802050265



Tablo 21: Orta Vade Bayes Doğrusal Regresyon Ortalama Hata Yüzdeleri Dağılımı

Sonuçlar Bayes doğrusal regresyon algoritmasının gerçek değerleri ortalama 10 kuruş hata ile tahmin ettiğini göstermektedir. Yapılan en yüksek hata 26 kuruş, en düşük hata durumunda ise değerler tamamen doğru tahmin edilmiştir. Determinasyon katsayısı değeri 0,80'dır. Grafikler incelendiğinde bayes doğrusal regresyon algoritmasının gerçek değerlerin üzerinde ve altında değerler ürettiği görülmektedir. Bu sonuç hata oranı yüzdeleri dağılımı grafiğindeki değerlerin pozitif ve negatif olmasından görülebilir. Negatif değerlerin fazlalıkta olması algoritmanın gerçek değerlerin üzerinde daha fazla sonuç ürettiğini göstermektedir. Doğru grafikleri 05-12-16 tarihi ve sonrasında üretilen tüm değerlerin gerçek değerlerin üstünde olduğunu göstermektedir. Bir önceki çalışma ile kıyaslandığında tüm performans metrikleri daha iyi sonuçlar göstermektedir. Determinasyon kat sayısı değerinin doğrusal regresyon algoritmasına kıyasla daha yüksek olması bu algoritmanın değişkenler arasındaki ilişkiyi daha iyi açıkladığını göstermektedir.

4.2.3 Karar Ağacı Regresyonu (Decision Tree Regression)



Tablo 22: Orta Vade Karar Ağacı Regresyonu / AKBNK

Maksimum Hata: 0,44

Minimum Hata: 0

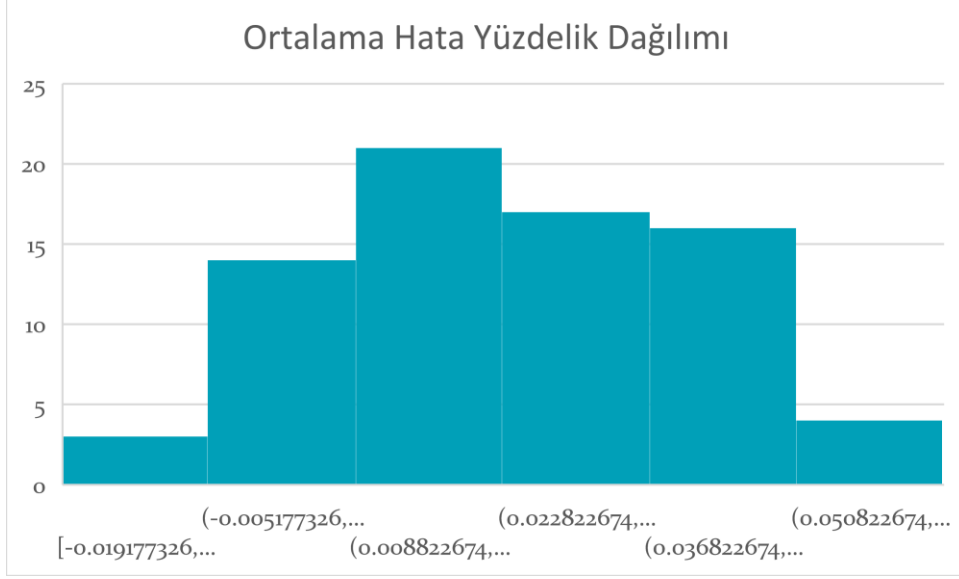
Ortalama Hata: 0,19

Maksimum Hata Yüzdesi: %5,6313

Minimum Hata Yüzdesi: %0

Ortalama Hata Yüzdesi: %2,4129

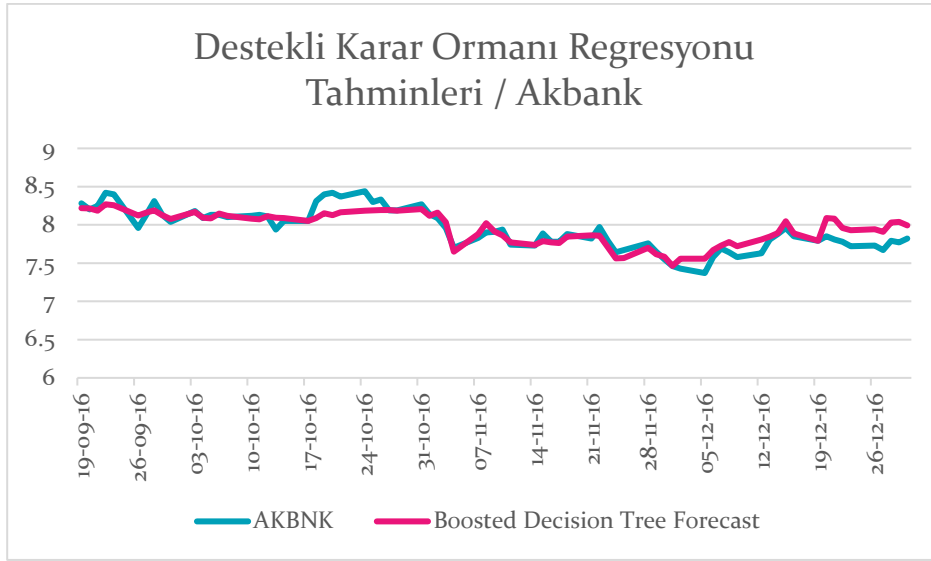
R²: 0.753092385



Tablo 23: Orta Vade Karar Ağacı Regresyonu Ortalama Hata Yüzdeleri Dağılımı

Sonuçlar karar ağacı regresyonu algoritmasının gerçek değerleri ortalama 19 kuruş hata ile tahmin ettiğini göstermektedir. Yapılan en yüksek hata 44 kuruş, en düşük hata durumunda ise değerler tamamen doğru tahmin edilmiştir. Determinasyon katsayısı değeri 0,75'tir. Grafikler incelendiğinde karar ağacı regresyonu algoritmasının gerçek değerlerin üzerinde ve altında değerler ürettiği görülmektedir. Bu sonuç hata oranı yüzdeleri dağılımı grafiğindeki değerlerin pozitif ve negatif olmasından görülebilir. Pozitif değerlerin fazlalıkta olması algoritmanın gerçek değerlerin altında daha fazla sonuç ürettiğini göstermektedir. Sonuçlar uzun vade çalışma sonuçları ile kıyaslandığında performansın arttığı görülmektedir.

4.2.4 Destekli Karar Ormanı Regresyonu (Boosted Decision Forest Regression)



Tablo 24: Orta Vade Destekli Karar Ormanı Regresyonu / AKBNK

Maksimum Hata: 0,29

Minimum Hata: 0

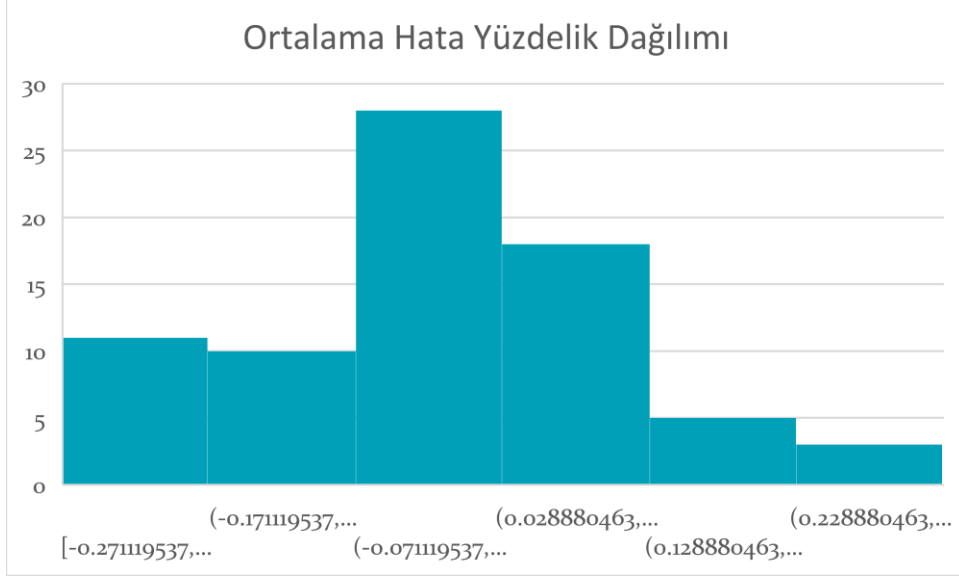
Ortalama Hata: 0,09

Maksimum Hata Yüzdesi: %3,4949

Minimum Hata Yüzdesi: %0

Ortalama Hata Yüzdesi: %1,1855

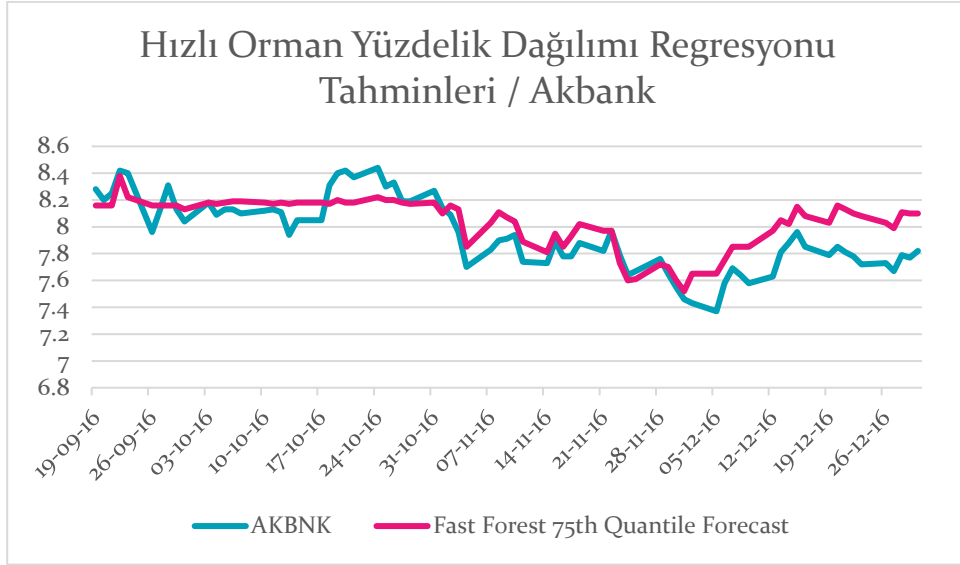
R^2 : 0.784449838



Tablo 25: Orta Vade Destekli Karar Ormanı Regresyonu Ortalama Hata Yüzdeleri Dağılımı

Sonuçlar destekli karar ormanı regresyonu algoritmasının gerçek değerleri ortalama 9 kuruş hata ile tahmin ettiğini göstermektedir. Yapılan en yüksek hata 29 kuruş, en düşük hata durumunda ise değerler tamamen doğru tahmin edilmiştir. Determinasyon katsayısı değeri 0,78'tir. Grafikler incelendiğinde destekli karar ormanı regresyonu algoritmasının gerçek değerlerin üzerinde ve altında değerler ürettiği görülmektedir. Bu sonuç hata oranı yüzdeleri dağılımı grafiğindeki değerlerin pozitif ve negatif olmasından görülebilir. Değerler çan eğrisine yakın bir sonuç sergilemektedir, bu sebeple alçak ve yüksek tahmin sayısının yakın olduğu söylenebilir. Sonuçlar uzun vade çalışma sonuçları ile kıyaslandığında performansın arttığı görülmektedir.

4.2.5 Hızlı Orman Yüzelik Dağılım Regresyonu (Fast Forest Quantile Regression)



Tablo 26: Orta Vade Hızlı Orman Dağılım Regresyonu / AKBNK

Maksimum Hata: 0,36

Minimum Hata: 0

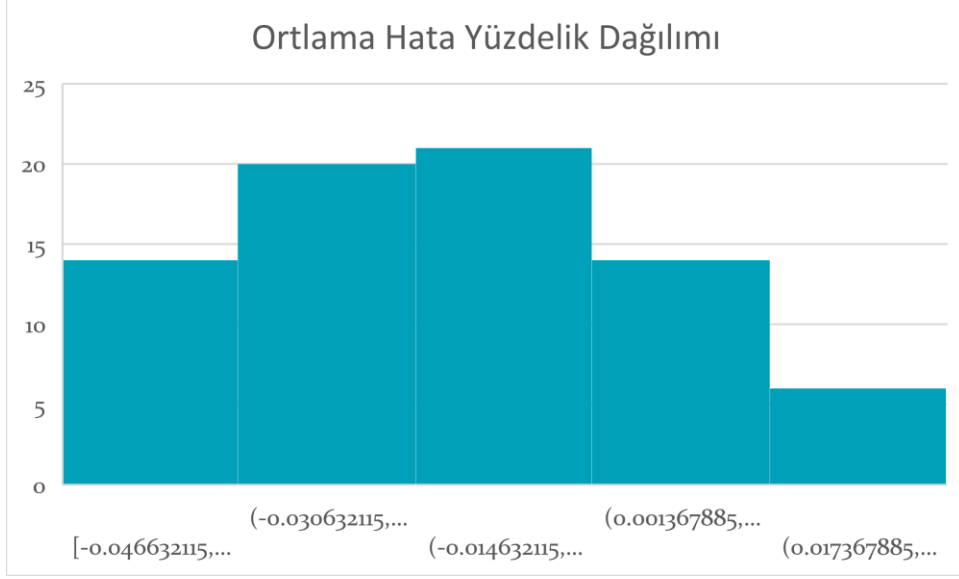
Ortalama Hata: 0,15

Maksimum Hata Yüzdesi: %4,6632

Minimum Hata Yüzdesi: %0

Ortalama Hata Yüzdesi: %1,8709

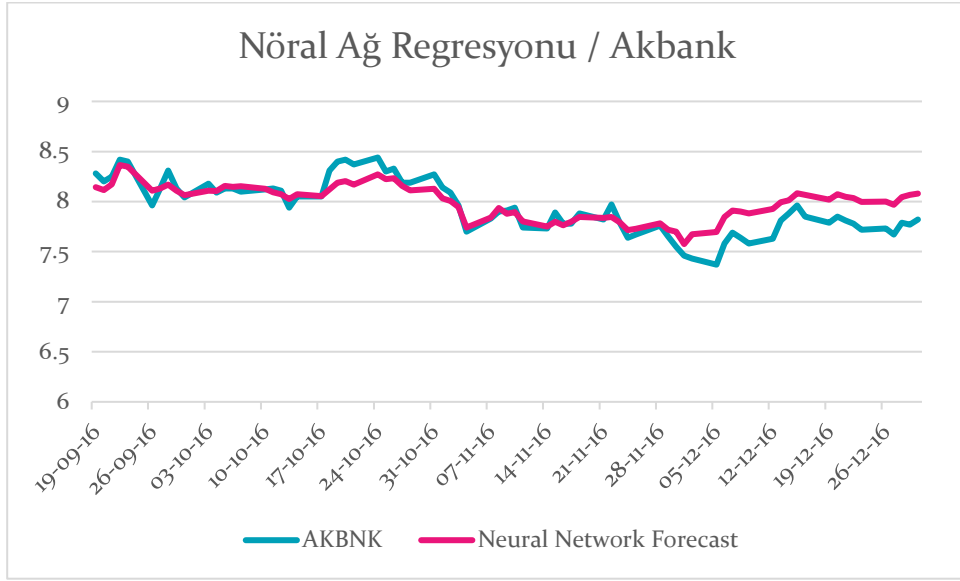
R^2 : 0.674864477



Tablo 27: Orta Vade Hızlı Orman Yüzdelerik Regresyonu Ortalama Hata Yüzdelerik Dağılımı

Sonuçlar hızlı orman yüzdelerik dağılım regresyonu algoritmasının gerçek değerleri ortalama 15 kuruş hata ile tahmin ettiğini göstermektedir. Yapılan en yüksek hata 36 kuruş, en düşük hata durumunda ise değerler tamamen doğru tahmin edilmiştir. Determinasyon katsayısı değeri 0,78'tir. Grafikler incelendiğinde hızlı orman yüzdelerik dağılım regresyonu algoritmasının gerçek değerlerin üzerinde ve altında değerler ürettiği görülmektedir. Bu sonuç hata oranı yüzdelerik dağılımı grafiğindeki değerlerin pozitif ve negatif olmasından görülebilir. Değerlerin büyük çoğunlukla negatif olması sebebi ile algoritmanın gerçek değerlerin üzerinde tahmin üretme eğiliminde olduğu söylenebilir. Sonuçlar uzun vade çalışma sonuçları ile kıyaslandığında performansın arttığı görülmektedir fakat şaşırtıcıdır ki determinasyon katsayısı değeri düşüş göstermiştir.

4.2.6 Nöral Ağ Regresyonu (Neural Network Regression)



Tablo 28: Orta Vade Nöral Ağ Regresyonu / AKBNK

Maksimum Hata: 0,33

Minimum Hata: 0

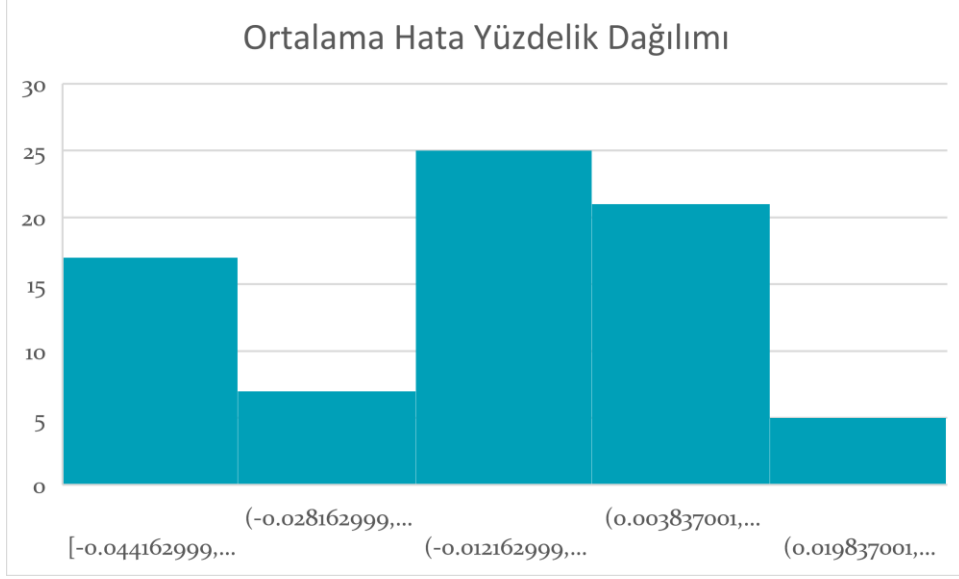
Ortalama Hata: 0,12

Maksimum Hata Yüzdesi: %4,4162

Minimum Hata Yüzdesi: 0

Ortalama Hata Yüzdesi: %1,5178

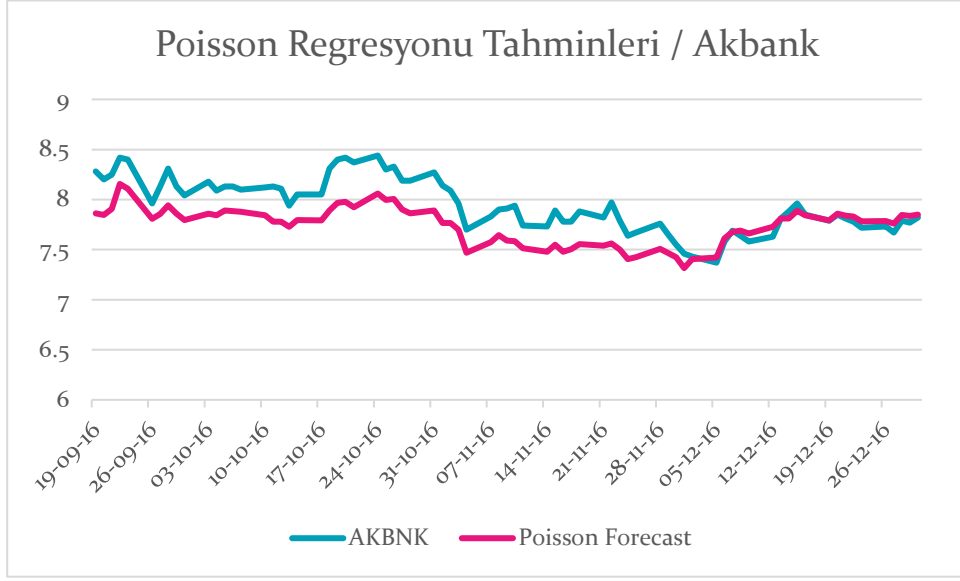
R^2 : 0.731668337



Tablo 29: Orta Vade Nöral Ağ Regresyonu Ortalama Hata Yüzdeleri Dağılımı

Sonuçlar nöral ağ regresyonu algoritmasının gerçek değerleri ortalama 12 kuruş hata ile tahmin ettiğini göstermektedir. Yapılan en yüksek hata 33 kuruş, en düşük hata durumunda ise değerler tamamen doğru tahmin edilmiştir. Determinasyon katsayısı değeri 0,78'tir. Grafikler incelendiğinde nöral ağ regresyonu algoritmasının gerçek değerlerin üzerinde ve altında değerler ürettiği görülmektedir. Bu sonuç hata oranı yüzdeleri dağılımı grafiğindeki değerlerin pozitif ve negatif olmasından görülebilir. Değerlerin büyük çoğunlukla negatif olması sebebi ile algoritmanın gerçek değerlerin üzerinde tahmin üretme eğiliminde olduğu söylenebilir. 29-11-16 tarihi sonrasındaki tüm değerler gerçek değerlerin üzerindedir. Sonuçlar uzun vade çalışma sonuçları ile kıyaslandığında performansın arttığı görülmektedir.

4.2.7 Poisson Regresyonu (Poisson Regression)



Tablo 30: Orta Vade Poisson Regresyonu / AKBNK

Maksimum Hata: 0,44

Minimum Hata: 0

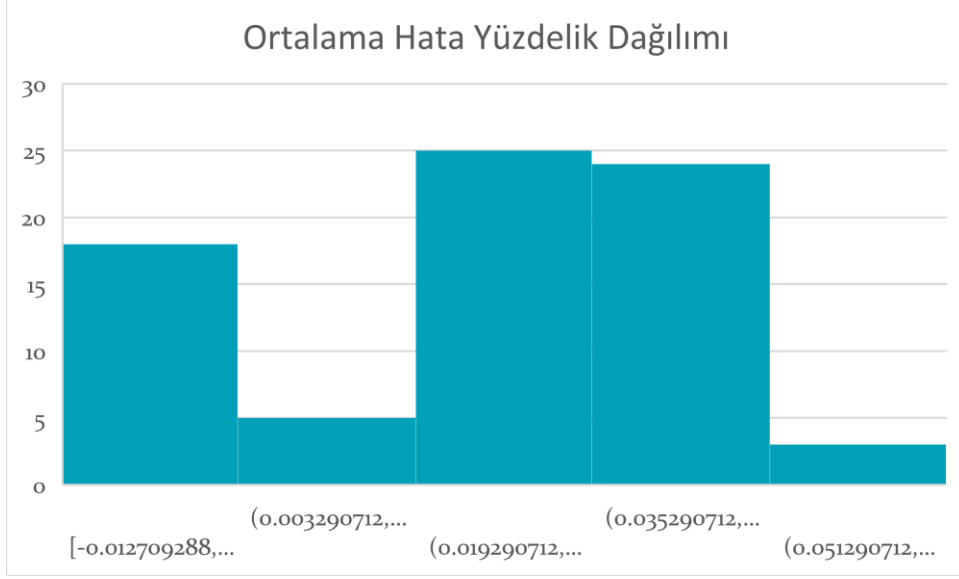
Ortalama Hata: 0,22

Maksimum Hata Yüzdesi: %5,3430

Minimum Hata Yüzdesi: %0

Ortalama Hata Yüzdesi: %2,7990

R^2 : 0.638519958



Tablo 31: Orta Vade Poisson Regresyonu Ortalama Hata Yüzdeleri Dağılımı

Sonuçlar Poisson regresyonu algoritmasının gerçek değerleri ortalama 22 kuruş hata ile tahmin ettiğini göstermektedir. Yapılan en yüksek hata 44 kuruş, en düşük hata durumunda ise değerler tamamen doğru tahmin edilmiştir. Determinasyon katsayısı değeri 0,63'tür. Grafikler incelendiğinde Poisson regresyonu algoritmasının gerçek değerlerin üzerinde ve altında değerler ürettiği görülmektedir. Bu sonuç hata oranı yüzdeleri dağılımı grafiğindeki değerlerin pozitif ve negatif olmasından görülebilir. Değerlerin büyük çoğunlukla pozitif olması sebebi ile algoritmanın gerçek değerlerin altında tahmin üretme eğiliminde olduğu söylenebilir. Sonuçlar uzun vade çalışma sonuçları ile kıyaslandığında performansın arttığı görülmektedir.

4.2.8 Değerlendirme

Sonuçlar karşılaştırıldığında aşağıdaki tablo ortaya çıkmaktadır:

	Doğrusal	Bayes Doğrusal	Karar Ağacı	Destekli Karar Ormanı	Hızlı Orman Yuzdelik	Nöral Ağ	Poisson
Max Hata	0.260889328	0.255078539	0.44375	0.294278069	0.359999924	0.3254813	0.447210158
Min Hata	0.000611409	0.001228983	0.0025	0.000746689	0	0.000574951	0.001001052
Ortalama Hata	0.104295674	0.096346734	0.192441856	0.094179507	0.14746664	0.119374047	0.225274606
Max Hata %	3.419257248	3.200483551	5.631345178	3.494988943	4.663211451	4.416299864	5.34301264
Min Hata %	0	0	0.032175032	0	0	0	0.012850475
Ortalama Hata %	1.327412413	1.225109118	2.412946633	1.185523631	1.87091774	1.517824993	2.799022892
R2	0.758842068	0.802050265	0.753092385	0.784449838	0.674864477	0.731668337	0.638519958

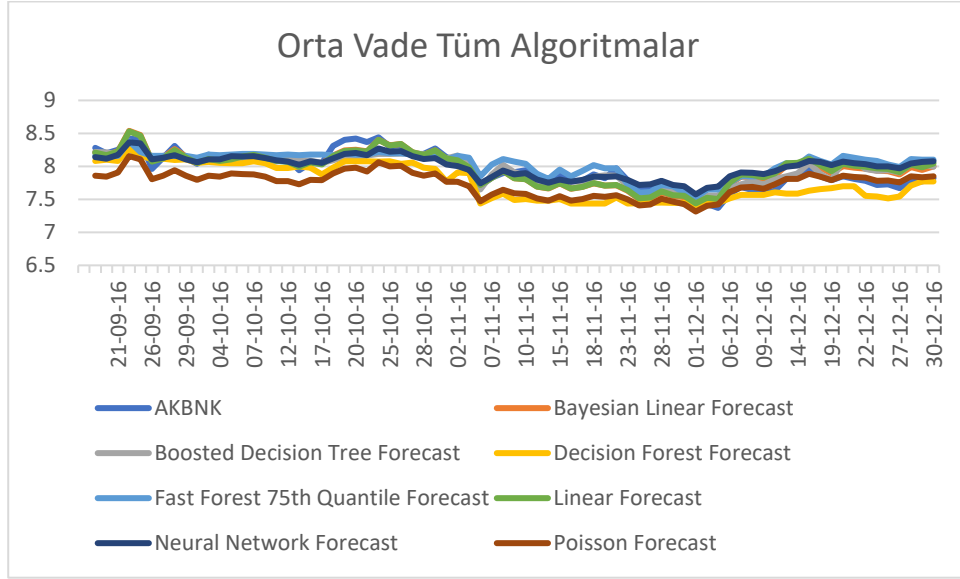
Tablo 32: Orta Vade Sonuç Değerlendirme

Ortalama hata kolunu karşılaştırıldığında bu deney için en etkin tahminleri 9 kuruş ortalama hata ve %1,18 ortalama hata yüzdesi ile Destekli Karar Ormanı Regresyonu algoritmasının ürettiği görülmektedir.

En iyi ikinci ve üçüncü algoritmalar ise %1,2251 ortalama hata ve %1,3274 ortalama hata yüzdesi ile Bayes Doğrusal ve Doğrusal Regresyon algoritmaları oldukları görülmektedir. Bu iki algoritmanın ortalama hata yüzdelik dağılımları incelendiğinde ise Bayes Doğrusal Regresyonun ufak bir farkla performansının daha yüksek olduğu görülmektedir. Bu sebeple Bayes Doğrusal Regresyon Algoritması ikinci, Doğrusal Regresyon Algoritması üçüncü sıradadır.

Dördüncü sırada ise doğrusal algoritmalara çok yakın bir ortalama hata oranı olan %1,5178 ile Nöral Ağ Regresyonu algoritması öne çıkmaktadır. %1,8709 hata ortalaması ile Fast Forest Quantile Regresyonu algoritması beşinci, %2,4129 hata oranı ile Karar Ağacı Regresyonu algoritması altıncı, %2,7990 hata oranı ile ise Poisson Regresyonu algoritması son sıradadır.

Deney sonuçları bir önceki deney ile karşılaştırıldığında tüm tahminlerin ilk deneye kıyasla daha etkin olduğu görülmektedir. Birinci deneydeki en düşük ortalama hata %2,14 ile Hızlı Orman Yüzelik Dağılım algoritması iken, ikinci deneyde %1,18 hata oranı ile Destekli Karar Ormanı algoritmasıdır. Dikkat çeken bir diğer sonuçta iki deneyde öne çıkan algoritmaların farklı algoritmalar olmasıdır. Tüm algoritmaların tahminleri aşağıdaki grafikte gösterilmiştir:

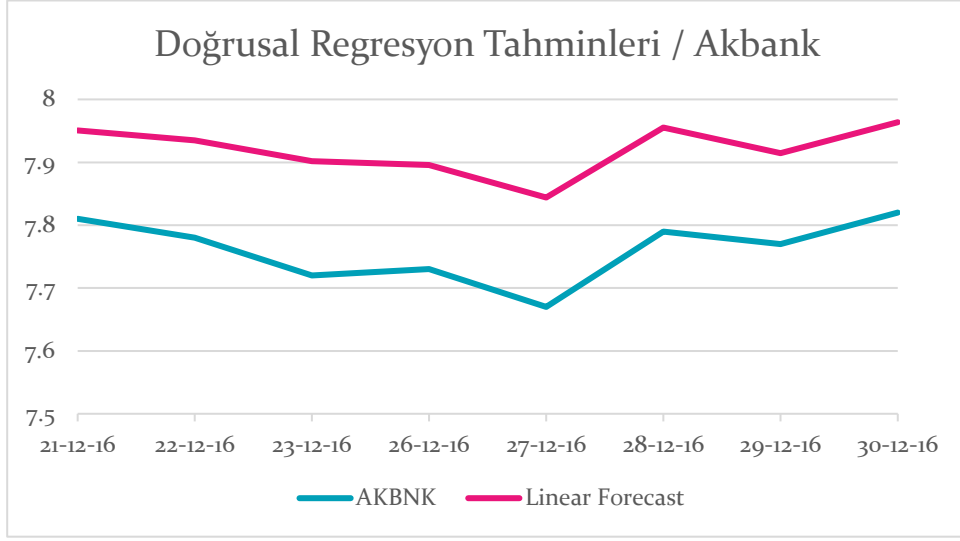


Tablo 33: Orta Vade Tüm Algoritma Tahminleri / AKBNK

4.3 Kısa Vade Öngörüleri

Son çalışmada maksimum öğrenmeyi gerçekleştirme durumu için verilerin %99'i öğrenme için kullanılmış, kalan %1 tahmin edilmiş ve gerçek veriler ile karşılaştırılmıştır. 755 günlük verinin 747 günü ile programlar eğitilmiş, kalan 8 gün tahmin edilmiştir. Tüm algoritmalar için tahmine karşılık gerçek değerlerin grafikleri, hata oranları, hata yüzdeleri, determinasyon katsayıları ve hata yüzdesi grafikleri gösterilmiştir. Tüm çalışmada görülen en yüksek performans beklenmektedir, bu sebeple bu sonuçlar ile yatırım yapılması durumu da incelenmiştir. Çalışma Microsoft Azure servisi üzerinde kiralanan bir sunucuda yapılmıştır. Tüm algoritmalar 1 saatten kısa bir sürede tamamlanmaktadır.

4.3.1 Doğrusal Regresyon (Linear Regression)



Tablo 34: Kısa Vade Linear Regresyon / AKBNK

Maksimum Hata: 0,1816

Minimum Hata: 0,1437

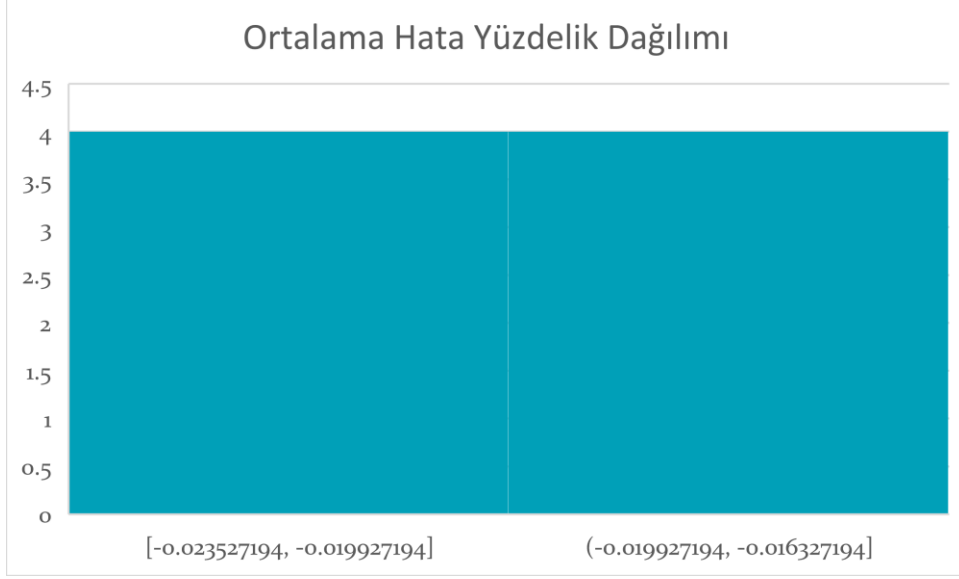
Ortalama Hata: 0,1613

Maksimum Hata Yüzdesi: %2,3527

Minimum Hata Yüzdesi: %1,8383

Ortalama Hata Yüzdesi: %2,0819

R²: 0.947917794



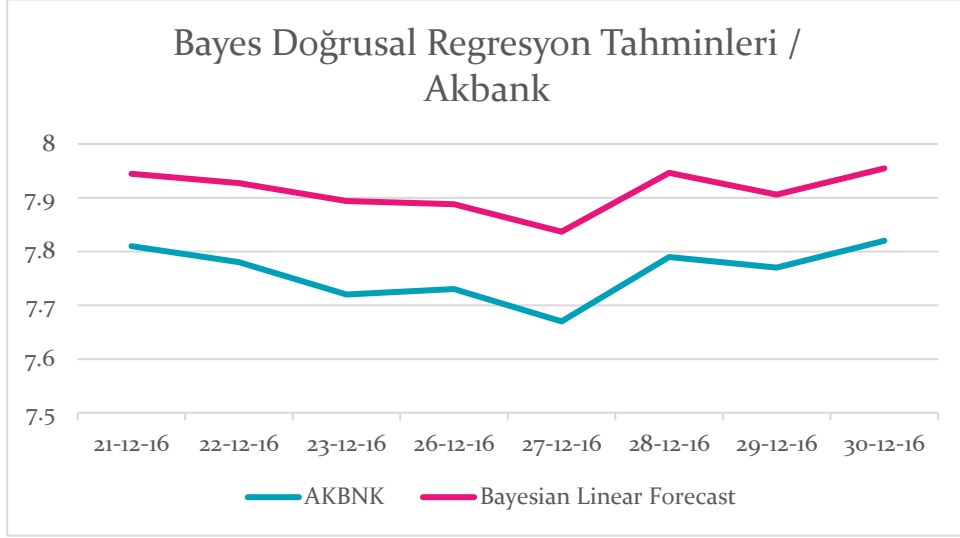
Tablo 35: Kısa Vade Doğrusal Regresyon Ortalama Hata Yüzdeleri Dağılımı

Sonuçlar doğrusal regresyon algoritmasının gerçek değerleri ortalama 16 kuruş hata ile tahmin ettiğini göstermektedir. Yapılan en yüksek hata 18 kuruş, en düşük hata durumunda ise 14 kuruşluk bir fark görülmektedir. Determinasyon katsayısı değeri 0,94'tür. Grafikler incelendiğinde doğrusal regresyon algoritmasının devamlı olarak gerçek değerlerin üzerinde değerler ürettiği görülmektedir. Bu sonuç hata oranı yüzdeleri dağılımı grafiğindeki tüm değerlerin pozitif olmasından görülebilir. Bir önceki çalışma ile kıyaslandığında determinasyon katsayısı ve maksimum hata metriklerinde gelişme vardır fakat ortalama hata bağlamında daha kötü sonuç verdiği görülmektedir. 8 hareketin 7'sinin yönü doğru tahmin edilmiştir. Fiyat hareket şiddeti de doğru tahmin edilmiştir fakat fiyatın kendisi gerçek değerden uzaklaşmıştır.

Elinde bu bilgiler bulunan ve 21-12-16 tarihinde elinde 100 birim parası olan mantıklı bir yatırımcı:

21-12-16 tarihinde gelecekte fiyatın düşeceği öngörüsü ile 27-12-16 tarihine kadar satın alma işlemi yapmaz. Fiyatın artacağı bilgisi ile 27-12-16 tarihinde 7,67 fiyatından satın alma işlemi gerçekleştirir ve fiyatın düşeceği bilgisi ile 28-12-16 tarihinde 7,79 fiyatından satar. Aynı mantık ile 29'unda 7,77 fiyatından satın alır ve 30'unda 7,82 fiyatından satar. Bu durum 8 günlük bir sürede %2.21 kar sağlar.

4.3.2 Bayes Doğrusal Regresyon (Bayes Linear Regression)



Tablo 36: Kısa Vade Bayes Linear Regresyon / AKBNK

Maksimum Hata: 0,1739

Minimum Hata: 0,1347

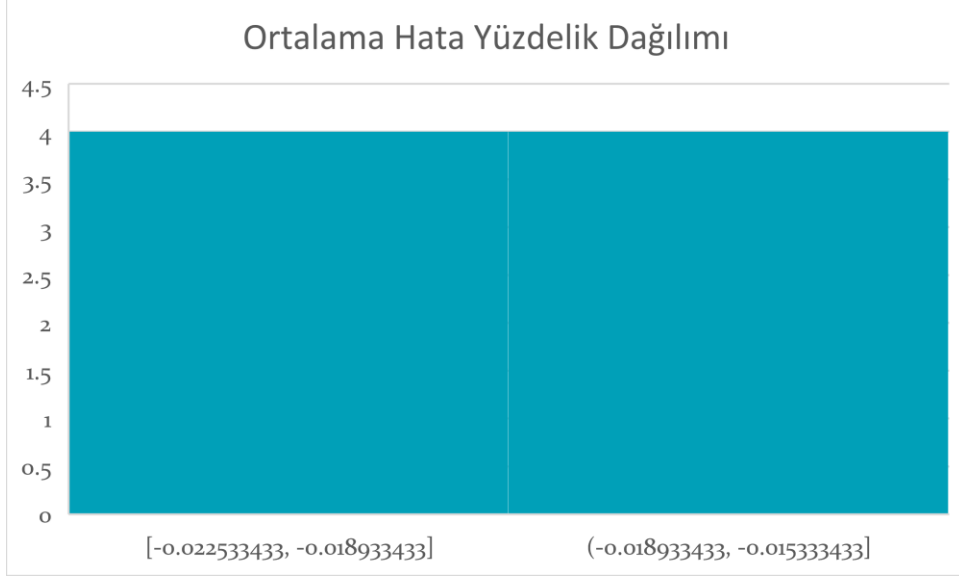
Ortalama Hata: 0,1533

Maksimum Hata Yüzdesi: %2,2533

Minimum Hata Yüzdesi: %1,7223

Ortalama Hata Yüzdesi: %1,9782

R^2 : 0.945942497

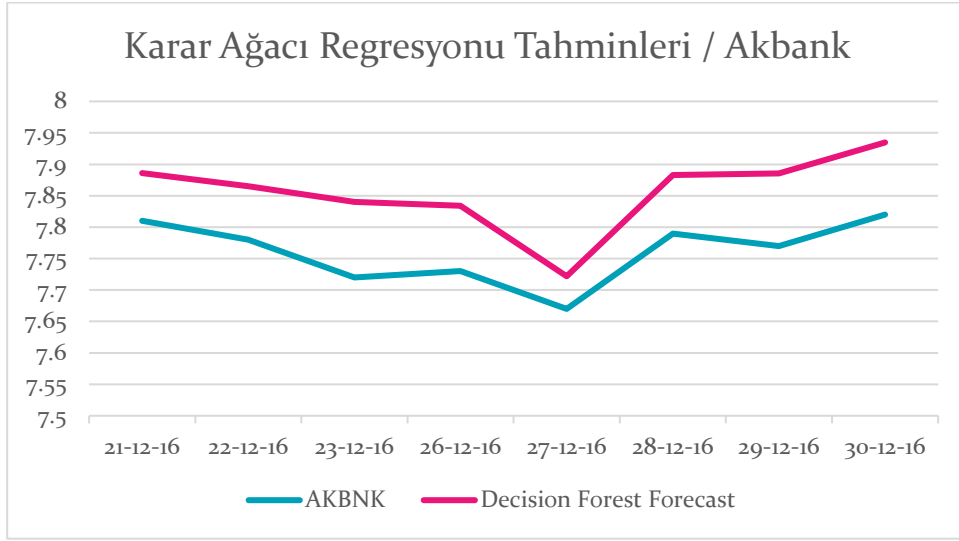


Tablo 37: Kısa Vade Bayes Doğrusal Regresyon Ortalama Hata Yüzdeleri Dağılımı

Sonuçlar Bayes doğrusal regresyon algoritmasının gerçek değerleri ortalama 15 kuruş hata ile tahmin ettiğini göstermektedir. Yapılan en yüksek hata 17 kuruş, en düşük hata durumunda ise değerler tamamen doğru tahmin edilmiştir. Determinasyon katsayısı değeri 0,94'tür. Grafikler incelendiğinde Bayes doğrusal regresyon algoritmasının devamlı olarak gerçek değerlerin üzerinde değerler ürettiği görülmektedir. Bu sonuç hata oranı yüzdeleri dağılımı grafiğindeki değerlerin tamamının negatif olmasından görülebilir. Bir önceki çalışma ile kıyaslandığında determinasyon katsayısı ve maksimum hata metriklerinde gelişme vardır fakat ortalama hata bağlamında daha kötü sonuç verdiği görülmektedir. 8 hareketin 7'sinin yönü doğru tahmin edilmiştir. Aynı doğrusal regresyon algoritmasında olduğu gibi fiyat hareket şiddeti de doğru tahmin edilmiştir fakat fiyatın kendisi gerçek değerden uzaklaşmıştır.

Elinde bu bilgiler bulunan ve 21-12-16 tarihinde elinde 100 birim parası olan mantıklı bir yatırımcı doğrusal regresyon algoritması ile aynı kararları verir ve 8 gün için %2.21 kar sağlar.

4.3.3 Karar Ağacı Regresyonu (Decision Tree Regression)



Tablo 38: Kısa Vade Karar Ağacı Regresyonu / AKBNK

Maksimum Hata: 0,1201

Minimum Hata: 0,0519

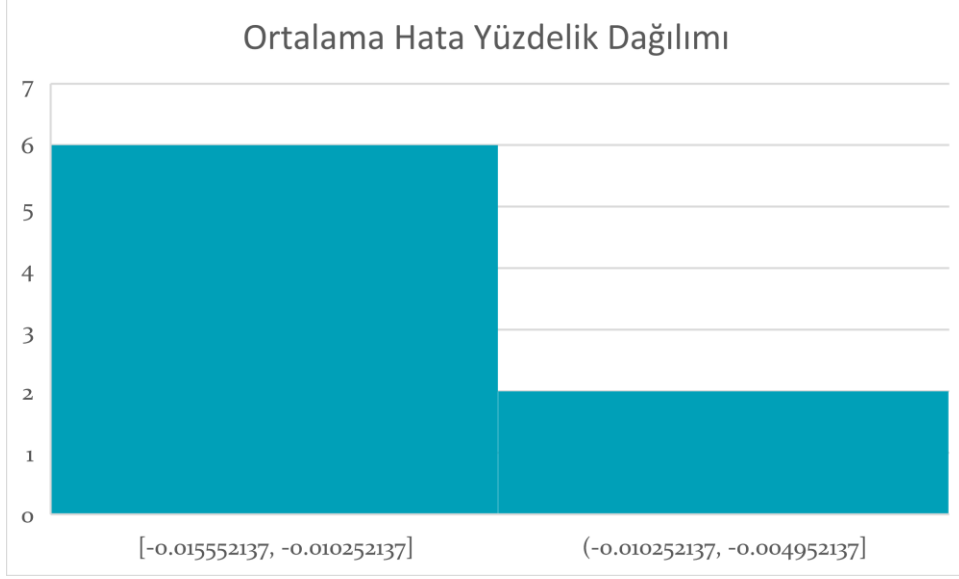
Ortalama Hata: 0,0977

Maksimum Hata Yüzdesi: %1,5552

Minimum Hata Yüzdesi: %0,6771

Ortalama Hata Yüzdesi: %1,2599

R²: 0.90691177



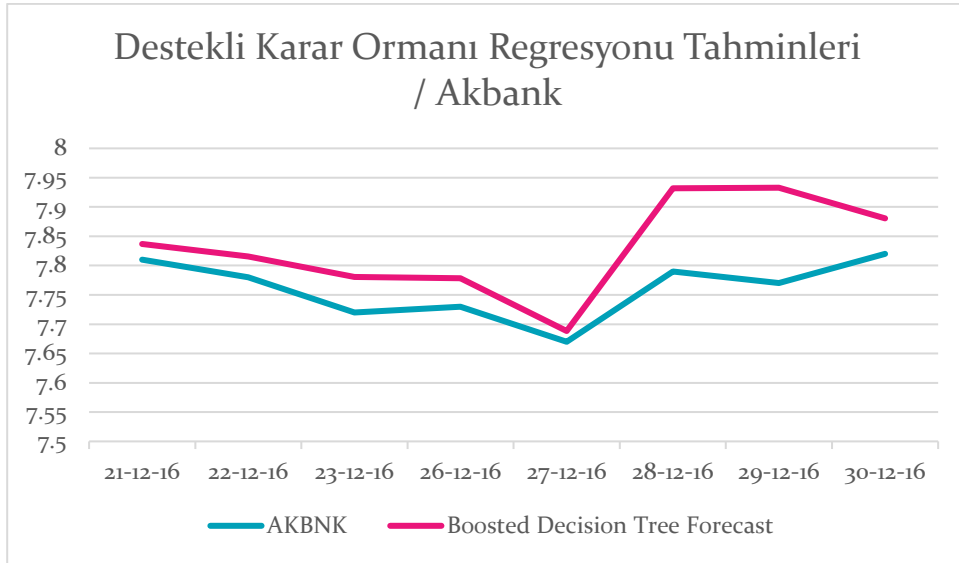
Tablo 39: Kısa Vade Karar Ağacı Regresyonu Ortalama Hata Yüzdeleri Dağılımı

Sonuçlar karar ağacı regresyonu algoritmasının gerçek değerleri ortalama 9 kuruş hata ile tahmin ettiğini göstermektedir. Yapılan en yüksek hata 12 kuruş, en düşük hata durumunda ise 5 kuruş hata yapılmıştır. Determinasyon katsayısı değeri 0,91'dir. Grafikler incelendiğinde karar ağacı regresyonu algoritmasının devamlı olarak gerçek değerlerin üzerinde değerler ürettiği görülmektedir. Bu sonuç hata oranı yüzdeleri dağılımı grafiğindeki tüm değerlerin negatif olmasından görülebilir. Sonuçlar uzun vade çalışma sonuçları ile kıyaslandığında performansın arttığı görülmektedir. Doğrusal algoritmalar ile kıyaslandığında ortalama hata sonuçlar daha iyi olsa da tahmin edilen 8 gün de 2 defa fiyat hareketinin yönü hatalı tahmin edilmiştir.

Elinde bu bilgiler bulunan ve 21-12-16 tarihinde elinde 100 birim parası olan mantıklı bir yatırımcı:

27-12-16 tarihine kadar alım işlemi yapmaz ve 23-26 tarihleri arasında 7.74'ten alıp 7.75'ten satma fırsatını kullanmaz. 27-12-16 tarihinde 7,67 fiyatından alım işlemi gerçekleştirir ve bir gün sonra 7,79 değerinde satar. Aynı mantık ile 28'inde 7.77 fiyatından alıp 29'unda 7.82 fiyatından satarak 8 günlük bir sürede %2.21 kar sağlar

4.3.4 Destekli Karar Ormanı Regresyonu (Boosted Decision Forest Regression)



Tablo 40: Kısa Vade Destekli Karar Ormanı Regresyonu / AKBNK

Maksimum Hata: 0,1628

Minimum Hata: 0,0183

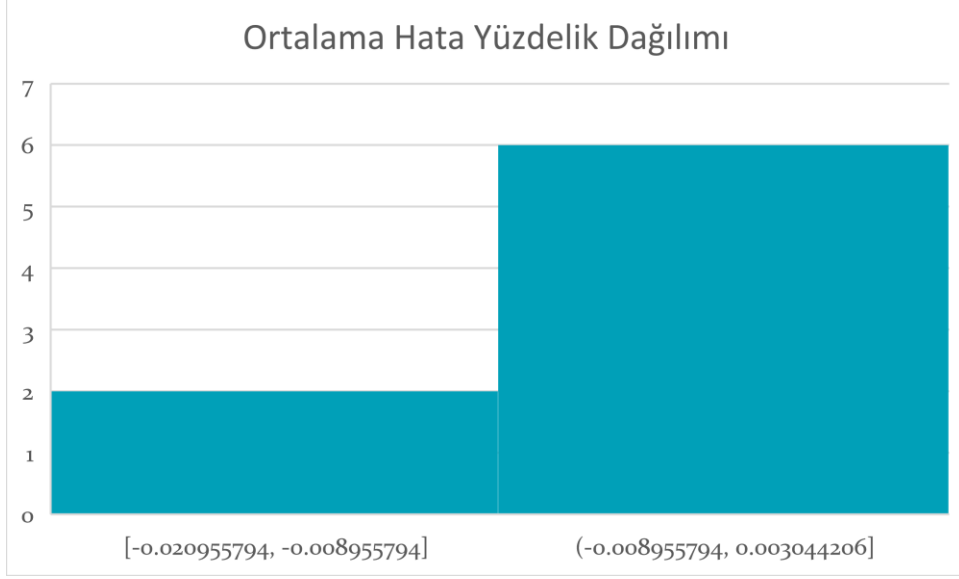
Ortalama Hata: 0,0755

Maksimum Hata Yüzdesi: %2,0955

Minimum Hata Yüzdesi: %0,2392

Ortalama Hata Yüzdesi: %0,972

R^2 : 0.709958244



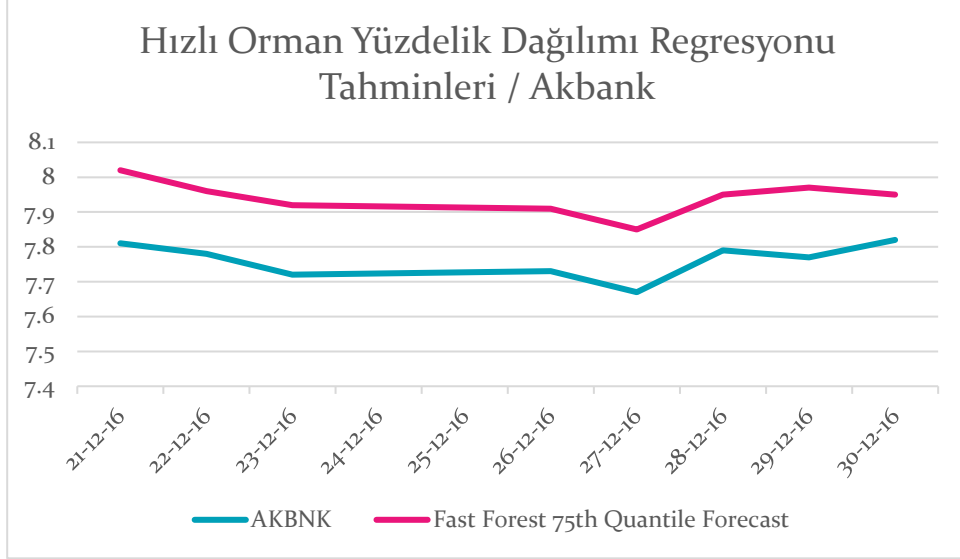
Tablo 41: Kısa Vade Destekli Karar Ormanı Regresyonu Ortalama Hata Yüzdeleri Dağılımı

Sonuçlar destekli karar ormanı regresyonu algoritmasının gerçek değerleri ortalama 7 kuruş hata ile tahmin ettiğini göstermektedir. Yapılan en yüksek hata 16 kuruş, en düşük hata durumunda ise 1 kuruş hata yapılmıştır. Determinasyon katsayısı değeri 0,7 'tir. Grafikler incelendiğinde destekli karar ormanı regresyonu algoritmasının devamlı gerçek değerlerin üzerinde değerler ürettiği görülmektedir. Bu sonuç hata oranı yüzdeleri dağılımı grafiğindeki tüm değerlerin pozitif olmasından görülebilir. Sonuçlar uzun vade çalışma sonuçları ile kıyaslandığında performansın arttığı görülmektedir. Sonuçlar önceki tüm algoritmalarından iyi olsa da bu algortmada da 8 hareket yönünün 2'si hatalı tahmin edilmiştir.

Elinde bu bilgiler bulunan ve 21-12-16 tarihinde elinde 100 birim parası olan mantıklı bir yatırımcı:

27-12-16 tarihine kadar alım işlemi yapmaz ve 23-26 tarihleri arasında 7.74'ten alıp 7.75'ten satma fırsatını kullanmaz. 27-12-16 tarihinde 7,67 fiyatından alım işlemi gerçekleştirir ve bir gün sonra 7,79 değerinde satar. Sonrasında fiyatın düşeceği düşüncesi ile alım satım işlemi yapmaz ve 8 günlük sürede %1.56 kar sağlar.

4.3.5 Hızlı Orman Yüzdelerik Dağılım Regresyonu (Fast Forest Quantile Regression)



Tablo 42: Kısa Hızlı Orman Yüzdelerik Regresyonu / AKBNK

Maksimum Hata: 0,2000

Minimum Hata: 0,1299

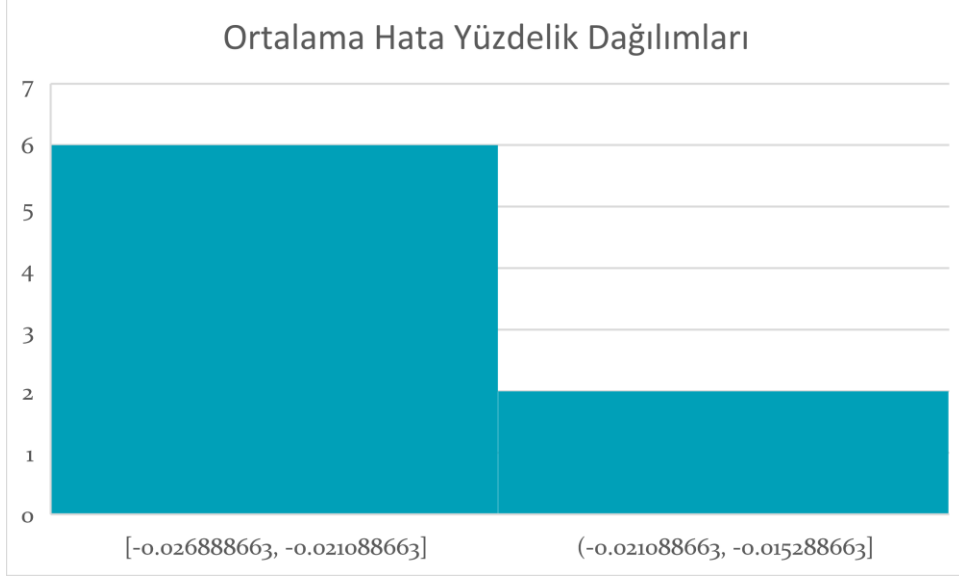
Ortalama Hata: 0,1757

Maksimum Hata Yüzdesi: %2,5906

Minimum Hata Yüzdesi: %1,6624

Ortalama Hata Yüzdesi: %2,2671

R^2 : 0.771738914



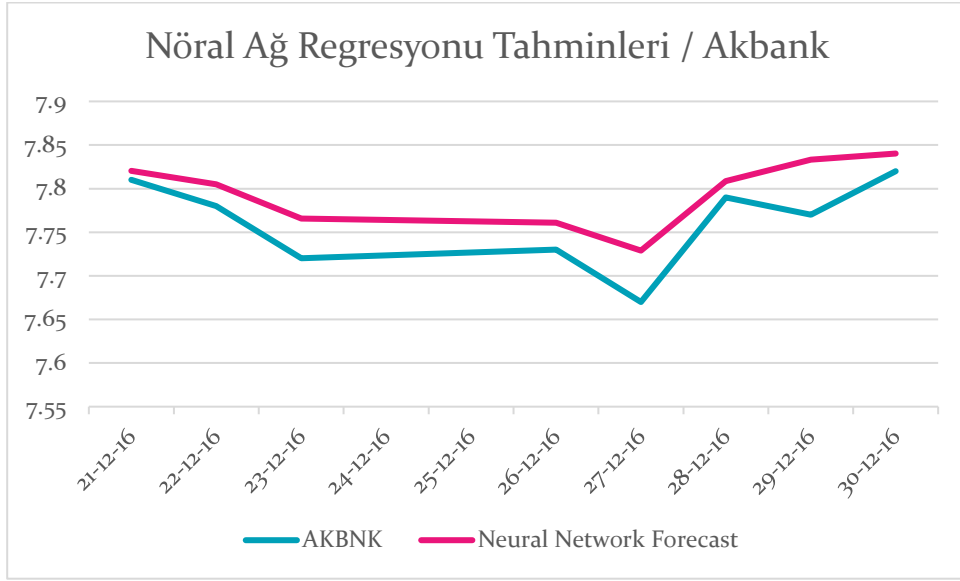
Tablo 43: Kısa Vade Hızlı Orman Yüzdelerik Regresyonu Ortalama Hata Yüzdelerik Dağılımı

Sonuçlar hızlı orman yüzdelerik dağılım regresyonu algoritmasının gerçek değerleri ortalama 17 kuruş hata ile tahmin ettiğini göstermektedir. Yapılan en yüksek hata 20 kuruş, en düşük hata durumunda ise 13 kuruşluk hata görülmektedir. Determinasyon katsayısı değeri 0,78'tir. Grafikler incelendiğinde hızlı orman yüzdelerik dağılım regresyonu algoritmasının devamlı gerçek değerlerin üzerinde değerler ürettiği görülmektedir. Bu sonuç hata oranı yüzdelerik dağılımı grafiğindeki tüm değerlerin pozitif olmasından görülebilir. Sonuçlar uzun vade çalışma sonuçları ile kıyaslandığında performansın düştüğü görülmektedir. Bu çalışmada ilk defa bir algoritmada artan öğrenme süresi ile düşen sonuçlar görülmüştür.

Elinde bu bilgiler bulunan ve 21-12-16 tarihinde elinde 100 birim parası olan mantıklı bir yatırımcı:

27-12-16 tarihine kadar fiyatın düşeceği düşüncesi ile alım satım işlemi yapmaz. 27'sinde 7,67 fiyatından satın alır ve fiyatın artacağı düşüncesi ile 29'una kadar kadar elinde tutar. Bu tarihte 7,77 fiyatından satar ve 8 günlük sürede %1.30 kar sağlar.

4.3.6 Nöral Ağ Regresyonu (Neural Network Regression)



Tablo 44: Kısa Vade Nöral Ağ Regresyonu / AKBNK

Maksimum Hata: 0,0609

Minimum Hata: 0,0185

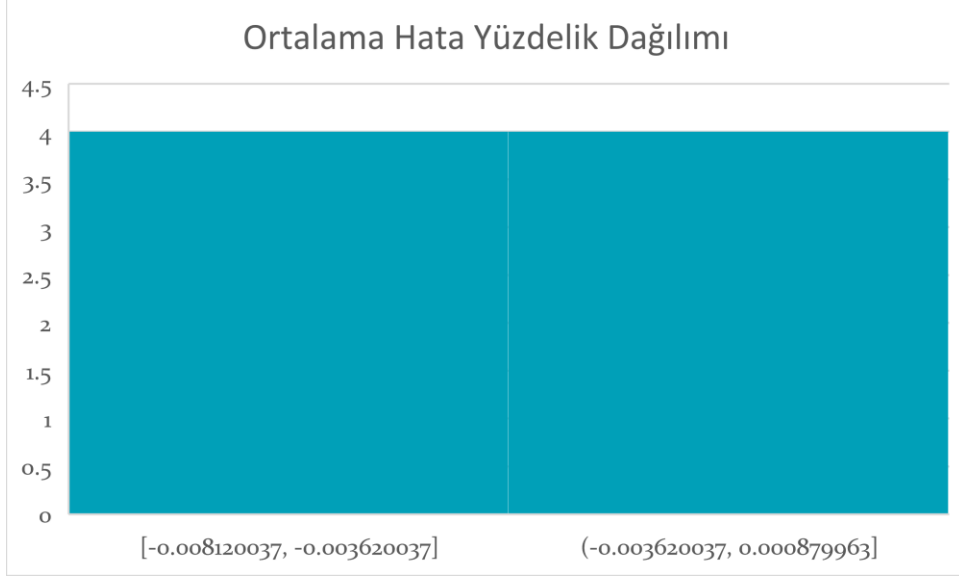
Ortalama Hata: 0,0374

Maksimum Hata Yüzdesi: 0,8120

Minimum Hata Yüzdesi: 0,2378

Ortalama Hata Yüzdesi: 0,4842

R²: 0.88445494



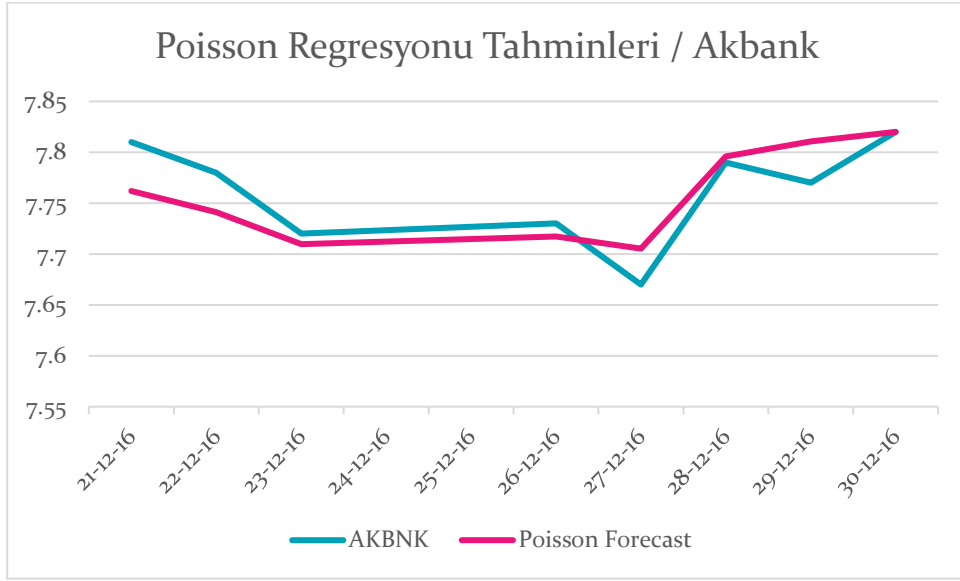
Tablo 45: Kısa Vade Nöral Ağ Ortalama Hata Yüzdeleri Dağılımı

Sonuçlar nöral ağ regresyonu algoritmasının gerçek değerleri ortalama 4 kuruş hata ile tahmin ettiğini göstermektedir. Yapılan en yüksek hata 6 kuruş, en düşük hata durumunda ise 2 kuruş hata görülmektedir. Determinasyon katsayısı değeri 0,88'tir. Grafikler incelendiğinde nöral ağ regresyonu algoritmasının devamlı olarak gerçek değerlerin üzerinde değerler ürettiği görülmektedir. Bu sonuç hata oranı yüzdeleri dağılımı grafiğindeki tüm değerlerin negatif olmasından görülebilir. Sonuçlar uzun vade çalışma sonuçları ile kıyaslandığında performansın arttığı görülmektedir. Öğrenme süresi arttıkça bu yöntemin performansı ciddi şekilde artmıştır. Buna rağmen Nöral ağ regresyonu da 8 hareket yönünden 2'sini hatalı tahmin etmiştir.

Elinde bu bilgiler bulunan ve 21-12-16 tarihinde elinde 100 birim parası olan mantıklı bir yatırımcı:

27-12-16 tarihine kadar fiyatın düşeceği düşüncesi ile alım satım işlemi yapmaz. 27'sinde 7,67 fiyatından satın alır ve 30'unda 7,82 fiyatından satarak 8 günlük sürede %1,95 kar sağlar.

4.3.7 Poisson Regresyonu (Poisson Regression)



Tablo 46: Kısa Vade Poisson Regresyonu / AKBNK

Maksimum Hata: 0,0408

Minimum Hata: 0

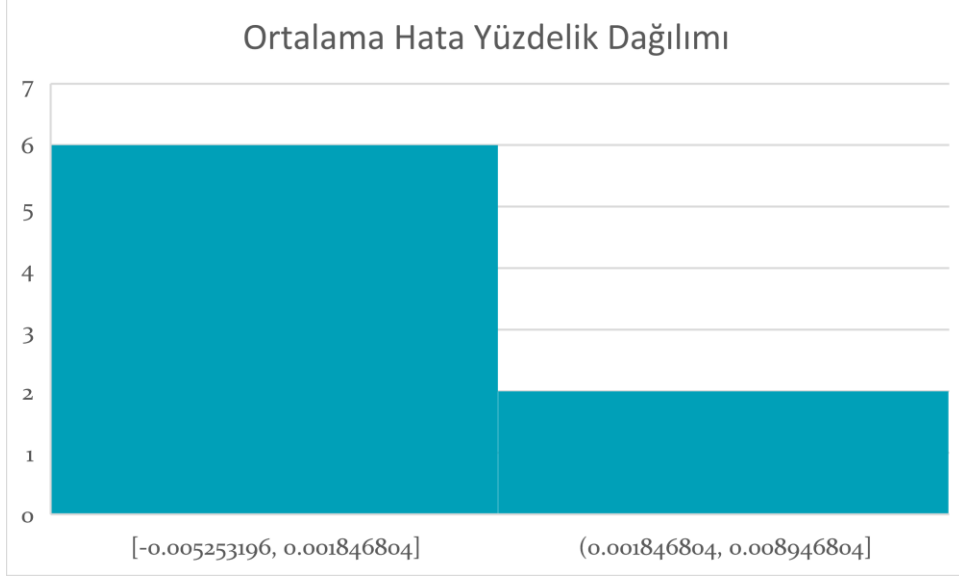
Ortalama Hata: 0,0205

Maksimum Hata Yüzdesi: 0,5232

Minimum Hata Yüzdesi: 0

Ortalama Hata Yüzdesi: 0,2654

R²: 0.718343493



Tablo 47: Kısa Vade Poisson Ortalama Hata Yüzdeleri Dağılımı

Sonuçlar Poisson regresyonu algoritmasının gerçek değerleri ortalama 2 kuruş hata ile tahmin ettiğini göstermektedir. Yapılan en yüksek hata 4 kuruş, en düşük hata durumunda ise değerler tamamen doğru tahmin edilmiştir. Determinasyon katsayısı değeri 0,72'dir. Grafikler incelendiğinde Poisson regresyonu algoritmasının gerçek değerlerin üzerinde ve altında değerler ürettiği görülmektedir. Bu sonuç hata oranı yüzdeleri dağılımı grafiğindeki değerlerin pozitif ve negatif olmasından görülebilir. Değerlerin büyük çoğunlukla negatif olması sebebi ile algoritmanın gerçek değerlerin üzerinde tahmin üretme eğiliminde olduğu söylenebilir. Sonuçlar uzun vade çalışma sonuçları ile kıyaslandığında performansın ciddi oranda arttığı görülmektedir. İlk çalışmada en kötü sonuçları veren Poisson regresyonu algoritması, son çalışmada hem en iyi tahminleri yapmış hem de 8 hareket yönünün 7'sini doğru tahmin etmiştir.

Elinde bu bilgiler bulunan ve 21-12-16 tarihinde elinde 100 birim parası olan mantıklı bir yatırımcı:

23-12-16 tarihinde 7,72 fiyatından satın alır ve 26'sında 7,73 fiyatından satar. Sonrasında 27'sinde 7,67 fiyatından satın alır 30'unda 7,82 fiyatından satarak 8 günlük sürede %2,08 kar sağlar.

4.3.8 Değerlendirme

Sonuçlar karşılaştırıldığında aşağıdaki tablo ortaya çıkmaktadır. Ortalama hata değeri 2-3 kuruş seviyesine kadar düşürülmüştür.

	Doğrusal	Bayes Doğrusal	Karar Ağacı	Destekli Karar Ormanı	Hızlı Orman Yuzdelik	Nöral Ağ	Poisson
Max Hata	0.18163	0.173958101	0.1200625	0.162826519	0.200000076	0.06309	0.040817
Min Hata	0.143757	0.134689462	0.0519375	0.018346863	0.129999809	0.01853	0
Ortalama Hata	0.161376	0.153325966	0.0977708	0.075512848	0.175714182	0.03748	0.02055
Max Hata %	2.352719	2.253343277	1.5552137	2.095579395	2.59067456	0.812	0.52532
Min Hata %	1.838323	1.722371637	0.6771512	0.239202907	1.66240165	0.23787	0
Ortalama Hata %	2.08198	1.978211167	1.2599896	0.972011062	2.267143786	0.48422	0.265449
R2	0.947918	0.945942497	0.9069118	0.709958244	0.771738914	0.88445	0.718343

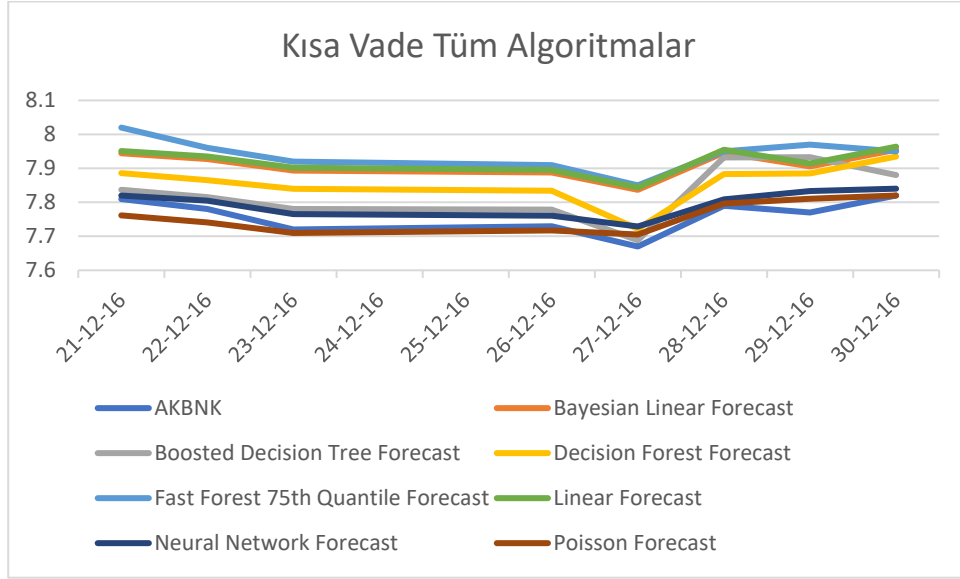
Tablo 48: Kısa Vade Sonuç Değerlendirme

Ortalama hata kolunu karşılaştırıldığında bu çalışma için en etkin tahminleri 2 kuruş ortalama hata ve %0,2654 ortalama hata yüzdesi ile Poisson Regresyonu algoritmasının ürettiği görülmektedir. İkinci sırada ise 3 kuruş ortalama hata ve %0,4842 ortalama hata yüzdesi ile Nöral Ağ Regresyonu algoritması bulunmaktadır. Bu iki algoritmanın bir önceki çalışmalardaki performansları incelendiğinde son sırada oldukları görülmektedir. İki algoritma da artan öğrenme süresi ile performanslarını yükseltmiştir.

Üçüncü ve dördüncü sırada 7 kuruş ortalama hata ile Destekli Karar Ağacı Regresyonu algoritması ve 9 kuruş ortalama hata ile Karar Ağacı Regresyonu algoritmaları bulunmaktadır. Bu algoritmalar da artan öğrenme süresi ile daha yüksek performans göstermişlerdir.

Beşinci ve altıncı sırada 15 kuruş ortalama hata ile Bayes Doğrusal Regresyon ve 16 kuruş ortalama hata ile Doğrusal Regresyon algoritmaları bulunmaktadır. Kısa vadede son sırayı 17 kuruş ortalama hata ile Hızlı Orman Yüzdilik Dağılım regresyonu almıştır. Tüm deneyler kıyaslandığında Bayes Doğrusal Regresyon algoritmasının düzenli olarak Doğrusal Regresyon algoritmasından üstün performans gösterdiği görülmektedir. Dikkat çeken bir özellik, doğrusal algoritmaların yüksek determinasyon katsayısı değerleri ve tüm hareket yönlerini doğru tahmin etmiş olmalarıdır.

İlginçtir ki Hızlı Orman Yüzdelik Dağılım algoritması eğitim / tahmin sürelerinden bağımsız olarak hep benzer sonuçlar göstermiştir: Ortalama hata oranı birinci çalışmada %2,14, ikinci çalışmada %1,87, üçüncü çalışmada ise %2.26'dır. Birinci çalışmada ilk sırayı alan bu algoritma, son çalışmada sonuncu sıradadır. Bu algoritma için artan öğrenme süresi performans artışı anlamına gelmemektedir. Tüm algoritmaların tahminleri aşağıdaki grafikte gösterilmiştir:



Tablo 49: Kısa Vade Tüm Algoritma Tahminleri / AKBNK

Tahminler alım satım kararları için kullanıldığında doğrusal bazlı algoritmalar ile 8 günlük sürede %2,21 kar sağlamanın mümkün olduğu gösterilmiştir. Belirtilen 8 günlük sürede sağlanabilecek maksimum kar %2,35'tir. Bu durum yapa zeka regresyonlarının arbitraj yaratma hedefi için kullanılabileceğini göstermektedir.

BÖLÜM V. SONUÇ, ÖNERİLER ve GENEL DEĞERLENDİRME

Bu çalışma finans alanında arbitraj yaratma hedefinin ilk adımı olan geleceği öngörebilme konusunu modern yapay zeka araçlarının yöntemleri ile incelemesi bağlamında bilgisayar mühendisliği ve finans disiplinlerini içeren özgün bir çalışmadır. Çalışma finans dünyasının iki temel hipotezi olan rassal yürüyüş ve etkin piyasalar hipotezlerine karşı çıkan hedefler içerir, fakat iki hipotezde de günümüz piyasasında doğru olmayan bölümler tespit edilmiştir.

Çalışma için IMKB 30 firmaları 2014-2016 tarihleri arası gün sonu fiyat verileri kullanılmış ve zaman serileri farklı özellikleri sebebi ile seçilen 7 regresyon algoritması gerekli donanımı sağlayan bir sunucu üzerinde çalıştırılmıştır.

Deneylerin sonucunda aşağıdaki bulgular elde edilmiştir.

- Her koşulda etkili tahmin yapan tek bir yapay zeka algoritması yoktur. Her algoritma farklı ortamlarda, farklı öğrenme süresi ve volatilité değerleri ile karşılaştığında farklı performans gösterir.
- Uzun dönem tahminlerinde Doğrusal Regresyon ve Hızlı Orman Yüzdélik Regresyonu gibi ortalama değerler üzerinden yola çıkan algoritmalar bu özellikleri sebebi ile öne çıkmaktadırlar. Uzun vadede alçak ve yüksek tahminler artacağı için ortalama tahminler daha etkin görünmektedir.
- Hızlı Orman Yüzdélik Regresyonu algoritması sonuçları her zaman daha uzun öğrenme süresinin daha etkin performans anlamına gelmediğini gösterir. Bu durum yapay zekada aşırı öğrenme (overfitting) olarak geçer. Öğrenme süresinde doyuma ulaşıldıktan sonra öğrenmeye devam edilmesi algoritmayı var olmayan paternler bulmaya zorlayabilir.
- Doğrusal regresyon algoritmalarının yeterince öğrenme gerçekleştikten sonra artan determinasyon katsayısı ile fiyat hareket yönlerini etkili tahmin ettikleri gözlemlenmiştir.
- Nöral ağ ve Poisson regresyonları yabancı kaynaklarda gösterilen çalışmalarda gözlemlendiği gibi yeterince öğrenme gerçekleştikten sonra etkin performans göstermektedir.
- Sonuç olarak fiyat hareket yönü ve hareketin şiddetini tahmin etmek için belirtilen yöntemler kullanılabilir, yöntemler Burton Marliel'in iddia ettiğinin aksine kullanılamaz değildir.

Bu alanda yapılacak gelecek çalışmalara öneriler:

- Bazı algoritmalar devamlı olarak gerçek değer üzerinde / altında tahminler yapmaktadır. Bu algoritmalar birlikte kullanılıp daha üstün performans göstermeleri sağlanabilir.
- Yeni bilgisayar donanımları çıkması ile bugün “sonuç alınamayacak süre” isteyen algoritmalar gelecekte kullanışlı olabilir. Yeni donanımlar çıktıkça bugüne kadar doğru veya yanlış kabul edilen birçok bilgi yeniden gözden geçirilmelidir. Son zamanlarda hız kazanan kuantum işlemci çalışmaları, işlem kapasitesi ve örnek uzay simülasyonları bağlamında mevcut bilgisayarlardan çok daha kuvvetli bir görüntü çizmektedir. Şüphesizdir ki bu işlemcilerle çok daha fazla veri çok daha hızlı işlenecek ve daha etkili sonuçlar elde edilecektir. Çalışmada belirtildiği gibi donanımsal özellikler sonuçlar için önemlidir.
- Bu çalışmada 7 farklı yapay zeka algoritması kullanılmıştır. Bunların haricinde denenebilecek birçok algoritma vardır. Referans gösterilen kaynaklarda Hidden Markow Model, Support Vector Machine, Long-Short Term Memory Neurons ve Resilient Back Propagation gibi algoritmaların farklı piyasalarda etkin performans gösterdiklerine değinilmiştir.
- Bu çalışmada farklı öğrenme süreleri denenmiş, bazı algoritmalarda performans öğrenme süresi ile artmış, bazı algoritmalarda düşmüştür. Farklı öğrenme süreleri denenerek her algoritmanın maksimum performansı için doyuma ulaşma süresi araştırılabilir.
- Bu çalışma arbitraj yaratma hedefinin ilk adımı olan gelecek fiyatları öngörme üzerinedir. Arbitraj yaratma amacını göstermek için fiyat artacak / azalacak bilgilerine bağlı alım satım emirlerinin karlılıkları gösterilmiştir. Arbitraj yaratma hedefi için elde edilen sonuçlar ile optimal yatırım ve maksimum dönüş sağlayan optimizasyon algoritmaları araştırılabilir, alma satma işlemleri karın maksimize edilmeye çalışıldığı bir “oyun” olarak yapay zekaya tanımlanıp bu optimizasyon bir kez daha yapay zeka algoritmaları sayesinde yapılabilir. Önceden programlanan koşullar ile kendi kendine alım satım yapan programlara “zincir emir” programları denmektedir. Bu tezde elde edilen tahmin verileri, belirtilen optimizasyon işlemi yapıldıktan sonra zincir emir programları için kullanılabilir.

KAYNAKLAR

- Adams, R. (2004). Intelligent Advertising. *AI & Society*, 68-81.
- Akın, F. (2002). *Kalitatif Tercih Modelleri Analizi*. Bursa: Seçkin Yayıncılık.
- Alpaydin, E. (2020). *Introduction to Machine Learning*. Massachusetts: Massachusetts Institute of Technology.
- Altay, E., & Satman, H. (2005). Stock Market Forecasting: Artificial Neural Network and Linear Regression. *Journal of Financial Management and Analysis*, 18-33.
- Bachelier, L. (2007). *Theory of Speculation: The Origins of Modern Finance*. Princeton: Princeton University Press.
- Bayraktar, A. (2012). Etkin Piyasalar Hipotezi. *Aksaray Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 37-47.
- Burris, S. N. (2019). *Boole's Algebra of Logic 1847*. Ontario: University of Waterloo.
- Campbell, M., Hoane, J. A., & Hsu, F.-h. (2001). Deep Blue. *Elsevier*, s. 57-83.
- Chen, A.-S., & Leung, M. (2004). Regression neural network for error correction in foreign exchange forecasting and trading. *Computers and Operations Research*, 1049-1068.
- Cornish, W. H., & Fowler, P. R. (1977). Coproducts of De Morgan algebras. *Bulletin of the Australian Mathematical Society* 16, 1-13.
- Delgrande, J. P., & John, M. (1986). *Fundamentals of Artificial Intelligence*. Springer-Verlag Berlin Heidelberg.
- DeMorgan, A. (1847). Formal Logic. A. DeMorgan içinde, *Formal Logic* (s. 150 - 165). London: Taylor and Walton.
- Deniz, O. (2005, 1). Poisson Regresyon Analizi. *İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi*, s. 59-72.
- Dondurmacı, G. A., & Çınar, A. (2014). Finans Sektöründe Veri Madenciliği Uygulaması. *Akademik Sosyal Araştırmalar Dergisi*, 258-271.
- Draper, N. R., & Smith, H. (1998). *Applied Regression Analysis*. New York: Wiley-Interscience.
- Ewald, W. (2019). The Emergence of First-Order Logic. *The Stanford Encyclopedia of Philosophy*. içinde Stanford: Stanford University.
- Fama, E. (1970). Efficient Capital Markets: A Review of Theory and Empirical Work. *Journal of Finance*, 283-417.
- Faria, S., & Goncalves, F. (tarih yok). Financial data modeling by Poisson mixture regression.

- Freund, R. J., Wilson, W. J., & Sa, P. (2006). *Regression Analysis: Statistical Modeling of a Response Variable*. Burlington: Elsevier.
- Galbraith, J. R. (1977). Organization Design. J. R. Galbraith içinde, *Organization Design* (s. 436). Pennsylvania: Addison-Wesley Publishing Company.
- George E. P. Box, G. M. (2015). *Time Series Analysis: Forecasting and Control*. New Jersey: Wiley.
- Goia, A., May, C., & Fusai, G. (2009). Functional clustering and linear regression for peak load forecasting. *International Journal of Forecasting*.
- Güdelek, M. U. (2019). *Zaman Serisi Analizi ve Tahmini: Derin Öğrenme Yaklaşımı*. Ankara: Tobb Ekonomi ve Teknoloji Üniversitesi.
- Gultekin, F. (2013). *Regresyon Analizi*. Balıkesir: Balıkesir Üniversitesi.
- Gun, O., & Bernard, G. (2011). Efficient Market Hypothesis: What are we talking about. *Real World Economics Review*, 19-30.
- Hao, L., & Naiman, D. (2007). *Quantile Regression*. California: Sage Publications.
- Heinen, A. (2000). *Modelling Time Series Count Data: An Autoregressive Conditional Poisson Model*. California: University of California.
- Heshmaty, V., & Kandel, A. (1985). Fuzzy linear regression and its applications to forecasting in uncertain environment. *Fuzzy Sets and Systems*, 159-191.
- Hromkovic, J. (2005). *Design and Analysis of Randomized Algorithms*. Berlin: Springer.
- Jayasekare, R., Gill, R., & Lee, K. (2016). Modeling discrete stock price changes using a mixture of Poisson distributions. *Journal of the Korean Statistical Society*, 409-421.
- Johnson, R., & Zhang, T. (2014). *Learning Nonlinear Functions Using Regularized Greedy Forest*. New Jersey: Rutgers University.
- Jordan, M. I., Ghahramani, Z., & Saul, L. K. (1997). Hidden Markov Decision Trees. *Neural Information Processing Systems* (s. 501-507). Vancouver: NIPS.
- Karan, M. B. (2001). *Yatırım Analizi ve Portföy Yönetimi*. Ankara: Gazi Kitabevi.
- Kolmogorov, A. N., & Uspenskii, V. A. (1987). Algorithms and Randomness. *Theory of Probability and its Applications*, s. 389-411.
- Koop, G. (2003). *Bayesian Econometrics*. London: John Wiley & Sons Inc.
- Koop, G., & Korobilis, D. (2010). *Bayesian Multivariate Time Series Methods for Empirical Macroeconomics*. Hanover: Now Publishers Inc.
- Laffont, J. J., & Maskin, E. (1990). The Efficient Market Hypothesis and Insider Trading on the Stock Market. *The Journal Of Political Economy*, 70-93.
- Leung, M., Chen, A.-S., & Daouk, H. (2000). Forecasting exchange rates using general regression neural networks. *Computers and Operations Research*, 1093-1110.

- Malkiel, B. (1973). *A Random Walk Down Wall Street*. New York: W. W. Norton & Company, Inc.
- Man-Chung, C., Chi-Cheong, W., & Chi-Chung, L. (2000). *Financial Time Series Forecasting by Neural Network Using Conjugate Gradient Learning Algorithm and Multiple Linear Regression Weight Initialization*. Hong Kong: The Hong Kong Polytechnic University.
- Masitrov, L. E. (2014). *Probability Theory: A Historical Sketch*. New York: Academic Press.
- Mazzola, E., & Muliere, P. (2011). Reviewing Alternative Characterizations of Meixner Process. *"Probability Surveys*, 127-154.
- Meek, C., Chickering, D. M., & Heckerman, D. (2002). Autoregressive Tree Models for Time-Series Analysis. *2002 SIAM International Conference on Data Mining*, (s. 229-244). Arlington.
- Meinshausen, N. (2006). Quantile Regression Forests. *Journal of Machine Learning Research*, 983-999.
- Microsoft. (2020, 03 11). *Microsoft*. Microsoft: <https://docs.microsoft.com/en-us/azure/machine-learning/how-to-deploy-fpga-web-service> adresinden alındı
- Nannavecchia, A. (2015). The Meixner Process for Financial Data. *Megatrend Revija*, 33-44.
- Nvidia. (2020, March 11). *Nvidia GeForce*. Nvidia: <https://www.nvidia.com/tr-tr/geforce/graphics-cards/rtx-2060-super/> adresinden alındı
- Patel, J. K., & Read, C. B. (1996). *Handbook of the Normal Distribution*. New York: Marcel Dekker.
- Pole, A., West, M., & Harrison, J. (1994). *Applied Bayesian Forecasting and Time Series Analysis*. New York: Chapman & Hall.
- Poyarkov, A., Drutsa, A., Khalyavin, A., Gusev, G., & Serdyukov, P. (2016). Boosted Decision Tree Regression Adjustment for Variance Reduction in Online Controlled Experiments. *22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (s. 235-244). New York: Association for Computing Machinery.
- Pradeepkumar, D., & Ravi, V. (2017). Forecasting financial time series volatility using Particle Swarm Optimization trained Quantile Regression Neural Network. *Applied Soft Computing*, 35-52.
- Rescher, N. (1964). Introduction to Logic. *Journal of Symbolic Logic*, 579.
- Rokach, L. (2016). Decision forest: Twenty years of research. *Information Fusion*, 111-125.
- Rosenberg, A. (2011). *Philosophy of Science*. Routledge.

- Rycroft, C. H., & Bazant, M. Z. (2005). Introduction to Random Walks and Diffusion. C. H. Rycroft, & M. Z. Bazant içinde, *Introduction to Random Walks and Diffusion* (s. 1-6). Massachusetts: MIT Department of Mathematics.
- Saygin, P., Cicekli, İ., & Akman, V. (2000). Turing Test: 50 Years Later. *Minds and Machines* 10, 463-518.
- Schoutens, W. (2002). *Theory and Applications in Finance: The Meixner Process*. Lueven: Eurandom.
- Şıklar, E. (2009). Regresyon Analizinde Bayeşçi Yaklaşım. *Anadolu Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 113-122.
- Siringano, J., & Spiliopoulos, K. (2017). Stochastic Gradient Descent in Continuous Time. *SIAM Journal on Financial Mathematics*, 933-961.
- Smith, C., McGuire, B., & Huang, T. (2006). The History of Artificial Intelligence.
- Smith, C., Ting Huang, Yang, G., & McGuire, B. (2006). *The History Of Artificial Intelligence*. Washington D.C.: University of Washington.
- Sowa, J. F. (2000). Knowledge Representation: Logical, Philosophical and Computational Foundations. J. F. Sowa içinde, *Knowledge Representation: Logical, Philosophical and Computational Foundations* (s. 45-83). California: Brooks/Cole.
- Specht, D. (1991). A General Regression Neural Network. *IEEE Transactions on Neural Network*, 568-576.
- Stordahl, K. (2007). The History Behind the Probability Theory and The Queing Theory. *Teletronikk*, 123-140.
- Taylor, S. J. (2008). *Modelling Financial Time Series*. London: World Scientific Publishing.
- Tunçel, A. K. (2007). Rassal Yürüyüş (Random Walk) Hipotezi'nin IMKB'de Test Edilmesi: Koşu Testi Uygulaması. *Gazi Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 1-18.
- Xu, M., Watanachaturaporn, P., Varshney, P., & Arora, M. (2005). Decision tree regression for soft classification of remote sensing data. *Remote Sensing of Environment*, 322-336.
- Yan, X., & Gang Su, X. (2009). *Linear Regression Analysis: Theory and Computing*. Singapore: World Scientific Publishing.
- Yu, K., & Moyeed, R. (2001). Bayesian quantile regression. *Statistics & Probability Letters*, 437-447.
- Zadeh, L. A. (1996). *Fuzzy Sets, Fuzzy Logic and Fuzzy Systems: Selected Papers by Lotfi A. Zadeh*. Danvers: World Scientific Publishing Co.