

**BAŐKENT ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ  
BİLGİSAYAR MÜHENDİSLİĐİ ANABİLİM DALI  
BİLGİSAYAR MÜHENDİSLİĐİ  
TEZLİ YÜKSEK LİSANS PROGRAMI**

**YAPAY SİNİR AĐLARI OPTİMİZASYON ALGORİTMALARININ  
BANKA ÖZKAYNAK KARLILİĐİ TAHMİNİ ÜZERİNDE  
KARŐILAŐTIRMALI PERFORMANS ANALİZİ**

**HAZIRLAYAN**

**TOLGAY BALCI**

**YÜKSEK LİSANS TEZİ**

**ANKARA - 2021**



**BAŐKENT ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ  
BİLGİSAYAR MÜHENDİSLİĐİ ANABİLİM DALI  
BİLGİSAYAR MÜHENDİSLİĐİ  
TEZLİ YÜKSEK LİSANS PROGRAMI**

**YAPAY SİNİR AĐLARI OPTİMİZASYON ALGORİTMALARININ  
BANKA ÖZKAYNAK KARLILIĐI TAHMİNİ ÜZERİNDE  
KARŐILAŐTIRMALI PERFORMANS ANALİZİ**

**HAZIRLAYAN**

**TOLGAY BALCI**

**YÜKSEK LİSANS TEZİ**

**TEZ DANIŐMANLARI**

**PROF. DR. NİZAMİ GAŐİLOV  
PROF. DR. HASAN OĐUL**

**ANKARA - 2021**

**BAŞKENT ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**

Bilgisayar Mühendisliği Anabilim Dalı Bilgisayar Mühendisliği Tezli Yüksek Lisans Programı çerçevesinde Tolgay BALCI tarafından hazırlanan bu çalışma, aşağıdaki jüri tarafından Yüksek Lisans Tezi olarak kabul edilmiştir.

Tez Savunma Tarihi: 02 / 04 / 2021

**Tez Adı:** Yapay Sinir Ağları Optimizasyon Algoritmalarının Banka Özkaynak Karlılığı Tahmini Üzerinde Karşılaştırmalı Performans Analizi

**Tez Jüri Üyeleri ( Unvanı, Adı - Soyadı, Kurumu)**

**İmza**

Prof. Dr. Nizami Gasilov, Başkent Üniversitesi

.....

Doç. Dr. Mustafa Sert, Başkent Üniversitesi

.....

Dr. Öğr. Üyesi Serdar Arslan, Çankaya Üniversitesi

.....

**ONAY**

Prof. Dr. Faruk ELALDI

Fen Bilimleri Enstitüsü Müdürü

Tarih : ... / ... / .....

**BAŞKENT ÜNİVERSİTESİ**  
**FEN BİLİMLER ENSTİTÜSÜ**  
**YÜKSEK LİSANS ÇALIŞMASI ORJİNALLİK RAPORU**

Tarih: .... / .... / .....

Öğrencinin Adı, Soyadı: Tolgay Balcı

Öğrencinin Numarası: 21810059

Anabilim Dalı: Bilgisayar Mühendisliği

Programı: Bilgisayar Mühendisliği Tezli Yüksek Lisans Programı

Danışmanın Unvanı/Adı, Soyadı: Prof. Dr. Nizami Gasilov, Prof. Dr. Hasan Oğul

Tez Başlığı: Yapay Sinir Ağları Optimizasyon Algoritmalarının Banka Özkaynak Karlılığı  
Tahmini Üzerinde Karşılaştırmalı Performans Analizi

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

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

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

Öğrenci İmzası: .....

**ONAY**

Tarih: .... / .... / .....

Prof. Dr. Nizami Gasilov

## TEŐEKKÜR

Tez süresince desteklerini esirgemeyen Danışman Hocalarım Prof. Dr. Nizami GASİLOV ve Prof. Dr. Hasan OĐUL'a, Tez Jürimde bulunan ve yönlendirmelerinden faydalandığım Doç. Dr. Mustafa SERT ve Dr. Öğr. Üyesi Serdar ARSLAN hocalarıma teşekkürü borç bilirim.

Tüm hayatım boyunca her zaman kendilerinden madden ve manen destek aldığım Annem Müşerref BALCI'ya, Babam Halil BALCI'ya ve Ağabeyim Tolga BALCI'ya tüm kalbimle teşekkür ederim.

# ÖZET

**Tolgay BALCI**

## **YAPAY SİNİR AĞLARI OPTİMİZASYON ALGORİTMALARININ BANKA ÖZKAYNAK KARLILIĞI TAHMİNİ ÜZERİNDE KARŞILAŞTIRMALI PERFORMANS ANALİZİ**

**Başkent Üniversitesi Fen Bilimleri Enstitüsü**

**Bilgisayar Mühendisliği Anabilim Dalı**

**2021**

Bir ülkenin finansal sisteminin en önemli parçası olan bankacılık sektörünün performansını ve karlılığını ölçmek her zaman önemlidir. Performans ölçümü sayesinde bankalar rekabet durumunu, büyüme potansiyellerini ve riskleri değerlendirebilir ve faaliyetlerini sürdürmede daha başarılı olabilirler. Literatürde, bankacılık performans değerlendirmesinde yapay sinir ağları (YSA) nadiren kullanılmıştır. Bu nedenle, mevcut tezde bankaların performanslarını kestirmek için YSA kullanımının derinlemesine incelenmesi hedeflenmiştir. Amaç, bir dizi finansal parametreler verildiğinde, YSA hesaplama yöntemlerini kullanarak bankaların özkaynak getirilerini tahmin etmektir. Tez, Türkiye'deki tüm mevduat bankaları ele alınarak gerçekleştirilmiştir. Bağımsız değişkenler olarak kabul edilen 6 dış ve 8 iç parametrenin 11 yıla ait çeyrek dönemlik verileri kullanılmıştır. Veri kümesi; Bankacılık Düzenleme ve Denetleme Kurumu, Türkiye Bankalar Birliği, Türkiye İstatistik Kurumu ve Türkiye Cumhuriyet Merkez Bankası bilgilerine dayanarak oluşturulmuştur. Çeşitli YSA optimizasyon algoritmalarının, bağımlı değişken olarak alınan banka özkaynak getirisini tahmin etme başarıları karşılaştırılmıştır. Yapılan sayısal deneyler sonucunda, kullanılan tüm YSA yöntemlerinin %80'in üzerinde doğruluk payı ile özkaynak karlılığını tahmin ettikleri görülmüştür. Ayrıca, en iyi YSA yönteminin her banka için farklı olduğu gözlemlenmiştir. Lineer regresyon yöntemi ile karşılaştırma, YSA yöntemlerinin daha başarılı olduklarını göstermiştir.

**ANAHTAR KELİMELER:** Yapay Sinir Ağları, Stokastik Gradyan İnişi, ADAM yöntemi, RMSprop yöntemi, Banka Özkaynak Karlılığı

# ABSTRACT

**Tolgay BALCI**

## **COMPARATIVE PERFORMANCE ANALYSIS OF ARTIFICIAL NEURAL NETWORK OPTIMIZATION ALGORITHMS ON THE ESTIMATION OF BANK RETURN ON EQUITY**

**Başkent University Institute of Science**

**Department of Computer Engineering**

**2021**

It is always important to measure the performance and profitability of the banking sector, which is the most important part of a country's financial system. Thanks to performance measurement, banks can evaluate the competitive situation, growth potential and risk, and they can be more successful to continue their activities. In the literature, artificial neural networks (ANN) have rarely been used for evaluating economic performance. Therefore, in the present thesis, it is aimed to examine the use of ANN in depth to estimate the performance of banks. The aim is to estimate returns on bank equity by using ANN calculation methods for given a set of financial parameters. The thesis was carried out by considering all deposit banks in Turkey. Quarterly data for 11 years of 6 external and 8 internal parameters, which are considered as independent variables, were used. The dataset was created based on information of Banking Regulation and Supervision Agency, the Banks Association of Turkey, Turkey Statistical Institute and the Central Bank of the Republic of Turkey. The success of various ANN optimization algorithms in predicting the bank return on equity, which is regarded as the dependent variable, has been compared. As a result of the numerical experiments conducted, it was seen that all the ANN methods predicted the return on equity with an accuracy of over 80%. Also, it has been observed that the best ANN method is different for each bank. Comparison with linear regression method showed that ANN methods are more successful.

**KEYWORDS:** Artificial Neural Network, Stochastic Gradient Descent, ADAM method, RMSprop method, Return on Equity (ROE)



# İÇİNDEKİLER

TEŞEKKÜR.....	i
ÖZET .....	ii
ABSTRACT .....	iii
İÇİNDEKİLER.....	iv
TABLolar LİSTESİ .....	vii
ŞEKİLLER LİSTESİ .....	viii
SİMGELER VE KISALTMALAR.....	ix
1. GİRİŞ.....	1
2. LİTERATÜR TARAMASI .....	4
3. YAPAY SİNİR AĞLARI VE OPTİMİZASYON ALGORİTMALARI.....	10
3.1. Yapay Sinir Ağlarına Giriş.....	10
3.2. Yapay Sinir Ağları Yapısı.....	14
3.2.1. Girdi katmanı .....	14
3.2.2. Gizli katman.....	14
3.2.3. Çıktı katmanı .....	14
3.3. Yapay Sinir Ağ Modelleri.....	15
3.3.1. Tek katmanlı algılayıcılar.....	15
3.3.2. Basit algılayıcılar .....	16
3.3.3. Uyarlanabilir ve çoklu uyarlanabilir doğrusal eleman.....	16
3.3.4. Çok katmanlı algılayıcılar .....	18
3.4. Yapay Sinir Ağlarında Öğrenme Yöntemleri.....	20
3.4.1. Danışmanlı öğrenme .....	20
3.4.2. Danışmansız öğrenme .....	20
3.4.3. Yarı danışmanlı öğrenme .....	21
3.5. Yapay Sinir Ağlarının Eğitimi ve Test Edilmesi .....	22
3.5.1. YSA eğitimi .....	22

3.5.2.	YSA'nın test edilmesi.....	23
3.6.	Yapay Sinir Ağlarının Temel Özellikleri .....	23
3.7.	Yapay Sinir Ağlarının Avantajları ve Dezavantajları .....	24
3.8.	Yapay Sinir Ağları Optimizasyon Algoritmaları .....	25
3.8.1.	Stokastik gradyan inişi metodu.....	25
3.8.2.	Uyarlamalı moment kestirimi metodu .....	25
3.8.3.	Karesel ortalamanın karekökü yayılımı metodu.....	27
3.8.4.	Uyarlamalı gradyan algoritması.....	28
3.8.5.	Uyarlamalı öğrenme oranı metodu.....	30
3.8.6.	Momentum algoritması uygulanmış stokastik gradyan inişi.....	31
3.8.7.	Hızlandırılmış Nesterov gradyan algoritması (NAG).....	33
3.8.8.	Hızlandırılmış uyarlamalı Nesterov moment kestirimi algoritması	34
4.	DENEY, GÖZLEM ve ANALİZ.....	35
4.1.	Metotlar .....	35
4.1.1.	Yapay sinir ağları.....	35
4.1.2.	Lineer regresyon.....	36
4.2.	İmplementasyon Detayları.....	37
4.3.	Sonuçlar.....	40
4.4.	Veri Kümesi .....	40
4.4.1.	Banka bilgileri .....	40
4.4.2.	Veri kümesi açıklamaları.....	40
4.5.	Problem Tanımı ve Çalışmanın Amacı .....	43
4.5.1.	Arka plan .....	43
4.5.2.	Çalışmada incelenen problem .....	44
4.5.3.	Veri kümesi ile ilgili kısıtlamalar .....	45
4.6.	Deneysel Kurulum.....	46
4.6.1.	Yapay sinir ağları eğitim süreci .....	47

4.6.2.	Lineer Regresyon Süreci.....	55
4.7.	Bulgular.....	57
4.7.1.	Uyarlamalı moment kestirimi metodu ile ROE hesaplanması.....	59
4.7.2.	Stokastik gradyan inişi metodu ile ROE hesaplanması.....	60
4.7.3.	Karesel ortalamanın karekökü yayılımı metodu ile ROE hesaplanması.....	62
4.7.4.	Lineer regresyon ile ROE hesaplanması.....	64
5.	GELECEK ÇALIŞMALAR.....	67
6.	TARTIŞMA ve SONUÇ.....	68
	KAYNAKLAR.....	71

## TABLULAR LİSTESİ

Tablo 2. 1. Literatürün farklı alanlarında optimizasyon algoritmalarının kullanımı.....	8
Tablo 3. 1. Aktivasyon fonksiyonları .....	12
Tablo 3. 2. Yapay sinir ağları ve biyolojik sinir ağları arasında benzerlikler .....	13
Tablo 4. 1. Ağ parametreleri.....	37
Tablo 4. 2. Banka bilgileri .....	40
Tablo 4. 3. Veri kümesinin değişkenleri .....	43
Tablo 4. 4. Epok değerlerine göre Loss, MAE, MSE değerleri .....	49
Tablo 4. 5. Farklı momentum katsayıları kullanarak elde edilen değerler .....	50
Tablo 4. 6. Modellerin doğruluk değer sonuçları .....	58
Tablo 4. 7. ADAM metoduna göre sonuç ve serpilme grafikleri .....	59
Tablo 4. 8. SGD metoduna göre sonuç ve serpilme grafikleri .....	61
Tablo 4. 9. RMSprop metoduna göre sonuç ve serpilme grafikleri .....	63
Tablo 4. 10. Lineer regresyon metoduna göre sonuç grafikleri.....	65

## ŞEKİLLER LİSTESİ

Şekil 3. 1. Biyolojik sinir hücresi .....	10
Şekil 3. 2. Yapay sinir hücresi .....	11
Şekil 3. 3. Yapay sinir ağları katmanları .....	15
Şekil 3. 4. Tek katmanlı algılayıcı örneği.....	15
Şekil 3. 5. Basit algılayıcı modeli.....	16
Şekil 3. 6. ADALINE modeli .....	17
Şekil 3. 7. İki ADALINE ağından oluşmuş MADALINE modeli .....	17
Şekil 3. 8. Çok katmanlı algılayıcı modeli .....	18
Şekil 3. 9. İleri beslemeli ağ yapısı.....	19
Şekil 3. 10. Geri beslemeli ağ yapısı .....	19
Şekil 3. 11. Danışmanlı öğrenme .....	20
Şekil 3. 12. Danışmansız öğrenme .....	21
Şekil 3. 13. Yarı danışmanlı öğrenme .....	21
Şekil 3. 14. Momentum uygulanmış SGD ile momentum uygulanmamış SGD.....	32
Şekil 4. 1. Lineer regresyon modeli.....	36
Şekil 4. 2. Çalışmadaki yapay sinir ağı yapısı.....	48
Şekil 4. 3. Eğitim – validasyon kaybı grafiği ile uygun epok seçimi .....	49
Şekil 4. 4. Optimum momentum katsayısı tespiti.....	51
Şekil 4. 5. Çalışmadaki çapraz doğrulama yapısı.....	52
Şekil 4. 6. Normalizasyon öncesi ve sonrası Epok sayısı MAE grafikleri .....	53
Şekil 4. 7. YSA optimizasyon algoritmaları ile ROE hesaplama akış diyagramı .....	55
Şekil 4. 8. Lineer regresyon ile ROE hesaplanmasında akış diyagramı.....	56

## SİMGELER VE KISALTMALAR

$\Sigma$ : toplama fonksiyonu  
 $\epsilon$ : hata terimi  
 $\mu$ : momentum katsayısı  
 $\eta$ : öğrenme katsayısı  
ADAM: Uyarlamalı Moment Kestirimi  
AdaGrad: Uyarlamalı Gradyan Algoritması  
AdaDelta: Uyarlamalı Öğrenme Oranı Metodu  
ADALINE: Uyarlanabilir Doğrusal Eleman  
BDDK: Bankacılık Düzenleme Denetleme Kurulu  
BM: Bulanık Mantık  
CNN: Evrimsel Sinir Ağları  
DoS: Hizmet Engelleme  
GA: Genetik Algoritma  
GSYİH: Gayri Safi Yurtiçi Hasıla  
MADALINE: Çoklu Uyarlanabilir Doğrusal Elemanlar  
MAE: Ortalama Mutlak Hata  
MAPE: Ortalama Mutlak Yüzdellik Hata  
MSE: Ortalama Kare Hatası  
NADAM: Hızlandırılmış Uyarlamalı Nesterov Moment Kestirimi  
NAG: Hızlandırılmış Nesterov Gradyan Algoritması  
ReLU: Düzeltilmiş Lineer Birim  
ROA: Aktif Karlılığı  
ROE: Özkaynak Karlılığı  
RMSprop: Karesel Ortalamanın Karekökü Yayılımı  
SGD: Stokastik Gradyan İnişi  
SGDM: Momentumlu Stokastik Gradyan İnişi  
SUE: Sanayi Üretim Endeksi  
TBB: Türkiye Bankalar Birliği  
TCMB: Türkiye Cumhuriyet Merkez Bankası  
TL: Türk Lirası  
TÜFE: Tüketici Fiyat Endeksi  
TÜİK: Türkiye İstatistik Kurumu  
USD: Amerikan Doları  
YSA: Yapay Sinir Ağları  
ZKTL: TL Bazında Zorunlu Karşılık Oranı

# 1. GİRİŞ

Bir ülkenin finansal sisteminde en önemli aktör olan bankaların, güçlü ve sağlam bir yapıda olması oldukça önem taşımaktadır. Bunun nedeni bankacılık sisteminin tüm ekonomik birimlerle doğrudan ilişkili olması ve tüm ekonomiyi doğrudan etkileyebilmesidir. Bankaların geleneksel faaliyetleri olarak kabul edilen mevduat toplama ve kredi verme işlevi sayesinde insanlar birikimlerini güvence altında tutarak gelir elde edebilmekte ve ihtiyacı olduğunda nakit talepleri karşılanabilmektedir. Bu sayede ekonominin canlılığı devam etmektedir.

Ülkemizde ve dünyada 1980'lerde başlayan finansal alandaki serbestleşme bankaların hem faaliyetlerini hem de örgütsel yapılarını ciddi şekilde değiştirmiştir. Yabancı sermayenin ülkelere girmesi sayesinde ülkelerin bankacılık sistemlerinde ulusal bankaların yanısıra yabancı bankalarında faaliyetlerinin büyüdüğü görülmektedir. Bunun bir sonucu olarak uluslararası büyük bankaların meydana geldiği görülmüş, dünyanın tüm ülkelerindeki yatırım araçlarına erişim imkânı gerçekleşmiş ve dünyadaki finansal sistemler üzerinde hiçbir zaman örneğine rastlanmayan ekonomik krizler ve çalkantılar gerçekleşmiştir.

Ülkemizde de finansal serbestleşmenin etkisiyle hem olumlu hem de olumsuz birçok durum meydana gelmiştir. Yeni sermaye kaynaklarının ülkeye girişiyle kredi işlemlerinin hacmi büyümekle birlikte, 1994 krizi ve akabinde 2000-2001 krizlerinde birçok bankanın battığı görülmüş ve tüm ekonomik birimler bu nedenle olumsuz etkilenmiştir. Örnek olarak; bankaların batması sonucunda mevduat sahiplerinin birikimlerini kaybetmesi, kredi mekanizmasının bozulması sonucu bankaların piyasaya kredi temin etmemeleri, kredi maliyetlerinin aşırı artması, devletin bu bankaları kurtarmak zorunda kalması nedeniyle devlet kaynaklarının heba olması, ekonomik büyümenin durması gibi durumlar sayılabilir.

Bankaların güçlü veya zayıf olarak değerlendirilmesini sağlayan çeşitli uluslararası göstergeler bulunmaktadır. Bu göstergeler karlılık, likidite, çeşitli riskler, sermaye yeterlilik oranları, kredi değerlilikleri, uluslararasılaşma seviyeleri, aktif büyüklükleri, aktiflerinin yapısı, mevduat tabanı vb. birçok kıstasla birbirleriyle kıyaslanmakta ve/veya yatırımcılar ve devlet kurumlarınca izlenebilmektedir. Günümüzde karlılık üzerine yapılan çalışmalar incelendiğinde özellikle literatürde daha çok özkaynak ve aktif karlılığının çalışmalarda kullanıldığı anlaşılmaktadır. Özellikle merkez bankalarının ve denetim

otoritelerinin aktif ve özkaynak karlılıklarını çok sık kullanmaları arařtırmacıları bu iki kıstasa yöneltmiştir.

Aktif karlılığı, bankanın varlıklarının tamamını kullanarak elde edebileceđi karlılık oranını ifade etmektedir. Özkaynak karlılığı ise, net karın özkaynaklara oranıdır. Özkaynakların ortaklık payını temsil ettiđi düşünülürse, öznaynak karlılığının ortakların net kardan elde ettiđi getiri payını temsil ettiđi düşünülür. Karlılık ölçümünde bu iki ölçütün ikisi de oldukça önem taşımaktadır.

Banka karlılığının hesaplanmasında birçok uzman genellikle Lineer Regresyon Analizi Lojistik Regresyon Analizi, Tam Logaritmik Regresyon, Çoklu Ayırma Analizi, Çoklu Regresyon Analizi gibi istatistiksel modeller kullanmaktadır. Bu teknikler sayesinde bankaların karlılık durumlarını tahmin etmek mümkün olmasına rağmen, çođu zaman bu tahminler doğruyu tam olarak yansıtamamaktadır. Bu durum çok daha tutarlı ve etkili modellerin kullanılmasına gerek duyulmasına neden olmuştur.

Bu noktada bilgisayar algoritmalarının bankacılık alanında oldukça öne çıkmasına ve gelecekte de bankaların hem karar almalarında hem de strateji üretmelerinde yapay zekânın rolünün oldukça önem taşıyacağı düşünülmektedir. Bu kapsamda arařtırmalar yapay zekânın finansal alanlarda uygulanabilirliğine dönük olarak gerçekleştirilmektedir.

Günümüzde karlılık ölçümlerinde farklı teknikler kullanılmaya başlanmıştır. Özellikle teknolojilerin gelişmesiyle yapay sinir ađları (YSA), genetik algoritmalar (GA), bulanık mantık (BM) gibi yapay zekâ teknikleri en çok tercih edilen teknikler olmaya başlamıştır.

Bu kapsamda tezde mevduat bankalarına yapay zekânın uygulanabilirliğine dönük bir çalışma yapmak amacıyla planlama yapılmıştır. Bu amaçla ilk olarak mevduat bankalarından uygun olanların seçilmesi amacıyla aktif büyüklükleri incelenmiş ve en uygun adaylar çalışmaya katılmıştır. Seçilen bankaların verileri çeşitli veritabanlarından (BDDK, TBB, TCMB ve TÜİK) alınarak 2006 son çeyreğinden, 2017 son çeyreğine kadar olan 3'er aylık dönemler olarak kullanılmıştır. Bu bankaların aktif karlılıkları yapay sinir ađları optimizasyon algoritmaları ile istatistiksel yöntemlerden Lineer Regresyon yöntemi Phyton programlama dili üzerinde kodlama yapılarak bankaların karşılařtırılabilir performans analizlerinin kıyaslamaları gerçekleştirilmiştir. Bu bankalar bankacılık sektörünün 2019 Aralık itibariyle yaklaşık % 75'ine hâkim konumdadır.



Bu 6rneklem ierisinde seilen baėımlı deėiŐken, baėımsız deėiŐkenler ve dıŐsal deėiŐkenler olarak 16 deėiŐken kullanılmıŐtır. Bu deėiŐkenler tezin Deney, G6zlem ve Analiz kısmında ayrıntılı bir Őekilde anlatılmıŐtır.

İlk b6l6mde yapay sinir aėları konusu detaylarıyla anlatılmıŐ ve yapay sinir aėları optimizasyon algoritmaları anlatılarak ilk b6l6m tamamlanmıŐtır.

İkinci b6l6mde literat6rdeki yapay sinir aėları ile banka performansı hesaplanmış ve detaylı Őekilde incelenmiŐtir.

66nc6 b6l6mde tezin deney, g6zlem ve analiz kısmı gerekleŐmiŐtir. Burada alıŐma ile ilgili banka bilgisi, veri k6mesi ve oluŐturulan model tanıtılarak deneylerde bulunan sonular ve karŐılaŐtırmalar g6rsellerle sunulup aıklanmıŐtır.

Son b6l6m tartıŐma ve sonu b6l6m6d6r. Genel itibariyle tezde anlatılanlar toparlanarak tez tamamlanmıŐtır.

## 2. LİTERATÜR TARAMASI

Literatürde bankaların karlılığı konusunda ulusal ve uluslararası alanda birçok çalışma farklı yöntemler kullanılarak gerçekleştirilmiştir. Kimi çalışmalar bir ülke verileri kullanılarak gerçekleştirilmişken, kimi çalışmalar birden fazla ülkenin verileri ile yapılmıştır. Genellikle banka karlılığı konusunda kullanılan istatistiksel yöntemler ve yapay sinir ağları başlıca yöntemlerdir. Literatür taraması konusu bu bağlamda ulusal ve uluslararası literatür olarak incelenecektir. Bu kapsamda literatür taraması için ilk olarak bankacılık alanında yapılan çalışmalar incelenmiş, daha sonrasında ise diğer alanlarda yapılan çalışmalar incelenmiştir.

**Ömürbek vd., [1]** çalışmalarında aktifleri açısından büyük olan bankaların kârlılık tahminini gerçekleştirmeyi amaçlamışlardır. Bu doğrultuda YSA yöntemini uygulamışlardır. Bağımlı değişken olarak özkaynak kârlılığını, bağımsız değişken olarak da 10 değişken kullanmışlardır. Yapılan analize göre bankaların tahmini özkaynak karlılık değerleri ile gerçek özkaynak karlılık değerlerinin yakın olduğunu belirlemişlerdir. Araştırılan dönemler 2017 Eylül-2018 Haziran aralığını kapsamıştır. Yapılan araştırmalar neticesinde kârlılık öngörülerinin, yapay sinir ağı modeli ile gerçekleştirilebileceği belirlenmiştir. Elde edilen test sonuçları ile fiili rakamlar arasında ortalama mutlak yüzdelik hata (MAPE) değerleri hesaplanarak bu değerler “çok iyi” ve “iyi” olarak sınıflandırılmıştır. Sınıflandırılan bu MAPE değerleri sonraki dönemlerde kârlılık öngörülerini için kullanılmıştır.

**Sönmez vd., [2]** çalışmalarında amaç olarak ROA ve ROE üzerinde etkili çeşitli değişkenleri kullanarak Türkiye’deki 24 mevduat bankasının karlılığını yapay sinir ağları ile hesaplamaya çalışmışlardır. Ayrıca klasik regresyon yöntemleri ile de ROA ve ROE değerleri tahmini gerçekleştirilmiştir. Tahmin sonuçları karşılaştırıldığında yapay sinir ağı sonuçlarının, bankaların gerçek ROA ve ROE değerlerine daha yakın olduğu görülmüştür.

**Şerbetli [3]** çalışmasında, öngörü modellemesi olarak ileri beslemeli yapay sinir ağları kullanılmıştır. Araştırma makroekonomik değişkenlerin banka karlılığı üzerindeki etki derecelerinin ne olduğu kapsamında ele alınmıştır. Çalışmada makroekonomik değişkenlerden farklı birçok değişken bir araya getirilip, eklenen her bir değişkenin doğruluk oranlarına etki edebileceği göz önüne alınmıştır. YSA’nın genelleme veya tahmin

becerilerini tam anlamıyla kullanabilmeleri sayesinde örneklem kümesi sınırlı olan veri setlerinde de başarılı olduğunu göstermişlerdir.

**Al-Osaimy [4]** çalışmasında İslami bankaların performansını tahmin etmek için sinir ağlarını kullanmıştır. 1991 - 1993 dönemi için 26 İslami bankanın veri kümesi örneğini kullanarak 7 finansal oran oluşturmuştur. Kohonen sinir ağını ilk kez İslami bankaların performansını ölçmek için kullanarak yüksek ve düşük performans grupları belirlemiştir. Bu sonuçları kullanarak, İslami bankaların performansını tahmin etmek için üç katmanlı bir geri yayılım sinir ağı oluşturmuştur. Yapay sinir ağının tahmin sonuçları ile gerçek banka değerleri arasında yüksek korelasyon görülmüştür.

**Bakar ve Tahir, [5]** banka performansını tahmin etmede çoklu doğrusal regresyon tekniğini ve ileri yönlü yapay sinir ağını kullanmış ve sonuçlarını karşılaştırmışlardır. Araştırmada 2001-2006 dönemine ait 13 bankanın verileri kullanılmıştır. Çalışmalarında, yapay sinir ağının banka performansını tahmin etmede daha güçlü bir yöntem olduğunu ispatlamışlardır.

**Olweny ve Shipho, [6]** çalışmalarında iki amaç bulunmaktadır. İlk amaç, bankaya özgü faktörlerin etkilerini belirlemek ve değerlendirmektir. Bu amaçla Kenya'daki ticari bankaların karlılığında sermaye yeterliliği, varlık kalitesi, likidite, operasyonel maliyet etkinliği ve gelir çeşitlendirmesi değişkenleri kullanılmıştır. İkinci amaç ise, piyasa yapısı faktörlerinin etkilerini belirlemek ve değerlendirmektir. Bu bağlamda Kenya'daki ticari bankaların karlılığına ilişkin yabancı mülkiyeti ve piyasa yoğunlaşması araştırılmıştır. Bu çalışma, yukarıdaki hedefleri gerçekleştirmek için çoklu doğal regresyon yöntemini kullanmıştır. Analiz, tüm bankaya özgü faktörlerin karlılık üzerinde istatistiksel olarak önemli bir etkiye sahip olduğunu, ancak piyasa faktörlerinin hiçbirinin önemli bir etkisinin olmadığını gösterilmiştir.

**Sapuan vd., [7]** çalışmasında aktif karlılığının tahmininde hem istatistiksel yöntemi hem de yapay sinir ağlarını kullanmıştır. Sonuç olarak yapay sinir ağları ile bulunan tahminlerin gerçek değerlere daha yakın olduğunu tespit etmişlerdir.

Tüm bu çalışmalar incelendikten sonra tezin amacı, bir dizi finansal parametre verildiğinde, YSA hesaplama yöntemlerini kullanarak bankaların özkaynak getirilerini tahmin etmek olarak belirlenmiştir. Yukarıda yararlanılan çalışmalarda YSA ve regresyon metotları kullanılmıştır. Özellikle doğrusal regresyon uygulanarak gerçekleştirilmiş

çalışmalarda hesaplanan özkaynak karlılıkları tahminleri ile gerçek özkaynak karlılıkları arasında doğruluk değerleri azalmaktadır. Ayrıca kullanılan değişkenlerin sayısı arttıkça oluşturulan modellerin doğruluk değerleri de düşebilmektedir [27]. Buna karşın YSA kullanımı ile bulunan doğruluk değerleri ve güvenilirliği artmaktadır. YSA kullanıldığında modeldeki değişken sayısı arttırılsa bile tahmin değerleri ile gerçek değerler arasındaki doğruluk değerlerinin çok fazla değişmediği belirlenmiştir.

Bu bağlamda tezdeki çalışmadan elde edilecek bulgular sonucunda YSA'ların diğer yöntemlere göre daha başarılı tahmin yeteneği olduğu ortaya koyulmaktadır. Modelimiz, literatüre uyumlu bir şekilde oluşturulduğundan özkaynak karlılığı tahminleri gerçek özkaynak karlılığı değerlerine yakındır. Tezde kullanılan optimizasyon algoritmaları ile desteklenmiş YSA modeli ülkemiz literatürüne yenilik katmaktadır.

Ortaya koyulan sonuçlar incelendiğinde ağıın bulduğu bankaların özkaynak karlılığı tahmin sonuçları ile bankaların gerçek özkaynak karlılık değerleri arasında doğruluk değeri oldukça yüksektir.

Tezin literatürdeki diğer çalışmalara göre farklı yönleri aşağıda maddeler halinde sıralanmıştır.

- Diğer çalışmalar tek bir yapay sinir ağı modeli kullanmışken, tezin yapay sinir ağı modeli 3 optimizasyon algoritması ile güçlendirilmiştir.
- Diğer çalışmalarda yer alan yapay sinir ağları modelleri daha az gizli katman sayısı kullanmışken, tezde daha fazla gizli katman sayısı kullanılmıştır.
- Veri kümemizde seçilen değişkenler, diğer çalışmalarda bulunan değişkenlerden daha fazladır.
- Diğer çalışmalara göre tezde daha geniş bir zaman aralığı incelenmiştir.
- Ülkemizde yapay sinir ağı optimizasyon modelleri ile regresyon yöntemleri kıyaslaması finans alanında ilk kez gerçekleştirilmiştir.

Sonuç olarak, tezde kullanılan model ile bankalara ait karlılıkları tahmin etme işlevi başarılı olmuştur. Bu sayede optimizasyon algoritmaları uygulanmış yapay sinir ağları modeli literatüre yenilik katarak katkıda bulunacağı düşünülmektedir. Ayrıca ülkemiz akademik literatüründe bankacılık sektörüne ilişkin ilk kez ADAM, RMSprop ve Stokastik Gradyan İnişi yöntemleri bir çalışmada kullanılmıştır.

Optimizasyon algoritmaları (RMSprop, ADAM, SGD, NADAM, v.b.) bankacılık alanında olduđu gibi birçok farklı alanda da kullanılabilir. Yapılan çalışmalarda bu algoritmalar; evrimsel sinir ađlarını eđitmede, Hizmet Engelleme (DoS) saldırılarını tespit etmede, parametre güncellemede ve derin sinir ađlarında kullanılmaktadır. Literatürdeki çalışmalar incelendiğinde; çeşitli problem çözümleri için farklı optimizasyon algoritmaları kullanıldığı görülmektedir. Bazı çalışmalarda var olan algoritmalar karşılaştırılmakta iken; bazıları da var olan algoritmaları bir araya getirerek yeni algoritmaların oluşturulmasını sağlamıştır.

Literatürde farklı alanlarda kullanılan çeşitli optimize edici yöntemler için başka yöntemlerin de başarılı sonuçlar verdiği görülmektedir. Bu literatür örneklerinin çalışmamızla yöntem yönünden benzerlikleri, çalışmayı gerçekleştirenler, ortaya çıkan sonuçları ile çalışmaların açıklamaları tablo şeklinde anlatılacaktır.

Tablo 2. 1. Literatürün farklı alanlarında optimizasyon algoritmalarının kullanımı

No	Araştırmacılar	Araştırma Alanı	Algoritmalar	Açıklamalar	Sonuçlar
1	Vani ve Rao [8]	CNN	SGD, RMSprop, ADAM, Adamax, AdaGrad, AdaDelta ve NADAM	Yazarlar, SGD, RMSprop, ADAM, Adamax, AdaGrad, AdaDelta ve NADAM algoritmalarını Indian Pines veri kümesine uygulamışlardır. Bu optimizasyon algoritmaları kullanarak CNN'deki hatayı minimize etmişlerdir.	Adamax optimizasyon yöntemi ile %99,58 başarı elde etmişlerdir.
2	Rajakumaran vd. [9]	Hizmet Engelleme (DoS) saldırılarının tespiti	SGD	Araştırmacılar çok değişkenli lineer regresyon ve Ortalama Kare Hatası (MSE) fonksiyonunu kullanarak hizmet engelleme (denial of service (DoS)) saldırılarının tespit etmeye çalışmışlardır.	Araştırma SGD optimizasyon yöntemi ile gerçekleştirilmiştir. DoS saldırılarındaki tespit oranı %99.7 olarak bulunmuştur. Diğer taraftan sistemdeki hata oranlarını düşürmeyi başarmışlardır.
3	Dogo vd. [10]	CNN	SGD, RMSprop, ADAM, AdaGrad, AdaDelta, Adamax ve NADAM	Yazarlar, her bir optimize edicinin performansını, elde edilen doğruluk yüzdesine göre değerlendirmiştir. Her bir optimize edicinin performans kalitesini belirlemek için, yazarlar üç gizli katman ve her katmanda 64 nöron içeren bir CNN mimarisi önermiştir.	Analiz sonucunda NADAM yöntemi ile en iyi sonucu bulmuşlardır, AdaDelta yöntemiyle ise en kötü sonucu elde etmişlerdir.
4	Yazan ve Talu [11]	Parametre Güncelleme	Momentumlu SGD, SGD, AdaGrad, RMSprop, AdaDelta ve ADAM	Yazarlar tarafından yapılan parametre güncelleme işlemlerinde; yakınsama süresi, yineleme sayısı ve belirli test fonksiyonu değerleri kullanılarak araştırma gerçekleştirilmiştir.	AdaDelta, deneysel bulgulara göre hızlı yakınsama açısından diğer optimize edicilerden üstün bir performansa sahip olduğu gösterilmiştir.
5	Papamarikios [12]	Lojistik ve softmax regresyon ile tanımlama	Gradyan inişi, Stokastik gradyan inişi, Yarı stokastik gradyan inişi ve Stokastik ortalama iniş	Yazar tarafından gerçekleştirilen iki deneyde stokastik gradyan inişi genellikle gradyan inişinden daha iyi performans göstermiştir. Ayrıca kullanılan veri kümesinde doğruluk yüzdesi, eğitim ve yakınsama süreleri performans bakımından gradyan inişi (GD) ve SGD için karşılaştırma yapılmıştır.	İki deneyde SGD genellikle gradyan inişinden daha iyi performans göstermiştir.

6	Shalev-Shwartz vd. [13]	CNN	Gradyan temelli algoritmalar	Yapılan deneyler sonucunda arařtırmacılar, derin öğrenmedeki gradyan iniři performans başarısızlıklarının kaynaklarını göstermişler ve bu hataları gradyan iniřlerin sinyal / gürültü oranı gibi aktivasyonlarla ilişkili olduğunu tespit etmişlerdir.	Deneyler karşılařtırmadan çok arařtırmaya dönük olarak gerçekleştirilmiştir.
7	Lecun vd. [14]	CNN	AdaDelta, ADAM	Yazarlar, ortalama kare hatasına (MSE) dayalı olarak, her bir CNN mimarisi üzerinde iki optimize edicinin performansını deęerlendirmişlerdir.	ADAM optimize edici, sıę aę mimarisi ve LeNet mimarisi için en düşük ortalama kare hatasının bulunmasını gerçekleřtirmiştir. AdaDelta, AlexNet mimarisinde en düşük ortalama kare hatasını yakalamıştır. Yazarlar ayrıca, ADAM optimize edicinin en iyi performansı elde ettięi sonucuna varmışlardır.
8	Jangid ve Srivastava [15]	CNN	ADAM, Adamax, RMSprop	Her optimize ediciyi deęerlendirmek için, yazarlar üç evriřimli katman ve 1000 nöronlu yoğun bir katman içeren bir CNN mimarisi tanıtmışlardır.	Bu mimari için RMSprop en iyi doęruluęu elde etmiştir.
9	Swastika vd. [16]	CNN	ADAM, AdaDelta ve SGD	Yazarlar, her optimize ediciyi deęerlendirmek için üç CNN mimarisi olan sıę bir aę, LeNet ve MiniVGGNet olmak üzere 3 aę kullanmışlardır.	İyileřtiriciler doęruluklarına göre deęerlendirilerek hangi optimize edicinin en iyi sonucu verdięi ortaya koyulmaya çalışılmıştır. AdaDelta optimize edicinin MiniVGGNet mimarisi için en iyisi olduęu yazarlar tarafından açıklanmıştır.
10	Zaheer ve Shaziya [17]	Derin Öğrenme	SGD, Nesterov momentum, ADAM, RMSprop, AdaGrad, AdaDelta	Deneylerde mnist, fashionmnist, cifar10 ve cifar100 olmak üzere dört veri kümesi seçilmiştir. Kullanılan veri setleri ile farklı epok deęerlerinde ilgili optimize ediciler ile çalıştırılmıştır.	En yüksek test sonuçları mnist, fashionmnist, cifar10 ve cifar100 için ADAM ile sırasıyla %98.26, %98.53, %98.55, %98.42 olmuştur. Analiz sonuçlarına göre ADAM algoritmasının test aşamasında daha başarılı bir performans gösterdięi görülmüştür.

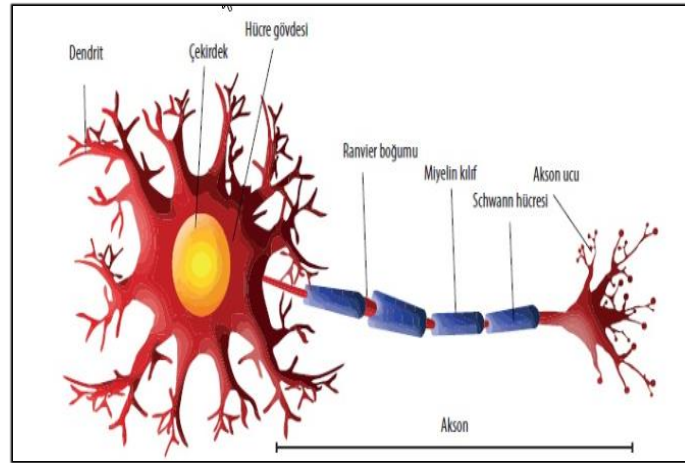
### 3. YAPAY SİNİR AĞLARI VE OPTİMİZASYON ALGORİTMALARI

#### 3.1. Yapay Sinir Ağlarına Giriş

1943 yılında insan beyninin modellenmesi için başlayan çalışmalar, 1980'li yıllarda biyolojik sinir yapısının bire bir oluşturulmaya çalışılmasıyla yapay sinir ağları modeli geliştirilmiştir. Her ne kadar yapı ve kapasite açısından biyolojik ve yapay sinir ağları arasında bazı farklılıklar olsa da, birçok yönden de benzerlikler taşımaktadır. Yapay sinir ağları, deneyime dayalı bir öğrenme süreci sayesinde çevreden bilgi toplayan ve bu bilgiyi işleyerek sonuçlar üreten işlemcilerdir [18, 19].

Tipik bir biyolojik sinir ağı soma, akson ve dendrit denilen kısımlardan oluşan ve her bir parçanın bilgilerin iletiminde rol oynadığı bir ağ yapısıdır. Soma, hücrenin denetimi ve yönetiminde, dendritler bilginin nöronlardan alınıp başka nöronlara iletilmesinde, aksonlar hücre gövdesindeki bilginin diğer nöronların dendritlerine taşınmasında görev üstlenirler.

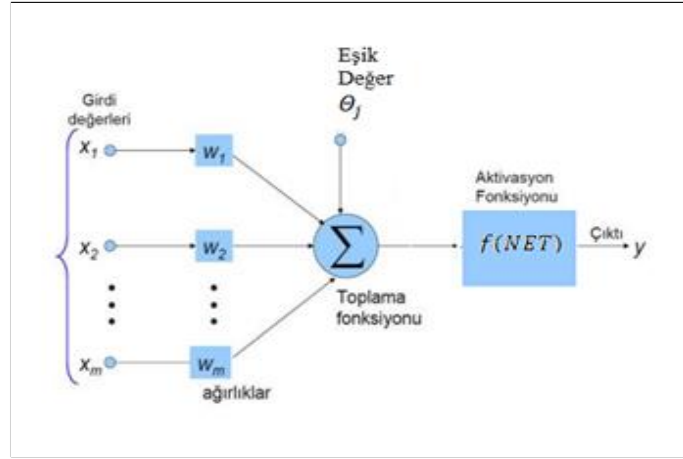
Biyolojik bir sinir hücresinin yapısı Şekil 3.1.'de gösterilmiştir.



Şekil 3. 1. Biyolojik sinir hücresi [20]

Yapay sinir hücresinin yapısı Şekil 3.2.'de gösterilmiştir.





Şekil 3. 2. Yapay sinir hücresi [3]

Yapay nöronlarda ise ister çok katmanlı olsun ister basit yapıda olsun aşağıdaki bileşenler bulunmaktadır [20, 21].

- **Girdi:** Dışarıdan ağa girilen değerler  $(x_1, x_2, \dots, x_m)$  olarak ifade edilmektedir. Girdiler  $(x)$ ,  $m$  elemanlı sütun vektörü olarak gösterilir.

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_m \end{bmatrix}$$

- **Ağırlık:** Dışarıdan ağa girilen değerlerin önemini ve hücredeki etkisini gösteren sayılar ağırlık  $(w_1, w_2, \dots, w_m)$  olarak ifade edilmektedir. Ağırlıklar,  $m$  elemanlı satır vektörü olarak düzenlenir.

$$W = [w_1, w_2, \dots, w_m]$$

- **Toplama Fonksiyonu:** Toplama fonksiyonu bileşeninde, yapay sinir hücresine giren tüm girdiler, ilgili ağırlıkları ile çarpılır. Bulunan tüm değerler toplanır ve böylece net girdi hesaplanır. Net girdi değerine eşik değeri  $(\theta)$  eklenerek NET değeri elde edilir. Toplam fonksiyonu sonucunda elde edilen NET değeri aktivasyon fonksiyonuna iletilir.
- **Aktivasyon Fonksiyonu:** Toplam fonksiyonundan iletilen NET değeri işleyerek yapay sinir hücresinin çıktısını oluşturan fonksiyondur.
- **Çıktı:** Aktivasyon fonksiyonu sonucunda üretilen değerlere çıktı denir.

Çıktı fonksiyonun belirlenmesi işlemleri izleyen adımlarla yapılmaktadır.

$$NET = \sum_{i=1}^m w_i x_i + \theta \quad (3.1)$$

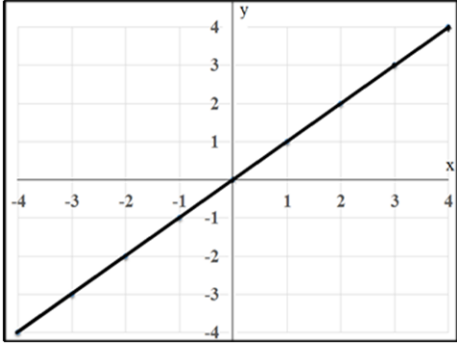
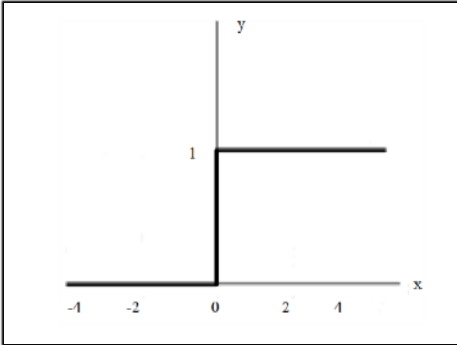
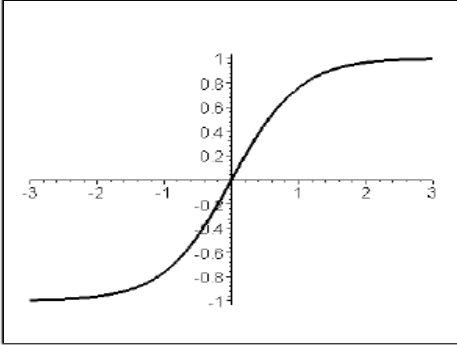
Buradak  $w$  ağırlıklar vektörünü,  $x$  ise girdi matrisini ifade etmektedir.  $m$  girdi sayısı olmak üzere;  $w = (w_1, w_2, \dots, w_m)$  ve  $x = (x_1, x_2, \dots, x_m)$  şeklinde yazılabilir. Formül haline getirilirse aşağıdaki çıktı fonksiyonu oluşmaktadır:

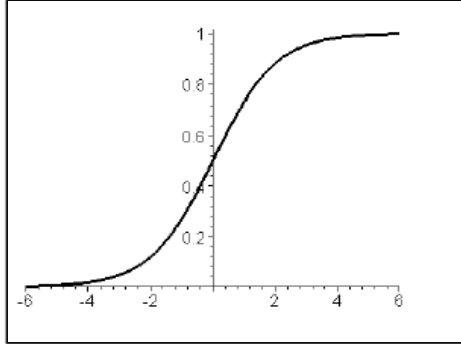
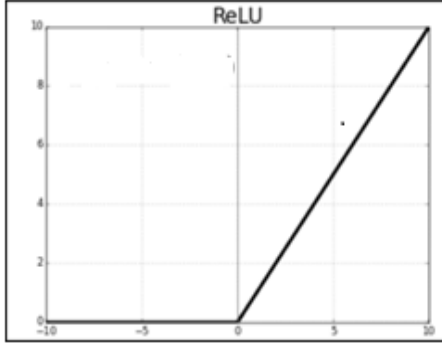
$$f(NE T) = y = f\left(\sum_{i=1}^m w_i x_i + \theta\right) \quad (3.2)$$

Burada  $f(NE T)$  olarak tanımlanmış değer aktivasyon fonksiyonu olarak literatürde tanımlanmaktadır.

Kullanılan modelin çeşidine göre değişik aktivasyon fonksiyonları kullanılmaktadır. Farklı aktivasyon fonksiyonları Tablo 3.1.'de gösterilmiştir.

Tablo 3. 1. Aktivasyon fonksiyonları [22]

Aktivasyon Fonksiyonu	Şekil	Formül
Doğrusal Fonksiyon		$f(NE T) = NE T$
Eşik Fonksiyonu		$f(NE T) = \begin{cases} 1, & NE T \geq 0 \\ 0, & NE T < 0 \end{cases}$
Hiperbolik Tanjant Fonksiyonu		$f(NE T) = \frac{e^{NE T} + e^{-NE T}}{e^{NE T} - e^{-NE T}}$

Sigmoid Fonksiyonu		$f(NET) = \frac{1}{1 + e^{-NET}}$
ReLU Fronksiyonu		$f(NET) = \max(0, NET)$ $\begin{cases} NET, & \text{eğer } NET \geq 0 \\ 0, & \text{eğer } NET < 0 \end{cases}$

Biyolojik sinir ağı ile yapay sinir ağı arasında çeşitli yönlerden benzerlikler bulunmaktadır.

Biyolojik sinir ağı ve yapay sinir ağı arasındaki sözkonusu benzerlikler Tablo 3.2.'de gösterilmiştir.

Tablo 3. 2. Yapay sinir ağı ve biyolojik sinir ağı arasında benzerlikler [23]

Biyolojik Sinir Ağı	Yapay Sinir Ağı
Stimulus	Girdi
Reseptörler	Girdi Katmanı
Sinir Ağ	İşlem Katmanı
Nöron	İşlem Elemanları
Efektörler	Çıktı Katmanı
Cevap	Çıktı
Sinapslar	Ağırlıklar
Dentridler	Özet İşlevi
Hücre Gövdesi	Aktivasyon Fonksiyonu
Akson	Çıktı
Eşik Değeri	Yanlılık

Biyolojik ve yapay sinir ağlarının farklılıkları ise aşağıda liste halinde gösterilmiştir [18]:

- Biyolojik sinirler hücresel yapıda bulunmaktayken, yapay sinirler birim yapıda bulunmaktadır.

- Biyolojik sinirlerde sinaps denilen sinir eklemi bulunmaktayken, yapay sinirlerde ara bağlantılı ağırlıklar bulunmaktadır.
- Biyolojik sinirler destekleyici girişe sahipken; yapay sinirler pozitif ara bağlantı ağırlığına sahiptir.
- Biyolojik sinirlerde hücre fiziği ile sınırlı aktiflenme varken; yapay sinirler de işlemci karakteristiğiyle sınırlı aktifleme aralığı bulunmaktadır.

### **3.2. Yapay Sinir Ağları Yapısı**

YSA girdi katmanı, gizli katman ve çıktı katmanı olmak üzere 3 katmandan oluşmaktadır [18].

#### **3.2.1. Girdi katmanı**

YSA'nın öğrenme sürecini gerçekleştirmesi için ağa sokulan girdilerin sisteme ilk giriş yaptığı katmandır. Yapay sinir ağı için iş akışının başlangıcıdır. Giriş katmanında hesaplama yapılmamakta sadece bilgiler gizli katmanlara aktarılmaktadır.

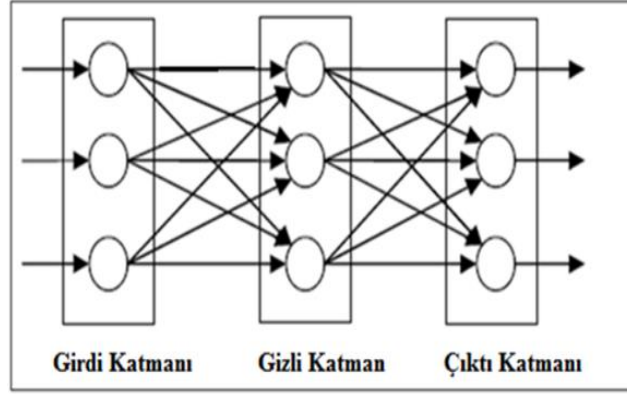
#### **3.2.2. Gizli katman**

Giriş katmanı ile çıktı katmanı arasında yer alan katmandır. Gizli katmanın, YSA'nın çevresi ile doğrudan bir bağlantısı yoktur. Gizli katmanda giriş katmanından alınan bilgiler işlenerek çıktı katmanına aktarılır. Gizli katmanlarda ileri ve geri yönlü işlemler gerçekleşebilir. Bir YSA'nın gizli katman sayısı ve her bir gizli katmanın sahip olduğu nöronların sayısı farklı durumlar için değişebilir. Gizli katmanların sayısı yanlış belirlendiği takdirde hesaplama süresinin artması, hesaplama hatalarının gerçekleşmesi gibi çeşitli olumsuz sonuçlar meydana gelebilmektedir.

#### **3.2.3. Çıktı katmanı**

Yapay sinir ağına giriş yapan girdilerden sonuçların üretildiği katmandır. Çıktı katmanı hesaplamalardan ve ağdan dış dünyaya doğru bilgi aktarımından sorumludur.

YSA'da bulunan tüm katmanlar Şekil 3.3.'te gösterilmiştir.



Şekil 3. 3. Yapay sinir ağları katmanları [24]

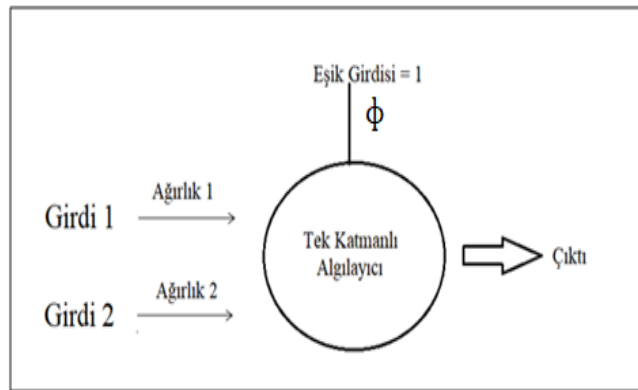
### 3.3. Yapay Sinir Ağ Modelleri

YSA modelleri genel olarak aşağıda sırasıyla başlıklar halinde sunulmaktadır.

#### 3.3.1. Tek katmanlı algılayıcılar

Tek katmanlı algılayıcılar, YSA tarafından öğrenmesi istenen olayların, girdi ve çıktuları arasındaki ilişki doğrusal ilişki olduğu durumlarda kullanılmıştır. Bu ağlarda ara katmanlar bulunmamaktadır. Girdi ve çıktı katmanlarında ise en az bir nöron bulunmaktadır. Tek katmanlı algılayıcılar, literatürde ayrıca basit sinir ağı modeli olarak da bilinmektedir.

Tek katmanlı bir algılayıcı örneği Şekil 3.4.' te gösterilmiştir.



Şekil 3. 4. Tek katmanlı algılayıcı örneği [25]

Burada değeri daima 1'e eşit olan ve ağıın çıktısının 0 olmasını önleyen bir eşik girdisi bulunmaktadır. Ağıın çıktısı ağırlıklar ile çarpılan girdi değerlerinin toplam değerine

eşik değerin eklenmesi ile bulunur. Çıktı fonksiyonu -1 ila 1 arasında değışen değeri alabilmektedir.

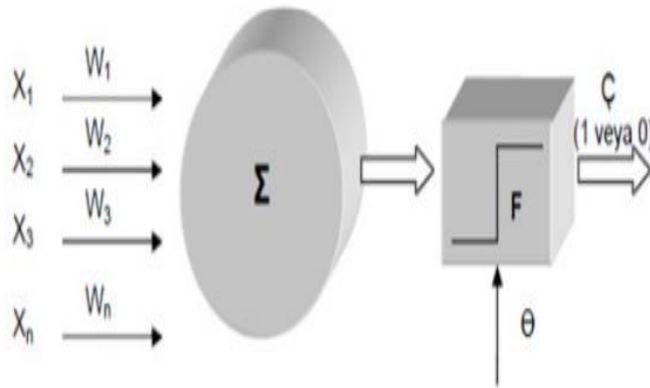
$$\zeta = f\left(\sum_{1}^n w_i x_i + \phi\right) \quad (3.3)$$

Eşitlik (3.3.)'te  $x_i$  ağın girdilerini tanımlar. Aynı eşitlikte  $w_i$  değeri bu girdilere karşılık gelen ağırlık değeri ifade eder.  $\phi$  ise eşik değeri. Bu işlemlerin fomülüze edilmiş hali ise Eşitlik (3.3.)' te gösterilmiştir.

### 3.3.2. Basit algılayıcılar

F. Rosenblatt tarafından 1950'lerde önerilmiş algılayıcılar, öğrenme algoritması olarak diğer gelişmiş öğrenme yöntemleri için temel oluşturmaktadır. Burada önemli olan nokta girdiler ve onlara karşılık gelen ağırlık değeri ile bu çarpımların toplamının eşik değeriyle karşılaştırılmasıdır. Çarpımların toplam değeri eşik değeriyle karşılaştırıldığında eşik değeri büyükse çıkış değeri üretilmektedir. Yapay sinir ağına öğrenmesi bu ağırlıkların değeri değiştirilmesi ile gerçekleşmektedir.

Basit algılayıcı modeli Şekil 3.5.'te gösterilmiştir.



Şekil 3. 5. Basit algılayıcı modeli [26]

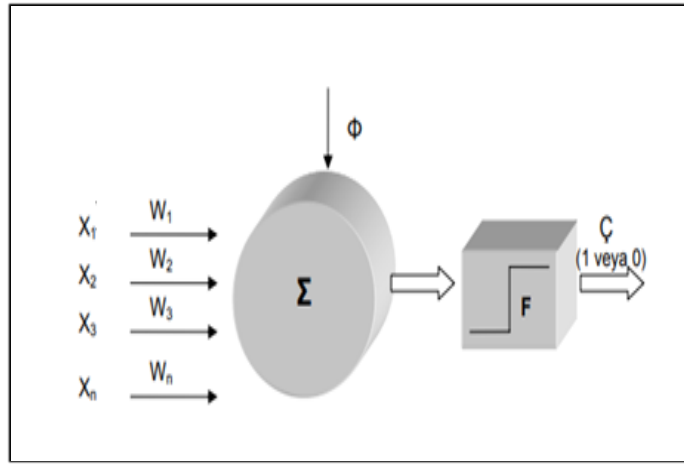
### 3.3.3. Uyarlanabilir ve çoklu uyarlanabilir doğrusal eleman

Widrow ve Hoff tarafından geliştirilmiş olan bu modeller, ilk yapay sinir ağı modellerindedir. ADALINE Modeli, YSA'nın hesapladığı çıktı değeriyle en az hata

payıyla tahmin edebilmesi için girdi ağırlıklarının değiştirilme prensibi olan delta kuralına ve en küçük ortalamaların karesi yöntemini benimser.

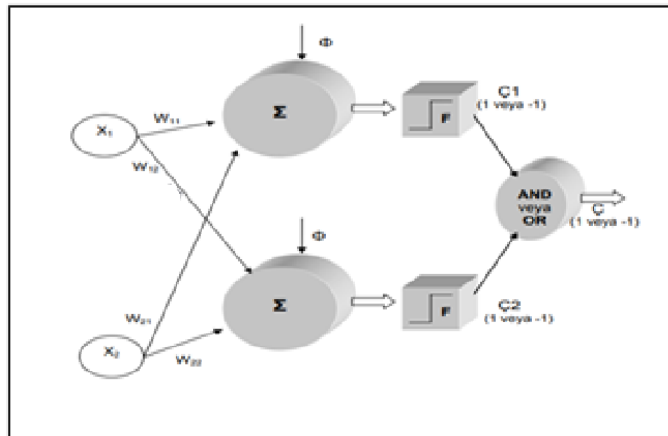
MADALINE Modeli ise birçok ADALINE ünitesinin bir araya gelmesiyle oluşan bir yapay sinir ağı modelidir. Genellikle iki katmandan meydana gelir. Her katmanda farklı sayılarda ADALINE ünitesi bulunmaktadır. Ağın çıkışı tıpkı ADALINE Modeli'nin çıkışında olduğu gibi -1 ve +1 değerleri ile ifade edilir.

ADALINE modeli Şekil 3.6.'da gösterilmiştir.



Şekil 3. 6. ADALINE modeli [27]

İki ADALINE ağından oluşmuş MADALINE modeli Şekil 3.7.'de gösterilmiştir.

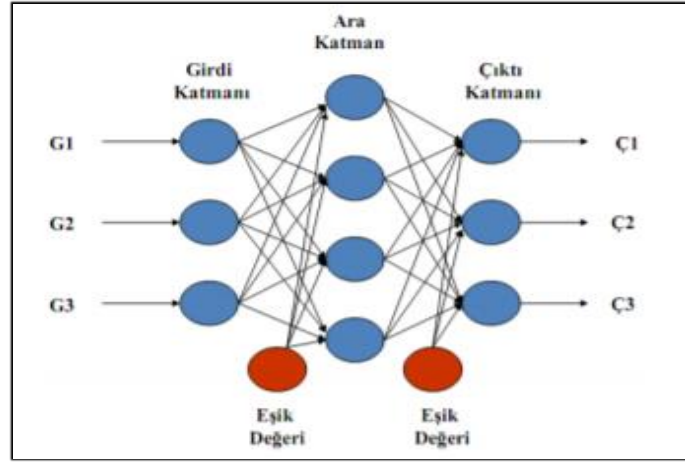


Şekil 3. 7. İki ADALINE ağından oluşmuş MADALINE modeli [27]

### 3.3.4. Çok katmanlı algılayıcılar

Bir yapay sinir ağının öğrenmesi istenen olaylarda girdi ve çıktılar arasında doğrusal olmayan ilişkiler bulunduğunda tek katmanlı algılayıcılar öğrenme için yetersiz kaldığından çok katmanlı algılayıcılar geliştirilmiştir. Temel amaçları ağa giren girdilere en uygun çıktıları bulmaktır. Ağdaki katmanlarda bulunan nöronlar bu nedenle birbirlerine bağlı bir yapıdadır.

Çok katmanlı algılayıcı modeli Şekil 3.8.'de gösterilmiştir.



Şekil 3. 8. Çok katmanlı algılayıcı modeli [28]

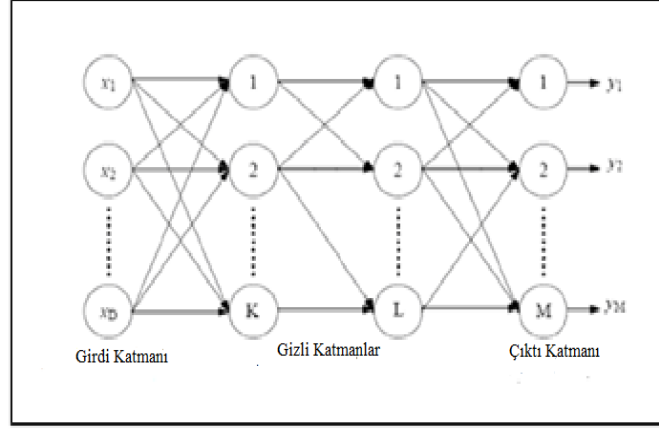
Bu model, ileri beslemeli yapay sinir ağları veya geri beslemeli yapay sinir ağları şeklinde çalışmaktadır [28].

#### 3.3.4.1. İleri beslemeli yapay sinir ağları

İleri beslemeli yapay sinir ağlarında nöronlar girdi katmanından çıktı katmanına doğru sıralı katmanlardan oluşmaktadır. Bir katman, sadece bir sonraki katmanla bağlantılıdır. Yapay sinir ağına giriş yapan girdiler hiçbir şekilde değişmeden gizli katmandaki nöronlara aktarılır. Gizli katmanda işleme tabi tutulan girdiler, çıktı katmanına ulaştığında çevreye verilir. İleri beslemeli ağlarda geri dönüşler bulunmadığı için üretilen çıktılar hızlı bir şekilde oluşmaktadır [29].

İleri beslemeli yapay sinir ağlarının yapısı Şekil 3.9.'da gösterilmiştir [21].



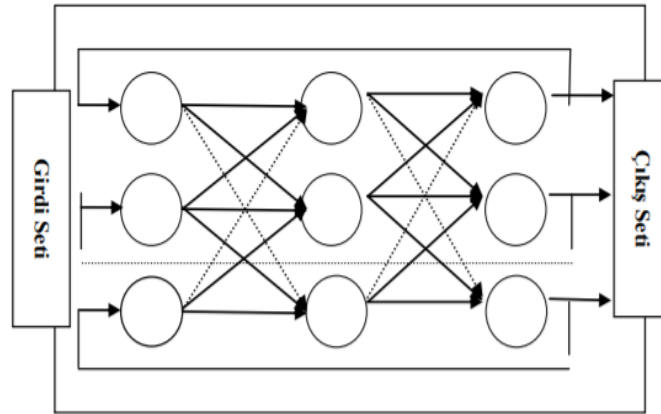


Şekil 3. 9. İleri beslemeli ağ yapısı [21]

### 3.3.4.2. Geri beslemeli yapay sinir ağları

Bu ağlarda nöronlar ileri yönde işlemler yapabildikleri gibi geri yönde de işlemler gerçekleştirebilmektedir. Bir başka deyişle bir katmandaki nöronlar hem ilerisindeki katmanlardaki nöronlarla, hem kendi katmanındaki nöronlarla, hem de gerisindeki katmanlardaki nöronlarla ilişkileri bulunmaktadır. Böylece girdiler her iki yönde iletilebilir. Bu ağlar dinamik bir yapıda bulunur. Dinamik yapıda olması sayesinde ağ eğitildikten sonra kendini sürekli olarak yeniler ve devamlı yeni bilgiler ile güncellenmesi sağlanmaktadır [21].

Geri beslemeli ağ yapısı Şekil 3.10.'da gösterilmiştir.



Şekil 3. 10. Geri beslemeli ağ yapısı [29]

Geri beslemede, döngüler belirli bir gecikmeye neden olacağı için bu ağlarda iletim hızı ileri beslemeli ağlara göre daha yavaş gerçekleşmektedir.

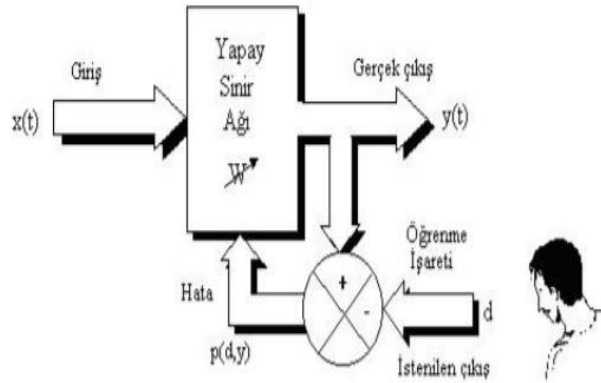
### 3.4. Yapay Sinir Ağlarında Öğrenme Yöntemleri

Yapay sinir ağlarında öğrenme yöntemleri çok çeşitli olabilmektedir. Örneğin; danışmanlı öğrenme, danışmansız öğrenme, yarı-danışmanlı öğrenme, transdüktif akıl yürütme, çevrimiçi öğrenme, takviyeli öğrenme ve aktif öğrenme gibi çok çeşitli öğrenme yöntemleri bulunmaktadır [30]. Bununla birlikte yapay sinir ağlarında öğrenme yöntemleri temel olarak danışmanlı, danışmansız ve yarı danışmanlı olmak üzere üç farklı sınıfta ele alınmaktadır.

#### 3.4.1. Danışmanlı öğrenme

YSA öğretilen yöntemlerinden biri olan bu yöntemde bir uzman tarafından problemin çözümüne ulaşmak için çeşitli girdiler ve bu girdilere karşılık gelen çıktılar modele yerleştirilir. Model eğitilirken, eğitim verileri arzulan doğruluk derecesine varıncaya kadar işlem sürdürülür. Bu işlemlerden sonra modele yeni girdiler verilerek tahminde bulunması istenir. Danışmanlı öğrenme, aslında regresyon ve sınıflandırma olarak iki sınıftan oluşur [31].

Danışmanlı öğrenme yöntemi Şekil 3.11.'de gösterilmiştir.

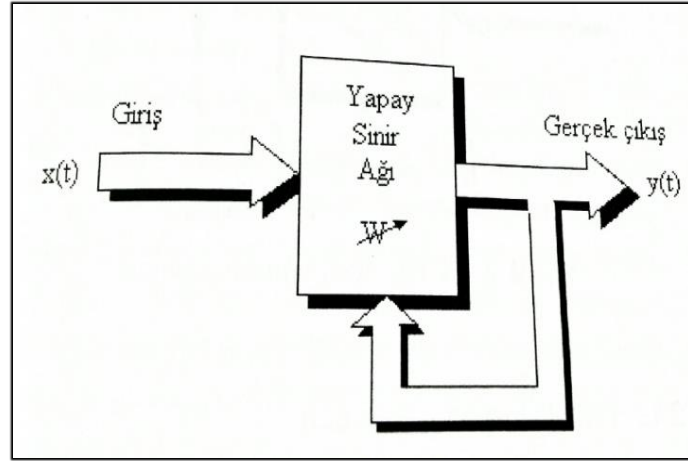


Şekil 3. 11. Danışmanlı öğrenme [32]

#### 3.4.2. Danışmansız öğrenme

Bu yöntemde, model eğitilirken örnek girdiler ağıya verilir. Bununla birlikte YSA'ya herhangi bir beklenen çıktı sunulmaz. Buna bağlı olarak örnek girdilerin sınıfının ne olduğunu ve parametreleri ile ilişkilerini öğrenmesi arzulanır. Kümeleme işlemleri, bu öğrenme modelinin en bariz örneklerindedir [31].

Danışmansız öğrenme yöntemi Şekil 3.12.'de gösterilmiştir.

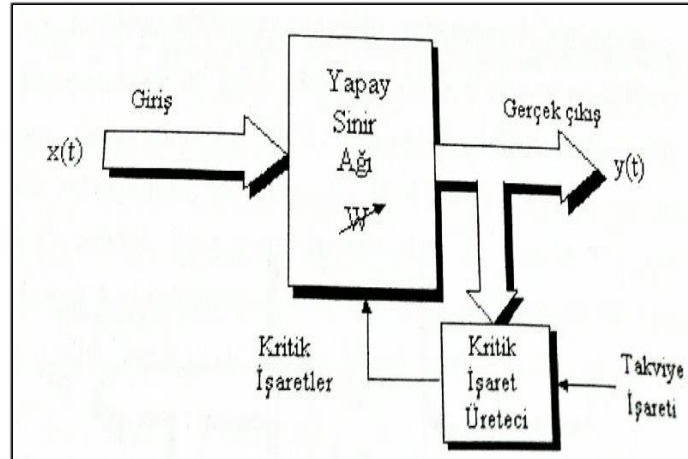


Şekil 3. 12. Danışmansız öğrenme [32]

### 3.4.3. Yarı danışmanlı öğrenme

Yarı danışmanlı öğrenme kuralı, danışmanlı öğrenme metoduna yakın bir yöntem olarak bilinmektedir. Giriş olarak verilen veride çok miktarda etiketlenmemiş veri ve az miktarda etiketlenmiş veri bulunmaktadır [33]. Bu metot genellikle etiketlenmiş verinin az olduğu, etiketlenmemiş verinin kolaylıkla elde edilebildiği durumlarda kullanışlı olmaktadır.

Yarı danışmanlı öğrenme yöntemi Şekil 3.13.'de gösterilmiştir.



Şekil 3. 13. Yarı danışmanlı öğrenme [32]

### 3.5. Yapay Sinir Ağlarının Eğitimi ve Test Edilmesi

YSA'ların eğitimi ve test edilmesi süreci, YSA'ların performansını etkileyen en önemli süreçtir. Eğitim aşamasında YSA'daki nöronların ağırlık değerleri, çıktının doğru gerçekleşebilmesi için belirli düzenlemelere tabi tutulmaktadır. Test işlemi aşamasında ise YSA'daki parametreleri etkileyen herhangi bir değişiklik yapılmaksızın ağın belirlenen amaç doğrultusunda hazırlanan prosedürlere uygun şekilde davranıp davranmadığı belirlenmektedir.

#### 3.5.1. YSA eğitimi

YSA eğitim sürecinde işlem test doğruluk oranının istenilen en iyi sonuca ulaşmasına kadar devam etmektedir. Bazı nedenler neticesinde eğitimin istenilen sonuca ulaşamadığı görülmektedir. Bu durumlarda eğitime bazı sınırlamaların koyulması gerekmektedir. Bu sınırlamalar YSA eğitim sürecinde eğitim parametreleri ile öğrenme kuralları olarak adlandırılan algoritmalar olarak ifade edilmektedir.

YSA tasarımında ağın tanımlanması için kullanılan değerlere eğitim parametreleri denir. Aşağıda belli başlı eğitim parametreleri listelenmiştir [20].

- Yineleme sayısı: Tüm veri kümesinin sinir ağları boyunca bir kez döngüsünü tamamlayıp, ağın ağırlıklarının güncellenmesi olayına yineleme sayısı (epoch) denilmektedir. Yineleme sayısının artırılması eğitim verisinin ezberlenilmesine yol açar. Bu durumda eğitim verisinde olmayan yeni bir veri sisteme gösterildiğinde istediğimiz sonuçlar alınamayabilir. Aynı şekilde yineleme sayısı az belirlenirse; amaçlanan başarı sağlanamamaktadır.
- Parça: Bir yineleme süreci boyunca bilginin sinir ağı boyunca ilerlemesi ve bu şekilde ağın ağırlıklarının yeniden güncellenerek hesaplanmasına denilmektedir.
- Hedef: Yapay sinir ağları çıktısının hata oranına ne kadar yaklaşılacağını belirleyen parametredir. Eğitim süresi istenen hata oranına kadar devam eder.
- Öğrenme Oranı: Hangi öğrenme kuralı kullanılıyorsa ona göre belirlenen eğitimin yoğunluğunu ifade eder.

YSA'larda nöronların arasındaki ağırlıkların değiştirilme işlemi çeşitli öğrenme kurallarının kullanımı ile gerçekleştirilmektedir. Bu kurallar aşağıda şu şekilde tanımlanabilir:

- Hebb Kuralı: Bir YSA'nın başka bir YSA'dan girdi almasıdır. Eğer ağların polariteleri aynı ise bağlantı ağırlığı artırılır. Hebb tarafından 1949 yılında geliştirilmiştir.
- Delta Kuralı: Widrow – Hoff tarafından geliştirilmiştir. YSA'daki beklenen çıktı ile mevcut çıktının farkının karesi alınıp bağlantı ağırlıkları minimize edilir.
- Hopfield Kuralı: 1982 yılında Hopfield tarafından bulunmuş olan algoritmadır. Aynı polariteye sahip girdi ve çıktı nöronlarının öğrenme katsayısı artırılır. Buna karşın farklı polariteye sahip girdi ve çıktı nöronlarının öğrenme katsayısı azaltılır.
- Kohonen Kuralı: Kohonen tarafından 1982 yılında geliştirilmiştir. Kohonen kuralında nöronların ağırlıkları ağırlık çıktısına göre ayarlanır. Çıktılara yakın olan nöronların ağırlık değerleri 1'e, uzak olan nöronların ağırlık değerleri 0'a yakınsanmaktadır.

### 3.5.2. YSA'nın test edilmesi

YSA giriş ve çıkışlarının eğitiminden sonra ağ eğitim setinde yer almayan örneklerden oluşan verilerle test edilir. Burada YSA ağırlık değerleri ile ayarlama olmamaktadır. Test aşamasında sadece ağırlık çıktısı hesaplanmaktadır.

Yapay sinir ağlarında verisetinin yeterince büyük olduğu durumlarda öğrenmenin daha kolay gerçekleştirilmesini sağlar. Ancak ağ eğitimi sırasında gereğinden fazla nöron kullanılmışsa; ağ eğitim setini ezberler. Bu duruma aşırı uydurma veya aşırı öğrenme denir. Diğer taraftan ağ gereğinden az sayıda nöron ile eğitilip ağırlık çıktısında kararsızlıklar görülmesi durumuna yetersiz uydurma denir.

### 3.6. Yapay Sinir Ağlarının Temel Özellikleri

YSA'nın temel özellikleri aşağıda belirtilmiştir [34]:

- **Öğrenme:** YSA'nın çıktıyı doğru tahmin edebilmesi için öğrenme işlemini düzgün şekilde gerçekleştirilmesi gerekir.
- **Genelleme:** YSA, öğrenmeyi gerçekleştirdikten sonra daha önceden karşılaşmadığı test verilerini analiz ederek doğru zamanda doğru tepkiyi verebilir.
- **Uyarlanabilirlik:** Tıpkı uyarlamalı örnek tanıma, sinyal işleme, sistem tanıma ve denetim örneklerinde olduğu gibi ilgilendiği problemleri çözmek için yapısındaki ağırlıkları tekrar tekrar işleyerek eğitir.

- **Hata Toleransı:** Her YSA'nın hesapladığı çıktı ile beklenen çıktı arasında belli bir ölçüde farklılık (hata) bulunmaktadır. Bu nedenle YSA oluşturulurken ortaya çıkabilecek bu hataları telafi edebilecek şekilde oluşturulurlar.
- **Doğrusal Olmama:** YSA hücreleri doğrusal yapıda bulunmamaktadır. Bu hücrelerin birleşmesinden meydana gelen YSA modelleri de doğrusal yapıda değildir.
- **Paralellik:** Çoğu algoritmanın aksine yapay sinir ağları paralel olarak çalışmaya uygundur.
- **Hız:** YSA paralel çalıştığından problemleri hızlı bir şekilde çözebilir.
- **Analiz ve Tasarım Kolaylığı:** YSA'nın temel yapısı tüm sinir ağlarında aynı olduğundan dolayı kolaylıkla oluşturulup, problemlerin çözümünde aktif rol oynarlar.

### 3.7. Yapay Sinir Ağlarının Avantajları ve Dezavantajları

YSA'nın avantajlarından bazıları aşağıda verilmiştir [35]:

- Çeşitli örnekler üzerinden genelleme yapabilir.
- Algılamaya yönelik olaylarda Uzman Sistemlerle birlikte kullanılarak oldukça başarılı sonuçlar ortaya koymaktadır.
- YSA'lar yeni koşullara uyum sağlar ve buna ilişkin durumları öğrenebilmesi oldukça kolaydır. Her ne kadar bilgiler eksik de olsa kendi kendini eğitebilmesi sayesinde eksik bilgiler ile performanslı bir şekilde yapay sinir ağları çalışabilir.

YSA'nın dezavantajlarından bazıları aşağıda verilmiştir [18]:

- Yapay Sinir Ağlarının ağ yapısı genel olarak deneme yanılma yöntemiyle belirlendiğinden dolayı yerel çözümlere takılabilmektedir.
- Öğrenme katsayısı hücre sayılarının belirlenmesinde kesin bir kural olarak tanımlanmadığı için bazı durumlarda nöron sayısı tanıma işlemini ve eğitim hızını etkilemektedir.
- Yapay Sinir Ağlarının eğitilmesinde örnekler önem taşırlar ancak bir örnek seçim kuralı bulunmamaktadır.
- Bilgiler ağa gömülü oldukları için yorumlanması ve çözümü elde etmek ve ağın takibini yapmak oldukça zordur.

### 3.8. Yapay Sinir Ağları Optimizasyon Algoritmaları

Optimizasyon algoritmaları, kayıpları azaltmak için sinir ağının ağırlık ve öğrenme hızı gibi özelliklerini değiştirmek için kullanılan algoritmalar veya yöntemlerdir. Kayıpları azaltmak ve mümkün olan en doğru sonuçları sağlamaktan optimizasyon algoritmaları veya stratejileri sorumludur. Sinir ağlarının öğrenmesinde, maliyetini, hızını, süresini, verimini en optimal seviyeye getirmek için optimizasyon algoritmaları kullanılmaktadır. Bu algoritmalarından en önemlileri şunlardır:

#### 3.8.1. Stokastik gradyan inişi metodu

Bu yöntem, yapay sinir ağları optimizasyon algoritmalarında en çok kullanılan yöntemlerden birisidir. Yapay sinir ağı öğrenme sürecini yerine getirirken her aşamada ağırlık değerlerinin belirli prosedürlere göre değişmesi gerekir. Bu prosedürlere en çok uygulanan metotlardan birisi “Stokastik Gradyan İnişi” yöntemidir.

Bu yöntemde tüm eğitim setleri için ağırlık değerlerinin değişimi gerçekleştirilir. Bundan dolayı sinir ağı daha etkin ve hızlı çalışır. Optimizasyon noktasına erişim süresi kısaldır. Aşağıdaki formül ile SGD algoritması ifade edilir:

$$\theta_{t+1} = \theta_t + \eta \nabla_{\theta} J(\theta_t; x^{(i)}; y^{(i)}) \quad (3.4)$$

Burada

$\theta \in \mathbb{R}^d$  model parametreleri,  $\eta$ : Öğrenme katsayısı,  $x^{(i)}$  : Eğitim örneği,  $y^{(i)}$ : Etiket değeri,

$\nabla_{\theta} J(\theta)$ : Parametrelere bağlı olarak hedef fonksiyonunun gradyanı olarak ifade edilir.

Bununla birlikte yöntemin en önemli olumsuz yanı sabit bir öğrenme katsayısının belirlenmek zorunda kalınmasıdır. Eğer katsayı küçük seçilirse öğrenme işlemi yavaş gerçekleşecek ve buna bağlı olarak optimum değere ulaşma süreci uzayacaktır. Diğer taraftan öğrenme katsayısı büyük seçilirse optimum değere ulaşma imkansız hale gelebilir ve minimum değer etrafında salınımlar oluşabilir [36].

#### 3.8.2. Uyarlamalı moment kestirimi metodu

Bu yöntemde her döngüde öğrenme katsayısı yenilenir. ADAM yönteminde iki parametre ön bellekte saklanır. Bunlar ağırlıklandırılmış ortalamalar ( $v_t$ ) ve momentum

değişiklikleridir ( $m_t$ ). Bu metotta ilk olarak önceki eğimlerin karelerinin üssel ağırlıklı ortalamaları alınır, sonrasında ise momentum değişiklikleri ile ön bellekte birleştirilerek saklanır.

ADAM optimizasyon algoritması, son zamanlarda bilgisayarla görme ve doğal dil işlemede derin öğrenme uygulamaları için daha geniş bir şekilde benimsenen stokastik gradyan inişinin bir uzantısıdır. Uygulaması kolay, hesaplama açısından verimli sonuçlar üreten, küçük hafıza gereksinimleri isteyen, eğimlerin diyagonal olarak yeniden ölçeklendirilmesine gerek olmayan bir metottur. Veri ve/veya parametreler açısından büyük problemler, durağan olmayan hedefler ve çok gürültülü veya seyrek gradyanlı sorunlar için çok uygundur.

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \quad (3.5)$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \quad (3.6)$$

$$m'_t = \frac{m_t}{1 - \beta_1^t}, v'_t = \frac{v_t}{1 - \beta_2^t} \quad (3.7)$$

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{v'_t + \epsilon}} m'_t \quad (3.8)$$

Model parametrelerinin şu şekildedir [37]:

$\eta$ : Öğrenme katsayısı,

$v_t$ : Geçmiş eğimlerin karelerinin üssel olarak ağırlıklandırılmış ortalamaları,

$m_t$ : Momentumdaki geçmiş eğimlerin üssel olarak ağırlıklandırılmış ortalamaları,

$\beta_1$  : Varsayılan değer 0.9,  $\beta_2$ : Varsayılan değer 0.999,  $\epsilon$ :  $10^{-8}$  değerine,

$g_t$  : t anında,  $\theta$  parametresine göre hesaplanmış, maliyet fonksiyonunun eğitimi,

$\theta$  :  $\mathbb{R}^d$  model parametreleridir.

ADAM algoritmasına ait sözde kod aşağıda gösterilmiştir [38, 40].



---

**Algoritma 1** ADAM algoritması sözde kodu

---

**Gereklilik :** Öğrenme katsayısı  $\eta$

**Gereklilik :** Bozulma katsayısı  $\beta_1, \beta_2$

**Gereklilik :** Stokastik Objektif Fonksiyon  $f(\theta)$

$m_0 \leftarrow$  ilk momente ata

$t_0 \leftarrow$  ilk momente ata

$v_0 \leftarrow$  ilk momente ata

**while**  $\theta_t$  yakınsama yapmazsa **do**

$t \leftarrow t + 1$

$g_t \leftarrow \nabla_{\theta} f_t(\theta_{t-1})$

$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$

$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2$

$m'_t = \frac{m_t}{1 - \beta_1^t}$

$v'_t = \frac{v_t}{1 - \beta_2^t}$

$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{v'_t + \epsilon}} m'_t$

**end while**

---

### 3.8.3. Karesel ortalamanın karekökü yayılımı metodu

Karesel ortalamanın karekökü yayılımı metodu (RMSprop), momentumlu SGD algoritmasına benzer. Karesel ortalamanın karekökü yayılımı metodu, salınımları dikey olarak sınırlayarak öğrenme katsayısını arttırır. Algoritma yatay olarak daha hızlı birleşir ve daha fazla ilerleyebilir. Bu metot ile SGD metodu arasındaki fark, gradyanların hesaplanışının farklılığına dayanır [38].

Yöntem, AdaDelta yöntemine benzemekle birlikte öğrenme katsayısı parametresini kullanır. Metot oluşturulurken,  $\eta$  öğrenme katsayısını,  $E[g^2]_t$  geçmiş eğimlerin karelerinin ortalamasıdır.  $g_t$  t anında,  $\theta$  parametresine göre hesaplanmış, maliyet fonksiyonunun eğitimidir.  $g_t^2$  t anında,  $\theta$  parametresine göre hesaplanmış, maliyet fonksiyonunun eğitim karesidir.  $\theta$  parametresi,  $\mathbb{R}^d$  model parametreleridir.

$$E[g^2]_t = 0.9E[g^2]_{t-1} + 0.1g_t^2 \quad (3.9)$$

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{E[g^2]_t + \epsilon}} \cdot g_t \quad (3.10)$$

RMSprop algoritmasına ait sözde kod aşağıda gösterilmiştir [38].

---

**Algoritma 2** RMSprop algoritması sözde kodu

---

**Gereklilik :** Global öğrenme oranı  $\eta$ , bozulma oranı  $\beta$

**Gereklilik :** İlk ağırlık  $\theta$

**Gereklilik :** Numerik durağanlık için  $\epsilon$

İlk olarak eğim hesaplama değerini  $r = 0$  yap.

**while** durdurma kriteri karşılanmayana kadar **do** (yap)

X eğitim setinden bir mini grup  $m$  örneği, Y'ye karşılık gelir

Eğimi hesapla:  $g \leftarrow \frac{1}{m} \sum_{i=1}^m \nabla_{\theta} L(f(\theta_t; x^{(i)}), y^{(i)})$

Kare eğim için:  $r \leftarrow \beta_r + (1 - \beta) g \cdot g$

Hesaplama güncellemesi:  $\nabla_{\theta} = \frac{\eta}{\epsilon + \sqrt{r}} \cdot g$

Güncellemeyi uygula:  $\theta \leftarrow \theta + \nabla_{\theta}$

**end while**

---

### 3.8.4. Uyarlamalı gradyan algoritması

AdaGrad, SGD yöntemindeki sabit öğrenme katsayısının oluşturduğu çeşitli problemleri ortadan kaldırmak amacıyla geliştirilmiştir. Öğrenme oranı, geçmiş gözlemlerin bilgisi dahil edilerek parametrelere bileşen olarak uyarlanır. Seyrek özelliklerle ilgili parametreler için daha büyük güncellemeler (örneğin yüksek öğrenme oranları) ve sık olan için daha küçük güncellemeler (örneğin düşük öğrenme oranları) gerçekleştirir.

AdaGrad'ın algoritmaları, daha bilgilendirici gradyan tabanlı öğrenme gerçekleştirmek için önceki yinelemelerde gözlemlenen verilerin geometrik bilgilerini dinamik olarak birleştirir. AdaGrad iki versiyonda yayınlanmıştır. Çapraz AdaGrad versiyonu uygulamada kullanılan versiyon olup; ana özelliği boyut başına bir öğrenme

oranını koruması ve uyarlamasıdır. Tam AdaGrad olarak bilinen ikinci versiyonu ise her bir direktif için bir öğrenme oranını muhafaza eder.

Sonuç olarak, seyrek verilerle uğraşırken bu yöntem çok uygundur. Her parametrenin, seyrek gradyanlı problemlerde performansı artıran kendi öğrenme hızı vardır. AdaGrad'ı kullanmanın avantajları kısaca öğrenme hızı yakınsamasının manuel olarak ayarlanması ihtiyacını ortadan kaldırması ve böylece ağırlıkların ölçeklendirilmesi eşit olmadığında; SGD'den daha hızlı ve daha güvenilir sonuçlar vermesidir. Bunun yanısıra ana parametrenin boyutuna çok duyarlı değildir.

AdaGrad yönteminin çalışma prensibi aşağıdaki formüllerde adım adım gösterilmektedir.

$$g_{t,i} = \nabla_{\theta_t} J(\theta_{t,i}) \quad (3.11)$$

$$\theta_{t+1,i} = \theta_{t,i} - \eta g_{t,i} \quad (3.12)$$

$$\theta_{t+1,i} = \theta_{t,i} - \frac{\eta}{\sqrt{G_{t,ii} + \epsilon}} g_{t,i} \quad (3.13)$$

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{G_t + \epsilon}} \odot g_t \quad (3.14)$$

$g_{t,i}$  objektif fonksiyonun gradyanı,  $\eta$  öğrenme oranı,  $G_t$ ,  $\mathbb{R}^{d \times d}$  diyagonal matris olup; geçmiş gradyanların karesinin toplamıdır.  $\theta \in \mathbb{R}^d$  model parametreleridir.

Bu şekilde eşitlik yeniden ayarlandığı takdirde aşağıdaki gibi AdaGrad'ın sözde kodu ortaya çıkacaktır [8].

AdaGrad algoritmasına ait sözde kod aşağıda gösterilmiştir [38].

---

**Algoritma 3** AdaGrad algoritması sözde kodu

---

**Gereklilik :** Küresel öğrenme oranı  $\alpha$ , bozulma oranı  $\beta$

**Gereklilik :** İlk ağırlık  $\theta$

**Gereklilik :** Numerik durağanlık için  $\epsilon \in (10^{-7})$

İlk olarak eğim hesaplama değerini  $r = 0$  yap.

**while** durdurma kriteri karşılanmayana kadar **do** (yap)

X eğitim setinden bir mini grup  $m$  örneği, Y'ye karşılık gelir

Eğimi hesapla:  $G \leftarrow \frac{1}{m} \sum_{i=1}^m \nabla_{\theta} L(f(\theta_i; x_i), y_i)$

Kare eğim için:  $r \leftarrow r + G \cdot G$

Hesaplama güncellemesi:  $\nabla_{\theta} \leftarrow \frac{\epsilon}{\delta + \sqrt{r}} \cdot G$

Güncellemeyi uygula:  $\theta \leftarrow \theta + \nabla_{\theta}$

**end while**

---

### 3.8.5. Uyarlamalı öğrenme oranı metodu

AdaDelta optimizasyonu, AdaGrad'ın iki önemli problemini çözmek için oluşturulan uyarlanabilir öğrenme oranına dayanan stokastik gradyan inişi yöntemidir. Bu problemler; eğitim boyunca öğrenme oranlarının sürekli olarak azalması ile manuel olarak seçilen bir küresel öğrenme oranına duyulan ihtiyaçtır.

AdaDelta, tüm geçmiş gradyanları biriktirmek yerine öğrenme oranlarını hareketli gradyan güncellemeleri prensibine göre uyarlayan daha sağlam bir AdaGrad uzantısıdır. Bu şekilde, AdaDelta birçok güncelleme yapıldığında bile öğrenmeye devam etmektedir. AdaGrad ile karşılaştırıldığında, AdaDelta'nın orijinal versiyonunda bir başlangıç öğrenme oranı belirlenmesine gerek yoktur.

Etkili öğrenme oranları göz önüne alındığında AdaDelta optimizasyon algoritması sayesinde yapay sinir ağının öğrenme oranı eğitiminin sonuna doğru 1'e yaklaşır ve bu rakam oldukça yüksek bir öğrenme oranıdır. Yöntemde bunu sağlayan imkanlar; gradyanlar ve adım boyutlarının küçük olması ile epsilon sabitinin geçmiş gradyanlara ve parametre güncellemelerine hakim olmasıdır.

İlgili yöntemin çalışma prensibi aşağıdaki formüllerde gösterilmiştir.

$$E[g^2]_t = \rho E[g^2]_{t-1} + (1 - \rho) g_t^2 \quad (3.15)$$

$$\Delta\theta_t = - \frac{RMS[\Delta\theta]_{t-1}}{RMS[g]_t} \cdot g_t \quad (3.16)$$

$$RMS[g]_t = \sqrt{E[g^2]_t + \epsilon} \quad (3.17)$$

$$E[\theta^2]_t = \rho E[\theta^2]_{t-1} + \Delta\theta_t^2 \quad (3.18)$$

$$\theta_{t+1} = \theta_t - \Delta\theta_t \quad (3.19)$$

$E[g^2]_t$  geçmiş eğimlerin karelerinin ortalamasıdır.  $\rho$  bozulma sabitidir.  $RMS[g]_t$  parametre güncellemeleri için  $t$  zamanına kadar önceki gradyanların karakök ortalamasıdır.  $\theta \in \mathbb{R}^d$  model parametreleridir.

AdaDelta algoritmasına ait sözde kod aşağıda gösterilmiştir [39].

---

**Algoritma 4** AdaDelta algoritması sözde kodu

---

**Gereklilik:** Bozulma oranı  $\alpha$ , Sabit  $\epsilon$

**Gereklilik:** İlk parametre  $x_1$

Biriktirme değişkenlerini başlat  $E[g^2]_0 = 0, E[\Delta x^2]_0 = 0$

**for**  $t = 1: T$  **do** %% Döngüyü güncelleştir

**Eğimi hesapla:**  $g_t$

**Gradyan biriktir:**  $E[g^2]_t = \rho E[g^2]_{t-1} + (1 - \rho) g_t^2$

**Hesaplama güncellemesi:**  $\nabla_{x_t} = - \frac{RMS[\Delta\theta]_{t-1}}{RMS[g]_t} \cdot g_t$

**Biriktirmeyi güncelle:**  $E[\Delta x^2]_t = \rho E[\Delta x^2]_{t-1} + (1 - \rho) \Delta x_t^2$

**Güncellemeyi uygula:**  $x_{t+1} = x_t + \Delta x_t$

**end for**

---

### 3.8.6. Momentum algoritması uygulanmış stokastik gradyan inişi

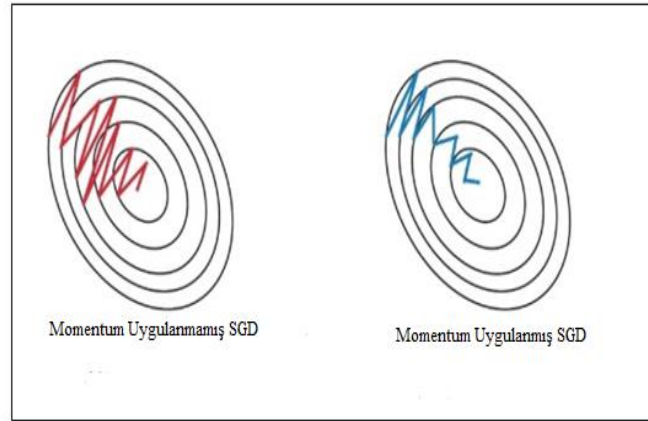
Bu yöntem, normal SGD yönteminin hızlanması ve var olan salınım problemlerini azaltmak için SGD yöntemine momentum katsayısı uygulanması ile oluşturulmuştur.

Aşağıdaki şekil Momentum uygulanmış SGD ile normal SGD arasındaki farkı göstermektedir. Momentum uygulandığından salınımlarda yumuşama ve daha hızlı yakınsama gerçekleşmektedir. Momentum yöntemi geçmiş parametreleri yenilerken sadece ilgili gradyanı değil, geçmiş gradyanları da hesaba katmaktadır [40]. SGDM' nin çalışma prensibi aşağıdaki formülde belirtilmektedir [40]:

$$v_t = \gamma v_{t-1} + \eta \nabla_{\theta} J(\theta, x^{(i)}, y^{(i)}) \quad (3.20)$$

$$\theta_t = \theta_{t-1} - v_t \quad (3.21)$$

Eşitlikte görülen  $\gamma$  parametresi 0 ile 1 arasında bir değer olup, önceki gradyanların ne kadarının hesaba katılacağını belirler ve genellikle 0.9 civarında bir değer olarak kabul edilmektedir.  $\theta : \mathbb{R}^d$  model parametreleri,  $\eta$ : Öğrenme katsayısı,  $x^{(i)}$  : Eğitim örneği,  $y^{(i)}$ : Etiket değeri,  $\nabla_{\theta} J(\theta)$ : Parametrelere bağlı olarak hedef fonksiyonunun gradyanı olarak ifade edilir. Momentum uygulanan ve uygulanmayan SGD Şekil 3.14.'te gösterilmiştir.



Şekil 3. 14. Momentum uygulanmış SGD ile momentum uygulanmamış SGD [40]

Momentum uygulanmış SGD algoritmasına ait sözde kod aşağıda gösterilmiştir [38].

---

**Algoritma 5** Momentum uygulanmış Stokastik Gradyan İnişi algoritması sözde kodu

---

**Gereklilik :** Global öğrenme oranı  $\alpha$

**Gereklilik :** İlk ağırlık  $\theta$ , ilk hız  $v$

**Gereklilik :** Numerik durağanlık için, sabit değer  $\delta$  ( $10^{-7}$ )

İlk olarak eğim hesaplama değerini  $r = 0$  yap.

**while** durdurma kriteri karşılanmayana kadar **do** (yap)

X eğitim setinden bir mini grup  $m$  örneği, Y'ye karşılık gelir

Eğimi hesapla:  $g \leftarrow \frac{1}{m} \sum_{i=1}^m \nabla_{\theta} L(f(\theta; x(i)), y(i))$

Hız güncellemesini hesapla:  $v_{t+1} \leftarrow \beta_t \cdot v_t - (1 - \beta_t) g_t$

Güncellemeyi uygula:  $\theta_{t+1} \leftarrow \theta_t + \alpha_t v_t$

**end while**

---

### 3.8.7. Hızlandırılmış Nesterov gradyan algoritması (NAG)

Momentum yöntemiyle yakından ilgili bir diğer yöntem de Hızlandırılmış Nesterov Gradyan Algoritması yöntemidir. Bu yöntemin, Momentum yönteminden ayrıldığı nokta gradyan hesaplama aşamasında gerçekleşmektedir. Momentum yönteminde gradyan, mevcut  $\theta_t$  parametreleri kullanılarak hesaplanır. Buna karşın NAG yönteminde, ara parametreleri  $\theta$  hesaplamak için  $\theta$  parametrelerine  $v_t$  hızını ekleriz.

$$\theta = \theta_t + \alpha v_t \quad (3.22)$$

$$g_{NAG} = \frac{1}{n} \sum_{i=1}^n \nabla_{\theta} L(x^{(i)}, y^{(i)}, \tilde{\theta}) \quad (3.23)$$

Gradyanları aldıktan sonra, Momentum metodu ile benzer güncelleme kuralı kullanarak parametreler güncellenir.

$$v_{t+1} = \alpha v_t - \eta g_{NAG} \quad (3.24)$$

$$\theta_{t+1} = \theta_t + v_{t+1} \quad (3.25)$$

NAG, Momentum yöntemi için düzeltme faktörü olarak görülebilir. Momentum yönteminde, alınan optimizasyon yolu büyük salınımlar sergilediğinden çok yavaş olabilirken; NAG hız güncellemesi yaparak salınımlardaki kayıplardan kaçınacaktır [41].

### 3.8.8. Hızlandırılmış uyarlamalı Nesterov moment kestirimi algoritması

NADAM yöntemi, ADAM ve NAG yöntemlerinin bir araya getirilmesinden oluşturulmuştur. Bu iki algoritma birbirlerine eklenirken momentum ifadesi farklılaştırılır. Bu sayede mevcut güncelleştirme kuralı oluşturulur. Çalışma prensibine ilişkin formül aşağıdadır [42]:

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{v_t + \epsilon}} \left( \beta_1 m_t + \frac{(1 - \beta_1) g_t}{1 - \beta_1^t} \right) \quad (3.26)$$

Burada;

$\eta$ : Öğrenme katsayısı,  $\theta$  :  $\mathbb{R}^d$  model parametreleri,  $\epsilon$ : hata terimi.  $m_t$  momentum vektördür.  $v_t$  önceki gradyanların karelerinin ortalamasıdır.  $\beta_1 = 0.9$  değeridir.  $g_t$  objektif fonksiyonun gradyanıdır.



## 4. DENEY, GÖZLEM ve ANALİZ

### 4.1. Metotlar

Çalışmada kullanılacak programlama dili için çeşitli araştırmalar yapılarak matematiksel işlemlerde gösterdiği yüksek performans, matris ve vektörlerdeki kolay kullanımı, sahip olduğu kütüphaneler ile hızlı, performanslı, etkili ve kullanıcı dostu programlamaların yapılabilmesi gibi özelliklerinden dolayı Phyton programlama dili seçilmiştir.

Phyton bugün birçok alana ilişkin akademik çalışmalardan, kodlamalara kadar birçok farklı alanda kullanılmaktadır. Phyton, kolay bir şekilde okunabilme özelliği olan, çalışma sırasında nesnelere inceleyebilme özelliği olan, nesne tabanlı programlama dili ile kolay anlaşılabilmeyi sağlayan, ifade yeteneğinin güçlü olan, modüler yapıya sahip olan, kuraldışı (exception) tabanlı hata yönetim sistemine sahip olan, dinamik veri yapısına sahip olan, oldukça etkin kullanılabilen kütüphane yapıları olan, C ve C++ ile ek modüller yazabilme özelliği olan, diğer programlara arayüz olarak dahil edilebilme yeteneklerine sahip bir programlama dili olarak listelenmektedir [43].

Tüm bu özellikleri ile Phyton programlama dili birçok farklı alanda kullanılmaktadır. Phyton programlama dilinin kodlama yapabilme, makine öğrenmesi yapabilme, web uygulamaları yapabilme, örümcek tipli yazılımlar gerçekleştirme, bilim dünyasında sık kullanılmama, veri analizi gerçekleştirme, ağ ve soket programcılığı yapma, sistem yönetimi gerçekleştirme, veritabanı erişimini sağlama, oyun geliştirme, masaüstü programları olarak kullanılmama gibi oldukça geniş bir kullanım alanına sahiptir [43].

#### 4.1.1. Yapay sinir ağları

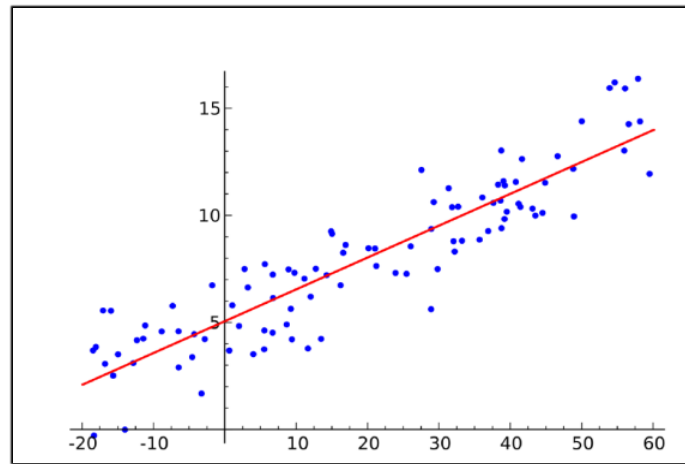
YSA, tıpkı bir insan beyni gibi öğrenme işlevini yerine getiren bilgisayar sistemleri olarak tanımlanır. Öğrenme sürecini örneklerle yaparlar. Bu ağlar, birbirleri ile bağlantılı yapay sinir hücrelerinden oluşur. Yapısındaki bağlantıların tümünün belli bir ağırlık değeri bulunmaktadır. YSA'nın sakladığı bilgiler bu ağırlık değerleri içerisinde bulunur. YSA'lar diğer hesaplama yöntemlerinden farklı bir hesaplama yöntemine sahiptir. Bu hesaplama yöntemi, içinde buldukları çevreye uyum sağlama, eksik bilgilerle çalışma, belirsizlikler altında kararlar alabilme özellikli sistemlerdir. Bu özellikler sayesinde, YSA'lar sınıflandırmada, örüntü tanımada, sinyal filtrelemede, veri sıkıştırımda ve optimizasyon

çalışmalarında en üstün tekniklerdir. YSA'lar ile veri madenciliği [44], optik karakter tanıma [45], önceden tanımlanmış bir gelecekte bir depremin büyüklüğünü tahmin etme [46], parmak izi tanıma [47], avuç içi tanıma [48], iris tanıma [49], malzeme bilimi analizi [50], tıbbi görüntü analizi [51] gibi birçok alanda başarılı örnekler gerçekleştirilebilmektedir.

Tezde gerçekleştirilen özkaynak hesaplanmasında yapay sinir ağları optimizasyon algoritmalarından ADAM metodu, Stokastik Gradyan İnişi metodu ve RMSprop metodu kullanılmış olup, bu algoritmalar istatistiksel yöntemlerden Lineer Regresyon algoritması ile karşılaştırılmıştır. Literatürde incelenen kaynaklarda genellikle özkaynak karlılığı analizinde yapay sinir ağları iyileştirme gerçekleştirilmeden kullanılmaktadır. Tezde ise özykaynak karlılığı hesaplanırken daha gerçekçi ve bilimsel tahminlere ulaşılması için optimizasyon teknikleri kullanılmış olup istatistiksel metot ile karşılaştırmalar gerçekleştirilmiştir.

#### 4.1.2. Lineer regresyon

Lineer regresyon, doğrusallığı ve X ve Y değişkenleri arasındaki ilişkiyi temsil eden istatistiksel bir modeldir. Lineer regresyon, eğilimlerin tahmin edilmesi ve oluşturulması gibi analizlerin tahmin edilmesinde yaygın olarak kullanılan modellerden biridir. Şekil 4.1. X ve Y değişkenleri arasındaki ilişkiye göre regresyon çizgisine uyan basit bir lineer regresyon modelini göstermektedir [22].



Şekil 4. 1. Lineer regresyon modeli [22]

Basit lineer regresyonda bağımlı deęişken (y) ile tek bir bağımsız deęişken (x) arasındaki ilişki incelenir. Deęişkenlerden birinin deęeri bilindiğinde dięerinin deęeri bulunur. Basit lineer regresyon formülü Eşitlik 4.1.'de verilmiştir [78].

$$Y = \beta_0 + \beta_1 X \quad (4.1)$$

Birden çok bağımsız deęişken ile bağımlı deęişken arasındaki ilişki ise çoklu regresyon ile incelenir. Burada  $\beta$  deęeri sabit deęer ve modelde yer alan bağımsız deęişkenlerin katsayılarını gösterir.  $\epsilon$  deęeri ise tesadüfi hata terimidir. Çoklu lineer regresyonun formülü Eşitlik 4.2.'de verilmiştir [78].

$$Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_p x_{ip} + \epsilon_i \quad (4.2)$$

## 4.2. İmplementasyon Detayları

Çalışmada bulunan optimum sonuçlar Tablo 4.1.'de gösterilmiştir.

Tablo 4. 1. Ağ parametreleri

Tanımlamalar	Açıklamalar
Ağın Türü	İleri Beslemeli Çok Katmanlı Algılayıcı
Öğrenme Algoritması	Geri Yayılım
Öğrenme Tipi	Denetimli Öğrenme
Öğrenme Kuralı	ADAM, RMSprop, Stokastik Gradyan İnişi(Momentum)
Girdi Katmanı Sayısı	15 (nöron)
Gizli Katman Sayısı	5
1. Gizli Katmandaki Nöron Sayısı	32 (nöron)
2. Gizli Katmandaki Nöron Sayısı	64 (nöron)
3. Gizli Katmandaki Nöron Sayısı	32 (nöron)
4. Gizli Katmandaki Nöron Sayısı	32 (nöron)
5. Gizli Katmandaki Nöron Sayısı	64 (nöron)
Çıktı Katmanı Sayısı	1 (nöron)
Öğrenme Oranı	0.001
Epok Sayısı	1000
Transfer Fonksiyonu	ReLU

**Transfer Fonksiyonu:** Çalışmada, Transfer fonksiyonlarından Düzeltilmiş lineer birim (ReLU) kullanılmıştır. Düzeltilmiş lineer birim (ReLU) aktivasyon fonksiyonu, Nair ve Hinton (2010) tarafından geliştirilmiş olup derin öğrenme uygulamaları için en iyi sonuçlar veren ve bu nedenle en yaygın kullanılan aktivasyon fonksiyonu haline gelmiştir [52]. ReLU, en başarılı ve yaygın olarak kullanılan işlev olduğu kanıtlanan ve daha hızlı öğrenen bir aktivasyon fonksiyonudur [53, 54]. Sigmoid ve tanh aktivasyon fonksiyonlarına kıyasla derin öğrenmede daha iyi performans ve genelleme sunmaktadır [55, 56, 57]. ReLU, neredeyse doğrusal bir işlevde en nitelikli seçenek olduğundan ve doğrusal modellerin özelliklerini gradyan-iniş yöntemleriyle optimize etmeyi en basit hale getirdiğinden tercihimiz tezde bu aktivasyon fonksiyonunun kullanılmasıdır.

**Epok Sayısı:** Epok sayısı modellerde geri yayılım yapma sayısıdır. Ağırlık değerlerini düzeltmek amacıyla sürekli olarak modelde geri gidilir ve ağırlık değerleri her epoktan sonra güncellenir. Epok sayısının istenilen değerden az belirlenmesi sonucunda ağırlık değerleri yeterince düzeltilemeyeceği için, hata sayısını minimuma düşürmek mümkün olamayacağından model tarafından doğru tahminler gerçekleştirilemeyecektir. Optimal epok sayısının bulunması durumunda ise yeterli ağırlık değeri düzenlemesi gerçekleştirilir ve hata sayısı minimuma düşürülerek modelin gerçeğe yakın tahminlerde bulunması sağlanır. Optimal epok sayısından daha büyük bir iterasyon değerlerinin seçilmesi durumunda ise hata değeri zaten daha önce optimum hata değerine ulaştığı için daha fazla iyileştiremeyecektir ve boşuna zaman kaybı meydana gelecektir. Ayrıca bu noktadan sonra ağ girdileri öğrenmek yerine ezberlemeye başlayacaktır. Bu sebeplerden probleme uygun epok sayısının belirlenmesi modelin başarısı için büyük öneme sahiptir.

**Öğrenme Oranı:** Öğrenme oranı, model içindeki en önemli parametrelerden biridir. Ağ ağırlıklarını güncelleyerek hatayı en aza indirir. Çok düşük veya çok yüksek öğrenme oranı seçmek, model performansını düşürecektir. Düşük öğrenme oranı seçildiğinde ağ ağırlıklarında küçük güncellemeler yapıldığından eğitim süreci yavaşlayacaktır. Diğer taraftan öğrenme oranı yüksek seçilirse hata değerleri optimal seviyeden uzaklaşacaktır. Uygulamada, eğitim yüksek öğrenme oranından düşük öğrenme oranına doğru ayarlanmalıdır. Literatürde genellikle öğrenme oranları büyük bir değer olarak “0.1” ile başlar daha sonra “0.01”, “0.001” vb. gibi daha düşük değerler ile devam eder. 0,001 üzerinde öğrenme oranının kullanılması eğitim süresini artırır ve aynı zamanda eğitim süresinin varyansını arttıracığından [59]; modelde “0.001” öğrenme oranı kullanılmasına

karar verilmiştir. Modelde tüm bankalar için diğer değerlerde (0,1 ve 0,01) denenmiş olup en yüksek tahmin değerlerine öğrenme oranı 0.001’de ulaşılmıştır.

**Çıktı Katmanı:** Modelimizde özkaynak karlılığı değişkeninin tahmin edilmesi amaçlandığından çıktı katmanı değeri 1 nöron olarak belirlenir.

**Girdi Katmanı:** Özkaynak karlılığı değişkeni dışında kalan tüm değişkenler girdi katmanını oluşturduğundan, ilgili katman sayısı 15 nöron olarak belirlenir.

**Gizli Katman Sayısı:** Deneylerimizde yapay sinir ağları 5 katman üzerine kurulmuştur. En iyi tahmin sonuçları 5. katmanda gerçekleşmiştir. Bu katmandan itibaren değerlerimiz stabil bir hal almıştır. Bu nedenle gizli katman sayımız 5 olarak kabul edilmiştir.

**Gizli Katmandaki Nöron Sayıları (32, 64, 32, 32, 64):** Çalışmada nöron sayıları parantez içerisindeki gibi verilmiştir. Genel olarak Izgara Arama (Grid Search) yöntemi kullanılmıştır. Izgara arama yöntemi kullanılmasının nedeni tezde kullanılan veri kümesinin sınırlı olması durumunda yöntemin sonuçlar üzerinde olumlu etkisi sağladığıdır [60]. Bu bağlamda da nöronlar deneme yanılma yöntemi ile ağa katılmıştır.

**Ağın Türü:** Amacımız bir model oluşturmak ve bu model ile bir output değeri almak olduğu için İleri Beslemeli Çok Katmanlı Algılayıcı kullanılmıştır.

**Öğrenme Algoritması:** Model oluşturulurken rastgele ağırlık değerleri verilir, bu değerler ile oluşturulan modelden aldığımız sonuçtaki hata oranına bakılarak geriye dönük olarak ağırlık değerleri alpha değeri ile düzeltilmeye çalışılır. Bu nedenden dolayı öğrenme algoritması olarak geriye yayılım algoritması uygun görülmüştür.

**Öğrenme Tipi:** Veri kümesinde giriş ve çıkış değerleri belli olduğundan dolayı modelde denetimli öğrenme kullanılmıştır.

**Öğrenme Kuralı:** Öğrenme kuralı olarak tezde Uyarlamalı Moment Kestirimi (ADAM), Karesel Ortalamanın Karekökü Yayılımı (RMSprop), Stokastik Gradyan İnişi (SGD) algoritmaları kullanılmıştır.

### 4.3. Sonuçlar

Bu bölümde, deneyimizde elde edilen sonuçlar grafikler gösterilerek yorumlanacak ve kullanılan optimizasyon algoritmalarının performansı karşılaştırılacak, ayrıca hangi algoritmanın veri kümemiz için daha iyi sonuçlar vereceği de karşılaştırılacaktır.

### 4.4. Veri Kümesi

#### 4.4.1. Banka bilgileri

Mevduat Bankaları, ticari bankalar ve vadesiz mevduat gibi transfer edilebilir mevduatları kabul eden diğer finansal kuruluşlardan oluşur. Türkiye’de mevduat bankaları kamu, yerli ve yabancı olarak 3 gruba ayrılmaktadır. Yurt içi özel banka mevduatı, Türkiye Cumhuriyeti’nin sermayesinin çoğunun gerçek veya tüzel kişilere olan banka mevduatlarıdır. Yabancı mevduat bankaları, birçok yabancı gerçek veya tüzel kişiliğe ait olup, banka mevduatları Türkiye’de faaliyet göstermektedir. Kamu mevduat bankalarının sermayesinin çoğu, Türkiye Cumhuriyeti devletlerine ait mevduat bankalarıdır. Mülkiyete göre, çalışmada devlet mevduat para bankaları kullanılmıştır.

Çalışmada seçilen bankalar ve ilgili bazı rakamsal büyüklükleri Tablo 4.2.’de gösterilmiştir.

Tablo 4. 2. Banka bilgileri

31.12.2019 İtibariyle Aktif Büyüklüklerine Göre Banka Sıralaması (Milyon TL)										
Banka	Kuruluş Yılı	Toplam Aktifler	Toplam Krediler	Toplam Mevduat	Toplam Özkaynaklar	Ödenmiş Sermaye	Net Dönem Kar/Zararı	Bilanço Dışı Hesaplar	Şube Sayısı (Adet)	Çalışan Sayısı (Adet)
Ziraat Bankası	1863	649,756	447,983	447,251	70,065	6,100	6,187	2,586,038	1,758	24,563
İş Bankası A.Ş.	1924	468,059	289,244	295,922	58,873	4,500	6,068	1,403,214	1,271	24,053
Halkbank	1938	457,045	309,208	297,734	32,197	1,250	1,720	1,760,198	1,006	18,967
Vakıfbank	1954	419,426	292,091	251,531	33,026	2,500	2,802	4,195,762	943	16,835
Garanti Bankası	1946	391,152	251,165	248,751	53,766	4,200	6,159	2,063,102	912	18,784
Yapı Kredi Bankası	1944	387,496	240,464	222,790	41,188	8,447	3,600	2,233,141	846	16,631
Akbank	1948	360,501	203,834	224,055	54,382	5,200	5,417	1,834,015	771	12,750

#### 4.4.2. Veri kümesi açıklamaları

Çalışmamızda veri kümemiz Türkiye Bankalar Birliği, BDDK, TCMB ve TÜİK ile bankaların yıllık finansal raporlarından oluşturulmuştur. Verilerimiz içerisinde 8 mikroekonomik değişken, yedi makroekonomik değişken ve bir bağımlı değişken vardır. Veri kümesini aldığımız dönemler Aralık 2006 - Aralık 2017 tarihleri arasında her üç aylık dönemi içeren Türkiye Kamu Mevduat Bankaları verileridir. Bu verilerdeki bağımlı

değişken olan özkaynakların kârlılığını yapay sinir ağı optimizasyon algoritmaları olan ADAM, Stokastik Gradyan İnişi ve RMSprop optimizasyon algoritmalarıyla tahmin edip ne kadar başarı elde edildiği algoritmaların birbirleri ile karşılaştırılarak gerçekleştirmeye çalışılmıştır.

- **Ortalama Özkaynak Kârlılığı:** Bankaya yatırılan sermayenin ne kadar karlı kullanıldığını gösteren kârlılıkla ilgili önemli kriterlerden biridir [61].

- **Özkaynaklar / Varlıklar:** Bu oran, özkaynaklar tarafından finanse edilen varlıkların oranını ifade eder. Oranın yükselmesi, bankanın kredi değerini arttıracaktır. Bu nedenle, oran arttıkça banka kârlılığının büyümesi beklenmektedir.

- **Mevduat / Aktifler:** Bankanın pasifinde gösterilen mevduatların toplam aktife oranıdır. Bu oranın büyüklüğü bankanın piyasadaki gücünü gösterir [62].

- **Krediler ve Alacaklar / Varlıklar:** Yüksek bir kredi-aktif oranı, bir bankanın daha fazla kredi verdiği ve daha fazla gelir elde ettiği gerçeğine işaret etmektedir. Öte yandan, düşük bir kredi-aktif oranı, bankanın daha az gelir elde ettiği anlamına gelmektedir. Ancak, yüksek bir kredi-aktif oranının bankayı yüksek likidite riskine soktuğunu inkâr edilememektedir.

- **Takipteki Krediler (brüt) / Krediler ve Alacaklar:** Bu oran, bankanın kredilerinin ne kadarını tahsil edemediğini gösterir. Bir nevi kredi kalitesine dair ipucu vermektedir. Oran düştükçe banka kârlılığı yükselmektedir [63].

- **Belirli Karşılıklar Sonrası Net Faiz Geliri / Varlıklar:** Net faiz gelirlerinin toplam varlıklara oranlanmasıyla elde edilir. Başka bir deyişle, elde edilen gelirin hangi oranda esas faaliyetlerden sağlandığını gösterir. Kısaca banka faaliyet etkinliğini ifade eder. Bu oranın yüksek olması istenmektedir.

- **Likit Aktifler / Aktifler:** Likit aktiflerin toplam aktiflere oranı bankanın likidite durumunu gösterir. Bu oran bir risk göstergesi olarak da ifade edilmektedir. Belirsizlik ortamı arttıkça, bankaların portföy çeşitliliğine giderek likit varlık oranlarını arttırmaları son derece önemlidir. Çünkü ekstra likite sahip olmak olası risklere karşı önemli bir koruma sağlar. Bununla birlikte, aşırı likit varlıkların banka kârlılığını da azaltabileceği unutulmamak zorundadır [61]

- **Personel Giderleri / Diğer Faaliyet Giderleri:** Bu oran işgücü verimliliğini gösteren bir değişkendir ve maliyet faktörü olarak da değerlendirilebilir. Bu nedenle, bankaların performansı üzerinde hem pozitif hem negatif olmak üzere iki yönlü bir etkisi vardır.

- **Faiz Dışı Gelir (net) / Varlıklar:** Bankanın ana faaliyetleri dışındaki faaliyetlerden kazanılan gelirin toplam varlıklara oranıdır. Bu oranın büyümesi genel olarak bankanın performansı üzerinde olumludur.

- **Döviz Kuru (USD / TL):** Döviz kuru, bankaların finansal performansı üzerinde önemli bir etkiye sahiptir. Çalışmada, döviz kuru aktif kârlılığı olan ROA ile pozitif bir ilişki içinde iken, özkaynak karlılığı olarak ROE ile negatif bir ilişki içerisinde [64].

- **Zorunlu Karşılık Oranı (ZKTL):** Bu oran arttığında bankaların daha düşük kredi vermeleri gerekecektir. Bu nedenle, banka kârı azalacaktır.

- **Sanayi Üretim Endeksi (SUE):** SUE, sanayi sektöründe yer alan şirketlerin üretimlerinde meydana gelen artma veya azalmayı gösteren bir endekstir. Endeks büyüdükçe banka karlılığının artması beklenilmektedir.

- **GSYİH Büyüme Oranı:** GSYİH birçok nedenden dolayı bankaların çeşitli büyüklüklerini etkileyebilir. Durgunluk dönemlerinde GSYİH'deki düşüş ile bankaların karları azaltmaktadır. GSYİH'deki büyümede bankaların karlılığı artar. GSYİH büyümesi ile banka karlılığı arasında pozitif bir ilişki doğru yönlü bir ilişki vardır [65].

- **Faiz Oranı (Gecelik Borçlanma Oranı):** Borç verme faiz oranı, bankanın merkez bankasından ihtiyaç duyduğu fon karşılığında ödeyeceği faiz oranıdır. Buna göre, Merkez Bankasının borç verme oranının artırılması, bankanın maliyetlerini artırabilir ve bu da banka kârlılığında bir azalmaya yol açabilir.

- **Enflasyon Oranı (TUFE):** İşletmelerin maliyetlerini ve getirilerini etkilemektedir. Enflasyon ve kârlılık arasındaki ilişkinin yönü, enflasyonun öngörülüp öngörülememesine bağlıdır. Enflasyon oranı öngörülebilirse bankalar faiz oranını, kârlarını artırmak için uyarlayabilirler. Aksine, enflasyon oranının öngörülmesi mümkün değilse, bankanın maliyetleri getirilerini aşabilir ve sonuç olarak banka karlılığı azalır.

- **Kukla (Kriz):** Krizin banka performansı üzerindeki etkisi için kukla değişken olarak kriz değişkeni, kriz dönemlerinde 0 diğer dönemlerde 1 şeklinde kullanılmıştır [62].

Tezde kullanılan değişkenler Tablo 4.3.'te toplu olarak gösterilmiştir.



Tablo 4. 3. Veri kümesinin değişkenleri

<b>Bağımlı Değişkenler</b>	<b>Bağımsız Değişkenler</b>
	<b>İçsel Değişkenler</b>
Ortalama Özkaynak Karlılığı	Özkaynaklar / Varlıklar
	Mevduat / Aktifler
	Krediler ve Alacaklar / Varlıklar
	Takipteki Krediler (brüt) / Krediler ve Alacaklar
	Likit Aktifler / Aktifler
	Belirli Karşılıklar Sonrası Net Faiz Geliri / Varlıklar
	Faiz Dışı Gelirler (net) / Varlıklar
	Personel Giderleri / Diğer Faaliyet Giderleri
	<b>Dışsal Değişkenler</b>
	Döviz Kuru (USD/TL)
	Zorunlu Karşılık Oranı (ZKTL)
	Enflasyon Oranı (TUFE)
	Kriz (dummy)
	GSYİH Büyüme Oranı
	Sanayi Üretim Endeksi (SUE)
	Faiz Oranı (gecelik borçlanma oranı)

#### 4.5. Problem Tanımı ve Çalışmanın Amacı

##### 4.5.1. Arka plan

Türk bankacılık sektörü, banka karlılığını detaylı bir şekilde incelemek için birçok yöntem geliştirmiştir. Sektör, son 20 yılda önemli değişikliklere uğramıştır. 1980'lerin ortalarından bu yana gelişimini ciddi şekilde engelleyen müdahaleler ve düzenlemelerin kaldırılmasıyla büyük ölçüde serbestleşmiştir. Bu serbestleşme ile birlikte kurumsal çerçevenin daha da iyileştirilmesi ve genel olarak bankaların ve mali piyasaların daha verimli işlemesi sağlanmıştır. Bu durum, günümüzde bankacılık sektörü için yeni ve daha rekabetçi bir ekonomik ortam yaratmıştır.

Son yıllarda, Türk bankaları iç piyasadaki konumlarını güçlendirmeye ve kısmen uluslararası finans piyasalarına daha kolay erişmelerine olanak sağlayan bir fayda elde etmeye çalışmıştır. Yeni teknolojinin benimsenmesi ve altyapının iyileştirilmesiyle birlikte bu değişiklikler, banka karlılığının performansına olumlu yönde etki yapmıştır.

Son 20 yılda bankacılık sektörü, çalışma ortamında dünya çapında büyük dönüşümler yaşamıştır. Hem dış hem de iç faktörler yapısını ve performansını oldukça etkilemiştir. Pek çok ülkede gözlemlenen bankaların rolü genel olarak ekonomik faaliyetlerin ve özel olarak da pazarın farklı bölümlerinin finansmanının merkezi olmaya devam etmektedir. Sağlam ve karlı bir bankacılık sektörü olumsuz şoklara daha iyi dayanabilir ve finansal sistemin istikrarına katkıda bulunabilir. Bu nedenle banka karlılığını etkileyen faktörler başta banka yöneticileri olmak üzere diğer araştırmacılarında ilgisini çekmektedir. Short [66], Bourke [67], Molyneux ve Thornton [68], Demircuc-Kunt ve Huizinga [69] ve Goddard vd., [70] banka karını açıklamada önemli olabilecek çeşitli faktörlerin etkisini tahmin etmek için doğrusal modeller kullanmışlardır. Bu araştırmalar banka karlılığı üzerinde birçok analizin yapılabileceğini gösterse de bazı konular yeterince ele alınmamıştır. Birincisi, literatürde esas olarak banka karlılığını etkileyen faktörler düzeyinde araştırmacılar çalışma gerçekleştirirken, değişkenlerin seçimi bazen iç tutarlılıktan yoksun olabilir veya makroekonomik ortamın etkisini kapsamlı yansıtmayabilir. İkincisi, literatürün çoğunda metodoloji yeterince tanımlanmamıştır bu da elde edilen tahminlerin yanlışlığı ve tutarsız olabileceği gibi bir durumu ortaya koymaktadır. Bu durumun meydana gelmemesi için tezde, temel bir denklem çerçevesinde bankaya özgü değişkenler ve makroekonomik değişkenler bankaların özkaynak karlılığını tespit etmede kullanılmıştır.

Mikroekonomik değişkenler bir başka deyişle bankaya özgü değişkenler, bankaların bilanço ve gelir tablolarında bulunan çeşitli muhasebe kalemlerinin oranlanmasıyla belirlenmiş değerlerdir. İkinci değişkenler olan makroekonomik değişkenler ise bir ülkeye ait genel ekonomiye ilişkin göstergelerden oluşan değişkenlerdir.

Tezdeki çalışma, yapay sinir ağları ile bu değişkenleri kullanarak banka karlılığının hesaplanmasına ve gerçek banka karlılığı verileriyle yakınlığını belirlemeye yönelik bir araştırmayı hedeflemektedir. Modeli belirlerken yapay sinir ağları optimizasyon algoritmaları metotlarını kullanarak özkaynak kârlılığı hesaplanmıştır.

#### **4.5.2. Çalışmada incelenen problem**

Çeşitli bankaya özgü mikro ekonomik değişkenler ve makroekonomik değişkenler kullanılarak yapay sinir ağları algoritmaları vasıtasıyla banka karlılığını tahmin edilmesi tezde ele aldığımız problemdir. Bu problemin çözümü için YSA tabanlı yöntem geliştirilmesinin ve bu yöntemin geleneksel analiz yöntemlerine göre birçok avantaja sahip

olması ve karmaşık modelleri hızlı ve yüksek doğrulukta analiz etme becerisine sahip olmasından dolayı tercih edilmiştir [71]. Ayrıca ekonometrik bir modelle karşılaştırıldığında, daha kısa bir sürede ve daha yüksek doğruluk payıyla tahminin elde edilmesi YSA kullanımını tez için gerekli kılmıştır.

Yapay sinir ağları kullanılarak, standart ekonomik modellerle analiz edilmesi çok zor olan birçok sorunu incelemek kolaylaşmaktadır [72]. YSA, geleneksel analiz yöntemlerine göre birçok avantaja sahiptir. Zaman serileri verilerinin dinamik yapıda olması nedeniyle; bu serilerin analizlerinde doğrusal araçlar yetersiz kaldığından, doğrusal olmayan araçları tercih etmek gerekmektedir. YSA bu tür analizler için oldukça başarılıdır. Çünkü yapay sinir ağları, tam veya eksik verilerle daha iyi sonuçlar elde etmektedir. Lineer regresyon modeli gibi bir ekonometrik/istatistiksel modelin aksine, YSA metotları ile ulaşılan tahminlerde doğruluk paylarının daha fazla olması nedeniyle günümüzde daha çok tercih edilmeye başlanmıştır.

Tezde incelenen problem, yapay sinir ağları optimizasyon tekniklerinden seçilmiş SGD, ADAM ve RMSprop ile istatistiksel yöntemlerden en çok kullanılan lineer regresyon modellerinin kullanılması suretiyle hesaplanan sonuçlarla gerçek sonuçların ne kadar yakın olduğu saptanarak en uygun yöntemlerin belirlenmeye çalışılmasıdır.

Bu tezde, öncelikle bankaların özkaynak karlılığını etkileyen bankaya ait çeşitli değişkenler ile makroekonomik değişkenler literatürden yararlanılarak çalışmaya katılmıştır. Daha sonra yapay sinir ağları ve ekonometrik/istatistiksel yöntemler ile bu veriler kullanılarak çeşitli dönemlere ait özkaynak karlılıkları hesaplanmıştır. Hesaplanan özkaynak karlılıkları, seçilen bankaların aynı dönemlerindeki özkaynak karlılıkları ile karşılaştırılmıştır. Hesaplanan özkaynak karlılıkları ile gerçek özkaynak karlılıklarının birbirlerine ne yüzdede yakın olduğuna bağlı olarak ilgili yapay sinir ağları optimizasyon yöntemlerinin ne derece isabetli tahminlerde bulunduğu gösterilmeye çalışılmıştır. Ayrıca lineer regresyon yöntemi kullanılarak aynı şekilde süreç gerçekleştirilmiş ve yapay sinir ağları optimizasyon yöntemleri ile istatistiksel yöntemlerin hangilerinin bu tür çalışmalarda tercih edileceği araştırılmıştır.

#### **4.5.3. Veri kümesi ile ilgili kısıtlamalar**

Tezde bazı kısıtlamalar ve sınırlılıklar bulunmaktadır. Bunlar;

- Veri kümesi 2006 Aralık ile 2017 Aralık arasına aittir. 2006 Aralık döneminden önce verilerin alınmamış olması paradan sıfırın atılması nedeniyledir. Diğer taraftan 2017 Aralık tarihinden sonra verilerimiz içinde bulunan mikroekonomik değişkenlerden takipteki krediler verisinin 2018 itibariyle donuk alacaklar olarak değişmesi ve hesaplanmasının farklılaşması nedeniyledir.
- Veriler Türkiye Cumhuriyet Merkez Bankası, Türkiye İstatistik Kurumu, Bankacılık Düzenleme ve Denetleme Kurumu ile Türkiye Bankalar Birliği'nden alınarak oluşturulmuştur. Bu veriler oluşturulurken zamansal periyot kısıtıyla karşı karşıya kalınmıştır. Bankalar verilerini 3'er aylık periyotlar halinde yayınlamaktadırlar. Bu bakımdan bakıldığında verilerimiz sayısal olarak kısıtlıdır.

Tüm bu kısıtlamalar ve literatüre uygunluk açısından veriler çalışmalarda genel olarak bahsi geçen 2006 ile 2017 yıllarını kapsamaktadır.

#### 4.6. Deneysel Kurulum

Tezde banka karlılığının tahminini gerçekleştirmek için yapay sinir ağları optimizasyon algoritmaları kullanılmıştır. Model genel olarak bankaların t (2006 Aralık - 2017 Aralık 3'er aylık dönem) zaman periyodundaki özkaynak karlılığını ( $\Pi_{it}$ ) hesaplarken, karlılığa etki eden bankaya özgü değişkenleri ( $X_{it}^b$ ), makroekonomik değişkenleri ( $X_{it}^m$ ), hata terimini ( $\epsilon_{it}$ ) ve sabit terimi (c) kullanır. Tezde kullanılan model Brooks'un 2008 yılında banka karlılığını etkileyen faktörlere dönük ürettiği model temel alınarak oluşturulmuştur [73].

$$\Pi_{it} = c + \sum_{b=1}^B \beta_b X_{it}^b + \sum_{m=1}^M \beta_m X_{it}^m + \epsilon_{it} \quad (4.3)$$

$\Pi_{it}$  = i bankasının t dönemindeki özkaynaklar üzerinden kârlılığını,

c = sabit terimi,

$X_{it}^b$  = bankaya özgü değişkenleri,

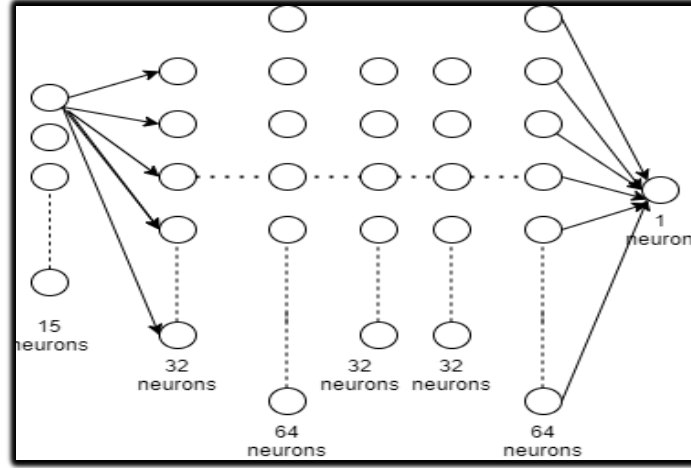
$X_{it}^m$  = makroekonomik değişkenlerle ilgili kontrol değişkenleri,

$\epsilon_{it}$  = hata terimini temsil etmektedir.

#### 4.6.1. Yapay sinir ağı eğitimi süreci

Tezde yapay sinir ağı optimizasyon algoritmalarında öz sermaye getirisinin hesaplanması Eşitlik 4.3'deki model kullanılarak gerçekleştirilmiştir. Tablo 3'de verilmiş parametrelerden de görülebileceği üzere modelimiz Çok Katmanlı Algılayıcı İleri Besleme, öğrenme algoritmamız Geri Yayılım, öğrenme tipimiz Denetimli Öğrenme'dir. Deneylerde ADAM, RMSprop ve Stokastik Gradyan İnişi optimizasyon algoritmaları kullanılmıştır. Ağ eğitimi süreci daha detaylı olarak şu şekilde açıklanabilir:

- **Katmanlar ile katmanlarda bulunan nöronların tespiti:** Modelimizde özkaynak karlılığı değişkeninin tahmin edilmesi amaçlandığı için çıktı katmanımızda 1 nörondur. Özkaynak karlılığı değişkeni dışında kalan tüm değişkenler girdi katmanını oluşturduğundan ilgili katmanımızda 15 nöron bulunmaktadır. Deneylerde en iyi tahmin sonuçları 5. katmanda gerçekleştiği ve 5. katmandan itibaren sonuçlarımız stabil olduğu için gizli katman sayımız 5 olarak kabul edilmiştir. Gizli katmanlarımızda sırasıyla 32, 64, 32, 32, 64 nöron bulunmaktadır. Aşağıdaki şekilde görüldüğü gibi tüm nöronlar birbirleri ile bağlantılıdır. Şekil 4.2. katman sayısını ve katmanlardaki toplam nöron sayısını gösterir. Çalışmada doğruluk değerini arttıran ilk faktör hem toplam katman sayısının hem de katmanlardaki toplam nöron sayısının artırılmış olmasıdır. Bu noktada, katmanlar içindeki düğüm sayısını belirlerken Izgara Arama yöntemi ile rastgele değerler kullanılmıştır. Seçtiğimiz veri kümesinin küçük bir arama uzayına sahip olması Izgara Arama yönteminin kullanılmasını için problemin uygun olduğu anlamına gelmektedir. Katmanlarda kullandığımız değerler rasgele atanmış nöron sayılarıdır. Bu yüzden deneylerde en uygun katman sayısı ve nöron sayıları denenecek bulunmuştur. Buna göre katman sayısı 5 ve katmanlardaki nöron sayıları (32, 64, 32, 32, 64) olarak belirlenmiştir.



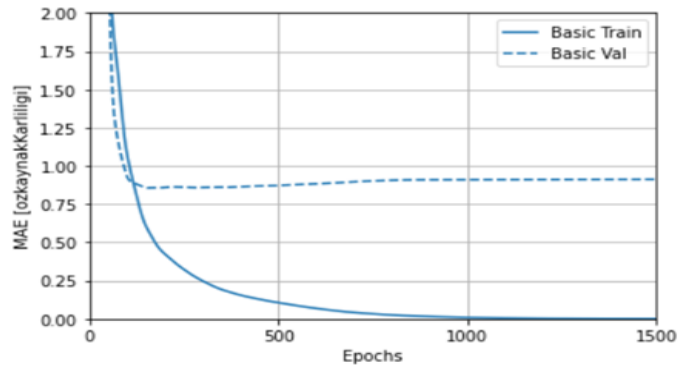
Şekil 4. 2. Çalışmadaki yapay sinir ağı yapısı

- **Öğrenme oranı seçilme aşaması:** Modelin öğrenme sürecini başarılı şekilde gerçekleştirmesi amacıyla öğrenme oranı seçimi literatüre uygun olarak gerçekleştirilmiştir. Sırasıyla 0.1, 0.01, 0.001 değerleri çalışmaya katılmıştır. 0,1 ve 0.01 değerleri kullanıldığında eğitim süresinin uzadığı, eğitim süresinin varyansının arttığı ve tüm bankalar için tahmin değerlerinin 0.001'e göre daha düşük gerçekleştiği görüldüğünden; literatürde küçük veri setlerine sahip olan çalışmalarda kullanıldığı gibi 0.001 öğrenme oranının kullanılmasına karar verilmiştir[59].
- **Döngü (Epok) sayısı seçilmesi aşaması:** Tez çalışması kapsamında optimal epok sayısının belirlenmesinde literatürdeki çalışmalar ve gerçekleştirilen deneylerde elde edilen sonuçlar temel alınmıştır. Literatürde genel olarak 1000 epok sayısının kullanıldığı görülmektedir. Ayrıca ağı eğiteceğimiz örnek sayısına göre 10, 50, 100, 500, 1000 ve daha büyük epok değerleri de modellerdeki hata sayısını minimum değere çekebilmek için kullanılmaktadır [58]. Tez çalışması kapsamında ele aldığımız 350 veri sayısına sahip veri kümesi üzerinde optimum değeri belirlemek için önerilen yapay sinir ağı modeli 1500 epok sayısını kapsayacak şekilde eğitilmiştir. Deney sonucunda elde edilen epok sayısı – ortalama mutlak hata (MAE) grafiği Şekil 4.3.'de gösterilmektedir. Ayrıca daha sağlıklı bir analiz için her 100 epok için MAE değerinin azalma miktarlarının karşılaştırıldığı veriler Tablo 4.4' te verilmiştir.

Tablo 4. 4. Epok değerlerine göre Loss, MAE, MSE değerleri

Epok Sayısı	Loss	MAE	MSE
0	92.7380	9.5292	92.7380
100	1.8082	1.0763	1.8082
200	0.3581	0.4240	0.3581
300	0.1247	0.2476	0.1247
400	0.0573	0.1548	0.0573
500	0.0265	0.1052	0.0265
600	0.0117	0.0681	0.0117
700	0.0047	0.0413	0.0047
800	0.0017	0.0247	0.0017
900	0.0006	0.0150	0.0006
<b>1000</b>	0.0002	<b>0.0086</b>	0.0002
1100	0.0001	0.0047	0.0001
1200	0.0000	0.0024	0.0000
1300	0.0000	0.0012	0.0000
1400	0.0000	0.0005	0.0000
.....	.....	.....	.....
1495	1.631300e-07	0.000246	1.631300e-07
1496	1.605582e-07	0.000244	1.605582e-07
1497	1.576266e-07	0.000242	1.576266e-07
1498	1.549062e-07	0.000240	1.549062e-07
1499	1.523791e-07	0.000238	1.523791e-07

Temel Eğitim (Basic Train) eğrisinin 1000 epok sayısına kadar sürekli azalarak 0'a yaklaştığı ve 1000 epok sayısına ulaştığında eğrinin x eksenini neredeyse 0'da kestiği görülmektedir. 1000 epok sayısında MAE değerinin ilk iki ondalık basamağının sıfırlandığı Tablo 4.4'te açıkça görülmektedir. Tablo ve grafik sonuçlarından da açıkça görüldüğü gibi 1000 epok sayısının üzerinde çok anlamlı bir iyileşme olmamaktadır ve daha fazla ağı çalıştırmak zaman kaybına neden olmaktadır. Bu sebeple tezde epok sayısı için 1000 değeri kabul edilmiştir ve bundan sonraki deneyler 1000 epok sayısına göre gerçekleştirilecektir.



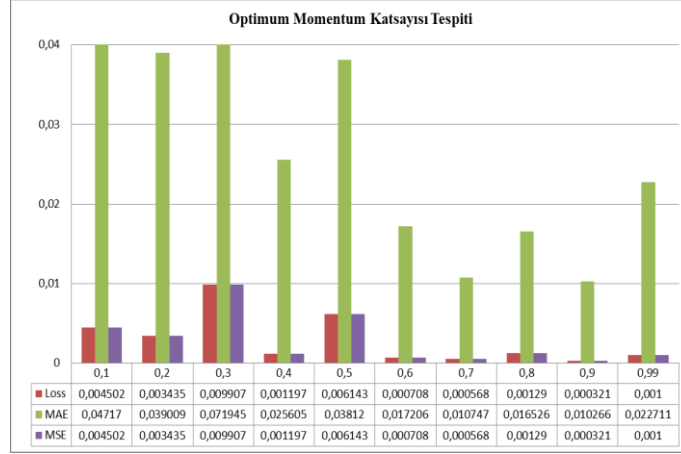
Şekil 4. 3. Eğitim – validasyon kaybı grafiği ile uygun epok seçimi

- **Transfer fonksiyonu seçilmesi:** Modelde transfer fonksiyonlarından Düzeltilmiş lineer birim (ReLU) kullanılmıştır. ReLU, derin öğrenme uygulamaları içinde en iyi sonuçlar veren aktivasyon fonksiyonu olması [52], hızlı şekilde öğrenen bir aktivasyon fonksiyonu olması [53, 54] ve sigmoid ile tanh aktivasyon fonksiyonlarına kıyasla derin öğrenmede daha iyi performans ve genelleme sunması [55, 56, 57] gibi nedenlerden dolayı tezde bu aktivasyon fonksiyonu seçilmiştir.
- **Stokastik Gradyan İnişi Algoritması için momentum seçimi:** Momentum katsayısı modelin öğrenmesini etkileyen önemli hiper parametrelerden birisidir. Bu oran bir önceki iterasyondaki değişiminin belirli bir oranının yeni değişim miktarına eklenmesi olarak açıklanmaktadır. Bu şekilde yerel çözümlere takılan ağların sıçrama ile daha iyi sonuçlar bulmasını sağlar. Seçilen momentum değerinin optimum seviyede olması ve bu değerden ne büyük ne de küçük olması önemlidir. Değer optimum seviyeden çok küçük seçilirse yerel çözümlerden kurtulmayı zorlaştırır, büyükse tek bir çözüme ulaşmakta sıkıntılar yaşanmaktadır [74]. Genellikle momentum katsayısı 0 ile 1 arasında alınır. Literatürde yapılan çalışmalar incelendiğinde genellikle en iyi tahmin sonuçlarına 0.9 momentum değerinde ulaşıldığı görülmüştür. Bu çalışma kapsamında optimum değer tespit edilebilmesi için 0.1 ile 0.99 arasında 0.1'lik adımlarla tüm momentum değerleri ile yapay sinir ağı eğitilerek 10 farklı deney gerçekleştirilmiştir. Deneyler sonucunda elde edilen Loss, MAE ve MSE değerleri Tablo 4.5 ve Şekil 4.4'te verilmiştir.

Tablo 4. 5. Farklı momentum katsayıları kullanarak elde edilen değerler

Momentum Katsayıları	Loss	MAE	MSE
0.1	0.004502	0.047170	0.004502
0.2	0.003435	0.039009	0.003435
0.3	0.009907	0.071945	0.009907
0.4	0.001197	0.025605	0.001197
0.5	0.006143	0.038120	0.006143
0.6	0.000708	0.017206	0.000708
0.7	0.000568	0.010747	0.000568
0.8	0.001290	0.016526	0.001290
<b>0.9</b>	<b>0.000321</b>	<b>0.010266</b>	<b>0.000321</b>
0.99	0.001000	0.022711	0.001000





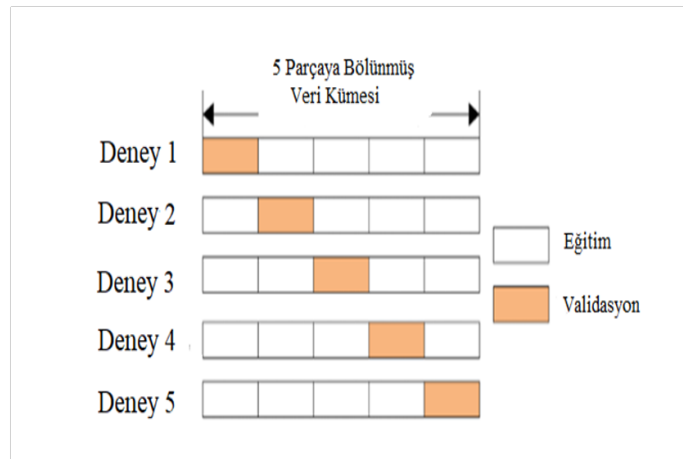
Şekil 4. 4. Optimum momentum katsayısı tespiti

Gerçekleştirilen deneylerde elde edilen sonuçlar incelendiğinde literatürdeki çalışmalarla bir paralellik olduğu gözlemlenmiştir. Geliştirilen yapay sinir ağı SGD optimizasyon yöntemiyle 0.9 momentum değerinde, 0.000321 Loss, 0.010266 MAE, 0.000321 MSE sonuçlarını elde etmiştir. 0.6 ve 0.7 momentum değerlerinde elde edilen sonuçlara yakın olsa da Tablo 4.5’ te açıkça görülmektedir ki önerilen yapay sinir ağı modeli en iyi sonuçlara 0.9 momentum değerinde ulaşmıştır. Gerçekleştirilen deneylerde elde edilen sonuçlara dayanarak bu tez çalışması kapsamında yapılacak tüm deneylerde 0.9 momentum katsayısı kullanılacaktır.

- **Bozuk veri kontrolü:** Modelde bozuk veriler kontrol edilmiştir. Veri kümesinde herhangi bir bozuk veri varsa, sistem bozuk verileri belirler ve mümkün olduğunca veri kümesinde bozuk veri kalmayana kadar bu verileri silmeye başlar. Öte yandan işlem öncesi veri kümesinden nümerik olmayan dönem değerleri ve banka isimleri çıkarılmıştır. Bu prosedür, tam olarak kaç tane girdi katmanının kullanılacağını gösterir. Böylece gereksiz giriş değerleri dışarıda bırakılarak doğruluk değeri arttırılmaya çalışılmıştır.
- **K katlı çapraz doğrulama aşaması:** Bir model geliştirme için verilerin az olduğu veri madenciliği ve makine öğrenimi alanlarında, modelin genelleme yeteneğini kontrol etmek için önerilen en yaygın prosedür k katlı çapraz doğrulama yöntemidir [75]. Bu yöntem, eğitim sürecinde sinir ağının aşırı uyumunu önlemek için gerçekleştirilir. K katlı çapraz doğrulama prosedürü gerçekleştirmek için model geliştirmeye yönelik veriler ilk olarak k eşit (veya neredeyse eşit) boyutlu segmente veya kata bölünür. Her k tane kat kaç segmente bölündüyse o kadar yineleme yaptıktan sonra, her yinelemede verilerin farklı bir katı doğrulama için dışarıda bırakılırken, kalan (k-1) kat öğrenme amacıyla kullanılır. Daha sonra öğrenilen

modeller hakkında tahminlerde bulunmak için kullanılır. Böylelikle, her seferinde bir model oluşturulur ve "görünmeyen" bir veri kümesiyle model test edilir. Bu yöntemin avantajı, daha küçük veri kümelerinde güvenilir ve tarafsız testler gerçekleştirmesidir çünkü bu işlem basit eğitilmiş ve test edilmiş prosedürden çok daha fazla hesaplama çabası gerektirir. Küçük örneklem boyutu tahminindeki yanlışlık, k kat çapraz doğrulama ile azaltılabilir [76]. Bu gerekçelerden dolayı tezde, eğitim sürecinde sinir ağının aşırı uyumunu önlemek için 5 kat çapraz doğrulama yöntemi benimsenmiştir. Kullanılan veri kümesi 5 katlı çapraz doğrulama yapılarak ayrılmıştır. Bu nedenle veri kümesi manuel yerine algoritma tarafından otomatik olarak 5 parçaya bölünmüştür. Her seferinde çapraz doğrulama algoritması, veri kümesinin 5 parçasından 1'ini validasyon olarak diğer 4'ünü ise eğitim olarak otomatik bir biçimde ayırmaktadır. Bu durum veri kümesinin %20'sinin validasyon, %80'inin eğitim olarak ayrılmış olması anlamına gelmektedir. Çapraz doğrulama yönteminde rasgele durum değeri (random state) "0", "1", "2", "3" ve "4" olarak atandığında, veri kümesi bu sayılar kadar karıştırılmıştır. Rastgele durum değerini "5" olarak atadığımızda, doğruluk değerimiz "0" rasgele durum değerinin doğruluk değerine eşit olmaktadır. Bundan dolayı rastgele durum değerlerimiz [0 - 4] arasında atanmasına karar verilmiştir.

5 katlı çapraz doğrulama aşaması Şekil 4.5.'te gösterilmiştir.

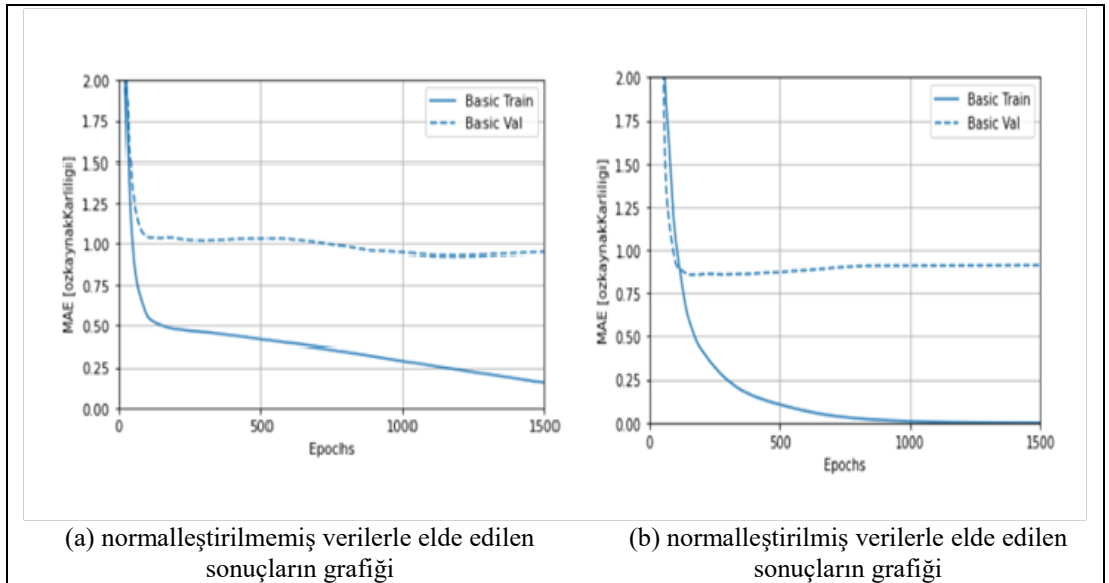


Şekil 4. 5. Çalışmadaki çapraz doğrulama yapısı

Sonuç olarak, çapraz doğrulama prosedüründe veri kümesinin % 20 test ve % 80 model eğitimi kısmı her seferinde değiştiğinden model eğitimi çok daha güvenilir hale gelmiştir. Nihai sonuçlar, validasyon sonuçlarının ortalaması olarak alınmıştır.

- **Normalleştirme aşaması:** Yapay sinir ağlarında eğitim ve test aşaması tamamlandıktan sonra belirli ön işlemler uygulanarak ağın eğitimi daha başarılı bir şekilde gerçekleştirilebilir. Verilerin ham halden normalleştirme süreci ile eğitime uygun hale getirilmesi sağlanır. Normalleştirmede sinir ağı içindeki yansılığını en aza indirmek için her bir veri aynı değer aralıklarında ölçeklendirilir. Aynı zamanda, normalleştirme ile aynı ölçek içindeki her özellik için eğitim süreci başlar ve eğitim süresi hızlanır.

Tez kapsamında verileri normalleştirmenin çalışmaya olan katkısını daha açık bir biçimde ortaya koyabilmek için veriler arasından rastgele seçilen Ziraat Bankası verileri üzerinde iki aşamalı deneyler gerçekleştirilmiştir. Birinci aşamada veriler üzerinde hiçbir normalleştirme yapılmadan yapay sinir ağı eğitilmiş ve sonuçlar alınmıştır. İkinci aşamada ise aynı şartlar altında aynı veriler üzerinde normalleştirme işlemi uygulanarak deney tekrarlanmış ve sonuçlar alınmıştır. Elde edilen sonuçlara ait epok sayısı - MAE grafikleri Şekil 4.6.'da verilmiştir.



Şekil 4. 6. Normalizasyon öncesi ve sonrası Epok sayısı MAE grafikleri

Şekil 4.6.' da epok sayısı - MAE grafikleri, solda normalleştirilmemiş verilerle elde edilen sonuçların grafiği ve sağda normalleştirilmiş verilerle elde edilen sonuçların grafiği olmak üzere gösterilmiştir.

Elde edilen deneysel sonuçlar incelendiğinde Şekil 4.6.'da görüldüğü üzere önerilen yapay sinir ağı normalleştirilmiş veriler üzerinde 1000 epok civarında 0 MAE değerine yaklaşırken, normalleştirme aşaması uygulanmadığında 1500 epokta bile 0 MAE değerine yakınsayamadığı görülmektedir. Normalizasyon işlemi

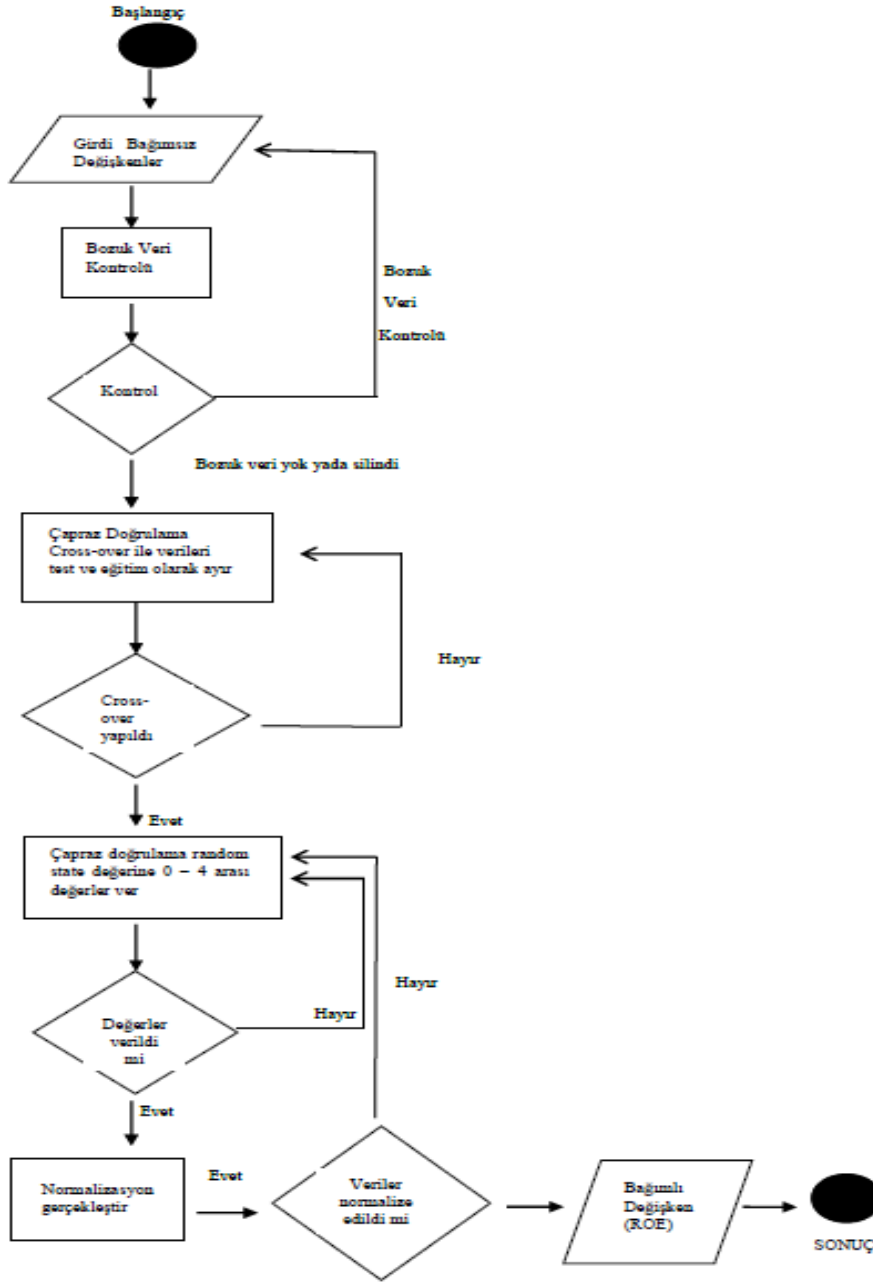
uygulandığında modelin öğrenmeyi daha hızlı ve iyi bir biçimde gerçekleştirdiği görülmüştür.

Bu özelliklerinden dolayı tezde normalleştirme işlemlerinden Min-maks normalleştirme yöntemi kullanılmıştır. Bu yöntemde göre veri özellikleri veya çıktıları yeni bir değer aralığında yeniden ölçeklendirilir. Literatürde sıklıkla veriler 0 ile 1 veya -1 ile 1 aralığında olacak şekilde normalize edilir. Min-maks normalleştirme uygulandığında, her özellik yeni değer aralığında aynı kalacaktır. Min-maks normalleştirme, verilerdeki tam olarak tüm ilişkileri koruma avantajına sahiptir. Bu bağlamda veri kümesi bölündükten sonra, ayrılan veriler normalleştirmeye koyulmuş ve özellikle normalizasyonda sayısal değerleri 0 ile 1 arasında tutarak deneyler sonucunda performansımız arttırılmıştır.

- **Geri yayılım (back propagation) gerçekleştirme:** Bir sinir ağını bazı görevleri yerine getirecek şekilde eğitmek için gizli katmanların ve çıktı katmanlarının ağırlıklarının ayarlanması gerekir. Böylece gerçek çıktının hedef çıktıya daha yakın olması sağlanır ve hedef çıktı ile gerçek çıktı arasındaki hata azaltılır. Bu süreç, sinir ağındaki ağırlıklara ait toplam hataların türevinin hesaplanmasıyla son bulur [77]. Geri yayılma algoritması, ağırlıkların hata türevini belirlemede en yaygın kullanılan yöntemdir. Geri yayılım algoritması aktivasyonların girişten çıkış katmanına yayıldığı ileri faz ve çıktı katmanında gözlemlenen gerçek ve istenen nominal değer arasındaki hatanın geriye doğru yayıldığı geri fazdan oluşmuştur. Tezde analiz aşamasında geri yayılım yapmak için ilk olarak ağırlık değerleri rastgele değerlerle başlatılmış, oluşturulan model için hedef çıkış değeri, çıkış nöron hatası, gizli nöron hatası ve delta olarak adlandırılan ağırlık varyasyonları hesaplanmıştır.

Tüm bu işlemler sonucunda doğruluk değerimiz deneylerde her banka için % 80' in üzerine çıkartılmıştır.

Deneysel aşamaları gösteren veri akış diyagramı Şekil 4.7.'de gösterilmiştir.



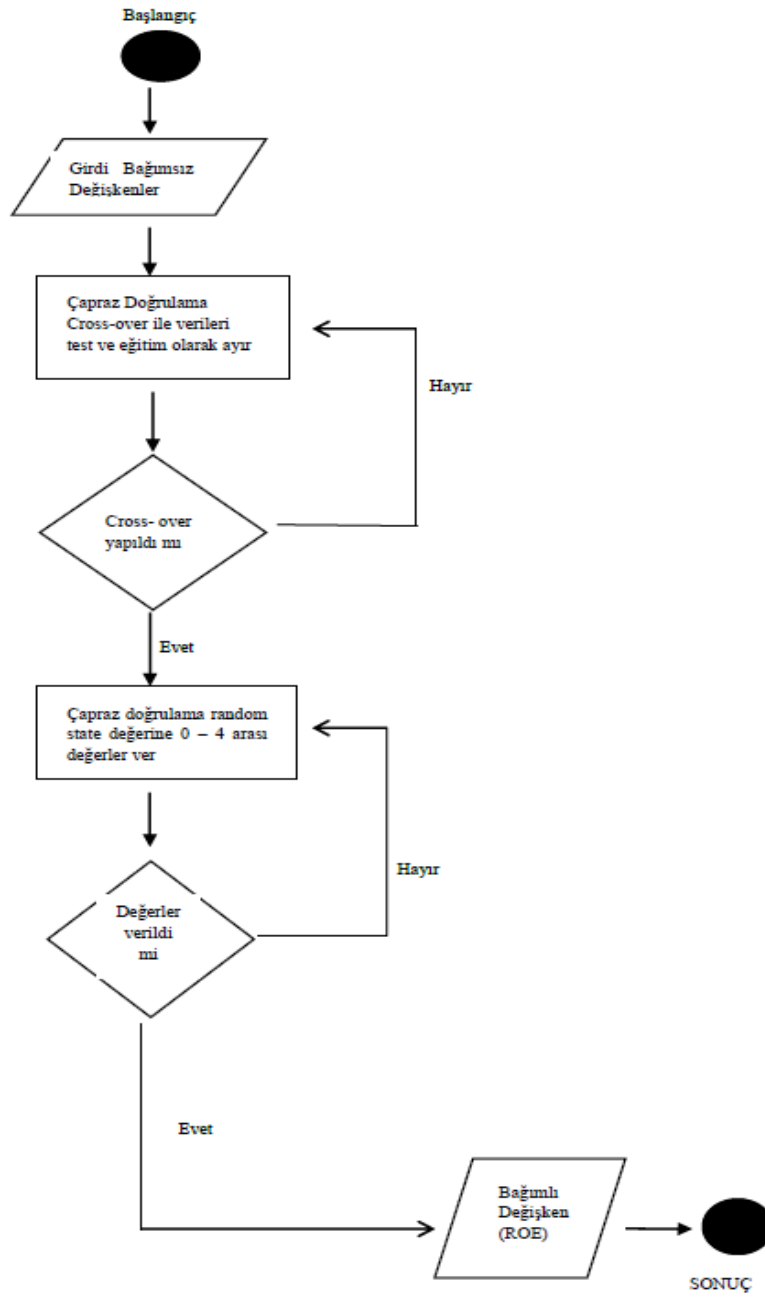
Şekil 4. 7. YSA optimizasyon algoritmaları ile ROE hesaplama akış diyagramı

#### 4.6.2. Linear Regresyon Süreci

Diğer taraftan özkaynak karlılığı yapay sinir ağları ile hesaplanmasının yanında, temel bir algoritma olan Linear Regresyon algoritması ile de hesaplama gerçekleştirilerek algoritmaların performansı karşılaştırılmaya çalışılmıştır. Bu şekilde gerçekleştirilen çalışma ile özkaynak karlılığına yapay zekâ ile istatistiksel algoritmaların hangisinin daha uygun olacağı araştırılmıştır.

Lineer Regresyon ile gerçekleştirilen kısımda bağımlı ve bağımsız verilerimiz sisteme doğrudan sokulacak, bozuk veri kontrolü ve normalizasyon gerçekleştirilmeyecek, çapraz doğrulama ile verilerimiz eğitim ve test veri setleri olarak ayrılarak doğrudan bağımlı değişkenimiz olan özkaynak karlılığı bulunacaktır.

İlgili algoritmanın gerçekleştirilmesini anlatan veri akış diyagramı Şekil 4.8.'de gösterilmiştir.



Şekil 4. 8. Lineer regresyon ile ROE hesaplanmasında akış diyagramı

#### 4.7. Bulgular

Bu bölümde, optimizasyon algoritmaları ile yapılan deneylerin çıktıları gösterilecektir. Öncelikle optimizasyon algoritmaları kullanılarak her bankanın ROE'si tahmin edilecek ve sonuçlar gerçek ROE değerleri ile birlikte grafik üzerinde gösterilecektir. Soldaki grafiklerde ilgili algoritmanın her bir banka için ürettiği tahmini ROE değerleri ile bankanın gerçek ROE değerleri gösterilmektedir. Sağdaki grafikte ise tahmini değerlerin gerçek değerlere yakınlığını ifade eden serpilme diyagramı gösterilmektedir.

Serpilme diyagramı kısaca iki veri arasındaki doğrusal ilişkinin görülmesini sağlayan diyagramlar olarak ifade edilebilir. Serpilme diyagramları 6 farklı şekilde yorumlanmaktadır. Bunlar güçlü pozitif ilişki, zayıf pozitif ilişki, güçlü negatif ilişki, zayıf negatif ilişki, karmaşık ilişki ve ilişki yoktur şeklindedir. Pozitif ilişkide değişkenlerin değerleri diyagonal çizginin yanında kümelenmiştir. Zayıf pozitif ilişki de noktalar diyagonal çizginin etrafında ama dağınıktır. Güçlü negatif ilişki de noktalar çizginin yakınında ancak bir değer artarken diğer değer azalmaktadır. Zayıf negatif ilişkide değişkenlerden biri artarken diğeri azalır ve noktalar dağınık haldedir. Karmaşık ilişkide değişkenler arası ilişki var olmasına rağmen ilişki ilk bakışta kestirilememektedir. İlişki yokta ise değişkenlerin arasında herhangi bir ilişki bulunmamaktadır.

Bankalar için yapay sinir ağları model sonuçlarını gösteren sonuç grafikleri ile serpilme grafikleri aşağıdadır. Soldaki grafiklerde bankaların gerçek ROE değerleri ile yapay sinir ağları tarafından tahmin edilen hesaplanan değerler gösterilmektedir. Sağdaki grafiklerde ise bankaların gerçek ROE değerleri ve tahmini ROE değerleri arasındaki ilişkinin yönünü ve derecesini gösteren serpilme grafiği verilmektedir.

Sonuç grafiğinde mavi çizgi ile gösterilen dönemsel olarak bankanın gerçek ROE değerleri iken; kırmızı çizgi ile gösterilen modelin tahmin ettiği ROE değerlerinin zaman düzleminde karşılaştırılmış halidir.

Serpilme grafiğinde zamansal düzlemin yok sayılarak YSA'nın tahmin ettiği bankaların ROE değerleri ile bankanın gerçek değerleri ile karşılaştırılmaktadır. Serpilme grafiğinde diyagonal çizgi ile gösterilen gerçek ROE değerleri iken; mavi ile belirtilmiş noktalar yapay sinir ağlarının aynı dönem için tahmin ettiği ROE değerleridir. Buna göre grafikte hesaplanan tahmini değerlerin diyagonal çizgiye yakın veya uzak olmasına göre

gerçek deęer ile hesaplanan deęerler arasında iliřkili olup olmadıęı ve iliřki varsa yn ve derecesi gsterilerek ifade edilecektir.

Sonuç grafiklerinde ADAM, SGD, RMSprop ve Lineer regresyon metotları kullanılarak hesaplanan doęruluk deęerleri incelendięinde en yksek ve en dřk doęruluk deęerleri sırasıyla ADAM metot iin Ziraat Bankası ve Garanti Bankası; SGD metot iin Vakıfbank ve Yapı Kredi Bankası; RMSprop metodu iin Vakıfbank ve Garanti Bankası; Lineer regresyon metodu iin Ziraat Bankası ve Garanti Bankası'dır.

Bankalara ait serpilme grafiklerinde ise bankaların tahmini ROE deęerlerinin diyagonal izgiye gre durumları gsterilmektedir. Bu grafiklerde her banka iin gerek ROE deęerleri ile tahmini ROE deęerleri arasındaki iliřkinin yn ve nitelięi incelenmiřtir. Sonuç olarak ADAM metodunda Garanti Bankası; SGD metot iin Yapı Kredi Bankası ve Garanti Bankası; RMSprop metodu iin Garanti Bankası zayıf ve pozitif iliřkiye sahipken dięer tm bankalar ilgili metotlarda gl pozitif iliřkiye sahiptir.

Tm bankalar iin kullanılan modellere ait doęruluk deęer sonuları toplu řekilde Tablo 4.6.'da gsterilmiřtir.

Tablo 4. 6. Modellerin doęruluk deęer sonuları

<b>Banka</b>	ADAM	SGD	RMSprop	Lineer Regresyon
Vakıf Bank	92,36	91,96	93,29	85,70
Halk Bank	85,17	89,06	89,99	89,44
Ziraat Bank	94,42	91,03	91,12	89,81
Yapı Kredi	86,62	82,94	83,18	73,00
Garanti Bankası	80,14	83,50	80,51	72,33
İř Bankası	87,70	86,21	89,03	88,25
Akbank	91,43	88,51	90,14	88,40
Ortalama	88,26	87,60	88,18	83,84

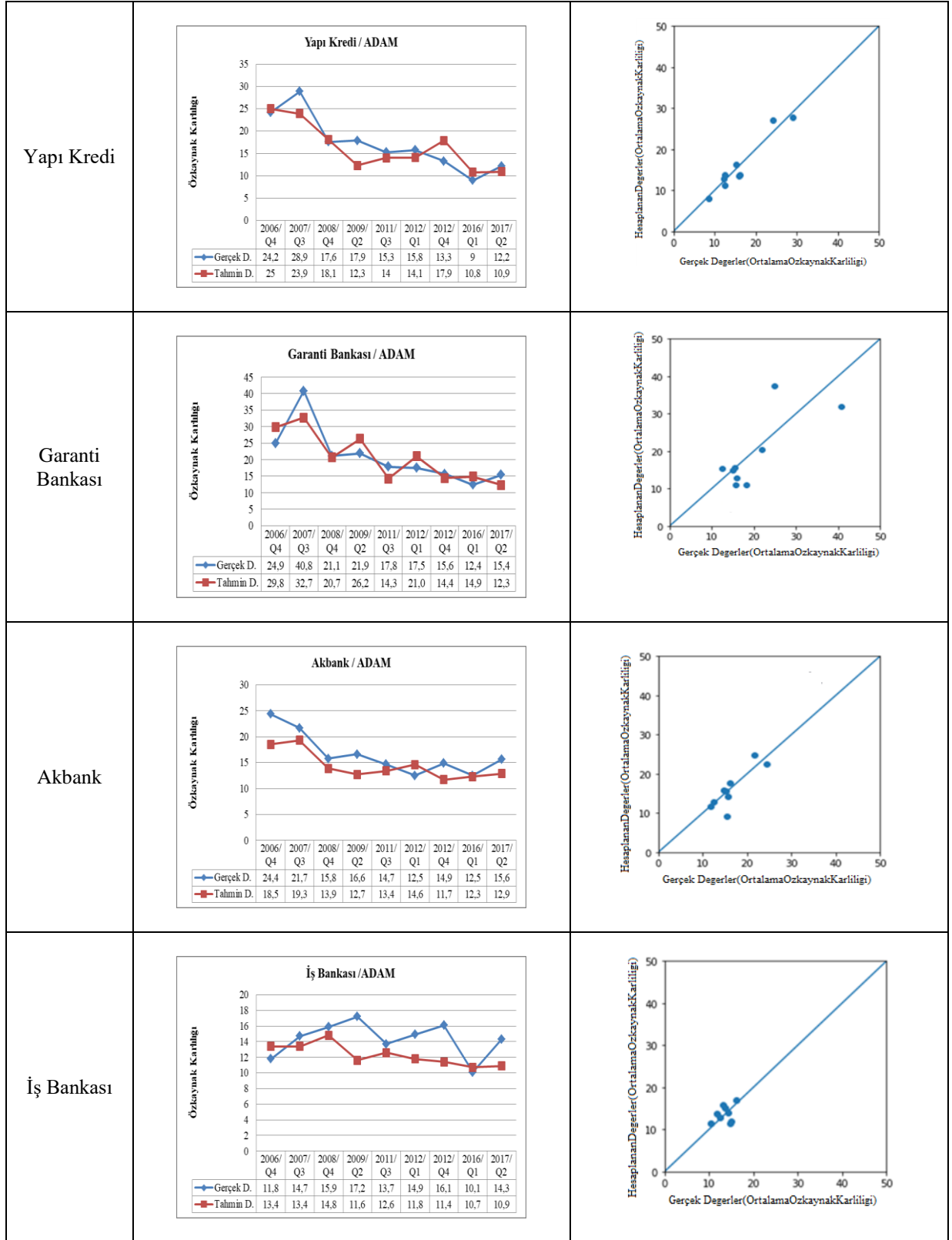


#### 4.7.1. Uyarlamalı moment kestirimi metodu ile ROE hesaplanması

Bankalara ait ADAM Metot ile gerçekleştirilen ROE tahminlerinin sonuç grafikleri ile serpilme grafikleri Tablo 4.7.'de verilmiştir.

Tablo 4. 7. ADAM metoduna göre sonuç ve serpilme grafikleri

Banka İsmi	Sonuç Grafiği	Serpilme Grafiği																														
Vakıfbank	<p><b>Vakıfbank / ADAM</b></p> <table border="1"> <thead> <tr> <th>2006/2007/2008/2009/2011/2012/2012/2016/2017/</th> <th>Q4</th> <th>Q3</th> <th>Q4</th> <th>Q2</th> <th>Q3</th> <th>Q1</th> <th>Q4</th> <th>Q1</th> <th>Q2</th> </tr> </thead> <tbody> <tr> <td>Gerçek D.</td> <td>18,6</td> <td>21,1</td> <td>14</td> <td>14,8</td> <td>15,3</td> <td>13,3</td> <td>13,6</td> <td>12,4</td> <td>18,5</td> </tr> <tr> <td>Tahmin D.</td> <td>17,2</td> <td>22,7</td> <td>12,9</td> <td>15,9</td> <td>18,7</td> <td>14,3</td> <td>12,6</td> <td>14,7</td> <td>17,1</td> </tr> </tbody> </table>	2006/2007/2008/2009/2011/2012/2012/2016/2017/	Q4	Q3	Q4	Q2	Q3	Q1	Q4	Q1	Q2	Gerçek D.	18,6	21,1	14	14,8	15,3	13,3	13,6	12,4	18,5	Tahmin D.	17,2	22,7	12,9	15,9	18,7	14,3	12,6	14,7	17,1	
2006/2007/2008/2009/2011/2012/2012/2016/2017/	Q4	Q3	Q4	Q2	Q3	Q1	Q4	Q1	Q2																							
Gerçek D.	18,6	21,1	14	14,8	15,3	13,3	13,6	12,4	18,5																							
Tahmin D.	17,2	22,7	12,9	15,9	18,7	14,3	12,6	14,7	17,1																							
Halk Bank	<p><b>Halkbank / ADAM</b></p> <table border="1"> <thead> <tr> <th>2006/2007/2008/2009/2011/2012/2012/2016/2017/</th> <th>Q4</th> <th>Q3</th> <th>Q4</th> <th>Q2</th> <th>Q3</th> <th>Q1</th> <th>Q4</th> <th>Q1</th> <th>Q2</th> </tr> </thead> <tbody> <tr> <td>Gerçek D.</td> <td>25,6</td> <td>27,1</td> <td>24</td> <td>25,6</td> <td>26</td> <td>23,8</td> <td>23,8</td> <td>12,7</td> <td>14,4</td> </tr> <tr> <td>Tahmin D.</td> <td>32,7</td> <td>26,9</td> <td>20,8</td> <td>18,8</td> <td>23,4</td> <td>21,8</td> <td>29,3</td> <td>14,2</td> <td>11,8</td> </tr> </tbody> </table>	2006/2007/2008/2009/2011/2012/2012/2016/2017/	Q4	Q3	Q4	Q2	Q3	Q1	Q4	Q1	Q2	Gerçek D.	25,6	27,1	24	25,6	26	23,8	23,8	12,7	14,4	Tahmin D.	32,7	26,9	20,8	18,8	23,4	21,8	29,3	14,2	11,8	
2006/2007/2008/2009/2011/2012/2012/2016/2017/	Q4	Q3	Q4	Q2	Q3	Q1	Q4	Q1	Q2																							
Gerçek D.	25,6	27,1	24	25,6	26	23,8	23,8	12,7	14,4																							
Tahmin D.	32,7	26,9	20,8	18,8	23,4	21,8	29,3	14,2	11,8																							
Ziraat Bankası	<p><b>Ziraat Bankası / ADAM</b></p> <table border="1"> <thead> <tr> <th>2006/2007/2008/2009/2011/2012/2012/2016/2017/</th> <th>Q4</th> <th>Q3</th> <th>Q4</th> <th>Q2</th> <th>Q3</th> <th>Q1</th> <th>Q4</th> <th>Q1</th> <th>Q2</th> </tr> </thead> <tbody> <tr> <td>Gerçek D.</td> <td>36,8</td> <td>38,6</td> <td>31,4</td> <td>36,6</td> <td>19</td> <td>16,4</td> <td>17,1</td> <td>18,2</td> <td>18,8</td> </tr> <tr> <td>Tahmin D.</td> <td>34,7</td> <td>40,8</td> <td>35,6</td> <td>34,6</td> <td>23,6</td> <td>15,5</td> <td>18,1</td> <td>17,2</td> <td>17,5</td> </tr> </tbody> </table>	2006/2007/2008/2009/2011/2012/2012/2016/2017/	Q4	Q3	Q4	Q2	Q3	Q1	Q4	Q1	Q2	Gerçek D.	36,8	38,6	31,4	36,6	19	16,4	17,1	18,2	18,8	Tahmin D.	34,7	40,8	35,6	34,6	23,6	15,5	18,1	17,2	17,5	
2006/2007/2008/2009/2011/2012/2012/2016/2017/	Q4	Q3	Q4	Q2	Q3	Q1	Q4	Q1	Q2																							
Gerçek D.	36,8	38,6	31,4	36,6	19	16,4	17,1	18,2	18,8																							
Tahmin D.	34,7	40,8	35,6	34,6	23,6	15,5	18,1	17,2	17,5																							

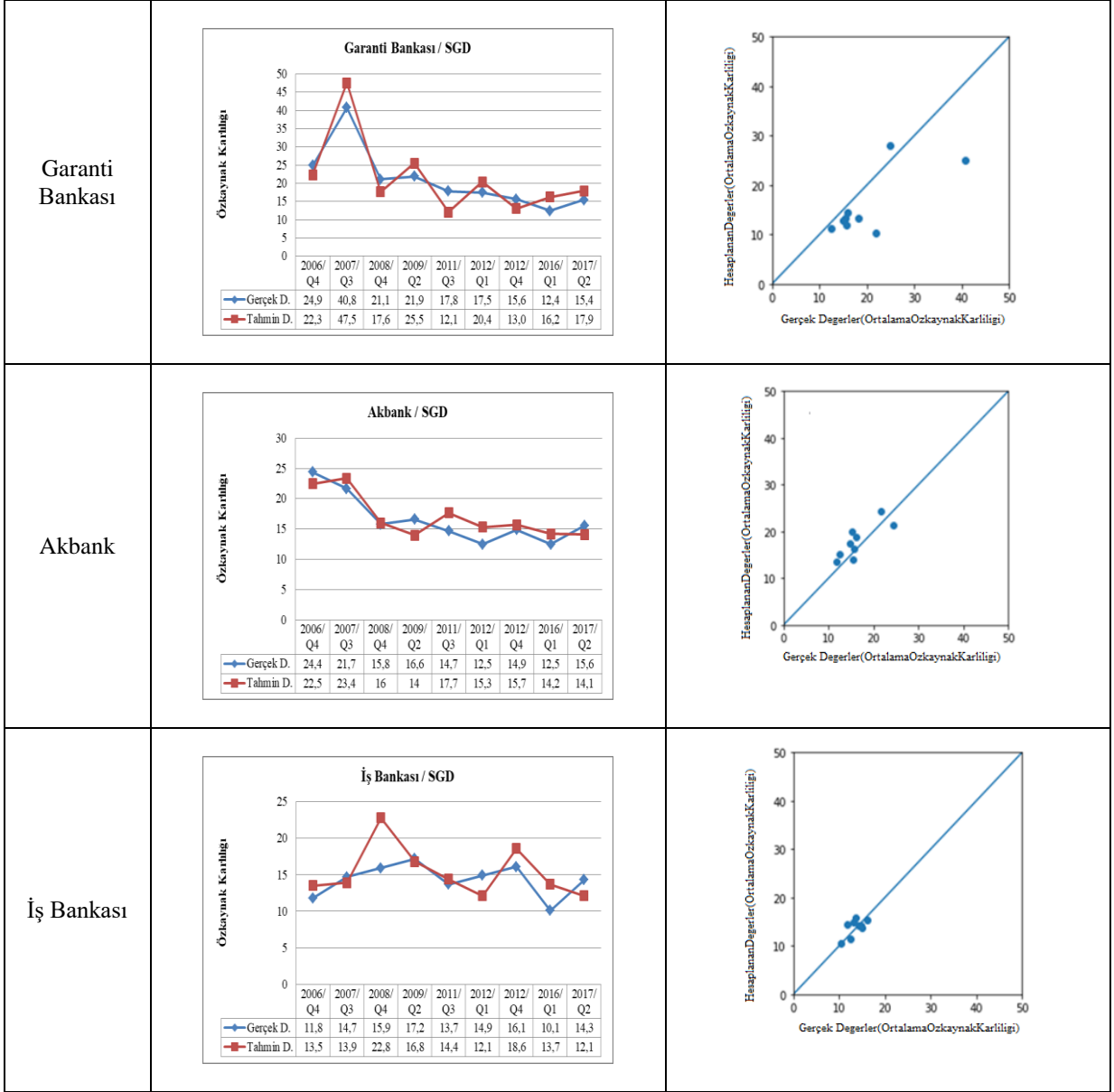


#### 4.7.2. Stokastik gradyan inişi metodu ile ROE hesaplanması

Bankalara ait SGD Metot ile gerçekleştirilen ROE tahminlerinin sonuç grafikleri ile serpilme grafikleri Tablo 4.8.'de verilmiştir.

Tablo 4. 8. SGD metoduna göre sonuç ve serpilme grafikleri

Banka İsmi	Sonuç Grafiği	Serpilme Grafiği																											
Vakıfbank	<p><b>Vakıfbank / SGD</b></p> <table border="1"> <thead> <tr> <th>2006/2007 Q4</th> <th>2007/2008 Q3</th> <th>2008/2009 Q4</th> <th>2009/2010 Q2</th> <th>2010/2011 Q3</th> <th>2011/2012 Q1</th> <th>2012/2013 Q4</th> <th>2016/2017 Q1</th> <th>2017/2018 Q2</th> </tr> </thead> <tbody> <tr> <td>18,6</td> <td>21,1</td> <td>14</td> <td>14,8</td> <td>15,3</td> <td>13,3</td> <td>13,6</td> <td>12,4</td> <td>18,5</td> </tr> <tr> <td>17,1</td> <td>22,8</td> <td>10,7</td> <td>16,0</td> <td>14,1</td> <td>12,2</td> <td>17,8</td> <td>11,4</td> <td>20,0</td> </tr> </tbody> </table>	2006/2007 Q4	2007/2008 Q3	2008/2009 Q4	2009/2010 Q2	2010/2011 Q3	2011/2012 Q1	2012/2013 Q4	2016/2017 Q1	2017/2018 Q2	18,6	21,1	14	14,8	15,3	13,3	13,6	12,4	18,5	17,1	22,8	10,7	16,0	14,1	12,2	17,8	11,4	20,0	
2006/2007 Q4	2007/2008 Q3	2008/2009 Q4	2009/2010 Q2	2010/2011 Q3	2011/2012 Q1	2012/2013 Q4	2016/2017 Q1	2017/2018 Q2																					
18,6	21,1	14	14,8	15,3	13,3	13,6	12,4	18,5																					
17,1	22,8	10,7	16,0	14,1	12,2	17,8	11,4	20,0																					
Halk Bank	<p><b>Halkbank / SGD</b></p> <table border="1"> <thead> <tr> <th>2006/2007 Q4</th> <th>2007/2008 Q3</th> <th>2008/2009 Q4</th> <th>2009/2010 Q2</th> <th>2010/2011 Q3</th> <th>2011/2012 Q1</th> <th>2012/2013 Q4</th> <th>2016/2017 Q1</th> <th>2017/2018 Q2</th> </tr> </thead> <tbody> <tr> <td>25,6</td> <td>27,1</td> <td>24</td> <td>25,6</td> <td>26</td> <td>23,8</td> <td>23,8</td> <td>12,7</td> <td>14,4</td> </tr> <tr> <td>23,7</td> <td>30,2</td> <td>23,6</td> <td>22,7</td> <td>29,5</td> <td>23,4</td> <td>22,8</td> <td>15,6</td> <td>14</td> </tr> </tbody> </table>	2006/2007 Q4	2007/2008 Q3	2008/2009 Q4	2009/2010 Q2	2010/2011 Q3	2011/2012 Q1	2012/2013 Q4	2016/2017 Q1	2017/2018 Q2	25,6	27,1	24	25,6	26	23,8	23,8	12,7	14,4	23,7	30,2	23,6	22,7	29,5	23,4	22,8	15,6	14	
2006/2007 Q4	2007/2008 Q3	2008/2009 Q4	2009/2010 Q2	2010/2011 Q3	2011/2012 Q1	2012/2013 Q4	2016/2017 Q1	2017/2018 Q2																					
25,6	27,1	24	25,6	26	23,8	23,8	12,7	14,4																					
23,7	30,2	23,6	22,7	29,5	23,4	22,8	15,6	14																					
Ziraat Bankası	<p><b>Ziraat Bankası / SGD</b></p> <table border="1"> <thead> <tr> <th>2006/2007 Q4</th> <th>2007/2008 Q3</th> <th>2008/2009 Q4</th> <th>2009/2010 Q2</th> <th>2010/2011 Q3</th> <th>2011/2012 Q1</th> <th>2012/2013 Q4</th> <th>2016/2017 Q1</th> <th>2017/2018 Q2</th> </tr> </thead> <tbody> <tr> <td>36,8</td> <td>38,6</td> <td>31,4</td> <td>36,6</td> <td>19</td> <td>16,4</td> <td>17,1</td> <td>18,2</td> <td>18,8</td> </tr> <tr> <td>33,5</td> <td>42,1</td> <td>25,4</td> <td>39,9</td> <td>17,3</td> <td>19,6</td> <td>15,6</td> <td>22,1</td> <td>17,1</td> </tr> </tbody> </table>	2006/2007 Q4	2007/2008 Q3	2008/2009 Q4	2009/2010 Q2	2010/2011 Q3	2011/2012 Q1	2012/2013 Q4	2016/2017 Q1	2017/2018 Q2	36,8	38,6	31,4	36,6	19	16,4	17,1	18,2	18,8	33,5	42,1	25,4	39,9	17,3	19,6	15,6	22,1	17,1	
2006/2007 Q4	2007/2008 Q3	2008/2009 Q4	2009/2010 Q2	2010/2011 Q3	2011/2012 Q1	2012/2013 Q4	2016/2017 Q1	2017/2018 Q2																					
36,8	38,6	31,4	36,6	19	16,4	17,1	18,2	18,8																					
33,5	42,1	25,4	39,9	17,3	19,6	15,6	22,1	17,1																					
Yapı Kredi	<p><b>Yapı Kredi / SGD</b></p> <table border="1"> <thead> <tr> <th>2006/2007 Q4</th> <th>2007/2008 Q3</th> <th>2008/2009 Q4</th> <th>2009/2010 Q2</th> <th>2010/2011 Q3</th> <th>2011/2012 Q1</th> <th>2012/2013 Q4</th> <th>2016/2017 Q1</th> <th>2017/2018 Q2</th> </tr> </thead> <tbody> <tr> <td>24,2</td> <td>28,9</td> <td>17,6</td> <td>17,9</td> <td>15,3</td> <td>15,8</td> <td>13,3</td> <td>9</td> <td>12,2</td> </tr> <tr> <td>22,5</td> <td>23,3</td> <td>17,2</td> <td>13,4</td> <td>18,8</td> <td>12,7</td> <td>18,4</td> <td>11,3</td> <td>12,9</td> </tr> </tbody> </table>	2006/2007 Q4	2007/2008 Q3	2008/2009 Q4	2009/2010 Q2	2010/2011 Q3	2011/2012 Q1	2012/2013 Q4	2016/2017 Q1	2017/2018 Q2	24,2	28,9	17,6	17,9	15,3	15,8	13,3	9	12,2	22,5	23,3	17,2	13,4	18,8	12,7	18,4	11,3	12,9	
2006/2007 Q4	2007/2008 Q3	2008/2009 Q4	2009/2010 Q2	2010/2011 Q3	2011/2012 Q1	2012/2013 Q4	2016/2017 Q1	2017/2018 Q2																					
24,2	28,9	17,6	17,9	15,3	15,8	13,3	9	12,2																					
22,5	23,3	17,2	13,4	18,8	12,7	18,4	11,3	12,9																					

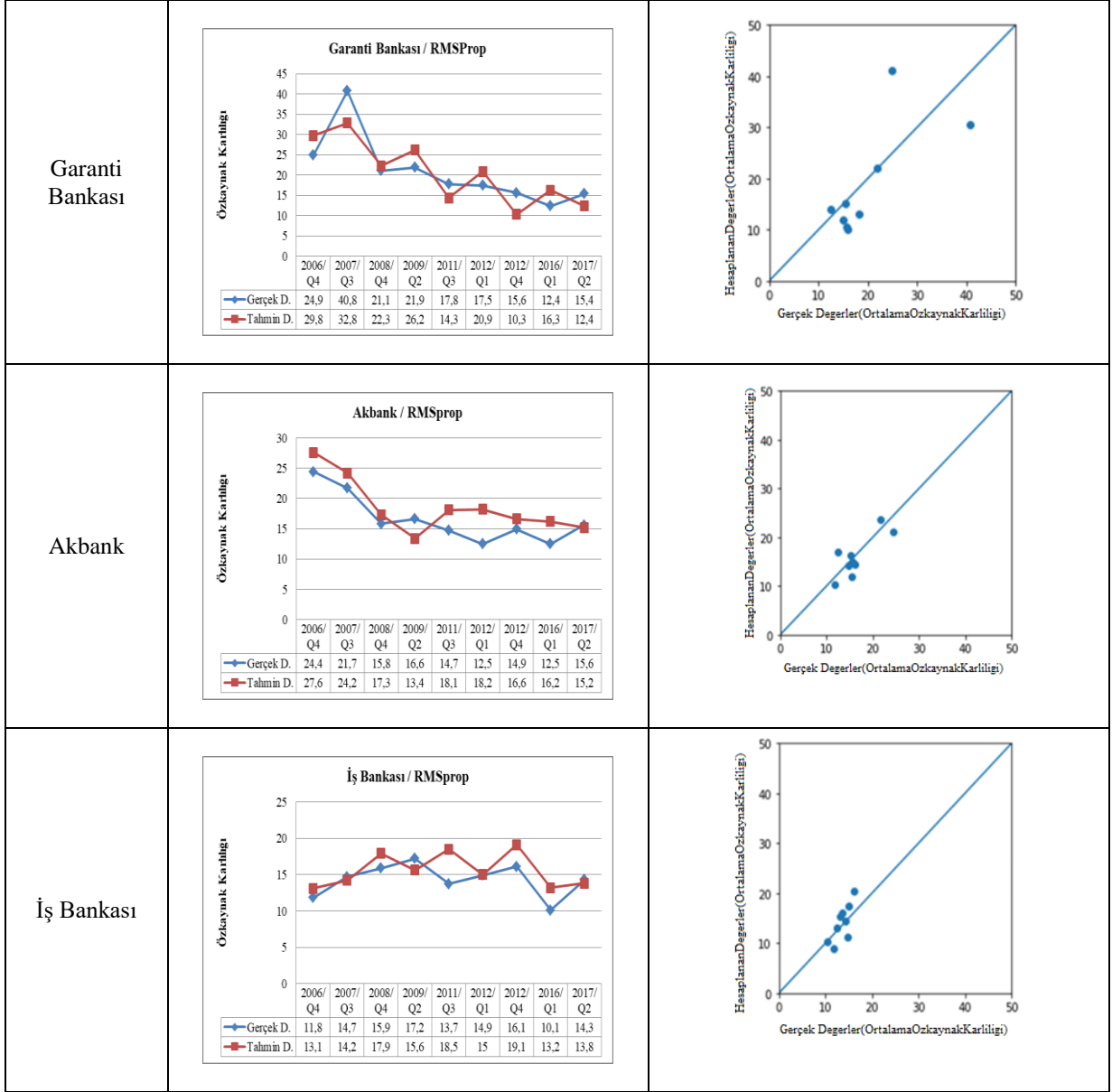


#### 4.7.3. Karesel ortalamanın karekökü yayılımı metodu ile ROE hesaplanması

Bankalara ait RMSprop Metot ile gerçekleştirilen ROE tahminlerinin sonuç grafikleri ile serpilme grafikleri Tablo 4.9.'da verilmiştir.

Tablo 4. 9. RMSprop metoduna göre sonuç ve serpilme grafikleri

Banka İsmi	Sonuç Grafiği	Serpilme Grafiği																											
Vakıfbank	<p><b>Vakıfbank / RMSprop</b></p> <table border="1"> <thead> <tr> <th>2006/ Q4</th> <th>2007/ Q3</th> <th>2008/ Q4</th> <th>2009/ Q2</th> <th>2011/ Q3</th> <th>2012/ Q1</th> <th>2012/ Q4</th> <th>2016/ Q1</th> <th>2017/ Q2</th> </tr> </thead> <tbody> <tr> <td>Gerçek D.</td> <td>18,6</td> <td>21,1</td> <td>14</td> <td>14,8</td> <td>15,3</td> <td>13,3</td> <td>13,6</td> <td>12,4</td> </tr> <tr> <td>Tahmin D.</td> <td>17,4</td> <td>22,5</td> <td>10,6</td> <td>13,8</td> <td>16,3</td> <td>14,2</td> <td>16,3</td> <td>13,2</td> </tr> </tbody> </table>	2006/ Q4	2007/ Q3	2008/ Q4	2009/ Q2	2011/ Q3	2012/ Q1	2012/ Q4	2016/ Q1	2017/ Q2	Gerçek D.	18,6	21,1	14	14,8	15,3	13,3	13,6	12,4	Tahmin D.	17,4	22,5	10,6	13,8	16,3	14,2	16,3	13,2	
2006/ Q4	2007/ Q3	2008/ Q4	2009/ Q2	2011/ Q3	2012/ Q1	2012/ Q4	2016/ Q1	2017/ Q2																					
Gerçek D.	18,6	21,1	14	14,8	15,3	13,3	13,6	12,4																					
Tahmin D.	17,4	22,5	10,6	13,8	16,3	14,2	16,3	13,2																					
Halk Bank	<p><b>Halkbank / RMSprop</b></p> <table border="1"> <thead> <tr> <th>2006/ Q4</th> <th>2007/ Q3</th> <th>2008/ Q4</th> <th>2009/ Q2</th> <th>2011/ Q3</th> <th>2012/ Q1</th> <th>2012/ Q4</th> <th>2016/ Q1</th> <th>2017/ Q2</th> </tr> </thead> <tbody> <tr> <td>Gerçek D.</td> <td>25,6</td> <td>27,1</td> <td>24</td> <td>25,6</td> <td>26</td> <td>23,8</td> <td>23,8</td> <td>12,7</td> </tr> <tr> <td>Tahmin D.</td> <td>37</td> <td>31,9</td> <td>30</td> <td>20,6</td> <td>25,6</td> <td>25,7</td> <td>28,9</td> <td>13,8</td> </tr> </tbody> </table>	2006/ Q4	2007/ Q3	2008/ Q4	2009/ Q2	2011/ Q3	2012/ Q1	2012/ Q4	2016/ Q1	2017/ Q2	Gerçek D.	25,6	27,1	24	25,6	26	23,8	23,8	12,7	Tahmin D.	37	31,9	30	20,6	25,6	25,7	28,9	13,8	
2006/ Q4	2007/ Q3	2008/ Q4	2009/ Q2	2011/ Q3	2012/ Q1	2012/ Q4	2016/ Q1	2017/ Q2																					
Gerçek D.	25,6	27,1	24	25,6	26	23,8	23,8	12,7																					
Tahmin D.	37	31,9	30	20,6	25,6	25,7	28,9	13,8																					
Ziraat Bankası	<p><b>Ziraat Bankası / RMSprop</b></p> <table border="1"> <thead> <tr> <th>2006/ Q4</th> <th>2007/ Q3</th> <th>2008/ Q4</th> <th>2009/ Q2</th> <th>2011/ Q3</th> <th>2012/ Q1</th> <th>2012/ Q4</th> <th>2016/ Q1</th> <th>2017/ Q2</th> </tr> </thead> <tbody> <tr> <td>Gerçek D.</td> <td>36,8</td> <td>38,6</td> <td>31,4</td> <td>36,6</td> <td>19</td> <td>16,4</td> <td>17,1</td> <td>18,2</td> </tr> <tr> <td>Tahmin D.</td> <td>33,5</td> <td>42,0</td> <td>28,6</td> <td>39,9</td> <td>21,9</td> <td>12,1</td> <td>18,6</td> <td>16,6</td> </tr> </tbody> </table>	2006/ Q4	2007/ Q3	2008/ Q4	2009/ Q2	2011/ Q3	2012/ Q1	2012/ Q4	2016/ Q1	2017/ Q2	Gerçek D.	36,8	38,6	31,4	36,6	19	16,4	17,1	18,2	Tahmin D.	33,5	42,0	28,6	39,9	21,9	12,1	18,6	16,6	
2006/ Q4	2007/ Q3	2008/ Q4	2009/ Q2	2011/ Q3	2012/ Q1	2012/ Q4	2016/ Q1	2017/ Q2																					
Gerçek D.	36,8	38,6	31,4	36,6	19	16,4	17,1	18,2																					
Tahmin D.	33,5	42,0	28,6	39,9	21,9	12,1	18,6	16,6																					
Yapı Kredi	<p><b>Yapı Kredi / RMSprop</b></p> <table border="1"> <thead> <tr> <th>2006/ Q4</th> <th>2007/ Q3</th> <th>2008/ Q4</th> <th>2009/ Q2</th> <th>2011/ Q3</th> <th>2012/ Q1</th> <th>2012/ Q4</th> <th>2016/ Q1</th> <th>2017/ Q2</th> </tr> </thead> <tbody> <tr> <td>Gerçek D.</td> <td>24,2</td> <td>28,9</td> <td>17,6</td> <td>17,9</td> <td>15,3</td> <td>15,8</td> <td>13,3</td> <td>9</td> </tr> <tr> <td>Tahmin D.</td> <td>26,9</td> <td>24,1</td> <td>18,5</td> <td>11</td> <td>15,6</td> <td>12,9</td> <td>20,3</td> <td>10,6</td> </tr> </tbody> </table>	2006/ Q4	2007/ Q3	2008/ Q4	2009/ Q2	2011/ Q3	2012/ Q1	2012/ Q4	2016/ Q1	2017/ Q2	Gerçek D.	24,2	28,9	17,6	17,9	15,3	15,8	13,3	9	Tahmin D.	26,9	24,1	18,5	11	15,6	12,9	20,3	10,6	
2006/ Q4	2007/ Q3	2008/ Q4	2009/ Q2	2011/ Q3	2012/ Q1	2012/ Q4	2016/ Q1	2017/ Q2																					
Gerçek D.	24,2	28,9	17,6	17,9	15,3	15,8	13,3	9																					
Tahmin D.	26,9	24,1	18,5	11	15,6	12,9	20,3	10,6																					

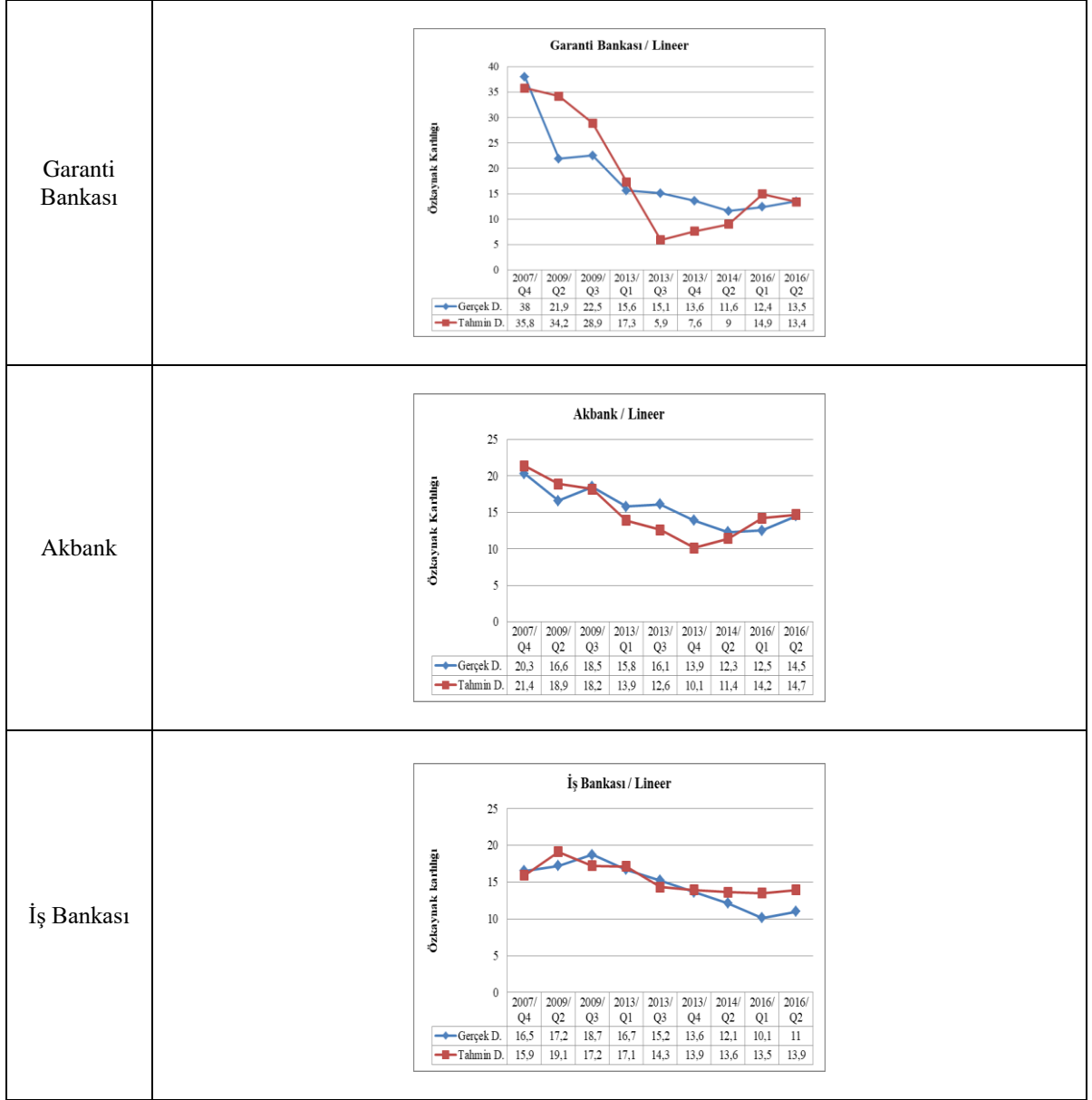


#### 4.7.4. Linear regresyon ile ROE hesaplanması

Bankalara ait Linear Regresyon Metot ile gerçekleştirilen ROE tahminlerinin sonuç grafikleri verilmiştir.

Tablo 4. 10. Lineer regresyon metoduna göre sonuç grafikleri

Banka İsmi	Sonuç Grafiği																														
Vakıfbank	<p style="text-align: center;"><b>Vakıfbank / Lineer</b></p> <table border="1"> <thead> <tr> <th></th> <th>2007/ Q4</th> <th>2009/ Q2</th> <th>2009/ Q3</th> <th>2013/ Q1</th> <th>2013/ Q3</th> <th>2013/ Q4</th> <th>2014/ Q2</th> <th>2016/ Q1</th> <th>2016/ Q2</th> </tr> </thead> <tbody> <tr> <td>Gerçek D.</td> <td>21,6</td> <td>14,8</td> <td>17,9</td> <td>13,8</td> <td>13,4</td> <td>13</td> <td>11</td> <td>12,4</td> <td>13</td> </tr> <tr> <td>Tahmin D.</td> <td>15,9</td> <td>15,9</td> <td>15,5</td> <td>15,2</td> <td>13,1</td> <td>13,4</td> <td>12,5</td> <td>16,3</td> <td>15,5</td> </tr> </tbody> </table>		2007/ Q4	2009/ Q2	2009/ Q3	2013/ Q1	2013/ Q3	2013/ Q4	2014/ Q2	2016/ Q1	2016/ Q2	Gerçek D.	21,6	14,8	17,9	13,8	13,4	13	11	12,4	13	Tahmin D.	15,9	15,9	15,5	15,2	13,1	13,4	12,5	16,3	15,5
	2007/ Q4	2009/ Q2	2009/ Q3	2013/ Q1	2013/ Q3	2013/ Q4	2014/ Q2	2016/ Q1	2016/ Q2																						
Gerçek D.	21,6	14,8	17,9	13,8	13,4	13	11	12,4	13																						
Tahmin D.	15,9	15,9	15,5	15,2	13,1	13,4	12,5	16,3	15,5																						
Halk Bank	<p style="text-align: center;"><b>Halk Bank / Lineer</b></p> <table border="1"> <thead> <tr> <th></th> <th>2007/ Q4</th> <th>2009/ Q2</th> <th>2009/ Q3</th> <th>2013/ Q1</th> <th>2013/ Q3</th> <th>2013/ Q4</th> <th>2014/ Q2</th> <th>2016/ Q1</th> <th>2016/ Q2</th> </tr> </thead> <tbody> <tr> <td>Gerçek D.</td> <td>28,7</td> <td>25,6</td> <td>27,4</td> <td>23,7</td> <td>21,5</td> <td>20,8</td> <td>17,4</td> <td>12,7</td> <td>13,4</td> </tr> <tr> <td>Tahmin D.</td> <td>26,7</td> <td>24,4</td> <td>29,9</td> <td>22,9</td> <td>24,8</td> <td>24,9</td> <td>15,3</td> <td>11,6</td> <td>11,4</td> </tr> </tbody> </table>		2007/ Q4	2009/ Q2	2009/ Q3	2013/ Q1	2013/ Q3	2013/ Q4	2014/ Q2	2016/ Q1	2016/ Q2	Gerçek D.	28,7	25,6	27,4	23,7	21,5	20,8	17,4	12,7	13,4	Tahmin D.	26,7	24,4	29,9	22,9	24,8	24,9	15,3	11,6	11,4
	2007/ Q4	2009/ Q2	2009/ Q3	2013/ Q1	2013/ Q3	2013/ Q4	2014/ Q2	2016/ Q1	2016/ Q2																						
Gerçek D.	28,7	25,6	27,4	23,7	21,5	20,8	17,4	12,7	13,4																						
Tahmin D.	26,7	24,4	29,9	22,9	24,8	24,9	15,3	11,6	11,4																						
Ziraat Bankası	<p style="text-align: center;"><b>Ziraat Bankası / Lineer</b></p> <table border="1"> <thead> <tr> <th></th> <th>2007/ Q4</th> <th>2009/ Q2</th> <th>2009/ Q3</th> <th>2013/ Q1</th> <th>2013/ Q3</th> <th>2013/ Q4</th> <th>2014/ Q2</th> <th>2016/ Q1</th> <th>2016/ Q2</th> </tr> </thead> <tbody> <tr> <td>Gerçek D.</td> <td>34,8</td> <td>36,6</td> <td>37,8</td> <td>17,6</td> <td>18,8</td> <td>18,5</td> <td>16,6</td> <td>18,2</td> <td>18,7</td> </tr> <tr> <td>Tahmin D.</td> <td>36,1</td> <td>33,5</td> <td>33,9</td> <td>18,7</td> <td>16,6</td> <td>16,9</td> <td>15,2</td> <td>16,4</td> <td>14,1</td> </tr> </tbody> </table>		2007/ Q4	2009/ Q2	2009/ Q3	2013/ Q1	2013/ Q3	2013/ Q4	2014/ Q2	2016/ Q1	2016/ Q2	Gerçek D.	34,8	36,6	37,8	17,6	18,8	18,5	16,6	18,2	18,7	Tahmin D.	36,1	33,5	33,9	18,7	16,6	16,9	15,2	16,4	14,1
	2007/ Q4	2009/ Q2	2009/ Q3	2013/ Q1	2013/ Q3	2013/ Q4	2014/ Q2	2016/ Q1	2016/ Q2																						
Gerçek D.	34,8	36,6	37,8	17,6	18,8	18,5	16,6	18,2	18,7																						
Tahmin D.	36,1	33,5	33,9	18,7	16,6	16,9	15,2	16,4	14,1																						
Yapı Kredi	<p style="text-align: center;"><b>Yapı Kredi / Lineer</b></p> <table border="1"> <thead> <tr> <th></th> <th>2007/ Q4</th> <th>2009/ Q2</th> <th>2009/ Q3</th> <th>2013/ Q1</th> <th>2013/ Q3</th> <th>2013/ Q4</th> <th>2014/ Q2</th> <th>2016/ Q1</th> <th>2016/ Q2</th> </tr> </thead> <tbody> <tr> <td>Gerçek D.</td> <td>17,4</td> <td>17,9</td> <td>16,4</td> <td>12,5</td> <td>19,3</td> <td>18,6</td> <td>16,9</td> <td>9</td> <td>10,8</td> </tr> <tr> <td>Tahmin D.</td> <td>25</td> <td>20,5</td> <td>20,8</td> <td>15,1</td> <td>12,8</td> <td>12</td> <td>14,9</td> <td>13,3</td> <td>11,7</td> </tr> </tbody> </table>		2007/ Q4	2009/ Q2	2009/ Q3	2013/ Q1	2013/ Q3	2013/ Q4	2014/ Q2	2016/ Q1	2016/ Q2	Gerçek D.	17,4	17,9	16,4	12,5	19,3	18,6	16,9	9	10,8	Tahmin D.	25	20,5	20,8	15,1	12,8	12	14,9	13,3	11,7
	2007/ Q4	2009/ Q2	2009/ Q3	2013/ Q1	2013/ Q3	2013/ Q4	2014/ Q2	2016/ Q1	2016/ Q2																						
Gerçek D.	17,4	17,9	16,4	12,5	19,3	18,6	16,9	9	10,8																						
Tahmin D.	25	20,5	20,8	15,1	12,8	12	14,9	13,3	11,7																						



Yapay sinir ağları optimizasyon algoritmalarının büyük çoğunluğunda Lineer Regresyon algoritmasından daha yüksek yüzde ile sonuç elde edilmiştir. Bu durumda genel olarak literatüre uygun bir sonuç olarak görülmektedir.



## 5. GELECEK ÇALIŞMALAR

Tezde çeşitli girdi parametreleri kullanılarak özkaynak karlılığı tahmin edilmiştir. Bu bağlamda çalışma aşağıda belirtilen konular kapsamında genişletilerek gelecek çalışmalar için yol gösterebilir.

- Çalışma mevduat bankalarından en büyük aktife sahip 7 banka seçilerek hazırlanmıştır. Bu bankalara başka bankalar da eklenerek örneklem genişletilebilir.
- Gelecekte araştırmalarda; gram külçe altın, açık piyasa, kredi riski, yıllık enflasyon oranı, Borsa İstanbul endeksi, yoğunlaşma, para arzı, vergilendirme, şüpheli krediler gibi değişkenleri de kullanarak çalışma genişletilebilir.
- Bankaları bazı önemli niteliklere göre gruplandırarak daha başarılı sonuçlar elde edilebilir.
- Çalışma benzer değişkenler esas alınarak başka ülkelere de uygulanabilir.
- Bu çalışmada, mikroekonomik ve makroekonomik değişkenler bütününe bankacılık karlılığını nasıl etkilediği sorusu ele alınmıştır. Gelecek çalışmalarda ayrı-ayrı değişkenlerin banka karlılığını olumlu veya olumsuz etkilediği, işaretlerinin (artı veya eksi) ne olduğu araştırılabilir ve literatürde yer alan sonuçlarla karşılaştırılabilir.
- Kriz, salgın hastalıklar gibi nedenlerle ekonominin darboğaza girmesi, döviz kurlarındaki anlık oynamalar, daraltıcı ve genişletici hükümet politikaları nedeniyle para arzındaki artış veya azalışlar, enflasyondaki değişimler gibi nedenlerden dolayı geleceğe yönelik tahminler yapmak zordur. Tezde gerçekleştirilen yöntemle dayanarak bu yönde kestirimler ve değerlendirmeler üretilebilir.

## 6. TARTIŞMA ve SONUÇ

Bankalar, bir ülkenin finansal ve ekonomik kalkınması için en önemli faktörlerden biridir. Bu nedenle ekonominin sağlıklı ilerleyebilmesi ve istikrarın sürdürülebilmesi için bankacılık sektörünün güçlü bir şekilde varlığını sürdürebilmesi gerekmektedir. Bu sürdürülebilirlik, özellikle mevduat toplayarak karlılıklarını ve pazar paylarını artırmaya çalışan, kredilendirme dışında çeşitli fonlar üreten, türev piyasalar ve hisse senedi piyasaları gibi pek çok alanda faaliyet gösteren bankalar tarafından gerçekleştirilmektedir.

1980 sonrası liberalleşme politikasından başlayarak bankacılık sektöründe büyük bir değişiklik olmuştur. Türkiye'de bankacılık sektörü genişlemiş ve bu dönemde yabancı bankaların Türkiye'de faaliyet göstermesine izin verilmiştir. Bankalar, sadece geleneksel bankacılık faaliyetlerinde (mevduat ve kredi) değil, aynı zamanda repo, menkul kıymetleştirme, denizaşırı faaliyetler vb. için de yüksek kar elde etmeyi hedeflemişlerdir.

Türkiye'deki bu durumun varlığı ve ülkenin dünyadaki diğer devletlerin durumundan etkilenmesiyle yerel veya küresel krizlere girilmiştir. 2000-2001 krizi ve 2008 küresel krizi, Türkiye'nin sırasıyla yerel ve küresel krizlerinin bir örneğidir. Bu krizler sonucunda karşılaşılan zorluklar neticesinde bilim insanları çalışmalarında çeşitli çözümler üretmeye çalışmışlardır.

Özellikle bankaların karlılığı üzerine yapılan çalışmalar makine öğreniminden bilgisayar bilimlerine kadar farklı alanlarda kendilerine yer bulmuştur. Banka karlılığı söz konusu olduğunda, çalışmalar genellikle varlık karlılığına (ROA), öz sermaye karlılığına (ROE) ve net faiz marjına (NIM) odaklanmasına rağmen, özellikle literatürde öz sermaye karlılığı ve varlık karlılığı genellikle çalışmalarda daha çok kullanılmaktadır. Bu bağlamda merkez bankalarının ve denetim otoritelerinin karlılığı varlık karlılığı ve öz sermaye getirisine yeniden odaklamaları, araştırmacıları bu iki kritere yönlendirmiştir.

Banka karlılığını hesaplarken, birçok uzman genellikle Lineer Regresyon Analizi, Lojistik Regresyon Analizi, Tam Lojistik Düzeltme, Çoklu Ayırma Analizi, Çoklu Regresyon Analizi gibi istatistiksel modeller kullanır. Bu teknikler sayesinde bankaların karlılığını tahmin etmek mümkün olsa da bu çok değişkenli istatistiksel teknikler bazı varsayımlar gerektirdiğinden dolayı gerçeği tam olarak yansıtamadıkları görülmüştür. Bu

durumun varlığı, daha tutarlı ve etkili modellerin bir zorunluluk haline gelmesine neden olmuştur.

Son yıllarda banka karlılığının hesaplanmasında daha yeni tekniklerin kullanıldığı gözlemlenmiştir. Özellikle, yapay zekâ alanındaki gelişmeler sayesinde yapay sinir ağları (YSA) veya genetik algoritmalar (GA) kullanılarak banka karlılığını hesaplamak mümkün hale gelmiştir. Bu bağlamda tezde Yapay Sinir Ağları üzerine odaklanılmıştır.

Tezde, 2006 ile 2017 yılları arasında 3'er aylık veriler kullanılarak, Türkiye'deki mevduat bankalarının özkaynak karlılığı tahmin edilmiştir. Tezde, mevduat bankalarından Ziraat Bankası, Vakıfbank, Halk Bankası, Garanti Bankası, İş Bankası, Yapı Kredi Bankası ve Akbank verileri kullanılmıştır. Veri kümesi olarak Türkiye Bankalar Birliği, Türkiye Cumhuriyet Merkez Bankası, Türkiye İstatistik Kurumu, Bankacılık Düzenleme ve Denetleme Kurumu kullanılmıştır. 8 içsel bağımsız değişken, 6 dışsal bağımsız değişken ve 1 bağımlı değişken kullanılarak çalışma gerçekleştirilmiştir. Bu banka verileri üzerinde, bankanın daha iyi sonuçlar elde ettiği üç yapay sinir ağı optimizasyon algoritması olan ADAM, Stokastik Gradyan İnişi ve RMSprop üzerine çalışmada eğilinmiştir. Çalışmalarda bulduğumuz sonuçlar Tablo 4.6.'da toplu şekilde gösterilmiş olup; bulunan doğruluk değerleri % 80,14 ile % 94,42 arasında olduğu görülmüştür. Ayrıca modellerin sonuçları incelendiğinde bankaların gerçek özkaynak karlılığı değerleri ile tahmini özkaynak karlılıklarının birbirlerine paralel ve yakın değerler gösterdiği anlaşılmıştır. Bu çalışma sayesinde yapay sinir ağları kullanılarak banka karlılığının tahmin edilebileceği ve finansal alanlarda çalışan araştırmacılar için yapay sinir ağlarını ve optimizasyon algoritmalarını kullanmayı tercih edebilecekleri anlaşılmaktadır.

Tezde gerçekleştirilen yöntemlerin sonuçları incelendiğinde; gerçek ve tahmini ROE değerleri arasında en yüksek doğruluk değerleri ADAM yönteminin kullanılması sonucunda gerçekleşmiştir. Bununla birlikte bazı bankalar için RMSprop yönteminin, bazı bankalar içinse Stokastik Gradyan İnişi algoritmasının en yüksek doğruluk değerlerine ulaştığı da anlaşılmaktadır. Literatür taraması kısmında bahsedildiği üzere farklı etki alanlarında gerçekleştirilen çalışmalarda da tezde kullandığımız algoritmaların iyi performans sergiledikleri görülmüştür. Bu durumun varlığı tez ile literatür örnekleri arasında genel bir benzerlik ve uyumluluk olduğu gerçeğini yansıtmaktadır.

Tez, Türkiye'de yapay sinir ağları yöntemi kullanılarak bankaların özkaynak karlılığı tahmininin gerçekleştirilmesinde literatüre bir yenilik katmaktadır. Sonraki çalışmalarda

çeşitli değişkenler ve optimizasyon yöntemleri kullanılarak, Türkiye’de ve dünyada bankaların gelecekteki önemli oranlarının doğru tahmin edilebilmesinin sağlanabileceği düşünülmektedir.

Sonuç olarak;

- Türkiye'deki bankaların karlılığının hesaplanmasına dönük tahminlerde hem yapay sinir ağları yöntemlerinin hem de istatistiksel yöntemlerin kullanılması uygundur.
- Yapılan çalışmalarda en etkili yöntem ADAM yöntemi olup ilk tercih edilmesi gereken yapay sinir ağı modeli olarak belirlenmiştir.
- Daha sonra sırasıyla RMSprop ve Stokastik Gradyan İnişi optimizasyon algoritmaları kullanılmasının uygun olduğu belirlenmiştir.
- Bu üç yapay sinir ağı modelleri Lineer Regresyon yöntemine göre daha iyi sonuçlar vermiştir.
- Tüm yöntemlerin bankaların geleceğe dönük karlılık oranlarının tahmininde de kullanılabileceği düşünülmektedir.

## KAYNAKLAR

- [1] V. Ömürbek, Ö. Akçakanat ve E. Aksoy, “Aktif büyüklüklerine göre değerlendirilen büyük ölçekli bankaların yapay sinir ağları ile karlılık tahmini,” *Atatürk Üniversitesi İktisadi ve İdari Bilimler Dergisi*, c. 33, sayı 2, ss. 451-466, 2019.
- [2] F. Sönmez, M. Zontul ve Ş. Bülbül, “Mevduat bankalarının karlılığının yapay sinir ağları ile tahmini: bir yazılım modeli tasarımı,” *BDDK Bankacılık ve Finansal Piyasalar*, c. 9, sayı 1, ss. 9-45, 2015.
- [3] A. Şerbetli, “Makroekonomik göstergelerin banka karlılığı ve sermaye/aktif getirisi üzerindeki etkisinin yapay sinir ağları ile test edilmesi,” Yüksek lisans tezi, Sosyal Bilimler Enstitüsü, İşletme Anabilim Dalı, Marmara Üniversitesi, İstanbul, Türkiye, 2008.
- [4] M. H. Al-Osaimy, “A neural networks system for predicting Islamic banks performance,” *Journal of King Abdulaziz University: Islamic Economic & Administration*, vol. 11, no. 1, pp. 33-46, 1998.
- [5] N. M. A. Bakar and I. M. Tahir, “Applying multiple linear regression and neural network to predict bank performance,” *International Business Research*, vol. 2, no. 4, pp. 176-183, 2009.
- [6] T. Olweny and T. M. Shipho, “Effects of banking sectoral factors on the profitability of commercial banks in Kenya,” *Economics and Finance Review*, vol. 1, no. 5, pp. 1-30, 2011.
- [7] N. M. Sapuan, S. Bakar, and H. Ramlan, “Predicting the performance and survival of islamic banks in Malaysia to achieve growth sustainability,” in *Proc. SHS Web of Conferences*, vol. 36, no. 16, 2017, pp.1-11.
- [8] S. Vani and T. M. Rao, “An experimental approach towards the performance assessment of various optimizers on convolutional neural network,” in *Proc. 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*, Tirunelveli, India, Apr. 23-25, 2019, pp. 331-336.

- [9] G. Rajakumaran, N. Venkataraman and R. R. Mukkamala, “Denial of service attack prediction using gradient descent algorithm,” *SN Computer Science*, vol. 1, no. 1, pp. 1-8, 2020.
- [10] E. M. Dogo, O. J. Afolabi, N. I. Nwulu, B. Twala, and C. O. Aigbavboa, “A comparative analysis of gradient descent-based optimization algorithms on convolutional neural networks,” in *Proc. International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS)*, Belgaum, India, Dec. 21–22, 2018, pp. 92-99.
- [11] E. Yazan and M.F. Talu, “Comparison of the stochastic gradient descent based optimization techniques,” *2017 International Artificial Intelligence and Data Processing Symposium (IDAP)*, Malatya, Türkiye, 2017, ss. 1-5.
- [12] G. Papamakarios, “Comparison of modern stochastic optimization algorithms,” *Edinburgh University Press*, 2014.
- [13] S. Shalev-Shwartz, O. Shamir, and S. Shammah, “Failures of gradient-based deep learning,” in *Proc. The 34th International Conference on Machine Learning*, Sydney, Australia, Aug. 6-11, 2017, pp. 3067-3075.
- [14] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” in *Proc. IEEE*, Piscataway, NJ, USA, vol. 86, 1998, pp. 2278–2324.
- [15] M. Jangid and S. Srivastava, “Deep ConvNet with different stochastic optimizations for handwritten devanagari character,” in *Proc. IC4S*, Patong Phuket, Thailand, Oct. 11–12, 2017, pp. 51–60.
- [16] W. Swastika, M.F. Ariyanto, H. Setiawan, and P.L.T. Irawan, “Appropriate CNN architecture and optimizer for vehicle type classification system on the toll road,” *Journal of Physics: Conference Series*, vol. 1196, pp.012044, March 2019.
- [17] R. Zaheer and H. Shaziya, “A study of the optimization algorithms in deep learning,” in *Proc. 3rd International Conference on Inventive Systems and Control (ICISC)*, Coimbatore, India, Jan. 10-11, 2019, pp. 536-539.

- [18] V. V. Nabiyev, *Yapay Zeka*. Ankara, Türkiye: Seçkin Yayıncılık, 2016.
- [19] S. Haykin, *Neural Networks: A Comprehensive Foundation*. New Jersey, USA: Prentice-Hall Inc., 1999.
- [20] H. H. Örkücü, “Ayırma analizine matematiksel programlama ve yapay sinir ağları yaklaşımları,” Doktora tezi, Fen Bilimleri Enstitüsü, İstatistik Anabilim Dalı, Gazi Üniversitesi, Ankara, Türkiye, 2009.
- [21] G. Dede, “Yapay sinir ağları ile konuşma tanıma,” Yüksek lisans tezi, Fen Bilimleri Enstitüsü, Elektronik Mühendisliği Anabilim Dalı, Ankara Üniversitesi, Ankara, Türkiye, 2008.
- [22] A. Vojvoda and H. Gold, “Neural networks in modelling maintenance unit load status,” *Promet-Traffic&Transportation*, vol. 14, no. 2, pp. 85-91, 2002.
- [23] E. Guresen ve G. Kayakutlu, “Definition of artificial neural networks with comparison to other networks,” *Procedia Computer Science*, vol. 3, pp. 426-433, 2011.
- [24] K. K. Çevik ve E. Dandıl, “Yapay sinir ağları için .Net platformunda görsel bir eğitim yazılımının geliştirilmesi,” *Bilişim Teknolojileri Dergisi*, c. 5, sayı 1, ss. 19-28, 2012.
- [25] A. Arı ve M. E. Berberler, “Yapay sinir ağları ile tahmin ve sınıflandırma problemlerinin çözümü için arayüz tasarımı,” *Acta Infologica*, c. 1, sayı 2, ss. 55-73, 2017.
- [26] F. Rosenblatt, “The perceptron, a perceiving and recognizing automaton Project Para,” *Cornell Aeronautical Laboratory*, 1957.
- [27] B. Widrow and M. A. Lehr, “30 years of adaptive neural networks: perceptron, madaline, and backpropagation,” *Proceedings of the IEEE*, vol. 78, no. 9, pp.1415-1442, 1990.
- [28] G. Ö. R. İclal. “Çok katmanlı algılayıcı yapay sinir ağı ile lineer diferansiyel denklem sisteminin çözümü,” *18. Akademik Bilişim Konferansı*, Adnan Menderes Üniversitesi, Aydın, Türkiye, 2016, ss. 738-745.
- [29] K. Öztürk ve M. E. Şahin, “Yapay sinir ağları ve yapay zekâ’ya genel bir bakış,” *Takvim-i Vekayi*, c. 6, sayı 2, ss. 25-36, 2018.

- [30] M. Er, “Güvenlik ikliminin iş tatmini ve örgütsel bağlılık üzerine etkisi, güvenlik performansının aracılık rolü: Kayseri mobilya imalat çalışanları üzerine bir araştırma,” Doktora tezi, Sosyal Bilimler Enstitüsü, Çalışma Ekonomisi ve Endüstri İlişkileri Anabilim Dalı, Süleyman Demirel Üniversitesi, Isparta, Türkiye, 2016.
- [31] G. Toz ve P. Erdoğan, “Meme kanserlerinin teşhisinde kullanılan görüntü işleme teknikleriyle ilgili bir derleme çalışması,” *Geleceğin Dünyasında Bilimsel ve Mesleki Çalışmalar*, 1. baskı. Bursa, Türkiye: Ekin Yayınevi, 2020, böl. 5, ss. 69- 83.
- [32] V. S. A. Kargı, *Yapay Sinir Ağ Modelleri ve Bir Tekstil Firmasında Uygulama*. Bursa, Türkiye: Ekin Yayıncılık, 2015.
- [33] M. Bilgin ve İ. F. Şentürk, “Danışmanlı ve yarı danışmanlı öğrenme kullanarak doküman vektörleri tabanlı tweetlerin duygu analizi,” *Balıkesir Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, c. 21, sayı 2, ss. 822-839, 2019.
- [34] H. Ergezer, M. Dikmen ve E. Özdemir, “Yapay sinir ağları ve tanıma sistemleri,” *PiVOLKA*, c. 2, sayı 6, ss. 14-17, 2003.
- [35] B. B. Vural, “Yapay sinir ağları ile finans tahmini,” Yüksek lisans tezi, Sosyal Bilimler Enstitüsü, İşletme Anabilim Dalı, Ankara Üniversitesi, Ankara, Türkiye, 2007.
- [36] L. Bottou, “Stochastic gradient descent tricks,” in *Neural Networks: Tricks of the Trade*, Springer, Berlin, Heidelberg, pp. 421-436, 2012.
- [37] S. E. R. Gazel ve C. T. Batı, “Derin sinir ağları ile en iyi modelin belirlenmesi: Mantar verileri üzerine Keras uygulaması,” *Yüzüncü Yıl Üniversitesi Tarım Bilimleri Dergisi*, c. 29, sayı 3, ss. 406-417, 2019.
- [38] I. Goodfellow, Y. Bengio, A. Courville and Y. Bengio, *Deep Learning*, vol. 1, no. 2, Cambridge: MIT Press, 2016.
- [39] M. D. Zeiler, “AdaDelta: an adaptive learning rate method,” *arXiv preprint arXiv:1212.5701*, 2012.
- [40] A. Alblwi, “Improving the adaptive moment estimation optimization methods for modern machine learning,” M.S. thesis, Dept. Electrical and Computer Eng., University of Delaware, Delaware, USA, 2020.



- [41] Y. Nesterov, "A method for unconstrained convex minimization problem with the rate of convergence  $O(1/k^2)$ ," In *Doklady An USSR* vol. 269, pp. 543-547, 1983.
- [42] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.
- [43] İ. Arslan, *Python ile Veri Bilimi*. Ankara: Pusula Yayıncılık, 2019.
- [44] M. Buscema, G. Ferilli, and P. L. Sacco, "What kind of 'world order'? An artificial neural networks approach to intensive data mining," *Technological Forecasting and Social Change*, vol. 117, pp. 46-56, 2017.
- [45] A. A. Desai, "Gujarati handwritten numeral optical character reorganization through neural network," *Pattern Recognit.*, vol. 43, no. 7, pp. 2582–2589, 2010.
- [46] H. Adeli and A. Panakkat, "A probabilistic neural network for earthquake magnitude prediction," *Neural Networks*, vol. 22, no. 7, pp. 1018-1024, 2009.
- [47] P. Baldi and Y. Chauvin, "Neural networks for fingerprint recognition," *Neural Computation*, vol. 5, no.3, pp. 402-418, 1993.
- [48] W. Gong, X. Zhang, B. Deng, & X. Xu, "Palmprint recognition based on convolutional neural network-Alexnet," in *Proc. 14th Federated Conference on Computer Science and Information Systems (FedCSIS 2019)*, Leipzig, Germany, Sep. 1-4, 2019, pp. 313-316.
- [49] R. Abiyev and K. Altunkaya, "Personal iris recognition using neural network," *International Journal of Security and its Applications*, vol. 2, no. 2, pp. 41-50, 2008.
- [50] M. Singh and H. Kaur, "Artificial neural networks in material science research," *Current Reports on Science and Technology*, vol. 2, pp. 145-159, 2016.
- [51] J. Jiang, P. Trundle, and J. Ren, "Medical image analysis with artificial neural networks," *Computerized Medical Imaging and Graphics*, vol. 34, no. 8, pp. 617-631, 2010.

- [52] V. Nair and G. E. Hinton, “Rectified linear units improve restricted Boltzmann machines,” in *Proc. The 27th International Conference on Machine Learning (ICML)*, Haifa, Israel, June 21-24, 2010, pp. 807–814.
- [53] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [54] P. Ramachandran, B. Zoph, and Q. V. Le, “Searching for activation functions,” in *Proc. 6th International Conference on Learning Representations (ICLR 2018)*, Vancouver, BC, Canada, Apr. 30 - May 3, 2018, pp. 1-13.
- [55] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
- [56] M. D. Zeiler, M. Ranzato, R. Monga, M. Mao, K. Yang, Q. V. Le, and G. E. Hinton, “On rectified linear units for speech processing,” in *Proc. The 38th International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Vancouver, Canada, May 26-31, 2013, pp. 3517–3521.
- [57] G. E. Dahl, T. N. Sainath, and G. E. Hinton, “Improving deep neural networks for LVCSR using rectified linear units and dropout,” in *Proc. The 38th International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Vancouver, Canada, May 26-31, 2013, pp. 8609-8613.
- [58] J. Brownlee, “What is the difference between a batch and an epoch in a neural network?,” *Deep Learning: Machine Learning Mastery*, 2018.
- [59] D. Mack, “How to pick the best learning rate for your machine learning project,” Medium.com [İnternet]. Erişim: <https://medium.com/octavian-ai/which-optimizer-and-learning-rate-should-i-use-for-deep-learning-5acb418f9b2> (Erişim tarihi: 8 Ocak 2021).
- [60] A. Gülcü ve Z. Kuş, “Multi-objective simulated annealing for hyper-parameter optimization in convolutional neural networks,” *Peerj Computer Science*, vol. 7, no. 338, 2021.

- [61] A. Anbar ve D. Alper, "Bank specific and macroeconomic determinants of commercial bank profitability: Empirical evidence from Turkey," *Business and Economics Research Journal*, vol. 2, no. 2, pp. 139-152, 2011.
- [62] H. Sarıtaş, S. G. Kangallı Uyar ve A. Gökçe, "Banka karlılığı ile finansal oranlar ve makroekonomik değişkenler arasındaki ilişkilerin sistem dinamik panel veri modeli ile analizi: Türkiye araştırması," *Eskişehir Osmangazi Üniversitesi İİBF Dergisi*, c. 11, sayı 1, ss. 87-108. 2016.
- [63] S. M. Miller and A. G. Noulas, "Portfolio mix and large-bank profitability in the USA," *Applied Economics*, vol. 29, no. 4, pp. 505-512, 1997.
- [64] E. Osuagwu, "Determinants of bank profitability in Nigeria," *International Journal of Economics and Finance*, vol. 6, no. 12, pp. 46-61, 2014.
- [65] A. Dietrich and G. Wanzenried, "Determinants of bank profitability before and during the crisis: Evidence from Switzerland," *Journal of International Financial Markets, Institutions & Money*, vol. 21, no. 3, pp. 310-320, 2011.
- [66] B. K. Short, "The relation between commercial bank profit rates and banking concentration in Canada, Western Europe, and Japan," *Journal of Banking & Finance*, vol. 3, no. 3, pp. 209-219, 1979.
- [67] P. Bourke, "Concentration and other determinants of bank profitability in Europe, North America and Australia," *Journal of Banking & Finance*, vol. 13, no. 1, pp. 65-79, 1989.
- [68] P. Molyneux and J. Thornton, "Determinants of European bank profitability: A note," *Journal of Banking & Finance*, vol. 16, no. 6, pp. 1173-1178, 1992.
- [69] A. Demirguc-Kunt & H. Huizinga, "Financial structure and bank profitability," in *Financial Structure and Economic Growth: A Cross-Country Comparison of Banks, Markets, and Development*. A. Demirguc-Kunt and R. Levine, Eds. Cambridge, MA, USA: MIT Press, 2001, pp. 243-261.

- [70] J. Goddard, P. Molyneux & J. O. Wilson, “The profitability of European banks: a cross-sectional and dynamic panel analysis,” *The Manchester School*, vol. 72, no. 3, pp. 363-381, 2004.
- [71] Y. Shachmurove, *Applying Artificial Neural Networks to Business, Economics and Finance*. University of Pennsylvania, Center for Analytic Research in Economics and the Social Sciences, 2002.
- [72] C. Gallo, C. Letizia and G. Stasio, “Artificial neural networks in financial modelling,” in *Proc. 36th EWGFM International Meeting*, Brescia, Italy, May 5-7, 2005, pp. 1-21.
- [73] C. Brooks, *Introductory Econometrics for Finance*. Cambridge, MA, USA: Cambridge University Press, 2008.
- [74] E. Öztemel, *Yapay Sinir Ağları*. İstanbul, Türkiye: Papatya Yayıncılık, 2020.
- [75] S. M. Weiss and C. A. Kulikowski, *Computer Systems That Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning, and Expert Systems*. San Mateo, CA, USA: Morgan Kaufmann Publishers Inc., 1991.
- [76] A. M. Molinaro, R. Simon and R. M. Pfeiffer, “Prediction error estimation: A comparison of resampling methods,” *Bioinformatics*, vol. 21, no. 15, pp. 3301-3307, 2005.
- [77] J. L. Raheja and U. Kumar, “Human facial expression recognition from detected in captured image using back propagation neural network,” *International Journal of Computer Science and Information Technology*, vol. 2, no. 1, pp. 116-123, 2010.
- [78] S. Kılıç, “Doğrusal regresyon analizi,” *Journal of Mood Disorders*, c. 3, sayı 2, ss. 90-92, 2013.