



T.C.

BURDUR MEHMET AKİF ERSOY ÜNİVERSİTESİ

SOSYAL BİLİMLER ENSTİTÜSÜ

YÖNETİM BİLİŞİM SİSTEMLERİ ANABİLİM DALI

**KRİPTO PARA DEĞERLERİNİN YAPAY ZEKÂ TEKNİKLERİ
İLE TAHMİNİ**

Furkan ATLAN

YÜKSEK LİSANS TEZİ

Danışman: Dr. Öğr. Üyesi İhsan PENÇE

BURDUR-2019

T.C.
BURDUR MEHMET AKİF ERSOY ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
YÖNETİM BİLİŞİM SİSTEMLERİ ANABİLİM DALI

KRİPTO PARA DEĞERLERİNİN YAPAY ZEKÂ TEKNİKLERİ
İLE TAHMİNİ

Furkan ATLAN
YÜKSEK LİSANS TEZİ

Danışman: Dr. Öğr. Üyesi İhsan PENÇE
Jüri: Dr. Öğr. Üyesi Melike ŞİŞECİ ÇEŞMELİ
Jüri: Dr. Öğr. Üyesi Sinan UĞUZ

BURDUR-2019



MAKÜ SOSYAL BİLİMLER
ENSTİTÜSÜ

YÜKSEK LİSANS JÜRİ ONAY FORMU

M.A.K.Ü Sosyal Bilimler Enstitüsü Yönetim Kurulu'nun 31.05.2019 tarih ve 2019/14 sayılı kararıyla oluşturulan jüri tarafından 24.06.2019 tarihinde tez savunma sınavı yapılan Furkan ATLAN'ın "Kripto Para Değerlerinin Yapay Zekâ Teknikleri ile Tahmini" konulu tez çalışması Yönetim Bilişim Sistemleri Anabilim Dalında YÜKSEK LİSANS tezi olarak kabul edilmiştir.

JÜRİ

ÜYE

(TEZ DANIŞMANI) : Dr. Öğr. Üyesi İhsan PENÇE

ÜYE

: Dr. Öğr. Üyesi Melike ŞİŞECİ ÇEŞMELİ

ÜYE

: Dr. Öğr. Üyesi Sinan UĞUZ

ONAY

M.A.K.Ü Sosyal Bilimler Enstitüsü Yönetim Kurulu'nun/...../..... tarih
ve/..... sayılı kararı.

İMZA/MÜHÜR

T.C.
BURDUR
MEHMET AKİF ERSOY ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ

ETİK BEYAN

Mehmet Akif Ersoy Üniversitesi Sosyal Bilimler Enstitüsü Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğine göre hazırlamış olduğum “Kripto Para Değerlerinin Yapay Zekâ Teknikleri ile Tahmini” adlı tezin hazırlanması sürecinde akademik etik ilkeleri ihlal etmediğimi taahhüt eder, tezimin kağıt ve elektronik kopyalarının Mehmet Akif Ersoy Üniversitesi Sosyal Bilimler Enstitüsü arşivlerinde aşağıda belirttiğim koşullarda saklanmasına izin verdiğimi onaylarım.

Sosyal Bilimler Enstitüsü Lisansüstü Eğitim-Öğretim Yönetmeliğinin ilgili maddeleri uyarınca gereğinin yapılmasını arz ederim.

☒ Tezimin tamamı her yerden erişime açılabilir.

☐ Tezim sadece Mehmet Akif Ersoy Üniversitesi yerleşkelerinde erişime açılabilir.

☐ Tezimin 3 yıl süreyle erişime açılmasını istemiyorum. Bu sürenin sonunda uzatma için başvuruda bulunmadığım takdirde, tezimin tamamı her yerden erişime açılabilir.

Furkan ATLAN

Tarih ve İmza

TEŞEKKÜR METNİ

Bu çalışmanın yapılmasında bana ilham veren, çalışmanın yapımı ve yazımı boyunca benden desteğini, bilgisini ve en önemlisi de zamanını esirgemeyen, beni yönlendiren ve geliştiren, bana yeni bir vizyon katan, talebesi olmaktan gurur duyduğum, üzerimdeki emeğini her daim minnetle hatırlayacağım değerli büyüğüm, sevgili hocam Dr. Öğr. Üyesi İhsan PENÇE'ye teşekkürü bir borç bilirim.

Lisans eğitimim boyunca bilişim ve programlamaya dair sahip olduğum tüm bilgi birikiminin mimarı ve bugünlere gelmemde büyük bir pay sahibi olan Dr. Öğr. Üyesi Melike ŞİŞECİ ÇEŞMELİ'ye teşekkür ederim.

Ayrıca, hayatım boyunca sevgilerini ve desteklerini benden esirgemeyen aileme ve dostlarıma teşekkür ederim.

Haziran, 2019

Furkan ATLAN

(ATLAN, *Furkan, Kripto Para Değerlerinin Yapay Zekâ Teknikleri ile Tahmini*, Yüksek Lisans Tezi, Burdur, 2019)

ÖZET

Bitcoin, blok zinciri teknolojisi altyapısı üzerine kurulmuş olan en popüler kripto para birimidir. Mevcut şartlarda kripto para borsasının en büyük hacmine sahip olan Bitcoin'in gelecekteki değerinin ne olacağı oldukça merak edilen bir konudur.

Bu çalışmada, Bitcoin/TL fiyatının Dolar/TL kuru ve tarih öznitelikleri ile değer tahmini yapılmıştır. Çalışmanın amacı, finans sektörü ve ticari işlemler ile yakından ilişkisi olan Bitcoin'in fiyat tahmininin gerçekleştirilerek hem kullanıcılara hem de yatırımcılara kısa ve orta vadede seçenekler sunarak, zamanında ve yerinde doğru yatırım yapılmasına yardımcı olmaktır.

Çalışmanın yapılmasında yapay sinir ağları (YSA), uyarlamalı sinirsel bulanık çıkarım sistemi (ANFIS), eğri uydurma ve uzun kısa süreli bellek (LSTM) yapay zekâ yöntemleri kullanılmıştır. Çalışmada birden fazla yapay zekâ ve derin öğrenme yönteminin kullanılmasının sebebi, özellikle görüntü işlemede çok yüksek performans değerlerine ve başarı oranlarına sahip olan derin öğrenmenin finansal verilerin analizinde başarı oranının analiz edilmek istenmesi ve kullanılan yöntemlerin performanslarının kıyaslanmasıdır.

Çalışmada kullanılan BTC/TL verileri Koineks şirketine ait web adresinden çekilip, Dolar/TL verileri ise Türkiye Cumhuriyeti Merkez Bankası'nın yayınladığı kur fiyatlarının günlük açılış değerleri baz alınarak oluşturulmuştur. Tarih özniteliği ardışık nümerik sayılar olarak başlangıç ve bitiş tarihleri dikkate alınarak oluşturulmuş olup, 08.12.2016 ve 08.12.2018 yılları arasını kapsamaktadır.

Yapılan analizler neticesinde, kullanılan yöntemler içerisinde özellikle eğri uydurma ve ANFIS'in Bitcoin/TL fiyat tahminindeki sonuçları oldukça anlamlı ve başarılı olarak değerlendirilmiştir. Deneysel çalışmalar sonucunda en iyi parametre ve zaman aralıklarında belirlenen algoritmalar ile gerçek zamanlı olarak diğer kripto para birimlerine ait değer tahmini de web ortamında yapılabilmektedir.

Anahtar Kelimeler: Kripto Para, Bitcoin, Yapay Zekâ, Derin Öğrenme, BTC Fiyat Tahmini, LSTM, ANFIS, Eğri Uydurma, YSA

(ATLAN, *Furkan, Forecasting of Cryptocurrency Prices Using Artificial Intelligence Techniques*, Master Thesis, Burdur, 2019)

ABSTRACT

Bitcoin is the most popular cryptocurrency unit built on the block chain technology infrastructure. Under current circumstances, the future value of Bitcoin, which has the largest volume of the cryptocurrency market, is highly curious.

In this study, the value of Bitcoin / TL price is predicted by using the dollar / TL rate and date attributes. The aim of the study is to provide a short and medium term options for both the users and the investors, by helping to make an accurate and timely investment in the price forecasting of Bitcoin, which is closely related to the financial sector and commercial transactions.

Artificial intelligence methods such as artificial neural network (ANN), adaptive neuro fuzzy inference system (ANFIS), curve fitting and long short-term memory (LSTM) are used in the study. The reason of using multiple artificial intelligence and deep learning method in the study is to analyze the success rate of analysis of financial data of deep learning which has very high performance values and success rates especially in image processing and to compare the performance of the methods used.

This study uses the data drawn from the web address of Koineks companies for getting BTC price and USD/TRY exchange rate are provided by the daily opening price value of the Central Bank of the Republic of Turkey. The date feature is based on the start and end dates as consecutive numerical numbers and covers the period between 08.12.2016 and 08.12.2018.

As a result of the analyzes, the Bitcoin/TL price forecasting is evaluated as significant and successful by using the curve fitting and ANFIS methods. As a result of experimental studies, it is able to forecast the rate of other cryptocurrencies in real time on the web with the algorithms determined within the best parameters and time intervals.

Keywords: Cryptocurrency, Bitcoin, Artificial Intelligent, Deep Learning, BTC Price Estimation, LSTM, ANFIS, Curve Fitting, ANN

İÇİNDEKİLER

İÇ KAPAK.....	i
TEZ ONAY SAYFASI.....	ii
ETİK BEYANI	iii
TEŞEKKÜR METNİ.....	iv
ÖZET.....	v
ABSTRACT	vi
İÇİNDEKİLER	vii
KISALTMALAR DİZİNİ	x
TABLolar DİZİNİ	xi
ŞEKİLLER DİZİNİ	xiii
GİRİŞ	1

BİRİNCİ BÖLÜM

BITCOIN VE BLOK ZİNCİRİ

1.1. Bitcoin ve Blok Zinciri ile İlgili Kavramlar.....	4
1.1.1. Kripto Para	4
1.1.2. Bitcoin	5
1.1.2.1. Bitcoin'in Tarihçesi.....	5
1.1.2.2. Bitcoin Sistem İşleyişi	6
1.1.3. Blok Zinciri (Blockchain)	8
1.1.3.1. Kriptografik Özet Fonksiyonu (SHA-256)	8
1.1.3.2. Eşler Arası Ağ Bağlantısı (Peer to Peer Network Connection)	9
1.1.3.3. İş Kanıtı (Proof of Work).....	10

1.1.3.4. Madenciler (Miners)	11
1.1.4. Kriptoloji.....	14

İKİNCİ BÖLÜM

LİTERATÜR ÇALIŞMASI

İKİNCİ BÖLÜM.....	15
LİTERATÜR ÖZETİ.....	15

ÜÇÜNCÜ BÖLÜM

MATERYAL VE METHOD

3.1. Kripto Para ve Öznitelik Verilerinin Elde Edilmesi	22
3.2. Yapay Zekâ	25
3.2.1. Sinir Ağları.....	26
3.2.1.1. Yapay Sinir Ağları (YSA).....	26
3.2.1.2. Tekrarlayan Sinir Ağları (RNN)	29
3.2.1.3. Uzun Kısa Vadeli Bellek (LSTM)	31
3.2.1.4. Konvolüsyonel Sinir Ağları (CNN)	34
3.2.2. Adaptif Ağ Tabanlı Bulanık Çıkarım Sistemi (ANFIS)	41
3.2.3. Eğri Uydurma (Curve Fitting).....	44
3.2.3.1. En Küçük Kareler Yöntemi.....	45
3.3. Öğrenme Algoritmaları	45
3.3.1. Stokastik Gradyan İnişi (Stochastic Gradient Descent)	46
3.3.2. RMSprop (Root Mean Square Prop) Algoritması.....	47
3.3.3. Adam (Uyarlanabilir Anlar-Adaptive Moments) Algoritması.....	48

DÖRDÜNCÜ BÖLÜM**DENEYSEL ÇALIŞMA**

4.1. En Uygun Yöntemlerin Belirlenmesi.....	53
4.2. Yöntemlerin Uygulanması ve Bulgular	57
4.2.1. YSA, ANFIS ve Eğri Uydurma Yöntemlerinin En İyi Parametrelerle Çapraz Doğrulama Kullanılarak Uygulanması	57
4.2.2. YSA, ANFIS ve Eğri Uydurma Yöntemlerinin En İyi Parametrelerle Zaman Serisi Kullanılarak Uygulanması	63
4.2.3. Gerçek Zamanlı Uygulama ve Sonuçları	76
SONUÇLAR	82
KAYNAKÇA	84
EKLER.....	92
ÖZGEÇMİŞ.....	93

KISALTMALAR DİZİNİ

ANFIS	: Adaptif Sinirsel Bulanık Çıkarım Sistemi
ARCH	: Oto Regresyon Koşullu Değişen Varyans
ARIMA	: Bütünleşik Oto Regresyon Hareketli Ortalama
BNN	: Bayesian Neural Network
BR	: Bayesian Regularization
BTC	: Bitcoin
CPU	: Merkezi İşlem Birimi
CRIX	: Kripto para indeksi
ENET	: Elastic-NET
ETH	: Ethereum
GARCH	: Genelleştirilmiş Oto Regresyon Koşullu Değişen Varyans
LSTM	: Uzun Kısa Vadeli Bellek
MA	: Hareketli ortalama
MSE	: Ortalama Karesel Hata
RMSE	: Ortalama Karesel Hataların Karekökü
RNN	: Tekrarlayan Sinir Ağları
SHA-256	: Security Hash Algorithm-256
TARCH	: Eşik Oto Regresyon Koşullu Varyans
TDNN	: Zaman Gecikmeli Sinir Ağları
USD	: Amerikan Doları
VAR	: Vektör Oto Regresyon
XGT	: Xgboosts
XRP	: Ripple
YSA	: Yapay Sinir ağları

TABLOLAR DİZİNİ

Tablo 1: 6 Ana Başlık Olarak Oluşturulan Veri Setleri ve Açıklamaları	23
Tablo 2: 1. Veri Setine Ait Bilgiler ve Açıklamalar	23
Tablo 3: 2. Veri Setine Ait Bilgiler ve Açıklamaları	24
Tablo 4: 3. Veri Setine Ait Bilgiler ve Açıklamaları	24
Tablo 5: 4. Veri Setine Ait Bilgiler ve Açıklamaları	24
Tablo 6: 5. Veri Setine Ait Bilgiler ve Açıklamaları	25
Tablo 7: 6. Veri Setine Ait Bilgiler ve Açıklamaları	25
Tablo 8: YSA Modeli ve Parametreleri.....	51
Tablo 9: ANFIS Yönteminde Elde Edilen Kurallar ve Parametreleri.....	52
Tablo 10: YSA'daki Algoritma ve Parametrelere Ait Toplam Test Hataları	53
Tablo 11: BR Algoritması İçin En İyi Parametrelerin Belirlenmesi.....	55
Tablo 12: ANFIS Kural Oranı ve Parametreleri	55
Tablo 13: ANFIS Yönteminde En Uygun Parametreler	56
Tablo 14: Eğri Uydurmaya Ait Parametreler ve Ortalama Test Hatası Karşılıkları	56
Tablo 15: Eğri Uydurma Yöntemi İçin Belirlenen En Uygun Parametreler.....	57
Tablo 16: Veri Setlerine Ait Hata Oranlarının Belirlenmesi	58
Tablo 17: En İyi Sonuçları Veren Veri Setleri.....	59
Tablo 18: Yöntemlere Ait Parametrelerin Birbirleri İle Kıyaslanması	60
Tablo 19: Zaman Serisi %90 Standart Veri Seti Sonuçları.....	63
Tablo 20: Zaman Serisi %90 Normalize Edilmiş Veri Seti Sonuçları.....	64
Tablo 21: Zaman Serisi %95 Standart Veri Seti Sonuçları.....	65
Tablo 22: Zaman Serisi %95 Normalize Veri Seti Sonuçları	66
Tablo 23: %95 Standart Zaman Serisi Verileri ve 5-Kat Çapraz Doğrulama Sonuçlarının Karşılaştırılması	67

Tablo 24: Yeniden Belirlenen En İyi Veri Seti Grupları ve Ortalama Hata Oranları.....	68
Tablo 25: Veri Seti Gruplarına Ait ANFIS ve Eğri Uydurma İstatistikleri	74
Tablo 26: Veri Seti Gruplarına Ait LSTM İstatistikleri	75
Tablo 27: Veri Seti Gruplarına Ait YSA İstatistikleri	76
Tablo 28: 1 Günlük Veriye Uygulanan Yapay Zekâ Yöntemlerine Ait Anlık Tahminlerin ve Gerçek Fiyat Verilerinin Karşılaştırılması	78
Tablo 29: 1 Aylık Verilere Uygulanan Yapay Zekâ Yöntemlerine Ait Anlık Tahminlerin ve Gerçek Fiyat Verilerinin Karşılaştırılması	79
Tablo 30: Web Ortamında Çalıştırılan BTC/TL Fiyat Tahmini Modelinin Sonuçlarının Karşılaştırılması	80
Tablo 31: Anlık Olarak Web Sitesi Üzerindeki Tahmin Aralıkları	80

ŞEKİLLER DİZİNİ

Şekil 1: Omuz Baş Omuz Formasyonu.....	1
Şekil 2: Ters Omuz Baş Omuz Formasyonu.....	2
Şekil 3: BTC'nin USD Cinsinden Fiyatının 10 Yıl İçerisindeki Değişimi	5
Şekil 4: İşlemin (Mesaj) Dijital İmza ile Şifrelenmesi ve Şifrenin Alıcı Tarafından Çözülmesi.....	7
Şekil 5: SHA-256 Algoritması ile Şifrelenen Mesaj.....	9
Şekil 6: Kullanıcı Katılımlı Ağ Çeşitleri.....	10
Şekil 7: Blok Zincirinde İşlemlerin Ardışık ve Bağlantılı Bir Şekilde Bloklara Yazılması	11
Şekil 8: BTC Üretme İşleminin Zorluk Derecesinin Belirlenmesi	13
Şekil 9: K-Kat Çapraz Doğrulama Modeli	20
Şekil 10: Yapay Sinir Ağı Mimarisi.....	27
Şekil 11: 3 Katmandan Oluşan İleri Beslemeli YSA	28
Şekil 12: Bir Giriş Ve Bir Çıkıştan Oluşan RNN Yapısı	29
Şekil 13: RNN'ye Ait Çoklu Döngüler Diyagramı.....	30
Şekil 14: RNN İşlem Döngüsü.....	30
Şekil 15: RNN'de Tek Katmanlı Tekrarlayan Ağ Yapısı	31
Şekil 16: Tek Modül İçerisinde 4 Katmana Sahip Bir LSTM Nöral Ağı	32
Şekil 17: LSTM'de Yer Alan Sigmoid Katmanı	33
Şekil 18: CNN' de Maksimum Havuzlama İşlemi	35
Şekil 19: 5x5 Boyutunda Bir Görüntüyü Temsil Eden Matris.....	36
Şekil 20: 3x3 Boyutunda Üretilen Örnek Filtre	36
Şekil 21: Görüntü ve Filtre Matrislerinin Çarpım Sonucu Olan Çıktı Matrisi	37
Şekil 22: Padding İşlemi Sayesinde Görüntü Matrisinin Korunması	38

Şekil 23: ReLu Fonksiyonunun Orijinal Görüntü Üzerine Uygulanması.....	39
Şekil 24: CNN’de Maksimum Havuzlama İşlemi	39
Şekil 25: Düzleştirme Katmanında Yapılan Dönüştürme İşlemi.....	40
Şekil 26: 2 Girdi, 1 Çıktı ve 4 Kural Oluşturularak Modellenmiş Bir ANFIS Yapısı....	41
Şekil 27: Gauss Eğrisi Üyelik Fonksiyonu	42
Şekil 28: 2 Saatlik Aralıklarla 1 Haftalık (2_2) Veri Setine Ait Sonuçlar.....	61
Şekil 29: 6 Saatlik Aralıklarla 1 Aylık (4_3) Veri Setine Ait Sonuçlar.....	61
Şekil 30: 3 Günlük 1 Yıllık (5_3) Veri Setine Ait Sonuçlar.....	62
Şekil 31: 2 Günlük Aralıklarla 2 Yıllık (6_4) Veri Setine Ait Sonuçlar.....	62
Şekil 32: 1_2 Veri Setine Uygulanan Yapay Zekâ Yöntemlerine Ait Tahminlerin Bir Arada Gösterilmesi.....	69
Test verilerine ait şekillerde yatay eksenle gösterilen “zaman bilgisi” ile zaman aralıklarının çarpımı sonucu veri seti grubuna ait tarih özniteliği ortaya çıkmaktadır. Örneğin, yarım saatlik 1 günlük verileri içeren 1_2 veri seti grubuna ait zaman bilgisi 4 saat, zaman aralıkları toplamda 6’dır. Yani, $4 \times 6 = 24$ saat ve 1 günlük verilere karşılık gelmektedir. Aynı hesaplama diğer veri seti grupları için de geçerlidir.....	69
Şekil 33: Zaman Serisi Kullanılarak Yapay Zekâ Yöntemlerinin Veri Seti 1_2’ye Uygulanması, (a) ANFIS, (b) Eğri uydurma, (c) LSTM, (d) YSA.....	70
Şekil 34: 2_2 Veri Setine Uygulanan Yapay Zekâ Yöntemlerine Ait Tahminlerin Bir Arada Gösterilmesi.....	70
Şekil 35: Zaman Serisi Kullanılarak Yapay Zekâ Yöntemlerinin Veri Seti 2_2’ye Uygulanması, (a) ANFIS, (b) Eğri Uydurma, (c) LSTM, (d) YSA.....	71
Şekil 36: 4_3 Veri Setine Uygulanan Yapay Zekâ Yöntemlerine Ait Tahminlerin Bir Arada Gösterilmesi.....	71
Şekil 37: Zaman Serisi Kullanılarak Yapay Zekâ Yöntemlerinin Veri Seti 4_3’e Uygulanması, (a) ANFIS, (b) Eğri Uydurma, (c) LSTM, (d) YSA.....	72
Şekil 38: 5_3 Veri Setine Uygulanan Yapay Zekâ Yöntemlerine Ait Tahminlerin Bir Arada Gösterilmesi.....	72

Şekil 39: Zaman Serisi Kullanılarak Yapay Zekâ Yöntemlerinin Veri Seti 5_3'e Uygulanması, (a) ANFIS, (b) Eğri Uydurma, (c) LSTM, (d) YSA	73
Şekil 40: 6_4 Veri Setine Uygulanan Yapay Zekâ Yöntemlerine Ait Tahminlerin Bir Arada Gösterilmesi.....	73
Şekil 41: Zaman Serisi Kullanılarak Yapay Zekâ Yöntemlerinin Veri Seti 6_4'e Uygulanması, (a) ANFIS, (b) Eğri Uydurma, (c) LSTM, (d) YSA	74
Şekil 42: 1 Günlük Verilere Uygulanan Yapay Zekâ Yöntemlerine Ait Anlık Tahminlerin Bir Arada Gösterilmesi, (a) BTC, (b) ETH, (c) XRP	77
Şekil 43: 1 Haftalık Verilere Uygulanan Yapay Zekâ Yöntemlerine Ait Anlık Tahminlerin Bir Arada Gösterilmesi, (a) BTC, (b) ETH, (c) XRP	78
Şekil 44: Web Ortamında Çalıştırılan BTC/TL Fiyat Tahmini Model Sonuçları.....	79

GİRİŞ

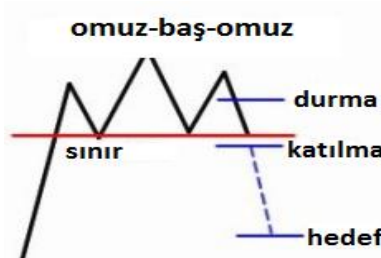
Bitcoin (BTC), 2008 finansal krizinden sonra Satoshi Nakamoto olarak bilinen kişi veya kişiler tarafından “Bitcoin: Eşten Eşe Nakit Ödeme Sistemi” isimli bir makale ile dünyaya tanıtılan kripto para birimidir (Nakamoto,2008). BTC, her ne kadar 2008 yılında bir fikir olarak ortaya atıldıysa da, BTC ile ilgili ilk transfer işlemi 2009 yılında, BTC ile ilgili ilk ticari işlem de 2010 yılında yapılmıştır, yani BTC 2010 yılında reel ekonomi ile tanışmıştır (Koçoğlu vd., 2016).

Geçmişe dönük olarak son 11 yıl incelendiğinde, 2008’den bu yana devam eden süreç “Finansal Teknolojiler 3.0” olarak isimlendirilmekte, bu süreç içerisinde BTC’yi de barındırmaktadır (Sakız ve Kutlugün, 2018; Zimmerman, 2016; <https://www.nytimes.com>: 02.05.2019).

2019 yılı itibari ile piyasadaki toplam 2169 kripto para arasında en büyük hacim, bilinirlik ve işleme sahip olan BTC (<https://www.coinmarketcap.com>: 02.05.2019), finans sistemlerine ciddi bir alternatif olacak şekilde oluşturulmuştur. Şu an piyasalarda yaklaşık 216.9 milyar dolarlık kripto para piyasa hacminin yaklaşık 128.2 milyar dolarını, yani %59’unu BTC oluşturmaktadır (<https://www.coinmarketcap.com>: 02.05.2019). BTC, bağlı olduğu herhangi bir merkez olmamasından dolayı çok ciddi fiyat değişimlerine maruz kalan bir kripto paradır.

Ekonomi alanında sıklıkla teknik analizlerde kullanılan ve geleceğe yönelik tahminler için yararlanılan formasyon grafikleri bulunmaktadır. Bu grafiklere örnek olarak; trend dönüş formasyonlarından “omuz baş omuz” ve “ters omuz baş omuz”, “fincan ve kulp”, dikdörtgen, takoz, bayrak, üçgen ve V formasyonları olarak adlandırılan “ikili zirve” ve “ikili dip” formasyonları bulunmaktadır (<https://www.investaz.com.tr/> : 13.05.2019). Şekil 1’de omuz baş omuz grafiği gösterilmektedir.

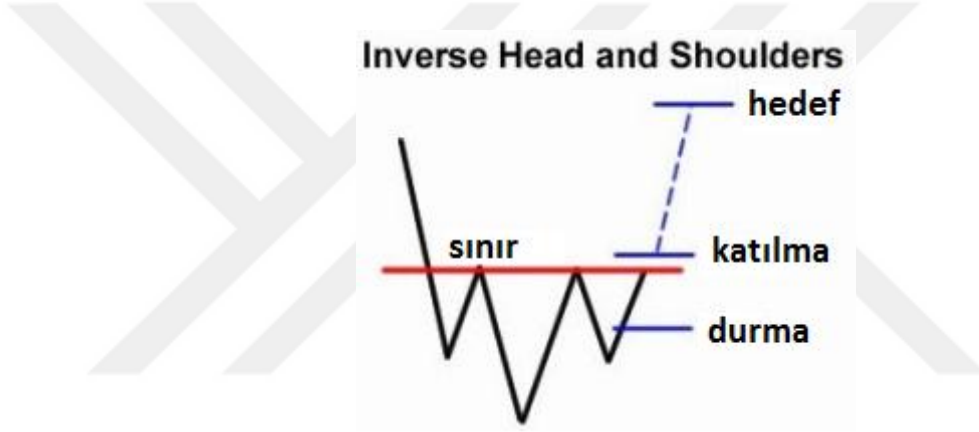
Şekil 1: Omuz Baş Omuz Formasyonu



Omuz baş omuz grafiği simetrik bir şekilde ise “üçlü tepe”, değilse omuz baş omuz formasyonu olarak adlandırılmaktadır. Şekil 1’de gösterilen grafikte geçilemeyen fiyatı, yani direnci üç defa test etmesine rağmen kıramayan (geçemeyen) hissenin, direnç ile destek arasındaki fiyat kadar fiyatın aşağı yönde bir düşüş göstermesi beklenir (<https://hardforkcafe.net>, 02.05.2019).

Omuz baş omuz formasyonu genellikle trendin sonunda oluşur ve trend sonu formasyonu olarak görülür. Omuz baş omuz grafiğinin tersi olan ve yükseliş trendi gösteren formasyon ters omuz baş omuz formasyonu olarak adlandırılmaktadır. Şekil 2’de ters baş omuz grafiği gösterilmektedir (<https://hardforkcafe.net>, 02.05.2019).

Şekil 2: Ters Omuz Baş Omuz Formasyonu



Ters omuz baş omuz formasyonunda eğer şekiller simetrik ise “üçlü dip”, değilse ters omuz baş omuz formasyonu olarak adlandırılmaktadır. Şekil 2’de gösterilen grafikte desteği üç defa test etmesine rağmen kıramayan (geçemeyen) hissenin direnç ile destek arasındaki fiyat kadar fiyatın artış göstermesi beklenir. Trendin başından oluşur ve kuvvetli bir yükseliş mesajı verir (<https://hardforkcafe.net>, 02.05.2019).

Bu grafikler matematiksel olarak çeşitli simetrik ve açı özelliklerine sahip olup çeşitli mertebeden eğrilere benzemektedir. Bu çalışmada hem bu eğrileri benzetebilecek hem de gizli kalmış özniteliklerden sonuçlar çıkarabilecek çeşitli yapay zekâ tekniklerinden faydalanılmıştır.

Bu tez çalışmasında, popülerliği her geçen gün artan ve kendi borsalarının kurulmasını sağlayan BTC’nin Dolar/Türk Lirası kuru da dikkate alınarak Türk Lirası cinsinden fiyat tahmini yapılmıştır.

BTC fiyat veya fiyat yönü tahmini ile ilgili literatür incelendiğinde özellikle Türkiye’de yapılan çalışmaların son yapay zekâ yöntemlerini kullanmada yaşadığı eksiklikler görülmüştür. Bu çalışmanın yapılmasının amacı, fiyat değişimi spekülasyonlara bağlı olan, deyim yerindeyse bir balon olan BTC’nin (Zeren ve Esen, 2018), BTC/TL cinsinden fiyat tahmininin Dolar/TL kuru ile ilişkisi üzerinden yapılmasıdır.

Çalışmada yapay zekâ yöntemleri olarak; ANFIS, YSA, LSTM ve polinomsal eğri uyurma (polynomial curve fitting) kullanılmış olup bu yöntemler birbirleri ile de karşılaştırılmıştır. Çalışmanın literatür özeti Bölüm 2’de verilirken, yapay zekâ yöntemleri 3. Bölüm olan Materyal ve Metot bölümünde açıklanmıştır. Çalışmanın 4. Bölümü olan Deneysel Çalışmalarda bu yöntemlerin hangi parametreler ile çalıştırıldığı ve elde edilen sonuçlara detaylıca yer verilmiştir. Son bölüm olan Sonuçlar bölümünde ise, elde edilen sonuçlar açıkça belirtilip, yorumlanmış ve çalışmanın daha iyi bir şekilde geliştirilip, daha doğru sonuçlar alınabilmesi için yapılabilecek olan adımlar paylaşılmıştır.

BİRİNCİ BÖLÜM

BITCOIN VE BLOK ZİNCİRİ

1.1. Bitcoin ve Blok Zinciri ile İlgili Kavramlar

Çalışmanın içeriğinin tam olarak anlaşılabilmesi için BTC ve BTC'ye dair kavramların açıklanmasında fayda görülmektedir.

1.1.1. Kripto Para

Kripto para, kriptoloji tekniği sayesinde şifrelenmiş olarak işlem yapma olanağı sağlayan, dijital değerlere sahip sanal varlıklardır (Çarkacıoğlu, 2016; Graydon, 2014).

Her ne kadar kripto paralar, geleneksel para ile yapılan tüm işlemleri karşılıyor olsa da bazı yönleri ile geleneksel paradan tamamen ayrılmaktadır. Bu ayrımların en önemlilerinden biri de kripto paraların doğası gereği merkezi olmayan bir yapıda oluşudur.

Bir kripto para, halka açık bir şekilde ve herkesin bildiği yöntemlerle sistem henüz kuruluş aşamasında iken kararlaştırılan oranlarda üretilmeye başlanır. Kripto paralar, bir hükümet veya şirket tarafından üretilmediği ve onu denetleyen bir merkez bankası olmadığı için piyasaya arz şekli ve miktarı ile tüm bunların ne zaman olacağı sistem kurulmadan önce planlanır (Çarkacıoğlu, 2016).

Kripto paranın kullanıcılara ve yatırımcılarına sunduğu hizmetlerden biri de gizlilik ve güvenlidir. Kullanıcılar, takma ad ve onlara özgü bir kullanıcı numarası ile istedikleri işlemi gerçekleştirirler. Kripto para sisteminde aracı bir üçüncü taraf ve kuruluş olmadığı için işlem ücretleri yok denecek kadar azdır.

Kripto paraların denetleyicisi olmadığı için ve kullanıcıların gerçek kimlikleri bilinmediği için yasa dışı işlem yapma fırsatı yaratır ve bu yönüyle uzmanlar tarafından eleştirilmektedir. Ayrıca yukarıda sayılan sebeplerden ötürü vergi kaçırma suçuna da ortam hazırlayabilmektedir (Marian, 2016).

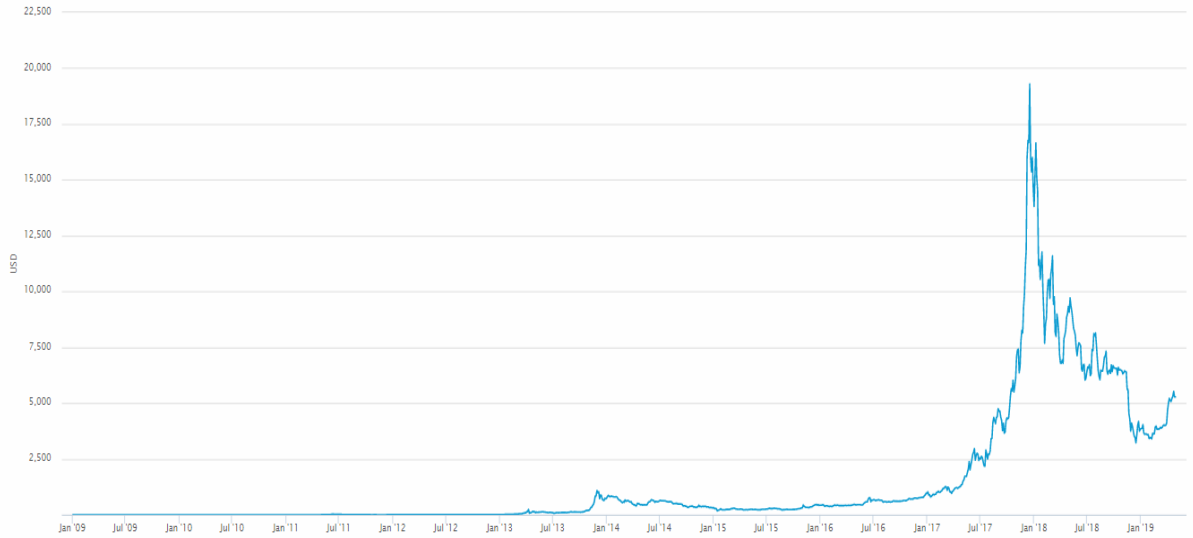
2019 yılı itibarıyla 2141 adet kripto para birimi olup, bu kripto paraların toplam hacmi 174.8 milyar dolardır (<https://coinmarketcap.com/>: 13.05.2019).

1.1.2. Bitcoin

BTC, herhangi bir otoriteye bağılı olmayan, Satoshi Nakamoto isimli kiři tarafından fikir olarak ortaya atılan eřler arası (peer-to-peer) bağılantı esasına dayanan (Nakamoto, 2008) ve kripto para piyasasında en yüksek hacme sahip kripto para birimidir (<https://coinmarketcap.com>, 02.05.2019).

BTC'nin arzı, sistem inşa edilirken belirlenmiştir. Üretilcek BTC adedi 21 milyon olarak sınırlı bırakılmıştır. Buna göre yaklaşık 2140 yılında bu arz bitecektir. 2019 yılı Mayıs ayı itibari ile toplam BTC sayısı 17.7 Milyon BTC, Pazar büyüklüğü \$95.1 Milyar Amerikan Doları (USD), günlük hacmi ise \$13.7 Milyar USD olarak görülmüştür. Şekil 3'te BTC'nin tüm zamanları içeren fiyat grafiğı USD cinsinden verilmiştir.

Şekil 3: BTC'nin USD Cinsinden Fiyatının 10 Yıl İçerisindeki Değıřimi



(<https://www.blockchain.com/tr>: 02.05.2019)

1.1.2.1. Bitcoin'in Tarihçesi

İnternet'in gündelik yařantıda aktif olarak yer alması ve bunun etkisinin her geçen gün artmasıyla beraber sanal dünyada bilinen ve bilinmeyen bazı topluluk ve gruplar ortaya çıkmaya başlamıştır (Koçoğılu vd., 2016).

BTC, ilk defa 1998'de geleneksel para sistemlerinin aksine "kriptoloji kullanılarak yaratılan ve iřletilen para" olarak Wei Dai tarafından paylaşılmıştır (BitcoinWiki, 2015).

Mevcut BTC sisteminin kurucusu ise “Peer to Peer Electronic Cash System” isimli makalenin yazarı olan Satoshi Nakamoto isimli kişidir. Bu kişinin gerçek kimliği bilinmemesiyle beraber, bunun pek de bir önemi yoktur. Çünkü Nakamoto 2010 yılında kendi isteği ile geriye çekilmiş ve sonrasında geliştiriciler tarafından BTC sisteminde bazı değişimler ve geliştirmeler yapılmıştır. Zaten BTC sisteminin önemli bileşenlerinden biri de kullanıcıların kendi arasındaki eşgüdumdür.

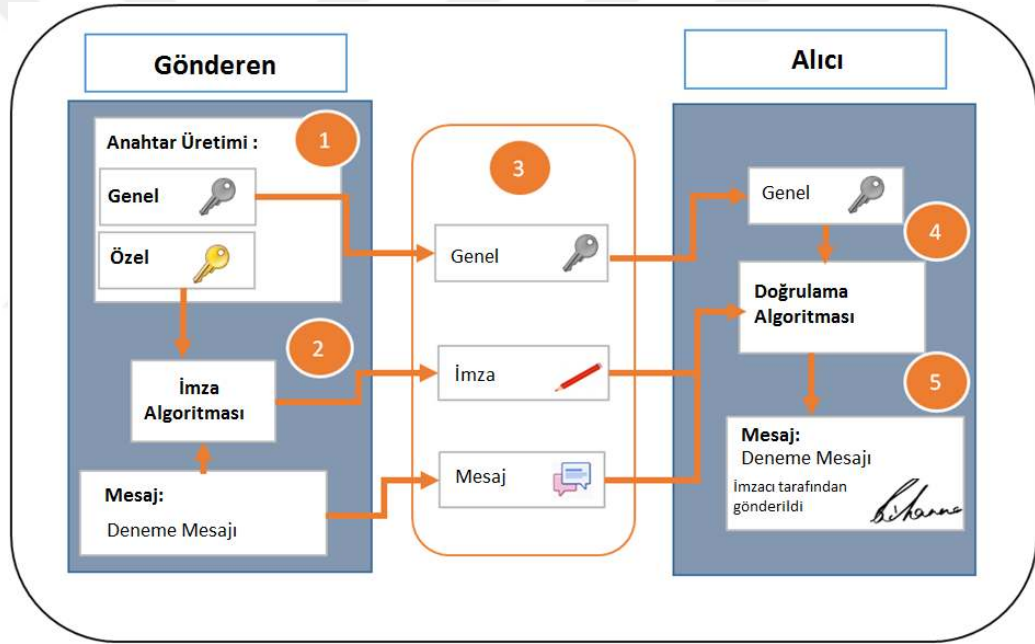
1.1.2.2. Bitcoin Sistem İşleyişi

BTC, kriptografik bir sistem üzerine inşa edilmiş bir kripto paradır. BTC kullanıcılarının mevcut düzende banka hesabına karşılık gelebilecek cüzdanları (wallet) bulunmaktadır. Kullanıcıların “cüzdan adresleri” rakam ve harflerden oluşan, kullanıcılarla ilgili gerçek bilgi içermeyen karmaşık bir dizidir. Bu sistemde bir kullanıcı başka bir kullanıcının “cüzdan adresini” biliyorsa, o kullanıcının önceden yapmış olduğu işlemleri görüntüleyebilir. Bu açıdan bakıldığında BTC şeffaf bir sistemdir (Antonopoulos, 2014; Brand, 2016; Çarkacıoğlu, 2016).

Bir kullanıcı diğer bir kullanıcıya BTC göndermek istediğinde öncelikle bu işlemi bir mesaj olarak tüm sisteme ve kullanıcılara yayar. Mesajın ulaştığı kullanıcılar bu mesajı alıp kimden geldiğine bakar, bunu onaylar ve diğer kullanıcılara da ulaşmasını sağlar. BTC açık kaynak yazılımında ve BTC sisteminde kullanıcı elindeki parasını harcamak için bir şifreye sahiptir. Bu şifre “Dijital İmza” olarak adlandırılmaktadır. Dijital imza, gerçek hayattaki imza olarak düşünülebilir. BTC bir algoritma kullanarak dijital imzanın hangi kullanıcıya ait olduğunu çözer ve bu sayede aynı işlemin birden fazla yapılmış gibi gösterilmesini, aynı para ile defalarca takas yapılmasını engeller. Dijital imza “gizli anahtar (private key)” ve “açık anahtar (public key)” olmak üzere iki adet anahtar kullanır. Gizli anahtar, kullanıcının imzasının oluşturulmasında kullanılır ve şifre gibi düşünülebilir. Açık anahtar ise BTC’nin gönderileceği kullanıcının adresidir. Aslında BTC transferinde gönderici, alıcının açık anahtarına BTC göndermektedir. Alıcı kişi, açık anahtarına gönderilen BTC’yi harcamak veya kullanmak için o adresin sahibi olduğunu ispatla yükümlüdür. Alıcı, bu işlemi gizli anahtarını ve gönderenin mesajını (sisteme ve tüm kullanıcılara gönderdiği mesaj) kullanarak attığı dijital imza ile gerçekleştirir. Ağdaki diğer kullanıcılar da bu imzayı bir fonksiyon sayesinde çözüp, alıcının açık anahtarı ile eşleştiğini doğrular.

Dijital imza kendi içerisindeki matematik sayesinde gönderenin gizli anahtara sahip olup olmadığını gizli anahtarı görmeden de anlayabilmektedir. İmza, ağı gönderilen mesaja bağlı olduğu için her işleme ait farklı bir imza üretilecektir. Bunun anlamı, aynı imza ile farklı işlemleri doğrulamak için kopyalama işleminin yani sahtekârlığın yapılamayacağıdır. İşlemin başında gönderilen mesajın bir fonksiyonunun olması, mesajın bir uçtan bir uca gönderilirken değiştirilmesini de önlemektedir. Mesajda yapılacak veya meydana gelebilecek en küçük bir değişiklik bile imzayı geçersiz kılmaktadır (<http://www.bitcoinhaber.net>: 03.05.2019; Nakamoto, 2008). Şekil 4'te BTC dijital imzasının gönderici ve alıcılarda ne gibi işlemlerden geçtiği gösterilmiştir.

Şekil 4: İşlemin (Mesaj) Dijital İmza ile Şifrelenmesi ve Şifrenin Alıcı Tarafından Çözülmesi



(https://medium.com/@craig_10243: 01.05.2019)

Gönderilmek istenen mesaj çok yer kaplamasın diye önce özet haline getirilir, daha sonra gizli anahtar ile şifrelenir ve böylelikle mesaj “imzalanmış” olur. Bu işlemlerden sonra alıcının açık adresi, gönderenin imzası ve mesajı alıcıya gönderilir, alıcı tarafında da açık anahtar eşleştikten sonra şifrelenmiş mesaj (encrypted), alıcı tarafından açılır (decrypted).

1.1.3. Blok Zinciri (Blockchain)

Günümüzde kişiler veya kurumlar arası para transferi işlemlerinde üçüncü bir taraf söz konusudur. Üçüncü taraf, gönderen ve alıcının bilgilerine vakıftır. Transfer işlemlerinde yaşanabilecek aksilikler olması durumunda gönderenin ve alıcının zararını temin etme garantisi vermekte ve bu tür hizmetler için özellikle uluslararası transferlerde ciddi komisyonlar almaktadır. Blok zincirinin herhangi bir merkezi yoktur ve sistemde güvenilir üçüncü bir taraf da yoktur. Yapılan transferlerin onayı dağıtıcıdır. Bu sistem dinamik bir yapıya sahiptir. Yani herhangi bir kullanıcı istediği zaman sistemden çıkabilir ve sisteme dâhil olabilir. Bu ağ üzerindeki tüm işlemler bir muhasebe defterinde (“open ledger”) tutulur (Kırbaş, 2018).

Blok zinciri teknolojisi, özellikle dijital paralardaki “çift harcama” sorununun önüne geçebilmek için geliştirilmiş; verileri özel kriptografik bir yöntemle şifreleyen, dağıtık bir veri tabanına sahip sistemdir (Pisa ve Juden, 2017; Yavuz, 2019).

1.1.3.1. Kriptografik Özet Fonksiyonu (SHA-256)

Özet Fonksiyonları, girilen veya gönderilmek istenen mesajın uzunluğuna bakmaksızın bu mesajları sabit bir uzunluğa hazır hâle getirirler. Bir özet fonksiyonu, hızlı olmalı, girilen mesajlardaki en ufak değişikliklerde bile önceki çıktıya benzer bir çıktı üretmemelidir. Şekil 5’teki mesajda yer alan bütün karakterler aynı olmasına rağmen sadece bir harfin büyük olması sebebiyle her iki Security Hash Algorithm-256 (SHA-256) fonksiyonunun çıktısının da farklı olduğu görülebilmektedir.

Girilen mesajın karakter sayısı ve karakter sıralaması aynıdır. Mesaj bir, “Merhaba Dünya” ve mesaj iki, “merhaba Dünya”dır. Sadece ilk karakter olan ‘m’ harfinin değişikliğinin algoritmadaki karşılığı oldukça karışık ve tespit edilemez olmuştur.

Şekil 5: SHA-256 Algoritması ile Şifrelenen Mesaj

Merhaba Dünya

SHA256 Hash of your string:
51D6575FD179FCD4894CBEF4AF6BEF4B06DE5B041390BD659CF209FF1CF3BE7D

merhaba Dünya

SHA256 Hash of your string:
0BDD492CC73EA339E601480CA8940C08239CA035633205FC92E9E418091F6434

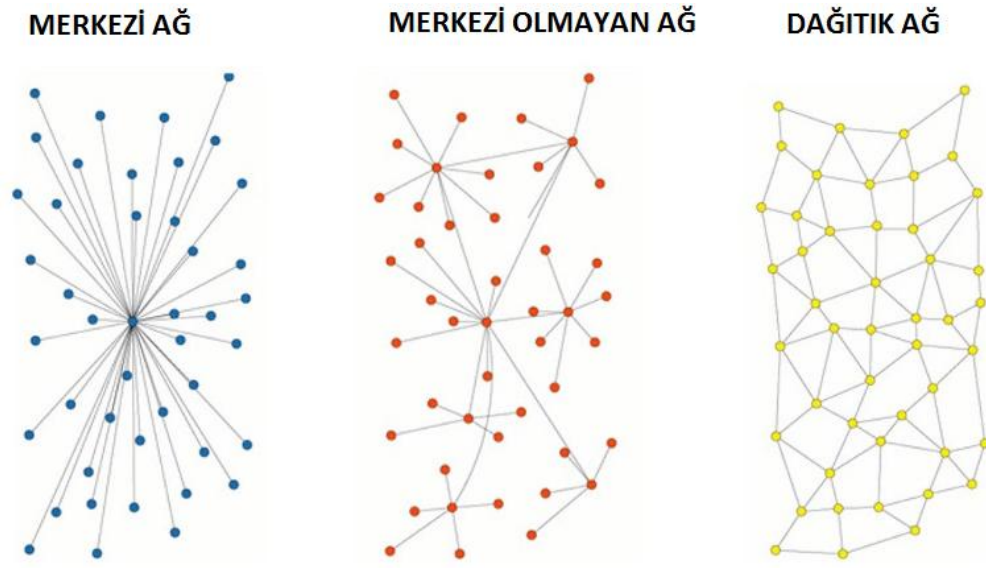
(<https://passwordsgenerator.net> : 01.05.2019)

BTC'nin üzerine inşa edildiği sistem olan blok zinciri, kriptografi olarak SHA-256 fonksiyonunu kullanır. Bu fonksiyon, girilen mesajın içeriğine ve uzunluğuna bakmaksızın 256-bit (32 byte) özet oluşturur ve en güvenilir şifreleme algoritmalarındandır. Şekil 5'te de görüldüğü üzere en ufak bir değişiklik sonucunda bile yeni üretilen özet asla tahmin edilemez. SHA-256, ardışık bir şekilde 1 ve 0'lardan meydana gelir. Okuma kolaylığı açısından, dörtlü gruplar olarak ve onaltılık sistemde şifrelenirler (Çarkacıoğlu, 2016)

1.1.3.2. Eşler Arası Ağ Bağlantısı (Peer to Peer Network Connection)

Blok zinciri sisteminde ağı dâhil olan her kullanıcının kendisine ait senkronize edilmiş bir kopyası mevcuttur. Bu sayede kullanıcılar ağdaki tüm işlemleri, tüm transferleri izleyip bunlara onay verebilir. Yani merkezi bir otoritenin, üçüncü bir tarafın olmadığı bu sistemde her kullanıcı, hem istemci hem de sunucu görevlerini yerine getirmektedir. Şekil 6'da ağ çeşitleri ve eşler arası ağı uygun dağıtık ağ modeli verilmiştir.

Şekil 6: Kullanıcı Katılımlı Ağ Çeşitleri



(<https://medium.com/@KarmaCoverage> : 01.05.2019)

Şekil 6’da görülen dağıtık ağ modelinde her bir nokta bir bilgisayarı yani bir kullanıcıyı temsil etmekte ve “düğüm (node)” olarak adlandırılmaktadır. Dağıtık ağ modelinde ağa yeni bir kullanıcının dâhil olması ya da bir kullanıcının ağdan ayrılması oldukça kolay ve hızlıdır.

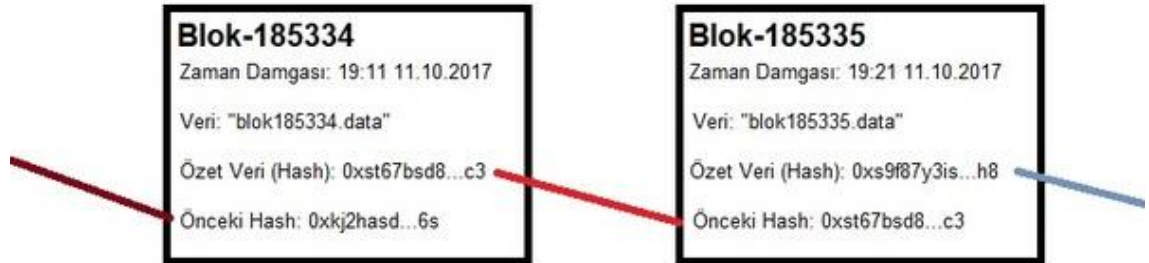
1.1.3.3. İş Kanıtı (Proof of Work)

Bilgisayarlar bazı işlemleri çok kısa sürede ve oldukça süratli bir şekilde gerçekleştirebilirler. Bir bilgisayar matematiksel hesaplamaları, bazı metin tabanlı işlemleri çok kısa sürede yapabilir. Bir web sitesine sürekli olarak istek gönderildiğini düşünün. Belli bir süre sonra bu web sitesi bilgisayardan biraz istek atmasını yani çalışmasını ve bu işlemi ispatlamasını isteyebilir. Bu sebeple biraz çalışacak ve sonra siteye istek gönderecektir ama istediği kadar çok istek gönderemeyecektir. Çünkü bunun için yeterli zamanı olmayacaktır.

Blok zinciri modelinde tüm işlemler adından da anlaşılabilceği gibi bloklara yazılır ve bu bloklar bir zincir ile birbirine bağlanır. Her blokta işlem bilgileri, o blok için özet fonksiyonu (hash) değeri ve önceki bloğun hash değeri vardır ve bunların üzerine de “zaman damgası (timestamp)” vurulup, blok diğer kullanıcılar tarafından da onaylanıp kabul edilirse zincire eklenir. Blok zincirinde herhangi bir işlem bu şekilde

gerçekleştirilir. Şekil 7’de blok zincirindeki bir işlemin bloklara nasıl kaydedildiği gösterilmiştir.

Şekil 7: Blok Zincirinde İşlemlerin Ardışık ve Bağlantılı Bir Şekilde Bloklara Yazılması



(<https://www.getrevue.co> : 01.05.2019)

Bu kısımda akla ilk gelen soru bu bloklara müdahale edilip, zincirinin kötü niyetli kişiler tarafından bozulup bozulmayacağı sorusu olabilir. Çalışmanın önceki kısımlarında blok zinciri teknolojisinin özellikle “çift harcama” sahtekârlığına karşı geliştirildiğinin hatırlanması gerekir. Blok zincirindeki bir bloğa yapılacak müdahalenin yani bir bloğun ve taşıdığı işlem bilgisinin saldırıya uğraması ve onun zorla değiştirilmesi, ağdaki kullanıcıların yüzde 50’den bir fazlasının elinde bulunan açık muhasebe defterindeki ilgili bloğun değiştirilmesi anlamına gelmektedir ki bunun için gerekli bilgisayar ve güçlü işlemci ve bu donanımlarla çözülmesi gereken SHA-256 algoritmasının özeti hesaba katıldığında böyle bir ihtimal mümkün olmayacak kadar zayıftır. Ayrıca bu saldırı işlemlerinin önüne geçebilecek ve buna izin vermeyecek olan sebeplerden biri de çalışmanın bir sonraki kısmında ele alınacaktır.

1.1.3.4. Madenciler (Miners)

Blok zinciri üzerine inşa edilen bir kripto para olan BTC’nin üretilmesi işlemi “madencilik (mining)”, bu işlemi gerçekleştiren bilgisayarlara (kullanıcılara) ise “madenci (miner)” adı verilmektedir (<https://www.bitcoinmining.com/tr>: 02.05.2019).

Madenciler BTC üretiminde aktif rol aldıkları gibi, BTC sisteminde yapılan transfer işlemlerini bloklara yazıp ve blokların zincire dâhil edilmesini onaylayarak çok küçük komisyonlar da almaktadır.

BTC sisteminde yeni bir transfer işleminin duyurusunu yapan mesaj tüm kullanıcılara iletdikten sonra madenci olarak adlandırılan bu kullanıcılar işlemi onaylayıp, işlem bloğunu blok zincirine dâhil etmek için matematiksel bir formül-problemi çözmeye başlarlar. İşlemin bu kısmında kullanıcıların sahip olduğu işlemci ve ekran kartının gücü oldukça önem arz etmektedir. Problemi çözerek işlem bloğunu zincire dâhil eden ilk madenci, yaptığı bu işin karşılığında ona verilecek olan BTC'yi elde eder ve böylece BTC üretimi sağlanmış ve madencilik işlemi gerçekleştirilmiş olur (Khalilov vd., 2017).

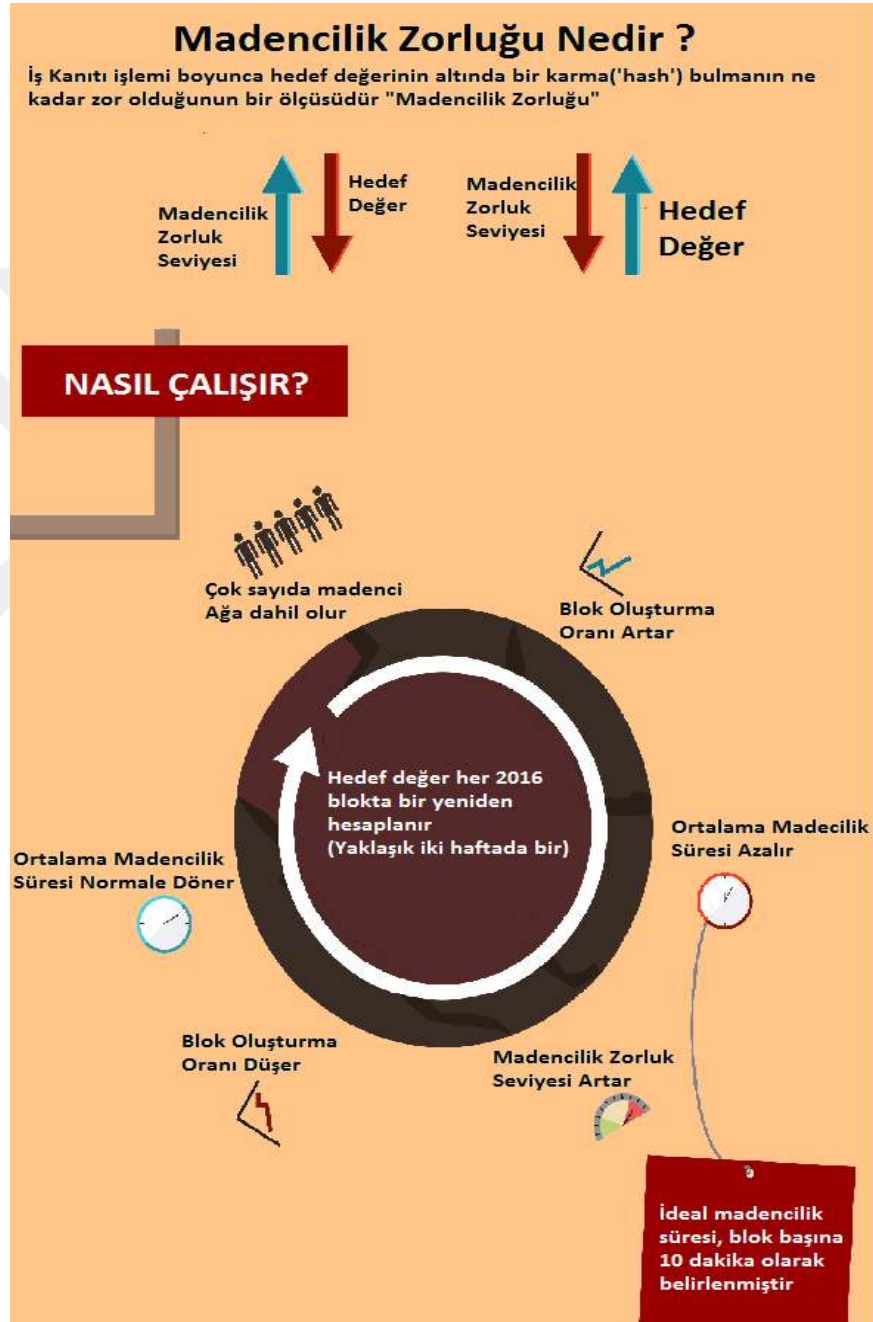
BTC üretiminde verilen ödüller her 4 yılda bir 210.000 blok tamamlandıktan sonra yarı yarıya düşmektedir. 2008 yılında ilk transfer işlemi gerçekleştiğinde, ödül 50 BTC tutarındaydı (Nakamoto, 2008). 2019 yılında ise bu ödül 12.5 BTC olmuştur (<https://coin-turk.com> : 02.05.2019).

Blok üretiminde 10 dakikada bir blok oluşturma hızı için çabalanmaktadır. 10 dakikalık zaman ortalamasını yakalamak için matematiksel problemin zorluğu kullanıcı ve yapılan işlem sayısına göre değişmektedir. Eğer ağdaki kullanıcı ve işlem sayısı fazla ise yani çok fazla kişi bir blok oluşturmaya çalışıp, ödülü almak istiyorsa matematiksel problemin zorluk derecesi artacak, tam tersi bir durum söz konusu olduğunda da problemin zorluk seviyesi daha düşük olacaktır ve böylece 10 dakikalık ortalama süresi yakalanmış olacaktır. Bahsedilen matematiksel formül-problem ise SHA-256 özet fonksiyonunun tahmin edilmesi, yani art arda 0 ile başlayacak olan 256 bitlik bir sayının hedef olarak belirlenip, madenci isimli kullanıcıların bu hedeften daha düşük bir sayı bulmaya çalışmalarıdır (Khalilov vd., 2017; <https://www.bitcoinmining.com/tr> : 02.05.2019).

Blok üretme işleminin 10 dakika olmasının sebebi bir örnekle açıklanacak olursa; eğer bir işlem çok kısa bir sürede hatta hemen o anda onaylanacak olursa sahtekârlık yapmak çok daha kolay olacaktır, kötü niyetli kişiler tarafından yapılacak bu olaylara itiraz hakkı da olmayacaktır. Çünkü işlem blok haline getirilmiş, matematiksel problem çok zor olmadığı için hemen çözülmüş ve blok çoktan zincire eklenmiş olacaktır. Ancak işlemin süresi değil de işlemin zorluğu değiştirilip, süre sabit tutulduğunda hem sahtekârlıkların önüne geçilmiş olacak, hem yarısından fazlasının kötü niyetli kullanıcıların eline geçmesi engellenecek, hem de işlem kullanıcılar

tarafından hatalı şekilde işlenmek istenmişse madenci kullanıcıları bu duruma zamanında müdahale edebilecektir. Şekil 8’de BTC üretiminin temel mantığı olan matematiksel formülün (“zorluk derecesi”) nasıl çalıştığına dair bileşenler ve bilgiler verilmiştir.

Şekil 8: BTC Üretme İşleminin Zorluk Derecesinin Belirlenmesi



(<https://www.bitcoinmining.com>: 02.05.2019)

Şekil 8’de de görüldüğü gibi hedef değer azaldıkça, zorluk seviyesi artmaktadır. BTC’nin piyasaya ilk çıkışının üzerinden 11 yıl geçmesi ve sisteme dâhil olan kullanıcı sayısının çok fazla artması sebebiyle bunun gayet normal olduğu düşünülebilir. Kullanıcı sayısı ile beraber doğal olarak işlem sayısı ve işlem gücü de artmıştır (<https://www.bitcoinhaber.net> : 02.05.2019). Böylesine bir gücün önüne geçmek için de blok oluşturma süresi sabitlenip, zorluk seviyesi arttırılmıştır. “Hedef” olarak adlandırılan çözülmesi gereken yeni *hash* değeri 2016 blokta bir tekrar hesaplanır. Her yeni blok için 10 dakika harcandığı bilindiğinde, zorluk derecesi ve hedefin 2 haftada bir yenilendiği anlaşılmaktadır.

1.1.4. Kriptoloji

Kişiler arası, kurumlar arası ve devletlerarası iletişim ve haberleşme de güvenlik her zaman önem arz etmiştir. İletişime söz konusu olan mesajın bozulmadan, çalınmadan, değiştirilmeden alıcıya ulaştırılması her zaman problem teşkil etmiştir. Mesajın bu tür tehlikelere karşı korunması ancak onu şifrelemekle ve alıcı tarafında bu şifreyi çözmesi ile mümkündür. Bu açıdan bakıldığında kriptoloji ikiye ayrılmaktadır: Mesajın şifrlenmesi (kriptografi) ve şifrelenmiş mesajın tahrif edilmeden çözülmesi (kripto analiz). Kriptoloji artık sadece devletler ve askeri bürokrasi için değil, başta finans sektörü olmak üzere tüm özel sektörün ilgilendiği ve kullandığı bir yöntem haline gelmiştir (Yerlikaya, 2006; Yılmaz, 2007).

Mesajın içeriğinin şifrlenmesi iki şekilde olabilir. Bunlardan ilki, mesajın belirli bir yöntem kullanılarak kısaltılması (stenografi); ikicisi ise, özel bir teknik kullanılarak mesajın içeriğinin anlaşılabilir bir hâle getirilmesidir (kriptografi) (Schneider ve Wiley, 1996).

Kriptolojinin temelinde güvenlik ve gizlilik olmak üzere iki prensip yatmaktadır. Güvenlik, gönderilmek istenen mesajın (bu mesaj herhangi bir şey olabileceği için veri olarak adlandırılabilir) deyim yerindeyse “sağ salim” bir şekilde güvenilir bir aracıyla (kişi ya da protokol) gönderene ulaştırılmasıdır. Gizlilik ise, mesajın içeriğinin sadece gönderen ve alıcı kişiler/kurumlar tarafında bilinmesi, mesajın iletim esnasında karşılaşılabileceği olası tüm sorunlara, engellemelere ve müdahalelere karşı içeriğinin değiştirilmemesi, bozulmamasıdır. Kriptoloji bir bilim dalı olup, kriptografi ise bu bilim dalı bünyesinde kullanılan şifreleme yöntemlerinden biridir (Yılmaz, 2007).

İKİNCİ BÖLÜM

LİTERATÜR ÇALIŞMASI

Tez çalışmasının bu bölümünde BTC fiyat tahmini ile ilgili yapılmış çalışmalara tarih sırasına göre değinilmiştir. Farklı öznitelikler, farklı yapay zekâ teknikleri kullanılarak yapılmış olan çalışmalarda yer alan modeller ve uygulanan modellerin neticeleri ele alınacaktır.

Shah ve Zhang (2014), BTC fiyat tahminini gerçekleştirmek için Bayes Regresyonunu (BR) kullandıkları çalışmada BTC yatırımcılarına bu yöntemin, yatırımlarını 60 günde iki misline çıkarabileceklerini önermektedir.

Graves ve Au (2015), Blok zincirinin sistem özelliklerinin BTC'nin fiyat tahmini üzerine yaptığı araştırmada kullanılan makine öğrenmesi optimizasyonunun verdiği sonuç itibari ile %55 civarında yukarı yönlü bir tahmin doğruluğu bulmuştur.

Madan vd. (2015), makine öğrenmesi algoritmalarını kullanarak BTC fiyat tahmini yapmaya çalışmıştır. Proje iki aşamalı bir şekilde kurulmuş olup; ilk olarak BTC fiyatını belirlediği düşünülen 25 öznitelik kullanılarak, ikinci aşamada ise sadece BTC fiyatına odaklanıp zaman değişkeni farkları ile incelemiştir. İlk aşamanın sonucunda günlük fiyat tahmini güvenilirlik oranı %98,7 olarak bulunmuştur. İkinci aşamanın sonucunda ise fiyat değişimi tahmini %50-55 civarında bulunmuştur.

Matta vd. (2015), BTC fiyat değişiminin sosyal medyada atılan tweet'ler ve Google Trends verileri ile bir ilişkisi olup olmadığını araştırmıştır. Araştırma neticesinde BTC fiyat değişimi ile Google Trends verileri arasında anlamlı bir ilişki olduğu kanıtlanmıştır.

Yang ve Kim (2015), BTC işlem akışı ile BTC fiyat değişimi arasındaki ilişkiyi karşılaştırdıkları çalışmada, BTC işlem ağı karmaşıklığı ile BTC piyasası oynaklığı arasında güçlü bir ilişki olduğu sonucunu elde etmiştir.

Indera vd. (2017), hareketli ortalama (MA) göstergeleri ve BTC geçmiş fiyat verilerine ait açılış, kapanış, en yüksek, en düşük değerlerini kullanarak çok katmanlı dışsal girişler (NARX) modelini önermektedir. Model, BTC geçmiş fiyat verilerine uygulandığında doğru bir fiyat tahmini yeteneği sunmaktadır.

Guo ve Antulov-Fantulin (2018), BTC fiyat dalgalanmasını incelemek ve tahminlemek için istatistiki ve makine öğrenmesi yaklaşımlarını kullanmışlardır. Sonuç olarak “topluluk yöntemi” Xgboosts (XGT) ve “düzenli regresyon” Elastic-NET (ENET) yaklaşımları kıyas yapılan diğer tekniklere göre daha başarılı sonuçlar vermiştir.

Alessandratti vd. (2018), kripto para borsalarındaki olağanüstü dalgalanmaların anormal ürünler üretmek için kullanılabileceği hipotezini savunmuşlar ve 2015-2018 yılları arasında 1681 kripto para ile yaptıkları bu çalışmanın sonunda makine öğrenmesine dayalı gelişmiş ve modern algoritmaların kısa vadede kripto para borsalarındaki “evrim”i tahmin edebilecek düzeyde olduğunu göstermişlerdir.

Şahin ve Özkan (2018), BTC günlük kapanış değerlerini dolar cinsinden tahmin etmeye çalıştığı projede 2015 ve 2018 yılları arasındaki fiyat verileri üzerinden çalışmışlardır. Çalışmanın neticesinde kullanılan modeller arasında en iyi sonucu veren modelin Eşik Oto regresyon Koşullu Varyans (TARCH) olduğu kanıtlanmıştır.

Ceyhan vd. (2018), BTC fiyat hareket yönünü metin madenciliği ve yapay zekâ tekniklerini birleştirerek hesaplamaya çalışmıştır. Yaptıkları çalışmada, BTC verilerinin tarih aralığı veri seti ve gündelik haber başlıkları ile birleştirilip, modern yapay zekâ algoritmaları ile çalıştırılması önerisinde bulunulmuş ve çalışmanın sonucu oldukça başarılı olmuştur.

Jang ve Lee (2018), çalışmalarında BTC fiyat tahmini yapmak için Bayes Sinir Ağı (BNN) modelini ve BTC sisteminin temel omurgası olan blok zinciri (Blockchain) ile ilgili özelliklerini kullanmışlardır. BNN'nin diğer doğrusal ve doğrusal olmayan modeller ile karşılaştırıldığı bu çalışmada BTC fiyat-zaman serisini analiz etmede ve son fiyat piyasa oynaklığının (volatilitésinin) yüksek olmasının sebebini açıklamada diğer modellerden daha başarılı bir performans gösterdiği bulunmuştur.

Akçora vd. (2018), BTC sisteminin işleyişinde önemli bir yer tutan blok zinciri üzerindeki işlem bloklarını analiz edip BTC fiyat tahmini üzerindeki etkisini Granger nedensellikleri ile açıklamaya çalışmıştır. Chainlet analizinin BTC fiyat oluşumuna dair iyi sonuçlar verdiği ve bazı tiplerinin BTC fiyat tahminine dair yüksek bir başarı oranı getirdiğini göstermişlerdir.

Guo vd. (2018), BTC alım-satım emirlerini inceleyerek bu işlemlerin BTC fiyatı üzerinde bir değişime neden olduğunu belirlemiştir. Çalışmada makine öğrenmesi

algoritmaları ve istatistik tabanlı teknikler birleştirilerek elde edilen bulgular ışığında BTC fiyat tahmini yapılmasına imkân sağlayan bir model oluşturulmuştur.

Spilak (2018), makine öğrenmesi, derin öğrenme ve LSTM algoritmalarını kullanarak kripto döviz piyasası indeksinde (CRIX) yer alan şifreleme metodolojisinin yönünü tahmin etmeye çalışmıştır. Araştırmanın sonucunda LSTM'nin yüksek bir güvenilirlik oranı ile tahmine dayalı en güçlü model olduğu ortaya konmuştur.

McNally vd. (2018), BTC'nin Amerikan Doları (USD) cinsinden fiyat tahmini yapmaya çalıştıkları araştırmada LSTM, Bütünleşik Oto Regresyon Hareketli Ortalama (ARIMA), Bayesian tabanlı RNN gibi teknikler sınanmıştır. LSTM'nin, %52 oranında sınıflandırma derecesi ile en yüksek başarıyı yakaladığı bu çalışmada ayrıca doğrusal olmayan derin öğrenme yöntemlerinin, ARIMA'dan daha başarılı tahminlerde bulunduğu görülmüştür.

Kılıç ve Çütcü (2018), BTC fiyat değişimleri ile Borsa-İstanbul (BIST) arasındaki eşbütünleşme ve nedensellik ilişkisini Engle-Granger ve Gregory-Hansen eşbütünleşme testleri ile Hacker-Hatemi-J ve Toda-Yamamoto nedensellik testleri ile açıklamaya çalıştıkları araştırmalarında Toda-Yamamoto nedensellik testine göre BIST'ten BTC fiyat değişimlerine doğru tek yönlü nedensellik ilişkisi olduğunu göstermişlerdir.

Çütcü ve Kılıç (2018), 2013-2018 yılları arasındaki BTC haftalık verileri ile yapısal kırılmalı testler kullanarak dolar kuru ile BTC fiyatları arasındaki ilişkiyi incelemiştir. Araştırma sonucunda dolar kuru ve BTC arasında uzun dönemli ilişki olduğu sonucuna ulaşılmıştır. Purbarani ve Jatmiko (2018), Büyük Veri (Big Data) analizi ile BTC fiyat tahmini yaptığı çalışmada doğrusal regresyonun (Linear Regression) makine öğrenmesi gerçekleştiren algoritmasını veri simülasyonu için kullanmıştır. Araştırma neticesinde Apache Spark küme mimarisi ve merkezi işlem birimi (CPU) arasında yapılan kıyaslamada Apache Spark simülasyonunun CPU'ya göre 2 kat daha hızlı çalıştırılabildiği sonucu elde edilmiştir.

Singh ve Agarwal (2018), BTC fiyatlarına ait son 8 yıllık veriler ile yaptıkları çalışmada yapay zekâ teknikleri arasından K-en yakın komşu regresyonunun (KNN Regression) 0.00021 ortalama karesel hata değeri ile en iyi sonucu verdiğini göstermişlerdir.

Gullapalli (2018), BTC günlük fiyat değişiminin tahmin edilmesinde günlük açılış ve kapanış değerlerinin etkili olduğunu savunmuştur. RNN, YSA ve zaman gecikmeli sinir ağlarını (TDNN) kullanarak yaptığı çalışmada BTC fiyat tahminini en iyi yapan algoritmanın TDNN olduğunu göstermiştir.

Sakız ve Kutlugün (2018), makine öğrenmesi algoritmaları ile geçmiş BTC fiyat verilerini kullanarak BTC fiyat tahmini yaptıkları çalışmada 2018 yılı şubat ayı BTC fiyat tahminini gerçeğin çok ötesinde bulmuşlardır. Çalışmada, uygun öznitelikler ile uygun yapay zekâ algoritmaları kullanılarak deneysel çalışmaların devam edebileceğini vurgulamışlardır.

Şahin (2018), ARIMA ve YSA kullanarak 2012-2018 yıllarına ait BTC fiyat verileri ile BTC fiyat tahmini yaptığı çalışmasında, çok katmanlı yapay sinir ağları (Multi Layer Perceptron) ve ARIMA(1.1.6) modeli karşılaştırılmış ve çok katmanlı yapay sinir ağlarının daha iyi bir tahmin sonucu verdiği belirtilmiştir.

Giudici ve Abu-Hashish (2019), Kripto para ve klasik varlık fiyatları arasındaki ilişkiyi vektör oto regresyon (VAR) kullanarak incelemiş ve BTC fiyat tahmini yapmaya olanak sağlayan bir model ortaya koymuştur. Çalışmada sonuç olarak BTC fiyat tahminini BTC gerçek fiyatına göre %11 hata oranı ile tahmin etmiştir.

Narayan vd. (2019), BTC fiyat dalgalanması ile Endonezya parasal büyüklükleri (enflasyon, reel döviz kuru, para hızı) arasındaki ilişkiyi 2011-2018 yılları arasındaki aylık BTC fiyatları üzerinden açıklamıştır. Çalışmada, BTC fiyat dalgalanmasının Endonezya'nın parasal büyüklüklerini etkilediği sonucuna ulaşılmış ve başka ülkelerin Merkez Bankalarına kripto para politikaları ile ilgili temel oluşturacak yararlı bilgiler sunmuştur.

Charles ve Darnê (2019), BTC geri dönüşlerini tahmin etmek için Genelleştirilmiş Oto Regresyon Koşullu Değişen Varyans (GARCH) tipi model kullanmıştır. Daha önce 2010-2016 yılları arasındaki BTC fiyat verileri kullanarak yapılmış modeli, Mart 2018 dönemine kadar uzatarak yeniden analiz etmiştir. Çalışmada GARCH tipi modellerin BTC geri dönüşlerini tahmin etmede kullanılamayacağı sonucu elde edilmiştir.

Polat ve Akbıyık (2019), BTC fiyatları ile Twitter yorumları arasındaki olumlu ya da olumsuz ilişkiyi metin madenciliği yöntemi kullanarak Granger Nedensellik

analizi ile incelemiştir. Sonuç olarak, Twitter yorumlarının BTC fiyatının Granger nedeni olmamasına karşın BTC fiyatlarının Twitter yorumlarının Granger nedeni olduğu ifade edilmiştir.

Bu tez çalışmasında, literatürden farklı olarak BTC fiyatlarının çok büyük artışlar yaşandığı dönem ile ortalama seyrine döndüğü dönemi kapsayan Aralık 2016-Aralık 2018 dönemi arasındaki veriler USD/TL döviz kuru ve zaman faktörleri ile ele alınarak yasa zekâ ve derin öğrenme yöntemleri ile çapraz doğrulama ve zaman serisi şeklinde incelenmiştir.

BTC fiyatının kripto para borsaları üzerindeki baskınlığı dikkate alınarak BTC/TL fiyat tahmini ve dolayısıyla BTC/TL fiyat yönü tahmini için farklı yapay zekâ ve farklı veri düzenleme yöntemlerinin kullanıldığı bu tez çalışmasında aynı zamanda diğer kripto para birimlerinin de anlık veriler kullanılarak farklı zaman aralıklarına ait fiyat tahminleri yapılabilmesi için web sitesi kurulmuştur.

BTC/TL fiyat tahmini için oluşturulan modelin, Ethereum (ETH) ve Ripple (XRP) kripto para birimleri için de kullanılabileceği sonucuna ulaşılmış olup gerçek zamanlı sisteme dahil edilmiştir.

Ayrıca birden fazla yapay zekâ yönteminin kullanılmasından dolayı yapay zekâ yöntemleri arasında finansal tahminlerde kullanılabilecek ve gerçek fiyat değerine ilişkin en yakın ve doğru tahmini verebilecek yöntem veya yöntemlerin elde edilmesi için kullanılan farklı yapay zekâ yöntemleri arasında karşılaştırma yapılması da tez çalışmasının literatüre katkısı açısından oldukça önemli görülmektedir.

ÜÇÜNCÜ BÖLÜM

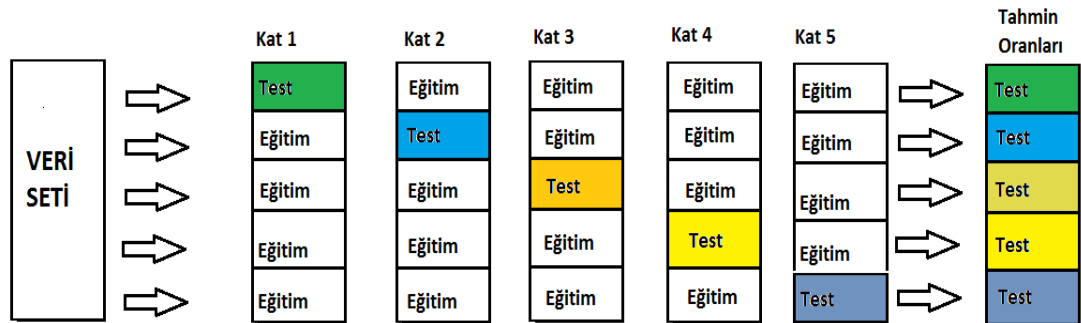
MATERYAL VE METHOD

Bu çalışmada, kripto para birimleri arasında en büyük hacme ve popülariteye sahip olan BTC'nin (<http://www.blockchain.com>: 01.05.2019) TL cinsinden fiyat tahmini yapılmıştır. Çalışmada “öznitelik (*feature*)” olarak tarih ve Dolar/TL kuru kullanılmıştır. Kullanılan öznitelikler toplam 6 veri setinde bir araya getirilmiş ve işlenmiştir. Çalışmanın amacı iki öznitelik bilgisi ile BTC fiyat tahmininin yapılmasıdır. BTC fiyat tahmininin TL cinsinden yapıldığı bu çalışmada veriler MATLAB R2018b ve JetBrains PyCharm Community Edition 2019.1 programları kullanılarak yapay zekâ yöntemlerine ait algoritmalar tarafından işlenmiştir.

Veriler çapraz geçerlilik (cross-validation) ve zaman serisi (time series) şeklinde olmak üzere iki farklı kategori altında yapay zekâ yöntemleri ile sınanmıştır. Çapraz doğrulama genellikle K-kat (K-fold) çapraz doğrulama ismiyle bilinir. K-kat çapraz doğrulama, veriyi veya veri setini K adet eşit alt parçalara (katlara) ayırır (Kohavi, 1995). Bu ayırma işlemi sonucunda parçalardan biri ‘test’ verisi olarak, geriye kalanlar da ‘eğitim’ verisi olarak nitelendirilir. Eğitim verisi, modelin uygulanıp eğitildiği veriler; test verileri de modelin eğitilmediği veriler üzerindeki performansının görüldüğü verilerdir.

K-kat çapraz doğrulamada veri seti K adet parçaya ayrıldıktan sonra bu parçalardan biri test verisi, geri kalanları da eğitim verisi olarak belirlenir ve model eğitilir. Geriye kalan her bir parça adım adım test verisi olur ve geriye kalan diğer tüm parçalar da eğitim verisi olur ve model tüm veri setini eğitir. Şekil 9’da K-kat çapraz doğrulamanın çalışma şekline örnek verilmektedir.

Şekil 9: K-Kat Çapraz Doğrulama Modeli



Şekil 9’da veri setine K-kat çapraz doğrulama 5-Kat (5-fold) olarak uygulanmıştır. Veri seti 5 eşit parçaya ayrılmış ve her bir parçada sıra ile bir parça test verisi olarak belirlenmiştir. Daha sonra test verileri bir araya getirilerek bu test verilerine ait hataların ortalaması alınmıştır. Tez çalışmasındaki verilere de 5-Kat (5-Fold) çapraz doğrulama uygulanmıştır ve performans ölçütü olarak ortalama karesel hataların karekökü (RMSE) kullanılmıştır. RMSE, ortalama karesel hatanın (MSE) kök içine alınmış halidir. Denklem (1)’de RMSE’nin nasıl elde edildiği verilmiştir.

$$RMSE = \sqrt{\frac{\sum_{j=1}^n e_j^2}{n}} \quad (1)$$

Denklem (1)’de e hedeflenen ve tahmin edilen değerler arasındaki farkı ifade etmektedir. MSE, bir regresyon eğrisinin bir noktaya ne kadar yakın olduğunu gösterir. MSE, bir modelin, tahminleme işleminin performansını ölçer. Ortalamaların kareleri alındığı için daima pozitif bir değere sahiptir. MSE, sıfıra ne kadar yakınsa, tahmin edilen oranın o kadar iyi performans gösterdiği anlaşılabılır. Bu çalışmada performans ölçütü olarak MSE’nin kök içine alınmış hali olan RMSE kullanılmıştır.

Bir veri grubunun zaman serisi olarak kabul edilebilmesi için zamana bağlı olması gerekmektedir (Seker vd, 2014; Seker, 2015). Bir örnekle açıklanacak olursa borsa verilerinin zaman serisi olduğu düşünülebilir. Çünkü her güne ait kapanış değeri bir sonraki günün açılış değerini etkilemektedir. Bu nedenle bir veri grubuna ait serinin zaman serisi olarak kabul edilmesi, mevcut veriler içerisinde en az birinin zamana bağlı olması ile mümkündür (Seker, 2015; Seker vd., 2014).

Zaman serisi, kronolojik sıra ile elde edilebilen verileri kapsamaktadır. ‘A’ örneklem büyüklüğü olmak üzere B_a , $a=1,2,3,\dots,A$ şeklinde ifade edilir. Burada B_1 ilk veriyi, B_2 ikinci veriyi ve B_a ’da son veriyi ifade etmektedir.

Zaman serisi analizi ile belirli zaman dilimlerine ait gözlemlenen ve elde edilen veriler istatistiksel olarak incelenebilmekte ve geleceğe ait tahminler yapılabilmektedir (<https://www.academia.edu/>: 11.05.2019).

Bu çalışmada veriler 5-Kat çapraz doğrulama ve zaman serisi yöntemi şeklinde çeşitli yapay zekâ yöntemleri ile sınanmıştır.

Son dönemlerde yapay zekâ yöntemlerinin kullanılmasında en çok tercih edilen Python (<https://spectrum.ieee.org/>: 07.05.2019) programlama dilinin de kullanıldığı bu çalışmada, kullanılan programlar ve işletim sistemi arasındaki eşgüdümün sağlanması için Anaconda3-2018.12 programı kullanılarak python3.6 sürümünde oluşturulan bir sanal ortam (*virtual environment*) kullanılmıştır. Sanal ortamda Python'un derin öğrenme işlemlerinde kullanılan ve Google tarafından geliştirilen açık kaynak kodlu bir kütüphane olan tensorflow (<https://www.tensorflow.org/>: 07.05.2019) kütüphanesi ve keras (<https://www.tensorflow.org/>: 07.05.2019) kütüphanesinin sunduğunu modellerden ve optimizasyon algoritmalarından faydalanılmıştır. BTC fiyat verileri ile öznitelik verilerinin nasıl ve hangi durumlara bağlı olarak elde edildiği ve BTC fiyat tahmini yapılırken kullanılan yapay zekâ yöntemleri çalışmanın ilerleyen bölümlerinde açıklanmıştır.

3.1. Kripto Para ve Öznitelik Verilerinin Elde Edilmesi

Çalışmada BTC fiyat tahmini analizinde kullanılan BTC/TL verileri zaman aralığı açısından 6 veri setine ayrılmıştır. Daha sonra yapılacak olan tahminler arasında bir kıyaslama yapılabilmesi için, yani özellikle hangi zaman aralıklarında BTC/TL tahmininin en iyi sonuçları verdiğini bulmak için oluşturulan bu 6 veri setinin her biri kendi içerisinde 4 farklı parçaya ayrılmış ve sonuç olarak toplam 24 veri seti üzerinden tahmin analizi yapılmıştır.

Öznitelik olarak kullanılan tarih ve Dolar/TL kuru verileri de BTC verileri ile aynı tarih aralığına sahip olacak şekilde oluşturulmuştur. Çalışmada kullanılan BTC/TL verileri <https://koineks.com/> adresinden, Dolar/TL kuru <https://tr.investing.com> ve <https://dolar.tlkur.com/> adreslerinden 10.12.2018 tarihinde günlük açılış değerleri baz alınıp, oluşturulmuştur. Anlık tahminlerin yapılabilmesi için de Visual Studio 2015 Community programı kullanılarak C# programlama dili ile güncel BTC/TL fiyat verileri <https://koineks.com/ticker> adresinden, Dolar/TL kuru ise Türkiye Cumhuriyet Merkez Bankası'ndan (<http://www.tcmb.gov.tr/kurlar/today.xml>) çevrimiçi (online) olarak çekilerek elde edilmektedir. Tarih özniteliği ise ilk ve son tarih aralıklarına göre oluşturulmuş olup çalışmada ardışık nümerik sayı olarak öznitelik olarak kullanılmıştır. 6 ana başlık ve zaman aralığında oluşturulan BTC/TL ve Dolar/TL fiyat verilerinin hangi tarihlere karşılık geldiği ve veri seti açıklaması Tablo 1'de verilmiştir.

Tablo 1: 6 Ana Başlık Olarak Oluşturulan Veri Setleri ve Açıklamaları

VERİ SETİ	AÇIKLAMA	BAŞLANGIÇ TARİHİ	BİTİŞ TARİHİ
1	Bir günlük veri	08.12.2018	08.12.2018
2	Bir haftalık veri	01.12.2018	08.12.2018
3	İki haftalık veri	24.11.2018	08.12.2018
4	Bir aylık veri	08.11.2018	08.12.2018
5	Bir yıllık veri	08.12.2017	08.12.2018
6	İki yıllık veri	08.12.2016	08.12.2018

6 ana başlıkta oluşturulan veri setlerinin numara ile isimlendirilmesinin sebebi, her biri kendi içerisinde 4 parçaya ayrılacak olan bu başlıkların karmaşıklığa sebep olmaması içindir.

Her bir veri setinin kendi içerisinde dört parçaya ayrılarak oluşturulmuş 24 farklı veri setine ait bilgiler Tablo [2-7] de verilmiştir. Tablo 2’de 1. veri setine ilişkin belirli aralıklardaki değerleri içeren dört farklı alt veri seti görülmektedir.

Tablo 2: 1. Veri Setine Ait Bilgiler ve Açıklamalar

Veri Seti=1	Açıklama	Başlangıç Tarihi	Bitiş Tarihi
1_1	1 saatlik 1 günlük veri	08.12.2018	08.12.2018
1_2	yarım saatlik 1 günlük veri	08.12.2018	08.12.2018
1_3	15 dakikalık 1 günlük veri	08.12.2018	08.12.2018
1_4	5 dakikalık 1 günlük veri	08.12.2018	08.12.2018

Tablo 2’de de görüldüğü gibi yapay zekâ algoritmasının öğrenme aşamasında daha iyi sonuçlar elde etmek amacıyla farklı sayıda ve periyottaki veriler kullanılmıştır. BTC fiyatlarının ani dalgalanmalarının oluşturulacak modeli etkilememesi için daha istikrarlı veriler içeren zaman dilimleri kullanılmıştır.

Tablo 3: 2. Veri Setine Ait Bilgiler ve Açıklamaları

Veri Seti = 2	Açıklama	Başlangıç Tarihi	Bitiş Tarihi
2_1	4 saatlik 1 haftalık veri	01.12.2018	08.12.2018
2_2	2 saatlik 1 haftalık veri	01.12.2018	08.12.2018
2_3	1 saatlik 1 haftalık veri	01.12.2018	08.12.2018
2_4	Yarım saatlik 1 haftalık veri	01.12.2018	08.12.2018

Tablo 4: 3. Veri Setine Ait Bilgiler ve Açıklamaları

Veri Seti = 3	Açıklama	Başlangıç Tarihi	Bitiş Tarihi
3_1	12 saatlik 2 haftalık veri	24.11.2018	08.12.2018
3_2	6 saatlik 2 haftalık veri	24.11.2018	08.12.2018
3_3	3 saatlik 2 haftalık veri	24.11.2018	08.12.2018
3_4	1 saatlik 2 haftalık veri	24.11.2018	08.12.2018

Tablo 5: 4. Veri Setine Ait Bilgiler ve Açıklamaları

Veri Seti = 4	Açıklama	Başlangıç Tarihi	Bitiş Tarihi
4_1	24 saatlik 1 aylık veri	08.11.2018	08.12.2018
4_2	12 saatlik 1 aylık veri	08.11.2018	08.12.2018
4_3	6 saatlik 1 aylık veri	08.11.2018	08.12.2018
4_4	2 saatlik 1 aylık veri	08.11.2018	08.12.2018

Tablo 6: 5. Veri Setine Ait Bilgiler ve Açıklamaları

Veri Seti = 5	Açıklama	Başlangıç Tarihi	Bitiş Tarihi
5_1	12 günlük 1 yıllık veri	08.12.2017	08.12.2018
5_2	1 haftalık 1 yıllık veri	08.12.2017	08.12.2018
5_3	3 günlük 1 yıllık veri	08.12.2017	08.12.2018
5_4	1 günlük 1 yıllık veri	08.12.2017	08.12.2018

Tablo 7: 6. Veri Setine Ait Bilgiler ve Açıklamaları

Veri seti = 6	Açıklama	Başlangıç Tarihi	Bitiş Tarihi
6_1	4 haftalık 2 yıllık veri	08.12.2016	08.12.2018
6_2	2 haftalık 2 yıllık veri	08.12.2016	08.12.2018
6_3	1 haftalık 2 yıllık veri	08.12.2016	08.12.2018
6_4	2 günlük 2 yıllık veri	08.12.2016	08.12.2018

Ana başlık olarak isimlendirilmiş olan her bir veri setinin içerisinde BTC/TL fiyat verileri, Dolar/TL fiyat verileri ve zaman bilgileri bulunmaktadır.

3.2. Yapay Zekâ

Yapay zekâ, insan davranışlarının makineler tarafından taklit edilmesini sağlayan sistemlerdir (Fürnkranz, 1999). Yapay zekâda karar verme veya tahmin yapma işlemleri makine öğrenmesi sayesinde gerçekleştirilmektedir (LeCun vd., 2015).

Yapay sinir ağlarında yer alan gizli katman (hidden layers) ve düğüm (node) sayıları arttırıldığı hâlde, donanımsal gelişmelerin eksikliğinden dolayı, yapay zekâ yöntemleri 2000'li yıllara gelindiğinde eskisi kadar popüler ve yaygın değildi. Fakat GPU ve diğer donanımsal gelişmelerin olumlu seyrinden dolayı çok sayıda gizli katmandan meydana gelen yapay sinir ağlarının maliyeti düştü ve yapay zekâ tekrar kullanılmaya başlandı (Şeker, 2017; Schmidhuber, 2015).

3.2.1. Sinir Ağları

3.2.1.1. Yapay Sinir Ağları (YSA)

YSA, insan beyninin öğrenme yeteneğinden esinlenerek geliştirilmiş sistemlerdir. Bir YSA, öğrenme işlemini örnekler sayesinde gerçekleştirir. YSA sistemleri birbirine bağlı işlem birimlerinden (yapay sinir hücrelerinden) meydana gelmektedir. Her bağlantı bir ağırlık değerine sahiptir ve yapay sinir ağının sahip olduğu bilgi ağırlık değerlerinde tutulup, tüm sisteme yayılmıştır (Öztemel, 2003).

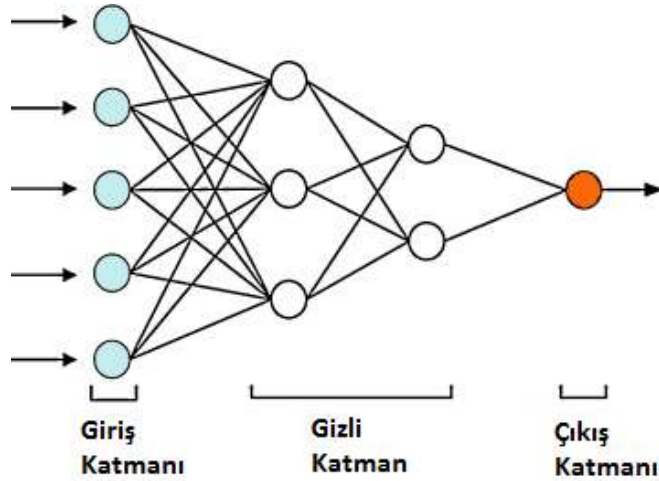
YSA, biyolojik sinir sistemleri baz alınarak oluşturulmuştur. Gerçek bir sinir sistemi hücreleri sinapsisler aracılığıyla iletişim sağlarlar. Herhangi bir sinir hücresi (nöron) sahip olduğu ve bir işleme tabi tuttuğu bilgileri aksonları kullanarak sistemdeki diğer birimlere (hücrelere) gönderir. YSA’larda da durum biyolojik sinir sisteminde yaşananlardan pek farklı değildir. Çünkü bu ağ yapısında da yapay sinir hücreleri çevreden (dışarıdan) gelen bilgileri bir birleştirme fonksiyonu kullanarak toplayıp, daha sonra aktivasyon fonksiyona tâbi tutarak, bir çıktı üretip, bu çıktıyı ağdaki bağlantıları kullanarak diğer hücrelere gönderir. YSA’nın kullandığı farklı aktivasyon ve toplama (birleştirme) fonksiyonları mevcuttur. YSA’yı birbirine bağlayan bağlantılara ait değerler “ağırlık değerleri” olarak adlandırılmaktadır. Yapay sinir hücreleri (proses elemanları) birbirlerine paralel olarak 3 katman şeklinde bir araya gelip, bir sistem meydana getirirler. Bu katmanlar; girdi katmanı, ara katman ve çıktı katmanıdır. Bilgiler YSA’ya girdi katmanı sayesinde gönderilir. Ara katmanlarda işlemlere tabi tutulup, çıktı katmanına iletirler. Ara katmanda bilgilerin işlenmesi, girdi katmanından alınan bilgilerin ağın ağırlık dereceleri ile bir çıktı haline getirilmesidir. Doğru sayılabilecek bir çıktı alınması için önce ağırlıkların doğru olarak belirlenmesi gerekmektedir. Ağın eğitilmesi, doğru ağırlık değerlerinin bulunması ile mümkündür. Ağırlık değerleri önce rastgele denebilecek bir şekilde belirlenir ama eğitim işlemi esnasında her bir örnek ağa iletildiğinde ağın öğrenme kuralına göre ağırlıklar değiştirilebilmektedir. Bu değiştirilme işlemi en doğru ağırlık değerleri bulununcaya kadar devam eder. Tüm bu süreç ağ eğitim setine ait bütün örnekler için doğru çıktı üretilinceye kadar tekrar edilir. Bu tekrar işlemi bittikten sonra, bu defa test setine ait örnekler ağa gösterilir. Bu işlem esnasında ağ, test setindeki örneklere doğru bir cevap verirse, bundan ağın eğitildiği sonucu anlaşılmaktadır. Ağ ağırlıkları belirlendiğinde, her bir ağırlığın tam olarak ne

anlama geldiği net bir şekilde bilinmemektedir. Bu yüzden YSA'ya “kara kutu (black box)” ismi verilmektedir. Ağırlıkların oluşturduğu karmaşık yapıya ve matematiksel olarak modelin anlaşılmasına rağmen, girdiler için üretilen sonuçların bu ağırlıklar sayesinde üretildiği bilindiğinden, ağı zekâsının ve kabiliyetinin bu ağırlıklara bağlı olduğu düşünülmektedir. Ağı bir olayı öğrenmesi, o olay veya veri seti için en uygun ağ modelinin seçilmesi ile mümkün olmaktadır. Bir YSA'nın modeli aşağıda verilen bilgiler sayesinde karakterize edilmektedir (Öztemel, 2003).

- Ağ topolojisi
- Toplama Fonksiyonu
- Aktivasyon Fonksiyonu
- Öğrenme Stratejisi
- Öğrenme Kuralı

3 Katmanlı bir YSA yapısı Şekil 10'da gösterilmektedir. Çıkış katmanındaki nöron sayısı sınıflandırma problemlerinde sınıf sayısına belirlenirken, eğri uydurma problemlerinde tek bir değerin hesaplanması gerektiğinden bir nöron olarak belirlenir. Bu çalışmada BTC değeri modellenip tahmin edileceği için çıkış katmanında tek nöron bulunur.

Şekil 10: Yapay Sinir Ağı Mimarisi



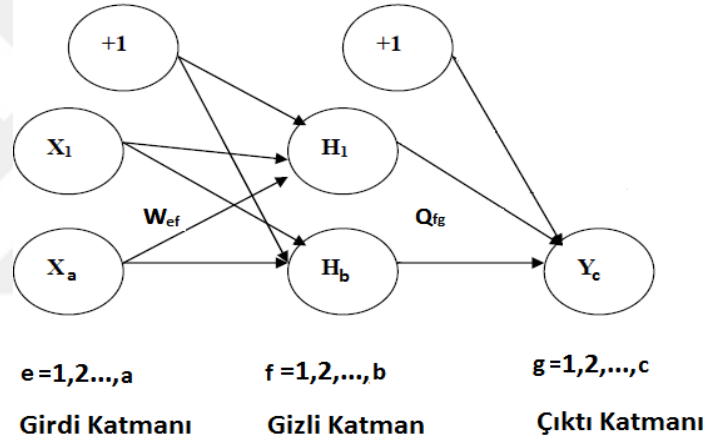
İleri Beslemeli ağlar, alınan girdiden üretilecek çıktıya doğru sadece tek yönde ilerlemeye izin veren ve giriş katmanı, ara (gizli) katman, çıktı katmanından meydana

gelen bir YSA'dır. Üç katmanın her birinde ele alınan soruna göre farklı sayıda nöronlar bulunmaktadır (Zhang vd., 1998; Kaastra ve Boyd, 1996; Hamzaçebi ve Kutay, 2004).

Şekil 11'de klasik bir ileri beslemeli YSA modeli gösterilmiş olup, girdi, gizli ve çıktı katmanlarında sırasıyla e, f, g adet nöron bulunmaktadır. Her katmandaki nöronların ağırlıklarının düzenlenmesi sayesinde nöronlar arasındaki bağlantı sağlanır ve ağırlık eğitimi tamamlanmış olur. Ağırlıkların düzenlenmesi, Denklem (2)'de gösterilmektedir.

$$E = \frac{1}{2} \sum_{q=1}^c (y_k - t_k)^2 \quad (2)$$

Şekil 11: 3 Katmandan Oluşan İleri Beslemeli YSA



Hata fonksiyonunun en aza indirgenmesi Denklem (2) ile sağlanmaktadır. Bu fonksiyonda yer alan y_k , ağda üretilen çıktıyı, t_k ise istenilen (hedeflenen) çıktıya karşılık gelmektedir. $\frac{1}{2}$ Sabit bir katsayı olup, hatanın türevini kolaylaştırmak içindir (Hamzaçebi ve Kutay, 2004).

Geri yayılım ağları, çok sık kullanılan bir ağ modelidir. Sıradan bir geri yayılım algoritması, ağdaki ağırlıkların, performans fonksiyonunun negatif yönde ilerlediği gradyan iniş algoritmasıdır (Hamzaçebi ve Kutay, 2004).

Geri yayılım algoritması, ilk olarak Werbos vd. (1974) ve Rumelhart vd. (1985) tarafından önerilmiştir. Özellikle Rumelhart vd. (1986) tarafından geliştirilip tekrar inşa edilen geri yayılım algoritması oldukça popüler hâle gelmiştir. Geri yayılım algoritması en çok kullanılan öğreticili ön öğrenme algoritmasıdır. Birçok farklı modeli olan geri

yayılım algoritması, gradyan iniş ve Newton metodu gibi standart optimizasyon tekniklerini kullanmaktadır. Bu algoritma, adını son katman olan çıktı katmanında oluşan hatayı en aza indirmek (minimize etmek) için ağırlıkları geriye dönük şekilde düzenlemesinden almaktadır (Hamzaçebi ve Kutay, 2004).

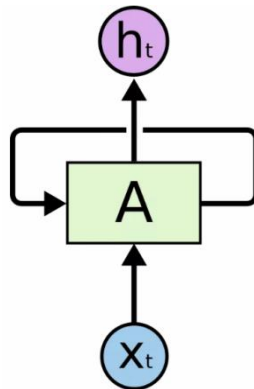
3.2.1.2. Tekrarlayan Sinir Ağları (RNN)

Basit tekrarlayan ağ (SRN), ilk olarak Elman (1990) tarafından tasarlanmıştır. Elman'ın geliştirdiği cümle simülasyonunda kelimeler isim ve fiil olarak kategorize edilmiştir (Elman, 1990).

Herhangi bir kişi bir kitap ya da yazı okurken, bir cümleyi o cümleden önceki harfler ve diğer cümleler ile birleştirip anlamlandırır. İnsan beyni öğrenmeyi bu şekilde gerçekleştirir. Önceki tüm okunan bilgiler silinip, her defasında yeniden okuma işlemine başlanmaz.

Geleneksel sinir ağları, maalesef yukarıda anlatıldığı gibi çalışmaz. İşte tam da bu noktada bu soruna bir cevap aramak için tekrarlayan sinir ağları (RNN) geliştirilmiştir. Tekrarlayan sinir ağları, bilginin devam etmesine olanak sağlayan döngülere sahip ağlardır (<http://colah.github.io> : 07.05.2019). Şekil 12'de basit bir tekrarlayan sinir ağı diyagramı gösterilmektedir.

Şekil 12: Bir Giriş Ve Bir Çıkıştan Oluşan RNN Yapısı

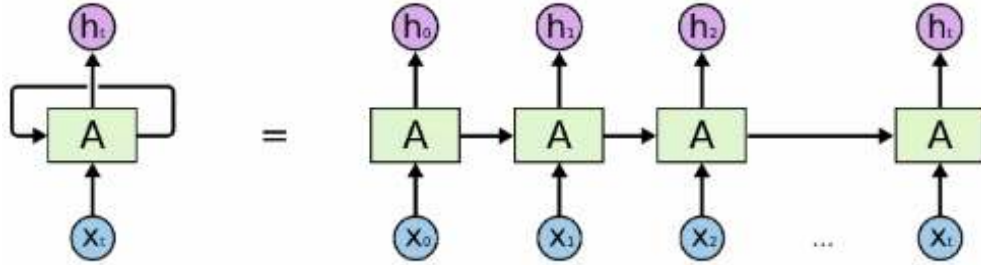


(<http://colah.github.io> : 07.05.2019)

Diyagramda görülen sinir ağının bir parçası (A), bazı giriş değerlerine (x_t) bakıp, bir çıktı (h_t) oluşturur. Bilgi, döngüler sayesinde ağın bir ucundan diğer ucuna geçirilmektedir. RNN'ler, her biri bir sonrakine bir mesaj ileten, aynı ağın çoklu

kopyaları gibi tasavvur edilebilir. Eğer Şekil 12’de gösterilen tek döngü açılacak olursa, Şekil 13’te gösterilen çoklu döngüler ortaya çıkacaktır.

Şekil 13: RNN’ye Ait Çoklu Döngüler Diyagramı

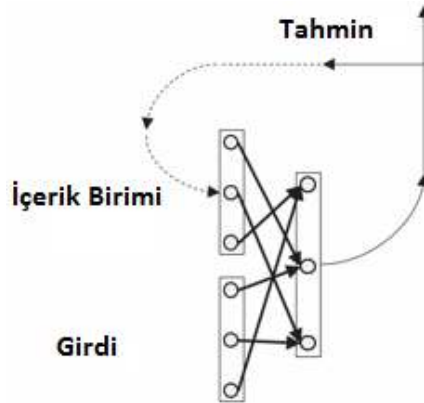


(<http://colah.github.io> : 07.05.2019)

Şekil 13’te görüldüğü gibi RNN’ler dizi ve listelerle ilgili zincirli bir yapıdadır.

RNN’ler çıktılarını bir sonraki işlemde girdi olarak kullanmaktadırlar ve aynı zamanda bir belleğe (memory) sahiptirler. Bir ağda bellek kullanılmasının sebebi, belirli bir düzende gelen girdi değerlerinin, çıktılar için bir anlamı olmasındandır. Şekil 14’te RNN’lerin işlem döngüsü diyagramı gösterilmektedir.

Şekil 14: RNN İşlem Döngüsü



(<https://medium.com/@hamzaerguder/> : 07.05.2019)

Şekil 14’te de görüldüğü üzere, girdi değeri RNN’nin gizli katmanında işlendikten sonra hem bir çıktı üretir, hem de içerik birimine (content unit) yazılır. Bu işlem, her yeni girdi değerinin etkilenmesi için kullanılır, yani tekrar edilir. Bu adımdan sonra belleğe alınan verilerde korelasyon bulunmakta ve buna da uzun vadeli (long term) bağımlılık adı verilmektedir. Bunun sebebi ise önceden gerçekleşmiş bir ya da daha fazla olayın işlenmesi olarak görülmektedir. İnsanların gündelik yaşamlarında da

bu ağda olduğu gibi daha önce bellekte tutulan bilgi, bir gizli katmanda yeni bir veri ile döngüye girip işleme tabi tutulur. Bu işlemin gerçekleşmesini sağlayan matematiksel formül, Denklem (3)'te verilmiştir (<https://medium.com/@hamzaerguder/> : 07.05.2019).

$$h_t = \phi(Wx_t + Uh_{t-1}) \quad (3)$$

Uzun vadeli bağımlılık ya da uzun vade bağımlılığı aslında uzun zaman önceki bilgiye erişim zorunluluğu demektir. Denklem (3) açıklanacak olursa;

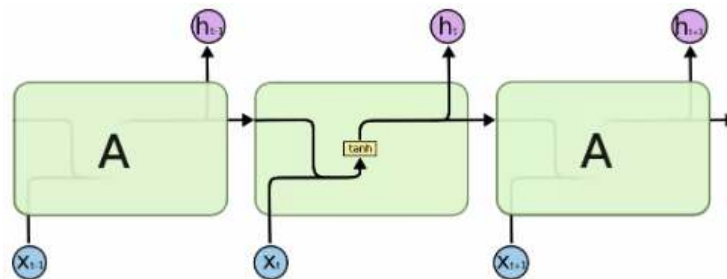
H_t , t anındaki gizli katmanın durumudur. X_t girdisi, W ağırlığı ile çarpılır, sonrasında t-1 anından içerik biriminde tutulan h_{t-1} U ağırlığı ile çarpılır ve Wx_t toplanır. Ağırlık matrisi, önceki ya da şimdiki verinin hangisinin sonuca etkisi daha çok ya da az ise ona göre değer alır. Bu hesaplamalar sonucunda ortaya çıkan hata, geri yayılım ile ağırlıkların tekrar düzenlemesi için kullanılır. Geri yayılım işlemi, hata en aza indirgenene kadar sürer (<https://medium.com/@hamzaerguder/> : 07.05.2019).

3.2.1.3. Uzun Kısa Vadeli Bellek (LSTM)

Uzun kısa vadeli bellek (LSTM) ağları uzun süreli (long term) bağımlılık sorununa karşı Hochreiter ve Schmidhuber (1997) tarafından geliştirilen, RNN'nin özel bir çeşididir.

Tüm RNN'ler, sinir ağındaki modüllerin kendilerini sürekli tekrar etmesinden meydana gelen zincirlerdir. Standart RNN'lerde, tekrar edilen modül, bir aktivasyon fonksiyonu olan tek bir “tanh” katmanına sahiptir. Şekil 15'te tekrarlayan modülün sahip olduğu tek katman gösterilmektedir.

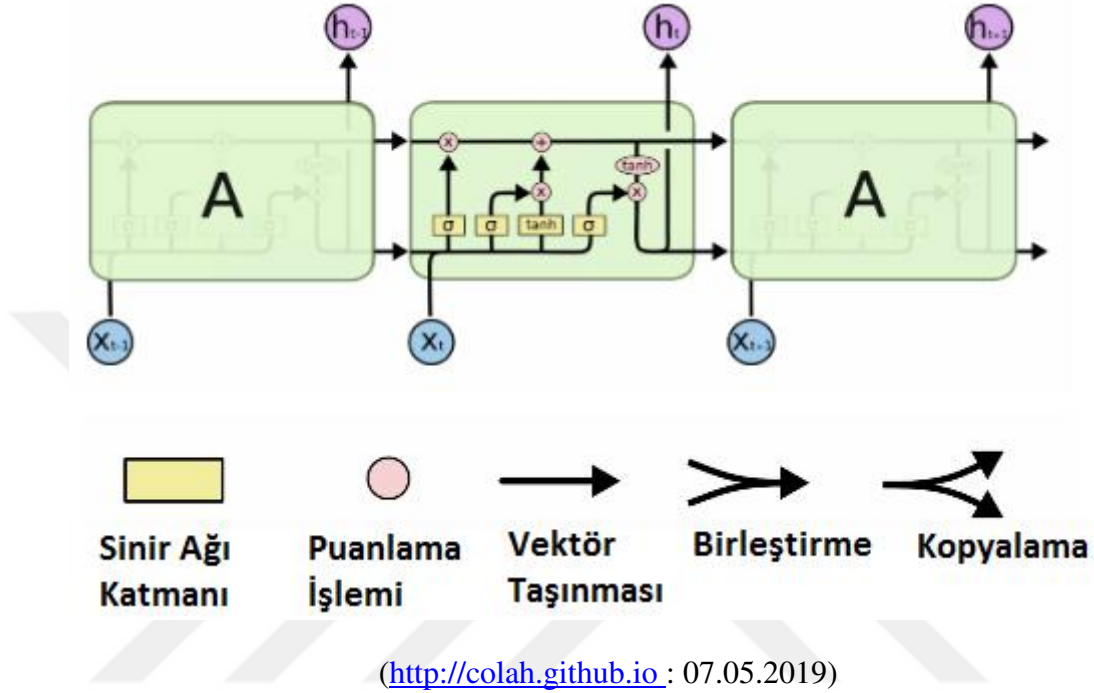
Şekil 15: RNN'de Tek Katmanlı Tekrarlayan Ağ Yapısı



(<http://colah.github.io> : 07.05.2019)

LSTM’de de bu zincirler hemen hemen aynıdır. Sadece tekrar eden modül, standart RNN’lerdeki modülün aksine bir değil dört katmana sahiptir. Şekil 16’da dört katmana sahip tekrarlayan bir modül yapısı gösterilmektedir.

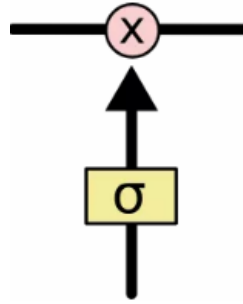
Şekil 16: Tek Modül İçerisinde 4 Katmana Sahip Bir LSTM Nöral Ağı



Şekil 16’da gösterilen ileri yönlü tek çizgiler, her katmanda üretilen çıktıları diğer bağlantıların girdilerine taşır, pembe renkli çemberler vektör eklendiği kısımlardaki hareketleri temsil etmektedir. Sarı renkli dikdörtgen şeklindeki kutular sinir ağ katmanları öğrenme işlemini gerçekleştirirken, birleştirme işlemini adından da anlaşılacağı gibi birleştirme çizgileri sağlar ve son olarak satır kopyaları her katmana kopyalama vektörü sayesinde iletilir (<http://colah.github.io> : 07.05.2019; <https://devhunteryz.wordpress.com/> : 07.05.2019).

LSTM, kapı (gate) olarak adlandırılan yapılar kullanılarak düzenlenmiş ve hücrelere bilgi ekleme, bilgi çıkarma özelliğine sahiptir. Kapılar, kullanıcının isteğine göre bilginin iletilmesi görevini üstlenmektedir. Sigmoid, nöral ağ katmanından ve puanlama işleminden (çarpma noktası) sorumludur. Şekil 16’da gösterilen sigmoid katmanı ayrıntılı olarak Şekil 17’de gösterilmektedir.

Şekil 17: LSTM’de Yer Alan Sigmoid Katmanı



(<http://colah.github.io> : 07.05.2019)

Sigmoid katmanı, bileşenlerden kaçının geçip geçmeyeceğine karar veren katmandır. Bu kararını 0 ile 1 arasında ürettiği sayılar sayesinde karar verir. Sıfır, “hiçbir şey kalmayın” anlamına gelirken, bir değeri ise “her şeyin içinden geçsin” anlamını taşımaktadır. Bir LSTM ağı, hücrelerin durumunu muhafaza ve kontrol etmek için bu kapılardan üç tanesine sahiptir (<http://colah.github.io> : 07.05.2019)

LSTM ağında yapılacak bir işlemin ilk etabı, hücre durumundan hangi bilgilerin atılacağını belirlenmesidir. Bu karar işlemi, “unutturma geçit katmanı” diye isimlendirilen bir sigmoid katman tarafından yapılmaktadır. Bu işlem, Denklem (4)’te verilen formüle göre yapılmaktadır.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (4)$$

Unutturma işlemi, h_{t-1} ’e ve x_t ’ye bakar ve C_{t-1} hücre durumundaki her bir sayı için 0 ile 1 arasında bir sayı üretir. Bir değeri, “bunu tamamen” gösterirken, sıfır değeri ise “bunu tamamen ortadan kaldırır” (<https://devhunteryz.wordpress.com> : 07.05.2019).

İkinci etap ise, kullanıcının yeni bilgilerden hangilerini saklayacağı kısımdır. Bu karar verme süreci ikiye ayrılmaktadır. Birincisi, “giriş kapısı katmanı” denilen sigmoid katman, hangi değerlerin güncellenmesi gerektiğine karar veren katmandır. İkincisi ise, hücre durumuna eklenebilecek yeni aday (C_t) değerlerin bir vektörünü oluşturan tanh katmanıdır. Bu işlemlerden sonra bu iki vektör, güncelleme oluşturulması için birleştirme vektörü ile birleştirilecektir (<https://devhunteryz.wordpress.com> : 07.05.2019; <http://colah.github.io> : 07.05.2019). Denklem (5) ve Denklem (6)’da güncelleme işlemini gerçekleştiren eşitlik verilmiştir.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (5)$$

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (6)$$

Şu aşamada hücre durumu, C_{t-1} 'i yeni hücre durumu olan C_t 'ye güncellenebilir. Önceki (eski) durumu, unutmaya karar verilen değerleri unuturak, f_t ile çarpılır. Daha sonra yeni aday değerleri eklemek için Denklem (5) ve Denklem (6)'daki sonuçlar eklenir (<https://devhunteryz.wordpress.com> : 07.05.2019; <http://colah.github.io> : 07.05.2019).

Gelenen son aşamada ise, ağı çıkı olarak ne vermesi gerektiğine karar verilmelidir. Üretilcek olan çıkı, belirlenen hücre durumlarına ve filtrele uygun olmak zorundadır. Bunun için, öncelikle hücre durumlarının hangi kısımlarının çıkartılacağına karar verecek bir sigmoid katmanı çalıştırılmalıdır. Sonrasında ise değerleri $[-1 \ 1]$ aralığında olmaya zorlamak için hücre durumu tanh aktivasyonuna eklenir, bu, sigmoid kapısının çıkılarıyla çarpılır ve bu şekilde yalnızca karar verilen parçalar çıkı olarak gösterilir (<http://colah.github.io> : 07.05.2019).

3.2.1.4. Konvolüsyonel Sinir Ağları (CNN)

Konvolüsyonel sinir ağları (CNN) ilk olarak LeCun tarafından karakter tanıma için oluşturulmuş bir derin öğrenme algoritmasıdır (LeCun vd., 1998). CNN, her ne kadar biyoloji ve bilgisayar bilimlerinin tuhaf bir sentezi gibi gözükse de, aslında resim tanımda çok sık kullanılan bir yapıdır.

İleri yönlü bir sinir ağı olan CNN, hayvanların görme merkezinden ilham alınarak geliştirilmiştir. Görme merkezindeki hücreler görselin tamamını kaplayacak bir biçimde alt kısımlara ayrılmıştır. Basit hücreler, kenar özellikleri gibi alanlar üzerinde yoğunlaşırken, karmaşık hücreler de görselin tamamına yoğunlaşmaktadır. Buradaki matematiksel işlem ise bir nöronun kendi alanındaki uyarıcılara verdiği cevap olarak düşünülebilir (Lecun vd., 1998; Fukushima,1980; Hubel ve Wiesel, 1968; Şeker, 2017).

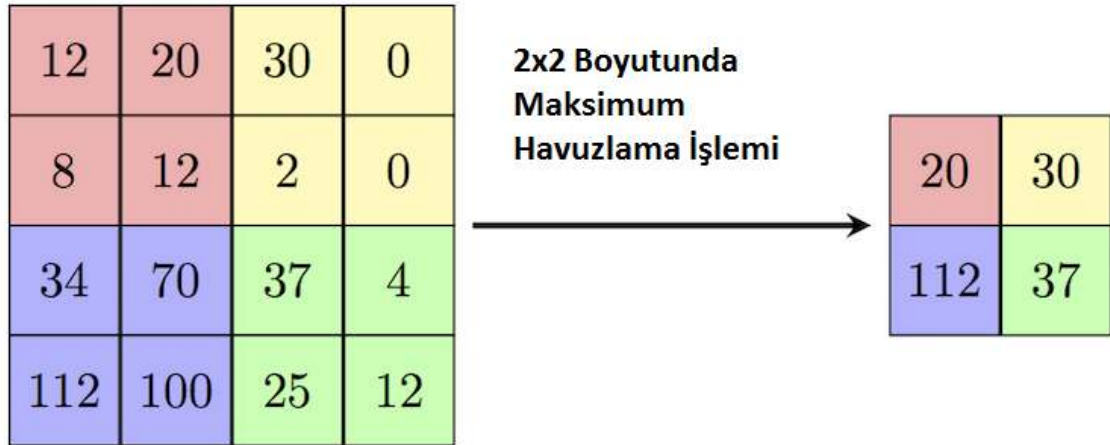
CNN, görüntüleri tanıma ve ayırt etme işlemleri için, o görüntüleri eşsiz kılan özelliklerini kullanır. Basitçe bir örnek verilecek olursa, bir buzdolabını buzdolabı yapan şeyler, dikdörtgen şekilli bir yapısı olması, genellikle beyaz renkli olması, buz ve soğutucu olmak üzere iki bölümü olup, bu iki bölümün kendi kapılarının olmasıdır. İşte CNN'lerde aynen bu şekilde çalışır. Bir görüntüyü ona ait özelliklerine göre işlerler ama

öncelikle eğriler ve kenarlar gibi alt özellikleri belirlerler ve çok daha soyut kavramlara kadar bu özellikleri oluştururlar.

CNN’i diğer standart sinir ağlarından ayıran en önemli şey konvolüsyonel ve altörnekleme (subsampling) katmanlarına sahip olmasıdır (Lecun, 2015). CNN’in diğer farkı ve faydası ise eş sayıda gizli birimle birbirine tamamen bağlı ağlara göre daha az sayıda eğitime ve parametreye sahip olması gösterilebilir (Şeker, 2017).

Kare bir görüntünün I olarak düşünüldüğü bir modelde, bir konvolüsyon katmanının girdisi $I_{a \times a \times r}$ ‘lık bir resimdir. Burada a değerleri sırası ile görüntüye ait yükseklik ve genişlik özellikleri, r ise kanal sayısıdır (bir RGB görüntüde kanal sayısı 3’tür). Konvolüsyonel katman ise $b \times b \times q$ boyutuna sahip k adet filtreden meydana gelmektedir. Filtre için b ve r değerleri ile çoğunlukla aynı seçilen q değeri seçilir. Seçilen bu değerlerden oluşturulan filtreler ile, her biri $[a-b+1 \ a-b+1]$ boyutunda k adet birbirine bağlı özellik haritaları üretilir (Ciresan vd., 2011; Şeker, 2017). Sonrasında ise Şekil 18’de gösterildiği gibi maksimum ya da ortalama havuzlama (pooling) işlemi sayesinde $p \times p$ ‘lık altörnekleme işlemi (p değeri, görüntü boyutuna göre çoğunlukla 2 ile 5 arasında) gerçekleştirilir (Şeker, 2017; Scherer vd., 2010).

Şekil 18: CNN’ de Maksimum Havuzlama İşlemi



(<https://computersciencewiki.org> : 08.05.2019)

Konvolüsyonel Sinir Ağları Katman Yapıları:

Çalışmanın bu bölümünde, yukarıda bir örnek ile anlatılan CNN yapısı, katmanları ile beraber ele alınacak ve ayrıntılı bir şekilde işlenecektir.

Konvolüsyonel Katman:

CNN'nin en önemli katmanı olarak görülen konvolüsyonel katman, verilen imajın (resmin) özelliklerini algılamak görevini yerine getirmektedir. Bu katmanda, resimdeki yüksek ve düşük seviyeli özellikleri çıkarmak için birkaç filtre uygulanmaktadır. Örneğin, uygulanacak olan bu filtre kenarlıkları algılayabilecek bir filtre olabilir. Bu filtreler ayrıca piksel değerleri de içermektedirler. 5x5x3 şeklinde gösterilen bir ifade de, 5, matris genişlik ve yüksekliği, 3 ise matris derinliğini ifade etmektedir. Şekil 19'da 5x5 filtresine uygun bir matris verilmiştir.

Şekil 19: 5x5 Boyutunda Bir Görüntüyü Temsil Eden Matris

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

(<https://medium.com/@tuncerergin> : 08.05.2019)

Şekil 19'da gösterilen 5x5'lik bir görüntü için oluşturulan 3x3'lük bir filtre Şekil 20'de gösterilmektedir.

Şekil 20: 3x3 Boyutunda Üretilen Örnek Filtre

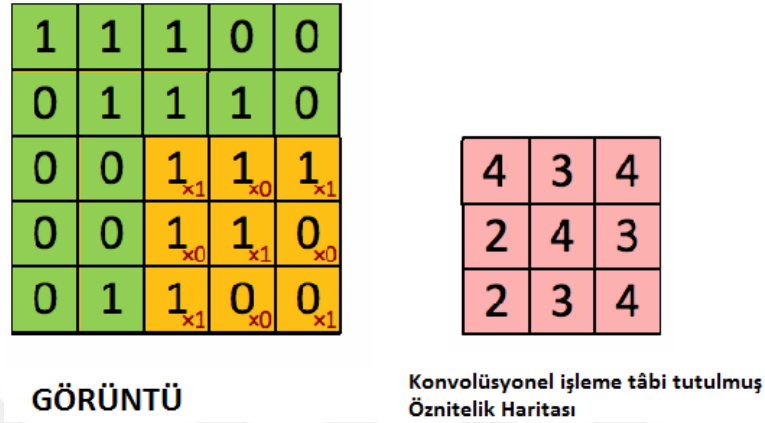
1	0	1
0	1	0
1	0	1

(<https://medium.com/@tuncerergin> : 08.05.2019)

Bu aşamada artık 3x3'lük filtre, 5x5'lik görüntüye uygulanabilecek hâldedir. Konvolüsyon işleminin adımları şu şekilde gerçekleştirilir; öncelikle filtre görüntünün sol üst köşesine yani matrisin ilk satır ve sütun hücrelerinin kesişim noktalarına (0,0) konumlandırılır. Daha sonra “görüntü” ve “filtre” matrisleri birbirleri ile çarpılır ve elde edilen sonuçlar toplanarak çıktı matrisine depolanır. Tüm bu işlemler 1 piksel (sağa doğru 1 basamak) kaydırılarak diğer satırlar için de tekrarlanır ve sonuç çıktısı olarak

3x3 boyutunda bir matris elde edilir. Şekil 21’de çıktı matrisi verilmiştir (<https://medium.com/@tuncerergin> : 08.05.2019).

Şekil 21: Görüntü ve Filtre Matrislerinin Çarpım Sonucu Olan Çıktı Matrisi



(<https://medium.com/@tuncerergin> : 08.05.2019)

Elde edilen çıktı matrisi genellikle “öznitelik haritası (feature map)” olarak adlandırılmaktadır. Filtre matrisinde temsil edilen özellikler, görüntü matrisinde ilgili kısmın bulunduğu yeri gösterir. CNN işlemlerinde çoğunlukla birden fazla özelliğin tespit edilebilmesi için birden fazla filtre kullanılmaktadır. Bu, bir CNN’de birden çok konvolüsyonel katman bulunduğu anlamına gelmektedir. Büyük adım (Stride), filtre matrisinin görüntü matrisi etrafında nasıl evrildiğini denetler ve sıklıkla “padding” kavramı ile birlikte kullanılır. Şekildeki örnekte stride büyüklüğü 1 piksel olup, istenilirse bu büyüklük değiştirilebilir. Fakat stride büyüklüğü öznitelik haritası boyutunu etkilemektedir. Padding, CNN’deki ilk filtreler uygulanırken, diğer konvolüsyonel katmanlar için bilginin korunmasını sağlamak için kullanılmaktadır. Sonuç çıktısının (feature map) boyutunun, girdi (görüntü matrisi) boyutundan daha düşük olmasından dolayı, padding, original boyutu korumak için sonuç çıktısına sıfırlar katar. Şekil 22’de padding’in matrisi sıfırlar ile doldurmasını temsil eden bir görüntü verilmiştir (<https://medium.com/@tuncerergin> : 08.05.2019).

Şekil 22: Padding İşlemi Sayesinde Görüntü Matrisinin Korunması

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	32 x 32 x 3						0	0
0	0							0	0
0	0							0	0
0	0							0	0
0	0							0	0
0	0							0	0
0	0							0	0
0	0							0	0
0	0							0	0
0	0							0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

(<https://medium.com/@tuncerergin> : 08.05.2019).

Doğrusal Olmayan Katman (Non-Linearity Layer):

CNN’de konvolüsyonel katmanlardan sonra doğrusal olmayan katmanlar gelir. CNN’nin tüm katmanları doğrusal bir fonksiyon olmasından ötürü, ağ tek bir algı (perception) olarak davranmaktadır. Yani sonuç, çıktı matrislerinin (feature map) doğrusal (linear) kombinasyonu olarak hesaplanır. Bu sebeple ağda, doğrusal olmayan katmana ihtiyaç duyulmaktadır. Bu katmanda aktivasyon fonksiyonu kullanılmasından dolayı “aktivasyon katmanı” olarak da bilinmektedir. Daha önce belirtilen diğer yapay zekâ algoritmalarında kullanılan aktivasyon fonksiyonları genellikle sigmoid ve tanh olmakta iken, CNN’ de ise aktivasyon fonksiyonu olarak eğitim hızı oldukça yüksek olan ve diğer aktivasyon fonksiyonları gibi doğrusal olmayan doğrultucu (Rectifier, diğer adıyla “ReLU”) fonksiyonu kullanılmaktadır. ReLu fonksiyonu Denklem (7)’de verilmiştir.

$$f(x) = \max(0, x) \quad (7)$$

Denklem (7)’deki formül örnek bir resim üzerine uygulandığında elde edilen sonuç Şekil 23’te görülmektedir.

Şekil 23: ReLu Fonksiyonunun Orijinal Görüntü Üzerine Uygulanması



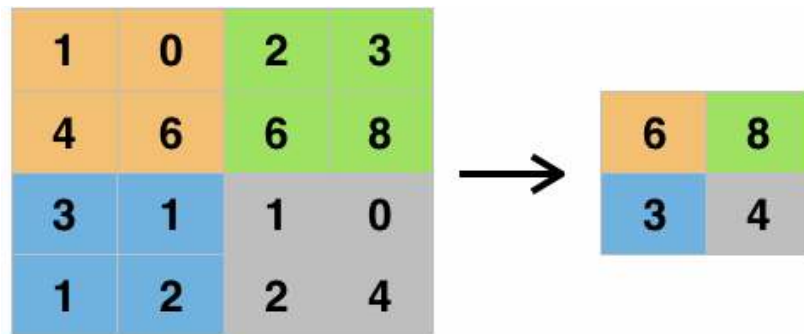
(<https://medium.com/@tuncerergin> : 08.05.2019).

ReLU fonksiyonu kısaca, negatif değerleri sıfır yaparken pozitif değerleri aynı çıkarmaktadır. Öznitelik haritasındaki siyah değerlere karşılık negatif değerler gelebilmekte olup, ReLu aktivasyonundan sonra siyah değerler kaldırılıp ve onun yerine sıfır eklenir (<https://medium.com/@tuncerergin> : 08.05.2019).

Havuzlama Katmanı (Pooling Layer):

Havuzlama katmanı, CNN'deki ardışık olan konvansiyonel katmanlar arasına; gösterimin kayma boyutunu, ağdaki kullanılan parametreler ile hesaplama sayısını azaltmak için eklenen katmandır. Bu işlem sayesinde ağ içerisindeki uyumsuzluklar denetlenmektedir. Birden fazla havuzlama (pooling) işlemi olmasına karşın genellikle tercih edilen çeşidi maksimum havuzlama (max pooling)dır. Şekil 24'te yer alan görsele göre maksimum havuzlama işlemi açıklanmıştır.

Şekil 24: CNN'de Maksimum Havuzlama İşlemi



(<https://medium.com/@tuncerergin> : 08.05.2019).

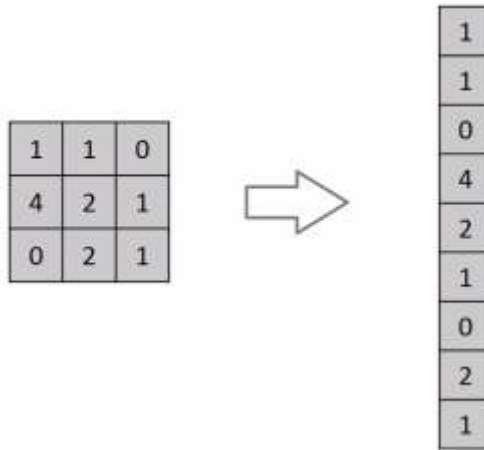
Şekil 24'te de görüldüğü üzere, biri görüntü matrisi (4x4 boyutunda) ve diğeri filtre matrisi (2x2 boyutunda) olmak üzere iki adet matris vardır. Filtre matrisi, görüntü

matrisi üzerine uygulandığında, işlem gördüğü alandaki sayılar içerisinde en büyüğünü alır ve tutar. Bu sayede öznitelik sayısı azaltılıp ağı eğitilmesi de kolaylaşır. Bunun yanında, eğitim setinin ezberlenmesinin de önüne geçilebilir. Maksimum havuzlama sayesinde ağı doğru karar verebilmesini sağlamak için gereken bilgiyi içeren çok daha küçük çıktılar kullanılmaktadır (<https://medium.com/@tuncerergin> : 08.05.2019).

Düzleştirme Katmanı (Flattening Layer):

Düzleştirme katmanının en büyük işlevi, ağı en önemli ve aynı zamanda son katmanı olan tam bağlı katmanın girişindeki verileri işlenmeye hazır hâle getirmektir. Sinir ağları, verileri çoğunlukla tek boyutlu bir diziden alır. Bu ağda yer alan veriler ise, konvolüsyon ve havuzlama katmanlarından gelen matrislerin tek boyuta dönüştürülmüş hâlidir. Dönüştürme işlemine bir örnek Şekil 25'te görülmektedir.

Şekil 25: Düzleştirme Katmanında Yapılan Dönüştürme İşlemi



(<https://medium.com/@tuncerergin> : 08.05.2019).

Tam Bağlı Katman (Fully-Connected Layer):

Bu katman, düzleştirme katmanından verileri alır ve sinir ağı yolu sayesinde öğrenme işlemini yerine getirir. Klasik yapay sinir ağı yapısını oluşturup gelen sinyalleri ağırlıklandırıp çıkışa iletmektedir.

Tam bağlı katmanda öznitelik haritası matrisi vektör olarak dönüştürdükten sonra bu özellikler bir model oluşturmak için bir araya getirilir ve çıktılar softmax veya sigmoid gibi aktivasyon fonksiyonları ile sınıflandırılır (<https://medium.com/@RaghavPrabhu> : 08.05.2019). CNN'de softmax katmanı tam

bağlı katman ile sınıflandırma katmanı arasına eklenir (<https://www.mathworks.com/> : 08.05.2019).

CNN'ler sayesinde görüntü sınıflandırılması yapıldığı gibi sürekli verilerin (açı, mesafe) tahmin edilmesi, regresyon modellerinin tahmini de yapılabilmektedir. CNN'de regresyon katmanı tam bağlı katman sayısının 1 nöron olarak belirlenmesinden sonra ağı en sonuna eklenir (<https://www.mathworks.com/> : 08.05.2019).

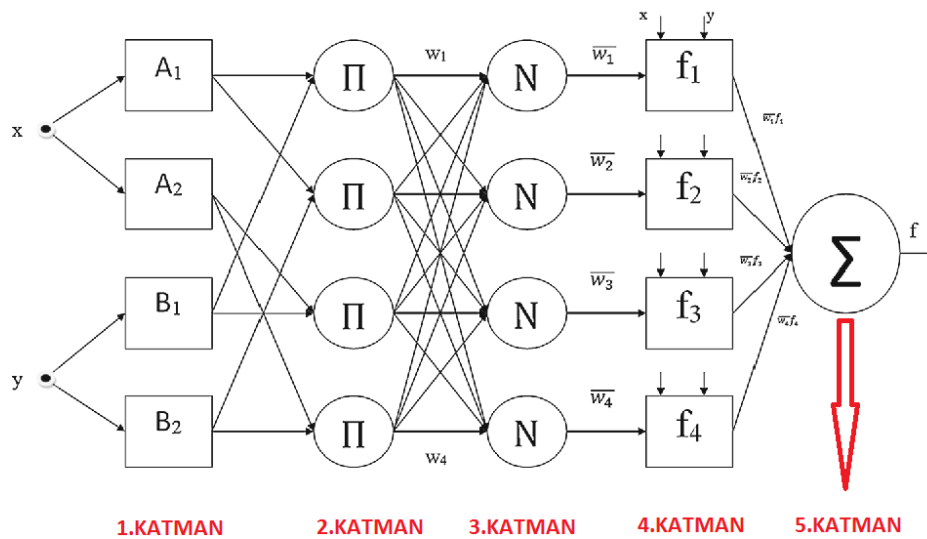
3.2.2. Adaptif Ağ Tabanlı Bulanık Çıkarım Sistemi (ANFIS)

Yapay sinir ağlarının paralel hesaplama ve öğrenme yeteneği ile bulanık mantığın çıkarım özelliğini kullanan melez (hibrid) bir yapay zekâ yöntemidir (Jang, 1993). Sugeno tipi bulanık sistemlerin sinirsel öğrenme kabiliyetine sahiptir. ANFIS model belirli fonksiyonları gerçekleştirmek için katmanlar hâlinde birbirine bağlı düğümlerin (node) birleşiminden oluşmaktadır (Akarslan ve Caner, 2009; Tsoukalas, 1996).

Adaptif ağlar, sistem tanımlamada kullanılmaktadır. ANFIS modelinde kullanıcı tarafından sisteme verilen giriş-çıkış (input-output) verileri, en iyi parametreler ve en iyi ağ yapısı belirlenerek, kullanıcıya en uygun modelleme imkânı sunulmaktadır (Jang vd., 1997; Demirel vd., 2010).

Şekil 26'da ANFIS model yapısı ve katmanları ile bu modelin nasıl işlediği gösterilmektedir.

Şekil 26: 2 Girdi, 1 Çıktı ve 4 Kural Oluşturulmuş Modellenmiş Bir ANFIS Yapısı

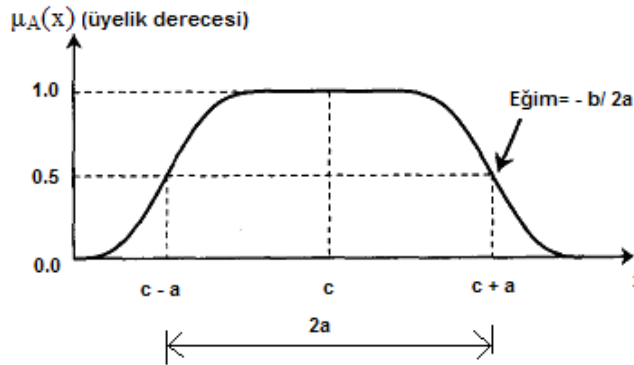


(<https://www.researchgate.net/> : 07.05.2019)

Şekil 26’da da anlaşılaçağı gibi ANFIS modeli 5 katman (*layer*) ve ileri beslemeli bir yapay sinir ağı mimarisine sahiptir.

Jang (1993)’ın ortaya koymuş olduğı ANFIS modelinde birbirine bağlanmış düğümlerden oluşan adaptif (uyumlu) ağların her biri bir işlemi temsil etmektedir. Düğümler arasındaki bağlar, değeri tam olarak bilinmeyen bir ağırlığı temsil eder. Adaptasyon, bu düğümlerin çıkış değerlerinin değiştirilebilir parametreler ile belirlenmesi sayesinde oluşmaktadır. Öğrenme kuralları, ağıdaki çıkış değerleri ile hedeflenen değerler arasındaki fark olan hatayı en aza indirgeyecek şekilde nasıl değiştirileceğı ile ilgilenir. ANFIS modelinde bulanıklaştırma işleminin gerçekleştirilmesi için Şekil 27’de gösterilen Gauss aktivasyon modeli kullanılmaktadır.

Şekil 27: Gauss Eğrisi Üyelik Fonksiyonu



(Demirel vd., 2010)

Şekil 27’te yer alan x , ilgili düğümün giriş değeri; A_i ise bu düğüme ait bulanık kümeye ait değerlerdir. $\mu_{A_i}(x)$, 0 ve 1 arasında normalize edilen bir Gauss eğrisi olarak değerlendirilir (Demirel, 2010).

ANFIS modelindeki katmanlarla, bu katmanlara bağlı düğümler ve işlevleri aşağıdaki gibidir.

1. Katman: Giriş değerlerinin verildiğı ve “bulanıklaştırma katmanı” olarak da bilinen katmandır. Bu katmanda yer alan hücreler adaptif hücreler olup, bu hücrelerin sayısı girdi değerlerinin sayısına eşittir. Bu katmanda yer alan hücrelerin çıktıları (O_{i1}) için Denklem(8) verilmektedir (Jang vd., 1995; Haznedar ve Kalınlı, 2015).

$$O_{i1} = \mu_{A_i}(x)_i = 1,2 \quad (8)$$

Denklem(8)'de yer alan A_i herhangi bir bulanık küme parametresi ve μ_{A_i} bu küme için belirlenen üyelik derecesidir.

Bu üyelik derecesi Gauss aktivasyon modeli ile kullanıldığında μ_{A_i} için aşağıdaki Denklem (9) verilir.

$$\mu_{A_i} = \frac{1}{1 + \left[\left(\frac{x - c_i}{a_i} \right)^2 \right]^{b_i}} \quad (9)$$

Denklem(9)'da yer alan a_i : üyelik fonksiyonu sigması, b_i : üyelik fonksiyonu eğimi ve c_i : üyelik fonksiyonunun merkezidir.

2. Katman: Kural katmanı olarak adlandırılır. Bu katmanda yer alan her düğüm Sugeno tipi bulanık çıkarım sistemine göre hesaplanan kurallardan meydana gelir. Bu katmanda bulunan hücre sayıları sabit olup, bu hücrelerin sayıları oluşturulan kuralların sayısına eşittir. Bu katmandaki hücrelerin girdileri, kuralların öncül kısmındaki değişken değerlerine ait üyelik fonksiyonu değerleridir. Hücre çıktıları (O_{2i}), kuralların ağırlıklarını (w_i) verir. Her bir kural düğümüne ait çıktı μ_i , 1. katmandan gelen üyelik derecelerinin çarpımıdır. Denklem (10)'da hücre çıktılarına ait eşitlik verilmiştir.

$$O_{2i} = w_i = \mu_{A_i}(x) \cdot \mu_{B_i}(y) \quad i = 1, 2 \quad (10)$$

3. Katman: Normalizasyon katmanıdır. Bu katmanda yer alan her düğüm, kural katmanından (2. Katman) gelen düğümleri birer giriş değeri olarak algılar ve her bir kuralın “normalleştirilmiş ateşleme seviyesini” ölçer. Bu katmandaki hücreler sabit hücreler olup, girdileri bir önceki katmandan aldığı ağırlık derecelerdir. Bu katmanda, ağırlık dereceleri normalize işlemine tabi tutulur. Denklem (11)'de normalize işleminin eşitliği verilmiştir.

$$\bar{\omega} = \frac{\omega_i}{\omega_1 + \omega_2}, i = 1, 2 \quad (11)$$

4. Katman: “Berraklaştırma” ya da “arındırma” katmanı olarak adlandırılır. Bu katmanda yer alan her düğümde verilen bir kurala ait ağırlıklandırılmış sonuç değerleri ölçülmektedir. Bu katmanda yer alan parametreler “sonuç parametreleri” olarak adlandırılmaktadır. Bu katmanda bulunan düğümlerin çıktı değeri Denklem (12)'de verilmiştir.

$$O_i^4 = \bar{\omega}_i f_i = \omega_i (p_i x + q_i y + r_i) \quad (12)$$

5. Katman: Bu katman toplam katmanı olarak isimlendirilir. Bu katmanda sadece bir tane düğüm vardır ve bu düğüm $\sum$ şeklinde ifade edilir. Bu katmanda, 4. katmanda bulunan her bir düğüm toplanır ve sonuç olarak ANFIS modelinin gerçek değerine ulaşılır. Sisteme ait çıktı ise Denklem (13)'te gösterilen eşitliğe göre elde edilmektedir (Haznedar ve Kalınlı, 2016; Hoccoğlu ve Kurban, 2005; Başkan, 2004; Fırat vd., 2008; Günay vd., 2006; Civelekoğlu, 2006; Özgan, 2009).

$$O_{5i} = f = \sum \bar{w}_i \cdot f_i = \frac{\sum w_i f_i}{\sum w_i} \quad i = 1, 2 \quad (13)$$

3.2.3. Eğri Uydurma (Curve Fitting)

Verilere eğri veya eğriler uydurma işlemi yapay zekânın temelini oluşturup, ilk yöntemler arasında gelmektedir. Eldeki verilere, matematiksel bir model oluşturmak için bilinen değerlerden yola çıkarak bilinmeyen ara değerlerinde hesaplanabilmesi amacıyla interpolasyon işlemi uygulanmaktadır. İnterpolasyon, regresyon ile yakından ilişkili olup eğri uydurma ile gerçekleştirilmektedir. Uydurulacak eğriye ait denklem katsayıları genellikle regresyon ile tespit edilmektedir. Bunun içinde literatürde oldukça popüler olan en küçük kareler yöntemine göre hesaplama yapılmaktadır.

Bağımlı ve bağımsız değişkenlerin sırası ile Y ve x olduğu varsayılan bir durumda, eğer bu iki değişken arasındaki ilişki parabolik ise; Denklem (14)'teki eşitlik kullanılmaktadır.

$$Y = a + bx + cx^2 \quad (14)$$

Bağımlı değişken ve bağımsız değişken arasındaki ilişki hiperbolik ise; Denklem (15)'teki eşitlik elde edilir.

$$Y = 1/(a + bx) \quad (15)$$

Y ve x arasındaki ilişki üstel ise; Denklem(16)'daki eşitlik, eğer bu ilişki logaritmik ise de genellikle Denklem (17) kullanılmaktadır (Adıyaman, 2007).

$$Y = ax^b \quad (16)$$

$$\ln Y = \ln a + b \ln x \quad (17)$$

Bazı problemlerin çözümünde bağımsız değişkenlerden meydana gelen fonksiyonlara veya X_i , Y_i noktalarına ait veri setlerine gereksinim duyulmaktadır. Eğri

uydurma yönteminde, fonksiyonlar polinomlara eğri uydurma için kullanılmaktadır (Bayılmış, Sayısal Analiz Ders Notları; <https://www.mathworks.com>, 06.05.2019).

Denklem (18)'de eğri uydurma işleminin uygulandığı n . Derece polinomsal bir fonksiyon eşitliği görülmektedir.

$$P_n(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n \quad (18)$$

3.2.3.1. En Küçük Kareler Yöntemi

Uydurma işleminin uygulandığı fonksiyon (uydurulan fonksiyon) ile ölçülerek elde edilen fonksiyon değerleri arasındaki farkın kareleri toplamı en aza indirgenmeye sağlanır. Burada amaç, ölçüm sonuçlarına ait değerlere mesafe açısından en az hatalı eğriyi veren fonksiyonu elde etmektir (Bayılmış, Sayısal Analiz Ders Notları).

Denklem (18)'de verilen polinoma ait katsayıların bulunması için kullanılan en küçük kareler yöntemi Denklem (19)'daki gibi hesaplanmaktadır.

$$c = (A' * A)^{-1} * A' * Y \quad (19)$$

Denklem (19)'de c katsayılar $[a_0, a_1 \dots a_n]$ 'i ifade ederken, Y bağımlı değişkeni, A ise; $[x^0, x^1 \dots x^n]$ değerlerini içeren Vandermonde matrisini ifade etmektedir.

3.3. Öğrenme Algoritmaları

Makine öğrenme algoritması, veriden öğrenebilen algoritmadır. Öğrenme algoritmaları kendilerine verilen verilerden gerçekleştirecekleri görevi ve bunun için kullanacakları performans ölçütünü tecrübelerine göre belirlerler. Bu tecrübeler daha önce hangi veriden nasıl bir performansla bir öğrenme gerçekleştirecekleridir (Goodfellow vd., 2016).

İnsan düşünce yapısının modellenmesini ve makinelere aktarılmasını hedefleyen yapay zekâ ile ilgili çalışmalar 1980'li yıllarda oldukça gerilemiştir. 1980'den sonra YSA'ların geliştirilmesi ile beraber yapay zekâ tekrar popüler bir alan haline gelmiştir. Özellikle CNN'lerin geliştirilmesi sayesinde ortaya çıkan derin öğrenme (LeCun vd, 2015) etiketlenmiş veriler arasında öznitelik saptayabilmek için sinir ağlarının katmanlarının derinleştirilmesi olarak tanımlanabilir (<http://www.derinogrenme.com> : 09.05.2019).

Her ne kadar derin öğrenme görüntü işleme tabanlı bir yarışmada bir kavram olarak ortaya atılsa da, özellikle doğrusal olmayan verilerin analizinde göstermiş olduğu en az hata ve maksimum başarı oranı ile birçok alanda kullanılmaktadır (<https://medium.com/deep-learning-turkiye> : 09.05.2019). Literatürde özellikle derin öğrenme alanında stokastik dereceli azalma yöntemi (SDA), RMSprop algoritması ve Uyarlanabilir Anlar (Adaptive Moments “Adam”) yöntemleri sıklıkla kullanılmaktadır.

3.3.1. Stokastik Gradyan İnişi (Stochastic Gradient Descent)

Stokastik gradyan inişi (SGD), YSA’ların ağırlıklarının güncellenmesinde kullanılan dereceli azalma algoritmasının bir uzantısıdır. Günümüzde makine öğrenme algoritmaları dereceli azalma yöntemi ile eğitilmektedir (Goodfellow vd., 2016). Çünkü makine öğrenmesinin amaçlarından biri, eğitim verileri düşünüldüğünde en iyi sonuç, en başarılı oran ve en düşük hatayı bulmaktır. Gradyan inişi, “maliyet fonksiyonunu” en aza indirgeyerek, en düşük hatayı bulmak için kullanılmaktadır. Dereceli azalma yönteminin türlerinden biri olan SGD, diğer dereceli azalma yöntemlerine göre nispeten daha hızlıdır. Çünkü SGD, eğitim verisi için ağırlıkları güncellerken tek bir ağırlık için bütün bir eğitim setini güncellemez, her bir eğitim seti için güncelleme yapılır. Bu yüzden çok hızlıdır ve bellek sorunu yaratmaz. Ayrıca, en düşük hatayı bulurken yerel en düşük değere (lokal minimum) takılmaması için bir eşik (hassasiyet) değeri belirlenir. Belirlenen eşik değeri aşıldığında çalışma durdurulur (Goodfellow vd, 2016; <https://medium.com/deep-learning-turkiye> : 09.05.2019). Denklem 20 ve Denklem 21’de k eğitim adımındaki SGD’nin güncelleme işlemi algoritma olarak verilmektedir (Goodfellow vd, 2018).

ϵ_k Öğrenme oranı, θ Parametresine ait ilk değer,

while algoritma çalıştığında **do**

$\{x^{(1)}, \dots, x^{(m)}\}$ eğitim kümesine ait m örneklik bir mini-yığını karşılayacak olan $y^{(i)}$ hedefleri ile beraber örneklenmesi

$$\text{Gradyan kestiriminin hesaplanması: } \hat{g} \leftarrow +\frac{1}{m} \nabla_{\theta} \sum_i L(f(x^{(i)}; \theta), y^{(i)}) \quad (20)$$

$$\text{Güncellenmenin uygulanması: } \theta \leftarrow \theta - \epsilon \hat{g} \quad (21)$$

end while

SGD algoritması için en önemli parametrelerden biri de öğrenme oranıdır. Denklem 20’de sabitlenmiş olarak verilen ϵ değeri, zaman içerisinde kademeli olarak azaltılmalıdır. Bu sebeple, k adımdaki öğrenme oranı ilerleyen bölümde ϵ_k olarak gösterilmektedir (Goodfellow vd., 2018).

Bu değişikliğin sebebi, SGD gradyan kestiriminin minimuma ulaştığı zaman ortadan kaybolmayan gürültü kaynağını (eğitim kümesinin m rastgele örneği) içermesinden kaynaklanmaktadır. Bu kestirime oranla, toplam maliyet fonksiyonuna ait gerçek gradyan, yığın gradyan inişini kullanıp bir minimum değere yaklaştığı zaman çok daha küçülür ve akabinde minimum noktasında sıfıra ulaşmış olur. Yani, yığın gradyan inişi sabitlenmiş öğrenme oranının kullanılmasında bir sakınca olmayacaktır. Denklem 22 ve Denklem 23’te SGD yakınsamasını garantileyen koşul verilmektedir (Goodfellow vd., 2018).

$$\sum_{k=1}^{\infty} \epsilon_k = \infty \quad (22)$$

$$\sum_{k=1}^{\infty} \epsilon_k^2 < \infty \quad (23)$$

Pratik olarak, τ adımına kadar, öğrenme oranının $a = \frac{k}{\tau}$ parametre ataması yapılarak, Denklem 24’teki gibi doğrusal bir şekilde azaltılması da oldukça sık kullanılmaktadır.

$$\epsilon_k = (1 - a)\epsilon_0 + a\epsilon_{\tau} \quad (24)$$

Genellikle, τ adımından sonra ϵ sabit tutulmaktadır (Goodfellow vd., 2018).

3.3.2. RMSprop (Root Mean Square Prop) Algoritması

RMSprop algoritması, bir diğer dereceli azalma yöntemi optimizasyonu olan AdaGrad’ın ağırlıklı hareketli ortalamasını değiştirerek dış bükey bir ortamda daha iyi bir performans sergilemesi için geliştirilmiştir (Hinton, 2012).

Derecenin (gradyan) boyutu çok büyük ya da çok küçük olabilir. Bu da global tek bir öğrenmeyi zor hale getirir. Toplu öğrenme gerçekleştirebilmek ve gradyan boyutunun çok ya da çok küçük olması durumunda ortaya çıkan hatayı önlemek için gradyanın boyutu yerine işareti kullanılır (Hinton, 2012). Bunun için;

- Ağırlık güncellemeleri aynı büyüklüktedir.
- Bu sebeple, çok zayıf, düşük gradyanlardan kaçış sağlanır.

RMSprop, her gradyanın ortalamasını almak yerine, bunların hareket ortalamalarını alır. Böylece gradyanın büyüklüğünden etkilenmez, adım adım ilerlemede yerel minimum değerine takılı kalmaz ve üstelik çok büyük ya da çok küçük tüm gradyanların eğitim setini çok hızlı bir şekilde öğrenir. Denklem 25-28 arasında verilen denklemlerde RMSprop algoritması verilmektedir (Goodfellow vd., 2018).

ϵ : Global öğrenme oranı, ρ : azalma oranı,

θ parametresinin ilk değeri,

δ küçük değerli sabiti (genellikle 10^{-6} değerinde varsayılan olarak ayarlanmıştır, küçük sayılarla bölünmeyi dengelemede kullanılmaktadır.),

Birikim değişkenine ait ilk değer atamasının gerçekleştirilmesi: $r = 0$

while durdurma koşulu sağlanmadığında (algoritma çalıştığı müddetçe) **do**

$\{x^{(1)}, \dots, x^{(m)}\}$ eğitim kümesine ait m örneklik bir mini-yığını karşılayacak olan $y^{(i)}$ hedefleri ile beraber örneklenmesi

$$\text{Gradyan hesaplamasının yapılması: } \hat{g} \leftarrow +\frac{1}{m} \nabla_{\theta} \sum_i L(f(x^{(i)}; \theta), y^{(i)}) \quad (25)$$

$$\text{Karesel gradyanın toplanması: } r \leftarrow \rho r + (1 - \rho) g \odot g \quad (26)$$

$$\text{Parametre güncellemesinin hesaplanması: } \Delta\theta = -\frac{c}{\sqrt{\delta+r}} \odot g \quad (27)$$

($\frac{1}{\sqrt{\delta+r}}$ eleman bazlı uygulanmaktadır)

$$\text{Güncellemenin uygulanması: } \theta \leftarrow \theta + \Delta\theta \quad (28)$$

Karpathy (2017)'e göre, RMSprop derin öğrenmede oldukça popüler ve kullanışlı olan en iyi algoritmalarından biridir ve sadece Adam algoritması'nın RMSprop algoritmasından daha iyi sonuçlar verebileceği belirtilmiştir.

3.3.3. Adam (Uyarlanabilir Anlar-Adaptive Moments) Algoritması

Adam öğrenme algoritması, İngilizce “uyarlanabilir anlar” anlamına gelen “adaptive moments” isminin kısaltmasıdır. Adam, adaptif bir öğrenme oranı optimizasyon algoritmasıdır (Goodfellow vd, 2016).

Derin öğrenmeye konu olabilecek işlemler genelde bir optimizasyon sorununa sahip konulardır. Doğrusal olmayan sorunların çözümü için optimizasyon tekniklerine başvurulmaktadır (<https://medium.com/deep-learning-turkiye> : 10.05.2019).

Adam, çeşitli derin öğrenme mimarileri için çalışan ve oldukça tercih edilen kullanışlı bir algoritma olup, dereceli azalma momentum (“gradient descent momentum”) algoritmasıyla RMSprop algoritmasının birleşimi olarak düşünülebilir (Kingma ve Ba, 2014; <https://engmrk.com> : 10.05.2019).

Adam algoritmasının çok kullanılmasının sebeplerinden biri de, tıpkı diğer dereceli azalma algoritmalarında olduğu gibi, maliyet fonksiyonuna en büyük doğruluk oranı, en düşük hata ve en iyi sonucu verecek şekilde uygulanmasıdır.

Adam algoritmasının beklenen performansı gösterebilmesi için varsayılan olarak belirlenmiş parametreleri vardır.

- Epsilon, sıfıra bölmeyi önlemek için çok küçük bir sayıdır (varsayılan değeri= 10^{-8}).
- Ortalamayı kontrol eden hiper parametreler vardır. Bu parametreler $p1$ ve $p2$ olarak kabul edilirse, varsayılan değer olarak $p1=0.9$ ve $p2=0.999$ değerlerini alırlar (Goodfellow vd., 2016).

Denklem 29-36 arasında verilen deklemlerde Adam algoritması verilmektedir (Goodfellow, 2018).

e adım büyüklüğü (Tavsiye edilen değer: 0.001)

Moment kestirimi için sırasıyla $p1$ ve $p2$ olarak adlandırılan üstel azalma oranları ($p1=0.9$, $p2=0.999$)

Sayısal denge için kullanılan δ küçük değerli sabiti (Tavsiye edilen değer: 10^{-8})

θ parametrelerinin ilk değerleri

Birinci ve ikinci moment değişkenlerine ait değer atamalarının yapılması:

$$s = 0, r = 0$$

Zaman adımının ilk değerinin atamasının yapılması: $t = 0$

while durdurma koşulu sağlanmadığında (algoritma çalıştığı müddetçe) **do**

$\{x^{(1)}, \dots, x^{(m)}\}$ eğitim kümesine ait m örneklik bir mini-yığı karşılayacak olan $y^{(i)}$ hedefleri ile beraber örneklenmesi

$$\text{Gradyan hesaplamasının yapılması: } \hat{g} \leftarrow +\frac{1}{m} \nabla_{\theta} \sum_i L(f(x^{(i)}; \theta), y^{(i)}) \quad (29)$$

$$t \leftarrow t + 1 \quad (30)$$

$$\text{Önyargılı ilk moment kestiriminin güncellenmesi: } s \leftarrow p_1 s + (1 - p_1) g \quad (31)$$

$$\text{Önyargılı ikinci moment kestiriminin güncellenmesi: } r \leftarrow p_2 r + (1 - p_2) g \otimes g \quad (32)$$

$$\text{Birinci momentteki önyargının düzeltilmesi: } \hat{s} \leftarrow \frac{s}{1-p_1^t} \quad (33)$$

$$\text{İkinci momentteki önyargının düzeltilmesi: } \hat{r} \leftarrow \frac{r}{1-p_2^t} \quad (34)$$

$$\text{Güncellenmenin hesaplanması: } \Delta\theta = -\epsilon \frac{\hat{s}}{\sqrt{\hat{r} + \delta}} \text{ (işlemler eleman bazlı uygulanır)} \quad (35)$$

$$\text{Güncellenmenin uygulanması: } \theta \leftarrow \theta + \Delta\theta \quad (36)$$

Adam, RMSprop yönteminde olduğu gibi öğrenme oranı ölçeklendirmesini gerçekleştirmek için kare gradyanları (gradyanların karesini) kullanır ve SDA momentum da olduğu gibi gradyanların yerine gradyanların hareketli ortalamasını kullanarak momentumdan faydalanır. Momentum, öğrenme algoritmasını yerel minimum değerinden kaçacak şekilde önceki yönde tutulması işlemidir (Karparthy, 2017; <https://towardsdatascience.com> : 10.05.2019).

Adam algoritması, uyarlanabilir (adaptif) bir öğrenme algoritması olup, verilen farklı parametreler için bireysel öğrenme oranlarının hesaplanmasında kullanılmaktadır (<https://towardsdatascience.com>:10.05.2019).

DÖRDÜNCÜ BÖLÜM

DENEYSEL ÇALIŞMA

BTC fiyatının TL cinsinden tahmin edilmesi için toplam 24 farklı zaman aralığı ve 24 farklı veri seti kullanılan bu tez çalışmasında, yapay zekâ tekniklerinden LSTM, ANFIS, Polinomsal Eğri Uydurma ve YSA kullanılmıştır. 24 veri setine uygulanan bu yöntemler arasında öncelikle model ve parametre olarak en iyilerinin belirlenmesi gerektiği üzerinde durulmuştur. BTC fiyatı dönemsel ve öngörülemez şekillerde değişim gösterdiğinden genel olarak en iyi sonucu veren modelin tespit edilmesi büyük önem arz etmektedir. Bu sebeple farklı parametreler ile yapay zekâ yöntemleri farklı zaman aralıklarında veri setleri ile test edilmiştir.

Çalışmanın Materyal ve Metot bölümünde kullanılacak olan veri setleri belirtilmiştir. Bu bölümde bu veri setlerine uygulanan yapay zekâ tekniklerinin model ve parametreleri detaylı bir şekilde tablolar ile sunulmaktadır. Farklı yöntemler ile elde edilen sonuçlar ışığında tüm veri setleri arasında en iyi sonucu veren parametreler istatistiki bir yöntem olan 5-kat çapraz doğrulama yöntemi ve zaman serisi şeklinde toplamda 5 kez çalıştırılarak belirlenmiştir. Her iki test yönteminde elde edilen test verisine ait RMSE değerlerinin ortalaması değerlendirilerek nihai parametreler ve veri seti için zaman aralıkları belirlenmiştir. Son olarak en uygun yöntemlerin gerçek zamanlı verilere uygulanabileceği bir model geliştirilmiştir.

Yöntemlere ait parametre çeşitleri ve alt algoritmaları aşağıdaki şekilde verilebilir.

Veri setlerinin YSA ile eğitilmesi için 3 farklı algoritma kullanılmıştır. Bu algoritmalar sırasıyla; ölçeklenmiş eşlenik gradyan (scaled conjugate gradient – SCG), bayes düzenlemesi (Bayesian Regularization-BR) ve levenberg-marquardt (LM) algoritmalarıdır. Tablo 8’de bu çalışmada test edilen YSA’ya ait parametreler görülmektedir.

Tablo 8: YSA Modeli ve Parametleri

Algoritma	İterasyon Sayısı	1.Katman Nöron Sayısı	2.Katman Nöron Sayısı
SCG-BR-LM	100	30	20
SCG-BR-LM	100	30	10

SCG-BR-LM	100	20	20
SCG-BR-LM	100	20	10
SCG-BR-LM	100	10	10
SCG-BR-LM	500	30	20
SCG-BR-LM	500	30	10
SCG-BR-LM	500	20	20
SCG-BR-LM	500	20	10
SCG-BR-LM	500	10	10
SCG-BR-LM	1000	30	20
SCG-BR-LM	1000	30	10
SCG-BR-LM	1000	20	20
SCG-BR-LM	1000	20	10
SCG-BR-LM	1000	10	10

ANFIS yönteminde kural oluşturmak için kullanılan MATLAB programındaki “genfis3” fonksiyonundan yararlanılarak, veri setlerinin sayısına uygun olarak 3 farklı kural yapısı oluşturulmuştur. Bu fonksiyon bulanık kuralların başlangıç yerlerini belirlemek için bulanık c-ortalamalar kümeleme yöntemini kullanmaktadır. Bu kural yapıları neticesinde toplamda 6 farklı kural sayısı elde edilmiştir. Elde edilen bu kurallar veri setlerinin tamamına uygulanarak 10, 100 ve 500 iterasyonda çalıştırılmıştır. Tablo 9’da ANFIS yöntemi ile ilgili parametreler detaylı bir şekilde verilmiştir.

Tablo 9: ANFIS Yönteminde Elde Edilen Kurallar ve Parametreleri

Kural Oranı	Kural Sayısı	İterasyon Sayıları
1	2	10,100,500
	3	10,100,500
	4	10,100,500
	6	10,100,500
	9	10,100,500
	11	10,100,500
2	2	10,100,500
	3	10,100,500
	4	10,100,500
	6	10,100,500
	9	10,100,500
	11	10,100,500
3	2	10,100,500
	3	10,100,500
	4	10,100,500
	6	10,100,500
	9	10,100,500
	11	10,100,500

Eğri Uydurma yönteminde polinom dereceleri arasında en iyi sonucun hangi derece ve/veya derecelerde bulunduğunu görmek için polinom dereceleri 3'ten başlayarak 20'ye kadar artırılıp veri setlerinin tamamına uygulanmıştır. Polinom derecesi çok yüksek olduğunda sıklıkla karşılaşılan “tam uydurma (over fitting)” hatası, yani öğrenme noktalarına yaklaşmak için eğilip, bükülme sorunundan dolayı çok yüksek dereceli eğriler kullanılmamaktadır (Alpaydın, 2018).

4.1. En Uygun Yöntemlerin Belirlenmesi

BTC/TL fiyat tahmininin yapılabilmesi için en uygun yöntemler ve bu yöntemlere ait parametrelerin belirlendiği bu bölümde, 24 veri setine uygulanan farklı yapay zekâ tekniklerinden elde edilen sonuçlar baz alınarak her teknik ve tekniğe ait en iyi parametreler alt başlıklar halinde verilmiştir. En uygun yöntemin belirlenmesindeki en önemli kriter olarak yapay zekâ tekniklerine ait algoritmaların MATLAB R2018b programında çalıştırıldığında elde edilen “en iyi değer (Best value)” yani en düşük RMSE değeri gelmektedir. En iyi değer, kullanıcıya tüm teknikler arasında en düşük değerin, başka bir deyişle en az hatanın olduğu noktanın hangi tekniğe ait olduğunu gösterir. Çalışmada en uygun yöntem ve parametreler belirlenirken eğitim ve test seti 5-kat çapraz doğrulama ile 5 kez çalıştırılıp test edilip, test veri setine ait ortalama en iyi değerlerin en çok bulunduğu setler dikkate alınmıştır.

YSA’da kullanılan 3 farklı algoritma arasından en iyi algoritma ve ona bağlı iterasyon ve nöron sayılarını belirlemek için tüm veri setlerinin test hataları toplamının (“test error”) 100,000’den küçük olduğu algoritmalar ve parametreler belirlenmiştir. Tablo 10’da YSA’ya ait algoritmalar ve parametrelerinin sahip olduğu toplam hatalar verilmiştir.

Tablo 10: YSA’daki Algoritma ve Parametrelere Ait Toplam Test Hataları

Algoritma	İterasyon Sayısı	1. Gizli Katman Nöron Sayısı	2. Gizli Katman Nöron Sayısı	Toplam Test Hatası
SCG	100	30	20	137899.4
		30	10	136956.3
		20	20	138862.8
		20	10	133730.0
		10	10	132966.6
	500	30	20	131741.2
		30	10	127505.9
		20	20	131021.4
		20	10	125403.7

	1000	10	10	123196.3
		30	20	126322.8
		30	10	126580.0
		20	20	122183.6
		20	10	124569.4
		10	10	116192.5
BR	100	30	20	113284.7
		30	10	115128.9
		20	20	108792.9
		20	10	113974.4
		10	10	111081.9
	500	30	20	97695.5
		30	10	100906.4
		20	20	89698.4
		20	10	101932.5
		10	10	95424.2
	1000	30	20	93099.4
		30	10	96253.0
		20	20	86300.2
		20	10	94874.9
		10	10	93949.8
LM	100	30	20	128773.2
		30	10	135307.0
		20	20	120837.1
		20	10	129565.9
		10	10	127742.8
	500	30	20	125466.3
		30	10	133745.4
		20	20	121297.0
		20	10	130179.8
		10	10	127986.5
	1000	30	20	129527.0
		30	10	132920.8
		20	20	122570.6
		20	10	132551.8
		10	10	126731.1

Tablo 10 incelendiğinde 24 veri setinde toplam test hatası 100,000'in altında olan algoritma ve parametreler koyu renk ile gösterilmiştir. İlk aşamada en iyi algoritma BR ve en iyi parametreler 500 ve 1000 iterasyon olduğu görülmüştür. 1000 iterasyonda çalışan BR algoritmasının tüm parametrelerinin en düşük toplam test hatası değerlerini verdiği görülmüş olup, BR algoritması için iterasyon sayısı 1000 olarak kararlaştırılmıştır. Ayrıca, nöron sayılarının toplam test hataları karşılaştırıldığında 20 nöron ve 10 nöron ile çalışılması belirlenmiştir. YSA yönteminde BR algoritması

kullanılarak 1000 iterasyonda ve toplamda 5 defa sinanan veri setlerine ait toplam test hatalarından en iyi ikisi Tablo 11’de verilmiştir.

Tablo 11: BR Algoritması İçin En İyi Parametrelerin Belirlenmesi

İterasyon Sayısı	1.Katman Nöron Sayısı	2.Katman Nöron Sayısı	Toplam Test Hatası
1000	20	20	86300.2
1000	10	10	93949.8

YSA’da elde edilen sonuçlara bakılarak, model eğitiminde en iyi algoritmanın BR olduğu, model eğitim başarısı için iterasyon sayısının fazla ama nöron sayısının çok fazla olmaması gerektiği yorumu yapılabilir.

ANFIS, 3 farklı kural tabanının her birinden 3 farklı kural sayısı oluşturularak veri setlerine uygulanmıştır. Kural sayılarına ait toplam test hatasına göre en iyi parametreler ve kural sayısı belirlenmeye çalışılmıştır. ANFIS kural modeline ait kural sayısı ve iterasyonları toplam test hataları ile birlikte Tablo 12’de verilmiştir.

Tablo 12: ANFIS Kural Oranı ve Parametreleri

Kural Oranı	İterasyon Sayısı	Toplam Test Hatası
1	10	58319.7
	100	62547.4
	500	63506.4
2	10	334304.2
	100	466495.8
	500	346516.7
3	10	90650.3
	100	97641.0
	500	106533.4

Tablo 12’de görüldüğü gibi en uygun parametreler toplam test hatasının 100,000’in altında olduğu 1.tip ve 3. tip kural oranlarının oluşturduğu kurallar olarak görülmektedir. Ancak bu iki kural tabanı arasında en iyi parametre tercihi yapıldığında bu açık bir farkla 1.tip kural yapısı olacaktır. 1. Tip kural tabanı belirlendikten sonra kendi içerisindeki iterasyon sayılarının karşılık geldiği toplam test hataları arasında çok büyük bir fark görülmemesine rağmen iterasyon sayısı arttıkça toplam test hatasının da orantılı bir şekilde arttığı görülmektedir. Bu sebeple en iyi model olarak belirlenen 1.tip kurala ait iterasyonlar arasında 10 iterasyon ve 100 iterasyonun veri setlerinin

eğitilmesinde kullanılmasına karar verilmiştir. ANFIS'in kural oluşturma yapısının sugeno ile sağlandığı tez çalışmasının önceki bölümlerinde açıklanmıştır. Bu tez çalışmasında verilerin ANFIS yöntemi ile eğitilmesi sağlanmadan önce kural sayılarının oluşturulmasında `genfis3` fonksiyonundan faydalanılmıştır. Bu fonksiyonun kullanımında veri seti büyüklüğü belirli bir sayının altında kaldığı zaman hata alınmaması için kural sayısının 3'ün altında olduğu veri setinde aynı kural sayısı otomatik olarak 2 defa oluşturulmuştur. Bu sebeple kural oranı terimi ile bu karmaşıklığa çözüm getirilmek istenmiştir. ANFIS'te seçilen 1 kural oranı ve ona ait iterasyon sayıları tekrar veri setinin eğitiminde kullanılmış ve bu defa iterasyon sayıları arasında en iyi ve/veya iyilerin seçimi yapılmış olup model ve modele ait parametreler Tablo 13'te gösterilmiştir.

Tablo 13: ANFIS Yönteminde En Uygun Parametreler

Kural Oranı	İterasyon Sayısı
1	10
1	100

Eğri uydurma yöntemine ait parametreler (polinom derecesi) ile veri setinin eğitilmesi ve tahmin edilmesi sonucunda ortaya çıkan toplam test hatası sonuçları Tablo 14'te verilmiştir.

Tablo 14: Eğri Uydurmaya Ait Parametreler ve Ortalama Test Hatası Karşılıkları

Polinom Derecesi	Toplam Test Hatası
20	135714733042.5
19	1214934.2
18	1026802.3
17	809792.4
16	625532.0
15	515696.9
14	408733.6
13	359803.0
12	522851.0
11	13495749.9
10	222164.8
9	174090.3
8	174045.1
7	115311.6
6	86014.2
5	75635.6
4	112950.3
3	187992.8

Tablo 14 incelendiğinde 75635.6 değeri ile 5. derece polinomsal eğri uydurmanın en iyi sonucu verdiği görülmüştür. Bunun yanında 7.derecenin ortalama test hatasının 100,000 den daha fazla olmasına karşın 7.derecenin de analizlere eklenilmesine karar verilmiştir. Çünkü daha önce bahsedilen “en iyi değer” yani en düşük hatayı 24 veri seti içerisinde 3 defa bulan 7.derecenin ortalama test hatasının da çok yüksek olmamasından dolayı seçilmiştir. Ayrıca ortalama test hatasının altında bir değere sahip olan 6.derecenin seçilmemesinin sebebi 24 veri seti içerisinde en iyi değeri tahmin edememiş olması olarak bilinmelidir. Unutulmamalıdır ki ortalama test hatası en uygun parametrelerin belirlenmesinde çok önemli bir faktör olarak kabul edilse de en iyi değer de tahmin edilip bulunduğu parametrelerde en uygun parametrelerin belirlenmesinde büyük rol oynamaktadır. Eğri uydurma yöntemi için seçilen polinom dereceleri ve bu derecelere ait ortalama test hataları Tablo 15’te verilmiştir.

Tablo 15: Eğri Uydurma Yöntemi İçin Belirlenen En Uygun Parametreler

Polinom Derecesi	Ortalama Test Hatası
7	115311.6
5	75635.6

4.2. Yöntemlerin Uygulanması ve Bulgular

BTC/TL fiyat tahmininin yapıldığı bu çalışmada, daha önceki bölümlerde yapay zekâ yöntemleri arasında YSA, ANFIS ve eğri uydurmaya ait en iyi parametreler belirlenip, 24 veri setine uygulanmıştır. Kullanılan yöntemlere ait teknikler ve algoritmalarda giriş verisi olarak verilen tarih ve Dolar/TL fiyatı öznitelikleri sonucunda BTC/TL fiyat tahmini yapılması amaçlanmıştır. Tahminleme işlemi aynı zamanda zaman serisi (time series) kullanılarak, aynı parametrelerle aynı veri setlerine uygulanmıştır. Bundan dolayı deneysel sonuçlar kısmı zaman serisi ve 5-katlı çapraz doğrulamanın kullanıldığı standart analiz olarak 2 başlıkta verilmiştir.

4.2.1. YSA, ANFIS ve Eğri Uydurma Yöntemlerinin En İyi Parametrelerle Çapraz Doğrulama Kullanılarak Uygulanması

Çalışmanın bu bölümünde YSA, ANFIS ve eğri uydurma yöntemleri için belirlenen en iyi parametrelerin 24 veri seti ile çalıştırıldıktan sonra, 3 yöntemin toplam

test hatalarının ortalaması 100 ile çarpılıp veri setinin fiyat farkına bölünüp veri seti hata oranı elde edilmiştir. Denklem (37)'de hata oranını elde eden eşitlik verilmiştir.

$$\chi = (100 * (\frac{1}{3} * \sum_{i=1}^3 a_i)) / (\beta_{max} - \beta_{min}) \quad (37)$$

Denklem (37)'deki χ veri seti hata oranını, α_i her bir yöntemin test hatasını, β_{max} en yüksek BTC fiyatı ve β_{min} ise en düşük BTC fiyatını ifade etmektedir.

YSA, ANFIS ve eğri uydurma yöntemlerine ait modellerin en iyi parametrelerinin 24 veri setine uygulandığında elde edilen sonuçlar Tablo 16'da gösterilmiştir. Test ve eğitim verileri için standart sapma değerleri ortalama test hatası ve ortalama eğitim hatası ile birlikte verilmiştir.

Tablo 16: Veri Setlerine Ait Hata Oranlarının Belirlenmesi

Veri Seti	En Yüksek Fiyat (BTC/TL)	En Düşük Fiyat (BTC/TL)	Fiyat Farkı (BTC/TL)	Ortalama Test Hatası (RMSE)	Ortalama Eğitim Hatası (RMSE)	Veri Seti Hata Oranı (%)
1_1	18700.00	17755.00	945.00	396.92±16.72	166.63±7.45	42.00
1_2	18700.00	17755.00	945.00	263.10±8.31	199.32±5.47	27.84
1_3	18700.00	17755.00	945.00	224.62±3.71	202.49±2.91	23.77
1_4	18700.00	17755.00	945.00	305.58±4.14	295.10±3.44	32.34
2_1	22140.00	17601.00	4539.00	424.02±35.19	309.10±30.54	9.34
2_2	22150.00	17601.00	4549.00	371.47±57.09	316.83±48.15	8.17
2_3	22150.00	17500.00	4650.00	887.15±250.53	471.61±34.68	19.08
2_4	22150.00	17500.00	4650.00	1068.22±485.11	656.01±83.58	22.97
3_1	23270.00	18100.00	5170.00	1016.21±70.03	565.40±45.00	19.66
3_2	23400.00	18100.00	5300.00	826.58±47.83	617.59±58.71	15.60
3_3	23400.00	17500.00	5900.00	761.29±38.35	640.82±43.49	12.90
3_4	23400.00	17500.00	5900.00	890.24±130.43	788.50±27.89	15.09
4_1	35128.00	18100.00	17028.00	1459.18±121.24	981.35±129.46	8.57
4_2	35128.00	18100.00	17028.00	1272.59±121.13	1033.77±186.28	7.47
4_3	35128.00	18100.00	17028.00	1117.19±183.51	1009.31±225.96	6.56
4_4	35128.00	17601.00	17527.00	1712.22±356.30	1645.49±342.01	9.77
5_1	77850.00	21940.00	55910.00	9370.72±696.80	4700.56±530.81	16.76
5_2	77850.00	19950.00	58300.00	6415.53±644.66	4818.79±521.11	11.00
5_3	77850.00	19950.00	58300.00	4767.80±404.25	4396.50±393.90	8.18
5_4	77850.00	18100.00	59750.00	7852.52±359.47	7703.93±370.85	13.14
6_1	58350.00	2616.00	55734.00	13607.58±479.96	7678.79±664.06	24.42
6_2	74509.00	2616.00	71893.00	10579.43±875.67	8511.49±899.35	14.72

6_3	74509.00	2616.00	71893.00	8243.69±933.15	7387.48±734.44	11.47
6_4	77662.00	2616.00	75046.00	8176.85±777.14	8023.93±705.25	10.90

Tablo 16’da elde edilen sonuçlar için kullanılan öznitelikler normalize edilmemiş standart verilerden oluşmakta olup, [0 1] aralığında minimax normalizasyonlu öznitelikler ile gerçekleştirilen testlerde herhangi bir iyileşme gözlenmeyip bazı yöntemlerde kötüleşme tespit edilmiştir. Bu sebeple çapraz doğrulama testlerinde normalizasyon uygulanmayan özniteliklerin kullanılması sonucuna ulaşılmıştır.

Denklem (37)’deki eşitlik veri seti 1_1 için uygulanacak olursa; $(100 \cdot 396.9246) / 945 = 42.00$ hata oranı elde edilecektir. Bu hata oranı, veri seti 1_1 içerisindeki BTC/TL fiyat tahmininde %42 oranında bir hata olduğunu ifade etmektedir.

Şu ana kadar YSA, ANFIS ve eğri uydurma yöntemlerinin en iyi parametreleri bulunmaya çalışılmıştır. Elde edilen en iyi parametrelere sahip üç farklı yapay zekâ yönteminin 24 farklı veri setine uygulanıp elde edilen sonuçlar ise Tablo 16’da gösterilmiştir. Bu aşamadan sonra ise, elde edilen Tablo 16’daki sonuçlar ışığında en iyi veri setlerinin belirlenmesi gerçekleştirilmiştir.

Tablo 16’da verilen sonuçlar incelendiğinde 6 ana veri seti arasından hangilerinin seçileceği, veri seti hata oranı sütunundaki veriler neticesinde belirlenmiştir. Buna göre koyu renkte yazılmış olan sonuçlar veri seti zaman aralığı da hesaba katıldığında tahmin oranının en iyi yapıldığı zaman aralıklarını temsil etmektedir. Tablo 17’de en iyi veri setleri verilmiştir.

Tablo 17: En İyi Sonuçları Veren Veri Setleri

Veri Seti	Hata Oranı (%)
2_2	8.17
4_3	6.56
5_3	8.18
6_4	10.90

Tablo 18’de daha önce belirlenen en iyi yöntem ve bu yöntemlere ait parametrelerin birbirleri arasında kıyaslanmaları sonucunda seçilen en iyi parametreler hata değerleri ile birlikte verilmiştir.

Tablo 18: Yöntemlere Ait Parametrelerin Birbirleri İle Kıyaslanması

Yöntem	1.Parametre	Toplam Test Hatası (RMSE)	2.Parametre	Toplam Test Hatası (RMSE)
YSA	1000 İterasyon 20 Nöron-20 Nöron	86300.17	1000 İterasyon 10 Nöron-10 Nöron	93949.78
ANFIS	2 Kural-10 İterasyon	58319.72	2 Kural-100 İterasyon	62547.41
Eğri Uydurma	5. Derece	75635.61	7.Derece	115311.56

Tablo 18'deki karşılaştırmalar neticesinde yöntemlerin daha az test hata oranına sahip parametreleri ile çalıştırılması kararlaştırılmış olup en iyi yöntem ve bu yöntemlere ait parametrelerin birbirleri arasında kıyaslanmaları sonucunda seçilen en iyi parametreler:

- YSA için; 1000 İterasyon-20 Nöron-20 Nöron,
- ANFIS için: 1 Kural oranı 10 İterasyon(X-Kural Sayısı 10 İterasyon),
- Eğri Uydurma için; 5. Dereceden eğri

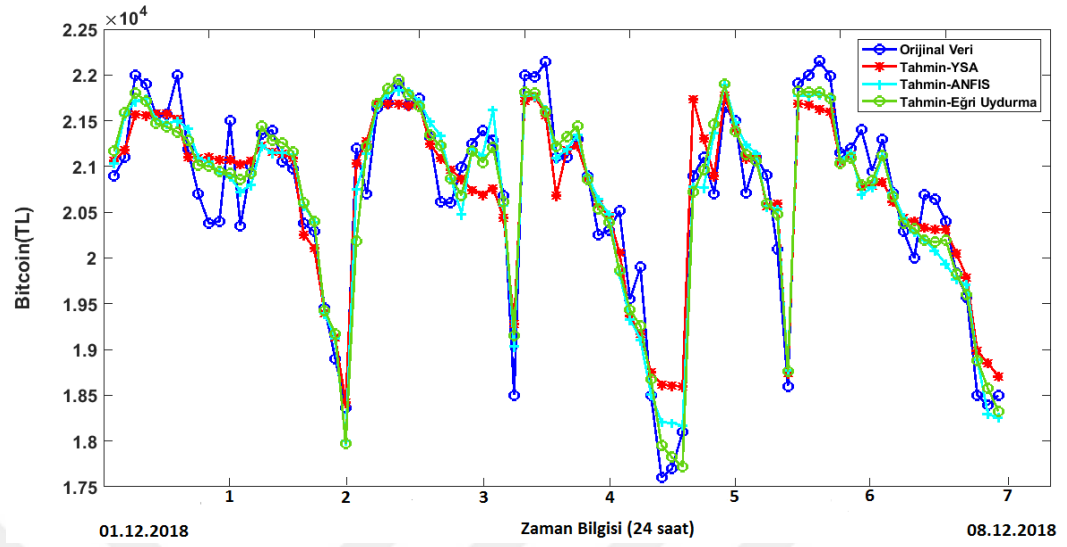
olarak bulunmuştur. ANFIS için en iyi parametre olarak bulunan 1 kural oranı, az veri sayısından dolayı genfis3 fonksiyonu ile oluşturulmuş kural yapısından kaynaklanan aynı 2 kuraldan birincisidir. Bunun yanında Tablo 18'de belirtilen en iyi ikinci parametrelerden bu aşamada tamamen vazgeçilmeyip zaman serisi analizinde de değerlendirilmiştir.

Tablo 17'de de yer alan ve çapraz doğrulama sonucunda belirlenen en uygun veri setlerinde, Tablo 18'de belirtilen en iyi yöntemlerin test seti sonuçları Şekil 28-31 arasında verilmiştir.

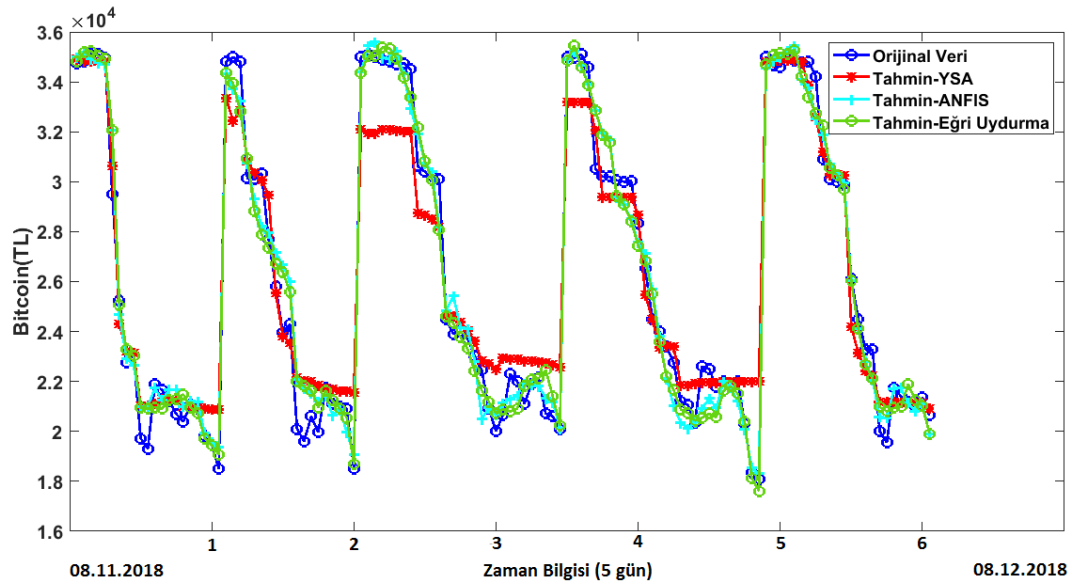
Şekil 28-31 arasında verilen sonuçlarda başlangıç tarihi sağ alt köşede, bitiş tarihi ise sol alt köşede verilmiştir. Şekillerin ait olduğu farklı veri seti grupları, farklı zaman aralıklarına karşılık geldiği için zaman aralıkları, yatay ekseninde “zaman bilgisi” olarak verilmiştir. Örneğin, 6 saatlik aralıklarla 1 aylık verilerin karşılık geldiği 4_3 veri seti grubunda zaman aralığı 5 gün olarak ve 6 farklı zaman dilimi olarak verilmiştir. Her bir zaman dilimi 5 gün ile çarpıldığında, son zaman dilimi olan 6'nın karşılık geldiği sayı 30, yani 1 ay olacaktır.

Yukarıda zaman bilgisi ve zaman dilimi ile yapılan hesaplama Şekil 28-31 arasında verilen tüm şekiller için geçerlidir.

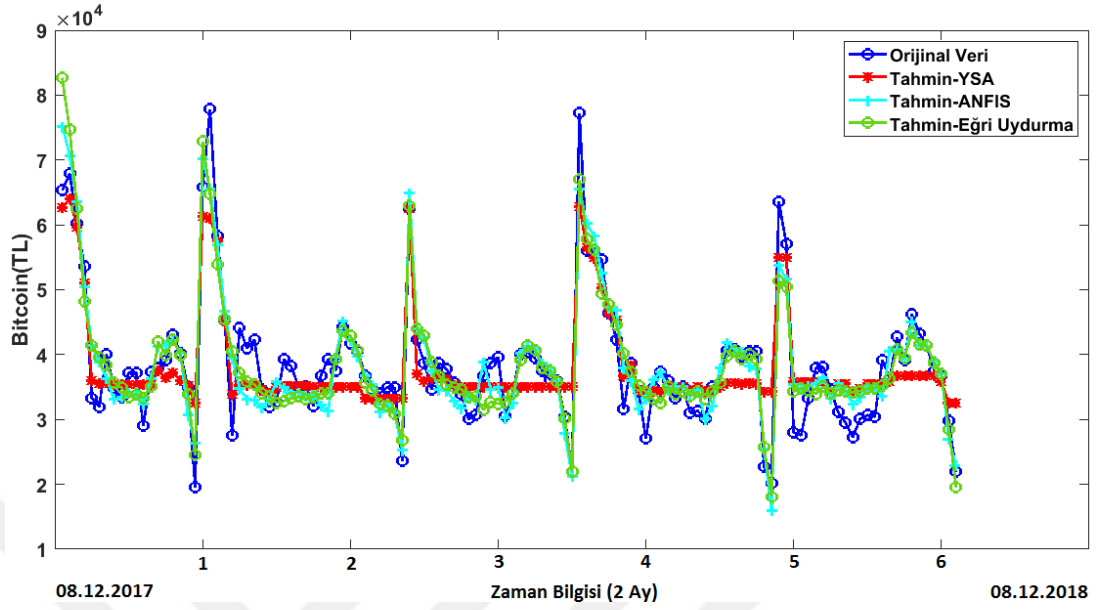
Şekil 28: 2 Saatlik Aralıklarla 1 Haftalık (2_2) Veri Setine Ait Sonuçlar



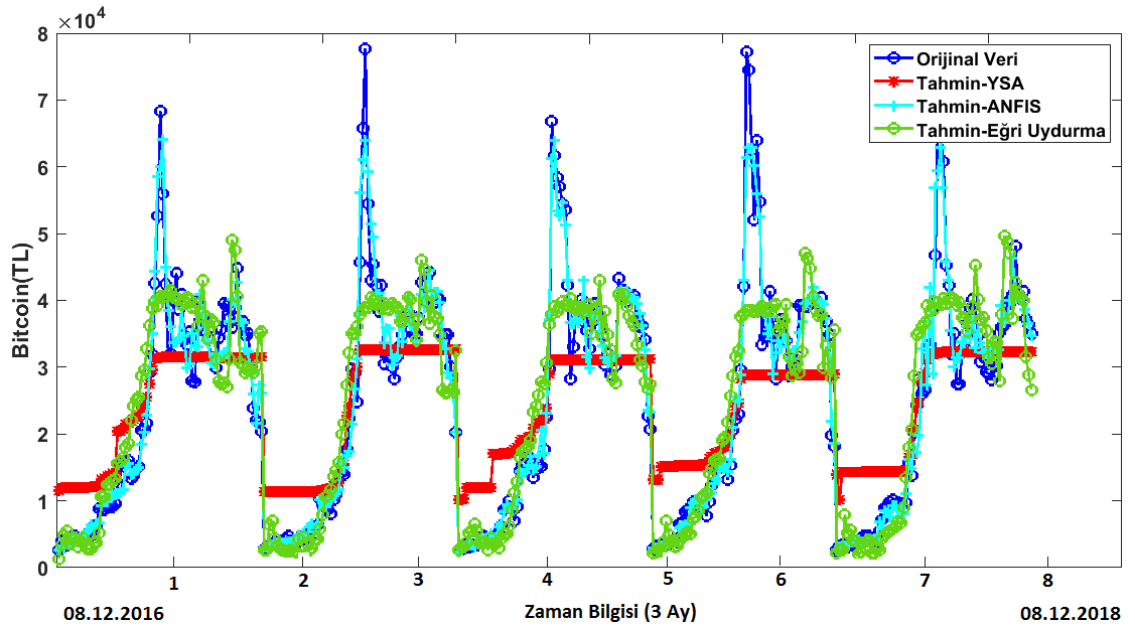
Şekil 29: 6 Saatlik Aralıklarla 1 Aylık (4_3) Veri Setine Ait Sonuçlar



Şekil 30: 3 Günlük 1 Yıllık (5_3) Veri Setine Ait Sonuçlar



Şekil 31: 2 Günlük Aralıklarla 2 Yıllık (6_4) Veri Setine Ait Sonuçlar



Şekil 28-31 incelendiğinde Tablo 17’de verilen düşük hata değerlerine ait grafikler yer almakta olup tahmin eğrilerinin orijinal eğriyi takip edebildiği görülmektedir.

4.2.2. YSA, ANFIS ve Eğri Uydurma Yöntemlerinin En İyi Parametrelerle Zaman Serisi Kullanılarak Uygulanması

BTC fiyatları zamana bağlı değişim gösterebilmekte olup bu çalışmada çapraz doğrulamanın yanında veri setleri zaman serileri gibi düşünülüp yapay zekâ yöntemleri ile de test edilmiştir. Bu çalışmada tarih ve Dolar/TL kuru ile BTC/TL fiyat tahmini olabilecek en doğru şekilde yapılmak istenildiğinden öznitelik olarak dolar kuru son değerine sabitlenmiştir. Yani geleceğe yönelik tahmin yapılırken dolar kurunun gelecekteki değişimi bilinemeyeceğinden son kur bilgisi kullanılmıştır.

Zaman serisi problemini oluşturmak için 24 adet veri seti bu sefer, sıralı bir şekilde ilk %90 ve %95'lik kısımları eğitim için, geriye kalan son %10 ve %5 lik kısımları ise test için ayrılmıştır. Normalizasyonun zaman serisine etkisinin olup olmadığı da incelenmiş olup [0 1] aralığında minmax normalizasyonu kullanılmıştır. Bu kısımda toplamda 4 farklı test gerçekleştirilmiştir. Yöntem olarak daha önce en uygun parametreleri ile belirlenen 1000 iterasyon-20 Nöron-20 Nöronlu YSA, X-Kural Sayısı 10 iterasyonlu ANFIS ve 5. dereceden polinomsal eğri uydurma kullanılmıştır. Toplamda 5 defa çalıştırılan yöntemlere ait test seti ortalama hata sonuçlarından Tablo 19'da %90 standart eğitim setine ait sonuçlar yer alırken, %90 normalize edilmiş eğitim setine ait sonuçlar Tablo 20'de, %95 standart eğitim setine ait sonuçlar Tablo 21'de ve son olarak %95 normalize eğitim setine ait sonuçlar Tablo 22'de verilmiştir. Test verileri için standart sapma değerleri ortalama hata ile birlikte verilmiştir.

Tablo 19: Zaman Serisi %90 Standart Veri Seti Sonuçları

Veri Seti	Fiyat Aralığı (BTC/TL)	Fiyat Farkı (BTC/TL)	Ortalama Hata (RMSE)	Hata Oranı (Yüzde)
1_1	18700-17755	945	409.20±4.85	43.3013
1_2	18700-17755	945	239.19±5.55	25.3108
1_3	18700-17755	945	379.45±24.94	40.1535
1_4	18700-17755	945	1629.43±120.91	172.4262
2_1	22140-17601	4539	1830.24±223.13	40.3226
2_2	22150-17601	4549	1728.11±275.91	37.9887
2_3	22150-17500	4650	2913.41±371.05	62.6540
2_4	22150-17500	4650	2827.89±733.67	60.8148
3_1	23270-18100	5170	1359.07±138.39	26.2876

3_2	23400-18100	5300	1258.59±192.94	23.7470
3_3	23400-17500	5900	2931.55±123.92	49.6872
3_4	23400-17500	5900	3198.16±77.57	54.2061
4_1	35128-18100	17028	1891.26±251.50	11.1068
4_2	35128-18100	17028	2030.11±145.57	11.9222
4_3	35128-18100	17028	3661.34±400.35	21.5019
4_4	35128-17601	17527	3292.62±1329.88	18.7860
5_1	77850-21940	55910	26698.98±604.08	47.7535
5_2	77850-19550	58300	13069.35±710.66	22.4174
5_3	77850-19550	58300	8883.79±220.44	15.2381
5_4	77850-18100	59750	33990.66±2377.85	56.8881
6_1	58350-2616	55734	313872.28±446.52	563.1612
6_2	74509-2616	71893	29012.99±667.64	40.3558
6_3	74509-2616	71893	23490.44±848.64	32.6742
6_4	77662-2616	75046	20273.91±673.67	27.0153

Tablo 19 incelendiğinde Denklem (19)'a göre hesaplanan hata oranı yüzdesi olarak %10'nun altında değere sahip yönteme rastlanılmamıştır.

Tablo 20: Zaman Serisi %90 Normalize Edilmiş Veri Seti Sonuçları

Veri Seti	Fiyat Aralığı (BTC/TL)	Fiyat Farkı (BTC/TL)	Ortalama Hata (RMSE)	Hata Oranı (Yüzde)
1_1	18700-17755	945	860.31±181.90	91.0385
1_2	18700-17755	945	605.59±301.78	64.0841
1_3	18700-17755	945	533.27±14.43	56.4312
1_4	18700-17755	945	1662.26±137.77	175.9009
2_1	22140-17601	4539	1787.04±327.86	39.3708
2_2	22150-17601	4549	1886.58±241.20	41.4724
2_3	22150-17500	4650	2565.49±750.86	55.1718
2_4	22150-17500	4650	2235.24±745.82	48.0696
3_1	23270-18100	5170	8608.22±328.54	166.5034
3_2	23400-18100	5300	2806.35±311.71	52.9500
3_3	23400-17500	5900	2725.54±214.21	46.1957
3_4	23400-17500	5900	1673.80±440.98	28.3695

4_1	35128-18100	17028	3740.15±325.95	21.9647
4_2	35128-18100	17028	1944.11±1444.47	11.4172
4_3	35128-18100	17028	1552.27±1378.60	9.1160
4_4	35128-17601	17527	1777.29±1776.30	10.1403
5_1	77850-21940	55910	40332.97±1393.28	72.1391
5_2	77850-19550	58300	22214.72±263.15	38.1041
5_3	77850-19550	58300	15711.04±973.00	26.9486
5_4	77850-18100	59750	20318.11±1415.28	34.0052
6_1	58350-2616	55734	293064.28±666.21	525.8268
6_2	74509-2616	71893	32723.51±572.48	45.5170
6_3	74509-2616	71893	32336.61±1244.97	44.9788
6_4	77662-2616	75046	37061.97±3036.96	49.3857

Aynı şekilde Tablo 20 incelendiğinde de sonuçlar açısından çokta iyi sayılamayacak değerler elde edilmiştir.

Tablo 21: Zaman Serisi %95 Standart Veri Seti Sonuçları

Veri Seti	Fiyat Aralığı (BTC/TL)	Fiyat Farkı (BTC/TL)	Ortalama Hata (RMSE)	Hata Oranı (Yüzde)
1_1	18700-17755	945	733.25±13.28	77.5931
1_2	18700-17755	945	481.71±19.67	50.9743
1_3	18700-17755	945	351.40±5.92	37.1847
1_4	18700-17755	945	280.86±24.74	29.7205
2_1	22140-17601	4539	605.04±222.36	13.3299
2_2	22150-17601	4549	574.52±163.57	12.6296
2_3	22150-17500	4650	1159.65±345.01	24.9387
2_4	22150-17500	4650	1532.86±538.05	32.9648
3_1	23270-18100	5170	563.77±428.34	10.9046
3_2	23400-18100	5300	895.85±420.35	16.9028
3_3	23400-17500	5900	1186.51±553.11	20.1103
3_4	23400-17500	5900	1359.89±598.02	23.0491
4_1	35128-18100	17028	1847.52±291.48	10.8499
4_2	35128-18100	17028	2238.11±328.49	13.1437

4_3	35128-18100	17028	1215.65±349.95	7.1391
4_4	35128-17601	17527	2549.88±657.13	14.5483
5_1	77850-21940	55910	19761.91±447.73	35.3459
5_2	77850-19550	58300	13323.74±416.42	22.8538
5_3	77850-19550	58300	9862.31±1520.21	16.9165
5_4	77850-18100	59750	16017.61±1509.98	26.8077
6_1	58350-2616	55734	19266.65±961.88	34.5689
6_2	74509-2616	71893	21805.40±1137.14	30.3303
6_3	74509-2616	71893	16495.88±1264.05	22.9450
6_4	77662-2616	75046	10106.24±2117.16	13.4667

Tablo 21’de elde edilen sonuçlar incelendiğinde %10 hata oranına yakın çok daha fazla veri setinin yer aldığı görülebilmektedir.

Tablo 22: Zaman Serisi %95 Normalize Veri Seti Sonuçları

Veri Seti	Fiyat Aralığı (BTC/TL)	Fiyat Farkı (BTC/TL)	Ortalama Hata (RMSE)	Hata Oranı (Yüzde)
1_1	18700-17755	945	893.49±149.10	94.5493
1_2	18700-17755	945	555.06±49.54	58.7368
1_3	18700-17755	945	451.95±55.31	47.8252
1_4	18700-17755	945	402.50±109.35	42.5930
2_1	22140-17601	4539	774.91±353.64	17.0722
2_2	22150-17601	4549	624.45±170.17	13.7272
2_3	22150-17500	4650	906.37±438.36	19.4919
2_4	22150-17500	4650	1579.19±389.60	33.9610
3_1	23270-18100	5170	1016.90±538.48	19.6693
3_2	23400-18100	5300	999.00±367.13	18.8491
3_3	23400-17500	5900	1119.12±447.66	18.9682
3_4	23400-17500	5900	850.11±292.83	14.4086
4_1	35128-18100	17028	2086.88±641.87	12.2556
4_2	35128-18100	17028	2332.91±787.65	13.7004
4_3	35128-18100	17028	2155.80±825.26	12.6603
4_4	35128-17601	17527	1579.43±518.22	9.0114
5_1	77850-21940	55910	24375.25±763.67	43.5973
5_2	77850-19550	58300	13928.07±920.92	23.8903
5_3	77850-19550	58300	11618.61±920.92	19.9290

5_4	77850-18100	59750	10133.63±1848.01	16.9600
6_1	58350-2616	55734	58192.58±2876.75	104.4113
6_2	74509-2616	71893	23242.65±1078.70	32.3295
6_3	74509-2616	71893	18560.90±1700.00	25.8174
6_4	77662-2616	75046	17016.06±3152.09	22.6742

En uygun veri setini bulmak için çapraz doğrulamanın yanında zaman serisi şeklinde de test yapıldığından son sonucu belirlemede bu iki farklı deneysel çalışmaya ait sonuçların ortalaması kabul edilmiştir. 5-K çapraz doğrulama ve zaman analizi hata oranı arasındaki fark iki hatanın ortalaması olarak kabul edilip ortalama hata farkı ismiyle belirtilmiştir. Daha önce Tablo 16’da verilen standart 5-K çapraz doğrulama yönteminin sonuçları ile zaman serisi sonuçlarının karşılaştırılması Tablo 23’te verilmiş olup, elde edilen sonuçlara göre %95 oranında standart (normalize edilmemiş) eğitim verilerinin yapay zekâ yöntemleri ile çalıştırılmasında diğer oranlara göre en iyi sonucu verdiği görülmüştür.

Tablo 23: %95 Standart Zaman Serisi Verileri ve 5-Kat Çapraz Doğrulama Sonuçlarının Karşılaştırılması

Veri Seti	Zaman Serisi Analizi Hata Oranı (%)	5-Kat Çapraz Doğrulama Hata Oranı (%)	Ortalama Hata Farkı (%)
1_1	77.59	42.00	59.80
1_2	50.97	27.84	39.41
1_3	37.18	23.77	30.48
1_4	29.72	32.34	31.03
2_1	13.33	9.34	11.34
2_2	12.63	8.17	10.40
2_3	24.94	19.08	22.01
2_4	32.96	22.97	27.97
3_1	10.90	19.66	15.28
3_2	16.90	15.60	16.25
3_3	20.11	12.90	16.51
3_4	23.05	15.09	19.07
4_1	10.85	8.57	9.71
4_2	13.14	7.47	10.31
4_3	7.14	6.56	6.85
4_4	14.55	9.77	12.16

5_1	35.35	16.76	26.05
5_2	22.85	11.00	16.93
5_3	16.92	8.18	12.55
5_4	26.81	13.14	19.98
6_1	34.57	24.42	29.49
6_2	30.33	14.72	22.52
6_3	22.95	11.47	17.21
6_4	13.47	10.90	12.18

Çapraz doğrulama testlerinde üç farklı yapay zekâ yönteminin neticesinde veri setleri arasından Tablo 17’deki gibi sırası ile 2_2, 4_3, 5_3 ve 6_4 veri setleri en iyi veri seti olarak belirlenmişti. Daha önce 1.veri setinin tamamında istenilen hata oranı yakalanmazken, zaman serisi analizi ile 1_3 veri seti için tolere edilebilir bir hata oranı bulunmuştur. Ancak çalışmanın bir web sayfası sitesi kurularak çevrimiçi (online) veriler ile çalışması ve anlık tahminler yapması aşamasında sunucu ile ilgili teknik sebeplerden dolayı 1_2 (yarım saatlik günlük veriler) veri seti kullanılmış olup Tablo 23’e göre sonuçlar birbirlerine yakındır. Zaman serisi hâline getirilen veri setleri içerisinde karşılaştırmalı sonuçlar da baz alınarak oluşturulan yeni en iyi veri seti grupları Tablo 24’te verilmiştir.

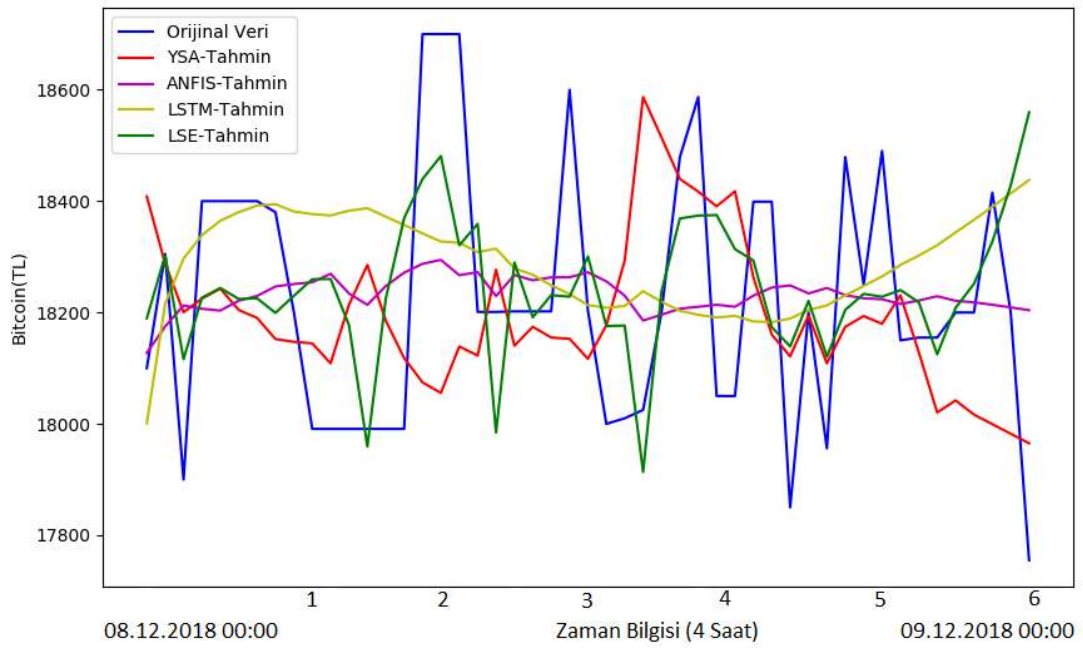
Tablo 24: Yeniden Belirlenen En İyi Veri Seti Grupları ve Ortalama Hata Oranları

Veri Seti	Ortalama Hata Oranı
1_2	39.41
2_2	10.40
4_3	6.85
5_3	12.55
6_4	12.18

Çalışmanın bu bölümünde en iyi yöntem ve parametrelerin belirlenmesi ve belirlenen yöntem ve parametreler uygulandığında en başarılı sonuçları veren veri seti grupları belirlenmiştir. Matlab R2018b programı kullanılarak elde edilen en iyi yöntem-parametreler ve veri seti grupları PyCharm Community Edition 2019.1 programı kullanılarak Python ile de çalıştırılmıştır. Python programlama dili kullanılarak yapılan test etme işlemi daha önce belirlenen zaman serisi şeklinde %95 oranında standart eğitim verisi kullanılarak yapılmıştır. Şekil 32-41 arasında Tablo 24’te nihai olarak

belirlenen veri setleri için daha önce belirlenen parametrelerde YSA, ANFIS ve eğri uydurma yöntemlerinin sonuçları yer almaktadır. Bunu yanında günümüzde oldukça popüler olup iyi sonuçlar veren derin öğrenme metodu da çalışmanın bu aşamasından sonra Python’a dahil edilmiştir. Derin öğrenme algoritması olarak zaman serilerinde iyi sonuçlar veren LSTM kullanılmıştır. Şekillerde, tarih etiketinin solunda verilen tarih, veri setinin ilk verisini yani başlangıç tarihini, sağındaki tarih ise veri setinin son verisini yani bitiş tarihini belirtmektedir.

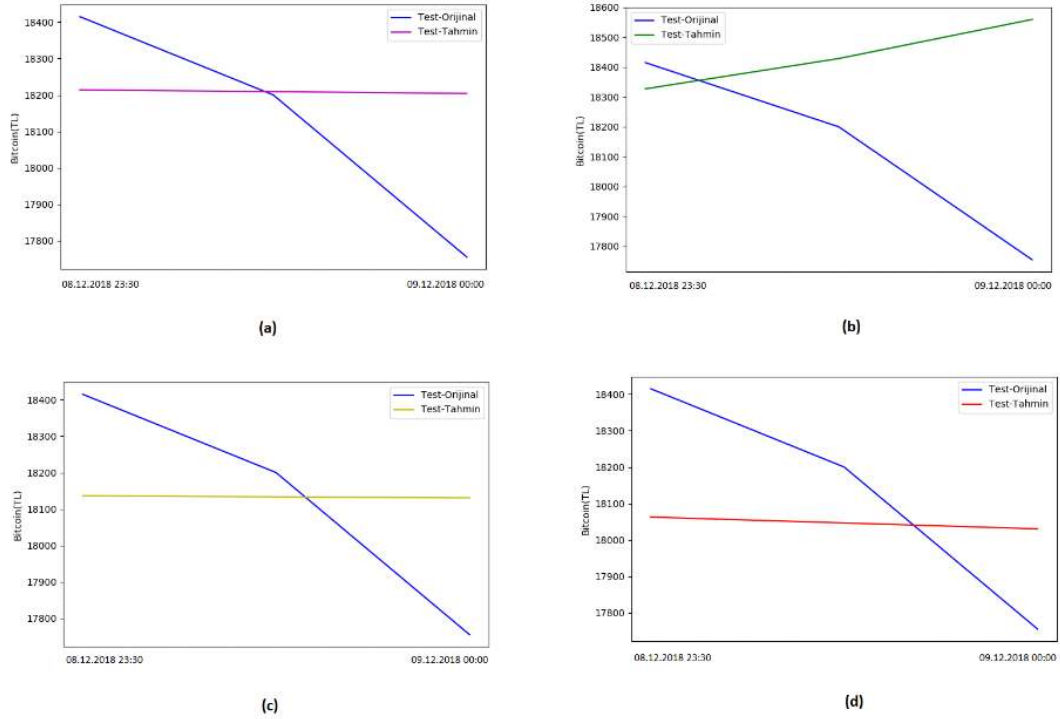
Şekil 32: 1_2 Veri Setine Uygulanan Yapay Zekâ Yöntemlerine Ait Tahminlerin Bir Arada Gösterilmesi



Şekil 32’de veri seti 1_2’ye ait olan verilerin eğitilip-test edilerek bir tahmin modeli oluşturmak için kullanılan 4 farklı yapay zekâ yönteminin doğru sonuca ne kadar yaklaştıkları ve birbirileri ile kıyaslamaları görülmektedir.

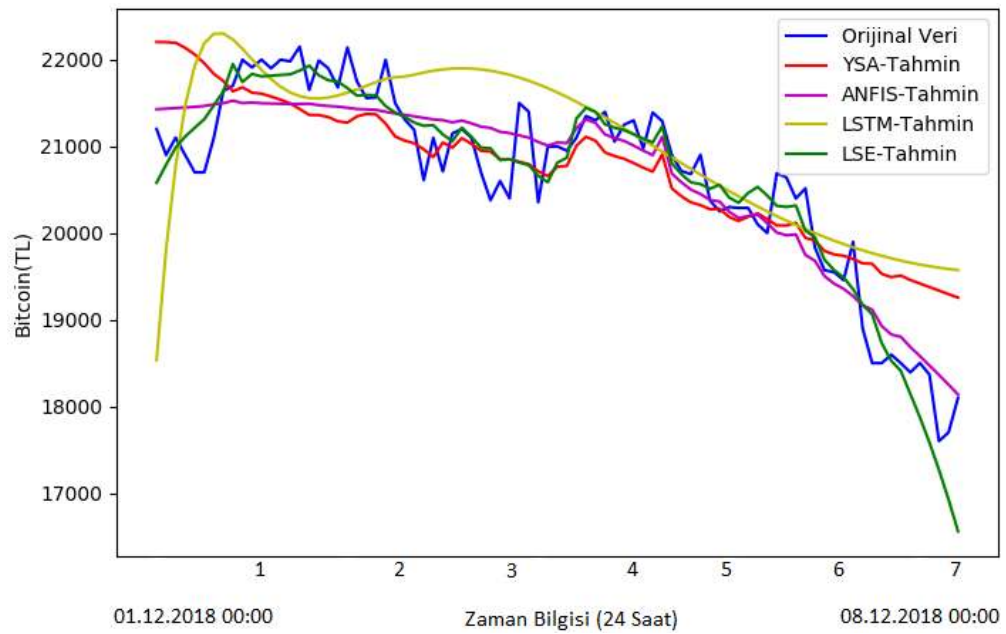
Test verilerine ait şekillerde yatay eksenle gösterilen “zaman bilgisi” ile zaman aralıklarının çarpımı sonucu veri seti grubuna ait tarih özniteliği ortaya çıkmaktadır. Örneğin, yarım saatlik 1 günlük verileri içeren 1_2 veri seti grubuna ait zaman bilgisi 4 saat, zaman aralıkları toplamda 6’dır. Yani, $4 \times 6 = 24$ saat ve 1 günlük verilere karşılık gelmektedir. Aynı hesaplama diğer veri seti grupları için de geçerlidir.

Şekil 33: Zaman Serisi Kullanılarak Yapay Zekâ Yöntemlerinin Veri Seti 1_2'ye Uygulanması, (a) ANFIS, (b) Eğri uydurma, (c) LSTM, (d) YSA

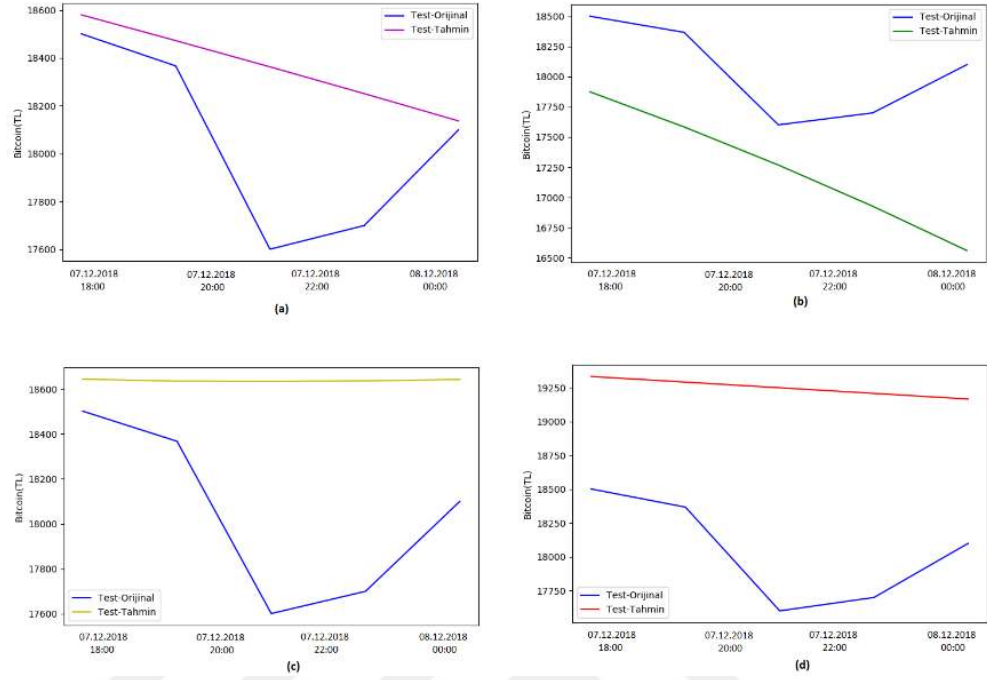


Şekilde 33'de görüldüğü gibi toplamda 4 farklı yapay zekâ yöntemi veri seti 1_2 üzerinde çalıştırılmıştır. Şekilde gösterilen veriler sadece test verilerine aittir.

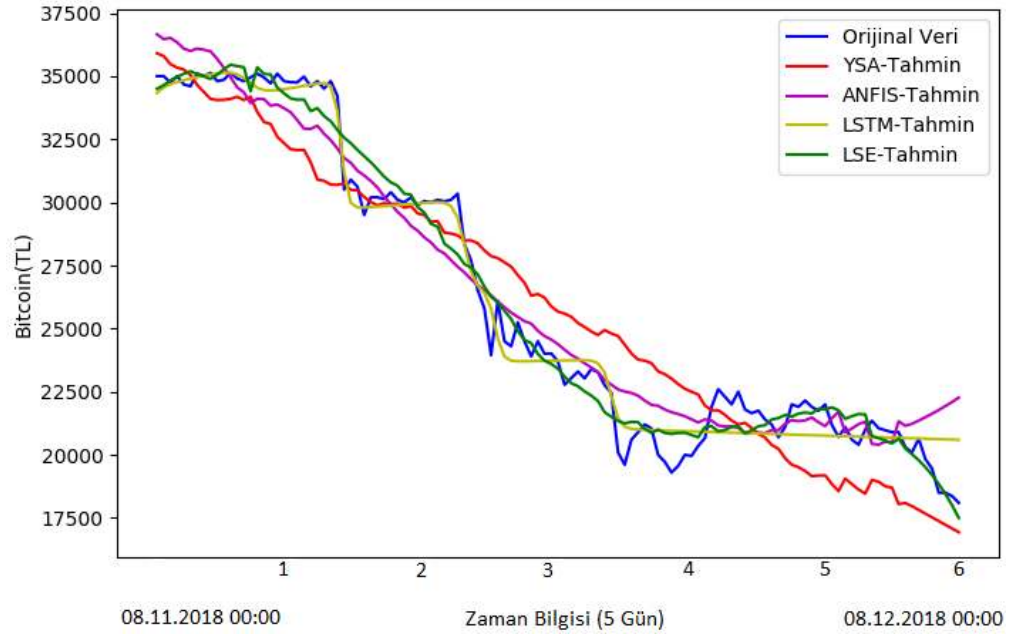
Şekil 34: 2_2 Veri Setine Uygulanan Yapay Zekâ Yöntemlerine Ait Tahminlerin Bir Arada Gösterilmesi



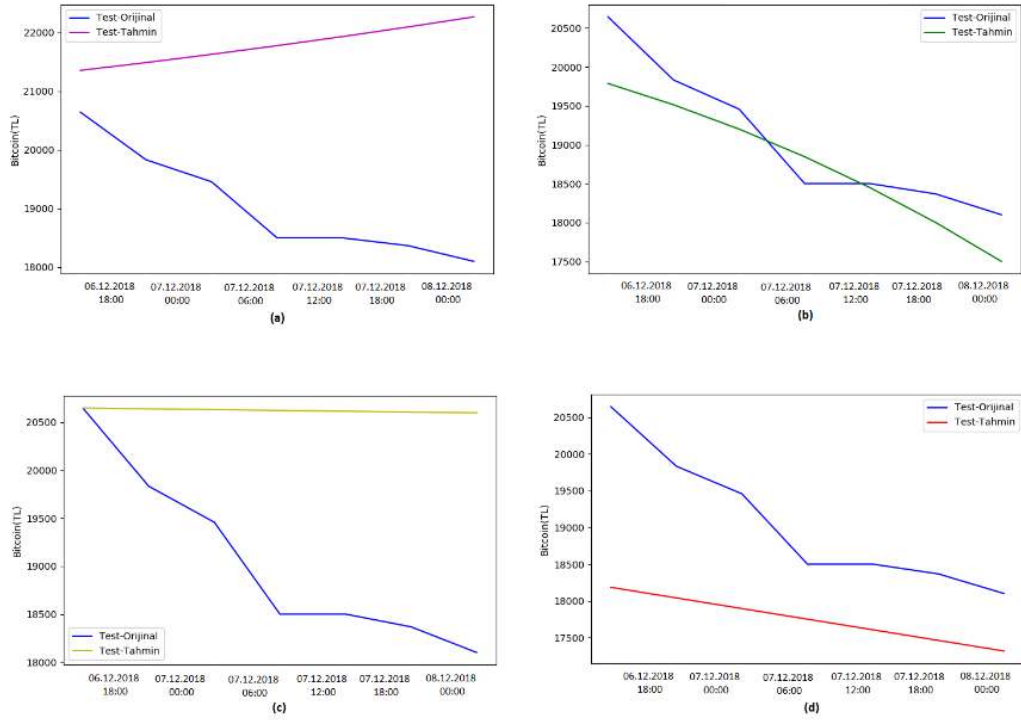
Şekil 35: Zaman Serisi Kullanılarak Yapay Zekâ Yöntemlerinin Veri Seti 2_2'ye Uygulanması, (a) ANFIS, (b) Eğri Uydurma, (c) LSTM, (d) YSA



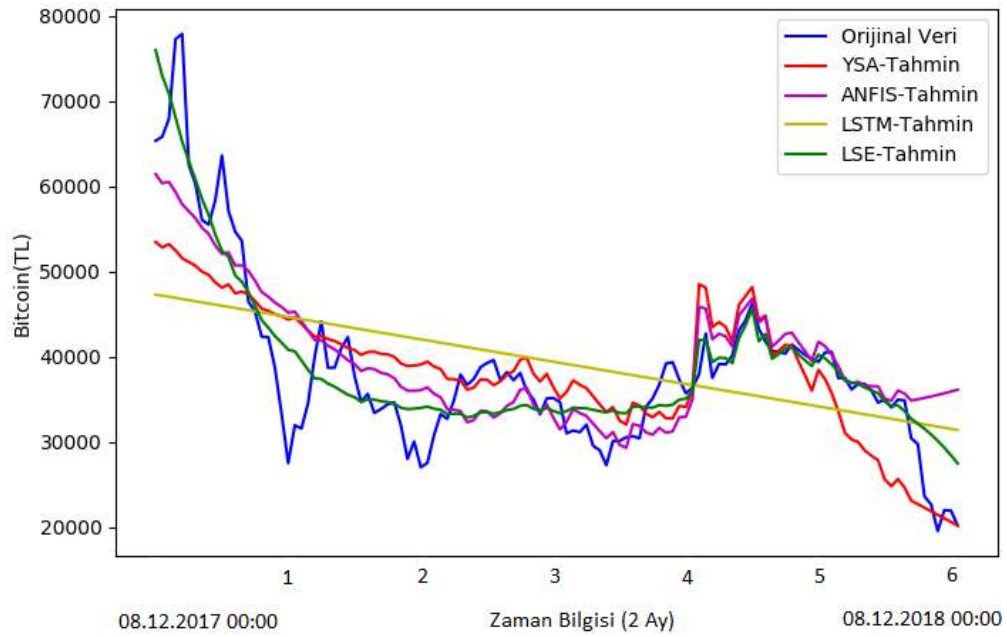
Şekil 36: 4_3 Veri Setine Uygulanan Yapay Zekâ Yöntemlerine Ait Tahminlerin Bir Arada Gösterilmesi



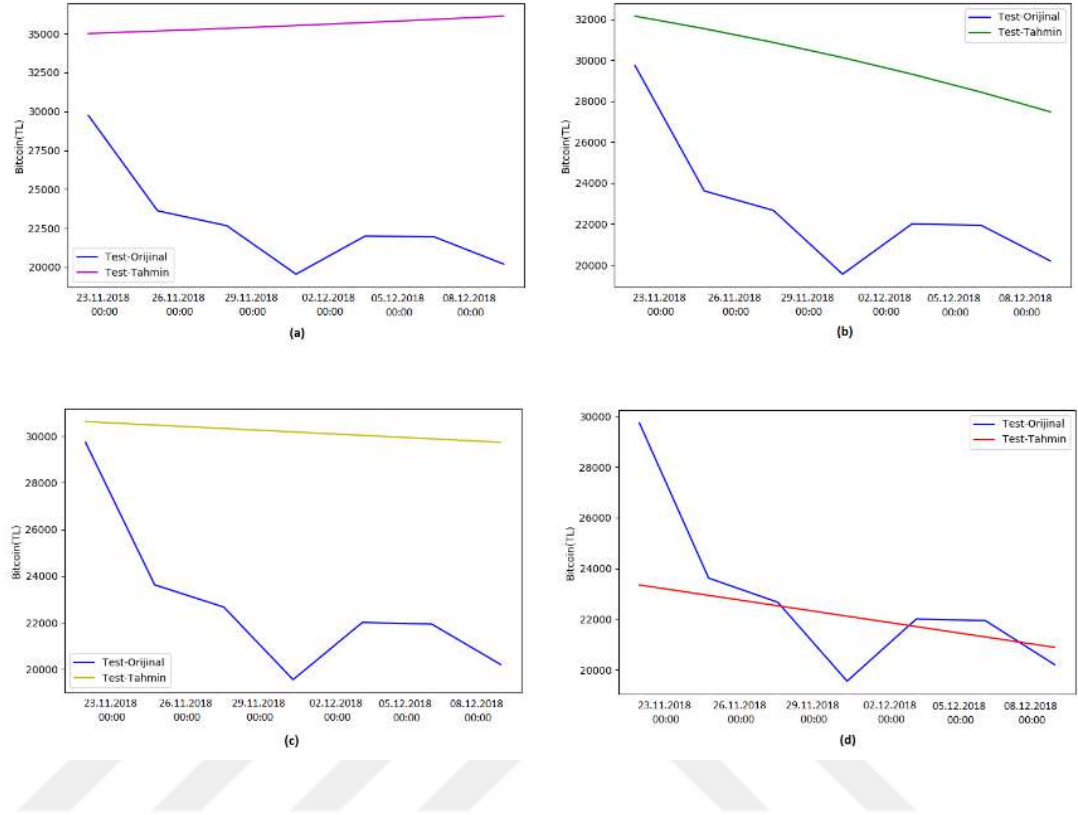
Şekil 37: Zaman Serisi Kullanılarak Yapay Zekâ Yöntemlerinin Veri Seti 4_3'e Uygulanması, (a) ANFIS, (b) Eğri Uydurma, (c) LSTM, (d) YSA



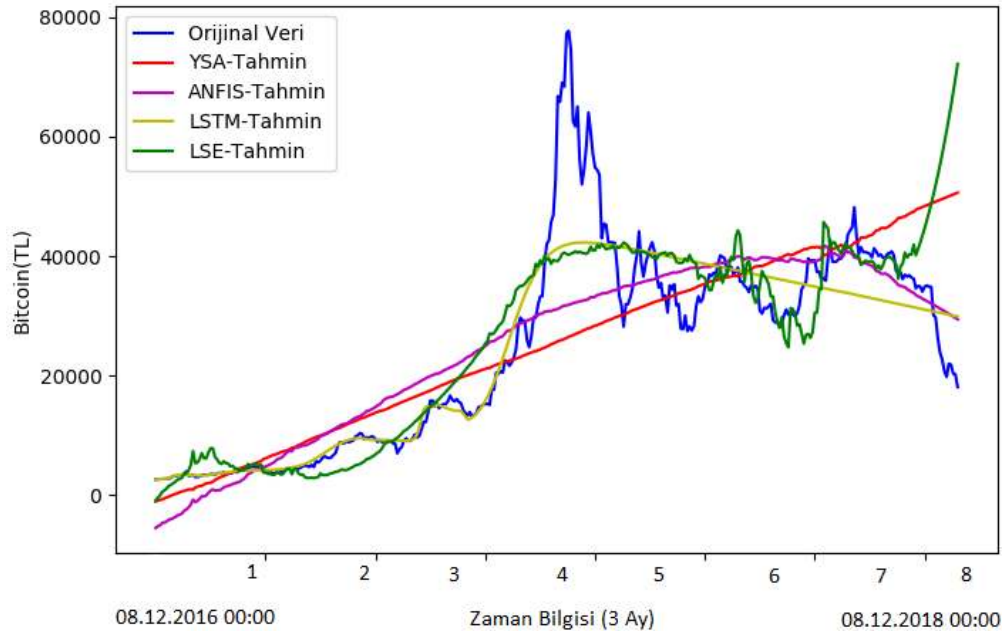
Şekil 38: 5_3 Veri Setine Uygulanan Yapay Zekâ Yöntemlerine Ait Tahminlerin Bir Arada Gösterilmesi



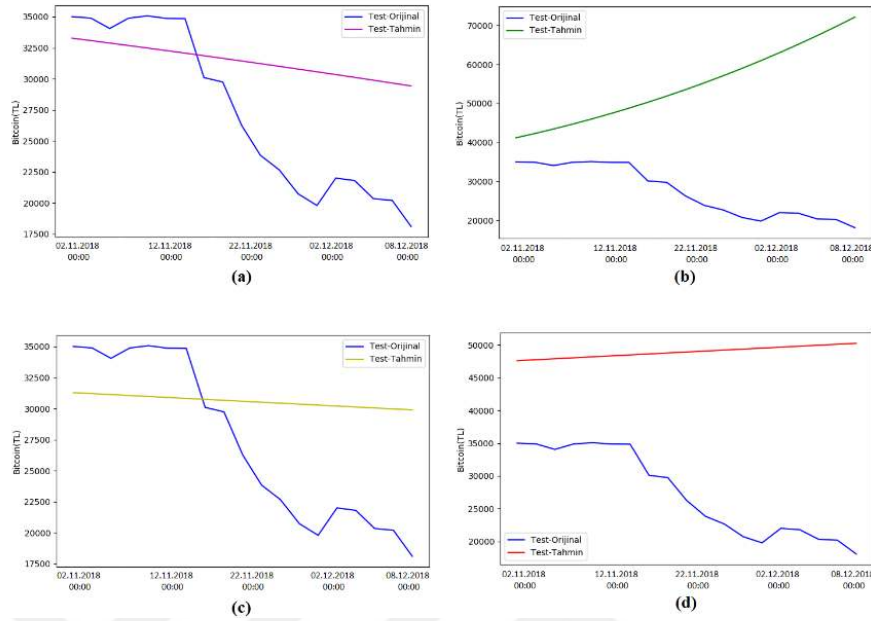
Şekil 39: Zaman Serisi Kullanılarak Yapay Zekâ Yöntemlerinin Veri Seti 5_3'e Uygulanması, (a) ANFIS, (b) Eğri Uydurma, (c) LSTM, (d) YSA



Şekil 40: 6_4 Veri Setine Uygulanan Yapay Zekâ Yöntemlerine Ait Tahminlerin Bir Arada Gösterilmesi



Şekil 41: Zaman Serisi Kullanılarak Yapay Zekâ Yöntemlerinin Veri Seti 6_4'e Uygulanması, (a) ANFIS, (b) Eğri Uydurma, (c) LSTM, (d) YSA



BTC/TL fiyat tahmini için oluşturulmuş farklı zaman aralıklarına sahip veri seti gruplarına uygulanan 4 farklı yapay zekâ yöntemine ait sonuçlar ve kıyaslamalar şekil 32-41 ile ifade edilmiştir. Tablo 25'te yapay zekâ yöntemlerine ait eğitim-test hataları ve standart sapma değerleri, belirlenen veri seti gruplarına göre kategorize edilerek verilmiştir. Python'da ANFIS yöntemi en küçük kareler yöntemine göre çalıştığı için Tablo 25'te standart sapması 0 olarak bulunan iki yapay zekâ yöntemi ANFIS ve eğri uydurmaya ait istatistikler birlikte verilmiştir. Bunun yanında yine Python çalışması için elde edilen veriler ışığında sadece ANFIS için normalize özniteliklerin kullanılmasının çok daha iyi sonuçlar verdiği görülmüştür. Diğer yöntemler için standart öznitelikler aynı veya daha iyi sonuçlar vermişlerdir. Dolayısıyla ANFIS örnekleri [0 1] aralığında minmax normalizasyonu kullanılarak çalıştırılmıştır. LSTM ve YSA yöntemlerine ait sonuçlar da sırayla Tablo 26 ve Tablo 27'de verilmiştir.

Tablo 25: Veri Seti Gruplarına Ait ANFIS ve Eğri Uydurma İstatistikleri

Veri Seti	ANFIS Eğitim Hatası (RMSE)	ANFIS Test Hatası (RMSE)	Eğri Uydurma Eğitim Hatası (RMSE)	Eğri Uydurma Test Hatası (RMSE)
1_2	217.78±0	284.14±0	192.69±0	485.53±0
2_2	400.14±0	425.15±0	295.4±0	903.85±0
4_3	1216.28±0	2970.55±0	890.79±0	465.44±0

5_3	5250.39±0	13164.36±0	4013.97±0	7527.52±0
6_4	9379.67±0	6712.14±0	7389.77±0	31502.59±0

Tablo 26: Veri Seti Gruplarına Ait LSTM İstatistikleri

Veri Seti	LSTM Ortalama Eğitim Hatası (RMSE)	LSTM Ortalama Test Hatası (RMSE)
1_2	247.47±44.76	414.56±203.35
2_2	531.45±108.98	1206.65±215.09
4_3	896.51±177.99	1951.50±215.09
5_3	6013.47±2236.77	12863.89±3538.99
6_4	6756.79±61.35	8657.30±1436.89

LSTM, özellikle zaman serilerinde sıklıkla kullanılan özel bir tekrarlayan sinir ağıdır. Bu çalışmada zaman serisi verilerinde kullanılan LSTM'nin beklenenden fazla hata skoruna sahip olmasının sebebi olarak; tam anlamıyla gerçek bir BTC/TL fiyat tahmini yapmak için öznitelik olarak kullanılan Dolar/TL verisinin sabitlenmesidir. Çünkü gerçek hayatta bir sonraki Dolar/TL kuru bilinemeyeceği için veri setindeki son Dolar/TL verisi sabitlenip zaman serisi şeklinde çalıştırılmıştır. Çalışmada LSTM ile tahmin yapılırken belli bir veriden (zaman aralığından) sonra LSTM'nin sabit, değişmeyen bir grafik görünümü sergilemesi Dolar/TL kurunun sabitlenmesine dayalı “kaos teorisi” ile ilişkilendirilmiştir.

Normalde kaos denildiğinde insanlar ilk olarak özgürlük, rassallık, anarşi gibi kavramları düşünmektedirler. Ancak bilimsel anlamda kaos bu kavramlarla pek de ilişkili değildir. Kaos için daha ziyade “düzensizliğin içindeki düzen” ifadesi kullanılmaktadır (Tosun, 2006). Kaos terimi ilk defa Henri Poincare tarafından kullanılmıştır. Poincare, güneş sisteminin kararlılığı üzerine yapmış olduğu bir çalışmada güneş sisteminin hareketini tespit etmeye dayanan denklem sisteminin çözümünün başlangıç koşullarına bağımlı olduğunu ama başlangıç koşullarının da tam olarak doğru bir şekilde bilinemeyeceği bulgusuna ulaşmıştır. Poincare, bu durum için “kaos” ifadesini kullanmıştır (Ertürk, 2012; Çıraklı vd., 2017). Poincare'e göre bir sistem, birbirine entegre ve birbiri ile ilişkili bölümlerden ve bileşenlerden meydana geliyorsa eğer, bu beklenmeyen davranışlara neden olur ve “kaos” tam olarak burada başlar (Fettahoğlu, 2009; Çıraklı vd., 2017).

Tablo 27: Veri Seti Gruplarına Ait YSA İstatistikleri

Veri Seti	YSA Ortalama Eğitim Hatası (RMSE)	YSA Ortalama Test Hatası (RMSE)
1_2	263.83±0.32	287.31±8.27
2_2	560.63±1.32	1331.92±50.81
4_3	2089.97±1.85	1624.11±157.73
5_3	7143.33±3.06	2853.34±121.13
6_4	10463.43±9.72	23499.96±481.88

Bu çalışmada LSTM, ANFIS ve YSA'nın ani fiyat değişimlerini dinamik bir şekilde iniş-çıkışlar göstererek tahmin edemeyip son tahmin ettiği değere bağlı kalıp aynı açı yönünde ilerlemesinin en büyük sebebi olarak kaos teorisi düşünülmektedir. Öyleki eldeki son güncel veriler ışığında yapılacak ilk tahmin daha sonrasındaki tahminlerin tümünü etkilemektedir. Bunun yanında Dolar/TL öznitelik verisinin sabitlenmesi de modeli sadece zaman verisini kullanması yönünde yalnız bırakmaktadır. Yine de doların TL üzerindeki dalgalanması göz önüne alındığında BTC-TL modellemesi için dolar kurunun kullanılması da bir gereklilik oluşturmaktadır. Çünkü Türkiye borsaları işlem hacmi olarak yetersiz olup BTC'nin değeri dolara bağımlı yurtdışı borsalarında oluşmaktadır.

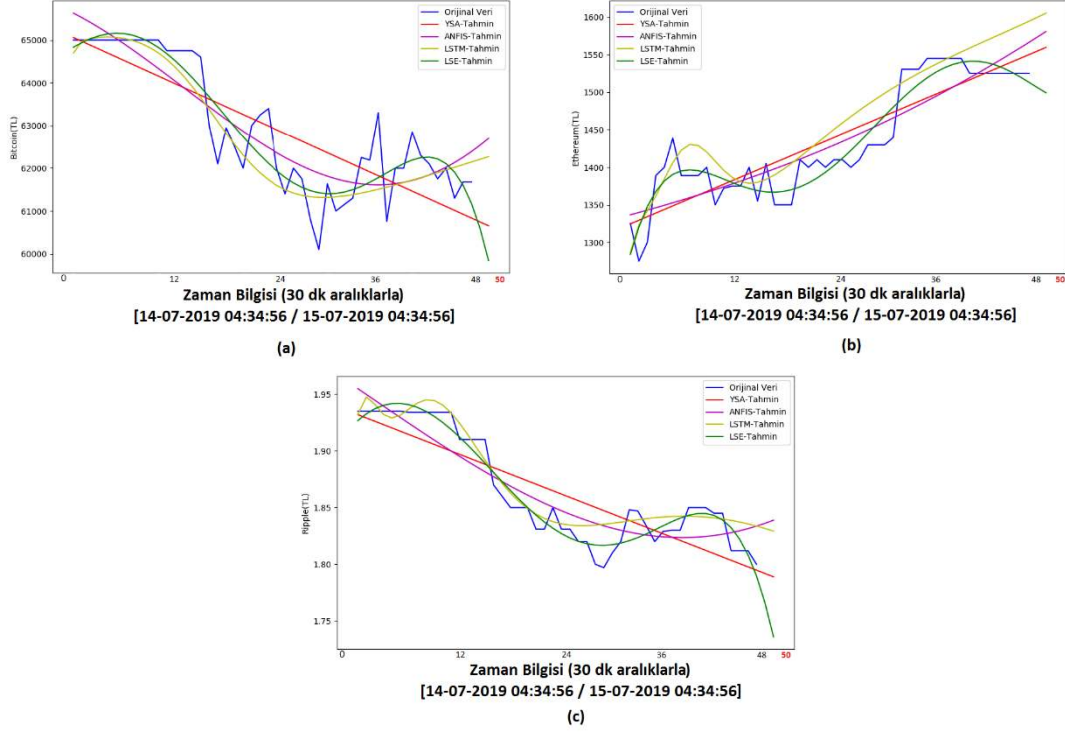
Tüm bunların yanında hem klasik paraların, hem de kripto paraların değerlerinin belirlenmesinin altında çok fazla bileşen ve etken yatmaktadır.

4.2.3. Gerçek Zamanlı Uygulama ve Sonuçları

Elde edilen tüm sonuçlar ışığında son model oluşturulmuş olup, bu model en iyi parametreleri kullanan YSA, ANFIS, eğri uydurma ve LSTM yöntemleri olarak öne çıkmaktadır. ANFIS dışındaki yöntemler standart özniteliklerle çalışırken, ANFIS için ise [0 1] aralığında minimax normalizasyonu kullanılmıştır. Gerçek zamanlı olarak BTC, Ethereum ve Ripple'a ait elde edilen kripto para değerlerinin yanında Dolar/TL değeri de elde edilebilmektedir. Son model için elde edilen tüm veriler eğitim seti olarak kabul edilip tamamı eğitilmekte olup, hiçbir şekilde bilinmeyen dolar kur değeri ve zaman bilgisi kullanılarak gerçek zamanlı tahminleme yapılmaktadır. Zaman serisi sonuçları göz önüne alındığında veri seti boyutunun %5'lik kısmı kadar ileri tarih ve sabit kur bilgisi kullanılarak tahminleme yapılabilmektedir. Şekil 42 ve Şekil 43'te her

üç kripto para çeşidi için de 15.07.2019 tarihi 04:30 saatinde çalıştırılan ve farklı zaman aralıklarına ait gerçek zamanlı bir tahminleme görülmektedir.

Şekil 42: 1 Günlük Verilere Uygulanan Yapay Zekâ Yöntemlerine Ait Anlık Tahminlerin Bir Arada Gösterilmesi, (a) BTC, (b) ETH, (c) XRP

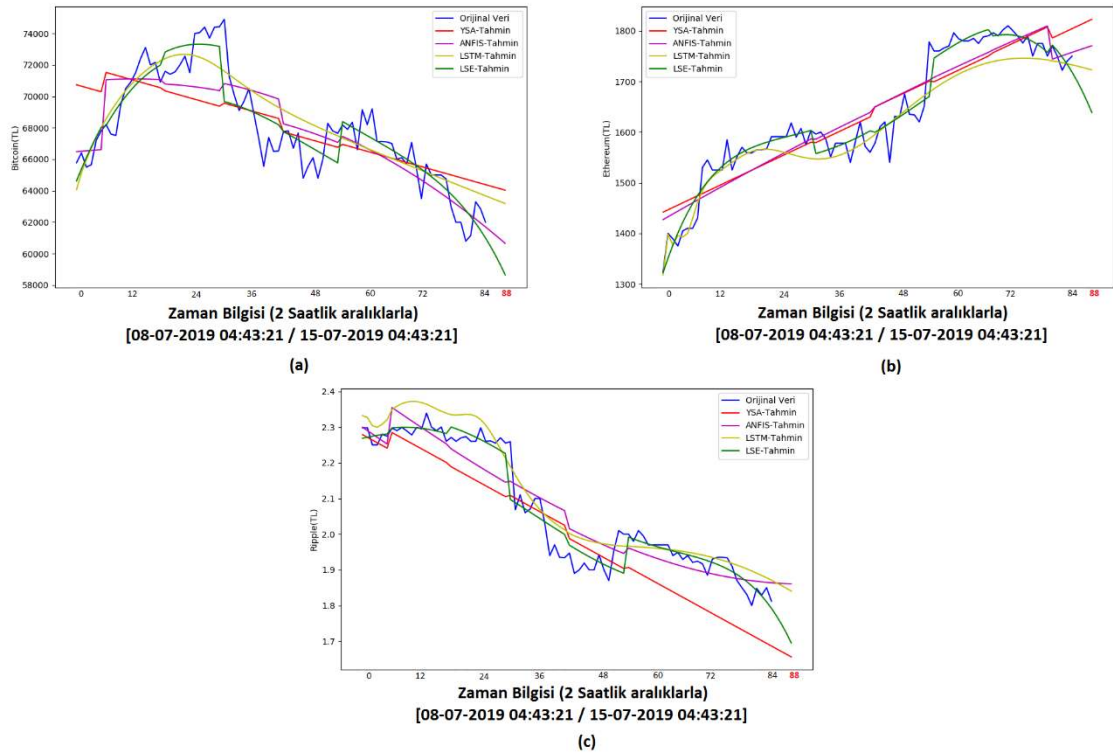


Şekil 42’de gösterilen anlık tahmin grafiklerinde 3 adet kripto para birimi (BTC, ETH, XRP) için 15.07.2019 05:00 ve 15.07.2019 05:30 zaman aralığını kapsayacak şekilde 30 dakikalık aralıklarla toplam 1 saatlik (2 adet) tahmini TL cinsinden yapılmıştır. %5 Test verisinin ileri tahmin için kullanıldığı bu çalışmada, yarım saatlik 1 günlük verilerin toplam sayısı 48’dir. Yani, 48 (yarım saatlik 1 günlük) verinin tamamı eğitim verisi olarak kabul edilmiş ve 48’in %5’i test verisi olarak hesaplanmıştır. Sonuç olarak 2 adet test verisi elde edilmiştir. Yarım saatlik veriler kullanıldığı için 2 adet yarım saatlik veri ile 1 saat sonrası tahmin edilmiştir. Şekil 42’de son eğitim verisi olan 48’den sonra gelen ve koyu kırmızı olarak verilen 50 sayısı aslında tahmin edilen test verisini belirtmektedir. Tahmin edilen fiyat ve gerçek fiyat değerlerine ait son veri olan ve 15.07.2019 05:30 tarihini kapsayan veriler Tablo 28’de gösterilmiştir. Tablo 28’de yer alan Güncel Fiyat sütunu, ilgili kripto paraların 15.07.2019 04:30 tarihinde o an ki fiyat verisine karşılık gelmektedir.

Tablo 28: 1 Günlük Veriye Uygulanan Yapay Zekâ Yöntemlerine Ait Anlık Tahminlerin ve Gerçek Fiyat Verilerinin Karşılaştırılması

Kripto Para	Güncel Fiyat (TL)	ANFIS Tahmin (TL)	Eğri Uydurma Tahmin (TL)	LSTM Tahmin (TL)	YSA Tahmin (TL)	Gerçekleşen Fiyat (TL)	Zaman Bilgisi (Saat)
BTC	60000	62695	59845	62269	60654	61750	05:30
ETH	1.325	1580	1500	1605	1560	1390	05:30
XRP	1.8001	1.8388	1.7362	1.8293	1.7889	1.8200	05:30

Şekil 43: 1 Haftalık Verilere Uygulanan Yapay Zekâ Yöntemlerine Ait Anlık Tahminlerin Bir Arada Gösterilmesi, (a) BTC, (b) ETH, (c) XRP



Şekil 43'te 3 kripto para birimine (BTC, ETH, XRP) ait 2 haftalık 2 saatlik veriler ile 15.07.2019 04:30 - 15.07.2019 12:30 zaman aralığını kapsayacak şekilde 2 saatte bir olmak üzere toplam 8 saatlik fiyat tahmini yapılmıştır. %5 Test verisinin ileri tahmin için kullanıldığı bu çalışmada, iki saatlik 1 haftalık verilerin toplam sayısı 84'tür. Yani, 84 (iki saatlik 1 haftalık) verinin tamamı eğitim verisi olarak kabul edilmiş ve 84'ün %5'i test verisi olarak hesaplanmıştır. Sonuç olarak 4 adet test verisi elde edilmiştir. İki saatlik veriler kullanıldığı için 4 adet iki saatlik veri ile 8 saat sonrası tahmin edilmiştir. Şekil 43'te son eğitim verisi olan 84'ten sonra gelen ve koyu kırmızı

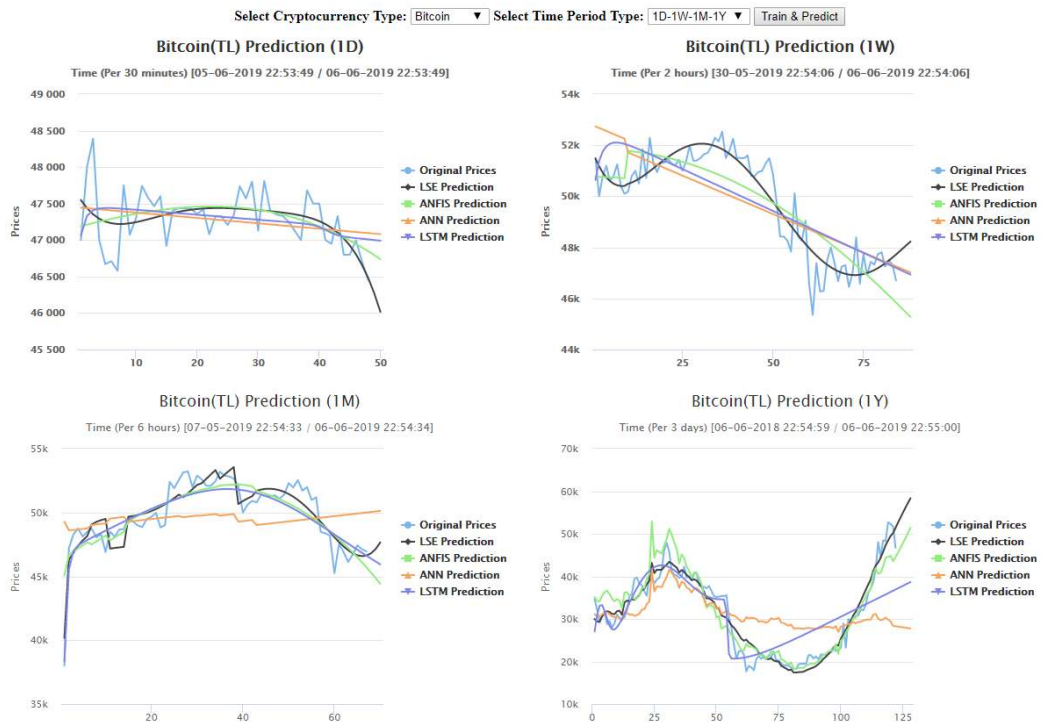
olarak verilen 88 sayısı aslında tahmin edilen test verisini belirtmektedir. Tahmin edilen fiyatlar ile gerçek fiyat değerlerine ait son veri olan ve 15.07.2019 12:30 tarihini kapsayan veriler ait veriler Tablo 29’da verilmiştir. Tablo 29’da yer alan Güncel Fiyat sütunu, ilgili kripto paraların 15.07.2019 04:30 tarihinde o an ki fiyat verisine karşılık gelmektedir.

Tablo 29: 1 Aylık Verilere Uygulanan Yapay Zekâ Yöntemlerine Ait Anlık Tahminlerin ve Gerçek Fiyat Verilerinin Karşılaştırılması

Kripto Para	Güncel Fiyat (TL)	ANFIS Tahmin (TL)	Eğri Uydurma Tahmin (TL)	LSTM Tahmin (TL)	YSA Tahmin (TL)	Gerçekleşen Fiyat (TL)	Zaman Bilgisi (Saat)
BTC	60000	60656	58700	63183	64034	60100	12:30
ETH	1.325	1770	1639	1722	1822	1287	12:30
XRP	1.8001	1.8607	1.6956	1.8407	1.6563	1.8000	12:30

Gerçek zamanlı çalışabilen modelin son olarak bir web ortamında çalıştırılması gerçekleştirilmiştir. Django kütüphanesi ve Google’ye ait bulut desteği kullanılarak www.futureofbtc.com adresi üzerinde oluşturulan web sitesine ait 06.06.2019-22:54 tarihinde gerçekleştirilen eğitim ve tahmin sonuçları Şekil 44’te görülmektedir.

Şekil 44: Web Ortamında Çalıştırılan BTC/TL Fiyat Tahmini Model Sonuçları



06.06.2019-22:54 tarihinde BTC=46750 iken gerçekleştirilen tahmin sonrası gereken süre kadar beklenilip BTC'nin gerçekte aldığı değerler gözlenmiştir. Gözlemler ile tahminler kıyaslanarak elde edilen sonuçlar Tablo 30'da gösterilmiştir.

Tablo 30: Web Ortamında Çalıştırılan BTC/TL Fiyat Tahmini Modelinin Sonuçlarının Karşılaştırılması

	1D	1W	1M	1Y
LSE Tahmin	46000 (-)	48250 (+)	47500 (+)	59000 (+)
ANFIS Tahmin	46750 (N)	45250 (-)	44750 (-)	52000 (+)
YSA Tahmin	47100 (+)	47000 (N)	50250 (+)	28000 (-)
LSTM Tahmin	47000 (+)	47000 (N)	46000 (-)	39000 (-)
Gerçekleşen	46750 (N)	47250 (+)	47200 (+)	63550 (+)
Tahmin Edilen Zaman	06.06.2019 23:30	07.06.2019 06:30	08.06.2019 10:30	24.06.2019 22:30

Tablo'da (N): Değişim yok, (+): Artış, (-): Azalış anlamına gelmekte olup BTC'nin son değerine kıyaslanarak yaklaşık olarak belirlenmiştir. İşaretler, başlangıç değeri olan 46.750 TL'ye göre değerlendirilmiştir.

Artış yönleri dikkate alındığında özellikle LSE'nin daha önceki deneysel çalışmalarda da olduğu gibi başarılı sonuçlar verdiği görülmektedir.

BTC/TL fiyat değeri ve fiyat yönü tahmininin anlık veriler ile yapılmasını sağlayan www.futureofbtc.com internet adresinde; 1 günlük, 1 haftalık, 1 aylık, 1 yıllık ve 2 yıllık dinamik veriler ile farklı zaman aralıklarına ait tahminler gerçekleştirilmektedir. Tahminleri kapsayan zaman aralıkları Tablo 31'de verilmiştir.

Tablo 31: Anlık Olarak Web Sitesi Üzerindeki Tahmin Aralıkları

Veri Seti	Açıklama	Tahmin Aralığı	Tahmin Adedi
1 Günlük	30 Dakikalık aralıklarla 1 günlük veri	Sonraki 1 Saate kadar	2
1 Haftalık	2 Saatlik aralıklarla 1 haftalık veri	Sonraki 8 saate kadar	4
1 Aylık	6 Saatlik aralıklarla 1 aylık veri	Sonraki 1 buçuk güne kadar	6
1 Yıllık	3 Günlük aralıklarla 1 yıllık veri	Sonraki 18 güne kadar	6

2 Yıllık	2 Günlük aralıklarla 2 yıllık veri	Sonraki 36 güne kadar	18
----------	------------------------------------	-----------------------	----

Tahmin adedi, eğitim ve test verilerinin yüzdeleri üzerinden hesaplanır. Bu çalışmada, eğitim verilerinin %95'i, yani test verilerinin %5'ini kapsayacak şekilde ileri dönemlerin tahminlenmesi gerçekleştirildiği için farklı zaman aralıklarının farklı tahmin adedi ortaya çıkmıştır. Örneğin, www.futureofbtc.com adresinde 1 aylık verileri kullanarak tahminleme yapmak isteyen bir kullanıcı, aslında 6 saatlik aralıklarla (6 saatte bir) alınan 1 aylık veri üzerinden bir tahminleme gerçekleştirmiş olur. 6 saatlik bir aylık verinin toplam sayısı 120'dir. Bu veri eğitim verisi olarak değerlendirilir ve eğitim verisinin %5'i, yani 6 adet tahmin gerçekleştirilir. 6 saatlik veriler ile 6 adet tahmin gerçekleştirildiğinde, toplam 36 saatlik (1 buçuk gün) bir zamanı kapsayan bir zaman aralığı ortaya çıkmaktadır.

SONUÇLAR

Kripto para birimleri arasında en büyük borsa hacmine ve fiyat değerine sahip olan BTC'ye olan ilgi gün geçtikçe artmaktadır. 2008 küresel finans krizinden sonra ortaya çıkan BTC, özellikle son iki buçuk yılda olağanüstü bir şekilde değerlendirilmiştir. Normal para ile birçok ortak özelliğe sahip olan kripto paraların sayısı ve finans sektöründeki hacmi her geçen gün artmaktadır. Türkiye'de de son yıllarda kripto para birimlerinin takas edildiği kripto para borsaları kurulmuştur. Teknolojinin finans sektörünü dönüştürmeye başladığı bu dönemde kripto paraların, bilhassa BTC'nin gelecekteki değeri, pozisyonu ve karşılığı oldukça belirsiz olmasına rağmen gelecekte var olacağına kesin gözüyle bakılmaktadır.

Bu çalışmada, BTC/TL fiyatının Dolar/TL kuru ve tarih öznitelikleri kullanılarak yapay zekâ ve derin öğrenme algoritmaları ile fiyat tahmini yapılmıştır. Seçilen verilerin zaman aralığı BTC/TL fiyatının büyük bir sıçrayış yapmaya başladığı dönem olan 08.12.2016 ve 08.12.2018'dir. Veriler 5-kat çapraz doğrulama ve zaman serisi analizleri ile aynı yapay zekâ yöntemleri ve parametreleri ile MATLAB R2018b ve PyCharm Community Edition 2019.1 programları kullanılarak 5 kez çalıştırılmıştır.

5-Kat çapraz doğrulama yöntemi daha başarılı bulunmasına rağmen, BTC/TL fiyat tahmini gibi zaman serisi olarak kabul edilen bir gerçek zamanlı tahminleme modeli için zaman serisi yöntemi de tercih edilmiştir. 5-Kat çapraz doğrulama yöntemi kullanılarak 4 farklı yapay zekâ yöntemine ait en iyi parametreler ve bu parametrelerin uygulanması sonucunda en iyi veri seti grupları yani BTC/TL tahmininin en iyi sonucu verdiği zaman aralığı belirlenmiştir.

Standart ve normalize edilmiş ve %90 ve %95 eğitim veri seti arasından %95 standart eğitim seti, zaman serisi için en iyi sonucu vermiştir. Zaman serisi analizlerinde sıkça kullanılan ve derin öğrenme algoritmalarından kabul edilen LSTM yöntemi tahminleme için PyCharm Community Edition 2019.1 programında kullanılmıştır.

Çalışmada gerçek zamanlı tam tahminleme yani gerçek hayata uyarlanabilen bir model geliştirildiği için BTC/TL fiyat tahmininde öznitelik olarak kullanılan Dolar/TL veri değeri gelecekte bilinemeyeceği için eğitim veri setinin son değerine sabitlenmiştir. LSTM'nin zaman serisine uygun şekilde ani inişli-çıkışlı bir grafik yerine daha çok

dalgalanmalı bir sonuç vermesinin sebebi olarak da doların sabitlenmesi ve Kaos teorisi gösterilmektedir. Tıpkı “kaos teorisinde” olduğu gibi değişimi etrafındaki bağlı olduğu etkenlerin değişimine bağlı olan nesneler veya durumlar gibi fiyat değişimindeki etkenlerden biri olan Doların sabitlenmesi, LSTM modelini etkilemiştir.

Sonuç olarak eğri uydurma ve ANFIS yöntemlerinin diğer yöntemlere göre oldukça başarılı sonuçlar verdiği görülmüştür. Mevcut kripto para borsalarını domine edecek kadar güçlü olan BTC’ye uygulandığında başarılı sonuçlar alınan bu modellerin, diğer kripto para birimlerinden Ethereum ve Ripple’a da uygulanabileceği sonucuna varılmıştır.

Bu sebeple, www.futureofbtc.com adresinde hizmet veren bir web sitesi kurulmuş olup, bu web sitesi ile anlık verilerle çalışarak kullanıcılara, çalışmada belirlenen en iyi veri seti gruplarına ait tarih aralıkları içerisinde yatırım ve kripto para alım-satımı yapma fırsatı sunulmaktadır. Ayrıca, web sitesi ile çalışacak olan bu modelin zamanın şartlarına göre yeni kripto para birimleri ve zaman aralıkları belirlenerek çalıştırılması düşünülmektedir.

Bu çalışmada kısıt olarak düşünülecek konular arasında Dolar/TL değerinin zaman serisi problemlerinde sabitlenmesi, yapay zekâ modellerine ait parametreler belirlenirken sadece BTC fiyatından yararlanılması gösterilebilir.

İleriki çalışmalarda daha fazla kripto para çeşidinin kullanılması, ekonomide uygulanan diğer yardımcı bilgilerinde modele öznitelik olarak eklenilmesi ile tahmin çalışılabilir.

KAYNAKÇA

- Akcora, C. G., Dey, A. K., Gel, Y. R., & Kantarcioglu, M. (2018), "Forecasting Bitcoin Price with Graph Chainlets", In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, Vol.10939, (765-776).
- Alessandretti, L., ElBahrawy, A., Aiello, L. M., & Baronchelli, A. (2018), "Anticipating Cryptocurrency Prices Using Machine Learning", *Complexity*.
- Alpaydın, E. (2018), "Yapay Öğrenme", Boğaziçi Üniversitesi Yayınevi, 4.Baskı.
- Antonopoulos, A. M. (2014). "Mastering Bitcoin: Unlocking Digital Cryptocurrencies", O'Reilly Media, Inc."
- BitcoinWiki, (2015). "Bitcoin History", (05.05.2019). <http://en.bitcoinwiki.org/>
- Brand, A. W. (2016), "Bitcoin for Dummies".
- Ceyhan, K., Kurtulmaz, E., Sert, O. C., & Özyer, T. (2018) "Bitcoin Movement Prediction with Text Mining", In *2018 26th Signal Processing and Communications Applications Conference (SIU)*, (1-4).
- Charles, A., & Darné, O. (2019), "Volatility Estimation for Bitcoin: Replication and Robustness", *International Economics*, Vol.157, (23-32).
- ÇARKACIOĞLU, A. (2016), "Kripto-Para Bitcoin", *Sermaye Piyasası Kurulu Araştırma Dairesi Araştırma Raporu*.
- Çıraklı, Ü.; Dalkılıç, S. & Hacıhasanoğlu, T. (2017), "Kaos Teorisi, Karmaşıklık Teorisi, Karmaşık Uyarlamalı Sistemler: Sağlık Hizmetleri Açısından Bir Derleme", C.3, S.16, (330-343).
- Çütcü, Ö. Ü. İ., & Kılıç, Ö. Ü. Y. (2018), "Bitcoin Fiyatları İle Dolar Kuru Arasındaki İlişki: Yapısal Kırılmalı Zaman Serisi Analizi", C.16, S.4, (349-366).
- Ertürk, A. (2012), "Kaos Kuramı: Yönetim ve Eğitimdeki Yansımaları", *Kastamonu Eğitim Dergisi*, C.20, S.3, (849 – 868).
- Fettahoğlu, S. (2009), "Pay Senedi Fiyatlarının Tahmin Edilebilirliği: Kaos Kuramı Yaklaşımı", *Mufad Dergisi*, C.43, (237-243).

- Giudici, P., & Abu-Hashish, I. (2019), "What Determines Bitcoin Exchange Prices? A Network VAR Approach", *Finance Research Letters*, Vol.28, (309-318).
- Goodfellow, I., Bengio, Y., & Courville, A. (2016), "Deep learning", MIT press.
- Goodfellow Ian vd., *Derin Öğrenme*, 1.Baskı, Buzdağı Yayınevi, 2018.
- Graydon, C. (2014), "[What is Cryptocurrency?](#)" (05.05.2019)
- Greaves, A., & Au, B. (2015), "Using The Bitcoin Transaction Graph to Predict The Price of Bitcoin", *No Data*.
- Gullapalli, S. (2018), "Learning to Predict Cryptocurrency Price Using Artificial Neural Network Models of Time Series".
- Guo, T., & Antulov-Fantulin, N. (2018), "An Experimental Study of Bitcoin Fluctuation Using Machine Learning Methods", *arXiv preprint arXiv:1802.04065*.
- Guo, T., Bifet, A., & Antulov-Fantulin, N. (2018), "Bitcoin Volatility Forecasting with a Glimpse Into Buy and Sell Orders", In *2018 IEEE International Conference on Data Mining (ICDM)*, (989-994).
- Hamzaçebi, C., & Kutay, F. (2004), "Yapay Sinir Ağları ile Türkiye Elektrik Enerjisi Tüketiminin 2010 Yılına Kadar Tahmini", *Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi*, C.19, S.3.
- Indera, N. I., Yassin, I. M., Zabidi, A., & Rizman, Z. I. (2017), "Non-Linear Autoregressive with Exogeneous Input (NARX) Bitcoin Price Prediction Model Using PSO-Optimized Parameters and Moving Average Technical Indicators", *Journal of Fundamental and Applied Sciences*, Vol.9, Issue3S, (791-808).
- Jang, H., & Lee, J. (2018), "An Empirical Study on Modeling and Prediction of Bitcoin Prices with Bayesian Neural Networks Based on Blockchain Information", *IEEE Access*, Vol.6, (5427-5437).
- Jang, J. S. (1993), "ANFIS: Adaptive-Network-Based Fuzzy Inference System", *IEEE transactions on systems, man, and cybernetics*, Vol.23, Issue.3, (665-685).
- Karparthy, A. (2017), "A Peek At Trends In Machine Learning".

- Khalilov, M. C. K., Gündebahar, M., & Kurtulmuşlar, İ. (2017), “Bitcoin ile Dünya ve Türkiye’deki Dijital Para Çalışmaları Üzerine Bir İnceleme”, *Proceedings of 19. Akademik Bilişim Konferansı*.
- Kılıç, Y., & Çütcü, İ. (2018), “Bitcoin Fiyatları ile Borsa İstanbul Endeksi Arasındaki Eşbütünleşme ve Nedensellik İlişkisi”, *Eskişehir Osmangazi Üniversitesi İktisadi ve İdari Bilimler Dergisi*, C.13, S.3, (235-250).
- Kırbaş, İ. (2018), “Blokzinciri Teknolojisi ve Yakın Gelecekteki Uygulama Alanları”, *Mehmet Akif Ersoy Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, C.9, S.1, (75-82).
- Kingma, D. P., & Ba, J. (2014), “Adam: A Method for Stochastic Optimization”, *arXiv preprint arXiv:1412.6980*.
- Koçoğlu, Ş., Çevik, Y. E., & Tanrıöven, C. (2016), “Bitcoin Piyasalarının Etkinliği, Likiditesi ve Oynaklığı”. *Journal of Business Research-Türk*, C.8, S.2, (77-97).
- Kohavi, R. (1995), “A Study Of Cross-Validation And Bootstrap For Accuracy Estimation And Model Selection”, In *Ijcai*, Vol.14, Issue.2, (1137-1145).
- LeCun, Y., Bengio, Y., & Hinton, G. (2015), “Deep Learning”, *nature*, Vol.521, Issue.7553, (436).
- Madan, I., Saluja, S., & Zhao, A. (2015), “Automated Bitcoin Trading via Machine Learning Algorithms”, URL: <http://cs229.stanford.edu/proj2014/Isaac%20Madan>, (20).
- Marian, O. (2016), “Kripto Para Birimi Üstün Vergi Cenneti mi?”, *İstanbul Üniversitesi Hukuk Fakültesi Mecmuası*, C.74, S.2, (919-930).
- Matta, M., Lunesu, I., & Marchesi, M. (2015), “Bitcoin Spread Prediction Using Social and Web Search Media”, In *UMAP Workshops*, (1-10).
- McClelland, J. L., Rumelhart, D. E., & PDP Research Group, (1986), “Parallel Distributed Processing”, *Explorations in the Microstructure of Cognition*, Vol.2, (216-271).

- McNally, S., Roche, J., & Caton, S. (2018), "Predicting The Price Of Bitcoin Using Machine Learning", In *2018 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP)*, (339-343).
- Nakamoto, S. (2008), "Bitcoin: A Peer-To-Peer Electronic Cash System".
- Narayan, P. K., Narayan, S., Rahman, R. E., & Setiawan, I. (2019), "Bitcoin Price Growth and Indonesia's Monetary System", *Emerging Markets Review*, Vol.38, (364-376).
- Öztemel, E. (2003). Yapay sinir ağları. *Papatya Yayıncılık, İstanbul*.
- PISA, M. ve JUDEN, M. (2017), "Blockchain and Economic Development: Hype vs. Reality", *Center For Global Development*, Vol.107, (5-7).
- Polat, M., & Akbıyık, A., (2019), "Sosyal Medya ve Yatırım Araçlarının Değeri Arasındaki İlişkinin İncelenmesi: Bitcoin Örneği", *Akademik İncelemeler Dergisi (AID)*, C.14, S.1, (443-462).
- Purbarani, S. C., & Jatmiko, W. (2018), "Performance Comparison of Bitcoin Prediction in Big Data Environment", In *2018 International Workshop on Big Data and Information Security (IWBIS)* (99-106).
- Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1985), "Learning Internal Representations by Error Propagation", (No. ICS-8506). California Univ San Diego La Jolla Inst for Cognitive Science.
- Sakiz, B., & Kutlugün, E. (2018), "Bitcoin Price Forecast via Blockchain Technology and Artificial Intelligence Algorithms", In *2018 26th Signal Processing and Communications Applications Conference (SIU)*, (1-4).
- Schneider, B. (1996), "Applied Cryptology: protocols, algorithms, and source code in C."
- Seker, S. E., Cankir, B., & Arslan, M. L. (2014), "Information and Communication Technology Reputation for XU030 Quote Companies", *arXiv preprint arXiv:1406.5073*.

- Seker, S. E., Mert, C., Al-Naami, K., Ozalp, N., & Ayan, U. (2014), "Time Series Analysis on Stock Market for Text Mining Correlation of Economy News", *arXiv preprint arXiv:1403.2002*.
- Shah, D., & Zhang, K. (2014), "Bayesian Regression and Bitcoin", In *2014 52nd annual Allerton conference on communication, control, and computing (Allerton)*, (409-414).
- Singh, H., & Agarwal, P. (2018), "Empirical Analysis of Bitcoin Market Volatility Using Supervised Learning Approach", In *2018 Eleventh International Conference on Contemporary Computing (IC3)*, (1-5).
- Spilak, B. (2018), "Deep Neural Networks for Cryptocurrencies Price Prediction", (Master's thesis, Humboldt-Universität zu Berlin).
- Şahin, E. E. (2018), "Kripto Para Bitcoin: ARIMA ve Yapay Sinir Ağları İle Fiyat Tahmini", *Fiscaoeconomia*, C.2, S.2, (74-92).
- Şahin, E. E., & ÖZKAN, O. (2018), "Asimetrik Volatilitenin Tahmini: Kripto Para Bitcoin Uygulaması", *Bilecik Şeyh Edebali Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, C.3, S.2, (240-247).
- Tieleman, T., & Hinton, G. (2012), "Lecture 6.5-rmsprop: Divide the Gradient by a Running Average of Its Recent Magnitude", *COURSERA: Neural networks for machine learning*, Vol.4, Issue.2, (26-31).
- Tosun, T. (2006), "Türev Araçlar, Kaos Teorisi Fraktal Yapıların Vadeli İşlem Zaman Serilerinde Uygulanması", Marmara Üniversitesi, Bankacılık ve Sigortacılık Enstitüsü, Yüksek Lisans Tezi, İstanbul.
- Werbos, P.J., (1974), "Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences", PhD thesis, Harvard University.
- Yang, S. Y., & Kim, J. (2015), "Bitcoin Market Return and Volatility Forecasting Using Transaction Network Flow Properties", In *2015 IEEE Symposium Series on Computational Intelligence*, (1778-1785).

- Yavuz, M. S. (2019), “Ekonomide Dijital Dönüşüm: Blockchain Teknolojisi ve Uygulama Alanları Üzerine Bir İnceleme”, *Finans Ekonomi ve Sosyal Araştırmalar Dergisi (FESA)*, C.4, S.1, (15-29).
- Yerlikaya, T., Buluş, E., & BULUŞ, N. (2006), Kripto Algoritmalarının Gelişimi ve Önemi”, *Akademik Bilişim Konferansları*, (9-11).
- Yılmaz, Y. (2007), “Kriptoloji Uygulamalarında Hukuki Boyut”.
- Zeren, F., & Esen, S. (2018), “Geleceğin Para Birimi Ya Da Sadece Bir Balon: Bitcoin”, *Balıkesir Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, C.21, S.39, (433-448).
- Zimmerman, E., "The Evolution of Fintech" , The New York Times, 2016.

İNTERNET

<http://colah.github.io/posts/2015-08-Understanding-LSTMs/> (07.05.2019)

<http://www.bitcoinhaber.net/2013/08/bitcoin-nedir-nasil-calisir.html> (Erişim Tarihi: 02.05.2019)

<http://www.derinogrenme.com/2015/07/21/derin-ogrenme-deep-learning-nedir/> (09.05.2019)

<https://coin-turk.com/bitcoin-blok-odulunun-yarilanmasi-piyasalari-nasil-etkileyecek> (02.05.2019)

<https://computersciencewiki.org/index.php/Max-pooling / Pooling> (08.05.2019)

<https://devhunteryz.wordpress.com/2018/07/14/uzun-kisa-sureli-bellek-long-short-term-memory/> (07.05.2019)

<https://engmrk.com/adam-optimization-algorithm/> (10.05.2019)

https://medium.com/@craig_10243/digital-signature-rules-and-their-relationship-to-bitcoin-b1faeae1f446 (01.05.2019)

<https://medium.com/@KarmaCoverage/blockchain-in-the-risk-markets-4983965f33ff> (01.05.2019)

<https://medium.com/@RaghavPrabhu/understanding-of-convolutional-neural-network-cnn-deep-learning-99760835f148> (08.05.2019)

<https://medium.com/@tuncerergin/convolutional-neural-network-convnet-yada-cnn-nedir-nasil-calisir-97a0f5d34cad> (08.05.2019)

<https://medium.com/deep-learning-turkiye/derin-ogrenme-uygulamalarinda-en-sik-kullanilan-hiper-parametreler-ece8e9125c4> (09.05.2019)

<https://medium.com/deep-learning-turkiye/derin-ogrenme-uygulamalarinda-en-sik-kullanilan-hiper-parametreler-ece8e9125c4> (10.05.2019)

<https://medium.com/deep-learning-turkiye/sinir-a%C4%9Flar%C4%B1-ve-derin-%C3%B6%C4%9Frenme-v-grandyan-d%C3%BC%C5%9F%C3%BC%C5%9F%C3%BC-8c6d15a3d965> (09.05.2019)

<https://passwordsgenerator.net/sha256-hash-generator/> (01.05.2019)

<https://towardsdatascience.com/adam-latest-trends-in-deep-learning-optimization-6be9a291375c> (10.05.2019)

[https://www.academia.edu/5792640/ZAMAN SERI ANALIZINDE TEMEL KAVR AMLAR](https://www.academia.edu/5792640/ZAMAN_SERI_ANALIZINDE_TEMEL_KAVR_AMLAR) (11.05.2019)

<https://www.bitcoinmining.com/images/what-is-bitcoin-mining-difficulty-high-resolution.png> (02.05.2019)

<https://www.bitcoinmining.com/tr/> (02.05.2019)

<https://www.blockchain.com/tr/> (02.05.2019)

<https://www.blockchain.com/tr/charts/market-price?timespan=all> (01.05.2019)

<https://www.getrevue.co/profile/Hakan/issues/blok-zinciri-5-blok-ve-mutabakat-77189> (01.05.2019)

<https://www.investaz.com.tr/> (13.05.2019).

<https://www.mathworks.com/help/deeplearning/ref/classificationlayer.html> (08.05.2019)

<https://www.mathworks.com/help/deeplearning/examples/train-a-convolutional-neural-network-for-regression.html> (08.05.2019)

https://www.mathworks.com/help/matlab/math/polynomial-curve-fitting.html?searchHighlight=Polynomial%20Curve%20Fitting&s_tid=doc_srcht
[itle](#) (06.05.1019)

<https://www.nytimes.com/2016/04/07/business/dealbook/the-evolution-of-fintech.htm>

https://www.researchgate.net/figure/ANFIS-Structure-with-two-inputs-one-output-and-two-membership-function-for-each-input_fig2_223914624 (05.05.2019)

<https://www.sciencedirect.com/science/article/pii/S1674775513000437> (06.05.2019)

<https://www.tensorflow.org/> (07.05.2019)

<https://www.tensorflow.org/tutorials/keras?hl=tr> (07.05.2019)

EKLER

EK 1**ÖZGEÇMİŞ****KİŞİSEL BİLGİLER :****Ad:** Furkan**Soyad:** ATLAN**Adres:** Şehitlik mahallesi Mehmet Akif Ersoy caddesi Turgut Özal sitesi 22/B Blok
Daire: 5 Yenişehir/Diyarbakır**Cep Telefonu:** +90 544 576 18 28**E-mail:** furkanatlann@gmail.com**Doğum Tarihi/Yeri:** 24.01.1994 – Diyarbakır**Medeni Durumu:** Bekâr**EĞİTİM BİLGİLERİ :**

- **2017 – 2019:** Mehmet Akif Ersoy Üniversitesi Sosyal Bilimler Enstitüsü Yönetim Bilişim Sistemleri Anabilim Dalı
- **2013 – 2017:** Mehmet Akif Ersoy Üniversitesi Yönetim Bilişim Sistemleri Bölümü
- **2008 – 2012:** Türk Telekom Mesleki ve Teknik Anadolu Lisesi Bilişim Teknolojileri Bölümü

YABANCI DİL BİLGİSİ :

- İngilizce: Orta Derecede

İŞ DENEYİMLERİ :**Operatör Haberleşme Stajyeri** – İnser Mühendislik ve Dış Ticaret Ltd.Şti.**AKADEMİK ÇALIŞMALAR :**

- F. ATLAN, İ. PENÇE, M. ŞİŞECİ ÇEŞMELİ, and A. KALKAN, “Estimation of Bitcoin Market Value in Turkey Using Artificial Intelligence Techniques,” presented at the 5th International Management Information Systems Conference, ANKARA, 2018.