T.C.
ANKARA YILDIRIM BEYAZIT UNIVERSITY
GRADUATE SCHOOL OF SOCIAL SCIENCES

THE EFFECTIVENESS OF ARTIFICIAL INTELLIGENCE IN FINANCIAL ANALYSIS: ISE100

PHD THESIS

Fikriye KARACAMEYDAN

THE DEPARTMENT OF FINANCE AND BANKING

Ankara, 2023

T.C.
ANKARA YILDIRIM BEYAZIT UNIVERSITY
GRADUATE SCHOOL OF SOCIAL SCIENCES

THE EFFECTIVENESS OF ARTIFICIAL INTELLIGENCE IN FINANCIAL ANALYSIS: ISE100

PHD THESIS

Fikriye KARACAMEYDAN

THE DEPARTMENT OF FINANCE AND BANKING

Supervisor
Assoc. Prof. Dr. Erhan ÇANKAL

Ankara, 2023

PAGE OF APPROVAL

The thesis study prepared by Fikriye KARACAMEYDAN and title as "The Effectiveness of Artificial Intelligence in Financial Analysis: ISE100" is unanimously accepted as a PhD thesis at the Department of Finance and Banking of the Institute of Social Sciences of the Ankara Yıldırım Beyazıt University, by following jury.

Juror		Institution	Signature
Assoc. Prof. Dr. Erhan ÇANKAL		Ankara Yıldırım Beyazıt University	
Accept [X]	Rejection		
Prof. Dr. Yüksel Akay UNVAN		Ankara Yıldırım Beyazıt University	
Accept [X]	Rejection		
Asst. Prof. Dr. Haroon MUZAFFAR		Ankara Yıldırım Beyazıt University	
Accept [X]	Rejection		
Prof. Dr. M. Cengiz TAPLAMACIOĞLU		Gazi University	
Accept [X]	Rejection		
Asst. Prof. Dr. Cevat RAHEBI		Istanbul Topkapı University	
Accept [X]	Rejection		

Date of Defence: 08/02/2023

I approve that this thesis satisfies all the requirements to be accepted as a thesis for the degree of doctor of philosophy at the Department of Finance and Banking of the Institute of Social Sciences of the Ankara Yıldırım Beyazıt University.

Director of the Institute of Social Sciences Prof. Yaşar YİĞİT

PLAGIARISM

I hereby declare that all information in this thesis has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work; otherwise I accept all legal responsibility. (08/02/2023)

Fikriye KARACAMEYDAN

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my supervisor Assoc. Prof. Dr. Erhan ÇANKAL for his endless support and guidance during my study. I also would like to thank to The Council of Higher Education for the support and permission during my studies.

Besides, I am appreciative to the members of my thesis committee, Prof. Dr. Yüksel Akay UNVAN, Prof. Dr. Atilla GÖKÇE and Asst. Prof. Dr. Haroon MUZAFFAR, for their contribution to my thesis.

Finally, I am grateful to my beloved mother *Saliha* and my dear daughters *Doğa a*nd *Defne* for their patience and support.

ABSTRACT

The Effectiveness of Artificial Intelligence in Financial Analysis: ISE100

In this thesis, the effects of macroeconomic indicators on the Borsa Istanbul (BIST100) index are analyzed using a deep learning method based on artificial neural networks, which is one of the technologies of artificial intelligence.

All the variables selected in the present study consist of the monthly closing prices for the period January 2001-January 2022. The effects of macroeconomic indicators consisting of the Dollar Rate (ERUSD, TL/$), Money Supply (M2), Producer Price Index (PPI), Industrial Production Index (IPI), Active Bond Interest Rate (TYBY), Brent Oil Price (BRT), and Gold Prices (GP, TL/Gr) on the BIST100 index are examined with both Logit and Probit regression models with qualitative responses and LSTM (Long-Short Term Memory) architecture and a deep learning model that is a machine learning method. The performances of the methods are compared with each other. The research has revealed that the effects of selected macroeconomic variables on the BIST100 index increase during crisis periods and during the COVID-19 pandemic. It has been determined that the change in the prices of stocks in BIST100 is largely explained by money supply, exchange rate, industrial production index, and consumer price index. It has been determined that the changes in the real sector have a significant impact on the capital sector. It has been determined that the deep learning method is more successful in predicting financial crises.

The results of the analysis revealed that the LSTM deep learning model, developed to analyze the effects of macroeconomic indicators on the Borsa Istanbul (BIST100) index, has a very low error level and gives more effective and satisfactory results than Logit and Probit regression models.

Keywords: Artificial Intelligence, Artificial Neural Networks, Deep Learning, Financial Forecast, ISE100, Logit-Probit Model, LSTM, Stocks.

ÖZET

Finansal Analizde Yapay Zekanın Etkinliği: BIST100

Bu tez çalışmasında, makroekonomik göstergelerin Borsa İstanbul (BIST100) endeksi üzerindeki etkileri yapay zekâ teknolojilerinden yapay sinir ağlarına dayalı derin öğrenme yöntemi kullanılarak analiz edilmiştir.

Mevcut çalışmada seçilen tüm değişkenler Ocak 2001- Ocak 2022 dönemine ait aylık kapanış fiyatlarından oluşan verileri kapsamaktadır. Dolar Kuru (ERUSD, TL/$), Para Arzı (M2), Üretici Fiyat Endeksi (PPI), Sanayi Üretim Endeksi (IPI), Aktif Tahvilin Faiz Oranı (TYBY), Brent Petrol Fiyatı (BRT) ve Altın (GP, TL/Gr) fiyatlarından oluşan makroekonomik göstergelerin BIST100 endeksi üzerindeki etkileri, hem Logit ve Probit nitel tepkili regresyon modelleri, hem de LSTM (Long- Short Term Memory) mimarisi kullanılarak oluşturulan ve bir makine öğrenme yöntemi olan derin öğrenme modeli ile incelenmiştir. Her yöntem için elde edilen sonuçların performansları karşılaştırmalı olarak değerlendirilmiştir. Seçilen makroekonomik değişkenlerin kriz dönemlerinde ve Covid-19 pandemisi döneminde BIST100 endeksi üzerindeki etkilerinin arttığı ortaya konulmuştur. BIST100'de yer alan hisse senetleri fiyatlarındaki değişikliğin, büyük oranda para arzı, döviz kuru, sanayi üretim endeksi ve tüketici fiyat endeksi tarafından açıklandığı saptanmıştır. Reel sektördeki değişimlerin sermaye sektörünü önemli ölçüde etkilediği tespit edilmiştir. Finansal krizlerin tahmini konusunda derin öğrenme yönteminin daha başarılı olduğu belirlenmiştir.

Analiz sonuçları, makroekonomik göstergelerin Borsa İstanbul (BIST100) endeksi üzerindeki etkilerini analiz etmek için geliştirilen LSTM derin öğrenme modelinin hata düzeyinin düşük, Logit ve Probit regresyon analizi yöntemlerinden daha etkin ve tatmin edici sonuçlar verdiği ortaya konulmuştur.

Anahtar Kelime: BIST100, Derin Öğrenme, Finansal Tahmin, Hisse Senedi, Logit-Probit Model, LSTM, Yapay Sinir Ağları, Yapay Zekâ.

TABLE OF CONTENTS

ABSTRACT i
ÖZET ii
TABLE OF CONTENTS iii
LIST OF ABBREVIATIONS vii
LIST OF FIGURES ix
LIST OF TABLES xi
1. INTRODUCTION 1
1.1. Stock Exchanges 7
1.2. Istanbul Stock Exchange (ISE) 9
1.3. Purpose of the Research 11
1.4. Contribution of the Research 12
1.5. Definition and Limits of the Problem 13
1.6. Problem Solving 13
1.7. Organization of the Thesis 14
2. QUALITATIVE RESPONSE MODELS 16
2.1. Linear Probability Model (LPM) 16
2.2. Logit Model 17
2.3. Probit Model 19
3. ARTIFICIAL INTELLIGENCE 22
4. DEVELOPMENT PROCESS OF ARTIFICIAL INTELLIGENCE 25
5. ARTIFICIAL INTELLIGENCE TECHNOLOGIES 31
5.1. Expert System 31
5.2. Machine Learning 31
5.3. Fuzzy Logic 32
5.4. Genetic Algorithms 32
5.5. Artificial Neural Networks 33
5.6. Deep Learning 34

6. ARTIFICIAL NEURAL NETWORK 35
6.1. Definitions of Neural Networks in the Literature 35
6.2. Biological Nervous System 36
6.3. Artificial Neural Networks Elements 37
6.3.1. Inputs 38
6.3.2. Weights 39
6.3.3. Summation Function 39
6.3.4. Activation Function 41
6.3.5. Output 42
6.4. General Structure of Artificial Neural Networks 43
6.4.1. Input Layer 43
6.4.2. Hidden (Intermediate) Layer 44
6.4.3. Output Layer 44
6.5. Architectural Structures of the Artificial Neural Networks 45
6.5.1. Classification of ANNs According to Connection Structures 45
6.5.2. Artificial Neural Network Classification Based on Learning Styles 48
6.5.3. Artificial Neural Network Classification Based on the Number of Layers 51
6.6. Learning Algorithms in Artificial Neural Networks 55
6.7. Learning Rules 62
6.7.1. The Hebbian Learning Rule 62
6.7.2. The Hopfield Learning Rule 62
6.7.3. The Delta Rule 62
6.7.4. The Gradient Descent Learning Rule 63
6.7.5. The Kohonen Learning Rule 64
6.8. Application Areas and Properties of Artificial Neural Networks 64
6.9. The Design of Artificial Neural Networks 68
6.10. Training and Testing of Artificial Neural Networks 68

6.11. Selection of Network Structure 72
6.11.1. Determining the Number of Input Neurons and Output Neurons 73
6.11.2. Determination of the Hidden Layer and the Number of Hidden Neurons 73
6.11.3. Data Standardization 74
6.12. Determination of Neural Network Performance 75
6.12.1. Determining the Stopping Criteria 77
6.12.2. Selection of Learning Algorithm 77
6.13. Advantages and Disadvantages of Artificial Neural Networks Applications 77
6.14. Disadvantages of Artificial Neural Network Applications 78
7. DEEP LEARNING 80
7.1. Fundamentals of Deep Learning Architecture 81
7.1.1. Input Layer 82
7.1.2. Convolution Layer 82
7.1.3. Activation Layer 83
7.1.4. Pooling Layer 83
7.1.5. Dropout Layer 83
7.1.6. Fully Connected Layer 83
7.1.7. Classification Layer 84
7.2. Deep Learning Architectures 84
7.2.1. Multi Layer Perceptron (MLP) 85
7.2.2. Convolutional Neural Network (CNN or ConvNet) 85
7.2.3. Recurrent Neural Network (RNN) 86
7.2.4. Long Short-Term Memory (LSTM) 87
7.2.5. Restricted Boltzmann Machine (RBM) 87
7.2.6. Deep Belief Network (DBN) 88
7.2.7. Deep Auto-Encoder (DAE) 88
8. LITERATURE REVIEW 89
9. METHODOLOGY AND DATA 106

10. EMPIRICAL ANALYSIS 121
10.1. Descriptive Statistics 121
10.2. Unit Root Test Results 122
10.3. Logit ve Probit Models and Results 128
10.4. Deep Learning Models and Analyses 134
10.4.1. Preprocessing of Data Used in the Study 134
10.4.2. Deep Learning Network Architecture Used in the Study 135
10.4.3. Training and Test Phase of the Network 139
10.4.4. Deep Learning Analysis Results 140
10.5. Comparison of Model Results 146
11. ARGUMENT 149
12. CONCLUSION AND PROPOSAL 150
13. REFERENCES 153

LIST OF ABBREVIATIONS

ADALINE : Adaptive Linear Elements
ADF : Augmented Dickey-Fuller Test
AI : Artificial Intelligence
ANN : Artificial Neural Networks
AIC : Akaike Information Criterion
ARIMA : Autoregressive Integrated Moving Average
ART : Adaptive Resonance Theory
BIC : Bayesian Information Criteria
CAE : Contractive Autoencoder
CAPM : Capital Asset Pricing Model
CML : Capital Market Line
CSD : Central Securities Depository
DAE : Denoising Autoencoder
DW : Durbin-Watson d statistic
DL : Deep Learning
EFAMA : European Fund and Asset Management Association
EMT : Efficient Market Hypothesis
GRNN : General Regressions Networks
GNP : Gross National Product
GDP : Gross Domestic Product
ICI : Investment Company Institue
IPI : Industry Price Index
ISE100 : Istanbul Stock Exchange 100
IEEE : Institute of Electrical and Electronics Engineers
LMS : Least Mean Squares
LVQ : Linear Vector Quantization
MAE : Mean Average Error
MSE : Mean Apsolute Error
MADALINE : Multiple Adaptive Linear Elements
MAPE : Mean Absolute Percentage Error
ML : Machine Learning
MLP : Multi Layer Perception

MPT : Modern Portföy Theory
PPI : Wholesale Price Index/Domestic Producer Index
RBF : Radial Basis Funcitons
SML : Security Market Line
SPK : Capital Markets Borad of Türkiye
OM : Self-Organizing Feature Maps
TCCB : The Central Bank of the Republic of Türkiye
TUIK : Turkish Statistical Institute
UCITS : Undertaking for Collective Investment in Transferable Securities
LSTM : Long Short Term Memory
ML : Machine Learning
MAPE : Mean Absolute Percentage Error
MLP : Multi Layer Perceptron
RCNN : Recurrent Neural Network National
SE : Stock Exchange
NYSTE : New York Stock Exchange
SAE : Sparse Autoencoder
VAE : Variational Autoencoder

LIST OF FIGURES

Figure 1.1. Stock market performances (2022/03-2021/12) 9

Figure 1.2. Number of investors and portfolio value in CSD 10

Figure 1.3. Market capitalization of stock exchange companies (million dollars) 11

Figure 2.1. Linear probability model 16

Figure 2.2. Conditional probabililty model 16

Figure 2.3. The expected value of the dependent variable 17

Figure 2.4. The conditional expected value of the dependent variable 17

Figure 2.5. Logistic distribution function 18

Figure 2.6. Probability of failure 18

Figure 2.7. Odds ratio 18

Figure 2.8. Logistic regression model 18

Figure 2.9. Utility index 19

Figure 2.10. The probability that the threshold value is less than or equal to the utility index 20

Figure 2.11. The standard normal cumulative distribution function 20

Figure 2.12. The graphs of the Logit and Probit models 21

Figure 3.1. AI, ML, ANN and DL 23

Figure 6.1. Structure of a typical neuron 37

Figure 6.2. General structure of artificial neuron 38

Figure 6.3. Summation function 40

Figure 6.4. General structure of the ANN 44

Figure 6.27. Graident descent 63

Figure 7.2. Elements of the deep neural network architecture 82

Figure 9.1. ISE100-TYBY 110

Figure 9.2. ISE100-ERUSD 111

Figure 9.3. ISE100-M2 112
Figure 9.4. ISE100-IPI 113
Figure 9.5. ISE100-PPI 113
Figure 9.6. ISE100-GP 114
Figure 9.7. ISE100-BRT 115
Figure 9.8. Logit model 117
Figure 9.9. Probit model 117
Figure 10.1. The probabilistic unit root process 123
Figure 10.2. ADF random walk equation 124
Figure 10.3. PP model 124
Figure 10.4. Original graphics of variables (level) 125
Figure 10.5. Graphs of variables with normalization 135
Figure 10.6. Simple architecture of deep learning network 138
Figure 10.7. Training progress with *adam* 142
Figure 10.8. Train and test results with *adam* 143
Figure 10.9. Training progress with sgdm 144
Figure 10.10. Train and test results with *sgdm* 144

LIST OF TABLES

Table 4.1. Some important advantages in neural network research 30

Table 6.1. Summation functions in ANNs 40

Table 6.2. Some activation functions 42

Table 6.3. Network types and areas of success 72

Table 6.4. Performance criteria 76

Table 10.1. Some descriptive statistics for the dependent and independent variables 122

Table 10.2. Unit root test results at level 126

Table 10.3. Unit root test results at ln 127

Table 10.5. Logit model compatibility results 130

Table 10.6. Probit model compatibility results 130

Table 10.7. Results of the Logit model & the Probit model 132

Table 10.8. Correct classification rate table for the Logit model 133

Table 10.9. Correct classificaiton rate table for the Probit 133

Table 10.10. Deep learning network architecture parameters 139

Table 10.11. Simulation results for *adam* 141

Table 10.12. Simulation results for *sgdm* 141

Table 10.13. Deep learning results 142

Table 10.14. Deep learning performance results 145

Table 10.15. Logit model performance 147

Table 10.16. Probit model performance 148

Table 10.17. DL model performance 148

1. INTRODUCTION

The main purpose of economics and economic activities, which try to find the cause-and-effect relationships related to the subjects it deals with, and to reveal them in the form of scientific rules, is to increase the level of social welfare. Social welfare and economic development are indicators of the social and economic development of societies. The easier it is for individuals in a society to reach basic needs such as social security, education, health, and housing opportunities, the more they feel safe and free, and the more savings turn into investments, the higher the level of social welfare in that society. For this reason, the concept of social welfare has become one of the issues that play an important role in scientific platforms as well as in the political field for both developed and developing countries.

Social welfare and economic development are closely related to the growth of the money market and capital markets, which constitute the financial markets of the countries. However, there is no common view on the extent of the development of capital markets, a good financial system has five components; a reliable public finance and public debt management, reliable and stable financial regulations, the presence of various banks, national and international, or both, a central bank that can balance domestic finances and manage international financial relations, and a well-functioning stock market. A regular financial system with these components plays an important role in the economic growth and development of countries (Rousseau & Sylla, 2003, pp. 374–375). As it can be understood from these defined components, the more developed of the financial market of a country, the lower of the risk level and vulnerability in the society. In addition, this predictable structure will play an important role in the development of countries, thereby increasing their social welfare and by encouraging the private sector, and savings will be transformed into investments, leading to an increase in economic growth.

Developed and developing countries have made various reforms in order to protect their economies and minimize the effects of financial fluctuations due to the economic crises experienced in the world from the 1980s to the 1990s. These reforms have been in the form

of practices that include financial liberalization policies such as liberalizing interest rates, removing credit ceilings, reducing or completely removing the deposit reserve ratios that banks have to keep at the Central Bank, opening the banking sector to both foreign and residents and liberalizing capital movements (Güloğlu & Altunoğlu, 2011, pp. 107–134). It has been observed that the financial crises experienced in this period, especially in developing countries, were experienced due to the financial liberalization steps taken to sustain growth and the hot money that left the country due to economic and political risks, or due to international capital movements, excessive borrow money, exchange rate policies and high inflation rates. Also, some important examples of the crises and the costs caused by these crises in the country's economies arising from speculative capital movements, are the crises experienced in Turkey in 1994 and 2001, the Asian crisis in 1997, and the Russian crisis in 1997-98.

In addition to these economic crises, the Mortgage crisis, which emerged in the USA in 2007 and then spread to England and turned into a global crisis as of the second quarter of 2008, is an important example as it shows the reflections of the crises in the world (Eğilmez, 2009, pp. 52–53). It has been shown this and the other financial crises mentioned above, regardless of the level of development, the markets of the countries greatly affect each other, cause great financial fluctuations and ruins in the world, and that financial shocks in large economies can lead to the bankruptcy of various financial institutions, especially banks, and even the bankruptcy of countries.

These developments in the world financial market have shown how important it is to forecast the possible chaos and uncertainties in financial markets and the fluctuations that may occur in micro and macroeconomic factors affecting price movements, in order to prevent and/or reduce the effects of financial breaks and uncertainties in the markets. This situation has revealed the fact that economic decision-makers can gain significant advantages and opportunities if they can accurately forecast the future value of capital market instruments and asset prices.

The unexpected changes in stock prices in the United States, many European countries, and Japan in the 1980s and 1990s attracted the attention of researchers and it was concluded that these fluctuations may be due to macroeconomic factors (Kaymaz & Yilmaz, 2022). On the other hand, Fama (1990), Barro (1990), and Schwert (1990) concluded in their

studies that the changes in the prices of stocks representing the firm value are mainly caused by the changes in the future cash flows and the changes in the discount rates (Barro, 1990; Fama, 1990; Schwert, 1990). Similarly, Chakravarty and İlahi et al., in their studies have concluded that basic macroeconomic variables affect stock prices (Chakravarty, 2005; Ilahi et al., 2015).

In traditional estimation fusion methods, various methods including averaging, least squares, and mean absolute deviations are chosen, and a single superior estimation is tried to be produced from the linear combination of individual estimations. In this case, the possible disadvantage of this type of modeling is that it does the linear modeling. In other words, it is the process of assuming a linear relationship between the variables. In reality, there may be a linear or curvilinear relationship between the variables. Therefore, making estimations based on linear relationships shows that traditional models do not guarantee the accuracy of their estimations (Donaldson & Kamstra, 1999, p. 228). For this reason, model estimation is made by combining models arising derived from different econometric and/or computer technologies, and thus the model is tried to be strengthened by avoiding specification errors, measurement errors, and deviations and minimizing these effects.

The developments and innovations in computer technology and capital markets in recent years have led to an increase in the diversity of products in the financial markets, thus the diversification of the study subjects in the field of finance and an increase in the interest in forecasts for economic analysis. The processing of data that symbolizes real-life facts and transforms them into a meaningful and decisionable form is called “information”. In the process of transforming information into “knowledge” and knowledge into “decisions”, the necessity of making financial analyses by means of statistical and econometric models in a fast, economical, and detailed manner has led to the emergence of new analysis methods. In financial forecasting, the methods used in the information processing process are: general grouping and regression analysis methods; volatility methods; parametric statistical techniques such as autoregression analysis, where the probability distribution depends on certain parameters; they can be listed as non-parametric statistical techniques such as the Friedman test, Sperman’s rank correlation test, and covariance estimation, where the probability distribution does not depend on certain parameters. In addition to these methods, various Artificial Intelligence (AI) technologies, which are very compatible with the

structure of financial data based on computer technologies, have been widely used in recent years (Henrique et al., 2019).

The fact that accurate estimations that reduce financial uncertainty and costs will bring successful results and thus play an important role in reducing the problems and costs that may arise for both investors and governments, and that the benefit functions can be maximized, has increased interest in forecasting modeling and led to the emergence of new forecasting techniques. The Artificial Neural Networks (ANN) method from AI technologies, which is one of these new prediction techniques, has the ability to learn, generalize, and work with incomplete, faulty, flawed, and even wrong data. It has become a prominent method because it gives very successful results in classification, optimization, pattern recognition processes, and especially in estimating nonlinear time series (Cheng et al., 2022a).

It is formed by ordering the data observed in the financial markets and following each other according to the order of occurrence; data such as stock prices, stock market index, exchange rates, and interest rates constitute financial time series, which are the main source of econometric studies (Çil, 2018, p. 5). Time series are defined as numerical quantities in which the values of the variables are observed consecutively from one period to the next (Sevüktekin & Nargeleçekenler, 2010, p. 41). Financial time series are not stationary because they are statistically composed of trends, seasonal movements, cyclical movements, and random movements (Kennedy, 1998, p. 288). For this reason, it is important to be able to make forecasts based on the behavior of the series by using methods that will purify the series from these components. In this context, AI technologies have become a very popular method used in financial analysis.

ANNs can be defined as computer software developed to imitate the working mechanism of the human brain to realize the brain's abilities such as learning, remembering, and generating new information by generalizing without any assistance (Donaldson & Kamstra, 1999, p. 228). They can accurately and perfectly model the relationship between input and output through learning, despite being software that mimics the human brain, without relying on any prior information or assumptions. Because of their ability to learn from data without any prior knowledge, neural networks are very suitable for classification and regression models. In addition, the ANNs are not linear in nature. Unlike many

traditional statistical and econometric methods, ANNs produce successful models using complex, incomplete, faulty, and flawed data, both qualitative and quantitative data, and for analyses where traditional methods cannot find a solution and produce weak or ineffective results. For the reasons listed above, ANNs have started to be used as the best alternative for data analysis (Kamruzzaman et al., 2006a, pp. 2–3). It is still successfully applied in many fields such as engineering, industry, business, finance, education, military defense, and health (Henrique et al., 2019; Quazi, 2022).

ANNs, which find a wide area of use in the field of economics and finance as well as in many other fields, have become a very interesting method due to their features such as measuring and analyzing without the need for prior knowledge, making easily continuous function estimation, and making generalizations. Because of these features, it has started to be used in many studies (Henrique et al., 2019; Patil et al., 2021). This interest in ANNs, which have been shown to give successful results in modeling and estimating complex problems, has also led to the diversification and enrichment of AI applications.

It is thought that applications produced using artificial neural network technology in the field of corporate finance, such as financial simulation, investor behavior forecasts, financial evaluation, loan approval, stock and asset portfolio management, pricing of public offerings, and determination of optimal capital structure, can be developed with artificial neural network technology (J. Hsieh et al., 2019, p. 12). Major fluctuations in the world economies, major company bankruptcies, financial crises, and the increase in the importance of determining the chaos and uncertainties experienced in financial markets have revealed the necessity of benefiting from applications developed and/or to be developed by AI technologies.

It is very important to make the right investment decisions for investors who want to turn their savings into investments. Therefore, they need data or a signal to help them make a decision. Accurate preliminary indicators to be obtained in estimating the prices of stocks traded on the stock exchange or the direction of price movements will offer many opportunities to investors. Political events, general economic conditions, basic economic data, and investors' expectations are important factors affecting the prices of instruments traded on the stock exchange. However, by simplifying the analysis and assuming that each variable behaves linearly prevents us from reaching the correct results. This situation leads

to the use of more accurate calculation and estimation methods offered by advanced computer technologies in time series analysis due to their non-linear, high-frequency, polynomial, dynamic, and complex structures. Thus, in order to solve complex real-world problems, it is attempted to overcome the shortcomings of a single method by utilizing various advantages of AI technologies. At the same time, the disadvantages arising from the use of hybrid models and techniques are trying to be eliminated. Therefore, the number of such studies is increasing (Enke et al., 2011, pp. 201–206; Enke & Mehdiyev, 2013, p. 636; Quazi, 2022).

Capital markets contribute to economic growth and development by transferring the funds held by investors who want to transform their savings into investments to people in need of funds. Stocks are one of the various investment instruments offered by the capital markets, especially the stock markets, which are regulated according to specific rules. The stock market is significantly affected by economic, social, and political developments at both national and international levels. Therefore, any information to be obtained about these high-risk investment instruments will undoubtedly also affect the decision-making process.

Two types of risk that affect the capital markets are systematic risk and unsystematic risk. Financial risk, management risk, and industry risk are also included in the list of unsystematic risks while inflation risk, interest rate risk, political risk, market risk, and currency risk are regarded to be systematic risks (Hull, 2007, pp. 9–12). Macroeconomic variables are one of the most important findings that will provide an understanding of the effects of these risk factors. Because of these characteristics, it can be observed that studies on stock market indices in the literature frequently use fundamental macroeconomic indicators such as inflation, interest rates, money supply, the industrial production index, the gross national product, gold prices, oil prices, and exchange rates.

In this study, the predictive efficiency of these methods will be tested by using the Logit and Probit models, which are traditional econometric methods, and a deep learning neural network model from AI technologies to determine whether the changes in the value of the Stock Exchange Istanbul 100 Index can be explained by selected macroeconomic variables. Thus, it will be revealed from the methods used, that which modeling technique can best predict the change in economic and financial system dynamics.

1.1. Stock Exchanges

Stock exchanges are markets where long-term investment instruments such as stocks and bonds are bought and sold. The direct confrontation between fund suppliers and fund demanders is the main feature of these markets. Fund transfers are made in a transparent environment using standard methods in exchanges that have been established and operate according to certain rules. Therefore, stock markets are a safe and easy investment area for investors (Karan, 2013, p. 37). The fact that investors can take risks easily is very important for the development of the economies of countries. So, the necessary fund transfer is provided for the savings, which form the basis of economic activities, to be transformed into investments and investments into production.

Although there are different opinions about the establishment of modern stock exchanges, stock market activities started with the trading of shares of the The United East India Company (*Vereenigde Oost-Indische Compagnie (VOC) is considered the first multinational company in the world and was the first company to issue stock*) in the early 1600s. In the 1700s, London emerged as the financial and stock market center followed by America and New York in the early 1800s. The world stock markets, which started to develop all over the world after the 1950s, gained a global character after the 1900s (Karan, 2013, p. 37). As a result of technological developments, the increase in communication opportunities and the developments in computer technologies have led to the development and growth of financial institutions and the expansion of the borders within which they operate. These developments have brought local investors' investment areas to global markets. In addition, the globalization of financial markets has increased both capital movements and the diversity of securities, thus the depth of the capital markets.

The world's most important stock exchanges are located in a group of countries located in Europe, America, and Asia, which have come a long way in the industrialization process. As of 2022, the world's four largest stock exchanges in terms of trading volume can be listed as follows. The New York Stock Exchange (NYSE) in the United States with a valuation of $20 trillion and by a wide margin is in first place. The National Association of Securities Dealers Automated Quotations (NASDAQ), an over-the-counter exchange founded in 1971, is in second place. The third place is the London Stock Exchange, founded

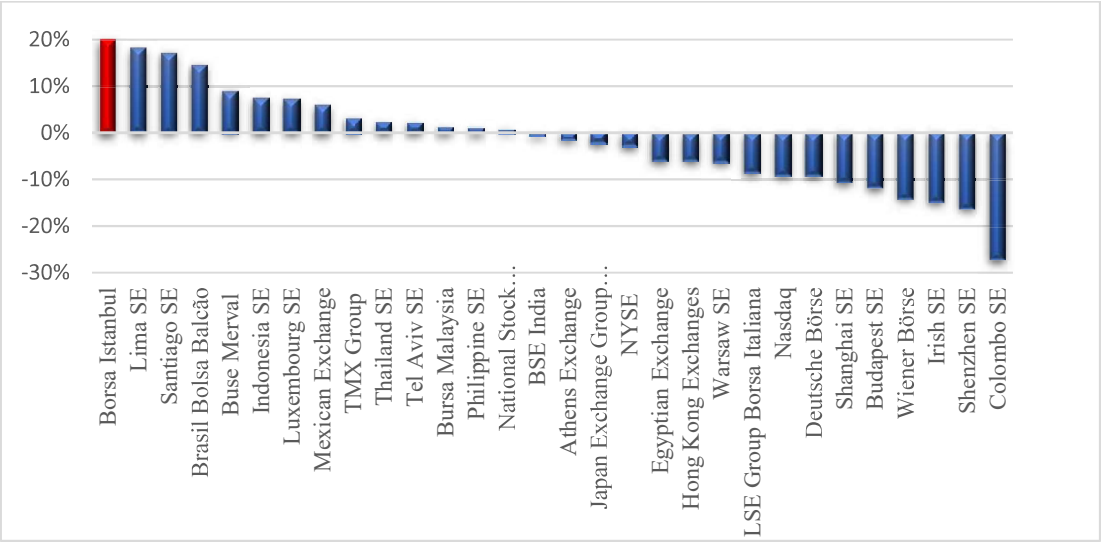
in 1801 and the fourth largest stock exchange is the Far East, Tokyo Stock Exchange (TSE), which was established in the 1870s.

Emerging stock markets are the stock markets of the developing BRICS countries (Brazil, Russia, India, China, and South Africa). These countries are rich in natural resources such as oil and natural gas and have high growth potential. A significant part of the basic capital needs of developing countries is met from stock portfolio investments, that is, from stock market activities. In other words, businesses, especially public joint stock companies, meet their medium and long-term funding needs from the capital markets. Thus, stock exchanges, which enable individuals to transfer their savings to companies as capital, also prevent companies from turning to high-interest, high-risk, and short-term loans. In this case, resources turn into investments and contribute to the national economy. In addition, the book values and equities of companies listed on the stock exchange are increasing.

The main function of capital markets is to optimize resource distribution in the economy by directing savings to rational investment areas and to balance income distribution by spreading ownership of the means of production to the base. In order for capital markets to fulfill this function, they must have an effective market functioning. This issue, depends on the transparency and trustworthiness of the market and the protection of investors' rights. The mechanism to ensure this is the existence of an institution that will effectively supervise the functioning of capital market (Sayılgan, 2011, pp. 22–23).

The chart below, Figure 1.1, shows the performance comparison of selected world stock markets between 2022/03 and 2021/12. In this chart, it is seen that the stock markets of Brazil, Indonesia, Mexico, and Luxembourg have been increased and the biggest increase was experienced in ISE100 at a rate of %20.

On the other hand, it is observed that the performance of the stock markets such as NASDAQ, NYSE, Hongkong, Shanghai, and Deutsche, which are among the few world stock exchanges with high trading volumes, dropped quite high in the range of %1-%16. This situation can also be interpreted as the reflections of the Covid-19 pandemic experienced all over the world and the effects of the pandemic on the world economies, which are already struggling with economic crises, to the capital markets.



Source: https://www.world-exchanges.org/our-work/statistics /WFE.

Figure 1.1. Stock market performances (2022/03-2021/12)

Although stock market indices are a basic indicator for national economies, it is necessary to know what the factors affecting the index are before making an evaluation. Internal factors such as the economic data of the countries, unemployment figures, interest rates, geopolitical position, and risks, and external factors such as global data and the economic relations of the countries with each other, are the factors that affect the economic course of the countries and, of course, the stock market index.

1.2. Istanbul Stock Exchange (ISE)

The markets where precious metals, financial instruments, and capital market instruments are bought and sold in a reliable and audited environment based on laws and certain legislation and prices are formed in this transparent environment, are called the Stock Exchange. After the financial liberalization movements that started in Turkey in the 1980s, the Istanbul Stock Exchange was established in 1986. Since its establishment, it has developed continuously and has become one of the important stock exchanges in the regional area. With the Capital Markets Law dated December 30, 2012, and numbered 6362, the capital markets, the Istanbul Stock Exchange market, the Izmir Futures and Options Exchange market, and the Istanbul Gold Exchange market were gathered under one roof under the name of "Borsa Istanbul". With this law, Istanbul Stock Exchange was removed from the status of a public institution as of April 3, 2013, and continued its activities as an autonomous and independent company (*Borsa İstanbul A.Ş.*, n.d.).

According to the statistics from the Central Securities Depository system show that in July 2022, there were 28.941.853 investors who transacted a total of 20.386.350 million TL in market value. A total of 449 companies are listed on the Istanbul Stock Exchange. Index created to measure the sectoral and overall performances of the prices and returns of these stocks are also alternative investment tools for the portfolio of investors. Therefore, individual and corporate portfolio managers, who carry out their investments with an index-linked portfolio strategy, tend to create their mutual funds according to the returns of the ISE index.

As of June, there were 5.351.006 investors with a portfolio of 13.411.636.000.000₺ registered in the Central Securities Depository (CSD) system. The chart below, Figure 1.2, illustrates how the number of investors and portfolio size has changed over the years. It is seen that there has been a rapid increase in the number of investors and portfolio value after 2020. It can be observed that a record surge brought it to the 5 million mark in 2021. The 52 firms that went public and their shares' strong performance in the initial trading days are believed to be the leading causes of this.

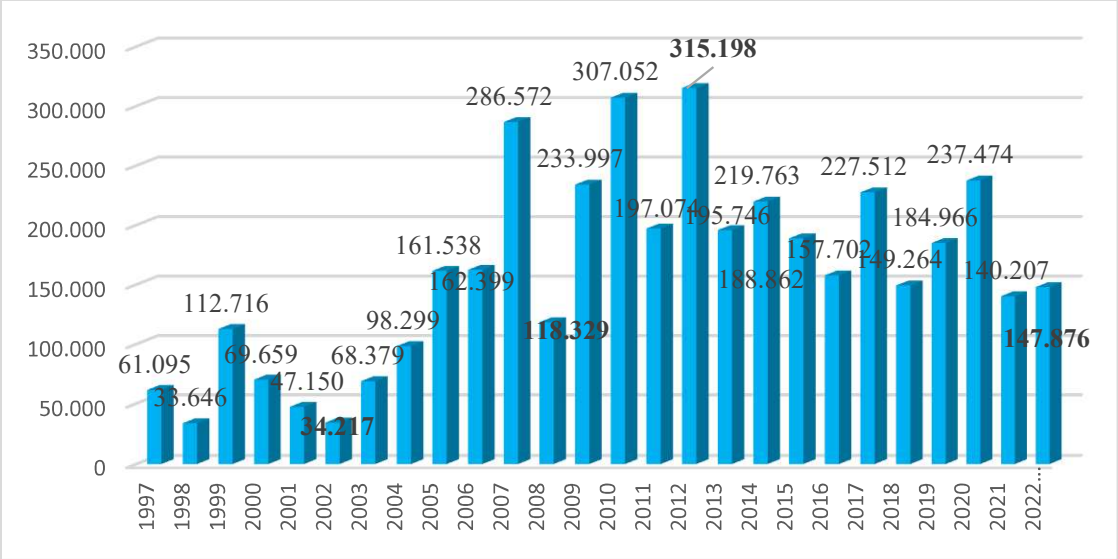


Source: https://www.mkk.com.tr/en/trade-repository-services/e-veri/fund-management-fees

Figure 1.2. Number of investors and portfolio value in CSD

The market capitalization values of the companies registered in the Istanbul stock market by year are given in Figure 1.3. Market capitalization is the value of a company as determined by the stock market. It is observed that the market value of companies, which had a continuous upward trend from 1997 to 2021, started to decrease significantly after 2012. It is observed that the value of all stocks in actual circulation, which experienced

decreases and increases between 2013 and 2019, increased again in 2020. Due to the Covid-19 pandemic experienced all over the world in 2020, there were significant reflections on the Turkish market as well as in the world markets, and there were decreases in the capital markets. It has been observed that the reflections of the Covid-19 pandemic on the Turkish stock market as a decrease with the data between 2020-2022.



Source: https://cmb.gov.tr

Figure 1.3. Market capitalization of stock exchange companies (million dollars)

1.3. Purpose of the Research

AI, which became an industry in 1980, emerged as a science in 1987 and started to take over every aspect of our lives with the developments in computer technologies in today's world. It is widely and effectively used in many fields, from health to the military and defense industry, industrial applications, and even the construction industry to archeology. In the field of finance, AI applications have become more popular in recent years due to their increasing accuracy, speed, and data size.

The aim of this research is to predict the long-term relationship between the ISE100 index value and the selected macroeconomic variables in the January 2001-January 2022 period by using Logit and Probit analysis and the deep neural networks method and to compare the estimation performances of the results obtained for both methods. Thus, the effectiveness of AI technologies, which emerged from developments in traditional

econometric methods and computer technologies and continued to develop rapidly in recent years, in time series analysis will also be measured. For the models used in the research, the most commonly used variables in studies on the market index in the literature were selected (W. Chen et al., 2017; Cheng et al., 2022a; Donaldson & Kamstra, 1999; Kara et al., 2011; Staub et al., 2015) and the meaningful variables obtained from the created meaningful models were given as input to the models created by ANNs and DL methods, and the effectiveness of the modeling methods was tested.

1.4. Contribution of the Research

Especially after the global economic crisis of 2008, researchers, policymakers, and investors have realized how important financial forecasts are in preventing financial crises and uncertainties. Therefore, economic decision makers needed modeling in order to develop policies that would minimize future uncertainties and create a portfolio.

Although there are many empirical studies in the literature examining the relationship between stock market indexes and macroeconomic factors (Henrique et al., 2019, pp. 226–251), the study estimating whether macroeconomic variables have any long-term effects on stock market index values using the Logit and Probit regression methods and the deep neural network method is very superficial and limited in number (Aggarwal, 2020; Davis & Karim, 2008; Kantar & Akkaya, 2018).

In the empirical study, both the degree of interaction between the real sector and the financial sector in Turkey will be measured, and a comparison of the DL method, which is a ML method from AI technologies and a new estimation method, with qualitative response of the Logit and Probit regression models from econometric methods will be made. The results of econometric and AI models created for this purpose are presented comparatively. Thus, a decision-making system that can be used to predict the short-term movement, trend, and price of the stock market has been developed and evaluated. It also offers a survey of the literature on the use of AI in financial analysis, with an emphasis mostly on the modeling process.

1.5. Definition and Limits of the Problem

In this research, the effectiveness of the DL modeling technique has been tested. The DL modeling technique is a ML method that has become increasingly important in our country and around the world in recent years and has been widely used in many fields but is used very limitedly in financial time series estimations for Turkey.

Qualitative regression models were created by using the monthly values of seven macroeconomic variables in order to show the relationship between ISE100 index values and macroeconomic variables in the 11-years (253-month) period between January 2001 and January 2022 in Turkey. Then, using the obtained models, DL models were created using the macroeconomic variables used.

1.6. Problem Solving

The first parts of the research will be formed as a result of theoretical and literature research. The conceptual and theoretical framework of financial time series and ANNs and DL methodology will be drawn from the information to be obtained from the Turkish and international literature, such as articles, theses, books, papers, and reports published in these sections.

In the application part, statistical analysis will be made using the ISE100 index and macroeconomic variable data. Eviews12, STATA15, and MATLAB R2021b package programs will be used in the application.

The data sets used in the analysis were compiled as monthly series from the Central Bank of the Republic of Turkey Electronic Data Distribution System (EVDS), the Turkish Statistical Institute (TUIK) statistics, the monthly statistical bulletins of the Capital Markets Board, the statistics of the Ministry of Treasury and Finance, and the Eurostat databases.

The macroeconomic variables used in this study were chosen by considering the studies by Chen, Roll, and Ross (1986) and Prantik and Vani (2004). The studies of Chen, Roll and Ross (1986) show that macroeconomic variables such as long- and short-term

interest rates, expected and unexpected inflation rates, industrial production, and price indices systematically affect securities prices and, thus, net asset values and returns. In addition, studies conducted in recent years have shown that various macroeconomic variables such as exchange rates, oil prices, gold prices, interest rates, production and price indices, and money supply significantly affect stock market indices (Aggarwal, 2020; Henrique et al., 2019; Kantar & Akkaya, 2018; Peng et al., 2021).

The ISE100 index was determined as the dependent variable in the established models, and seven macroeconomic variables were determined as the independent variables. The selected variables are the main variables that are most widely used in the literature and have a direct impact on the pricing of capital market instruments and this effect has been proven in the literature. In addition, these macroeconomic variables in the model are also used as indicators (benchmarks or fund benchmarks) for the ISE100 index.

1.7. Organization of the Thesis

In the first part of the research, in addition to the introduction, a brief overview of the world stock markets was made and information were given about the Istanbul Stock Exchange. After briefly introducing the purpose, scope, methodology, and data set of the research, in the second part, information is given about Logit and Probit models, which are qualitative response regression models and econometric modeling techniques that constitute the application part of the study.

From the third to the end of the seventh chapter, the concept of AI and its history, the development of AI, AI technologies, and the latest DL method are covered. In these sections, first of all, ANNs, which constitute the basic working system of the DL method, are theoretically examined, including their definition, history, general structure, types, learning algorithm, and learning rules of ANNs are given. Then, from the steps showing how to design an ANN model, information is provided on how to determine the input, output, and hidden layers of the network and the number of neurons in these layers, how to enter data, how to determine the parameters of the network, or, in other words, how the structure of the network can be created. In addition, the application areas, advantages, and disadvantages of ANNs are also given in this section. In the seventh chapter, which discusses the DL method,

which is based on ANNs, the elements that make up the architecture of the network and the methods used in learning are given wide coverage.

In the eighth chapter, detailed and current literature studies are given, and the references to the current thesis study are examined. Thus, a wide literature review, including financial studies using AI technologies, is included. Especially within the scope of the thesis, financial time series and AI techniques were analyzed, and the methods to be used in the thesis were determined.

In the last chapters, after the methodology and data set of the study were introduced, the application of analysis techniques was started. By establishing Borsa Istanbul Index prediction models, Logit and Probit regression analyses were made, and a DL design is made using the variables used in these prediction models, Logit and Probit regression analyses are made, and a DL design is made using the variables used in these models. The effectiveness of the techniques was tested with the findings obtained from the prediction models created using analysis techniques. A performance evaluation is carried out within the framework of the purpose of the study, and the prediction results of the models used are presented comparatively.

As a general evaluation, in this thesis study, the ability of the DL method to forecast the future trend of the nonlinear, variable, and complex ISE100 index will be tested. The DL method is one of the AI technologies that has become increasingly important in recent years all over the world and has been widely used in many fields but is seen to be used very limitedly in our country in financial time series estimations. It has also been observed that DL methods are used to a limited extent or are not used at all in the field of finance, and so this fact brings forward to new approaches in the field of finance.

2. QUALITATIVE RESPONSE MODELS

In regression models, it is assumed that the dependent variable, or explained variable, Y, is quantitative and, in contrast, the explanatory variables are quantitative and qualitative variables, or a mixture of them. On the other hand, qualitative response models are models in which the dependent variable is a qualitative variable and takes only two values. Three regression models developed for the two-valued response variable are briefly explained below.

2.1. Linear Probability Model (LPM)

Linear probability models are simple models that consist of two-valued dependent variables and can be estimated by the ordinary least squares method. It indicates the probability of choosing one of the two options presented. Since the dependent variable takes binary values 0 or 1, it is a restricted qualitative variable defined as a probability.

$$Y_t = \beta_1 + \beta_2 X_i + u_i$$

Figure 2.1. Linear probability model

The model expressed as a linear regression model is a typical linear regression model, and it is called the Linear Probability Model since the independent (explanatory) variable has two choices. Because when X_i is given, the conditional expected value of Y_i can be interpreted as E (Y_i /X_i) conditional probability of the occurrence of the event when X_i is given. Here, when Y_i =1, the first option is preferred and when Y_i =0, the second option is preferred, so the dependent variable will be able to take the values of 0 ve 1. Assuming E(u_i) = 0 when the expected value is taken, the obtained model will be as follows:

$$E(Y_i \mid X_i) = \beta_1 + \beta_2 X_i$$

Figure 2.2. Conditional probabililty model

In this model, the probability of choosing the first option will be P_i=P (Y_i =1) and if the second option is preferred, $1 - P_i$=P (Y_i =0). In this case, the expected value of the dependent variable is (Gujarati & Porter, 2009, p. 543);

$$\mathrm{E}(Y_i) = 0(1 - P_i) + 1(P_i) = \mathrm{P}(Y_i = 1) = P_i$$

Figure 2.3. The expected value of the dependent variable

$$\mathrm{E}(Y_i \,|X_i) = \beta_1 + \beta_2 X_i = P_i$$

Figure 2.4. The conditional expected value of the dependent variable

In classical regression models, the parameters are found by the least squares method. However, in the Linear Probability Model (LPM), like the dependent variable, the error term also fits the Bernoulli distribution as it only takes a binary value of 0-1. Therefore, although the model parameters are estimated by the least squares method, they do not meet the linear deviation-free estimator specifications. Therefore, there are some criticisms regarding the estimation and interpretation of this model. Because there is a problem of varying variance in error terms, parameter estimates will not be efficient, and this will make traditional significance tests questionable. For these reasons, the logit and probit models discussed in the literature are preferred as alternative modeling options instead of linear probability models to model binary variables (Gujarati, 2011, p. 154).

2.2. Logit Model

Although the linear probability model is simple to forecast and use, it is thought that the limitations arising from its disadvantages, such as the fact that the appropriate probabilities can be less than or greater than zero and the partial effect of any explanatory variable being fixed, can be overcome by using more complex binary response models (Wooldridge, 2010, p. 584).

When the linear probability model described in the previous section is expressed as $P_i = \beta_1 + \beta_2 X_i$ and (cumulative) logistic distribution function P_i, is the *i*'th probability of the second individual making one of the two choices;

$$P_i = \frac{1}{1 + e^{-(\beta_1 + \beta_2 X_i)}} = \frac{1}{1 + e^{-Z_i}} = \frac{e^Z}{1 + e^{Z_i}}$$

Figure 2.5. Logistic distribution function

Here, e = 2.7182 while the variable Z_i takes values in the range of -∞ to +∞, since P_i takes values between 0 and 1, the relationship between Z_i (or X_i) and P_i will not be linear and will not meet the two requirements mentioned earlier. For this reason, since this relationship will need to be converted to a linear form to be estimated can be written as $1 - P_i$;

$$1 - P_i = \frac{1}{1 + e^{Z_i}}$$

Figure 2.6. Probability of failure

Since the probability of an event occurring is P_i , the probability of failure will be $1 - P_i$ in this case will be;

$$\frac{P_i}{1 - P_i} = \frac{1 + e^{Z_i}}{1 + e^{-Z_i}} = e^{Z_i}$$

Figure 2.7. Odds ratio

This ratio is the probability of an event occurring to the probability of not occurring and is expressed as the Odds Ratio when the natural logarithm of this ratio is taken.

$$L_i = \ln\left(\frac{P_i}{1 - P_i}\right) = Z_i = \beta_1 + \beta_2 X_i$$

Figure 2.8. Logistic regression model

The resulting model is called the Logit regression model. So the logarithm L of the odds ratio is linear not only with respect to X but also with respect to the population coefficients (in terms of coefficient estimation). L is called Logit. The probability of making a particular choice while estimating the parameters of this model will be L_i = ln (1/0), if not L_i = ln (0/1). These expressions are obviously meaningless, so Logit models cannot be

estimated using the standard least squares method. In this case, the highest likelihood method can be used to estimate the population parameters (Gujarati & Porter, 2009, p. 556).

In Logit models, according to the number of independent variables, each slope coefficient is a partial slope coefficient and measures the change in the estimated logit when there is a one-unit change in the value of a particular explanatory variable (other variables are constant). A more meaningful interpretation of the slope coefficients is in terms of the odds obtained by taking the inverse logarithm of the slope coefficients (Gujarati & Porter, 2009, p. 563). Again, in binary models whose dependent variable is 0 and 1, the traditional measure of goodness of fit R^2, does not matter, what is important is the expected signs of the regression coefficients and their statistical significance (Gujarati, 2011, p. 158).

2.3. Probit Model

As mentioned earlier, an appropriately chosen cumulative distribution function should be used to explain the behavior of a binomial dependent variable. The error term of the linear probability model exhibits a nonlinear distribution, while the error term of the Logistic model exhibits a logistic distribution. The Probit model error term shows a normal distribution. Therefore, while a cumulative logistic distribution function is used in the Logit model, as in Figure 2.5, the normal cumulative distribution function is used in the Probit model, which is based on the utility theory developed by McFadden and is sometimes called the Normit Model. In the Probit model the *i*'th individual's decision to make or not make a particular choice depends on an unobservable *utility index* I_i (also called a latent variable) and I_i is determined by one or more explanatory variables such as X_i, the higher the value of I_i the index, the more likely the individual will make a choice. This I_i index is expressed as follows (Gujarati, 2011, p. 161; Gujarati & Porter, 2009, pp. 566–567):

$$I_i = \beta_1 + \beta_2 X_i$$

Figure 2.9. Utility index

What is the relationship of this unobservable index to the individual's decision to make a choice? Since Y = 1 if the individual makes a choice, and Y = 0 if he does not, assuming that there is a threshold value called I_i^*, for this index for each individual, if I_i

exceeds this threshold value, the individual makes the choice, if not, he does not. The threshold value cannot be observed like the utility index, but if we assume that they have the same mean and variance, not only predicts population coefficients given in Figure 2.9, but also obtains information about the unobservable index itself.

Under the assumption of normality, the probability that I_i^* and I_i are less than or equal to the standardized normal cumulative probability distribution is expressed and calculated as follows:

$$P_i = P(Y = 1 | X) = P(I_i^* \leq I_i) = P(Z_i \leq \beta_1 + \beta_2 X_i) = F(\beta_1 + \beta_2 X_i)$$

Figure 2.10. The probability that the threshold value is less than or equal to the utility index

Here, Z_i is the standardized normal variable that is, Z~N (0, σ^2). F , being the standard normal cumulative distribution function, it can be written as:

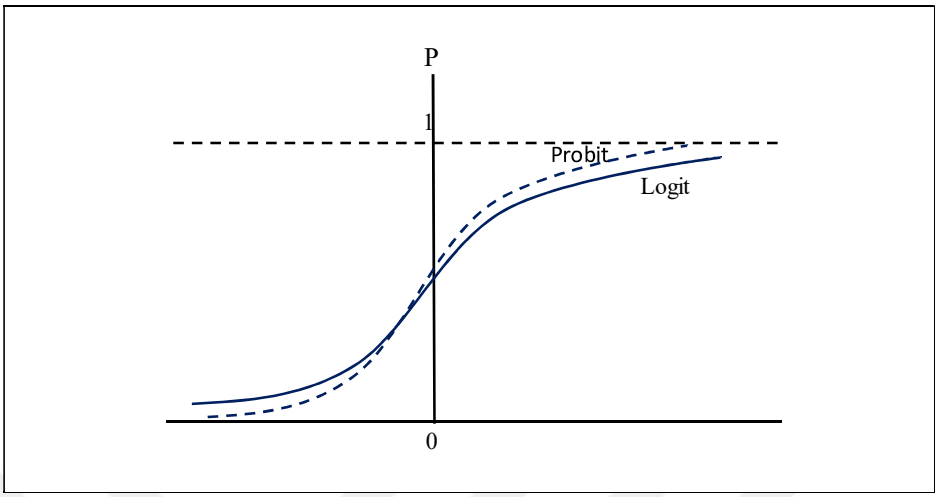
$$F(I_i) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{I_i} e^{-z^2/2} dz = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\beta_1 + \beta_2 X_i} e^{-z^2/2} dz$$

Figure 2.11. The standard normal cumulative distribution function

Similar to the Logit model, it is not appropriate to use the classical least squares method for the estimation of the Probit model. With the highest likelihood method, it is possible to obtain consistent estimates of the parameters of the Probit model (Gujarati & Porter, 2009, p. 567).

Although the Logit and Probit models give similar results, the main difference between the two models is that the logit distribution has slightly thicker tails. A logistically distributed random variable has a variance of about $\pi^2/3$ hereas a standard normally distributed variable has a variance of 1. However, in practice, there is no reason to prefer one over the other, and the Logit model is preferred over the Probit model because of its comparative mathematical simplicity (Gujarati, 2011, p. 163). The response probabilities of

these models are complex and difficult to forecast due to their nonlinear nature, but they have become popular recently (Wooldridge, 2010, p. 596).



Source:(Gujarati & Porter, 2009, p. 572).

Figure 2.12. The graphs of the Logit and Probit models

While natural logarithms of odds ratios are used in Logistic regression analysis, the cumulative normal distribution is used in Probit regression analysis.

3. ARTIFICIAL INTELLIGENCE

The work of human beings for self-knowledge has made significant progress with technological developments. At the latest point reached today, an AI chat robot called ChatGPT has been put into use by a research company called OpenAI. In the modern world, where computers and computer systems have become an indispensable part of life, major developments have occurred in software and hardware systems and data science. These developments have led to the widespread use of AI technologies, which have the ability to process information, learn, reason, solve problems, and make decisions quickly, accurately, and effectively in many areas where more information is obtained every day and the obtained information is used in many areas where computer technologies are used.

Studies of ANNs in neurobiology were drawn to the field of engineering with the creation of the first artificial nerve cell in 1943 by neurologist Warren McCulloch and mathematician Walter Pitts (Haykin, 1999, p. 38). The concept of AI has also taken its place in the literature after this study. The concept of AI, whose ability to solve problems using the ability to think and learn, which is one of the most important features of the human brain, has been discovered, has led to the emergence of new and more complex technologies due to the intense interest it has received from both researchers and commercial vendors.

A neural network is a computer program or wired machine designed to learn in a manner similar to the human brain. AI, on the other hand, is a combination of neural networks and neural network theory that was born from AI research or from research on designing machines with cognitive ability (Kutsurelis, 1998, p. 2). The AI research started with the Turing Test, which is the first software example that the British mathematician AM Turing, who managed to crack the Enigma code used by the German army for communication in the Second World War, explained in his article “Computing Machinery and Intelligence” published in the philosophy journal “Mind” in 1950. According to the test, which is known by its name today, if a person talking to a computer does not understand whether the person they are talking to is a computer or a human, that computer should be considered “intelligent” (Turing, 1950, pp. 433–460). Turing’s work deeply influenced AI studies, and the modern age of AI began. After this developments, AI and ML have turned into studies aimed at ensuring that machines have the ability to perform complex tasks. The

first attempt at learning neural networks was with the Hebbian learning rule, a neurophysiological theory developed by Donald Hebb in 1949. According to the theory, learning occurs through the exchange of messages between neurons in the human brain. In other words, he suggested that learning ability can be assigned to network structures by performing logical operations through networks of nerve cells (Haykin, 1999, p. 38).

Neural networks (NN), or more precisely ANN, are a technology with roots in a variety of fields, including neurological sciences, mathematics, statistics, physics, finance, computer science, and engineering (Haykin, 1999, p. 1). These computer systems, called intelligent systems, have continued their development with the contribution of AI. The developments in these two areas trigger each other, and their development processes continue.

ANNs are neural networks, also known as "simulated neural networks". It is a subset of ML and is at the heart of DL algorithms. These systems were inspired by the human brain, and they took their names and formed their structures by mimicking the way biological nerve cells (neurons) send signals to each other. Artificial neural networks (ANNs), Machine Learning (ML), and Deep Learning (DL) are called artificial intelligence (AI) (Quazi, 2022). These three concepts are shown in the literature in the Figure 3.1 below.

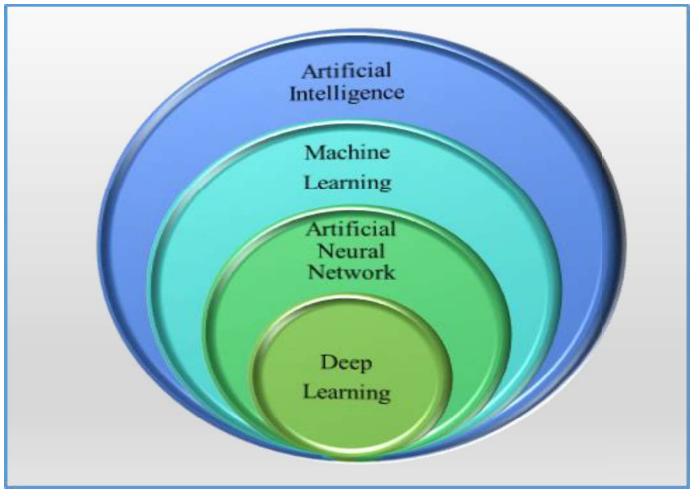


Figure 3.1. AI, ML, ANN and DL

AI and machine learning (ML) are two emerging and interrelated technologies in finance. Concepts such as AI, ML, DL, and data science are confusing and intertwined. ANNs, a sub-branch of AI, are a statistical approach created to develop forecasting models.

Basically, ML is a particular embodiment of AI that develops techniques and uses algorithms that enable machines to recognize patterns in datasets. On the other hand, DL is a subset of ML that equips machines with the techniques needed to solve complex problems. Data science is a separate field of study that applies AI, ML, and DL to solve complex problems (Goodell et al., 2021, p. 2; Öztemel, 2012, pp. 12–15).

AI, defined as understanding the human thinking structure and attempting to develop computer operations that will reveal the likes of it, is an attempt to give programmed computers the ability to think. AI studies, which have been going on since the 1950s for the development of systems that think and act like human beings, have spread to fields such as engineering, neurology, and psychology since they were aimed at imitating humans at one point. Despite significant advances, the point reached today in studies for developing systems that can think and behave like humans is that AI has not yet been fully developed. Leaving aside the discussions on the possibility of AI, studies on this subject continue along with studies in different fields that support this field.

ANNs technology is one of the various fields that emerged within the scope of AI studies and that at some point provided support to those studies. Therefore, ANNs, which is a sub-branch of AI, forms the *basis of systems* that can learn. ANNs that imitate the neuron, which is the basic processing element of the human brain, in a formal and functional way, are programs created for a simple simulation of the biological nervous system in this way. Therefore, AI technology, which is thought to be able to transfer the ability of learning by experimenting (experience) to the computer environment, provides an incredible "learning from input data" capacity to a computer system and offers many advantages. This technology, which offers various advantages and develops day by day, is used in the fields of economy and statistics, as it is in many other fields today. They are frequently used in areas that require the definition of the structure contained in the data, such as forecasting and prediction, because they are known as "Universal Function Approximators".

Technically, the most basic task of an ANN is to determine an output set that can correspond to an input set shown to it. In order to do this, the network is trained with examples of the relevant event and gains the ability to generalize. ANNs are also called *correlated networks, parallel distributed networks, and neuromorphic systems* (Öztemel, 2012, p. 30).

4. DEVELOPMENT PROCESS OF ARTIFICIAL INTELLIGENCE

The history of AI begins with an interest in neurobiology and the application of the obtained information to computer science. When AI studies are examined, it is discovered that the majority of the research is a continuation of each other. In other words, the oldest developed AI structures and learning algorithms are still used today, and developments in this field are strongly related to previous developments. Due to the succession of these studies, AI studies developed rapidly, but some studies conducted later showed that previous studies were insufficient, which caused the studies in the field of AI and the support given to these studies to pause until these problems were resolved (Öztemel, 2006, p. 37).

For many years, scientists have studied how the human brain works and its functions. The first work providing information on brain functions was published in 1890. It is known that before 1940, some scientists such as Helmholtz, Pavlov, and Poincare worked on the concept of an ANN. However, it cannot be said that these studies have engineering value (Öztemel, 2006, p. 37). The first foundations of ANNs were laid in an article published in 1943 by neuropsychologist McCullogh and mathematician Pitts, who started research in the early 1940s. McCullog and Pitts made a model in the form of a simple mathematical representation of the biological neuron (Zurada, 1992, p. 30). Thus, inspired by the computational ability of the human brain, the first neuron model or simple neural network was modeled, and the first mathematical model of the artificial nervous system was developed. They tried to determine the learning rules by revealing that cells work in parallel with each other (McCulloch & Pitts, 1943, p. 118).

In 1949, Donald Hebb, a psychologist at McGill University, explained a learning rule that improves the weight values of an artificial neural network consisting of artificial cells in his book called “The Organization of Behavior” (Haykin, 1999, p. 38). This rule, called the Hebb Learning Rule, still forms the basis of many learning rules today. The concept of generating adaptive responses with random networks was introduced by Farely and Clark in 1954 (Anderson & McNeill, 1992, p. 12). This concept was developed by Rosenblatt in 1958 and by Caianiello in 1961. The perceptron model, especially developed by Rosenblatt, was

developed later and formed the basis of multi-layer perceptrons, which were revolutionary in ANNs (Zurada, 1992, p. 19).

With the development of computers in the 1950s, it became possible to model the foundations of theories about human thought. Nathanial Rochester, a researcher at IBM research labs, led efforts to create a neural network simulation. Although the first attempt was unsuccessful, subsequent attempts were successful. After this stage, the search for traditional computational methods has left its place for the investigation of neural computation methods. Until then, Rochester and his team had defended "thinking machines" by citing evidence of their own work. The "Artificial Intelligence Dartmouth Summer Research Project", which was held in 1956 and later on, increased the talk and therefore the evidence on AI and ANNs. One of the results of this research is that it encourages interest in both AI and ANNs research (Anderson & McNeill, 1992, p. 17).

In 1957, Frank Rosenblatt, a neurobiologist at Cornell University, developed the Simple Perceptron Model (Perceptron), which will have an important role in both the development and pause of artificial neural network studies. Rosenblatt was interested in the eye processes of a fly. Inspired by the fact that most of the processes that tell a fly to escape are done inside the fly's eye instead of its brain, a hardware device called the perceptron was established and the network structure developed, which is the oldest artificial neural network model of today (Anderson & McNeill, 1992, p. 17). Perceptron, developed by Rosenblatt, pioneered an important development in the history of ANNs. Because this model forms the basis of Multilayer Perceptrons that will be developed later and will be revolutionary in ANNs, this simple perceptron model is still used today (Zurada, 1992, p. 19).

In 1959, Bernard Widrow from Stanford University and his student Marcian Hoff developed two Perceptron-like models, which they named ADALINE (Adaptive Linear Elements) and MADALINE (Multiple Adaptive Linear Elements (Öztemel, 2006, p. 38).

The work done by Marvin Minsky and Seymour Papert in 1969 turned into a book called "Perceptrons". The authors specifically claimed in this book that sensors based on ANNs have no scientific value and cannot solve nonlinear problems, and they used the

example of not solving the XOR[1] logic problem to prove their point. This situation has caused AI studies to enter a period of stagnation. Although the studies were interrupted in 1969 and the necessary financial support was cut, some scientists continued their studies. In particular, the studies of researchers such as Shun-Ichi Amari, Stephen Grossberg, Gail A. Carpenter, Teuvo Kohonen, and James A. Anderson came to fruition in the 1980s, and new studies on AI began to be put forward. Adaptive Resonance Theory (ART), developed by Grossberg and Carpenter in 1976, has been the most complex artificial neural network developed for teacherless learning (Öztemel, 2006, p. 39; Zurada, 1992, p. 19).

Studies by Hopfield in 1982 and 1984 showed that ANNs can be generalized and can produce solutions to problems that are difficult to solve, especially with traditional computer programming. Hopfield used similar studies created in 1972 by electrical engineer Kohonen and neuropsychologist Anderson, working in different disciplines and unaware of each other. In order to solve technical problems such as optimization, he developed the nonlinear dynamic Hopfield network and the Kohonen "SOM, Self-Organizing Feature Maps" network, which is a learning network without a trainer (Akel & Karacameydan, 2018, p. 53). Hopfield also revealed the mathematical foundations of ANNs in these studies (Anderson & McNeill, 1992, p. 18). These studies formed the basis for unsupervised learning rules that will be developed later.

In 1986, David Rumelhart and James McClelland, in their work called "Parallel Distributed Processing", showed that the previously claimed flaws on this subject could be overcome by developing a new learning model, the Back Propagation Algorithm, which is a learning model in feedforward models. Because the XOR problem, which the single-layer perceptron could not solve, was solved by the discovery of multi-layer perceptrons. Today, various versions of this learning method are used in a variety of settings (Öztemel, 2006, p. 38). The development of this algorithm, which is still one of the most used training algorithms, has revolutionized the field of ANNs (Zurada, 1992, p. 20). These developments have led to an increased interest in ANN.

[1] (Executive OR) a logic problem can be defined as a special case of the problem of classification of points in a unit hypercube. Each point in the hypercube belongs to either class 1 or 0. For detailed information, see (Zurada, 1992).

As a result of the increasing interest in AI, various conferences were held on this subject. One of these conferences is the "US-Japan Joint Conference on Cooperative/Competitive Neural Networks" organized by Japanese and American participants in Kyoto. During the conference, the Japanese presented their fifth-generation research hereupon, there was a fear among Americans that they are falling behind Japan. Because of this fear, funds were quickly transferred to AI research in the United States, and AI was developed on a regular basis (Anderson & McNeill, 1992, p. 18).

By 1985, the American Institute of Physics began to organize annual meetings on "Neural Networks for Computing" (Anderson & McNeill, 1992, p. 18). With the 1st International Conference on ANNs (1st International Conference on Neural Networks) held by the Institute of Electrical and Electronics Engineers (IEEE) in 1987, studies in the field of ANNs started to become widespread all over the world.

In 1988, as an alternative to multilayer sensors, Broomhead and Lowe developed the Radial Basis Functions (RBF) model and produced very successful results, especially for filtering problems. Later, Specht developed Parabolistic Networks (PNN) and General Regression Networks (GRNN), which are more advanced forms of these networks. Since 1987, ANNs have been discussed in different symposiums and conferences, and new learning techniques and models have been put forward. Today, with the decrease in the size of computers and the increase in their capacities, ANNs have ceased to be theoretical and laboratory studies and have begun to create systems used in daily life and to be useful to people practically (Öztemel, 2006, p. 40).

The widespread use of ANNs in the field of finance is occurring in parallel with the development of the backpropagation algorithm by Rumelhart and McClelland. Multi-layered networks are needed in the solution of financial problems since financial problems include a linear as well as a non-linear structure. Although there are multi-layered networks that have been developed before, these networks have not been trained. However, with this algorithm developed by Rumelhart and McClelland, this problem has been overcome, and ANNs have attracted great interest in the field of finance. Today, ANNs are effectively applied in financial forecastings, such as for financial indices, stock prices, portfolio diversification, bond valuation, loan repayment rates, real estate prices, and the bankruptcy of enterprises.

In summary, AI was used in customer orders by DEC (Digital Equipment Corporation), an American computer and technology company, and when it was seen that 40 million dollars were saved, the interest in AI increased again, and countries started to make serious investments in this field. In 1975, Holland produced the genetic algorithm based on the principle of natural development with living things, and in 1976 he developed the Sejnowski Boltzman Machine and applied the backpropagation algorithm to it. In 1978, Sutton and Barto developed a reinforcement learning model, and in 1982, hundreds of studies in this field were presented at the international ANN conference on PC-based neuro, with Hecht-Nielsen's TRW MARK III, the first modern electronic neuro computer (Kubat, 2019, p. 621).

Martens (2010) and Martens and Sutskever (2011) developed powerful deep and recurrent neural network (RNN) models that employ momentum stochastic gradient descent in datasets with well-designed long-term dependencies (Long Short Term Memory, LSTM) Hessian-Free found that it showed higher training performance than the performance obtained from optimization (Sutskever et al., n.d., pp. 1–9). AI is a structure that is directly affected by these disciplines, which affect many disciplines such as computer science, engineering, biology, psychology, mathematics, logic, philosophy, business, finance, and linguistics. For this reason, its applications are seen in many fields, from medicine to the military, from economics to meteorology. Of course, its use in so many different fields appears impressive, but it has found such a wide range of applications as a result of years of research from the 1940s to the present (Metlek & Çetiner, 2021, p. 3).

Table 4.1. Some important advantages in neural network research

Researcher	Contribution
McCulloch & Pitts	ANN models with adjustible weights (1943)
Rosenblatt	The Perceptron Learning Algorithm (1957)
Widrow & Hoff	Adaline (1960), Madaline Rule I (1961) & Madaline Rule II (1988)
Minsky & Papert	The XOR Problem (1969)
Werbos (Doctoral Dissertation)	Backpropagation (1974)
Hopfield	Hopfield Networks (1982)
Rumelhart, Hinton & Williams	Renewed interest in backpropagation: multilayer adaptive backpropagation (1986)
Vapnik, Cortes	Support Vector Networks (1995)
Hochreiter& Schmidhuber	Long Short Term Memory Networks (1997)
LeCunn et.al.	Convolutional Neural Networks (1998)
Hinton & Ruslan	Hierarchical Feature Learning in Deep Neural Networks (2006)

In the Table 4.1, the main developments in AI studies are summarized by year. After the 2000s, advances in AI research accelerated. The increasing interest in this field has led to the emergence of new AI techniques based on ANNs. One of them is DL technology. However, in order to understand DL, it is necessary to understand the structure and workings of ANNs. These issues will be discussed in detail in the following sections.

5. ARTIFICIAL INTELLIGENCE TECHNOLOGIES

AI systems, also called intelligent systems in the literature, are technology applications developed to perform tasks similar to human cognitive abilities, although there is no generally accepted definition of their concept (Alaloul & Qureshi, 2020, p. 1). Developments in software and hardware systems have led to access to more information and data and the efficient use of the obtained data, making the work of decision units easier. The fact that problems that cannot be solved or that can be difficult to solve with human intelligence can be solved and decisions can be made thanks to algorithms in code sequences for many fields has increased interest in this field and led to the emergence of many new technologies in the field of AI. In these systems, the ways and stages of people's decision-making are imitated. The most prominent of these technologies today are summarized below.

5.1. Expert System

Computer programs are technologies that have been developed and can solve a problem in the same way that experts in a certain field can solve based on their expertise. Its main fields of application are medicine and biomedicine (Staub et al., 2015, p. 1478). Experts use their knowledge and experience to solve problems. The computer should understand and store this knowledge and experience (Öztemel, 2006, p. 15). It is used in applications that need both machine and human intervention. These systems consist of three parts: The Rule Base, The Database, and The Rule Analyzer (Kubat, 2019, p. 625).

5.2. Machine Learning

ML is the field of computer science that creates algorithms to process large amounts of data and learn from it. It can learn the relationships between the inputs and outputs of events using examples. By interpreting the learned information and similar events, decisions are made or problems are solved. In order to understand ML, it is necessary to understand learning. Learning is the process of "improving behaviors through the discovery of new information over time" (Öztemel, 2006, p. 17-21). Since this is classical programming in AI technology, linear or non-linear relationships cannot be established between the data and the results, and rules cannot be created. Since this is classical programming in AI technology,

linear or non-linear relationships cannot be established between the data and the results, and rules cannot be created. In this case, the solution cannot be found. For this reason, ML algorithms, which have mostly been used in military and commercial fields in recent years, are divided into three subgroups: supervised learning, unsupervised learning, and reinforcement learning algorithms, due to their ability to establish a relationship between data and results (Metlek & Çetiner, 2021, p. 5).

It contains classical algorithms for various tasks such as classification, regression, and grouping in ML technology. Since it is a technology that trains algorithms on data, the more data that is provided, the better the algorithm learns. In other words, ML is an optimization algorithm.

5.3. Fuzzy Logic

Fuzzy logic, which was first used by L. Zadeh in 1965, is based on the combination of the advantages of ANNs such as learning and decision-making. In addition, fuzzy logic-neural networks (ANNs) are one of the most suitable methods for time series calculations (Staub et al., 2015, p. 1478). Today, many events take place in uncertain conditions. For events that are not known with certainty, experts use definitions such as “normal”, “high, approximate and low”. Fuzzy sets are developed to represent and use non-statistical uncertainties with data and information. Every problem has its own uncertainty. This method tries to describe this uncertainty mathematically. Therefore, these systems are suitable for uncertain or approximate inference and especially for systems whose model cannot be expressed mathematically (Kubat, 2019, p. 623). They are technologies that make it easier to process uncertain information and make decisions in situations where precise numbers cannot be expressed (Öztemel, 2006, p. 27).

5.4. Genetic Algorithms

The genetic algorithm process, which is likened to the natural evolution process, includes operators used in natural evolution such as reproduction, crossover, and mutation. It is a technology used in solving complex optimization problems. In order to solve a problem, random initial solutions are determined, and better solutions are sought by

matching these solutions with each other. This search continues until the best result is produced. In the technology of genetic algorithms, it is accepted that the features that will produce the desired result in solving problems pass to new solutions obtained from initial solutions, and from them to later solutions, through inheritance (Öztemel, 2006, p. 17). In other words, it is a method dependent on natural genetics and natural selection mechanisms by coding parameter sets and using objective function and functioning information. Mechanical learning, economics, social sciences, and information systems are application areas (Staub et al., 2015, p. 1478).

5.5. Artificial Neural Networks

ANNs, or in other words, "neural networks", are systems that examine the structure of the human brain, consisting of neurons and learning methods. In the 19th century, the studies of psychologists and neuropsychologists to understand the human brain formed the basis of ANNs. The first scientific study on artificial neural networks started with McCulloch and W. Pitts, as mentioned earlier.These systems try to fulfill the functions of the brain, developed by taking into account the brain structure and the interaction of neurons with each other (Kubat, 2019, p. 623). These systems also focus on learning. They learn the relationships between events from examples, and then make decisions using the information they have learned about examples they have never seen (Öztemel, 2006, p. 27).

All of the AI technologies were developed to serve people in their daily lives and are still developing rapidly. The basic structure of AI technologies is an ANN system, which is created by simulating the working style of the system consisting of simple biological nerve cells (neurons), and an ANN system created according to this structure also forms *the basis of all artificial intelligence studies*. ANNs, which enable computers to learn, will learn to operate by trial and error and use this knowledge in problem-solving.

In the literature, it is seen that ANNs are used in many fields, from medicine to defense, from psychology to finance, due to their superior success in forecasting and classification studies. ANNs, which are briefly introduced here, will be discussed in more detail in a separate section, since DL, which forms the basis of our study, is a sub-branch of ANNs and works according to this system.

5.6. Deep Learning

Deep learning is a complex ML algorithm modeled on the workings of the human brain (Bengio et al., 2015, p. 8). DL describes algorithms developed through supervised and unsupervised learning that analyze data with a logical structure similar to how a human would draw conclusions from an event. Therefore, it has to use a layered algorithm called a deep learning ANN to achieve the main goal. For this reason, the interest in this field has increased even more, as it has been seen that the human brain gives better results than the standard ML models inspired by the biological neural network, and the studies reach results that were thought to be impossible (Fathi & Maleki Shoja, 2018b, pp. 229–230).

Since DL technology and the ANNs on which it is based are the main subjects of the study, they will be explained in detail in the following sections.

6. ARTIFICIAL NEURAL NETWORK

6.1. Definitions of Neural Networks in the Literature

There is no single, agreed-upon definition for ANNs. Many definitions can be found in broad and narrow contexts. In fact, some researchers argue that instead of giving a general definition for ANN, ANN types should be defined within themselves. However, some of the definitions in the literature related to ANN are as follows:

ANNs are logical software developed to imitate the working mechanism of the human brain and to perform the basic functions of the brain such as learning, remembering, and deriving new information through generalization (Yazici et al., 2007, p. 65).

ANNs can be defined as a computer program written for a mathematical formula that will enable the parameters to be adapted with the help of a set of examples, in the shortest and simplest way without going into technical details (Anderson & McNeill, 1992, p. 4).

The ANN is a system formed by connecting neurons in different ways, which are a simple processing elements that, in essence, mimic the way the human brain works. Each neuron receives signals from other neurons or from outside, combines them, transforms them, and produces numerical results (G. Zhang et al., 1998, p. 37).

This model, inspired by the biological working system of the human brain, is an information-processing model that describes processes and gives examples. ANN is a network of highly connected and organized neurons in layers (G. P. Zhang, 2003, pp. 163–164).

According to the definition of an artificial neural network made by Kohonen, who is well-known in the artificial neural network literature, “ANNs are a hierarchical organization of many simple elements connected in parallel, interacting with real-world objects in a way similar to the biological nervous system” (Kohonen, 2001, pp. 71–72).

An ANN is a parallel and distributed single- or multi-layered computing system consisting of many simple processing elements interconnected with one-way signal channels in a weighted form (Kamruzzaman et al., 2006a, p. 3).

A more comprehensive and generally accepted definition made by Haykin is: "A neural network is a densely parallel distributed processor composed of simple processing units that has a natural tendency to accumulate experiential information and enable it to be used" (Haykin, 1999, pp. 1–2).

This processor is similar to the brain in two ways:

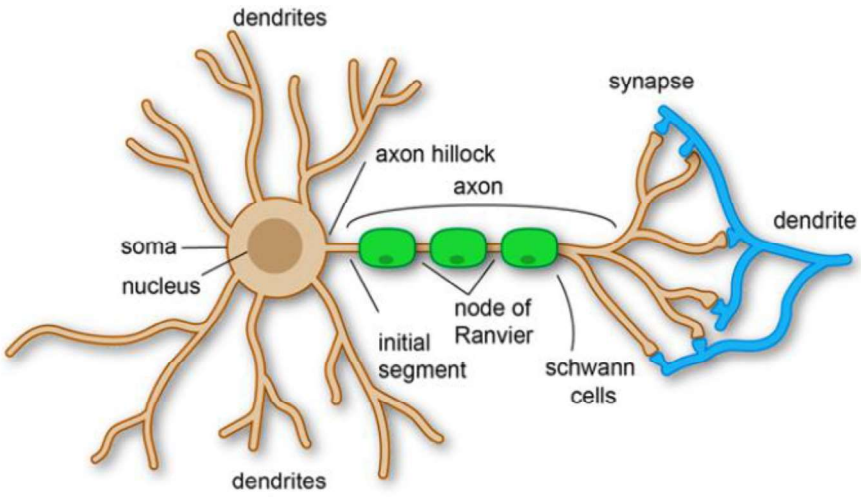
- ✓ Knowledge is acquired by the network from the environment through a learning process.
- ✓ Interneuron connection strengths, also known as synaptic weights, are used to accumulate the obtained information.

6.2. Biological Nervous System

The human nervous system has a very complex structure. The brain is the central element of this system, and it is estimated that there are about 10^{10} neurons (nerve cells) connected to each other by subnets, and they have more than $6x10^{13}$ connections. The human nervous system has a very complex network structure. Nerve cells are specialized cells that carry information through an electrochemical process. These nerve cells come in different shapes and sizes. Some are only 4 microns (4/1000 millimeters) wide, while others are 100 microns wide. Although different types of nerve cells have differences in terms of shape and function, as shown in Figure 6.1, they all consist of 4 different regions: the dendrite, the nucleus and soma or cell body, the axon, and the junction or synapse (Anderson & McNeill, 1992, p. 3).

Synapses can be viewed as connections between nerve cells. These are not physical connections, but rather gaps that allow electrical signals to pass from one cell to another. These signals go to the soma, which processes them. The nerve cell creates an electrical signal and sends it to the dendrites via the axon (Öztemel, 2006, p. 47). Dendrites form the

input channels for a cell. Dendrites convert these signals into small electric currents and transmit them to the cell body. The cell body processes the signals coming through the dendrites and converts them into output. These outputs produced by the cell body are sent via axons to be input to other neurons (Anderson & McNeill, 1992, p. 4).



Source: hhttps://en.wikipedia.org/wiki/Dendritetml

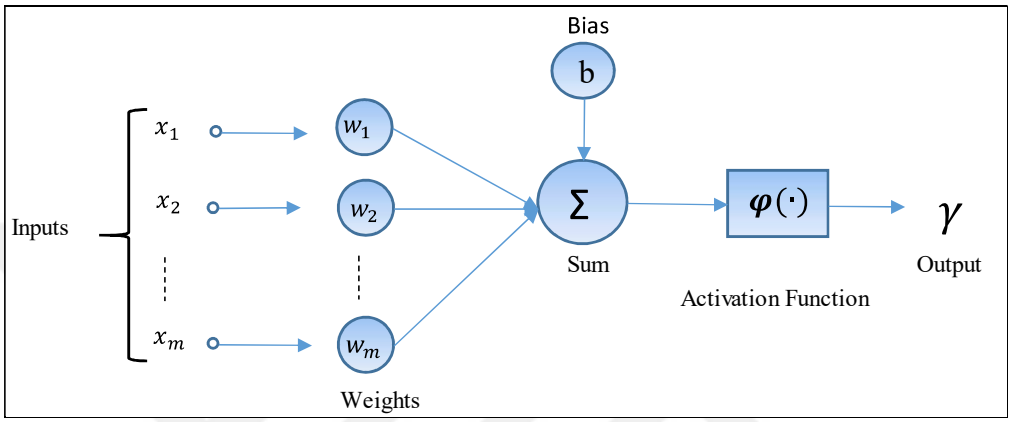
Figure 6.1. Structure of a typical neuron

In a neuron, the dendrite receives input signals, the soma processes input signals, the axon converts input signals into output signals, and finally, the synapses provide electrochemical contact between neurons. In the simulation of this structure adapted to ANNs, each process element is named as follows: Synapses represent weights; dendrites represent the summation function, the soma activation function, and the axon output.

6.3. Artificial Neural Networks Elements

An ANN consists of a large number of simple processing units called neurons, units, cells, nodes, and process elements. Just as biological neural networks have nerve cells, artificial neural networks also have artificial nerve cells. ANNs are a program designed to simulate the way the simple biological nervous system works. Artificial neural networks contain simulated nerve cells, and these cells connect to each other in different ways to form the network. Figure 6.2 presents the structure of a general artificial neuron (Kamruzzaman et al., 2006a, p. 3).

In the artificial neuron in Figure 6.2, the inputs are given by X_i and i=0, 1, 2,…, n; and the weights are shown with mathematical symbols as W_j ve j=0,1,2,…,n. Each of these input values is multiplied by a link weight. In the simplest form, these products are summed up and sent to a transfer function, where the result is produced. This result is then converted to output (Y). Here, the input vector is shown as $\{X_1, X_2, \ldots, X_n\}$ and the appropriate weights vector is shown as $\{W_{j1}, W_{j2},\ldots,W_{jn}\}$.



Source: Haykin, 1999, p.11.

Figure 6.2. General structure of artificial neuron

ANNs, inspired by the biological nervous system, have the capacity to learn, memorize, and reveal the relationship between data. Just as biological nervous systems are made up of nerve cells, ANNs are made up of artificial nerve cells. An artificial nerve is a simple structure that mimics the basic functions of a biological nerve. Neurons in the network take one or more inputs and give a single output. This output can be output to the outside of the artificial neural network, or it can be used as an input to other neurons. Artificial nerve cells are called *process elements (neurons)* in engineering science. Although there are differences in the cell models developed, in general, each processing element consists of 5 basic components: inputs (X), weights (W), summation function (Σ), activation function (Ψ(.)) and cell output (γ). These components can be explained as follows (Haykin, 1999, pp. 11–12; Öztemel, 2006, pp. 48–51):

6.3.1. Inputs

It is the information coming to an artificial nerve cell from the outside world or other cells. These inputs are determined by the examples the network is asked to learn. The

artificial neuron can receive information from the outside world, as well as from other cells or from itself. A neuron can usually receive many simultaneous inputs. Each input has its own relative weight.

6.3.2. Weights

Inputs show the information coming to the artificial neuron, while weights show the importance of the information coming to the artificial neuron and its effect on the cell. The weight 1 (w_1), in Figure 6.2 shows the effect ot input 1 (x_1) on the cell. The fact that the weights are large or small does not mean that they are important or unimportant. A weight value of zero may be the most important event for that network. Weights can have positive or negative values. A weight of value plus or minus indicates whether the effect is positive or negative, while a zero indicates that there is no effect. Weights can have variable or fixed values. All connections that provide the transmission of inputs between neurons in the artificial neural network have different weight values. Thus, the weights act on each input of each neuron. They are connected to a cell with information, inputs, and weights received from the external environment. The value of the net input is calculated by using the activation function by calculating with different methods and the net output value is obtained. The important thing here is that the error between the actual values and the output values is the least.

6.3.3. Summation Function

The summation function is a function that calculates the net input to a cell. Although different functions are used for this, the most common one is to find the weighted sum. Here, each incoming input value is multiplied by its own weight. Thus, the net incoming input is found. However, in some cases, the addition function may not be such a simple operation. Instead, it can be much more complex, such as minimum (min), maximum (max), mode, product, majority, or a few normalization functions. The algorithm that will combine the inputs is usually also determined by the chosen network architecture. These functions can generate values differently, and these values are forwarded. The total input of any neuron is equal to the weighted sum of the values from other neurons and the threshold value (Kamruzzaman et al., 2006a, p. 3).

$$NET = W_j X_i + b$$

Figure 6.3. Summation function

The summation function is written as follows:

NET : Network, the net input of the neuron,

W_{ji} : the weight of the connection between neurons j and i,

X_i : the output of the neuron i,

W_{j0} : b: represents the threshold value (bias)

Table 6.1. Summation functions in ANNs

Summation Function	Equation	Explain
Weighted Total	NET =$\sum w_{ij}x_i$	Inputs and weight values are multiplied.
Multiplication	NET = $\Pi w_{ij}x_i$	Inputs and weight values are multiplied.
Minimum	NET = Min ($w_{ij}x_i$)	The lowest calculated value is taken.
Maximum	NET = Max ($w_{ij}x_i$)	The highest calculated value is taken.
Incremental Total	NET = NET (old)+ $\sum w_{ij}x_i$	Weighted total is calculated.

Source: Öztemel (2006), p.50.

In Table 6.1, examples of different addition and combination functions used in research in the literature are given. Usually, the best addition function is determined by trial and error. Besides, determining the aggregation function completely depends on the designer's own foresight.

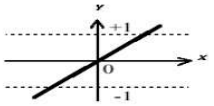
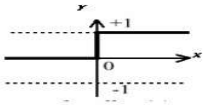
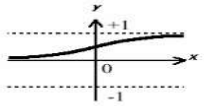
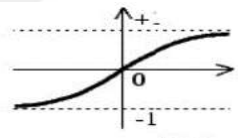
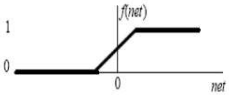
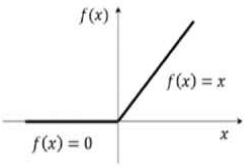
6.3.4. Activation Function

It is the function that determines the output that the cell will produce in response to this input by processing the net input obtained by the summation function. The purpose of the activation function is to allow the output of the addition function to change when time is involved. As in the addition function, various types of activation functions can be used in the activation function, depending on the function the cell will perform. For example, in multilayer perceptron models, the activation function must be a differentiable function.

With the NET value obtained by the summation function in ANNs, the next step which is the activation function, is passed. It determines the output to be produced by the current network structure by processing the obtained NET input value with the selected activation function. By means of nonlinear functions that can be used instead of a threshold function or a linear function of the inputs, it is possible to model nonlinear input-output relations with ANNs. The activation function is the function that provides linear or nonlinear matching between the input and the output (Eğrioğlu et al., 2019, p. 19).

The choice of the activation function largely depends on the data of the neural network and what information it is desired to learn. Generally, the chosen activation function is a nonlinear one rather than a linear one. Because in linear functions, the output is linearly proportional to the input. The most suitable function for a problem is determined as a result of the designer's trials. Some activation functions used in the literature are given in Table 6.2. The sigmoid function (Logsig) or hyperbolic tangent function (Tansig) is preferred as the activation function in the Multilayer Perceptron (MLP) model, which is the most widely model used today. There is no strict rule about which of the join or activation functions to use. The researcher decides which one to use in which layer by trial and error.

Table 6.2. Some activation functions

Activation Function	Equation	Graph
Linear Function	F(NET) = NET	
Step (Heaviside) Function	$F(NET) = \begin{cases} 1, & if \quad NET > \theta \\ 0, & if \quad NET \leq \theta \end{cases}$	
Sigmoid Function (Logsig)	$F\ (NET) = \frac{1}{1 + e^{-NET}}$	
Hyperbolic Tangent (Tansig)	$F(NET) = \frac{e^{NET} + e^{-NET}}{e^{NET} - e^{-NET}}$	
Threshold Value Function	$F(NET) = \begin{cases} 0, & if \quad NET \leq 0 \\ NET, & if\ 0 < NET < 1 \\ 1, & if \quad NET \geq 1 \end{cases}$	
Rectifier Linear Unit, ReLu	$F(x) = \max(0, x)$	

Source: Howard Demuth and Mark Beale (2004); p.1-7; Kamruzzman Joarder, Rezaul K. Beggand and Ruhul Sarker (2006); p.4.

6.3.5. Output

It is the output value determined by the activation function. The output produced is sent to the outside world or to another cell. The cell can also send its own output as input to itself. In fact, there is only one output value from a neuron. The same value goes to more than one neuron as input. An artificial neural network can have a single or multi-layered structure. The number of cells in the layers may differ from each other. As the number of cells in the intermediate layers increases, the process may become more difficult, but the

result may be better. Therefore, the number of neurons to be used in the study, the number of layers, the number of neurons in the layers, and the selection of the addition and activation functions will enable the network to learn and make the closest estimation to the truth with the least error. This is possible only by trial and error. This procedure is carried out using computer programs that have been created. Matlab, Python, and Neurosolution are some of these programs.

6.4. General Structure of Artificial Neural Networks

Through connections, millions of biological nerve cells (neurons) combine to form the brain; ANNs are formed by the combination of many artificial nerve cells. In other words, layers are formed when neurons come together in the same direction, and ANNs are formed by the combination of these several layers. Some neurons in this structure are in contact with the external space to receive inputs and some to transmit outputs. All other neurons are hidden layers. In this case, connecting the layers to each other in different ways leads to the emergence of different network architectures. In the early years of ANNs, some researchers randomly created connections between neurons and encountered negative results (Kamruzzaman et al., 2006b, p. 5; Kubat, 2019, pp. 668–669).

The easiest way to design a structure is to separate the elements into layers. There are three parts to the layering here. These are grouping *neurons* into layers, grouping *connections* between layers, and finally grouping *summation* and *transfer functions*. This situation is illustrated in detail in Figure 6.4. In other words, the coming together of nerve cells cannot be random. In general, they come together in 3 layers: the *input layer*, *hidden layer* or *intermediate layer*, and *output layer*, and in each layer, in parallel, they form the network (Kubat, 2019, pp. 668–669; Öztemel, 2006, pp. 52–53). These layers are:

6.4.1. Input Layer

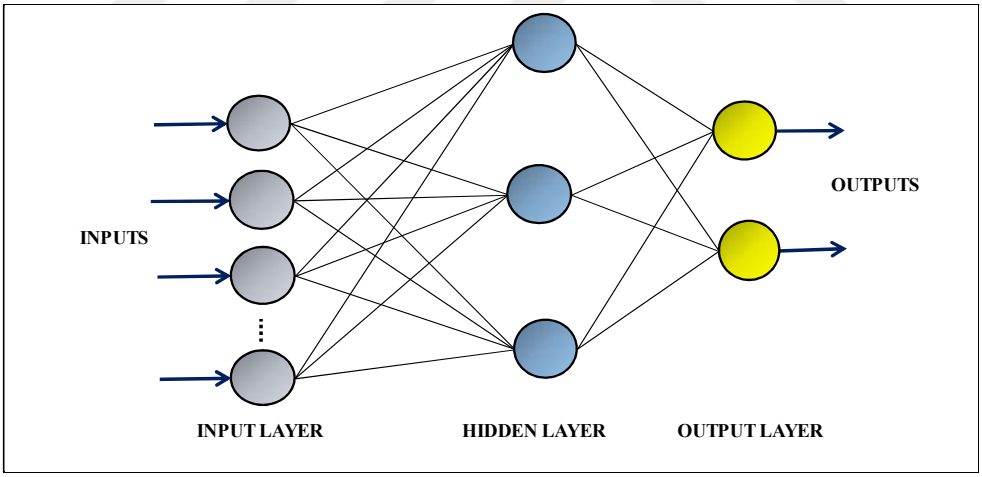
Neurons in this layer receive information from the outside world and transfer it to the hidden layers. In other words, inputs are information coming to neurons. In some networks, there is no information processing at the input layer. They just pass the input values to the next layer. For this reason, some researchers do not add this layer to the number of layers of networks.

6.4.2. Hidden (Intermediate) Layer

Information from the input layer is processed and sent to the output layer. The processing of this information is carried out in hidden layers. In a network, the input and output layers consist of a single layer, while the hidden layer can consist of more than one layer. The hidden layers contain a large number of neurons, and these neurons are fully connected with other neurons. The selection of the number of neurons in the hidden layer is very important in terms of defining the size of the network and knowing its performance. In addition, increasing or decreasing the number of neurons and layers in the hidden layer affects whether the network is simple or complex.

6.4.3. Output Layer

The neurons in this layer process the information from the hidden layer and produce the output that the network should produce for the input set (sample) presented from the input layer. The output produced is sent to the outside world.



Source: Öztemel (2006); p.53.

Figure 6.4. General structure of the ANN

The neurons in each of these three layers and the relationships between the layers are shown in Figure 6.4. The round shapes in the figure show neurons. There are parallel elements in each layer, and the lines connecting these elements show the connections of the network. Neurons and their connections form an artificial neural network. These connections' weights are determined during learning.

6.5. Architectural Structures of the Artificial Neural Networks

In general, ANNs are classified according to three main criteria. One of these criteria is the connection structure of ANNs, which is also called the architecture of the network. The artificial neural network's connection structure differs from one another based on the directions of the connections between the neurons or the flow directions of the signals in the network. While some networks are configured as feedforward, some networks contain a feedback network structure.

The second classification criterion is the classification of the network according to its learning patterns. According to this criterion, there are three basic learning styles. These are supervised (teacher) learning, unsupervised (teacher) learning, and supportive learning.

The third classification criterion is the classification made according to the number of layers. According to this classification, networks are divided into single-layer and multi-layer networks.

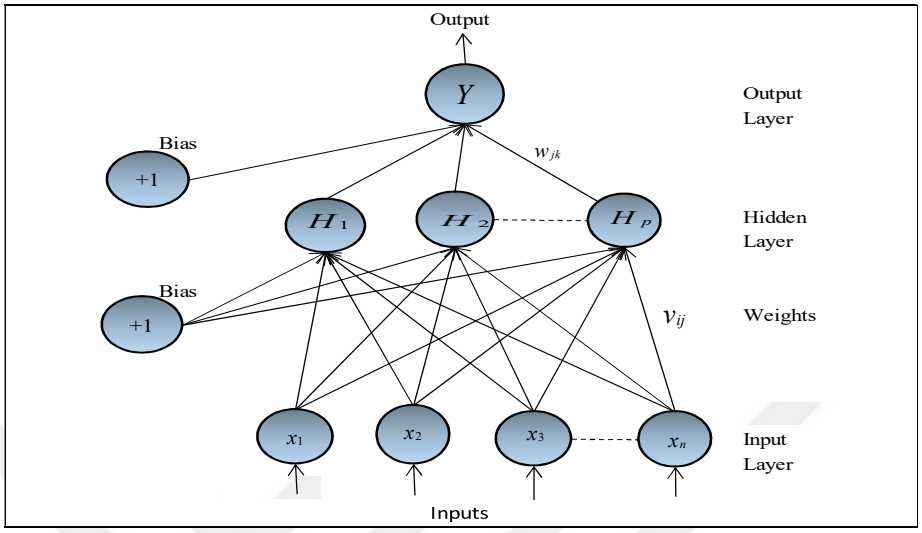
6.5.1. Classification of ANNs According to Connection Structures

In ANNs, some networks are configured as feedforward, while others contain feedback structures. In feedforward neural networks, the connections between neurons do not form a loop, and these networks usually respond quickly to input data. In feedback networks, connections contain loops and can use new data each time the loop is closed. These networks generate the input response slowly due to the loop. Therefore, the training process for such networks is longer. There are also network structures that can be defined as both feedforward and backward propagation (Paul D. McNelis, 2005, p. 21). These two types of networks are discussed in detail below.

Feedforward Networks

Neurons in a feedforward network are usually divided into layers. Neurons in each layer are associated with neurons in the next layer by connection weights. However, the layers do not have any connection among themselves. In feedforward networks, information flow is carried out in one direction from the input layer to the output layer without feedback

(Fathi & Maleki Shoja, 2018a, p. 251). This is also called the activation direction. An example of this type of artificial neural network is a single or multi layer perceptron. Such networks are trained using the supervised learning technique.



Source: Kaastra and Boyd (1996), p. 218.

Figure 6.5: Feedforward network structure

The feedforward network structure is given in Figure 6.5. There is no feedback in such networks. Here, input neurons are shown as $\{X_1, X_2, \ldots, X_n\}$, and hidden neurons are shown as $\{H_1, H_2, \ldots, H_p\}$, Y represents output neuron, and V_{ij} shows, the weight of the connection from the input neuron i to the hidden neuron j and W_{jk}, shows the weight of the connection from the hidden neuron j to the output neuron k. The units shown as +1 are the threshold values.

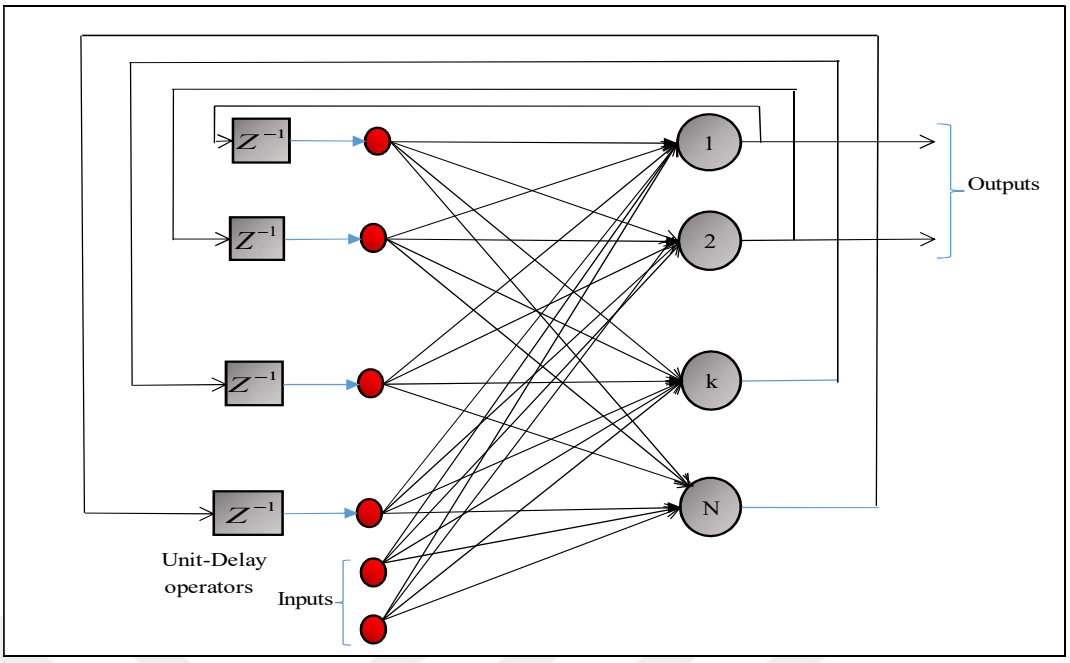
The input layer transmits the information it receives from the external environment to the neurons in the hidden layer without making any changes. This information is processed in the input layer and output layer to determine the network output. With this structure, networks perform a nonlinear static function. The most well-known backpropagation learning algorithm is used effectively in the training of this type of artificial neural network, and sometimes these networks are also called backward propagation networks. The feedforward backpropagation architecture was developed in the 1970s. Generally, a backpropagation network has an input layer, an output layer, and at least one hidden layer. There is no restriction on the number of hidden layers, but usually, one or two hidden layers are used (Haykin, 1999, p. 22).

In backward propagation networks, the number of layers and the selection of the number of neurons in each layer are important decisions in terms of affecting the performance of the network. There are no clear selection criteria for what these numbers will be. However, there are general rules that must be followed. These rules can be expressed as:

1. As the relationship between input data and output becomes more complex, the number of neurons in the hidden layers also increases.
2. If the subject matter can be divided into several stages, it may be necessary to increase the number of layers.
3. The amount of training data used in the network, constitutes the upper limit criterion for the number of processes in the hidden layers.

Recurrent Networks

A recurrent neural network is a network structure in which the outputs in the output layer and hidden layers are recurrent to the input units or previous intermediate layers. In this network, there are turns or recurrent connections between neurons and therefore they are said to have dynamic memory. In recurrent networks, the output of any cell can be sent directly to the input layer and used as input again. Recurrent can be between cells in a layer as well as between neurons between layers. ANNs with recurrent exhibit nonlinear dynamic behavior. As examples of these networks, Hopfield, Elman, and Jordan nets can be given. Figure 6.6 shows the structure of a recurrent artificial neural network structure (Haykin, 1999, p. 24).



Source: Haykin (1999), p.2; Öztemel (2006), p. 171.

Figure 6.6: Recurrent network structure

Recurrent ANNs can be developed with models with richer dynamics than feedforward ANNs. However, feedforward networks are more applied than recurrent networks in academic and practical fields. This is because recurrent networks are difficult to implement in practice. In particular, the fact that recurrent networks can be created with many different structures may prevent specialization in a particular model structure and make the training phase difficult due to the inconsistency of the training algorithms (Kaastra & Boyd, 1996, p. 217). In addition, the training of recurrent networks takes a long time. In particular, as the number of data points in the training set increases, the training time gets longer. For this reason, a feedforward network structure is preferred for solving problems related to multivariate and long time series.

6.5.2. Artificial Neural Network Classification Based on Learning Styles

The main feature of ANNs is their ability to learn. The basic philosophy of learning is to learn the relationships between the inputs and outputs of an event by using actual examples of that event, and then determine the outputs of new examples that will occur later according to these relationships. Here, it is accepted that the relationship between the inputs and outputs of the examples related to an event contains information that will represent the overall event (Öztemel, 2006, pp. 23–24).

The process of determining the weight values of the connections between the neurons of an artificial neural network is called training the network. The learning process is started by randomly assigning these weight values at the beginning. As the determined samples are introduced to the network, the weight values of the network change, and the network continues this process until it produces the desired outputs. Training of the network is complete when the network is able to produce the desired outputs and make generalizations about the events it represents. This is called network learning. Learning styles that enable finding the best weight set during the solution of the problem are divided into three categories: supervised (with the teacher), unsupervised (without a teacher), and supportive learning (Öztemel, 2012, pp. 25–26). In the literature, these learning methods are also referred to as "*machine learning algorithms*", which are frequently used in time series data analysis and are also a sub branch of ANNs.

Supervised Learning

In the form of supervised learning, a teacher or superviser helps the learning system to learn the event. The supervisor illustrates the event that will be taught to the system as an input/output set. In other words, for each sample, both the inputs and the outputs that need to be generated in return for those inputs are sent to the system. The task of the system is to map the inputs to the outputs determined by the consultant. In this way, the relationships between the inputs and outputs of the event are learned.

In this learning, the difference between the outputs produced by the network and the target outputs is considered as an error, and this error is tried to be minimized. This process is tried continuously until acceptable accuracy is reached. In these trials, the weights of the connections are changed to give the most appropriate output. If the intended output cannot be produced against the given input, the connection weights should be changed to minimize the error in the output value of the network. For this reason, supervised learning requires a mentor. The consultant evaluates the performance of the control and guides the learning process to gradually improve performance. Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) performance criteria are used in supervised learning algorithms. As examples of supervised learning algorithms, Widrow-Hoff's Delta Rule (Delta Rule) and Rumelhart and McClelland's Generalized Delta Rule (backpropagation algorithm) can be given. The operation of a supervised learning algorithm is shown below.

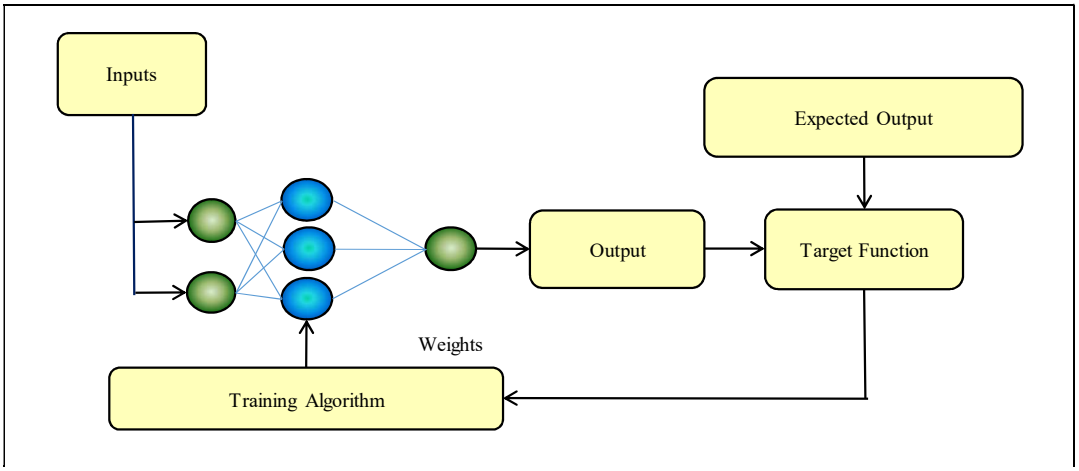


Figure 6.7. Supervised learning algorithm

Unsupervised Learning

In unsupervised learning, there is no teacher helping the system learn. Only input values are shown to the system. The system is expected to learn the relationships between the parameters in the examples by itself. This is an algorithm mostly used for classification problems. Only after the learning of the system is finished, the labeling that shows what the outputs mean be done by the user. Examples of this type of learning are Hebb, Hopfield, and Kohonen learning algorithms and ART networks. The forecasting of such networks is not evaluated by how well the network predicts a particular readily observed target variable (Anderson & McNeill, 1992, p. 41).

Reinforcement Learning

In this type of learning, a teacher assists the network. However, instead of showing the system the output set that should be produced for each input set, the supervisor expects the system to produce the output corresponding to the inputs shown to him and generates a signal indicating whether the produced output is true or false. The system continues the learning process by taking this signal from the supervisor into account. As an example of networks where this learning method is used; Linear Vector Quantization Model (LVQ) can be given. It is used in many fields such as game theory, statistics, information theory, simulation-based optimization, and genetic algorithms.

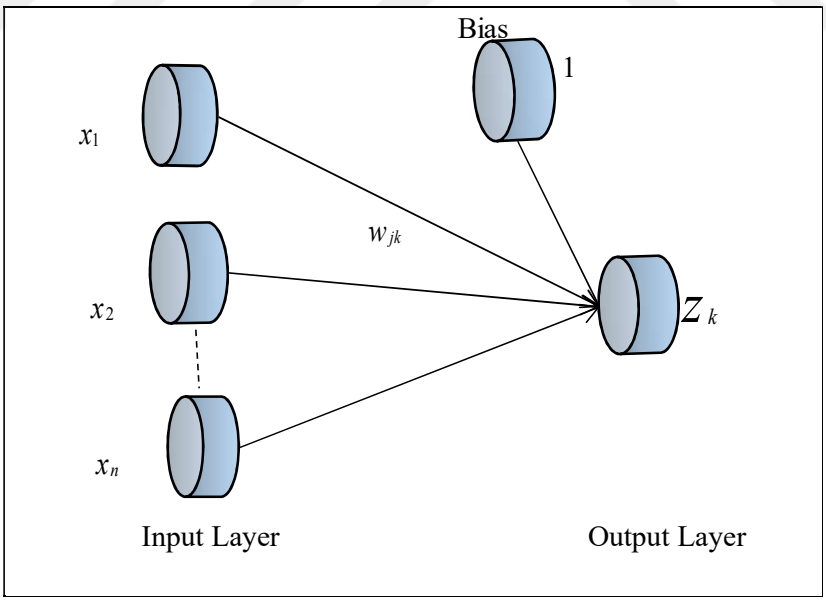
6.5.3. Artificial Neural Network Classification Based on the Number of Layers

According to the number of layers, ANNs are divided into two types: single-layer neural networks and multilayer neural networks.

Single Layer Neural Networks

Single-layer neural networks consist of only input and output layers. Every network has one or more inputs and outputs. Output units are connected to all input units. Every link has a weight. In these networks, there is also a threshold value (Φ) that prevents the values of the neurons and thus the output of the network from being zero. The input of the threshold value is always +1. Because there is no hidden layer to provide curvature, this type of mesh is mostly used for linear problems.

In single-layer networks, each network has one or more inputs (X_i ; i=1,2,…,n) and only one output (Z_k; k=1), and each link has a weight shown with (W_{ik}; i=1, 2 ,…, n; k=1). The artificial neural network model in Figure 6.8 has a single-layer structure.



Source: Öztemel (2006) p. 59.

Figure 6.8: Single layer networks

The output of the network is found by summing the weighted input values with the threshold value (Bias) (θ). This input value is passed through a transfer function to calculate the output of the network. This process is formulated as below (Öztemel, 2006, pp. 59–60):

$$Ç = f\left(\sum_{i=1}^{m} w_i x_i + \theta\right)$$

Figure 6.9. Output function

In single-layer perceptrons, the output function is linear. In other words, the examples shown to the network are shared between two classes, and the line that separates the two classes is tring to be found. For this reason, the threshold value function is used. The most important single-layer ANNs are the Simple Perceptron Model (SPM), the Adaptive Linear Element Model (ADALINE), and the Multiple Adaptive Linear Element Model (MADALINE).

The Simple Perceptron Model was first developed by Rosenblatt in 1958 to classify successive processes in a certain order. It is a single-layered, trainable, and single-output model that emerged as a result of studies carried out to model brain functions. It is based on the principle that a nerve cell takes more than one input and produces an output. The output of the network is a logical value consisting of one or zero. The threshold function is used to calculate the value of the output (Öztemel, 2006, p. 61).

It is the first artificial neural network to be trainable. In the simple perceptron model, the output value is derived when the net value, which is the sum of the products of the inputs and their corresponding weight values, exceeds the threshold value. Learning about the artificial neural network is possible by changing these weights. The most important feature of the simple perceptron model is that it has the property of converging the correct weights from the input variables entered into the network in the event of a solution to the problem. The simple perceptron model is the basis for multilayer networks that will be developed later and will be revolutionary for artificial neural network models (Öztemel, 2006, p. 37).

Multilayer Neural Networks

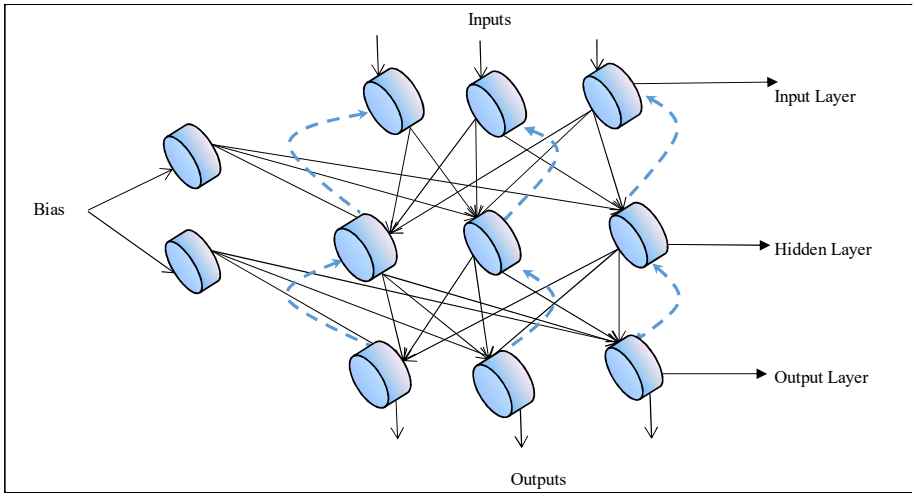
A multilayer perceptron or neural network is a typical network consisting of several layers formed by many neurons that are interconnected. In multilayer perceptrons, there is

an input layer, an output layer, and one or more hidden layers between these two layers. This feature is the main feature that distinguishes multi-layer perceptrons from single-layer perceptrons. Multilayer perceptrons are used in solving complex problems, especially in predictions. Because this network has the ability to automatically transform into a non-linear structure with a series of operations performed in the hidden layer of its structure (G. Zhang et al., 1998, pp. 37–38).

The inadequacy of single-layer perceptrons in solving linear problems has led to the idea that ANNs do not work. However, multilayer perceptrons developed as a result of studies to solve the XOR problem have been an important step in the historical development of ANNs. Research has resulted in the emergence of MLP and its learning rule, the back propagation algorithm. The Multilayer Perceptron Model (MLP) is also called the "Error Propagation Model" or "Back Propagation Model".

Obtaining successful results in solving nonlinear problems with multilayer perceptrons and the back propagation algorithm has led to an increase in its use in many fields, including business science. Multilayer perceptrons are a very important solution tool, especially for problems that require classification, recognition, and generalization. Its main purpose is to minimize the "error" between the expected output of the network and the output it produces (Öztemel, 2006, p. 76). Multilayer ANNs can be illustrated by the Linear Vector Quantization Model (LVQ) and the Multilayer Perceptron Model.

The multi-layer artificial neural network structure is given in Figure 6.10. ANNs have an MLP structure. The first layer of MLP is the input layer. The last layer is the output layer. Between the input layer and the output layer, there is at least one hidden layer with hidden neurons. Each neuron in the hidden layer has a nonlinear transfer function, and the results obtained through these functions provide input to the next neurons (Smith & Gupta, 2000, p. 1025). The two most commonly used transfer functions in MLP are the sigmoid function and the hyperbolic tangent function.



Source: Öztemel (2006), p. 76.

Figure 6.10: The multi-layer artificial neural network structure

The Figure 6.10 illustrates a simple backpropagation network structure that includes an input layer, a hidden layer, and an output layer. Circles arranged in layers represent processing elements, or neurons. There are three neurons in the input layer, and three variables are introduced as input to the network. There are three neurons in the hidden layer and three neurons in the output layer. Therefore, three variables are outputs from the network. The values transmitted from the input layer to the hidden layer and the values transmitted from the hidden layer to the output layer are weighted with a weight set. In the network structure, the thick arrows represent the information flow during recall. Recall is the process of presenting and outputting new input data to a trained network. Therefore, back propagation is not used during the recall process. Back propagation is used only in the training process, so the information flow in the training process is indicated by all the arrows in the figure. There are no connections between neurons in the same layer. The links only go forward (Eğrioğlu et al., 2019, p. 77).

They work according to a multi-layered, perceptive, supervised (teacher) learning strategy. Both inputs and outputs to be produced in response to those inputs are shown to these networks during training. The learning rule is a generalization of the Delta Learning Rule based on the least squares method (Öztemel, 2006, p. 77).

6.6. Learning Algorithms in Artificial Neural Networks

Before an MLP can be used for foresight, it must first be trained. In MLPs, connection weights are determined that allow the expected output to be calculated during the training phase (Öztemel, 2006, p. 82). In a linear regression, the computation is easily done by finding the sum of the squares of the error. However, ANNs include non-linear optimization by nature, which causes the training process to be more difficult and complex than linear regression. Although there are many training algorithms developed in the literature, the most widely used among these algorithms is the back propagation algorithm developed by Werbos (1974) and Rumelhart (1986) et al (G. P. Zhang, 2004a, p. 5)

The backpropagation algorithm has been one of the most important developments in the history of ANNs. This algorithm can be applied to multilayer perceptrons consisting of neurons with a continuous, differentiable activation function. The backpropagation algorithm minimizes total error by adjusting weights with a gradient descent approach or gradual decrease (G. P. Zhang, 2004b, p. 5).

The learning rule of the back propagation network is a generalization of the Delta learning rule based on the least squares method. For this reason, the learning rule is also called the "Generalized Delta Rule" (Öztemel, 2012, p. 77).

The back propagation algorithm allows one to find the weights that will produce the most suitable solution for the given training set. The backward propagation algorithm consists of two steps, forward computation and backward computation (Öztemel, 2012, pp. 78–80; Smith & Gupta, 2000, p. 6):

The stage of forward computation is the stage of computing the output of the network. In the backward calculation phase, it is the rearrangement of the weights in the model based on the errors in the outputs. This algorithm allows for finding the most suitable weight values for the training set presented to the network. Weights are regulated based on gradient descent.

The implementation process of the back propagation algorithm of a single hidden layer and feed-forward MLP is as follows:

Step 1: After samples are collected and the topological structure of the network and learning parameters are determined, the initial values of the weights are randomly assigned.

Step 2: Steps 3 through 9 are repeated until the necessary condition for the training to stop is met.

Step 3: For each set of training data, steps 4 through 8 are repeated.

Forward Propagation Calculation

Step 4: It starts the input layer of the network by displaying the inputs (G_1, G_2,…) from a sample selected from the training set. Inputs are transmitted to the middleware without any processing. In other words, if the output of k'th neuron in the input layer is $Ç^{i}_{k}$; then neuron's output is formed as $Ç^{i}_{k} = G_k$.

Step 5: Each input to the neurons in the hidden layer is multiplied by the weights {W_1, W_2,…, W_n} and the net input is calculated as follows:

$$NET^{a}_{j} = \sum_{k=1}^{n} w_{kj} Ç^{i}_{k}$$

Figure 6.11. Net input function

Here W_{kj}: represents the weight value that connects the k'th input layer element to the j'th hidden layer element.

The sigmoid function is the activation function, and the important thing here is to use a differentiable function and when used, the output is as given below. β_j in this equation is the weight of the threshold value of the j'th element in the hidden layer.

$$Ç_j^a = \frac{1}{1+e^{-\left(NET_j^a + \beta_j^a\right)}}$$

Figure 6.12. Output function

When these calculations are made by all neurons and finally the output values of the output layer are found, the forward calculation phase ends. The activation function in the hidden layer and the activation function in the output layer do not have to be the same. One of them can be a sigmoid function and the other a tangent function or some other function (Öztemel, 2006, p. 50).

Backward Propagation Calculation

Step 6: For the input presented to the network, the output produced by the network and the expected outputs (E_1, E_2,....) are compared, and the difference, or error value, is distributed to the weight values of the network, and the error is tried to be reduced in the next iteration. Error for the m'th neuron in the output layer (e_m);

$$e_m = E_m - O_m$$

Figure 6.13. Error for the m'th neuron in the output layer

This value is the error obtained for a processing element (the neuron). To find the Total Error (TE) for the output layer, all errors must be summed. The main purpose of training the MLP network is to minimize this error. TE is formulated as:

$$TE = \frac{1}{2}\sum_m e_m^2$$

Figure 6.14. Total error

The error is distributed to the neurons to minimize the total error. This is done by changing the neuron weights. This is achieved by changing the weights between the hidden layer and the output layer, and between the hidden layer and the hidden layer input layer.

Step 7: If we call Δw^a the amount of change in the connection weights connecting the j'th neuron of the intermediate layer to the m'th neuron in the output layer, the amount of change in the weight in the t'th iteration is calculated as follows:

$$\Delta_{w}{}^{a}_{jm}(t) = \lambda\delta_m O_j^a \alpha \Delta_{w}{}^{a}_{jm}(t-1)$$

Figure 6.15. Amount of changes (Δw^a)

In this equetion (λ) is the learning coefficient, and, (α) is the momentum coefficient. The learning coefficient is effective in improving the performance of the network. A small learning coefficient extends the training time, while a large learning coefficient shortens it. However, too large a learning coefficient prevents convergence. The momentum coefficient, on the other hand, is effective in preventing the mesh from getting stuck at a local minimum point during knitting. It ensures that the weight change value is added to the next change at a certain rate, and the total error tends more towards zero. δ_m in the equation is the error of the m'th output unit.

$$\delta_m = f'(NET).\ e_m$$

Figure 6.16. m'th output unit error

Here, $f'(NET)$ is the derivative of the activation function. When the sigmoid activation function is used, this equation is;

$$\delta_m = O_m\ (1 - O_m).\ e_m$$

Figure 6.17. Output unit error

Step 8: After calculating the amount of change, the new values of the weights in the t'th iteration will be calculated as follows:

$$w_{jm}^{a}(t) = w_{jm}^{a}(t-1) + \Delta w_{jm}^{a}(t)$$

Figure 6.18. Value of weights

In addition, the weights of the threshold value are changed. If the threshold value of the neurons in the output layer is represented by their weights θ^c the amount of change is given as the threshold output is constant and 1;

$$\Delta\theta_m^c(t) = \lambda\delta_m + \alpha\Delta\theta_m^c(t-1)$$

Figure 6.19. The amount of changes

The new weight value of the threshold value in the t'th iteration is as follows:

$$\theta_m^c(t) = \theta_m^c(t-1) + \Delta\theta_m^c(t)$$

Figure 6.20. The new weight value

In the second case, the errors of all elements in the output layer must be taken into account when changing the weights between the input layer and the middle layer or between the two middle layers. If this weight change, for example, is represented by Δwi between the input layer and the middle layer, the amount of change is;

$$\Delta_{w}{}_{kj}^{i}(t) = \lambda\delta_j^a O_k^i + \alpha\Delta_{w}{}_{kj}^{i}(t-1)$$

Figure 6.21. The amount of weight change

The δ^a factor is in the form of;

$$\delta_j^a = f'(NET)\sum_m \delta_m w_{jm}^a$$

Figure 6.22. δ^a factor value

and when the sigmoid function is used δ^a factor value is;

$$\delta_j^a = O_j^a(1 - O_j^a)\sum_m \delta_m w_{jm}^a$$

Figure 6.23. δ^a factor value

The new values of the weights are;

$$w_{kj}^{i}(t) = w_{kj}^{i}(t-1) + \Delta w_{kj}^{i}(t)$$

Figure 6.24. The new values of the weights

If the interlayer threshold weights are denoted by θ^{a} the amount of change;

$$\Delta\theta_{j}^{a}(t) = \lambda\delta_{j}^{a} + \alpha\Delta\theta_{j}^{a}(t-1)$$

Figure 6.25. The amount of change

The new values of the weights for the t'th iterastion are provided by the equation below,

$$\theta_{j}^{a}(t) = \theta_{j}^{a}(t-1) + \Delta\theta_{j}^{a}(t)$$

Figure 6.26. The new values of the weights

As a result, the forward and backward calculation steps are performed for one iteration. All of the weights are rearranged, and the process of editing the weights is continued until the termination criterion is met.

Step 9: If the error has reached the predetermined fault tolerance or the termination criterion has been met, the training is stopped. Otherwise, the process is repeated starting from Step 4.

The above-mentioned steps continue until the learning of the multilayer perceptron is complete, that is, until the errors between the realized outputs and the expected outputs are reduced to an acceptable level. For the network to learn, there must be a stopping criterion. This is generally taken as the error falling below a certain level.

Input data is presented to the network and propagates throughout the network until it reaches the output layer. This forward process produces the calculated output.

An error value is calculated for the network by comparing the calculated output with the target output. To train the network, back propagation algorithms from supervised

learning algorithms are used. This algorithm propagates backwards through the network, starting with the weights between the output layer neurons and the last hidden layer neurons.

When backpropagation is finished, the forward process starts again, and this cycle continues until the error between the calculated output and the target output is minimized.

When the back propagation algorithm is chosen as the learning algorithm, two parameters become important. The first of these is the learning rate (λ), and the other is the momentum coefficient (α). These two parameters will be briefly described below.

Learning Rate (λ)

The learning rate has a significant impact on the performance of the network and determines the amount of change in the weight values of the links. This coefficient usually has a value between 0 and 1. In the case where the learning rate is small, the training process takes a long time, while it takes a shorter time with the growth of this value. While very small values of the learning rate cause the learning process to slow down unacceptably, very high values of the learning rate cause instability. In a case where the learning rate is greater than 1, the network oscillates between local minima, and no convergence event occurs. In addition, the fact that this ratio is too low may not provide the opportunity to find the minimum (Öztemel, 2012, p. 99; G. Zhang et al., 1998, p. 48).

Momentum Coefficient (α)

This coefficient, which affects the learning performance of the network, is defined as the addition of a certain rate of the change in the previous one iteration to the new change amount (Öztemel, 2012, p. 99). This coefficient has been specifically proposed to enable networks stuck in local solutions to find better results with a jump and is a factor that helps the network recover faster. Taking into account the momentum coefficient, which has values between 0 and 1, produces a reduction in the number of steps and the total network error (G. Zhang et al., 1998, p. 48). The momentum coefficient helps to speed up the educational process with a large learning rate, while minimizing the tendency to oscillate.

Determining the optimal learning rate and momentum coefficient is largely experimental and heuristic. In addition, these parameters vary greatly depending on the problem area of interest.

6.7. Learning Rules

In the literature on neural networks, there are many learning rules used in learning systems. The majority of these learning rules are based on the oldest and best-known Hebb Learning Rule. Some important learning rules in use are given below.

6.7.1. The Hebbian Learning Rule

It is the oldest and most famous learning rule, developed on a biological basis by Canadian psychologist Donald Hebb in 1949. According to this rule, if a neuron receives input from another neuron and both neurons are highly active (i.e., they have the same mathematical sign), the weight of the connection between neurons must be increased. In other words, if a cell is active, it tries to make the cell it is connected to active, and if it is passive, it tries to make it passive (Hebb, n.d., pp. 60–78).

6.7.2. The Hopfield Learning Rule

This rule, which is similar to the Hebb rule, determines how much the connections between the artificial neural network elements should be strengthened or weakened. The main difference is that it also determines the size of the change to be made in the connection weight. Accordingly, if both the input and the desired output are active or both are inactive, the connection weight is increased by the learning coefficient; otherwise, it is decreased by the learning coefficient. Strengthening or weakening of the weights is carried out with the help of the learning rate. The learning coefficient is a fixed and positive value assigned by the user, which generally takes a value between 0 and 1 (Öztemel, 2012, p. 26).

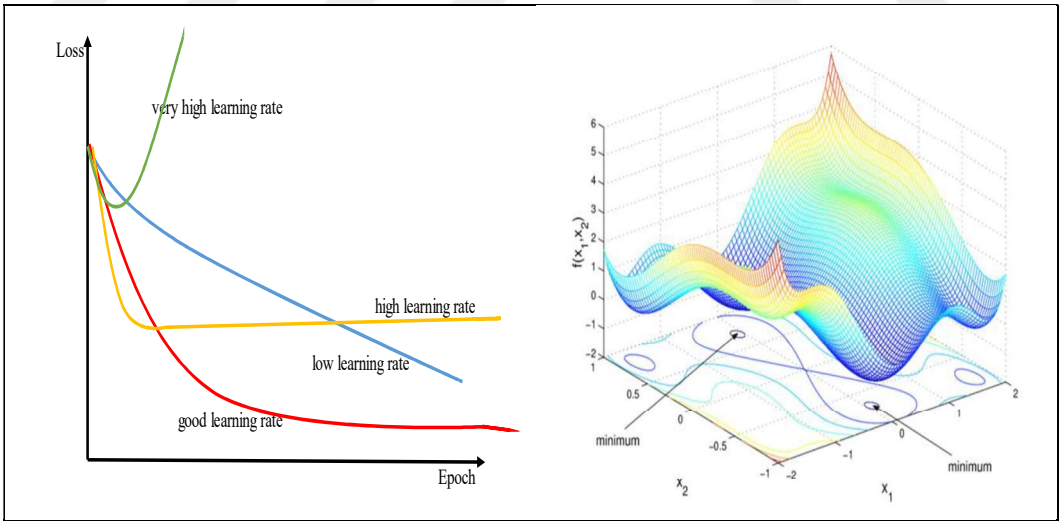
6.7.3. The Delta Rule

One of the most commonly used learning rules is the Delta Rule. This rule is an improved version of the Hebb rule. It was developed by Widrow and Hoff (1960). It is based

on switching input connections in order to reduce the difference (delta) between the desired output and the actual output of the processing unit. For this reason, this algorithm is also known as the least-squares rule. The error is reduced by simultaneously propagating back from one layer to the previous layers. The process of debugging the network continues from the output layer to the input layer (Anderson & McNeill, 1992, p. 30).

6.7.4. The Gradient Descent Learning Rule

This rule is similar to the Delta rule. The adjustment of the weights can be performed in a manner proportional to the first derivative of the error (gradient) between the desired output and the actual output for a processing unit. The aim is to get rid of the local minimum and catch the general minimum by reducing the error function (Kamruzzaman et al., 2006b, p. 131). It is one of the most used optimization algorithms in ML. The size or smallness of the learning rate is an important factor in achieving the optimal result. A low learning rate increases the processing time required to achieve the optimal result, whereas a high one does not reduce error because the steps are too large and deviate from the optimal. Therefore, the learning rate is chosen to be large enough to reach the global optimum by trial and error. The



Source: https://medium.com/deep-learning-turkiye/gradient-descent-nedir-3ec6afcb9900

Figure 6.27. Graident descent

6.7.5. The Kohonen Learning Rule

This rule was developed by Teuvo Kohonen (1982), inspired by learning in biological systems. Neurons are thought to compete to learn to adjust their weight. According to this rule, the elements of the mesh compete with each other to change their weight. The cell producing the largest output becomes the winning cell (neuron), and the connection weights are changed. This means that, this cell becomes stronger than the cells next to it. The cell has the capacity to warn and ban its neighbors. The Kohonen rule does not require a target output. Therefore, it is an unsupervised learning method. This rule, also known as self-organizing, is used especially in studies on distributions. However, due to the fact that its theoretical infrastructure is not fully developed, it has not become widespread in practice yet (Anderson & McNeill, 1992, p. 30; Kohonen, 2001, pp. 71–72).

6.8. Application Areas and Properties of Artificial Neural Networks

ANNs, which have gained a wide application area in real-life problems, have been used in the solution of problems in many different areas, especially those that are difficult and complex to solve, and generally very successful results have been achieved. It is used in many industries today. There is no limit to the application area. However, it is mainly used in some areas, such as forecasting, modeling, and classification. ANNs, which emerged in the 1950s, became sufficiently powerful for general-purpose use only in the 1980s. Since it is the method that best describes the trend or structure in the data, it is very suitable for forecasting and forecasting operations. The following examples can be given for the common uses of ANNs in real life (Öztemel, 2012, p. 203-206).

- Industrial applications (quality control, inspection, system modeling, etc.),
- Financial applications (economic and financial forecasting, credit rating, macroeconomic forecasts, risk analysis, etc.),
- Military and defense applications (target recognition and tracking systems, etc.),
- Health applications (medical image processing, urology applications, etc.),
- Applications in other fields (oil and gas exploration, handwriting recognition, etc.).

ANNs need special environments in which they can work due to their densely connected and complicated processing structures. Therefore, ANNs run on computers with special software prepared for this purpose. Today, special hardware is being developed to run increasingly dense and complex neural networks and to process information faster.

Properties of the Artificial Neural Networks

ANNs can produce results according to the information they have without any intervention by performing learning, association, classification, generalization, prediction, feature determination, and optimization processes. ANNs have the ability to make correct decisions for subsequent inputs by arranging themselves with the information given during the learning process. ANNs, which are used in many fields today, are used effectively in many fields due to the following general features. ANNs have the ability to make correct decisions for subsequent inputs by arranging themselves with the information given during the learning process. ANNs, are used effectively in many fields today due to the following general features (Öztemel, 2012, p. 31).

ANNs perform ML: The main function of networks is to enable computers to learn. By learning from the events, it tries to make similar decisions in the face of similar events.

The working style of the programs is not similar to the known programming methods: There are different information processing methods than the ones in which traditional programming and AI methods are applied.

Storage of information: In ANNs, information is measured by the values of the connections in the network and stored in the connections.

Nonlinearity: The neuron, which is the basic processing element of ANNs, is not linear. Therefore, ANNs, which are formed by the combination of neurons, are also non-linear. However, the fact that the activation function used is not a linear function turns ANNs into a non-linear structure. Therefore, ANNs can be used to solve nonlinear problems in real life.

Learning: Neural networks learn and solve problems using examples from real events. In order to learn about events, examples related to those events should be determined. Artificial neural network gains the ability to make generalizations about the event by using examples. In order for ANNs to show the desired behavior, they must be programmed in accordance with their purpose. In other words, correct connections must be made between neurons, and connections must have appropriate weights. Due to the complex nature of neural networks, connections and weights cannot be preset or designed. As a result, ANN should learn the problem by using training examples from the problem it is interested in, so that it exhibits the desired behavior.

In order to operate ANNs safely, they must first be trained and their performance tested: Training the network means showing the existing samples to the network one by one and determining the relationships between the events in the example by running its own mechanisms. The samples are divided into two sets, the training set and the test set. Each network is first trained with the training set, and when the network starts to give correct answers to all examples, the training work is complete.

Ability to generate information about unseen examples: The network can generate information about examples it has not seen by making generalizations from the examples shown to it.

Usability in detection events: Networks are mostly used to process information for detection.

Pattern association and classification: The purpose of most networks is to associate patterns that are given as examples to themselves or others. Another purpose is classification.

Ability to self-organize and learn: It is possible for the network to adapt to new situations shown by examples and to learn about new events continuously.

Ability to work with incomplete information: Networks can work with incomplete information after they are trained and can produce results even though there is missing information in new samples. This does not degrade their performance. The network itself can learn what information is important during training.

Having fault tolerance: Networks' ability to work with incomplete information ensures that they are fault-tolerant. Since the ANN consists of many neurons connected in various ways, it has a parallel distributed structure, and the information of the network is distributed over all connections in the network. Therefore, the inactivation of some connections or even some neurons in a trained ANN does not significantly affect the network's ability to produce accurate information. Therefore, their ability to fix the error is higher than other methods.

Ability to process uncertain and incomplete information: After learning the events, networks can make decisions by establishing relationships with the events they learned under uncertainty.

Gradual degradation: The fact that networks are fault-tolerant causes their degradation to be gradual. A network breaks down slowly and gently over time. This is due to incomplete information or disruption of neurons. Networks do not degrade as soon as a problem arises, they degrade gradually.

Distributed memory: In neural networks, information is spread over the network. The values of the connections of neurons with each other indicate the information of the network. A single link makes no sense. The network as a whole characterizes the entire event it learns. Therefore, the information is distributed over the network. This results in a messy memory.

Ability to work with only numerical information: Information represented by symbolic expressions must be converted to numerical representation in order to be interpreted and to produce solutions.

Hardware and Speed: ANNs can be implemented with large-scale integrated circuit (VLSI) technology due to their parallel structure. This feature is expected in real-time applications as it increases the fast information processing capability of the neural network.

6.9. The Design of Artificial Neural Networks

In this part of the study, information will be given about how to design ANNs and which criteria to consider during the creation of network architecture. The number of layers in ANNs, how many neurons these layers will consist of, how the neurons should be positioned relative to each other, and the flow directions of signals between neurons determine the structure of the networks.

There are differences in performance and characteristics between the structures of ANNs. The structures of ANNs are very important, especially since they determine the modeling ability of the network. At the design stage of the artificial neural network, the most suitable one is selected among these network structures. Although there are some suggested techniques for determining the optimal network architecture, these methods are quite complex and difficult to implement. Also, none of these methods guarantees an optimal solution for real estimation problems. In short, there are no precise and clearly applied methods for determining these parameters. Therefore, neural network design is more of an art than a science (G. Zhang et al., 1998, p. 42).

6.10. Training and Testing of Artificial Neural Networks

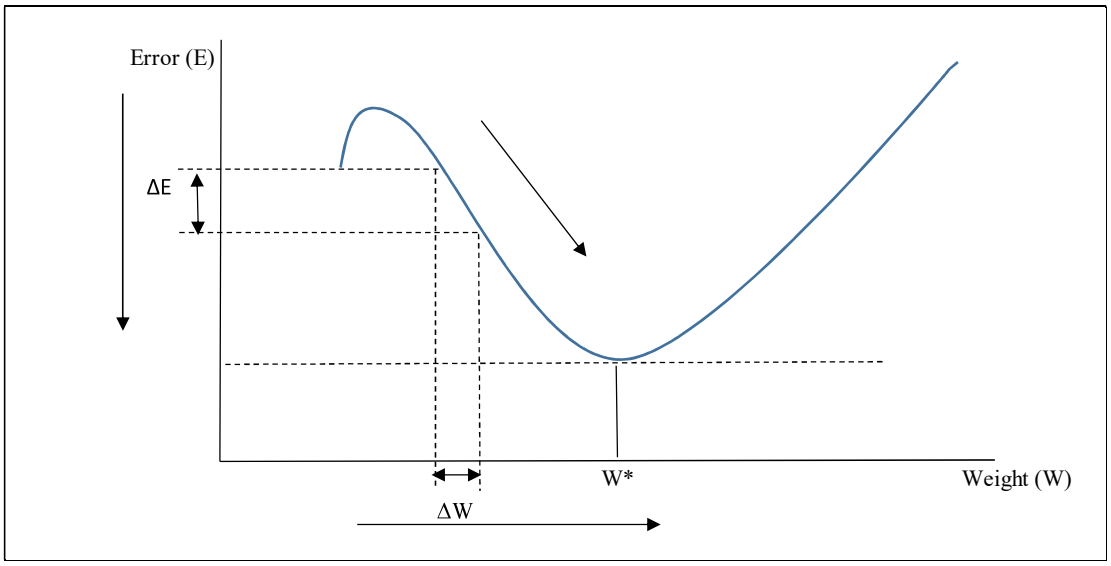
The process of determining the weight values of the connections of neurons in ANNs is called “training the network”. Initially, these values are randomly assigned. Networks change these weight values as examples are shown to them (Öztemel, 2012, p. 55).

In the setup phase of the neural network estimator, the sample dataset is divided into two datasets for training and testing of the network. There is no general rule for separating data. However, the data type, amount of data, and the characteristics of the problem are important factors in separating the data set. Inaccuracies in the selection of the training and test dataset will affect the performance of the network (G. Zhang et al., 1998, p. 50). The selected datasets should be at a level that can describe the sample space. There are few suggestions in the literature for the determination of training and test sets. Many researchers use 90% of the data as a training dataset, while the remaining 10% is used as a test dataset. Likewise, the rates of 80%, 20%, or 70%, 30% are frequently used in the literature to divide

the data into periods (G. Zhang et al., 1998, p. 50). Training samples from the data separated according to this rule are used to develop the artificial neural network model, while test samples are used to evaluate the predictive ability of the developed model.

In the learning process, weights are randomly assigned at the beginning, and the weight values change as the samples are shown to the network according to the chosen learning approach. The goal here is to find the weight values that will produce the correct outputs for the examples shown to the network. The network, which has reached the correct weight values, has become able to make generalizations about the event represented by the examples and has completed the network learning.

The philosophy of training multi-layer networks, which is the model to be considered in this study, is not different from other ANNs. For this reason, the problems encountered in the training and education of multi-layer networks will be discussed here. The important point in the training process of the network is to find the weight values that will give the least error in the problem space. Considering a simple problem to explain error minimization, suppose that the event (problem space) that the network is asked to learn has an error space as shown in Figure 6.28.

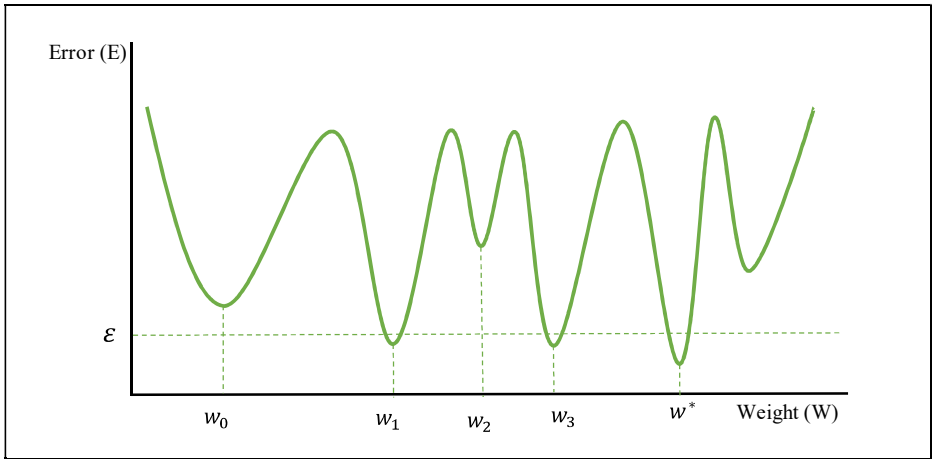


Source: Öztemel (2006); p..82.

Figure 6.28. Representation of learning in error space

W in the figure shows the weight vector with the least error. Multilayer networks are desired to reach the W* value. This weight value represents the point at which the error for the problem is the smallest. For this reason, by making a change by ΔW in each iteration, it is ensured that an error of up to ΔE is reduced at the error level.

It should be noted that the error level of the problem will not always be simple and two-dimensional, as seen in Figure 6.28. Complex problems in real life have many minimum and maximum points at the error level. The minimum points will only give the W* minimum weight vector, or global minimum point, at one level. As can be seen in Figure 6.28 although the weight vector W*gives the least error level, it may not always be possible to catch this point. This solution is the best the network can have. It tries to reach this solution during network training. However, instead of stopping the training process of multilayer networks at the global minimum, which is the best solution point, there is also the possibility of stopping at the minimum point (W_1, W_3) with weight values that will give a higher error rate (Zurada, 1992, pp. 206–207). In this case, it is said that multilayer networks fall into the local minimum trap during training (Kamruzzaman et al., 2006b, p. 131). The local minimum point may be at a level that is considerably higher than the global minimum level, or it may be close to this level (Öztemel, 2012, p. 83).



Source: Zurada (1992), p.207.

Figure 6.29. Multidimensional error space

As seen in Figure 6.29, although W* is the weight vector that gives the least error for solving the problem, it is often not possible to catch this error value in practice. It tries to capture the W* solution of multilayer networks during training. Sometimes it can be stuck

on a different solution (the local minimum), and it is not possible to improve performance. For this reason, users accept a margin of error up to a certain level by specifying a tolerance value (ε) in the performances of the networks. The problem is considered learned when it falls below the tolerance value. Since the errors of the W_0 and W_2 solutions in Figure 6.28 are above the acceptable error level, these solutions are unacceptable. These are called local solutions. Although solutions W_1, and W_3 are not the best, their error level is lower than the acceptable level. Although these are local solutions, they are acceptable solutions. As can be seen, more than one solution can be produced for a problem. Therefore, it cannot be said that multi-layer networks always produce the best solution. It would be more accurate to say that they produced an acceptable solution. Even if the solution produced is the best solution, it is difficult to know. In most cases, it is not possible to know (Öztemel, 2012, pp. 82–83).

Figure 6.28 also shows the following possible reasons why the best results could not be found (Öztemel, 2012, p. 83):

- The training set presented to the network while the problem is being trained may not represent 100% of the problem space.
- The correct parameters may not have been selected when the MLP was created.
- The network's weights may not have been determined exactly as desired at the start.
- Then, the next day, the next.

For these and similar reasons, MLP may not reduce the error below a certain value during training. For example, W_1 finds the weights and cannot lower the error any further. But W_1 is actually a native solution, not the best solution. It can be seen as the local best solution since the error has decreased to an acceptable level for the W_1 vector. It may also be possible that the global solution does not exist. It all depends on the design of the network, the nature of the samples, and the training process.

Another problem encountered in the training of multi-layer networks is that the network memorizes rather than learns. In this case, although the trained multilayer networks produce correct answers for all samples in the training set, they cannot produce correct answers for the samples in the test set. In other words, the network gives the impression that it has achieved the global minimum by memorizing the sample input. The network didn't actually learn; it just memorized the sample input set. Such a network will produce results

that are far from the performance of the training during the testing process. Users encountering this type of problem should investigate the cause of this situation (Kaastra & Boyd, 1996, pp. 229–231).

After the training of the network is completed, the attempts to measure whether it learns (performance) are called network testing. For testing, the samples that the network does not see during learning, which is the test set, are used. The weight values of the network are not changed during testing. The inputs in the test set are given to the model of the artificial neural network, and the output value of the artificial neural network is compared with the desired output value. The purpose of this process is to see whether the ANN model can make adequate generalizations. If the desired success is achieved in the training and testing stages, the artificial neural network model can be used.

6.11. Selection of Network Structure

While creating an ANN model, the stages of determining the structure and structure properties of the network, determining the properties of the functions in the neuron, and determining the parameters by selecting the learning algorithm should be done carefully and depending on the characteristics of the problem to be applied. Because these stages are very important for the success of the results to be achieved. When the learning algorithm to be used in the network is selected, the structure required by this algorithm will also be automatically selected. Table 6.3 shows the network types that are successful based on their intended use.

Table 6.3. Network types and areas of success

Purpose of Useage	Network Type	Use of the network
Prediction	Multilayer perceptron (MLP)	Estimating an output value from the inputs of the network
Classification	Vector Quantization Models (LVQ) Adaptive Resonance Theory Models (ART) Probabilistic Neural Networks (PNN) Counterpropagation	Determining which class the inputs belong to
Data Shaping	Hopfield Networks Boltzman Machine Bidirectional Associative Memory (BAM)	Finding incorrect information in the entries and completing the missing information

Source: Öztemel (2012), p.207.

6.11.1. Determining the Number of Input Neurons and Output Neurons

In ANNs, it is easy to determine the number of input neurons in estimation problems based on cause-effect relationships. In such studies, the number of input neurons is determined by the number of variables in the input vector. In this case, the number of input neurons is equal to the number of variables. In time series estimation problems, the number of input neurons is related to the number of delays. However, there is no definite way to determine how this situation will be resolved.

It is easier to determine the number of output neurons than to determine the number of neurons in other layers. In problems involving the future prediction of a time series, the number of output neurons is equal to the length of the prediction period. While the number of output neurons is equal to 1 in the single-period forecasting, in multi-period forecasting, forecasting can be done in two ways. A one-period forecast is an iterative forecast. The estimated period value is used as input for the next period. In this case, a single output neuron is sufficient. The direct method is used in multi-period forecasting when more than one period is estimated at the same time. In this case, the number of output neurons is equal to the number of periods to be estimated (G. Zhang et al., 1998, pp. 44–46).

6.11.2. Determination of the Hidden Layer and the Number of Hidden Neurons

The hidden layer and the hidden neurons in this layer are of great importance in the success of ANNs (G. Zhang et al., 1998, p. 42).

The number of hidden layers varies depending on the problem, the amount of data, and the design. Usually one or two hidden layers (intermediate layers) are sufficient. Using more hidden layers significantly reduces the speed of the network. It can also cause the network to memorize rather than learn. Preferably, a three-layer structure consisting of input-hidden-output layers can be used. If the result is not satisfactory, 2 or 3 intermediate layers can be tried later. Some studies have shown that there is no need for structures with more than two hidden layers to solve most prediction problems (G. Zhang et al., 1998, p. 44). In addition, applications have shown that more than four layers in total adversely affect the success of the network (Kaastra & Boyd, 1996, p. 225).

Increasing the number of hidden layers will increase the number of connections between all processing elements in the network, thus increasing the risk of memorization, leading to increased computation time for the network and poor prediction results. For this reason, when determining the number of hidden layers, the most appropriate number should be found by trial and error, taking the number of data into account.

Similarly, there is no magic formula for the number of hidden neurons. In this regard, the task falls to the designer's ability. It is generally preferred to work with a small number of hidden neurons. Because they have higher generalization abilities. However, some rules have been developed. These rules are not absolute, they are just a generally applied method. For a 3-layer network with n input cells and m output cells, the number of neurons in the hidden layer can be $\sqrt{nxm}$. The number of neurons in the hidden layer can vary between 1.5 and 2 times the above formula, depending on the type of problem and the number of data. This is called the geometric pyramid rule. Baily and Thompson suggest that the number of hidden cells for a 3-layer network is 75% of the number of cells in the input layer (Kaastra & Boyd, 1996, p. 225).

Although there is no absolute rule for determining the number of neurons in the hidden layer, the exact number depends on the structure of the network, the amount of data, the type of problem, and the experience of the designer. It is determined by trial and error.

6.11.3. Data Standardization

The most important factor that provides nonlinearity in ANNs is the standardization of the data. The data can be standardizated to recover from extreme values so that the available data can be better modeled. In addition, the method to be chosen for normalizing the data also affects the performance of the network. Data standardization ensures that the negative effects of the cumulative totals of the data used during the processing process are prevented. The activation functions used in the hidden and output layers, which determine the data normalization range, play the role of compressing the output of a neuron to a range of [0,1] or [-1,1]. Data standardization is done before the training process begins. The formulas that are frequently used in data standardization approaches are as follows (G. Zhang et al., 1998, p. 49):

Linear transformation in the range [0, +1]: $x_n = (x_0 - x_{min})/(x_{max} - x_{min})$

Linear transformation in the range [a,b]: $x_n=(b-a)(x_0-x_{min})/(x_{max}-x_{min})+a$

Statistical normalization: $x_n = (x_0 - \bar{x})/s$

Simple normalization: $x_n = x_0 /x_{max}$

X_n and X_0 in the formulas represent the normalized and original data, and X_{min}, X_{max}, $\bar{x}$ and s represent the minimum, maximum, mean, and standard deviation values along the row or column, respectively.

Studies have been conducted on how important data normalization is in learning ANNs, and they concluded that normalization is beneficial, but this benefit decreases as the sample size increases.

It is usually independent of normalizing the inputs and normalizing the targeted output values. In time series and forecasting problems, the normalization of inputs and targeted outputs is done together. The choice of the normalization range depends on the transfer function of the neurons in the output layer. If the sigmoid function is used in the output layer, the normalization range is chosen as [0,1], and if the hyperbolic tangent function is used, the normalization range is chosen as [-1,1]. However, if the identity function, which is a linear function, is used as the transfer function of the neurons in the output layer, then the selection of the normalization range is made according to the transfer function of the processing elements in the hidden layer. The targeted output values should be consistent with the normalization range of the observed network outputs as a result of normalization. Interpretation of the results from the network can be done after conversion of the outputs to the original range. The accuracy of the values produced by the network should be calculated based on the original data set. The performance measure should also be calculated after converting the outputs to the original range (G. Zhang et al., 1998, p. 50).

6.12. Determination of Neural Network Performance

Although there are many performance measures for a neural network estimator, such as modeling time or training time, the best and most important performance metric is the

accuracy of the estimation. The accuracy criterion is defined as the difference between the true value and the predicted value. This difference is called the prediction error (G. Zhang et al., 1998, p. 51).

Determining the performance of a model means measuring the accuracy of predictive modeling. Good forecasting leads to good decisions. When the performance of an artificial neural network is mentioned, it is understood that the learning ability is measured. In other words, it is measured whether the artificial neural network model learns the data well or not. The most commonly used predictive accuracy metrics are as follows (Gujarati, 2011, p. 77; G. Zhang et al., 1998, p. 51):

Table 6.4. Performance criteria

MAE	$\sum\|e_t\|/N$
SSE	$\sum(e_t)^2$
MSE	$\sum(e_t)^2/N$
RMSE	$\sqrt{MSE}$
MAPE	$(1/N)\sum\|e_t/y_t\|(100)$

Here;

e_t, is the estimation error,

y_t, is the observation value of the period t,

N represents the number of error terms.

Among these predictive accuracy measures, MSE is the most widely used. An important feature of this measure is that the prediction error can be decomposed into variance sums. This feature shows that the MSE criterion depends only on the second moment of the combined distribution of realization and prediction. Thus, it is a measure that provides useful information. It should be noted, however, that it does not provide complete information on the true distribution (G. Zhang et al., 1998, p. 52).

6.12.1. Determining the Stopping Criteria

The stopping criterion can be determined in two ways: if the error falls below a certain value or below an acceptable error determined by the researcher, and when the network completes the specified number of iterations, the training of the network is stopped.

6.12.2. Selection of Learning Algorithm

The two most important factors that determine the success of an artificial neural network application are the structure of the network and the learning algorithm. The network structure plays a decisive role in the selection of the learning algorithm. There are many learning algorithms to be used in the development of ANNs. It is known that some algorithms are more suitable for some applications. The most used algorithm is the back propagation algorithm (G. Zhang et al., 1998, pp. 47–48).

6.13. Advantages and Disadvantages of Artificial Neural Networks Applications

The advantages of ANNs are due to their non-linear structure and unique training process. Generally, strengths of ANNs compared to other models include the ability to model nonlinear structure, the ability to make generalizations, adaptability and flexibility, information storage, error tolerance, and the absence of prerequisites and assumptions in statistical or other modeling techniques (G. P. Zhang, 2003, p. 160).

The relationships between real-life events and the factors behind these events are non-linear, and it is very difficult to model these relationships. However, by modeling models more easily, ANNs can solve models that are difficult and complex to model mathematically (Öztemel, 2012, p. 207). Thanks to the transfer function they use, ANNs can produce models for nonlinear problems and perform effective predictive modeling. For this reason, ANNs are preferred as an effective predictive tool over traditional prediction methods.

Through their learning ability, ANNs can generalize about situations that they have not encountered before using known examples. By using the data in the learning phase, the

error between the input and the output is minimized, and an ANN model that gives the least error between the input and output variables can be established. Due to their adaptability and flexibility, artificial neural netw ANN can be retrained by changing the connection weights in the case of new information or changes in the environment. This is one of the most basic features that distinguish ANNs from traditional statistical methods (G. Zhang et al., 1998, p. 36).

In neural networks, information is measured by the value of the network's connections and stored in the connections. As with other programs, the data is not embedded in a database or program. In this way, ANNs have the power to store information compared to other models (Öztemel, 2012, p. 31). The fact that artificial neural networks, consisting of densely interconnected neurons, can work with incomplete information, ensures that they are tolerant to errors. There are limitations when modeling with traditional econometric modeling techniques such as Box-Jenkins, ARIMA, moving average, and the linear model. In addition, there is a loss of information since stagnation is required. However, ANN modeling does not have these drawbacks (Abraham & Malik, 1973, pp. 1–17).

In order to successfully solve the problem using artificial neural networks, the problem must be well modeled. ANNs do not need any prior knowledge other than examples for modeling. ANNs applications are both practical and cheaper in terms of cost. Just specifying examples and a simple program may be sufficient to solve the problem. The fact that ANNs can work in parallel facilitates their real-time use.

6.14. Disadvantages of Artificial Neural Network Applications

Apart from the advantages provided by the use of artificial neural networks, there are also some disadvantages that should be considered (Öztemel, 2012, p. 34). Hardware dependent operation of ANNs is an important problem. Most of today's machines have serial processors. The reason networks exist is to be able to work with parallel processors. ANNs need very fast parallel processors, especially because they do parallel processing. Performing parallel operations on serial machines causes a waste of time. A different ANN structure should be developed for each problem. This is done by trial and error. Neural networks also do not guarantee that the solution found is the best solution.

There is no certain set of rules in the creation of ANNs, in model selection, in determining the topology of the network, in determining the number of layers, or in determining parameters such as the learning rate and the number of neurons that should be included in each layer. It is determined entirely based on the experience of the researcher. This is also an important disadvantage. There is also no method to determine when the training of the network will end. Reducing the error of the network on the samples below a certain value is considered sufficient to complete the training. However, it cannot be said that an optimal trainer is provided. In this case, it may take a long time for the training to take place.

The ANN does not give any information on how it transforms any input vector into an output vector. From an engineering point of view, neural networks can be seen as a "*black box*". The black box receives the information from the outside and gives the outputs it produces to the outside. What happens inside is unknown (Cheng et al., 2022a, p. 108-218). In other words, the general rule or condition of the connection between input and output is unknown to the network, and the network has no ability to explain it. While this situation decreases trust in the obtained network, successful applications increase interest in ANNs. In cases where it is difficult to find examples or examples that accurately represent the problem cannot be found, it is not possible to produce healthy solutions to problems.

Despite all these disadvantages, it is possible to produce solutions for many problems and create successful applications with ANN. In order for ANNs to be able to produce solutions to problems by getting rid of these disadvantages, it is necessary to meticulously create networks. Having sufficient knowledge about both the problems to be solved and ANNs can provide successful results. It is necessary to know that it is possible to create a network without ignoring this fact in searching for solutions to problems with the ANN method, but it is not an easy process (Öztemel, 2012, p. 35).

7. DEEP LEARNING

DL is a branch of ML that uses ANNs and algorithms to process data in a way that is inspired by the structure of human nerve cells. DL generates results by learning from the processed data. Just as people learn from their life experiences, DL algorithms improve their learning abilities by learning a little more each time and making changes thanks to their many layers. Thus, they are able to produce results that were not possible before. The concept of "deep" here also refers to the number of layers in the artificial neural network. Unlike classical neural networks, deep neural networks consist of more layers. Since it is a sub-branch of ML, the terms "machine learning" and "deep learning" are often used as if they were the same term. But the two systems are different and have different capabilities.

The foundations of ANNs and DL, which are created by influencing the human nerve cell structure, date back to the 1940s. Bengio et al. have examined the general historical development of DL in three waves. The term "cybernetics" was used to describe the period between 1940 and 1960, "connectionism" to describe the period between 1980 and 1990, and "deep learning" to describe the period after 2006 (Bengio et al., 2015, pp. 11–12).

The concept of ANNs first started with the mathematical model developed by McCulloch-Pitts in 1943 based on the study of the human brain. Later, with the concept of ML by Turing in 1949, the concept of the perceptron by Rosenblatt in 1958, and the discovery of the Adaptive Linear Element-ADALINE by Widrow and Hoff in 1960, the process called "cybernetics" was experienced. However, since studies in this field declined until the 1980s, the period known as the "first winter of artificial intelligence" was experienced. In the second wave, called connectionism, gained momentum in studies after the 1980s gained momentum, and the concepts of DL networks, backpropagation algorithms and convolutional neural networks were introduced by LeCun (1998) in 1998. In the 1990s, AI studies continued increasingly, and with the Hebbian learning rule concept in 1949 and Schmidhuber's very DL concept in 1993, great steps were taken in the depth and complexity of neural networks. This process paved the way for the discovery of the Long Short Term Memory-LSTM network (Schmidhuber, 2015, pp. 93–94). During this period, deep networks were thought to be very difficult to train because the existing hardware was not structured to allow too many experiments and also the cost was too high. In 2006, the third

wave, which started with the concept of Goffrey deep belief networks, gradually increased its speed with the concept of Goodfellow Contested Producer Networks in 2014 and continued its development (Bengio et al., 2015, pp. 11–17; Schmidhuber, 2015, pp. 90–98).

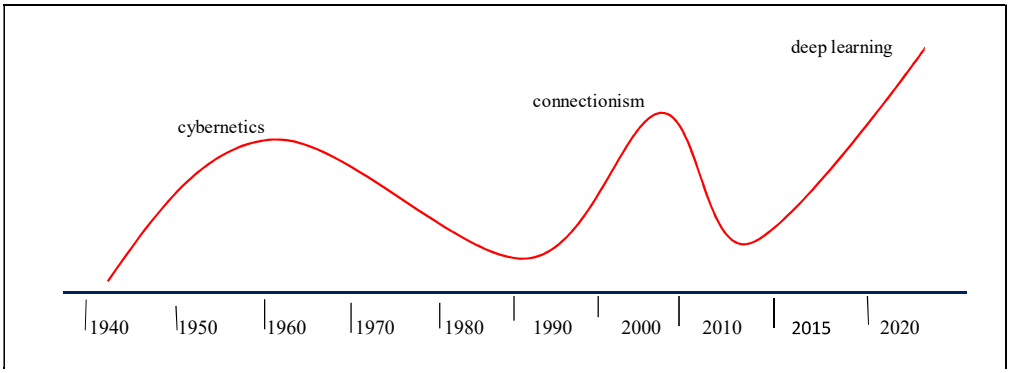


Figure 7.1. The historical process of the deep learning

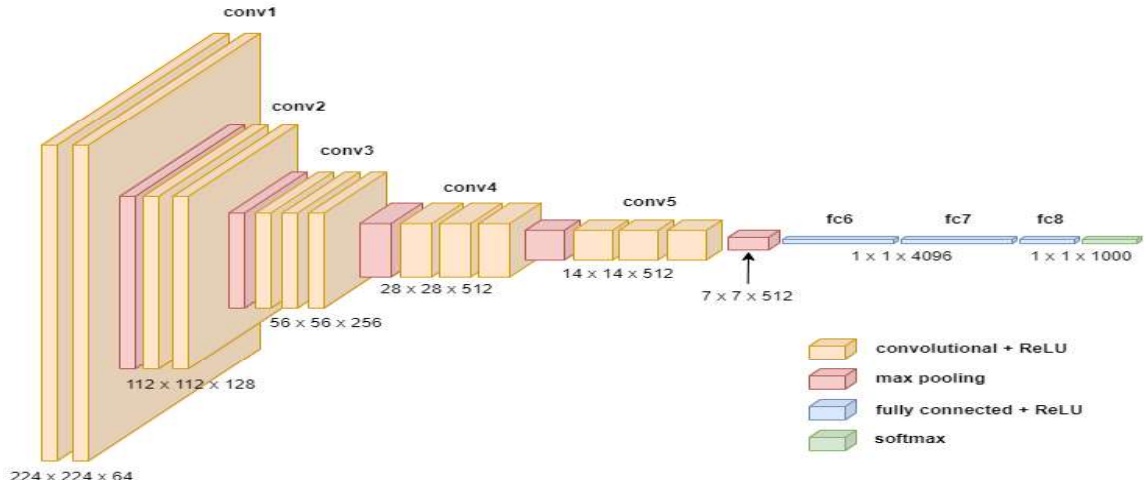
In general, this historical process of DL mentioned above is summarized by the graph in Figure 7.1. According to this figure, there have been three waves of development of deep learning: These are, "the cybernetic process" from the 1940s to the 1960s, "the connectionism process" from the 1980s and the 1990s, and finally, the process that continued rapidly after 2006 under the name "deep learning" (Bengio et al., 2015, p. 12).

Today, DL is used by the world's top technology companies, such as Google, Microsoft, Facebook, IBM, Apple, Baidu, Adobe, Netflix, NVIDIA, and NEC.

7.1. Fundamentals of Deep Learning Architecture

The ultimate point of ML is DL algorithms. Contrary to traditional learning methods, DL has a hierarchical structure. The general logic of the algorithm, in which supervised and unsupervised learning methods are used, is based on the convolution-based artificial neural network model (Metlek & Çetiner, 2021, pp. 6–7). Convolutional neural networks come together in different combinations and form the basis of modern DL architectures. Layers contain many non-linear processing units, and each layer uses the output of the previous layer as input. In DL, low-level features are extracted from layers without removing certain features beforehand, and a hierarchical representation is formed by deriving high-level features from low-level features. Deep neural networks are capable of processing voice,

image, speech, text, and video data. While doing so, the layers in the classical DL architecture created are listed below (Metlek & Çetiner, 2021, pp. 7–18).



Source: https://towardsdatascience.com/how-to-easily-draw-neural-network-architecture-diagrams-a6b6138ed875

Figure 7.2. Elements of the deep neural network architecture

An example of deep neural network structure is shown in Figure 7.2 above. A deep neural network structure consists of four main layers: the convolutional layer, max pooling layer, fully connected layer, and softmax layer.

7.1.1. Input Layer

The first layer in convolutional neural networks is the input layer. It is the layer where data is standardized and presented to the network. Data presented to the network is first served from this layer. If the data has been subjected to any sizing process before, this layer can be given to the network. If there is no standard for the size of the data or if there is a size that must be followed at the entrance of the network, the sizing process is also performed in this layer.

7.1.2. Convolution Layer

A series of filters (kernels) are used to circulate the input data. The content, size, and number of the filters change depending on the application to be made. The size of the filters to be used in the application should be smaller than the size of the data. It is the layer where

new attribute matrices are generated by circulating filters with dimensions and properties determined by developers on data from the previous layer.

7.1.3. Activation Layer

It is the layer that brings the matrix values obtained from the convolution layer into the range determined depending on the algorithm used. It is often used to convert negative values to positives. The most commonly used activation functions in the literature are; the logistic sigmoid function $f(z) = \sigma(z) = \frac{1}{1+e^{-z}}$, the hyperbolic tangent function $f(z) = tanh(z)$=$2\sigma(2z) - 1$, and the Recitified Linear Unit (ReLU) function $f(z) = max(0, z)$ (Fathi & Maleki Shoja, 2018a, pp. 258–259).

7.1.4. Pooling Layer

In many studies in the literature, the pooling layer is used after the activation layer. Basically, the purpose of this layer is to reduce the preceding data to matrices of smaller sizes. So, the network will work faster. However, downgrading may also cause data loss. This is a predictable process that will normally be welcomed. Many matrices, such as maximum value, minimum value, and mean value matrices, can be used in this layer for data reduction.

7.1.5. Dropout Layer

In general, while DL works with large data sets, when it needs to work with a smaller data set, the artificial neural network algorithm can memorize the data used during training. As a result, the data in the network architecture may need to be forgotten. In other words, it is the layer used to prevent the network from memorizing the training set in order to reduce this risk, since there is a risk of over-learning in cases where the network has few data.

7.1.6. Fully Connected Layer

This layer is the one that takes all the data from the previous layers and transforms it into a one-dimensional array matrix, or vector. Data from this layer goes directly to the

classification layer. It can be used after the pooling or memorization layer as well as after the classification layer.

7.1.7. Classification Layer

In DL networks, as in artificial neural network algorithms, it is the final layer and is the layer that creates the output value of the network by evaluating the data from the previous full link layer. The output of the generated network can be classification information as well as polynomial coefficients. Class information obtained from the classification layer is used by many different classification functions. One of them is the SoftMax classifier, which uses a probabilistic calculation method and can generate values in the range of 0-1 for each class.

7.2. Deep Learning Architectures

Since data science and the ease of accessing data increase the volume of data obtained, it is very difficult and requires a long process to analyze big data with classical programming methods. For this reason, many new AI algorithms have been developed that can easily work with big data. Basically, the common feature of these algorithms is that they can generate rules by creating a link between the data and the result. This rule concept expresses the weight coefficients of the network and the structure of the network. DL has found application in many areas where computer technologies are used. It is seen that it gives very successful results. This result has led to an increase in the diversity of studies in this field due to the interest in the deep artificial neural network method (Metlek & Çetiner, 2021, p. 20).

In DL networks, which generally use convolutional neural network architecture, the term “deep” refers to the number of layers in the network. The more layers, the deeper the network. While classical neural networks or convolutional neural networks have only two or three layers, deep neural networks can have hundreds of sublayers (Metlek & Çetiner, 2021, p. 21). However, deep neural networks are designed like typical feed-forward networks. Due to the diversity of the problems and areas studied, different structures or models of deep neural networks have been developed for the solution. Some of the main ones are described below.

7.2.1. Multi Layer Perceptron (MLP)

The first single-layer perceptron model was developed by Rosenblatt in 1958. This perceptron model, called a “perceptron”, is based on the principle of a neuron taking multiple inputs and producing an output using a threshold function. A perceptron is the simplest form of neural network and consists of only one artificial neuron. The output of the network is a logical value of 1 or 0 (Öztemel, 2012, p. 61). Later, in 1959, Widrow and Heff developed a single-layer network called Adaptive Linear Element (ADALINE), which learns using the least squares method. In 1990, Widrow and Lehr (1990) took this model one step further and developed Multiple Adaptive Linear Elements (MADALINE) networks, which are formed by combining more than one ADALINE unit (Öztemel, 2012, pp. 68–74). These early models of ANNs are used to solve problems with a linear structure but it was seen that they could not learn nonlinear relationships. For this reason, multilayer perceptrons have been developed.

Multi layer perceptrons (MLP), also called feedforward neural networks or deep feedforward neural networks, are models consisting of three or more layers or multiple perceptrons (McNelis, 2005, p. 251). MPL is also known as the core architecture of deep neural networks or DL. The learning phase of multi-layered DL networks is done with the backpropagation algorithm, which is a supervised learning technique, as in classical ANNs. Various optimization approaches, such as Stochastic Gradient Descent (SGD), Limited Memory BFGS (L-BFGS), and Adaptive Moment Estimation (Adam), are applied during the training process. The output of an MLP network is determined using transfer functions such as ReLU, Tanh, Sigmoid, and Softmax. The mean square error or logloss functions are used to calculate the loss rate (Sarker, 2021, pp. 6-7).

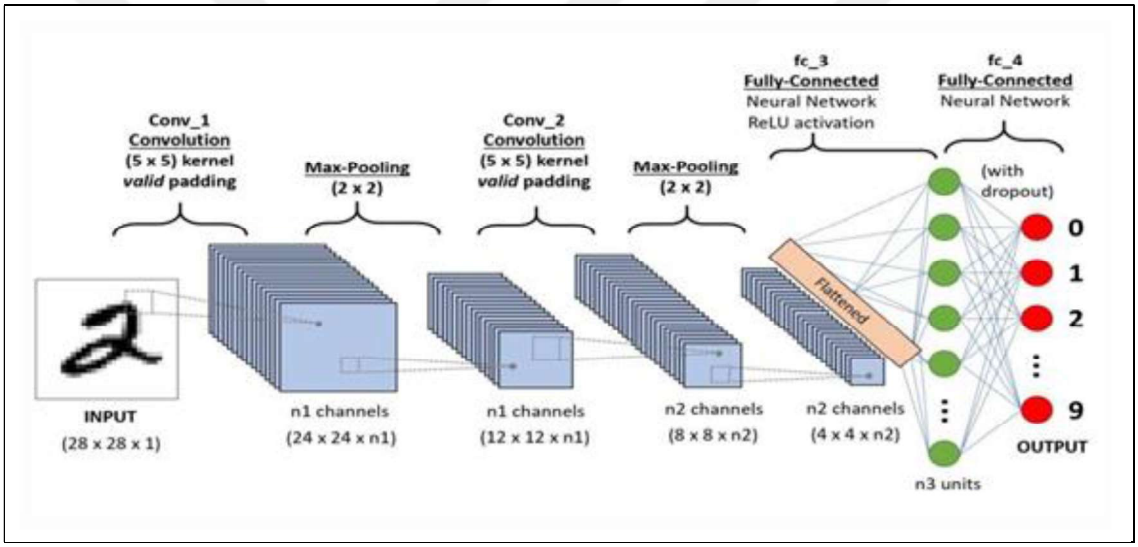
Due to the large number of layers, it is called the deep network and is used for solving more complex problems. As seen in the examples given in the literature, deep neural networks have been used successfully in time series analysis (Jing et al., 2021).

7.2.2. Convolutional Neural Network (CNN or ConvNet)

Convolutional neural networks are a kind of feedforward-supervised deep neural network that learns directly from input. Convolutional neural networks develop a classical

neural network design like multilayer networks. It creates the optimal parameters at each layer to achieve a meaningful output, thus reducing the model's complexity. It uses the dropout layer to reduce the memorization problems that can occur in a classical network. In addition, it reduces the matrix of the outputs by using the pooling layer and transforms the matrix into a one-dimensional vector by using the full link layer. Because of these features, they are generally used in the fields of visual recognition, medical imaging, and natural language processing. Convolutional network architecture is used to build applications such as AlexNet, Xception, ResNet, and GoogleLeNet (Sarker, 2021, p. 7).

A general convolutional neural network structure is shown in Figure 7.3. The convolutional neural network in the figure consists of six layers.



Source: https://teknoloji.org/derin-ogrenme-nedir-yapay-sinir-aglari-ne-ise-yarar/#Multilayer_Perceptron_Cok_Katmanli_Algilayici

Figure 7.3. Convolutional network architecture

7.2.3. Recurrent Neural Network (RNN)

Recurrent neural networks are another popular network structure that uses sequential or time-series data and feeds the output from the previous step as input to the next step. (Sarker, 2021, p. 7). Like feedforward and convolutional networks, iterative networks also learn from training inputs, but the difference in this network structure is that their internal memory is used to process information from previous inputs. The purpose of the model is to create a training model using sequential information. These networks are widely used for

complex tasks such as time series, handwriting learning, and language recognition due to their learning capabilities.

Unlike a typical deep neural network, which assumes that the inputs and outputs are independent of each other, the outputs in recurrent neural networks depend on previous elements in the sequence. However, the backward dependence of a standard recurrent network structure makes it difficult to train this network and causes the problem of disappearing gradients. Due to the problem of lost information in this network, whose main purpose is to measure long-term dependencies, it is difficult to store and learn information with this network structure. For this purpose, methods such as recurrent neural network architecture based on Long Short-Term Memory (LSTM), Bidirectional RNN/LSTM, Gated Recurrent Units (GRUs) have been developed (Sarker, 2021, p. 7).

7.2.4. Long Short-Term Memory (LSTM)

It is an iterative neural network architecture based on feedback connections, unlike feedforward neural networks. This architecture is a popular RNN architecture developed by Hochreiter et al. (1997) that uses special units to deal with the vanishing gradient problem. A memory cell in LSTM architecture can store data for a long time. It has a unit called a memory cell. RNN and LSTM network structures are the same but the only difference comes from the difference in the content of a node. In an LSTM network, information flow in and out of the cell can be done through three ports. These are the input gate, output gate, and forget gate. The data coming to these gates passes through a certain activation function (sigmoid or tanh), and the incoming data leaves the node as output after certain processes with the input content. Thus, LSTM is considered one of the most successful networks of RNNs, as it works with a memory that remembers the inputs for a long time and solves the training problem (Sarker, 2021, p. 7).

7.2.5. Restricted Boltzmann Machine (RBM)

A Constrained Boltzmann Machine (RBM) is also a generative stochastic neural network that can learn a probability distribution among its inputs. An RBM neural network typically consists of two layers, visible and hidden (Sarker, 2021, p. 10). The layers are not interconnected. Each neural node is a unit of computation that processes the input, and this

node makes stochastic decisions about whether to transmit the input. Certain threshold values are added to the inputs, multiplied by certain weights, and passed through the activation function to form the output. In the restructuring phase, the output is compared model consisting of several sequentially connected stacks of independent unsupervisewith the previous output by reprocessing the process in this way, and it is tried to reach the output value that gives the least difference by making more than one repetition (Chong et al., 2017a, p. 190).

7.2.6. Deep Belief Network (DBN)

A Deep Belief Network (DBN) is a multi-layered generative graphicd networks that use the hidden layer of each network as input for the next layer, such as RBMs. Clustering with unsupervised learning is the last layer, followed by classification with the SOTFMAX layer. The main idea in the DBN structure is to train unsupervised feedforward neural networks with unlabeled data before fine-tuning the network with labeled input. One of the key advantages of DBN, unlike typical surface learning networks, is that it allows for reasoning capabilities and the detection of deep patterns, allowing the deep difference between normal and erroneous data to be captured. Overall, the DBN model can play a key role in a wide variety of high-dimensional data applications due to its powerful feature extraction and classification capabilities and has become one of the hottest topics in the field of neural networks (Sarker, 2021, p. 10).

7.2.7. Deep Auto-Encoder (DAE)

A deep automatic encoder network, consisting of two parts (encoding and decoding), is a feed-forward but unsupervised learning model that sends data from the input layer to the output layer using its own tags during training. Autoencoder is widely used in many unsupervised learning tasks such as size reduction, feature extraction, efficient coding, generative modelling, noise removal, anomaly or outlier detection, etc. There are methods developed based on deep autoencoder architecture, such as Sparse Auto-Encoder (SAE), Noise Reduction Auto-Encoder (DAE), Contracted Auto-Encoder (CAE), and Variational Auto-Encoder (VAE) (Sarker, 2021, p. 9).

8. LITERATURE REVIEW

Stock exchanges are organized markets where long-term investment instruments such as stocks and bonds are bought and sold. In these markets, those who supply funds and those who demand funds come face to face. Thus, investors can take risks in a standard and transparent environment, and funds are transferred from those who supply funds to those who request funds. Since most of the capital needs of developing countries are provided by stock portfolio investments, the development and volume of the securities markets are very important for these countries. After the 1980s, rapid developments were experienced in the capital markets due to the globalization trend that started with the liberalization of capital markets in the USA and other developed countries. Mobility between capital markets accelerated, and financial markets grew. Although the financial markets of America and Europe fell into trouble with the global financial crisis in 2008, the growth in China and India had a positive impact on the world markets (Karan, 2013, pp. 37–38).

With the great economic crises experienced in 1929 and 2008, the world stock market activities all over the world have developed rapidly in an attempt to heal their wounds. Stock exchanges have a significant impact on the global economy because they contribute significantly to the economic development of countries by allowing a significant amount of savings to be transferred to companies as capital. In addition, companies can reach long-term, less risky, and less costly loans through securities exchanges instead of short-term, high-interest, high-risk loans to be taken from the banking system. Undoubtedly, the transformation of these funds arising from these savings into investments is a serious growth driver for countries. The stock exchanges that investors invest most in the world are the New York Stock Exchange (NYSE), an unorganized and over-the-counter stock market; NASDAQ; the Toronto Stock Exchange; the Amsterdam Stock Exchange; the Tokyo Stock Exchange; the Hong Kong Stock Exchange; and the Bombay Stock Exchange, which are thought to be the starting place of the stock market in some sources.

An index is a criterion that is used to measure the change that consists of the movements of one or more variables and thus provides general information about complex events that are difficult to understand based on many more variables by linking them to a single variable (Karan, 2013, p. 60). In other words, an index is a statistical composite

measure of movement in the market or industry. The price indices by which the stock prices of companies are determined in a market or market segment are called stock market indexes (Vui et al., 2013, p. 477). Since stock market indices include price movements in the sector, they are not linear in nature, and their volatility is high. Stock markets are highly volatile due to the effects of both micro and macroeconomic variables of countries, as well as psychological variables such as consumer behavior, expectations, and political risk. Due to these features, they are also important market indicators. Owing to their dynamic, complex, and non-parametric structure, stock market index prediction is difficult. Considering that the estimated size of the world stock market was 36.6 trillion dollars during the global crisis period in 2008, it can be understood why stock markets are so important for national economies and investors. For this reason, stock market indexes and stock price predictions have attracted a lot of attention from investors and researchers recently.

In the finance literature, the number of studies examining the effect of macroeconomic variables on asset prices is increasing. However, most of the studies are aimed at estimating the effects of macroeconomic variables on stock returns. Because stocks are the most risky investment instruments in the capital market, they can respond very quickly to economic developments. For this reason, examples of studies on stock market indexes and stock forecasts are given below.

The basis of the performance appraisal method, which is an indicator of effective forecasts, is based on the study of the market timing of money managers in the USA by Alfred Cowles in 1933. According to this study, Cowles concluded that 45 professional fund management organizations cannot provide a higher return from the stock market (Cowles, 1933, pp. 309–324). In this study, 16 financial advisory companies giving investment advice made approximately 7500 recommendations for common stocks from January 1, 1928, to July 1, 1932, but an annual average of worse than 1.43 percent was observed, and even this result was found to be a coincidence.

The modern portfolio theory, or Capital Asset Pricing Model (CPM), introduced by Markowitz in 1952, which assumes that investors choose the securities they will buy into their portfolios from a universe full of risky assets, is the oldest and most popular method used in determining the returns on securities (Markowitz, 1952). This theory was later developed by Sharpe (1964), Lintner (1965), and Mossin (1966) and used in many studies

(Lintner, 1965; Mossin, 1966; Sharpe, 1964). The fact that it considers the market factor as the only factor affecting asset returns ensures that the model can be easily predicted. In addition, Treynor (Treynor, 1965, pp. 63–75), Sharpe (1966), and Jensen (1968) have developed three different performance valuation models, named after themselves and consisting of a single parameter, which evaluate the performance of mutual funds using SVFM and Modern Portfolio Theory. However, the necessity of using multi-factor models in the analysis due to the ignoring of many factors has led to the emergence of the Arbitrage Pricing Model (AFM) (Roll & Ross, 1980).

Chen, Roll, and Ross (1986) concluded in their study that the difference between long-term interest rates, expected and unexpected inflation rates, industrial production, differences between high and low quality bonds, and the risk premium systematically affect stock returns. They concluded that oil prices do not have any effect on stock returns. In their linear regression analysis using monthly time series, they concluded that stock returns are affected by systematic economic news and changes in macroeconomic variables and are priced in line with these effects. In addition, in their study, they found that the effects of stock market indices such as the NYSE on asset prices are insignificant when compared to macroeconomic variables. Innovations in macroeconomic variables are risks and are priced meaningfully in the stock market (N.-F. Chen et al., 1986, pp. 383–403).

In contrast to Markowitz's equilibrium model and Sharpe's single index models, Fama and French (1996) proved that changes in macroeconomic factors such as the market index, firm size, and book value/market value ratio have an effect on security returns with the model they established using the time series approach of monthly securities returns. In addition, the value obtained by subtracting the risk-free interest rate from the market index, which is the first variable used in the model working with monthly data, will be the residual return of the index. Thus, the assumption that stock returns are based on index returns has also been proven (Fama & French, 1996, pp. 55–84).

Fama (1990) examined the interaction between stock prices and real activities, inflation, and money and found a strong positive correlation between stock returns and real variables such as industrial production, GNP, money supply, interest rate, and lagging values of inflation (Fama, 1990, pp. 1089–1108). However, most of the recent studies have drawn attention to the short-term relationships between macroeconomic variables and stocks.

Schwert (1990) analyzed the relationship between real stock returns and real activity between 1889 and 1988. Fama (1990), in his study with 65 years of data between 1953 and 1987, concluded that there is a high correlation between monthly, quarterly and annual stock returns and future production growth rates. He also compared the two measures of industrial production, the Miron-Romer index and the Babson index, and concluded that the new Miron-Romer index of industrial production is less related to stock price movements than the old Babson measure (Schwert, 1990, pp. 1237–1257).

Mukherjee and Naka (1995) used the Vector Error Correction Model in their study with stock prices and six macroeconomic indicators. They found that there is a long-term equilibrium relationship between exchange rate, money supply, inflation, industrial production, long-term government bond interest rates, demand loan interest rates, and the Tokyo Stock Exchange Index, and that the variables are co-integrated (Mukherjee & Naka, 1995, pp. 223–237).

Wongbangpo and Sharma (2002) investigated stock prices traded in five ASEAN countries (Indonesia, Malaysia, Singapore, the Philippines, and Thailand) and concluded that variables such as gross national product, consumer price index, money supply, interest rate, and exchange rate are affected and that there is a causal relationship between them. In the long run, a positive relationship was observed between the growth in production and stock prices and an inverse relationship between inflation and stock prices. It was also observed that the effects of exchange rates and interest rates had different negative and positive effects according to countries (Wongbangpo & Sharma, 2002, pp. 27–51).

The three-factor model of Fama and French (1998) was applied to ISE firms operating in the 1993–1998 period, and it was concluded that firms with low book value or market value outperformed firms with high book value or market value. It has also been observed that the performance of large firms is higher than that of small firms. These results also showed that the findings of similar studies in developed and emerging markets differ due to differences in national capital dynamics (Gonenc & Karan, 2003, pp. 1–25).

Chen, Leung, and Daouk (2003) conducted a study based on the idea that trading strategies guided by their predictions about the direction of the Taiwan stock market, one of the fastest-growing stock markets in Asian countries, can lead to greater effectiveness and

higher profits. In this study, they concluded that the performance of the Probabilistic Neural Network model they deveped is better than that of the Generalized Moments Method, Kalman Filter and Random Walk model (A.-S. Chen et al., 2003).

Chakravarty (2005) used monthly time series from 1991 to 2005 to investigate the relationship between stock price and some basic macroeconomic variables in India. It has been concluded that there is no causal relationship between stock price, gold price, and exchange rate, but there is a one-way relationship with money supply, and industrial production affects stock prices (Chakravarty, 2005, pp. 1–15).

Lui and Shrestha (2008) investigated the long-run relationship between the Chinese stock market index and macroeconomic factors such as the exchange rate, inflation, money supply, industrial production, and interest rate. They showed that there is a positive relationship between money supply and industrial production and the stock market index and a negative relationship between the exchange rate, inflation, and interest rates and the stock market index (Liu & Shrestha, 2008, pp. 744–755).

Boyer and Zheng (2009) selected seven investor groups in the US capital market during the 53-year period between 1952 and 2004 and found that there was a positive and significant relationship between cash flows and stock returns of these groups, especially mutual funds and foreign mutual fund groups (Boyer & Zheng, 2009, pp. 87–100).

Sayılgan and Süslü (2011) analyzed in their study, using quarterly data on selected variables between 1999 and 2006, 11 developing countries such as Turkey, Hungary, Poland, Russia, Argentina, Brazil, Chile, Mexico, Indonesia, Malaysia, and Jordan. In their analysis, the effects of selected macroeconomic variables on stock returns were analyzed using balanced panel data analysis. As a result, stock returns were affected by the exchange rate, inflation rate, and Standard and Poor's 500 index. It was determined that there is no significant relationship between the interest rate, gross domestic product, money supply, and oil prices (Sayilgan & Süslü, 2011, pp. 73–96).

In their study, Ilahi, Ali, and Jamil (2015) proved by linear regression analysis that the Karachi Stock Market Index is weakly affected by changes in macroeconomic variables such as the inflation rate, exchange rate, and interest rate (Ilahi et al., 2015, pp. 1–11).

In their studies using monthly data from the 2005–2015 period, Coşkun, Kirac, and Muhammed (2016) examined the relationship between the Borsa Istanbul Index and the variables of interest rate, exchange rate, export amount, import amount, industrial production index, and gold price by means of a causality test and an impulse-response function. While a one-way causality relationship was observed from BIST to the industrial production index, exports, and imports, there was also a one-way causal relationship from the exchange rate to BIST (Coşkun et al., 2016, pp. 61–74).

Gürsoy (2019), in his study conducted with the logistic regression method, it was seen that the stock returns of the banks traded in Borsa Istanbul 100 were positively affected by the changes in the Standard & Poors 500 Index, the exchange rate, and the US interest rate (Gürsoy, 2019, pp. 1–25).

In their study, Kocabıyık and Fattah (2020) tested the macroeconomic variables affecting the Borsa Istanbul 100 index and the New York Stock Exchange S&P 500 index for the period 2010–2019 with the Toda-Yamamoto model. They found a bidirectional causal relationship between the money and exchange rate and the BIST100 index, and a causal relationship was seen only in the money supply towards the S&P500 index. In addition, no causal relationship was found between the consumer price index, industrial production index, export-import coverage ratio, interest rate, oil price, and gold price (Kocabiyik & Fattah, 2020, pp. 116–151).

In their study, Kaymaz and Yılmaz (2022) applied the Vector Autoregressive Model to see the effect of macroeconomic variables such as credit risk premium, exchange rate, oil prices, and gold prices on Borsa Istanbul (BIST100) during the COVID-19 pandemic period. In the pre-Covid-19 period, changes in BIST100 were mainly affected positively by gold prices, while credit risk premium, exchange rate, and oil prices negatively affected BIST100. In the Covid-19 period, while the effect of credit risk premium and oil prices on BIST100 increased, gold prices had a negative effect (Kaymaz & Yilmaz, 2022, pp. 206–217).

After the great economic depression in 1929, the science of econometrics, which is based on testing economic theories and modeling them mathematically and statistically, gained importance. Because the experience gained from the crises in the capital markets around the world has shown that the modeling and forecasting of economic and financial

variables is extremely important for the economies of countries. Therefore, the science of econometrics has begun to be used more by policymakers to guide economic policies. These experiences have shown that modeling and predicting economic and financial variables are extremely important for the economy and finance. Since economic and financial variables are time series that are formed by ordering the observation values according to time, the use of econometric methods in economic and financial modeling has gradually increased due to the developments in the field of time series analysis in recent years. In addition, AI technologies, which have found a wide application area as a powerful statistical modeling technique, have become an alternative to econometric methods and a predictive method with comparable performance. This method, which can produce very successful results in the classification and prediction of time series, is also used extensively in the fields of statistics, economics, and finance (Kaastra & Boyd, 1996, p. 216).

There are many studies in the literature on the modeling and estimation of financial variables using AI methods. These studies deal with the financial performance and financial forecasting of the markets, the forecasting of financial crises, the forecasting of exchange rates, and the forecasting of stock prices. The following studies and their findings are examples of studies conducted with AI methods in the literature.

Wu and Lu (1993), in their study using ANNs and Box Jenkins' ARIMA model to estimate the Standard & Poor's 500 index for the period 1971–1990, they concluded that the ANNs outperformed the time series (Wu & Lu, 1993, pp. 257–264).

Hsieh (1993) showed that it gives very successful results in financial management by using artificial neural networks, which try to simulate the physical process on which intuition is based, or the adaptive biological learning process, and have the ability to produce solutions even for uncertain and incomplete data for the given problem (C. Hsieh, 1993, p. 12).

Yao, Poh and Jasic (1996) estimated the exchange rates between the US dollar and the five main currencies of the period, using the exchange rates for 2,910 days for the period from 18 May 1984 to 7 February 1995. They used the Japanese Yen, German Mark, British Pound, Swiss Franc and Australian Dollar as the five basic currencies of the period. They estimated the exchange rates using ARIMA and ANN models and found that the ANN model gave more effective results in estimating exchange rates (Yao et al., 1996, pp. 754–759).

In recent studies, it has been seen that AI technologies are used in analysis together with traditional statistical methods, so that better results can be obtained by reflecting the strengths and advantageous aspects of both methods. One of the studies that gave these successful results is the study of Hu and Tsoukalas in 1999, which estimated the volatility of the European Monetary System exchange rates by combining the GARCH, EGARCH, and IGARCH models with the moving average variance (MAV) model and an artificial neural network model. In this study, it was concluded that ANNs outperformed the least squares and simple averaging methods (Hu & Tsoukalas, 1999).

Donaldson and Kamstra (1999) estimated the price volatility of the S&P 500 stock index using ANNs with the GARCH and moving average variance (MAV) models. They showed that, compared to the traditional weighted least squares method, they could take into account the interaction effects in time series estimations and that ANNs were more effective in non-linear time series (Donaldson & Kamstra, 1999, pp. 227–236).

Dahlquist, Engström, and Soderlind (2000) measured the performance of 210 funds between 1993 and 1997 with the alpha coefficient obtained by linear regression of fund returns and various benchmarks and concluded that the funds generally did not perform well (Dahlquist, Engström ve Soderlind, 2000, 409-423).

Zhang (2001) tested the applicability of ANNs to linear time series problems. Zhang compared eight different ARIMA models with nonlinear multilayer ANN models for predicting IBM stock closing prices. As a result, the study revealed that ANNs are successful in solving linear problems as well as nonlinear problems (G. P. Zhang, 2001, pp. 1183–1202).

Bollen and Buse (2001), in their study on 230 funds during the 1985–1995 period, investigated whether there is a difference in timing ability when the daily returns and monthly returns of the funds are used. Regression analyses developed by Treynor and Mazuy (1966) and Henriksson and Merton (1981) were used as a method. In the method developed by Treynor and Mazuy, it was determined that 11.9% of the funds were used in the case of monthly returns and 34.2% of the funds showed timing ability in the case of daily returns. Similar results were obtained with the method developed by Henriksson and Merton (Bollen & Busse, 2001, pp. 1075–1094).

Chen, Leung, and Daouk (2003) have tested The direction of the Taiwan Stock Exchange Index, one of the fastest growing financial stock markets from developing Asian countries, with probabilistic neural networks and the Kalman Filter, which is one of the parametric statistical methods, using the generalized moment method and the random walk model, and the probabilistic neural. It has been observed that the network model makes predictions with superior performance compared to other traditional methods, and higher returns are obtained from investment strategies made according to this method (A.-S. Chen et al., 2003, pp. 901–923).

Kim, Oh, Sohn, and Hwang (2004) examined the effects of the financial crisis in South Korea in 1997 on the economic structure of Korea and developed an early warning system with multi-layer ANN models. They divided the year 1997 into three main periods: the stable period between January 3 and September 18, the unstable period between September 19 and October 21, and the crisis period between October 22 and December 27. They used the KOSPI stock index as an input variable in the model, and, considering that index volatility gives information about the direction of the market, they calculated the index's end-of-day closing value, daily return, 10-day moving average, variance, and variance ratio. In the study covering the period 1994-2001, they found that the volatility of the KOSPI index was a harbinger of the crisis and realized that ANN models were impressively successful in early detection of the 1997 economic crisis, in classifying the market movements, and in following the fundamental trend of the economy (T. Y. Kim et al., 2004, pp. 583–590).

Yu, Tresp, and Schwaighofer (2005) on the other hand, combined parametric linear models with non-parametric Gaussian processes and concluded that nonlinear estimations yield better results than traditional estimations (Yu et al., 2005, pp. 1012–1019).

Dutta, Jha, Kumar, and Mohan (2006) used the multi-layer ANN method in modeling the Mumbai Stock Exchange index. They concluded that the ANN model successfully predicted the index values (Dutta et al., 2006, pp. 283–295).

Panda and Narasimhan (2007) estimated the future value of exchange rates with a linear autoregressive model, a random walk model, and an ANN model using weekly data of Indian Rupee/US Dollar (INR/USD) exchange rates for the period January 1994–June

2003. They found that the ANN model gave more effective results for the firm and investors in estimating the exchange rate (Panda & Narasimhan, 2007, pp. 227–236).

Tseng, Cheng, Wang, and Peng (2008) estimated the volatility of Taiwan Stock Index (TXO) prices with a new hybrid asymmetric volatility approach, the multi-layer ANN option pricing model. They used EGARCH and Grey-GARCH models as comparison criteria. They concluded that the ANN model predicts market volatility more effectively (Tseng et al., 2008, pp. 3192–3200).

Liang, Zhang, Xiao, and Chen (2009) estimated option prices using Hong Kong options stock market data for the period 2006–2007 using ANN, finite difference, and Monte Carlo methods. Forecasting was done first using traditional option pricing methods, and then ANN and support vector regression (SVR) were used to reduce forecast errors. Thus, future option prices were estimated using parametric and non-parametric methods, and they concluded that the ANN method showed superior forecasting performance (Liang et al., 2009, pp. 3055–3065).

According to Tsai and Wang (2009), there are two methods used in estimating stock prices in the literature. These are fundamental analysis, which uses information from the company's financial statements, and technical analysis, which uses figures and graphs based on historical data. However, since fundamental and technical analysis alone are not sufficient to make the right decision, they used two analysis methods based on computer technologies that have better forecasting performance. With a hybrid model combining ANNs and decision tree management (DT), they achieved forecasts with an accuracy of 77%, which is higher than the individual models (Tsai & Wang, 2009, p. 60).

At the beginning of October 2008, the size of the world stock market was estimated at approximately $36.6 trillion in the United States. Considering that the total world derivatives market is 11 times the size of the entire world economy and has a nominal value of approximately 791 trillion dollars, it also shows why stock market forecasting is so important and why it is the subject of so many academic studies. Dase and Pawar (2010) compiled the studies in the literature that make stock market index prediction with the artificial neural network method, which gives successful results in large data sets. In these studies, it has been shown that the artificial neural network method has proven to have quite

high predictive performance in predicting the stock index and predicting whether it is best to buy, hold, or sell stocks (Dase & Pawar, 2010, pp. 14–10).

In Vadlamudi's (2017) study, in which studies in the literature on stock market estimation were examined, it was shown that AI methods made more effective predictions than traditional methods when methods such as linear regression, AI networks, and genetic algorithms were compared (Vadlamudi, 2017, pp. 123–128).

Chen, Zhang, Yeo, Lau, and Lee (2017) used the parametric statistical methods of linear regression (LR) and Support Vector Regression (SVR) to predict the volatility of stocks on the Chinese Stock Exchange, which is very popular in both the business world and the academic community. They estimated and analyzed with Repetitive Artificial Intelligence (RNN) and Gated Repetitive Units (GRU), which are AI methods, and it was seen that the estimation results of AI methods were more successful and showed higher estimation performance (W. Chen et al., 2017, pp. 1–6).

Chong, Han, and Park (2017) used three data representation methods with a data set consisting of high-frequency daily stock returns. Principal component analysis, an autoencoder and constrained Boltzmann machine, and a three-layer deep neural network (DNN) model to predict future stock returns were applied to the Korean stock market. They observed that DNNs performed better than the linear autoregressive model in the training set, but the advantage was mostly lost in the test set (Chong et al., 2017b, pp. 187–205).

According to Lin, Chien, and Cheng (2018), according to behavioral finance, investors' behaviors such as greed and fear during transactions in financial markets also affect market trends, so traditional financial analysis or technical analysis can be used to forecast short-term market trends from these highly complex data. argued that their method was inadequate. Using neural networks in their work, they have proven that financial trading markets follow a certain trading logic by integrating physical momentum behavior into financial engineering technology analysis and market profile theory. They showed that buying and selling behavior in financial markets can be explained by the physical trends of the quantitative and technical analysis of the market profile theory (C.-C. Lin et al., 2018, pp. 756–764).

Nunes, Gerding, McGroarty, and Niranjan (2019) showed that the multilayer AI model gave better results, being the first comprehensive study to use a multivariate linear regression model and a multilayer artificial neural network model to predict the European yield curve. This result also paved the way for the development of better forecasting systems for fixed income markets (Nunes et al., 2019, pp. 362–375).

Cao and Wang (2020) used principal component analysis and ANNs for an accurate and effective stock prediction model and proved that the artificial neural network model they created offers an effective stock selection strategy (Cao & Wang, 2020, pp. 7851–7860).

Jing, Wu, and Hefei (2021) created a hybrid model by combining the stock prices traded on the Shanghai Stock Exchange with investor sentiment analysis and DL analysis, which is an AI method. They concluded that the hybrid model they created by using the Long-Short-Term Memory Neural Network approach to classify the hidden emotions of the investors and to analyze the technical indicators in the stock market using the Convolutional Neural Network model outperformed the basic classifiers in classifying investor sentiment and outperformed the prediction of stock prices (Jing et al., 2021, p. 115019).

Cheng, Yang, Xiamg, and Lui (2022), noting that the financial market capitalization of listed companies in the US reached $30 trillion in 2019, which is more than 1.5 times the US Gross Domestic Product, emphasized how important it is for both investors and financial institutions to predict the price movements of stocks in this huge but volatile market. In their analysis using Multimodal Graph Neural Networks (MAGNN), they explained the construction process of the leading lag effect and heterogeneous graphs, which is a new approach for financial time series analysis that plays a major role in hedging market risks and optimizing investment decisions, and concluded that this method shows superior performance in market forecasting. They have shown that, this method provides investors with a profitable as well as interpretable option and enables them to make informed investment decisions (Cheng et al., 2022b, pp. 108–2018).

Many studies have been conducted using various econometric analysis methods to predict financial crises, detect chaos and uncertainties, and determine the mobility of capital markets, and the effectiveness of the methods has been compared. In this sense, besides the

econometric methods mentioned above, studies using Logit and Probit models, although few in number, have also taken their place in the literature. These studies are listed below.

Frankel and Rose (1996) used a probit model to identify leading indicators of currency crises for more than 100 developing countries using annual data for the period from 1971 to 1992. They concluded that high domestic credit growth, high foreign interest rates, and persistently low foreign direct investment to debt ratios indicate a high probability of a collapse (Frankel & Rose, 1996, pp. 351–366).

Kaminsky and Reinhart (1999) examined the sources and extent of 76 currency crises and 26 banking crises using monthly data on 16 macroeconomic variables, whose findings they considered likely to be indicative. They concluded that financial liberalization leads to a banking crisis, banking sector problems trigger the currency crisis, and the currency crisis caused by the banking crisis weaklened the already weak banking sector, and that this process, which triggered each other by deepening the banking crisis, became a vicious circle. They aslo concluded that when currency and banking crises occur together, the impact is deeper than when they occur separately (Kaminsky & Reinhart, 1999, pp. 473–500).

Kim (2003) forecasted the stock price index, a financial time series, with a logistic model and ANNs and concluded that ANNs are a promising method in stock market forecasting (K. Kim, 2003, pp. 307–319).

Uğurlu and Aksoy (2006) used the separation model and the logit model to identify the indicators of corporate financial crises in emerging markets during periods of economic depression and emphasized that the separation model, which was the primary method for predicting financial failure until the 1980s, has now been replaced by logistic regression and that ANNs have yielded better results in predicting financial failure in recent years (Uğurlu & Aksoy, 2006, pp. 277–295).

Kaya and Yılmaz (2007) performed a logit analysis with 31 explanatory variables for the period 1990–2002, using the logit and signal methods, which are widely used in estimating currency crises, for the monthly data of the 2003–2005 period in Turkey. Except for two of these variables, the findings of the signal analysis made with 29 variables were combined to form a common cluster, and they used five macroeconomic variables, which

they defined as "traditional indicators" for this period, containing the findings of this common cluster, to form a financial pressure index. They tested the performances of the models they established and concluded that both methods gave successful results, but the leading indicators differed in terms of periods and economies (Kaya & Yilmaz, 2007).

Davis and Karim (2008) used the polynomial logit model and an early warning system to predict leading indicators to enable the early detection of crises in the banking sector. They found that the polynomial logit model performed better in predicting a global banking crisis, while the early warning system for a country-specific banking crisis did better (Davis & Karim, 2008, pp. 89–120).

Lin, Khan, Chang, and Wang (2008); Kaminsky and Reinhart, in their 1999 study, created four models to predict the exchange rate crisis using data from 20 countries between 1970 and 1998. They measured the model performance of the early warning systems they created by using four modeling techniques, namely the logistic regression model which is one of the traditional methods, the KLR model, ANNs, and fuzzy logic. They concluded that the most successful models are fuzzy logic, ANNs, and the logistics models, respectively (C.-S. Lin et al., 2008, pp. 1098–1121).

Öğüt, Doğanay and Aktaş (2009) used data mining techniques ANNs, Support Vector Machine, and multivariate statistical techniques (discriminant analysis and logistic regression methods) to detect manipulations in the prices of stocks on the Istanbul Stock Exchange. They concluded that multivariate techniques performed better (Öğüt et al., 2009, pp. 11944–11949).

Kovacova and Klistik (2017) concluded that the probit model performed better in their study using logit and probit regression models, in which Slovak companies create their bankruptcy forecasts (Kovacova & Kliestik, 2017, pp. 775–791).

Kantar and Akkaya (2018) estimated the effects of financial liberalization in Turkey with the financial pressure index and logit and probit models they created using 19 macroeconomic variables for the period of January 2005-January 2017 and tried to reveal the leading indicators of the financial crisis. They concluded that the increase in deposit rates, the increase in the domestic debt stock, and the decrease in gross reserves cause an increase the probability of a crisis (Kantar & Akkaya, 2018, pp. 575–590).

According to Akkaya and Kantar (2019), the importance of predicting banking crises, which became a more important issue after the 2008 global crisis, has increased due to the effects they have on the economy. For this reason, in their study, they examined the fragility structure of the Turkish banking sector using annual data and using logit and probit models (limited dependent variable models) for the period 1996-2017. While the exchange rate and deposit interest variables, which have high explanatory power in all three models, are statistically significant in the logit model, they have reached the conclusion that the loan amount, deposit amount, and deposit interest variables are statistically significant in the probit model (Akkaya & Kantar, 2019, pp. 131–145).

Some studies, taking the Turkish capital markets into account, also reveal that artificial neural network modeling produces more successful results than other models. Yıldız (2001) worked on industrial, commercial, and service enterprises in Turkey between the years 1983 and 1997, which are subject to the Capital Markets Board and/or traded on the Istanbul Stock Exchange (ISE). He concluded that the ANN model is more successful than the separation analysis in predicting financial failure (Yildiz, 2001, pp. 47–62).

Diler (2003) estimated the direction of the ISE National-100 Index the next day by using the error back-propagation method with ANN modeling and concluded that the method can predict the next-day value of the ISE National-100 Index by 60.81% (Diler, 2003, pp. 65–82).

Benli (2005) developed financial failure prediction models based on logistic regression and an artificial neural network model by using data from 17 privately owned commercial banks transferred to the Savings Deposit Insurance Fund and 21 privately owned commercial banks in the 1997–2001 period. It has been determined that the power of the artificial neural network model to predict financial failure (82.4%) is superior to the logistic regression model (76.5%). Therefore, it has been determined that the artificial neural network model can be used as a important tool to predict financial failure for all information users (Benli, 2005, pp. 31–46).

Altay and Satman (2005) tried to estimate the returns of the ISE 30 and ISE all indexes using multilayer ANN and linear regression methods. It has been concluded that while ANN models do not give better results than linear regression for monthly and daily

returns, they are quite successful in estimating the direction of index returns (Altay & Satman, 2005, pp. 18–33).

Karaçor and Alptekin (2006), sought answers to the following questions in their study. Could the November and February crises have been prevented with the help of leading indicators? Have the leading indicators worked for the Turkish economy? If it did, could we predict possible crises? Can we avoid the crisis? Can we manage the crisis? By using macroeconomic indicators such as exports, international reserves, the real exchange rate, the real deposit interest rate, and the production index, which are accepted as leading indicators of the crisis, and concluded that the leading indicators predict the crisis process with little error (Karaçor & Alptekin, 2006, pp. 237–256).

Avcı (2007) estimated the daily and session returns of the ISE-100 index with the multi-layered ANN model and determined that the forecast performance is strong and can be used as a financial performance measure. With this study, it has been shown that effective and strong predictions are achieved with the ANN modeling technique (Avci, 2007, pp. 128–142).

Altan (2008) estimated the exchange rate with an ANN and a vector autoregressive (VAR) model using monthly data on the exchange rate (TL/USD) for the period January 1987–September 2007. It has been concluded that the predictions made with an ANN architecture that learns by multi-layer feed-forward and back-propagation yield very effective results (Altan, 2008, pp. 141–160).

Aladağ, Eğrioğlu and Kadılar (2009) modeled their study with ANNs, which are increasingly important in time series analysis and are used in many fields. They analyzed Canadian Lynx data with the hybrid model they created by combining ARIMA and Elman's reversible artificial neural network model and concluded that this proposed model has the best prediction accuracy performance (Aladag et al., 2009, pp. 1467–1470).

In their study, Yolcu, Eğrioğlu and Aladağ (2013) concluded that time series with linear and non-linear structures are better predicted by ANNs and hybrid artificial neural network models, which are more effective methods than traditional methods (Yolcu et al., 2013, pp. 1340–1347).

Avcı (2015) predicted the Istanbul Stock Exchange 30 Index with multilayer ANNs models and observed that the buy-and-hold strategy had a great advantage in most of the examined periods (Avci, 2015, pp. 443–461).

9. METHODOLOGY AND DATA

Especially since the global economic crisis in 2008, researchers, policymakers, and investors have understood how important financial forecasts are in preventing financial crises and uncertainties. Although there is no clear consensus on when the global economic crisis began, some researchers believe it began with the failure to repay subprime mortgage (low value hold and sell) loans in the United States in 2006, while others believe it began with the collapse of the large banking company Lehman Brothers in 2008. Everything began with the failure of the large banking firm, Brothers. However, the impact area of the crisis expanded and globalized in 2008, and the crisis spread to Europe and the rest of the world.

The aim of this study is to predict the long-term relationship between the ISE100 index value and the selected macroeconomic variables in the January 2001-January 2022 period by using Logit and Probit analysis and the deep neural networks method and to compare the estimation performances of the results obtained for both methods. Thus, the effectiveness of AI technologies, which emerged from developments in traditional econometric methods and computer technologies and continued to develop rapidly in recent years, in time series analysis will also be measured. For the models used in the research, the most commonly used variables in studies on the market index in the literature were selected (W. Chen et al., 2017; Cheng et al., 2022a; Donaldson & Kamstra, 1999; Kara et al., 2011; Staub et al., 2015) and the meaningful variables obtained from the created meaningful models were given as input to the models created by ANNs and DL methods, and the effectiveness of the modeling methods was tested.

In the study, after giving information about AI technologies, it is intended to develop and evaluate an alternative decision-making system that can be used to predict the short- and long-term movement, trend, and price of the stock market. It also provides a review of the literature on AI technologies applied to analyze financial problems, focusing mainly on the modeling process. For this purpose, macroeconomic variables, which are frequently used in studies on stock market indices in other domestic and foreign literature and proven to have effects on stock prices, were chosen based on the studies made by Chen, Roll, and Ross (1986) and Prantik and Vani (2004) on the basis of the literature review (W. Chen et al., 2017; Cheng et al., 2022a; Kara et al., 2011; Staub et al., 2015)

While the price movements of financial assets are observed in financial markets, the price of assets is generally used in empirical studies (Campbell et al., 1997, p. 9). They presented two main reasons for using returns in the analysis. First, the return on a financial asset provides a complete and independent summary of investment opportunities for an investor. In other words, the investor who invests 1000 TL in a financial asset with an annual return rate of 5% will have 1050 TL at the end of the year. The second reason is that asset returns are more useful than prices for both empirical and theoretical reasons. In other words, price series display a non-stationary appearance due to the presence of a stochastic and/or deterministic trend and constant deviations from the unconditional mean. Asset returns, on the other hand, are close to the average and only deviate from it in the short run. Therefore, return series are generally stationary (Çil, 2018, p. 7).

The study consists of 253 monthly observation values covering the time interval between 2001:01 and 2022:01. In order to estimate the value of the Borsa Istanbul 100 index selected as the input (dependent) variable, or, in other words, to investigate the effects on the stock market in Turkey, seven macroeconomic variables were determined. These are the Dollar Rate (TL/$), Money Supply, Producer Price Index, Industrial Production Index, Gold Price (TL/Gr), Active Bond Interest Rate and Brent Oil Price. In addition, since the effects of the global economic crisis experienced in the world in 2008 were more limited in Turkey than in developed countries, these periods were not excluded from the data set in order not to spoil the integrity of the data set and to see the effects of the crisis in Turkey in the 2001 period.

Seven macroeconomic variables were used as independent (input) variables in the models. ISE100 index values were used as the dependent (output) variable to represent the stock markets. This index is the basic index for the BIST stock market. In addition, since monthly series were used in the study, the Gross Domestic Product series could not be obtained, so the industrial production index was used to represent this variable. All variables were subjected to statistical analysis, and the Eviews 12, STATA 15, and MATLAB R2021b programs were used in the analysis.

The data sets used in the analysis were compiled as monthly series from the Central Bank of the Republic of Turkey Electronic Data Distribution System (EVDS), the Turkish

Statistical Institute (TUIK) statistics, the monthly statistical bulletins of the Capital Markets Board, the statistics of the Ministry of Treasury and Finance, and the Eurostat databases.

While the monthly values of the ISE100 index were used as the dependent variable in the research, the consumer price index, industrial production index, gold bullion prices, dollar rate, crude oil barrel prices, and monthly values of active bond interest rates were used as independent variables. The arguments used are as described below:

Index: It is an indicator that measures the proportional change of investment instruments traded on the stock exchange within a certain time period by collecting data such as price, cost, sales, and production. Stock indices, on the other hand, measure the price and return performance of stocks traded in the stock market. These measurements can be made on both a holistic and sectoral basis. In this way, it is possible to monitor the economic performance of stocks and sectors.

ISE100 Index: It is a fundamental index and fundamental indicator used to assess the performance of the top 100 stocks traded on the Borsa Istanbul in terms of market and trading volume. It consists of 100 stocks selected to be used in the futures market among the publicly offered stocks of companies traded in the national market, excluding investment trusts, with high market value and liquidity, taking into account their sectoral representation ability. The transaction code of the ISE100 index, which is also accepted as an indicator of the Turkish stock market, is XU100. Indices are calculated from prices. As the country's financial system improves, the stock market index also rises. The ISE100 index is used as a benchmark for other investment instruments. The ISE100 index allows one to measure the price movements of the stocks in the index under certain conditions and thus determine the general trend of the stock market. Price indices such as the wholesale price index and the consumer price index are indicators that measure the change in the general level of prices.

Although stock market indices are a basic indicator for national economies, it is necessary to know what the factors affecting the index are before making an evaluation. Internal factors such as a country's economic data, unemployment figures, interest rates, geopolitical position, and risks, as well as external factors such as global data and economic relations between countries, are factors that influence a country's economic course and, of course, stock market indices.

Two Years Bond Yield/ Active Bond Interest Rate: This rate is the 2-year bond rate. Bonds are debt instruments used by governments, companies, or individual investors in need of funds for various reasons, to provide financing for a certain period of time with a variable or fixed interest rate. Investors prefer this investment tool to protect themselves from economic risks and obtain a fixed return above inflation. While increasing interest rates and increasing economic uncertainties due to the implementation of tight monetary policies direct the investor to bonds in order to be protected from the said risks, as a result of stable economic growth, interest rates tend to fall, leading them to stocks due to the reduction of risks.

Interest is called the rent or price of money. If the interest rates that direct the entire country's economy decrease, there is an economic recovery, while if the interest rates increase, an economic slowdown occurs. Because when interest rates increase, fund owners prefer alternative investment instruments that they anticipate will bring higher returns and therefore do not turn into production. This will undoubtedly affect the banking sector as well. For these reasons, stock returns significantly affect investors' investment decisions in the markets. When the market interest rate increases, the bond price decreases. A stock is a security, like a bond. Therefore, when the Interest Rate of Active Bond increases, the stock market index is expected to decrease. In other words, there is a negative relationship between the two variables.

Since the investments to be made in stocks, bonds, and deposits are the types of placements fed from the same fund source, the investor has two investment preferences. The first is to buy bonds. The second is to earn interest income in return for deposits in the bank or to earn stock market income by purchasing stocks. Therefore, in a situation where the interest rates rise, the investor will turn to bonds and bills because he believes that he will protect himself from risks, and in a situation where the interest rates decrease, he will turn to stock investments (Eğilmez, 2009). The graph in Figure 9.1. below shows the direction of the relationship between the two variables.

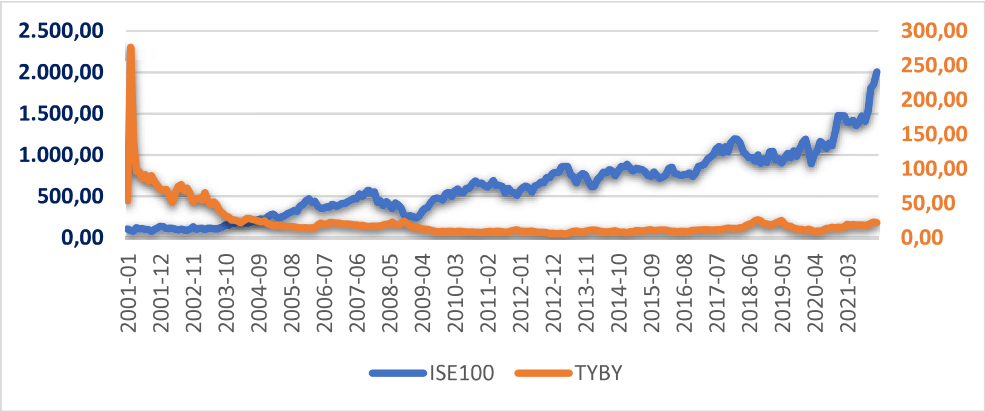


Figure 9.1. ISE100-TYBY

The graphic above shows the reflections of the financial crisis in Turkey in February 2001. The financial crisis was reflected in the money market, and there were serious increases in interest rates and exchange rates due to the increase in demand for TL and foreign currencies and the inability to meet this increasing liquidity demand. On February 21, the overnight interest rate in the interbank markets increased to 6200%. When the foreign exchange reserves of the central bank decreased significantly, it was unable to meet these sharp speculative movements and left the exchange rate to fluctuate.

Exchange Rate: The exchange rate is a price and relates to commodity markets. The stable course of the changes in the exchange rate affects economic stability positively. After the liquidity crisis on November 22, 2000, and the currency crisis on February 19, 2001, Turkey abandoned the disinflation program based on adjustable fixed rates and switched to floating rates. The equilibrium exchange rate has a linear relationship with the ratio of price levels. The exchange rate is also a macroeconomic variable that has been identified as a benchmark for some securities. Since the crises in the foreign exchange and stock markets cause uncertainties in both national economies and international markets, they have been the focus of studies. It is very important both for governments to determine policy and for investors to direct their investments. When the exchange rate increases, the returns on stocks decrease as investors' funds shift to other financial instruments. Therefore, empirical studies have shown that while there is a negative relationship between the two variables in the short term, there is also a positive relationship in the long term. The constant rise in foreign exchange prices will cause investors who bind their savings to stocks to sell their stocks and turn to foreign exchange. In this case, as the demand for stocks decreases, their returns will

also decrease. The graph in Figure 9.2 below shows the direction of the relationship between the two variables.

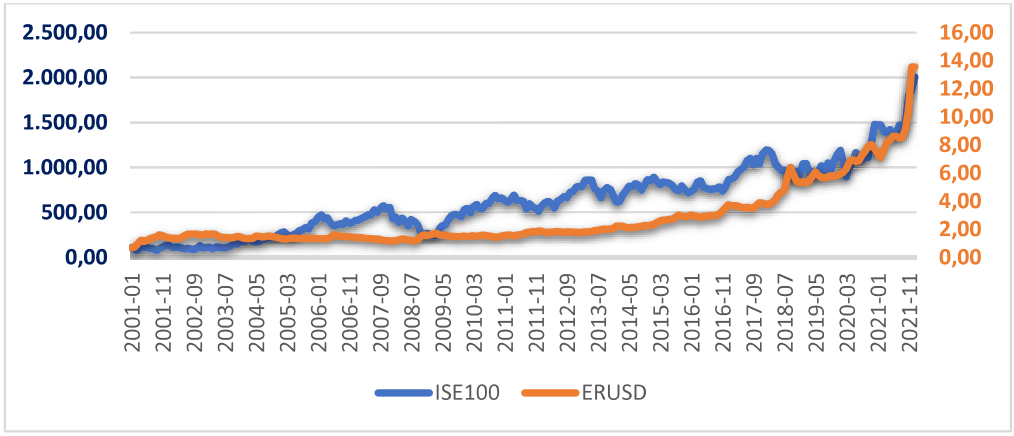


Figure 9.2. ISE100-ERUSD

The financial crisis experienced in Turkey in February 2001 was reflected in the money market led to an increase in the demand for TL and foreign currency in interest rates due to the due to the inability to meet this increasing liquidity demand. This situation is shown in Figure 9.2. Likewise, there were serious increases in the exchange rates, and while the dollar rate was 686.500 liras on February 19, 2001, it rose to 960.000 liras with an increase of around 40% per day. The Turkish lira depreciated by 11% in real terms. 19 banks were closed, economic contraction was experienced, and 1.5 million people were unemployed.

The exchange rate, which started to increase after 2015, reached 14.00 TL in the 11th month of 2021. Since 2022, when similar problems were experienced, exchange rates and inflation have increased rapidly, and the Central Bank's reserves have decreased. In order to prevent investors from turning to foreign exchange deposit accounts due to the increasing exchange rates, a new application called a "Currency Protected Account" has been started by the TCCB in order to support the Turkish Lira Deposit or Participation Account opened with banks as of May 23, 2022. This method is a type of deposit that aims to give a higher return by adding the difference of the changes that will occur in addition to the interest in TL deposit accounts. However, it can be seen from the definition that this method does not differ from paying interest on interest. It is observed that the rise in exchange rates also led to an upward movement in the stock markets.

Money Supply/M2: The M2/GNP ratio is an indicator of the financial depth of the country's economy. It is also an indicator of the use of the banking system by the public. When the money supply increases, interest rates will fall and the prices of bonds and stocks will rise. Therefore, investments will be directed toward stocks. Then there is a positive relationship between the two variables. The money supply is a size that includes money in circulation and time and demand deposits.

M1= Currency in circulation + Demand deposit

M2= M1 + Time deposit

The graph in Figure 9.3 below shows that the relationship between the two variables is in the same direction and they move together. Following the failure of 19 banks during the 2001 financial crisis, the public's use of the banking sector has increased dramatically in Turkey. It can be said that, as of January 2022, banking transactions have replaced those on the stock market.

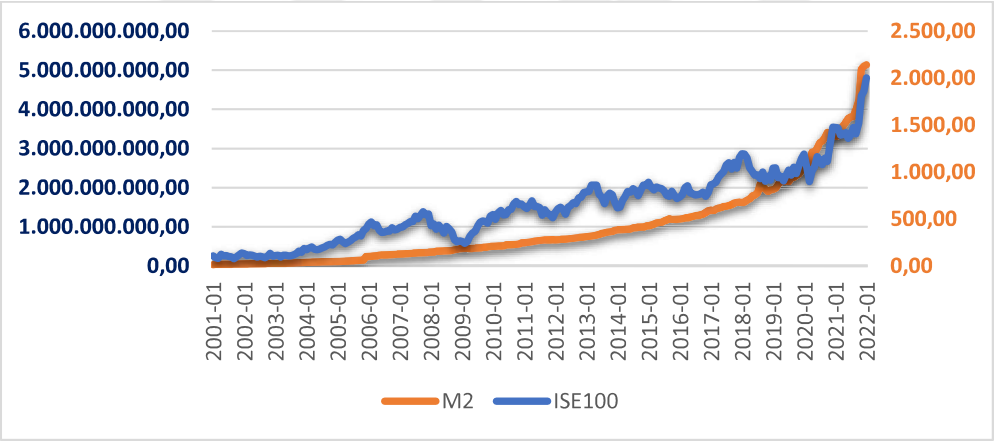


Figure 9.3. ISE100-M2

IPI (Industrial Production Index): The Industrial Production Index is one of the indices among the statistics announced by TURKSTAT. It is an index that gives the industrial production/GNP ratio. It consists of weighting all branches of the industry according to production classes (85% of manufacturing, 20% of electricity, 20% of mining). Increasing the industrial production index means more production and the growth of the Turkish economy. When this index, which measures economic activity, rises, it means that the Gross National Product (GNP) rises as well, implying that prices have risen. In this case,

the value of the stocks also increases. Therefore, there is a positive relationship between them. The relationship between the two variables is also seen in the graph below.

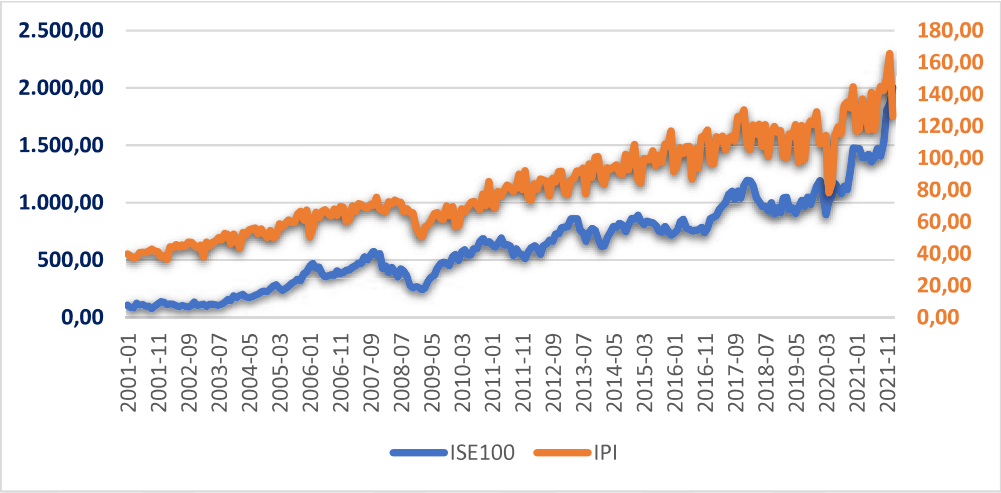


Figure 9.4. ISE100-IPI

PPI (Wholesale Price Index- Domestic Producer Index): In Turkey, the index calculated as the Producer Price Index (PPI) since November 10, 2005 and the Domestic Producer Price Index (D-PPI) since February 26, 2014, replacing the index known as the Wholesale Price Index (WPI) by the Turkish Statistical Institute, started to be used. This index, which measures the price level of the country, covers the goods and services that are subject to sale in the country. Therefore, it is a measure of inflation that shows the general level of prices. When the Domestic Producer Price Index increases, stock returns are expected to increase as prices increase. In other words, there is a positive relationship between the two variables. The graph in Figure 9.5 below shows the direction of the relationship between the two variables.

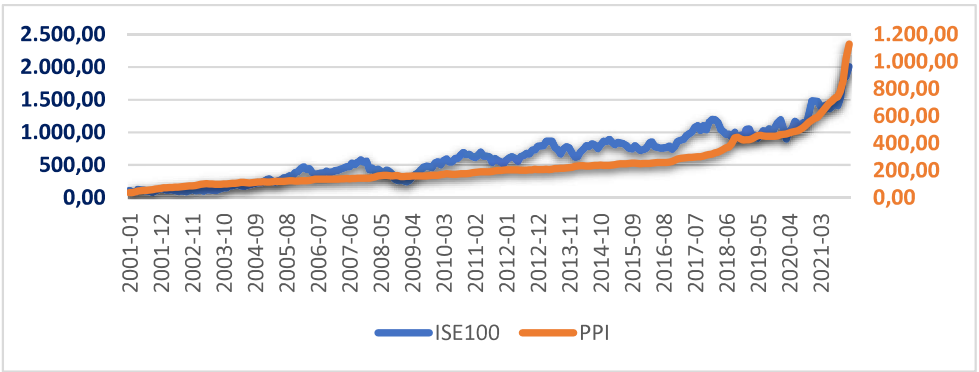


Figure 9.5. ISE100-PPI

Gold Prices: The monthly selling price in Turkish Lira of gram gold bullion, which is an alternative investment instrument, has been taken. Although gold has been around for hundreds of years, it is one of the most preferred alternative investment instruments, especially in recent years, since it has low risk. Since the variable is included in the analysis as Turkish Lira, it will be possible to observe how the changes in Turkish Lira affect the stock market index. The increase in gold prices will lead the investor to turn to gold as an alternative investment tool. In an inflationary environment, however, investors will prefer a low-risk, high-return investment instrument. Gold, as a precious metal, has always remained a safe haven for investors. The graph in Figure 9.6 below shows the direction of the relationship between the two variables.

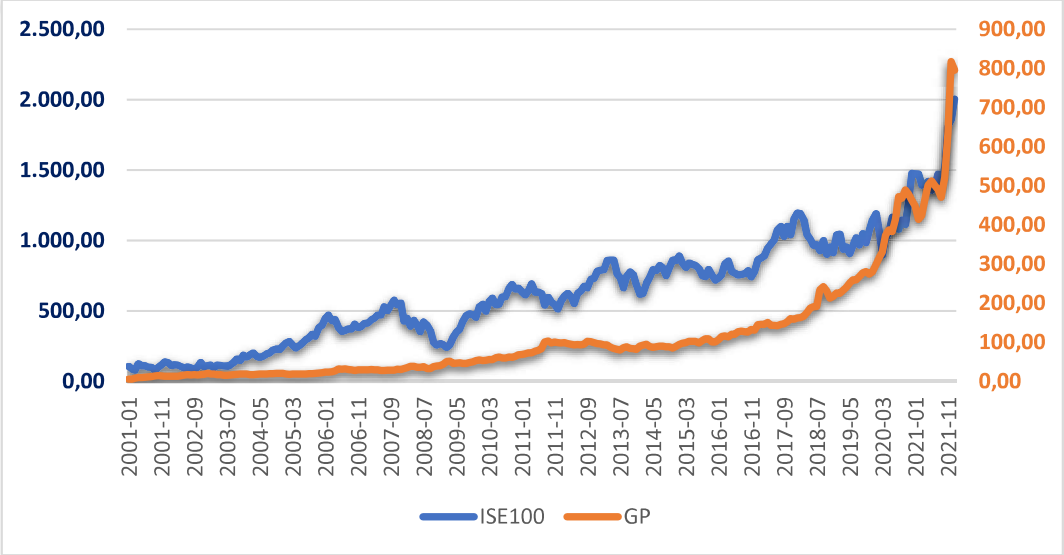


Figure 9.6. ISE100-GP

Brent Oil Price: Oil, which is one of the primary energy sources, is the basic and indispensable input of many sectors. This feature is therefore an important cost factor. The increase in oil prices will cause cost inflation and create inflationary pressure. Inflationary pressure will cause prices to rise and the central banks to raise interest rates to suppress price increases, leading to investors turning to Treasury bills and bonds and thus lowering stock prices. As can be seen, the components in question are like a loop that has an intricate interaction structure with each other.

Oil is not only an input but also an investment tool. Because fluctuations in oil prices cause changes in economic activity and stock prices. After the oil shock experienced in 1973

and because of the effects of this shock on the economies of the countries, oil started to be used as an important factor in economic analysis. It has been observed that changes in oil prices have a positive effect on the stock market in the long run.

Movements in Brent oil prices affect the country's economies through channels such as the foreign trade balance, monetary policy, and inflation. Rising oil prices negatively affect the foreign trade balance of oil-importing countries. It causes a wealth transfer to oil-exporting countries. Since Turkey is an oil-importing country, when oil prices increase, money demand increases, and if money supply does not increase in parallel with this demand, interest rates increase and investment costs increase. The lack of this resource, which is a very important input for many sectors, especially energy-intensive sectors, causes a contraction in these sectors. Therefore, increased costs lead to shrinkage and inflation. This continuity in price increases leads to a decrease in employment, which leads to a decrease in demand, and a decrease in demand leads to a decrease in production and the continuation of the spiral cycle. Again, this will lead to changes in the profitability ratios of the companies and thus the values of the stocks. Therefore, oil-importing countries such as Turkey are very sensitive to changes in oil prices. The graph below shows the direction of the relationship between the two variables. In Figure 9.7, the high increases in Brent oil prices experienced due to the global crisis in 2008 can be seen. It is seen that the price increase in 2011 almost approached the increase in 2008. Brent oil prices, which had a decreasing trend in 2020, increased rapidly in 2021 and continued to increase.

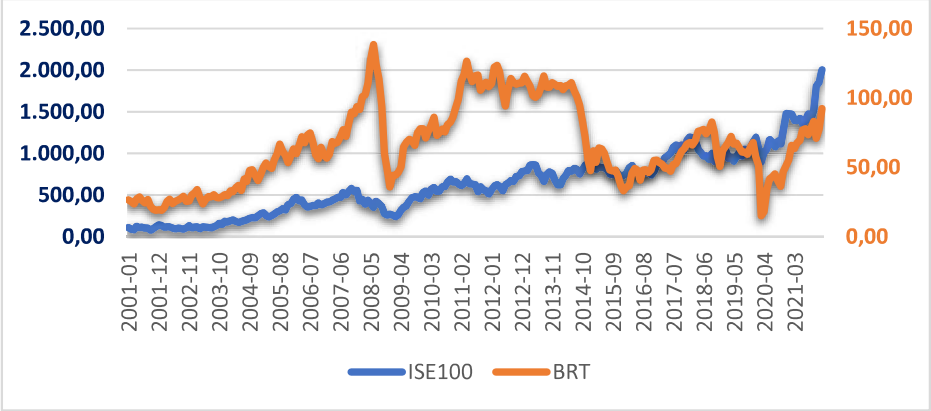


Figure 9.7. ISE100-BRT

Economic models are established by taking into account the equilibrium relations envisaged in economic theory. The existence of significant econometric and economic relations between the variables in the established model also depends on the stationarity of the series. Financial time series, on the other hand, are not stationary because they generally show a volatile structure. Therefore, first of all, the variables to be used in the analysis should be made stationary.

It is an important step to determine the stationarity of the series in the analysis made with time series, since it may seem like there are relationships that do not exist in the analysis made with non-stationary series and therefore may cause spurious regression problems. Whether the statistical tests are meaningful or not in the modeling depends on these analyses. The Efficient Market Hypothesis for the stock market is tested with unit root tests. If the ISE100 index is unit-rooted, the efficient market hypothesis is valid; if it is stationary, the efficient market hypothesis is not valid. Therefore, with unit root tests in financial time series, the trend of asset prices returning to the mean is investigated (Çil, 2018, pp. 288–289).

Since the time series of the macroeconomic variables utilized in the analysis have unit root and seasonality effects by nature, the Generalized Dickey-Fuller Test (ADF) and Phillips-Perron (PP) unit root tests were used to ascertain if the data sets were stationary or not.

In classical regression models, the dependent variable is random and the independent variable or variables are considered constant. When the dependent variable is defined according to whether a certain feature is present or not, it is converted into a binary structure as a qualitative variable. In this case, the dependent variable takes the value 0 or 1. Since these model parameters cannot be estimated by the classical least squares method, the estimations are made with the linear probability, Logit, and Probit models.

In the study, both Logit and Probit model estimations were made using the ISE100 index value as a dependent (output) variable and seven macroeconomic variables as independent (input) variables. Since the dependent variable must have a binary structure in these modeling techniques, a threshold value was calculated by taking the geometric mean of the series for the dependent variable BIST100 index. The variable was converted into a

qualitative or binary structure by assigning a value of 0 to values less than the threshold value and a value of 1 to values greater than the threshold value. Here, the geometric mean is preferred because it is a measure of central tendency that takes into account the geometric differences, rather than the arithmetic differences between the data. Thus, a structure was created that covers all the observations in the data set, conforms to algebraic operations, and allows the processing of relative numbers.

The variables used in the study are shown below:

ISE100	BIST 100 Index
ERUSD	Exchange Rate (TL/$)
WPI-PPI	Domestic Producer Price Index
IPI	Industrial Production Index
M2	Money Supply M2
GP	Gold TL/gr
TYBY	Active Bond Interest Rate or 2-year Government Bond Interest
BRT	Brent Oil Price (Brent/Dollar)

Based on this information and the logit and porbit regression model, which was created with the probability of the index value P_i, the i'th period being greater or less than the geometric mean, will be as in Figure 9.8:

$$\ln\left(\frac{P_i}{1-P_i}\right) = \beta_0 + \beta_1 \text{ERUSD} + \beta_2 \text{PPI} + \beta_3 \text{IPI} + \beta_4 \text{M2} + \beta_5 \text{GP} + \beta_6 \text{TYBY} + \beta_7 \text{BRT} + u_t$$

Figure 9.8. Logit model

$$P_i = 1 - \phi(-x, \beta) = \phi(\beta_0 + \beta_1 \text{ERUSD} + \beta_2 \text{PPI} + \beta_3 \text{IPI} + \beta_4 \text{M2} + \beta_5 \text{GP} + \beta_6 \text{TYBY} + \beta_7 \text{BRT})$$

Figure 9.9. Probit model

The time series of the variables used in the analysis were compiled as monthly series from the Central Bank of the Republic of Turkey Electronic Data Distribution System database (EVDS), the statistics of the Turkish Statistical Institute (TUIK), the monthly statistical bulletins of the Capital Markets Board, the bulletins of the Ministry of Treasury and Finance, and the Eurostat databases. Eviews 12 for the statistical and econometric tests of the time series data of the dependent and independent variables for the period January

2001 to January 2022, Stata 15 for Logit and Probit analysis, and MATLAB R2021b for DL analysis were used.

In the DL model, ISE100 was used as the dependent (output) variable, and seven macroeconomic variables were used as independent (input) variables. In the DL model, which works based on the artificial neural network architecture, the data set is divided into two parts: 70% training and 30% testing. In financial analyses in the literature, AI applications usually make a distinction between data sets in different ratios, but this distinction is determined by trial and error as with other parameters of the model (Chong et al., 2017a; Jing et al., 2021; C.-C. Lin et al., 2018). In this case, the 2001:01–2015:08 period consists of the first 177 observations as the training set and the last 76 observations as the 2015:09–2021:02 test set.

In order to solve the problems in their studies, researchers use modeling and analysis by trying to create the best and most explanatory parameter sets to represent the problem. In the studies carried out as a result of the development of the DL approach, the issues of how to best design a multi-layer artificial neural network structure, how many layers it will consist of, how many neurons it will contain, what the dropout value will be, and which optimization algorithm or activation function to choose have been seen as how important issues are for the solution. However, since there is no definite proposition in the selection of these parameters, the parameters were obtained by trial and error according to the problem and the data set (Jing et al., 2021; Kaastra & Boyd, 1996, pp. 220–224).

In the DL method, which is a ML method that learns from data, the most used parameters while creating the artificial neural network architecture are explained in detail in Chapter 7 of the study. Here, we will quickly touch on the key factors that must be taken into account while determining these characteristics (*Deep Learning Toolbox*, n.d.):

The size and diversity of the data set: The larger the dataset, the better the model learns, although this is not sufficient for a good model.

Mini-Batch Size: In each iteration of DL, gradient descent is calculated retrospectively on the network with the backpropagation process, and the weight values are updated in this way. The higher the number of variables in this calculation process, the longer

the calculation takes. For this reason, the data is divided into small groups, and learning is done through these small groups. It indicates how many data points the model will process at the same time by segmenting data or determining the learning mini-batch parameter.

Learning rate and momentum: In each iteration of learning in DL, the difference is found by backward derivatives with the backpropagation method, and the result obtained by multiplying the value with the learning parameter is subtracted from the weight values and new weight values are calculated. Here, the learning rate can be determined as a fixed value, or it can be learned as a step-by-step value, a momentum-dependent value, or during learning by adaptive algorithms. However, the learning rate usually defaults to 0.01 and is reduced to 0.001 after a certain epoch or cycle. It is seen that the momentum beta coefficient, which is generally used, is also 0.9 and varies in the range of 0.8–0.99.

Selection of the optimization algorithm: In DL, learning is an optimization process. In solving nonlinear problems, algorithms such as stochastic gradient descent (sgdm), adagard, adadelta, and adam are commonly used. The important feature of adaptive algorithms is that they learn the learning rate themselves and are dynamic. DL generally uses the sgdm algorithm by default.

Epoch number: While the model is being trained, the data is trained and tested in parts, not completely, and the weights are updated by back propagation according to the result. This is the name given to each of the training steps.

Weight value and activation function: There are various weight determination methods that affect the learning speed of the model. It is explained in detail in Section 6.3.2. In multilayer artificial neural network models, activation functions such as sigmoid, tanh, and ReLu are mostly used in hidden layers for nonlinear transformation operations. It is used for backing derivatives in hidden layers.

Dropout value: It has been observed that forgetting weak data below a certain threshold in fully connected networks increases the success of learning. In general, the value taken as 0.5 is defined as a value in the range [0,1] when taken as the threshold value.

Layer, number of neurons in each layer, and pooling: The feature that distinguishes DL from ANNs is the high number of layers. That's why it's called "deep learning". As the number of layers increases, other features of the data are learned during the learning stages, so learning is more successful. The number of neurons indicates the amount of information stored in memory. The better the computer technology and processor, the larger the size of the kernels that operate on the matrix in each layer, so the pooling process made on the output of the kernels, in other words, the filtering technique for the data, is also successful.

10. EMPIRICAL ANALYSIS

In this section, the models used during the study are introduced. In order to measure the change in the value of Istanbul Stock Exchange Index by the change in the selected macroeconomic variables, Logit and Probit models from qualitative response regression models and a DL model from AI technologies will be estimated. Thus, the effectiveness of these modeling techniques will be tested.

Istanbul Stock Exchange (BIST100 Index) index values and selected 7 macroeconomic variables for the period 2001:01-2022:01 in Turkey were subjected to statistical analysis and Eviews12, STATA15 and MATLAB R2021b programs were used in the analysis.

10.1. Descriptive Statistics

Descriptive statistics are statistics that describe the general characteristics of a data set statistically, summarize the series, show typical values at which units are stacked, and provide information about distributions.

Standard deviation is the most widely used measure of dispersion. Because the standard deviation expresses how close the observations in the data set are to the mean. Usually, the small standard deviation value indicates that the deviations from the mean are small. In the opposite case, when there is a large standard deviation, it is understood that the data diverge from the mean. The standard deviation is obtained by dividing the sum of the squares of the deviations from the arithmetic mean, which is the variance, to the total number of observations, and taking the square root of this value. In summary, the standard deviation is equal to the square root of the variance.

Summary statistics for the original series dependent and independent variables are given in Table 10.1 below. Analyzing the summary statistics, it is seen that macroeconomic variables are exposed to fluctuations in the 2001:01–2022:01 period, so the variability is high. It is seen that the series are skewed to the left or inclined to values higher than the average, also due to the effects of the economic crisis. This shows that the financial series

are quadratic series. While the averages of macroeconomic variables are different, their standard deviations also show significant differences.

Table 10.1. Some descriptive statistics for the dependent and independent variables

	ISE100	BRT	GP	ERUSD	TYBY	IPI	M2	PPI
Mean	629.4487	65.74174	116.4266	2.780042	21.96500	81.86682	9.86E+08	237.5752
Median	618.5821	63.52000	80.32000	1.712764	14.32000	77.58710	6.49E+08	189.6186
Maximum	2003.200	138.4000	817.6100	13.55286	276.1500	165.5610	5.14E+09	1129.030
Minimum	76.25870	14.85000	5.882500	0.673858	5.200000	36.35654	32202029	37.04517
Std. Dev.	383.8759	28.93278	139.6022	2.227352	25.78224	28.03729	1.06E+09	169.4207
Skewness	0.605083	0.353347	2.275737	2.054375	4.995143	0.369554	1.666464	2.016016
Kurtosis	3.232937	2.206649	8.639739	7.463964	41.35857	2.300925	5.591807	8.236685
Jarque-Bera	16.01029	11.89966	553.6755	388.0260	16562.91	10.91049	187.9144	460.4615
Probability	0.000334	0.002606	0.000000	0.000000	0.000000	0.004274	0.000000	0.000000
Sum	159250.5	16632.66	29455.92	703.3505	5557.144	20712.30	2.50E+11	60106.54
Sum Sq. Dev.	37134889	210950.7	4911174.	1250.197	167510.5	198094.6	2.81E+20	7233247.
Observations	253	253	253	253	253	253	253	253

10.2. Unit Root Test Results

Time series are numerical quantities in which the values of the variables are observed consecutively from one period to the next (Sevüktekin & Nargeleçekenler, 2010, p. 40). A time series is considered stationary if it does not continuously increase or decrease over a given time period and the data scatter around a horizontal axis over time. In other words, a time series is stationary if its mean, variance, and covariance do not change systematically (Sevüktekin & Nargeleçekenler, 2010, p. 229).

The stationarity of the series is a necessary assumption for efficient and consistent estimations. However, economic time series tend to increase over time, so they are not stationary. Regression model assumptions require both dependent and independent series to be stationary and errors to have a zero mean and finite variance. Otherwise, spurious regressions may occur with regression analysis on a non-stationary series. While there are high R^2 values and significant t-statistics in the spurious regression, the parameter estimation results are meaningless in terms of economic interpretation. If there is cointegration between the variables, even if the variables are not stationary when examined individually, taking the linear functions of these variables causes the series to become stationary by removing the

trend of each other and making the spurious regression disappear. Co-integration or cointegretation means that the error term obtained from linear combinations of non-stationary series is stationary. For cointegration to occur, all series must be equally stationary, and the error term obtained from the linear functions of these variables must also be stationary. This correction mechanism also includes time series AR (Autoregressive) corrections and MA (Moving averages) corrections for DF and ADF models. Therefore, we can say that the PP test is an ARMA (Autoregressive Moving Average) process.

While there are various methods for examining the stationarity of time series, the unit root test is one of them. A unit root in a series means that the series is not stationary. The most commonly used unit root tests for determining stationarity are the Dickey-Fuller Test and the Generalized Dickey-Fuller Test (ADF, Augmented Dickey-Fuller, 1979), developed as an extension of it. In addition, the Phillips Perron (1989) Test, which is created by adding a non-parametric addition, is used to correct the error terms, considering that these tests are insufficient when the economic time series are subjected to a break (Wooldridge, 2010, pp. 640–642). This correction mechanism also includes time series AR (Autoregressive) corrections and MA (Moving Averages) corrections for DF and ADF models. Therefore, it can be said that the PP test is an ARMA (Autoregressive Moving Average) process (Phillips & Perron, 1988, pp. 335–346).

The unit root tests and hypotheses used in the study are as follows:

Augmented Dickey-Fuller (ADF) Unit Root Test

The following process is used to explain the use of the test. The starting point is the probabilistic unit root process, which is an autoregressive process.

$$Y_t = \rho Y_{t-1} + u_t \qquad -1 \leq \rho \leq 1$$

Figure 10.1. The probabilistic unit root process

Here, u_t is the error term of the white noise. If ρ =1, which means the series contains a unit root, the above model will be a non-stationary probabilistic process. Therefore, since we cannot predict the coefficient in question with Ordinary Least Squares (OLS) estimators,

when we subtract Y_{t-1} from both sides of the model, Figure 10.2 shows the model with δ=(ρ-1) and Δ being the first difference processor;

$$\Delta Y_t = \delta Y_{t-1} + u_t$$

Figure 10.2. ADF random walk equation

This model is predicted and tested with the following hypothesis.

H_0: $\delta = 0$ There is a unit root in the series.

H_1: $\delta < 0$ There is no unit root in the series.

For a α=0.05 significance level, if, P< α, H_0 is rejected and is said to be the series is stationary. In other words, if the tau (τ) statistic is calculated and the absolute value of the calculated value exceeds the McKinnon DF critical values, the null hypothesis cannot be rejected (Sevüktekin & Nargeleçekenler, 2010, p. 345).

Phillips Perron (PP) Unit Root Test

It is stated that the Dickey-Fuller Test error terms are statistically independent and have constant variance. In the extended Dickey-Fuller unit root test, the lagged values of the dependent variable are added to the independent variable and the autocorrelation problem in the error term is eliminated. The Phillips-Perron Unit Root Test uses non-coefficient or non-parametric statistical methods without adding lagged difference values to account for the sequential relationship, i.e. autocorrelation, in the error term (Gujarati & Porter, 2009, p. 758).

The critical values of the Phillips-Perron (PP) test and the critical values of the ADF test are the same. The unit root test developed by Phillips-Perron (PP) adds new assumptions to Dickey-Fuller's error term. The regression equations of Phillips-Perron are expressed below (Phillips & Perron, 1988, pp. 335–346).

$$y_t = \hat{u} + \hat{\propto} y_{t-1} + \hat{\varepsilon_t} \text{ ve } y_t = \tilde{\mu} + \tilde{\beta}\left(t - \frac{1}{2}T\right) + \tilde{\alpha} y_{t-1} + \hat{\varepsilon_t}$$

Figure 10.3. PP model

Here $\widehat{\varepsilon_t}$ shows the white noise sequence process with an expected mean of zero and T represents the number of observations. No serial correlation or assumption of homogeneity between error terms is required here. Phillips-Perron (PP) produced test statistics to test the hypotheses about the $\hat{u}$ and $\widehat{\propto}$ coefficients. In Dickey-Fuller unit root tests, τ_τ is used to test a model and in Phillips-Perron test it is expressed as $Z_\propto$ (Sevüktekin & Nargeleçekenler, 2010, pp. 364–366).

Economic models are established by taking into account the equilibrium relations envisaged in economic theory. The existence of significant econometric relations between the variables in the established model also depends on the stationarity of the series. In a stationary process, the series fluctuates around a fixed long-term average and the effect of any shock is not permanent. On the other hand, a non-stationary series will not enter the long-term deterministic path and will permanently reflect the effects of shocks in the short-term to the long-term values. Time series are not stationary because they usually show a volatile structure. Therefore, the variables to be used in the analysis must first be made stationary.

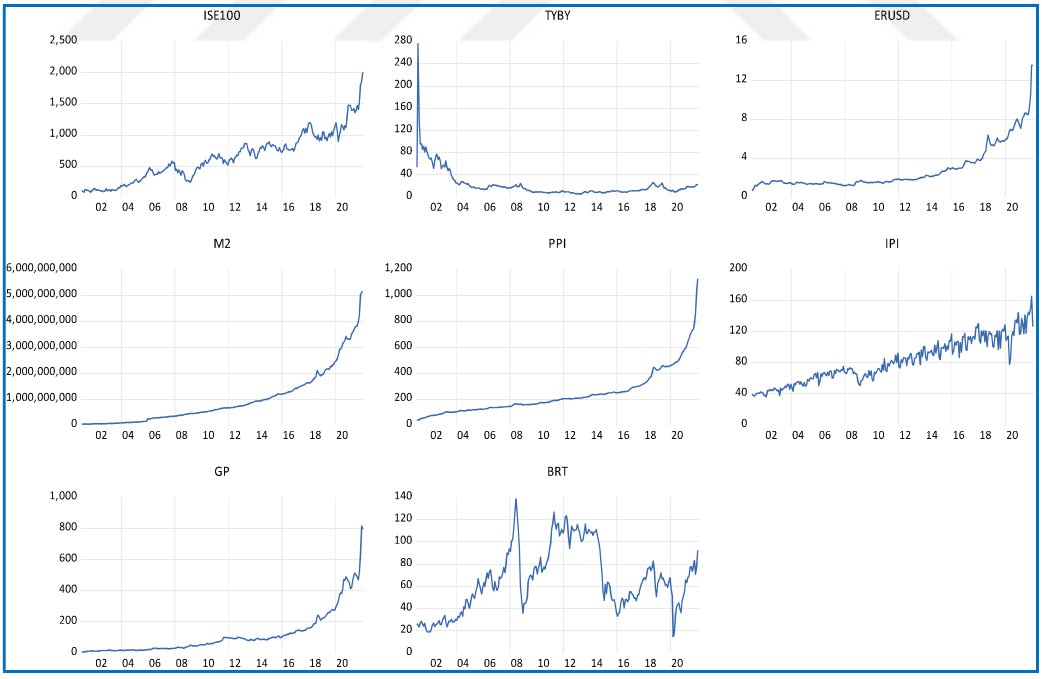


Figure 10.4. Original graphics of variables (level)

In order to determine the stationarity of a time series, inferences are made with the help of graphics. As can be seen in Figure 10.4, where the original graphics of the series of

the variables are given, it is seen that all the dependent and independent variables used in the regression models are not stationary, in other words, the series contain unit roots. Trend or seasonal mobility was observed in each variable.

Although there are many different unit root test techniques, the most popular ones are the ADF (1981) and PP (1988) unit root tests for determining whether financial time series are stationary. The stationarity test of the series was carried out by using three unit root test models: the fixed model, the fixed and trended model, and finally the none model without the constant term and trend. The results obtained from the stationarity analysis are shown in Table 10.2.

Table 10.2. Unit root test results at level

Variables	T-Statistics ADF	T-Statistics PP	Critical Value		
	Trend & Intercept	Trend & Intercept	1%	5%	10%
ISE100	-0.5926[0]	-0.6834	-3.9967	-3.4286	-3.1377
TYBY	-5.4044[15] *, **, ***	-6.4646 *, **, ***	-3.9967	-3.4286	-3.1377
ERUSD	4.7760[12]	4.3645	-3.9967	-3.4286	-3.1377
PPI	3.7943[6]	9.58833	-3.9967	-3.4286	-3.1377
IPI	-1.80939[12]	-10.6153*, **, ***	-3.9967	-3.4286	-3.1377
M2	7.86595[12]	10.3237	-3.9967	-3.4286	-3.1377
GP	7.2818 [10]	7.9621	-3.9967	-3.4286	-3.1377
BRT	-2.64021[0]	-2.02866	-3.9967	-3.4286	-3.1377

Note: All three models were tried in the level values and first order differences of the series and constant term and trend models were used in model selection as long as they were meaningful. The values in square brackets represent the appropriate lag length of the variables determined according to the SIC. The minimum lag length at which autocorrelation is removed was chosen. *, ** and *** indicate stationarity at 1%, 5% and 10% significance levels, respectively.

When the test statistics in Table 10.2 are examined, it is seen that all of the ADF and PP coefficients of the original series of the dependent and independent variables are not significant at the α=0.05 reliability level. It has been seen that the series where all variables (except the interest rate of the active bond) are not stationary at level contain a unit root at level. It is seen in Figure 10.4 above that the graphs of the series also support this result. In the analyses made, it was seen that when the first-order differences of the original series were taken, the series did not become stationary and the integration degrees were different. For this reason, stationarity analyses were performed again by taking the natural logarithms of the series.

If a time series becomes stationary when it is differentiated d times, the series is said to be d-integrated and is denoted as I(d). When the first differences of the series, except the interest of the active bond, are taken in the analysis, it is concluded that not all of them are integrated at the same level, the PPI series is integrated at I(2), and the other series are also integrated at the I(1) level. If the series is not stationary, the future prediction is not made by making generalizations. While the series are not singularly stationary, their linear combinations can be stationary. For this reason, after the ADF test, the Engle-Granger test was performed, and the ADF test was performed on the series consisting of the residues obtained from the regression and the null hypothesis was rejected. Therefore, according to Engle-Granger, the series are cointegrated. Therefore, the regression will be a regression showing a real relationship.

Table 10.3. Unit root test results at ln

Variables	T-Statistics ADF	T-Statistics PP	Critical Value		
	Trend & Intercept	Trend & Intercept	1%	5%	10%
lISE100	-2.7080[0]	-2.5823	-3.9951	-3.4279	-3.1373
TYBY	-5.4044[15] *, **, ***	-6.4646 *, **, ***	-3.9951	-3.4279	-3.1373
lERUSD	-0.0178[2]	-0.3388	-3.9951	-3.4279	-3.1373
lPPI	0.1964[2]	-1.5855	-3.9951	-3.4279	-3.1373
lIPI	-2.7569[12]	-9.5753 *, **, ***	-3.9951	-3.4279	-3.1373
lM2	-2.1832[0]	-2.1813	-3.9951	-3.4279	-3.1373
lGP	-3.0737[1]	-2.4055	-3.9951	-3.4279	-3.1373
lBRT	-2.8523[0]	-2.3922	-3.9951	-3.4279	-3.1373
dlISE100	-17.4022[0] *, **, ***	-17.5399*, **, ***	-3.9951	-3.4279	-3.1373
dlERUSD	-12.1062[1] *, **, ***	-9.8556*, **, ***	-3.9951	-3.4279	-3.1373
dlPPI	-7.2550[1] *, **, ***	-5.6983 *, **, ***	-3.9951	-3.4279	-3.1373
dlIPI	-6.1426[11] *, **, ***	-87.3719 *, **, ***	-3.9951	-3.4279	-3.1373
dlM2	-16.4110[0] *, **, ***	-16.4004 *, **, ***	-3.9951	-3.4279	-3.1373
dlGP	-11.4887[1] *, **, ***	-11.3133 *, **, ***	-3.9951	-3.4279	-3.1373
dlBRT	-13.7588[0] *, **, ***	-13.8688 *, **, ***	-3.9951	-3.4279	-3.1373

Note: All three models were tried in the level values and first order differences of the series and constant term and trend models were used in model selection as long as they were meaningful. The values in square brackets represent the appropriate lag length of the variables determined according to the SIC. d stands for first-order difference. The minimum lag length at which autocorrelation is removed was chosen. *, ** and *** indicate stationarity at 1%, 5% and 10% significance levels, respectively.

When the test statistics in Table 10.3 are examined, it is seen that all of the ADF and PP coefficients of the first-order differences of the dependent and independent variables are significant at the α=0.05 reliability level. It has been determined that all variables are

stationary at the I(1) level and their series do not contain unit roots. All series are I(1) co-integrated, and the Phillips-Perron unit root test results also confirm the ADF test results. For this reason, the first-order differences of the variables are taken into account for the estimation of the models.

Table 10.4. Correlation coefficient between the independent variables

	TYBY	BRT	ERUSD	GP	IPI	M2	PPI
TYBY	1	-0.03312	0.24945	0.1678	-0.00835	0.13426	0.36378
BRT	-0.03312	1	-0.09913	-0.0337	0.00648	-0.04088	0.13353
ERUSD	0.24945	-0.09913	1	0.733	0.03381	0.21283	0.60858
GP	0.16779	-0.03367	0.73304	1	-0.06753	0.26173	0.48518
IPI	-0.00835	0.00648	0.03380	-0.06753	1	0.09101	0.00243
M2	0.13426	-0.04088	0.21283	0.2617	0.09102	1	0.09195
PPI	0.36378	0.13353	0.60858	0.4852	0.00244	0.09195	1

As seen in Table 10.4, the correlation coefficients between the independent variables remain below 0.73 and mostly demonstrate weak linear relationships.

10.3. Logit ve Probit Models and Results

In the linear probability model, the conditional expected value of the dependent variable is the conditional probability of the event occurring given the independent variable values. The linear probability model can be estimated by the classical least squares method. However, the error term is not normally distributed, the error term has heteroscedasticity, the estimation values can fall outside the range of 0–1, and the error term generally has a low R^2 value. For this reason, Logit and Probit models have been developed as alternative models to get rid of these problems (Bengio et al., 2015, p. 584; Gujarati & Porter, 2009, pp. 552–553). The difference between the two models is that the Logit curve has a thicker tail or approaches the axis (0 or 1) more slowly than the Probit curve. The difference in the coefficients obtained from these models is due to the different functions used for probabilities (Gujarati & Porter, 2009, p. 571). The issue of which model will be selected in the application is presented according to the preference of the person who will use the model.

The parameters of the Logit and Probit models are estimated using the maximum likelihood method. In order to use this method, it is assumed that the probability distribution of the error term is in accordance with the normal distribution. In this case, when the series

are stationary, the normal distribution assumption regarding the error term is fulfilled. Before estimating the parameters of Logit and Probit regression models, which are qualitative dependent response models, a stationarity analysis for the variables used in the model is required.

Because the parameters of the Logit and Probit models cannot be estimated using the classical least squares method, the maximum likelihood method is used. In this method, the standard normal distribution which is the Z test statistic, is used for the statistical significance of the coefficients. In order to test the co-significance of the coefficients in the Logit and Probit models, Wald, Score (Lagrange Multiplier, LM), and Likelihood Ratio (LR) tests are used. In the analysis, the standard normal distribution Z statistic was used to test for the statistical significance of the coefficients, and the LR statistic was used to test the general significance of the models.

R^2, the coefficient of determination, which is used as a measure of goodness of fit or how explanatory the model is in classical linear regression models, is not a good measure for Logit and Probit models. Because these measures are not based on variance, they are not an accurate measure of whether the model is well explained or not, as they are calculated based on the change in the values of similarity ratios. The correct classification rate obtained from the classification table was also used to evaluate the goodness of fit of the models created. The obtained probability values are classified according to the determined cutoff point, and it is predicted which of the 0 or 1 values each unit will take. A value of 0.50 is usually taken for the cutoff point. If the probability value is greater than this value, the unit is assigned to group 0, and if it is less than this value, the unit is assigned to group 1.

In binary logit-probit models, the STATA program calculates the Pseudo R^2 value as a measure of goodness of fit, and the Eviews12 program calculates the McFadden R^2. Pseudo-R^2 was used in this study.

The coefficients obtained in the models made with Logit and Probit techniques cannot be interpreted directly as in the linear regression model. However, the sign of the examined coefficients indicates the direction of the relationship between the independent variables and the probability of the event occurring. If the sign of the coefficient is negative, there is an inverse relationship; if it is positive, there is a direct relationship.

In this study, both Logit and Probit model estimations were made using the macroeconomic variables in question, and for this, a threshold value was calculated for our independent variable, the BIST100 index. Thus, the transformation of the dependent variable into a binary variable with a qualitative response was achieved. The geometric mean of the BIST100 variable was taken and converted to a binomial to give a value of 0 for values below it and 1 for values above it. The geometric mean is a measure of central tendency that takes into account geometric differences between data, rather than arithmetic differences. The geometric mean, which covers all the observations in the data set and conforms to algebraic operations, allows the processing of relative numbers.

While investigating whether the variables used in this study are suitable for the Logit model and the Probit model, first of all, the dependent variable for each independent variable was added to the Logit and Probit models separately, and the relationship between the dependent variable and the independent variables was observed. The test results obtained in this context are given in Tables 10.5 and 10.6 below.

Table 10.5. Logit model compatibility results

	BinaryISE100				
Variables	**Coefficient**	**Prob**	**Log-Likelihood**	**LR Statistic**	**Prob (LR St.)**
TYBY	0.763828	0.0000	-94.087715	147.70	0.0000
LnERUSD	24294.1	0.0000	-82.945841	169.99	0.0000
LnPPI	553424.8	0.0000	-40.502838	254.87	0.0000
LnIPI	4.48e+07	0.0000	-44.480443	246.92	0.0000
LnM2	1.33e+50	0.0000	-38.382249	259.11	0.0000
LnGP	136.1367	0.0000	-41.043973	253.79	0.0000
LnBRT	12.36735	0.0000	-134.60854	66.66	0.0000

Table 10.6. Probit model compatibility results

	BinaryISE100				
Variable	**Coefficient**	**Prob**	**Log-Likelihood**	**LR Statistic**	**Prob (LR St.)**
TYBY	-0.160051	0.0000	-93.265235	149.35	0.0000
LnERUSD	5.747567	0.0000	-83.387448	169.10	0.0000
LnPPI	7.528884	0.0000	-39.850861	256.18	0.0000
LnIPI	9.398173	0.0000	-44.761852	246.36	0.0000
LnM2	66.07681	0.0000	-37.872111	260.13	0.0000
LnGP	2.776939	0.0000	-40.080625	255.72	0.0000
LnBRT	1.452791	0.0000	135.30463	65.27	0.0000

As can be seen in the Table 10.5 and Table 10.6 above showing the compatibility results, each variable is compatible with the dependent binary variable ISE100 for the Logit and Probit models. A 1% change in each variable causes the ISE100 to take a higher or lower value than its geometric mean. In other words, each variable in the model is an important variable that explains the dependent variable and is statistically significant. It was also in line with our expectations economically. Therefore, our variables are suitable for both the Logit and Probit models. However, since the changes in the interest rate and gold prices in the Probit model are in line with expectations, this model is considered more appropriate.

In this study, the Logit and Probit models were estimated with the highest likelihood method, taking into account the obtained data and evaluations, and the results are summarized in Table 10.7. It was seen that the variables used in the models were statistically significant according to the 1%, 5%, and 10% significance levels and were important variables affecting the dependent variable.

To test the overall significance of the Logit and Probit models, Hosmer-Lemeshow and Pearson goodness-of-fit tests, which are generally right-tailed tests, were performed. The Akaike Information Criterion (1973), which is an indicator of the goodness of fit of an estimated statistical model, and the Schwartz or Bayesian Information Criteria (1978), which enables model selection among a group of parametric models with different numbers of parameters, were also calculated. According to the these test, the model with the smallest AIC or BIC value is the best model. In addition, the correct classification rate was found to be 95.6 percent and 94.8 percent, respectively.

In the Logit and Probit models, since each of the coefficients obtained (log-odds) is a log-bet value, they cannot be interpreted directly as in linear regression models. However, the signs in the coefficients obtained indicate the direction of the relationship between the variables and the probability of the event occurring. If the coefficient obtained is negative, the relationship between the variables will be in the opposite direction; if it is positive, the relationship will be in the same direction.

Table 10.7. Results of the Logit model & the Probit model

	Logit Model		Probit Model	
Variables	**Odss Ratio**	**Prob**	**Coefficient**	**Prob**
c	8.1e-277	0.133***	-321.2098	0.066***
TYBY	0.540201	0.004*	-0.362669	0.002*
LnERUSD	13.79592	0.765***	2.213481	0.611***
LnPPI	9.424175	0.904***	1.408437	0.872***
LnIPI	2.43e+08	0.005*	10.29269	0.004*
LnM2	4.50e+85	0.231***	98.83031	0.145***
LnGP	0.000015	0.066***	-6.232234	0.056***
LnBRT	2.109222	0.872***	0.7311821	0.757***
Log-Likelihood	-22.420721		-22.222259	
LR Statistic	291.04		291.43	
Prob (LR St.)	0.0000		0.0000	
Pseudo R2	0.8665		0.8677	
Hosmer-Lemeshow chi2 (Prob>chi2)	3.52 (0.8974)		0.50 (0.9999)	
Pearson chi2 (Prob>chi2)	64.49 (1.0000)		53.70 (1.0000)	
AIC	60.84144		60.44452	
BIC	89.10856		88.71163	
Correct Classification Rate	95.65%		94.86%	

Note: *, **, ***; represent 1%, 5%, and 10% significance levels respectively.

As seen in Table 10.7, the coefficients obtained in the Logit and Probit models are statistically significant at 5% and 10% significance levels, respectively. When the coefficients of the variables were evaluated one by one, the results were obtained in a way that met our expectations. While an inverse relationship was expected with the interest rate of the active bond and gold prices, it was observed that this expectation was met and that there was a positive relationship between other variables, again meeting the expectations. As the interest rate of the active bond increases, the return will also increase, so the demand of the investor shifts from the stock market to this instrument. The investor will show the same reaction when gold prices increase and will choose gold as a safe haven. In other words, there is a negative relationship between the stock market, the two-year bond rate, and gold prices. It is seen that a 1% increase or decrease in other macroeconomic variables included in the model causes the dependent variable to take higher and lower values than the average, depending on the coefficients of the variables, by increasing or decreasing the dependent variable at different rates.

It is seen that these results are confirmed by four separate tests calculated for models, showing whether the model is completely significant or not. Since the probability values of Hosmer-Lemeshow and Pearson tests are higher than 0.05, our models are statistically significant. Again, AIC and BIC criteria are also undervalued. Accurate classification rates were found to be 95.65% for the logit model and 94.86% for the probit model. In addition, considering the LR test statistic and the probability values for this test statistic, it is seen that all coefficients in both the Logit and Probit models are significant together. Also, the Pseudo R^2 values, which are the output of the STATA program and show the significance of the whole model, were found to be 0.865 for the Logit model and 0.866 for the Probit model. The correct classification rate results for the two models are shown in Table 10.8 and Table 10.9.

Table 10.8. Correct classification rate table for the Logit model

Clasisified	D	F	Total
Positive	151	5	156
Negative	6	91	97
Total	157	96	253
Correctly Classified(%)	96.79	93.81	95.65

Table 10.9. Correct classificaiton rate table for the Probit

Clasisified	D	F	Total
Positive	150	6	156
Negative	7	90	97
Total	157	96	253
Correctly Classified(%)	96.15	92.78	94.86

In the Probit model, unlike the Logit model, to meet our economic expectations, gold prices appear as a variable that adversely affects the stock market index as well as the two-year bond interest. The variable is also statistically significant. Gold is an important investment tool all over the world and has been seen as a safe haven for investors. It is evident that many investors are attracted to this area due to the continuous increase in gold prices in recent years. There is no doubt that speculative attacks in the stock market, interventions in the capital market, the lack of transparency and reliability of the capital market, as well as speculative news on this subject, have a negative impact on all investors. Since all of these factors make small investors nervous, they cause investors to turn to safer

investment instruments, and the failure to transform savings into investments leads to a contraction in the economy. When this mechanism, which has a spiral structure, operates in the opposite direction, it affects social welfare negatively.

Comparing the performance of the Logit and Probit models, the compatibility test values of both models are very close to each other, but the main difference between the two models is that the logistic distribution has a slightly thicker tail. However, there is practically no reason to prefer one over the other. One is preferred over the other because of its comparative mathematical simplicity. The correct classification rate varies from 95.65% for the Logit model to 94.86% for the Probit model, which is quite low at 8 per thousand. However, the variables in the Probit model were estimated to meet economic expectations. The statistical tests of the variables included in the model are also an indication of this. In addition, the Pseudo R^2 values of both models were over 86%.

10.4. Deep Learning Models and Analyses

In this study, it was investigated how the DL model, which is a method based on the working mechanism of AI technologies and is one of the ANNs, should be used to measure the power of explaining the Istanbul Stock Exchange index values using the selected macroeconomic variables. It has been shown in many empicial studies that AI applications, especially in estimating non-linear time series, give as good or better results than well-known traditional econometric methods, and because of this success, they are used as good estimation tools (Akel & Karacameydan, 2018; Jing et al., 2021; G. Zhang et al., 1998, p. 40).

10.4.1. Preprocessing of Data Used in the Study

The preprocessing stage of the data means the normalization of the data groups consisting of training and test data in accordance with the established DL networks. Normalization is one of the most important factors that determine the success of the DL model. In addition, the entry of the data into the system in raw or processed form should be determined according to the researcher's purpose for the problem (Kaastra & Boyd, 1996, p. 222). Data normalization is the process of compressing the output of the processing

elements into the range of [0, 1] or [-1, 1] with the activation functions used in the hidden layer and the output layer (Altan, 2008; Gomes et al., 2016; G. Zhang et al., 1998, p. 49).

The data used in this study consists of non-linear, seasonally unadjusted, and trend-containing time series in which seven macroeconomic variables and Borsa Istanbul Index values are taken as independent variables. For this reason, the series are normalized in the range of [-1, 1] using the equation $(x_i = (x_i - \mu)/\sigma)$. The graphs of the normalized independent variable series obtained are given in Figue 10.5 below.

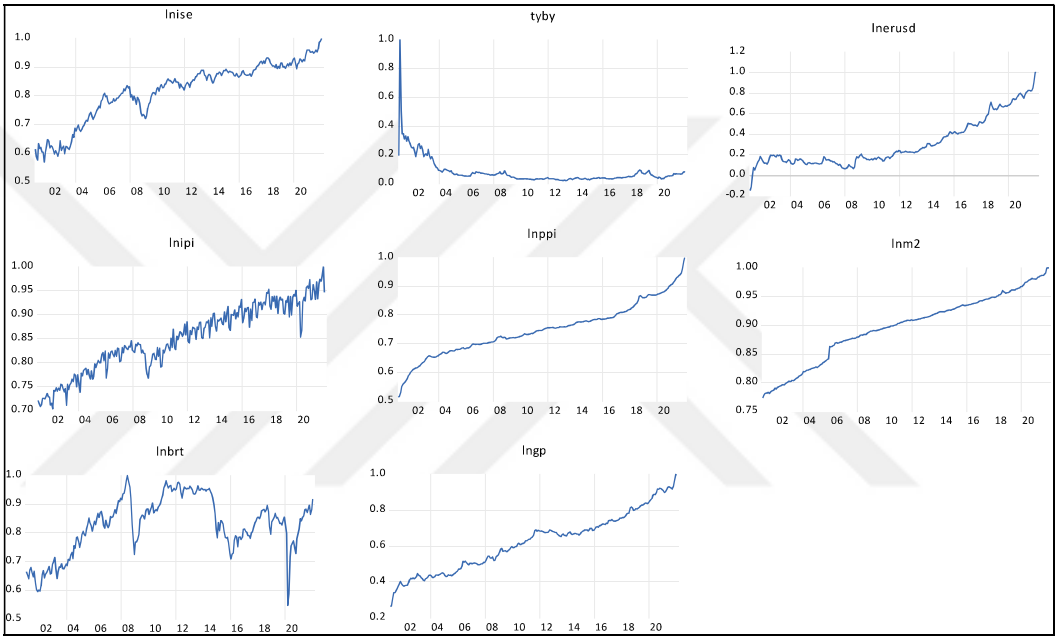


Figure 10.5. Graphs of variables with normalization

10.4.2. Deep Learning Network Architecture Used in the Study

Significant variables obtained from the meaningful models established as a result of the analyses made with Logit and Probit regression modeling methods; Dollar Rate, Industrial Production, Domestic Consumer Price Index, Active Bond Interest Rate, Gold Prices, Money Supply and Brent Oil Price are used as input variables in the established deep learning neural network models.

With the DL method based on working of ANNs, many different models can be produced using different criteria. The performances of the models created are depends on many parameters, such as the number of inputs, the number of input layers, the number of

hidden layers, activation functions, the learning method used, the learning rate, the number of mini-batches, the number of epochs, weight values, memorization value, and even the speed of the processor used in the analysis. It is closely related to critical components like capacity. The most widely used ANN model in the literature is the multilayer perceptron and the most widely used DL model is LSTM (W. Chen et al., 2017; Cheng et al., 2022a; Jing et al., 2021; Sarker, 2021).

DL, one of the applications of AI that provides the opportunity to design deep neural networks with pre-trained models, is a method used on image, time series, and text data. Convolutional neural networks (ConvNets, CNN) and Long Short-Term Memory (LSTM) networks can be used to classify and regress data. An LSTM network is a recurrent neural network (RNN) that processes input data by looping the time steps and updating the network state. The network contains information remembered at all previous time steps, and using data from these previous time steps, it predicts the value a time series might take in the next period. This network learns to predict the value of the next period. In this method, there are two estimation methods: open and closed loop estimation.

A standard reversible neural network architecture (RNN) contains a single layer, while LSTM network consists of four communicating layers. The basic core parts in a LSTM network structure are the sequence input layer and the LSTM layer. In the sequence input layer, the inputs are serial or time series data in the network. The LSTM layer learns the long-term dependencies of serial data between time steps. Starting with the serial input layer, the network architecture consists of the LSTM layer, the fully dependent layer and the output layer.

In this study, an AI architecture will be created by using a DL algorithm, which is the last point reached by ML algorithms from AI technologies. Convolutional neural networks are used in various combinations to form the foundation of modern DL architectures. The DL architecture has layers of input, convolution, activation, pooling, memory, full coupling, and classification. In the convolution layer, filters (kernels) are used to circulate the data in the input layer horizontally and vertically.

There are activation functions such as ReLu, Sigmoid, and Step (Step), which are the most commonly used in the literature, in the activation layer. The ReLu activation function

is used in this study. In the pooling layer, the maximum pooling function, which is the most used in the literature, is the maximum pooling, minimum pooling, and average pooling value functions. In the dropout layer, AI algorithms function to make the network forget the data in order to prevent the data from being memorized during training. For this reason, the dropout layer is used in network architecture.

Since the model we created is an LSTM model, the full connection (FullConnected, FC) layer and classification layers in the structure of this model are also used. Thus, the data is transformed into a one-dimensional vector and passed to the classification layer. As in ANNs, in the classification layer, which is the last layer in DL algorithms, it evaluates the data from the previous full link layer and creates the outputs of the network. SoftMax classifier, which is a probabilistic calculation method, is generally used in this layer. Thus, values in the range 0–1 are generated for each possible class (Metlek & Çetiner, 2021, pp. 4–18).

For an LSTM model, there must be a number of input neurons that match the numbers of the input data. In the created LSTM network, seven input neurons, 200 hidden units, 50 fully connected layers units, 50% dropout, and an output layer with the same layer as input data were created. In the training phase, several tries later, using adam and sgdm optimisation functions, 100 cycles per iteration and minibatch size 20 with an initial learning rate of 0.01 and gradient threshold 1 were used. Each cycle, the data is shuffled. The goal here is to prevent the network from memorizing. The general structure of the DL prediction models created according to the parameters used is given in Figure 10.6.

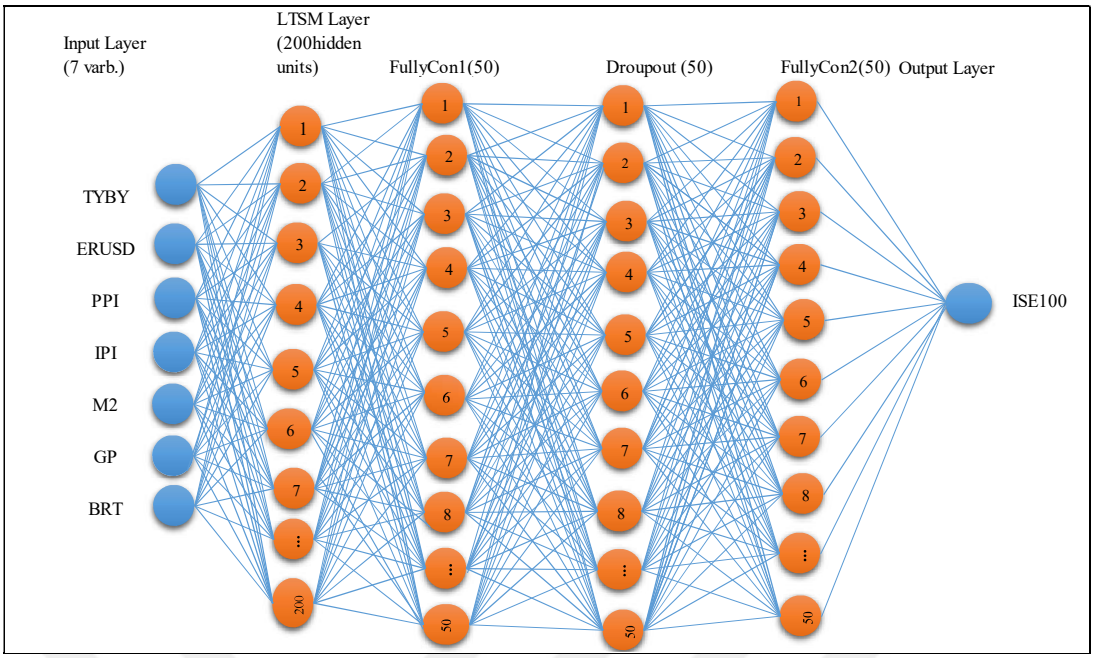


Figure 10.6. Simple architecture of deep learning network

Windows 10 Pro 64-bit is installed on hardware that includes an Intel(R) Core (TM) i7-10510U CPU @ 1.80 GHz (8 CPUs), 2.30 GHz processor, 16 GB of RAM, and an Intel(R) UHD Graphics graphics card. It was developed in the MATLAB R2021b environment with the operating system.

The parameters that make up the AI architecture of DL models are given in Table 10.10.

Table 10.10. Deep learning network architecture parameters

ANN type	Deep Learning LSTM
Number of input layer neurons	7
Number of output layer neurons	1
Number of fullyconnected layers	2
Number of hidden layer neurons	200
Optimization function	adam, sgdm
Activation function used in hidden layer	ReLU
Activation function used in the output layer	Softmax
Optimization algorithm	Gradient Descent
Scaling method used	Normalisation
Inıtial learning rate	0.01
Fullyconnected layer	50
Droupout	50
Gradient threshold	1
Epoch	100
Minibatchsize	20

10.4.3. Training and Test Phase of the Network

While the data used for the solution of the established DL creates the training set, the test set consists of data that the network did not see during the training process. The test set tests whether the network learns well. It has been seen in the literature that the training test set data is determined at various rates, such as 90%-10%, 80%-20%, or 70%-30%. However, the most preferred rate is 70%–30%, although it varies according to the problem (Chong et al., 2017a, pp. 187–205; Jing et al., 2021, pp. 5–8; G. Zhang et al., 1998, p. 50). In some studies, the data set is divided into three groups as training, validation and test sets (Kaastra & Boyd, 1996, p. 220). In this study, the data set was divided into training and test sets at different rates, and models were created for each ratio by using the *adam* and *sgdm* optimisation functions. Analyses were made with models created according to these parameters. The findings are presented in the tables below for comparison.

The data set is divided into 70% training and 30% test sets, which is the most widely used in the literature. The first 177 observation values (2001:01–2015:08 period) were determined as the training set, and the last 76 observation values (2015:09–2021:02 period) were determined as the test set. Another reason for determining the test set in this range is to show whether the global crisis and its seasonal effects, which started in 2008 and showed their effects until the beginning of 2009, could be comprehended by the model.

As an indicator of whether the established DL model is a good predictor or not, the corrected R^2 and performance criteria, whose equations for calculation methods are given in section 6.12 and which are frequently used in performance measurements of MLPs, are used. These metrics are performance metrics such as R^2, HKO-Mean Squared Errors, HKT-Sum Squared Errors, and HKO-Root Mean Squared Errors. The most appropriate network structure was obtained by comparing the findings in the training and testing phases, taking into account these performance criteria.

10.4.4. Deep Learning Analysis Results

In a multi-layered AI and DL model, determining parameters such as the number of hidden layers, the number of neurons in the hidden layer, the activation functions in the hidden layer and the output layer, and learning algorithms such as the learning rate and momentum coefficient are very important for the performance of the network. However, since there is no precise method for determining these parameters, they were determined by trial and error (Jing et al., 2021, pp. 5–8; Kaastra & Boyd, 1996, pp. 220–224; C.-C. Lin et al., 2018, pp. 756–764; Wang et al., 2018, pp. 1–17).

The MATLAB R2021b computer language's algorithms were used to build multilayer DL models using the settings listed in Table 10.7. The best performing models were selected from the models created by changing the parameters. These models were created using the *adam* and s*gdm* optimization algorithms, respectively, and by changing the training and test ratios of the data without normalizing the data set used in the study. In the next step, the data were normalized, and the same process was repeated. The analysis results of the fourteen DL models obtained are summarized in the tables below. The aim here is to show the effectiveness of normalization on model performance.

Table 10.11. Simulation results for *adam*

adam		Train		Test	
Train (%)	**Test (%)**	**MSE**	**RMSE**	**MSE**	**RMSE**
20	80	0.1267	0.3559	0.0851	0.2917
30	70	0.0466	0.2159	0.3496	0.5913
40	60	0.0317	0.1782	0.1555	0.3943
50	50	0.2027	0.4502	0.4081	0.6388
60	40	0.0588	0.2425	0.0749	0.2738
70	30	0.1245	0.3528	0.3409	0.5839
80	20	0.1379	0.3714	0.3697	0.6081

The effectiveness of each of the seven DL models developed was evaluated using the MSE and RMSE performance metrics. According to the results in Table 10.11, DL networks were created with an unnormalized data set using the man optimization algorithm, and as expected, the performance values differed according to the data set partitioning. The training set reached the smallest MSE value when it was 40%. However, the test set did not reach the global minimum, but the performance indicator values were high.

Table 10.12. Simulation results for *sgdm*

sgdm		Train		Test	
Train (%)	**Test (%)**	**MSE**	**RMSE**	**MSE**	**RMSE**
20	80	0.0481	0.2195	0.0361	0.1901
30	70	0.0279	0.1672	0.0992	0.3150
40	60	0.3653	0.6044	0.5380	0.7334
50	50	0.0287	0.1695	0.0830	0.2881
60	40	0.0334	0.1829	0.1357	0.3684
70	30	0.1222	0.3496	0.1975	0.4441
80	20	0.0547	0.2339	0.1981	0.4451

As can be seen in Table 10.12, seven DL network models were created using the *sgdm* optimization algorithm with the unnormalized data set. The training set reached the smallest MSE value when it was 40%. However, the test set did not reach the global minimum, but the performance indicator values were high. According to these two tables, the performance values of DL network models are high even if the optimization algorithm is changed, if the analysis is performed without normalizing the series.

Table 10.13 below provides a summary of the outcomes of the examination of DL models, which were developed by normalizing the research's data set in accordance with the normal distribution and employing the *adam* and *sgdm* optimization algorithms.

Table 10.13. Deep learning results

		Train		Test	
Train (70%)	**Test (30%)**	**MSE**	**RMSE**	**MSE**	**RMSE**
adam		**0.0440**	**0.2099**	**0.1396**	**0.3736**
sgdm		**0.0448**	**0.2117**	**0.1444**	**0.3800**

As can be seen from the tables above, the system that gives the best results is a linear system, and the best results are achieved by normalization.

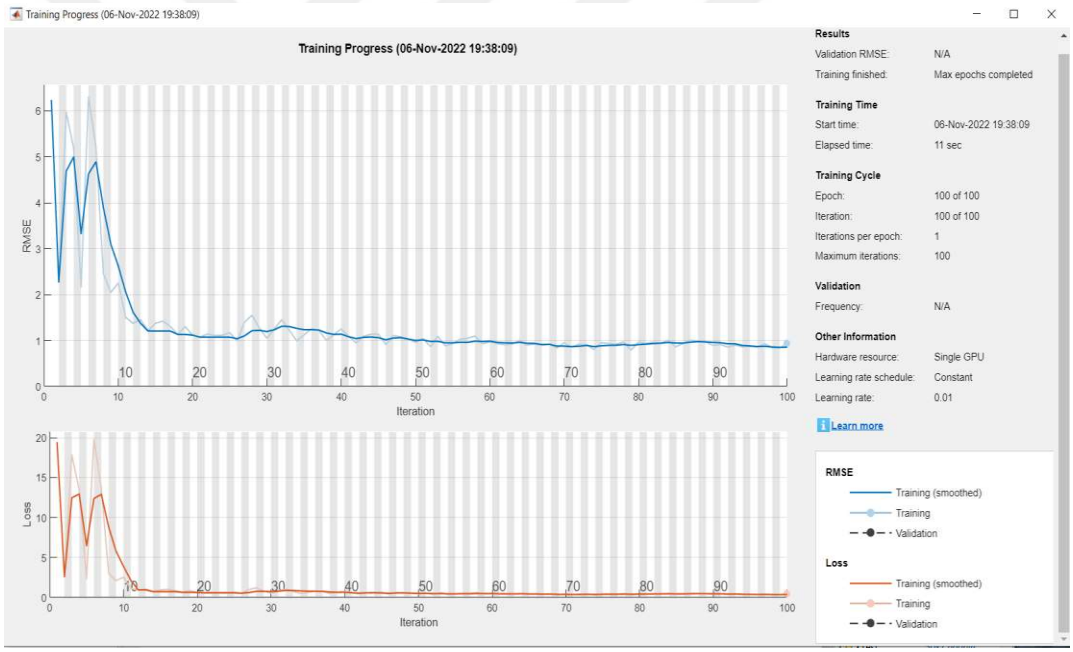


Figure 10.7. Training progress with *adam*

Figure 10.6 shows how effective learning is when the optimization algorithm determines the *adam* algorithm and the number of epoch as 100 during the training phase of the obtained six layers LSTM DL model (while taking into account the other determined parameters). As the number of epoch increases, the error between the original series gradually decreases and progresses towards the global minimum.

The graphs in Figure 10.8 below show the training and test curve graphs obtained from the analysis using the *adam* optimization algorithm. It is seen that the complexity of the system increases as the degree of mathematical expressions of the regression models chosen to model the training and test curves in the graphs increases, and thus this turns into an advantage. Thus, an obscure system becomes identifiable with a simpler expression.

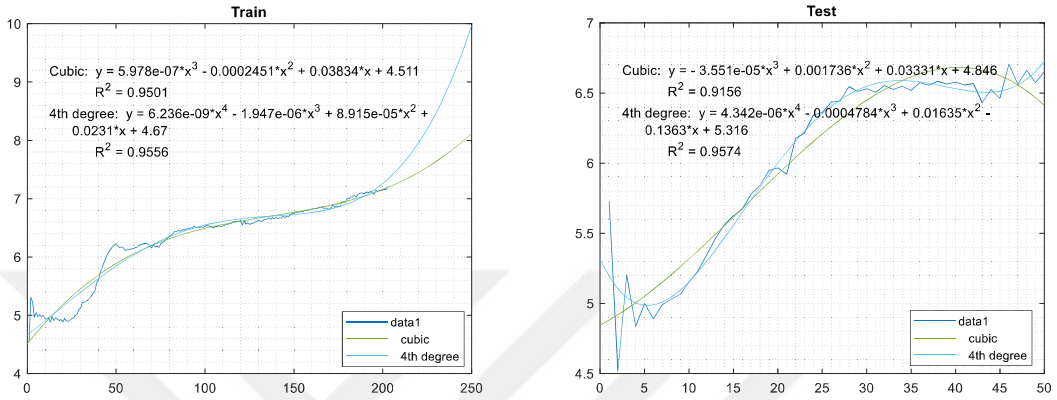


Figure 10.8. Train and test results with *adam*

It can be seen that the two fully connected layered LSTM models aid in the simplification of complex problems. It is also seen that the mathematical equation of the LSTM models has a fourth-order function, and the R^2 value is quite high at 0.95. This shows that the model is completely significant and the explanatory power of the variables is quite high. In other words, the network showed a very successful performance during the training and testing phase. The model is a good forecasting model, and the margin of error in the estimates was 0.0440.

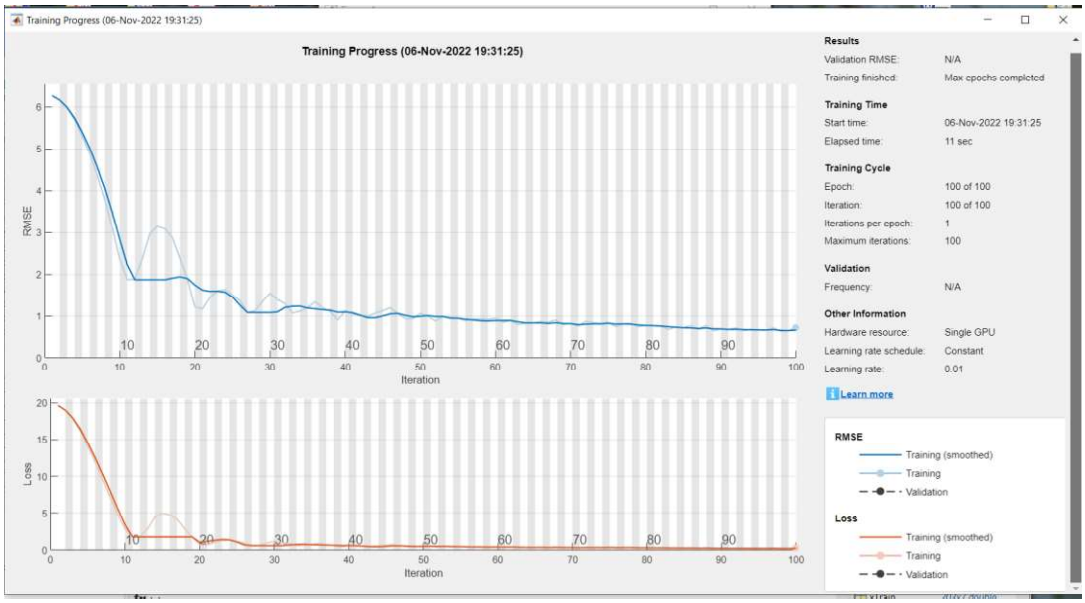


Figure 10.9. Training progress with sgdm

Figure 10.9 shows how effective learning is when the optimization algorithm determines the *sgdm* algorithm and the number of epoch as 100 during the training phase of the obtained six layers LSTM DL model (while taking into account the other determined parameters). As the number of epoch increases, the error between the original series gradually decreases and progresses towards the global minimum.

The graphs in Figure 10.10 below show the training and test curve graphs obtained from the analysis using the *sgdm* optimization algorithm. It is seen that the complexity of the system increases as the degree of mathematical expressions of the regression models chosen to model the training and test curves in the graphs increases, and thus this turns into an advantage. Thus, an obscure system becomes identifiable with a simpler expression.

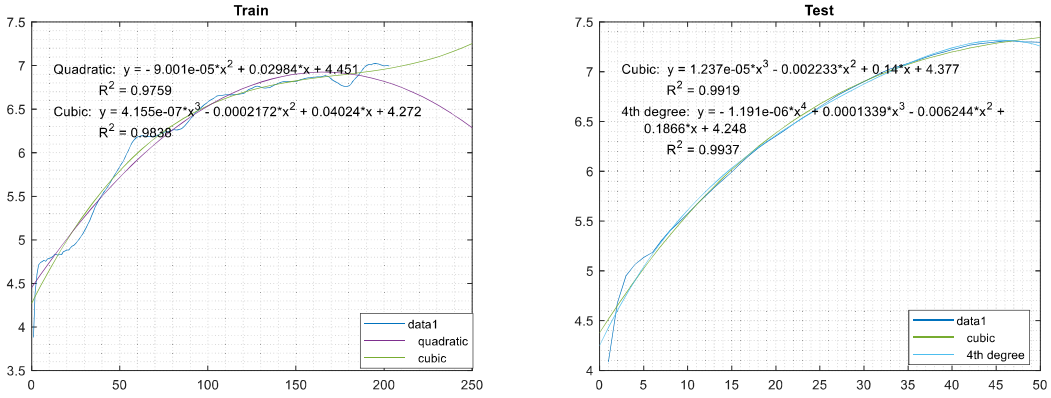


Figure 10.10. Train and test results with *sgdm*

As can be seen in the tables and figures given above, a deep neural network architecture with the best performance was created as a result of the trials using various parameters. It is an optimization process based on the learning process in DL applications. In DL models, the sgdm (stochastic gradient descent with momentum) optimization algorithm is used hypothetically. However, in this study, the financial time series was modeled both unnormalized and normalized, using both optimization algorithms. Model performances are compared. It has been seen that the optimization algorithm, which is slower than the global minimum and has a lower margin of error, is sgdm and predicts the models with the minimum error. According to the results obtained, it was seen that the selected macroeconomic variables were important variables explaining the change in the ISE100 index values and were statistically significant.

The correctness and sufficiency of the learning of an artificial neural network is tested with the test set. According to Figure 10.8 and Figure 10.10, which give the training and test phase results, the trainig and test phase has been successfully completed. As can be seen from these graphs, which show the estimation values alongside the real values, the estimation errors are quite low and gradually approach zero. In order to measure the prediction success of the deep neural network models created after the training and testing phases, the findings in Table 10.14 were obtained by calculating R^2, mean square error, and root mean square error.

Table 10.14. Deep learning performance results

	Train			***Test***		
	R^2	**MSE**	**RMSE**	**R^2**	**MSE**	**RMSE**
adam	0.9501	0.0440	0,2099	0,9156	0,1396	0,3736
sgmd	0,9759	0,0448	0,2117	0,9919	0,1444	0,3800

When the findings in Table 10.14 and the graphs showing the results of the training and testing phases are evaluated together, it is seen that the error margins of the networks in the training and testing phases are quite low and approach zero. In other words, it is seen that multilayer LSTM deep neural network models perform quite successfully in the in-sample and out-of-sample periods. During the in-sample period, the R^2 value of the multilayer deep neural network model developed to estimate the ISE100 index value was 97.59%, while the MSE value was at the lowest level at 0.0448.

Out-of-sample predictions are of greater importance in the comparison of prediction accuracies, as they consist of predictions from new data that are not used in model prediction. Considering this situation, it is seen that the created multilayer deep neural network model predicts the ISE100 index value most successfully, and the R^2 value of the model is realized at a very high level of 99.19%. This model's MSE value was at its lowest with 0.1444. In general, it is seen that there is a strong and positive relationship between the independent variables and the ISE100 value. It is seen that this relationship between the variables is over 50% in all variables, and especially the MSE is at a very low level.

Although the variables were not seasonally adjusted and dummy variables were not used during the creation of the models, the deep neural network models performed quite successfully. The deep neural network models developed in general have a good understanding of the effects of the economic crisis in Turkey in 2001 and the global economic crisis that emerged in the USA in mid-2007 and adversely affected all world economies in 2008 on macroeconomic variables and the ISE100 index.

The findings in Table 10.14 and the training and testing phase graphics given above show that the deep neural network models created gave very good results and that the networks successfully completed the training and testing phases. The positive performance indicators show that deep neural network models can be used for prediction. However, the prediction results will be presented comparatively in the last section.

10.5. Comparison of Model Results

Time series are numerical quantities in which the values of the variables are observed consecutively from one period to the next (Sevüktekin & Nargeleçekenler, 2010, p. 41). Statistically, the economic time series has a structure consisting of trend, seasonality, cyclical, and random movements (Kennedy, 1998, p. 288). Therefore, they are not stationary. It is very important for decision makers to be able to predict the future performance and behavior of economic time series due to uncertainties in the economy. In this case, the model established for a time series based on the observed values is expected to have a performance that can predict the possible values that the series may take in the future (one day, one month,

or one year later). In other words, the forecasting performance of the model is expected to be high.

Forecasting modeling is of great importance and widely used in the field of economics, as well as in many other fields. Forecasting modeling plays an important role in determining the future behavior of both governments and producers, consumers, and finance sectors, which are called micro-decision makers. It is of great importance that good forecasts can be made, especially when making the right decisions in the field of finance. Simply put, good foresight leads to good decisions.

In many empirical studies in the literature, econometric models have been used as predictive models, and AI and deep neural network models have been used in comparison with econometric models in recent years due to their high prediction performance. This is because AI models can outperform and even outperform classical models in predicting nonlinear, seasonal, and trending time series.

In this part of the study, the comparison of Logit, Probit, and DL methods has been made. The comparison was made based on the estimation performances of the methods used. The performances of the Logit and Probit models with the best performance were examined, and the prediction results obtained from the regression analysis are presented comparatively in Tables 10.15 and Table 10.16.

Table 10.15. Logit model performance

Clasisified	D	F	Total
Positive	151	5	156
Negative	6	91	97
Total	157	96	253
Correctly Classified (%)	96.79	93.81	95.65
Sensitivity (%)			96.18
Specifivity (%)			94.79
Error Rate (%)			4.35

Table 10.16. Probit model performance

Clasisified	D	F	Total
Positive	150	6	156
Negative	7	90	97
Total	157	96	253
Correctly Classified (%)	96.15	92.78	94.86
Sensitivity (%)			95.54
Specifivity (%)			93.75
Error Rate (%)			5.14

In this part of the study, the comparison of Logit, Probit and DL methods has been made. The comparison was made based on the estimation performances of the methods used. The performances of the Logit and Probit models with the best performance were examined, and the prediction results obtained from the regression analysis are presented comparatively in Tables 10.15 and Table 10.16.

Table 10.17. DL model performance

	Accuracy	Loss	Total(%)
Accuracy (%)	99.61	0	100
Loss (%)	0	0.39	100
Total (%)	99.61	0.39	100

The DL model learned the in-sample period series correctly at a rate of 98.81% with a loss of 1.19% during the training phase. In other words, the rate at which the model learned about the crises experienced in Turkey in 2000 and 2001 and the reflections of the 2008 global crisis is 98.81%.

As can be seen from Table 10.17, the DL model correctly learned the in-sample period series and predicted 100% of the periods in the test data set for the out-of-sample prediction. After 2015, the effects of the developments in the markets due to the political events in July 2016 in Turkey and the price increases due to the COVID-19 pandemic in 2019, and their reflections on the market indicators, have been estimated at 100%. In other words, the validation rate of the model is 100%.

Although the validation rates of the created models are high, the validation percentage of the DL model is 100%. The margin of error is zero. In other models, the margin of error is between 4.35% and 5.14%.

11. ARGUMENT

After the rapid developments in computer technologies, the use of AI technologies has become widespread in the field of finance, as well as in many other fields. The reason for the increase in interest in AI is that artificial intelligence technologies' ability to comprehend the problem, read the data correctly, and solve it quickly is quite advanced. Because of these characteristics and following its success in many fields such as health, defense, and engineering, it has begun to be used in the field of finance since the great financial crisis of 2008. Especially for policymakers and investors, it has started to turn towards methods that will ensure the elimination of financial uncertainties.

The effectiveness of AI has been proven once again in this study, in which the DL method, one of the new generation of AI technologies, was tested alongside traditional statistical and econometric methods such as the Logit and Probit regression modeling techniques. In this study, it was discovered that AI technologies outperform classical econometric methods in the analysis of financial time series with a dynamic and volatile structure. Changes in the seven selected macroeconomic variables will affect the future values of the ISE100 index in which direction and strength. In fact, studies have shown that even new modulation techniques developed by AI technologies outperform each other in estimating each other. The new formations that have occurred all over the world in recent years, the ongoing economic crises since 2008, the COVID-19 pandemic that started in 2020, the military crises between Russia and Ukraine in 2022, have actually revealed the necessity of evaluating the world as a whole, not as countries and borders.

These developments have demonstrated that a positive or negative situation in any part of the world has reached global proportions, affecting all countries. This situation has made all the components that countries face in maintaining or increasing the level of social welfare uncertain. In these days when there are many discussions about whether AI applications will bring benefit or harm in the future, it is obvious that change is inevitable and it is clear that countries should be open to new technological structures or methods in order to ensure their continuity and to predict and structure the future in every field.

12. CONCLUSION AND PROPOSAL

ANNs are computer software developed to imitate the working mechanism of the human brain and to realize the brain's abilities, such as learning, remembering, and generating new information through generalization, without any assistance. Due to this feature, AI applicaitons are successfully applied in many fields, such as industry, business, finance, education, the military, defense, and health. One of the application areas of AI is solving problems of future prediction.

ANNs have been able to obtain very successful results in non-linear time series analysis due to their non-linear structure. In this regard, AI has become a frequently used method in many areas of finance. In the field of finance, although AI models are applied in many studies examining issues such as predicting financial crises, determining the direction of exchange rates, general price levels, and especially measuring and selecting the performance of stocks, there are limited studies on the estimation of values of stocks market index.

Linear stochastic regression models can have an advantage over other models if they can understand and explain important details between variables. However, linear models are insufficient when the relationship between the variables in the studied problem is not linear. At this point, AI models can make successful predictions when the appropriate network structure for nonlinear relationships is determined.

In this study, it is aimed to estimate the value of the ISE100 index for the period January 2001–January 2022 by using the DL modeling technique, which is one of the non-linear estimation models, after the variables to be used as input variables are determined by using the Logit and Probit methods. General information about the size and development of stock markets in Turkey and around the world, or in other words, stock markets, is given. Then, AI technologies, such as ANNs, and finally DL modeling, which is ML, were examined in detail. The DL model was designed, and the ISE100 value was modeled using the appropriate network architectures and seven macroeconomic variables. Estimated models were evaluated within themselves and performance comparisons were made by performing Logit and Probit analyses. In Logit and Probit analysis, the ISE100 index was

used as the dependent variable, with seven independent variables: the Dollar Rate (TL/$), the Money Supply, the Producer Price Index, the Industrial Production Index, Gold Prices (TL/Gr), the Active Bond Interest Rate, and the Brent Oil Price.

The estimated DL models showed a consistent structure and good predictive performance. In the prediction comparison made with Logit and Probit models, it was concluded that the DL technique performed better than this method. In addition, it is the most striking point that the method of DL is more successful than the Logit and Probit methods in the prediction of financial crises.

When the Logit and Probit model results are examined, it is seen that the models created are the best models. Since the probability values of Hosmer and Lemeshow and Pearson goodness of fit tests were higher than 0.05, the models were statistically completely significant, and the lowest AIC and BIC values and the highest Pseudo R^2 values were found to be 0.865 for the Logit model and 0.866 for the Probit model. The accurate classification rates were found to be 95.65% and 94.86% for the Logit and Probit models, respectively. It has been concluded that the estimation results of the Logit and Probit models are very close to each other. According to the Logit and Probit analysis results, the coefficients obtained for each independent variable are significant, and the coefficient sizes and test statistics values are close to each other. It has been observed that the Probit model gives better results than the Logit model.

The fact that the predictions made with Logit and Probit analyses show great deviations from the real values and the better prediction performance of the network even though the seasonal effects are not reflected in the deep neural networks cause a generalization to be made that the nonlinear modeling for financial variables, that is, the deep neural network method, is more effective.

There are very few studies that use deep neural networks or DL models to predict financial markets and stock markets around the world. The method of ANNs has been used in several studies to predict the direction of the Turkish stock market, but the DL model has not been used. In this study, it is thought that the use of Logit and Probit analysis and the DL model to predict the direction of the ISE100 index and the large number and variety of macroeconomic variables included in the analysis add originality to the study.

Today, developments in the fields of communication technology and finance, the global economic crises that started in the US loan market and affected the national markets of all the countries of the world, and the socio-economic processes experienced after epidemics such as the COVID-19 pandemic have shown that international capital markets have gained a national capital market identity. It is seen that the models created with AI techniques, which have the ability to make successful predictions and model non-linear relationships and thus explain the changes in the capital market, are an effective estimation tool for investors who want to hedge risk and gain returns in such a wide market. Because of these features, artificial neural network techniques are becoming an increasingly important problem-solving tool in the field of finance.

Based on this study, the following can be suggested as future research topics for the ISE100 index: While expanding the basket of macroeconomic variables that affect the ISE100 value, the stock being studied can be limited. In other words, a similar analysis can be made for certain stocks in Turkey. The analysis of the value of the Turkish stock market over a longer period by comparing other econometric models with AI methods and/or using hybrid models can be another subject of study.

It has been determined that artificial intelligence technologies provide very successful optimizations compared to traditional methods in terms of how investors or decision makers will take a position in the capital markets, which investment instruments they will prefer, or which instruments will be included in the portfolio basket. Therefore, artificial intelligence technologies emerge as a very successful method in the field of finance. These models, which are created by learning from the data with the lowest error, will take their place in the literature as the best optimization and prediction models.

13. REFERENCES

Abraham, B., & Malik, H. J. (1973). Multivariate Logistic Distributions. *CRC Press.*, *1*(3), 588–590. https://doi.org/10.1214/aos/1176342430

Aggarwal, M. (2020). Probit and Nested Logit Models Based on Fuzzy Measure. *Iranıan Journal of Fuzzy Systems*, *17*(2), 169–181.

Akel, V., & Karacameydan, F. (2018). Yatırım Fonları Net Varlık Değerlerinin Yapay Sinir Ağları Yöntemi ile Tahmin Edilmesi. *Anadolu University Journal of Social Sciences*, *12*(2), 87–106.

Akkaya, M., & Kantar, L. (2019). Bankacılık Kırılganlık Endeksinin Logit-Probit Model İle Analizi: Türkiye Uygulaması. *Finans Politik& Ekonomik Yorumlar*, *56*(650), 131–145.

Aladag, C. H., Egrioglu, E., & Kadilar, C. (2009). Forecasting Nonlinear Time Series with a Hybrid Methodology. *Applied Mathematics Letters*, *22*(9), 1467–1470.

Alaloul, W. S., & Qureshi, A. H. (2020). Data Processing Using Artificial Neural Networks. In *Data Processing Using Artificial Neural Networks*. IntechOpen.

Altan, Ş. (2008). Döviz Kuru Öngörü Performansı için Alternatif Bir Yaklaşım: Yapay Sinir Ağı. *Gazi Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, *10*(2), Article 2.

Altay, E., & Satman, M. H. (2005). Stock market forecasting: Artificial neural network and linear regression comparison in an emerging market. *Journal of Financial Management & Analysis*, *18*(2), 18.

Anderson, D., & McNeill, G. (1992). *Artificial Neural Networks Technology*. Kaman Science Corporation,.

Avci, E. (2007). Forecasting Daily and Sessional Returns of the ISE-100 Index with Neural Network Models. *Doğuş Üniversitesi Dergisi*, *8*(2), Article 2.

Avci, E. (2015). Stock Return Forecasts with Artificial Neural Network Models. *Marmara Üniversitesi İktisadi ve İdari Bilimler Dergisi*, *26*(1), Article 1.

Barro, R. J. (1990). The Stock Market and Investment. *Review of Financial Studies*, *3*(1), 115–131.

Bengio, Y., Goodfellow, I., & Courville, A. (2015). *Deep Learning: Methods and Applications*. MIT Press.

Benli, Y. K. (2005). Bankalarda Mali Başarısızlığın Öngörülmesi Lojistik Regresyon Ve Yapay Sinir Ağı Karşılaştırması. *Gazi Üniversitesi Endüstriyel Sanatlar Eğitim Fakültesi Dergisi*, *16*, 31–46.

Bollen, N. P. B., & Busse, J. A. (2001). On the Timing Ability of Mutual Fund Managers. *The Journal of Finance*, *56*(3), 1075–1094.

Borsa İstanbul A.Ş. (n.d.). Retrieved 30 July 2022, from https://www.borsaistanbul.com /tr/sayfa/471/borsa-istanbul-hakkinda.

Boyer, B., & Zheng, L. (2009). Investor flows and stock market returns. *Journal of Empirical Finance*, *16*(1), 87–100. https://doi.org/10.1016/j.jempfin.2008.06.003

Campbell, J. Y., Lo, A. W., & MacKinlay, A. C. (1997). *The Econometrics of Financial Markets*. Princeton University Press. https://www.amazon.com/Econometrics-Financial-Markets-John-Campbell/dp/0691043019

Cao, J., & Wang, J. (2020). Exploration of Stock index Change Prediction Model Based on The Combination of Principal Component Analysis and Artificial Neural Network. *Soft Computing*, *24*(11), 7851–7860.

Chakravarty, S. (2005). Stock Market and Macro Economic Behavior in India. *Institute of Economic Growth [Cited 20 May 2011].*, 15.

Chen, A.-S., Leung, M. T., & Daouk, H. (2003). Application of Neural Networks to an Emerging Financial Market: Forecasting and Trading The Taiwan Stock Index. *Computers & Operations Research*, *30*(6), 901–923.

Chen, N.-F., Roll, R., & Ross, S. A. (1986). Economic Forces and the Stock Market. *The Journal of Business*, *59*(3), 383–403.

Chen, W., Zhang, Y., Yeo, C. K., Lau, C. T., & Lee, B. S. (2017). Stock market prediction using neural network through news on online social networks. *2017 International Smart Cities Conference (ISC2)*, 1–6.

Cheng, D., Yang, F., Xiang, S., & Liu, J. (2022a). Financial Time Series Forecasting with Multi-Modality Graph Neural Network. *Pattern Recognition*, *121*, 108218.

Cheng, D., Yang, F., Xiang, S., & Liu, J. (2022b). Financial time series forecasting with multi-modality graph neural network. *Pattern Recognition*, *121*, 108218.

Chong, E., Han, C., & Park, F. C. (2017a). Deep Learning Networks for Stock Market Analysis and Prediction: Methodology, Data Representations, and Case Studies. *Expert Systems with Applications*, *83*, 187–205.

Chong, E., Han, C., & Park, F. C. (2017b). Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies. *Expert Systems with Applications*, *83*, 187–205.

Çil, N. (2018). *Finansal Ekonometri*. Der Yayınları.

Coşkun, M., Kiraci, K., & Muhammed, U. (2016). Seçilmiş Makroekonomik Değişkenlerle Hisse Senedi Fiyatları Arasındaki İlişki: Türkiye Üzerine Ampirik Bir İnceleme. *Finans Politik ve Ekonomik Yorumlar*, *616*, Article 616.

Cowles, A. (1933). Can Stock Market Forecasters Forecast? *Econometrica*, *1*(3), 309–324.

Dase, R. K., & Pawar, D. D. (2010). Application of Artificial Neural Network for Stock Market Predictions: A Review of Literature. *International Journal of Machine Intelligence*, *2*(2), 14–17.

Davis, E. P., & Karim, D. (2008). Comparing Early Warning Systems for Banking Crises. *Journal of Financial Stability*, *4*(2), 89–120.

Deep Learning Toolbox. (n.d.). Retrieved 28 August 2022, from https://www.mathworks.com/products/deep-learning.html

Diler, A. I. (2003). Forecasting the Direction of the ISE National-100 Index by Neural Networks Backpropagation Algorithm. *Istanbul Stock Exchange Review*, *7*(25–26), 65–82.

Donaldson, R. G., & Kamstra, M. (1999). Neural Network Forecast Combining with Interaction Effects. *Journal of the Franklin Institute*, *336*(2), 227–236.

Dutta, G., Jha, P., Laha, A. K., & Mohan, N. (2006). Artificial Neural Network Models for Forecasting Stock Price Index in the Bombay Stock Exchange. *Journal of Emerging Market Finance*, *5*(3), 283–295.

Eğilmez, M. (2009). *Küresel Finans Krizi Piyasa Sisteminin Eleştirisi*. Remzi Kitabevi. https://www.nadirkitap.com/kuresel-finans-krizi-piyasa-sisteminin-elestirisi-mahfi-egilmez-kitap14581620.html

Eğrioğlu, E., Yolcu, U., & Baş, E. (2019). *Yapay Sinir Ağları* (1st ed.). Nobel Yayıncılık.

Enke, D., Grauer, M., & Mehdiyev, N. (2011). Stock Market Prediction with Multiple Regression, Fuzzy Type-2 Clustering and Neural Networks. *Procedia Computer Science*, *6*, 201–206. https://doi.org/10.1016/j.procs.2011.08.038

Enke, D., & Mehdiyev, N. (2013). Stock Market Prediction Using a Combination of Stepwise Regression Analysis, Differential Evolution-based Fuzzy Clustering, and

a Fuzzy Inference Neural Network. *Intelligent Automation & Soft Computing*, *19*(4), 636–648.

Fama, E. F. (1990). Stock Returns, Expected Returns, and Real Activity. *The Journal of Finance*, *45*(4), 1089–1108.

Fama, E. F., & French, K. R. (1996). Multifactor Explanations of Asset Pricing Anomalies. *The Journal of Finance*, *51*(1), 55–84.

Fathi, E., & Maleki Shoja, B. (2018a). Deep Neural Networks for Natural Language Processing. In *Handbook of Statistics* (Vol. 38, pp. 229–316). Elsevier.

Fathi, E., & Maleki Shoja, B. (2018b). Chapter 9—Deep Neural Networks for Natural Language Processing. In V. N. Gudivada & C. R. Rao (Eds.), *Handbook of Statistics* (Vol. 38, pp. 229–316). Elsevier.

Frankel, J. A., & Rose, A. K. (1996). Currency Crashes in Emerging Markets: An empirical Treatment. *Journal of International Economics*, *41*(3), 351–366.

Gomes, L. F. A. M., Machado, M. A. S., Caldeira, A. M., Santos, D. J., & Nascimento, W. J. D. do. (2016). Time Series Forecasting with Neural Networks and Choquet Integral. *Procedia Computer Science*, *91*, 1119–1129.

Gonenc, H., & Karan, M. B. (2003). Do Value Stocks Earn Higher Returns than Growth Stocks in an Emerging Market? Evidence from the Istanbul Stock Exchange. *Journal of International Financial Management and Accounting*, *14*(1), 1–25.

Goodell, J. W., Kumar, S., Lim, W. M., & Pattnaik, D. (2021). Artificial intelligence and machine learning in finance: Identifying foundations, themes, and research clusters from bibliometric analysis. *Journal of Behavioral and Experimental Finance*, *32*, 100577.

Gujarati, D. N. (2011). *Econometrics by Example*. Palgrave Macmillan.

Gujarati, D. N., & Porter, D. C. (2009). *Basic eEconometrics* (5th ed). McGraw-Hill Irwin.

Güloğlu, B., & Altunoğlu, E. (2011). Finansal Serbestleşme Politikaları Ve Finansal Krizler: Latin Amerika, Meksika, Asya ve Türkiye Krizleri. *İstanbul Üniversitesi Siyasal Bilgiler Fakültesi Dergisi*, *27*, Article 27.

Gürsoy, A. (2019). Makroekonomik Değişkenlerin Hisse Senedi Getirilerine Etkisi: Bankacılık Sektörü Örneği. *Ekonomi ve Finansal Araştırmalar Dergisi*, *1*(1–2), Article 1–2.

Haykin, S. S. (1999). *Neural Networks: A comprehensive Foundation* (2. ed., [Nachdr.], internat. ed). Prentice Hall.

Hebb, D. O. (n.d.). The Organization of Behavior. In *Chapter 4. "The First Stage of Perception: Growth of the Assembly"*.

Henrique, B. M., Sobreiro, V. A., & Kimura, H. (2019). Literature Review: Machine Learning Techniques Applied to Financial Market Prediction. *Expert Systems with Applications*, *124*, 226–251.

Hsieh, C. (1993). Some Potential Applications of Artificial Neural Systems in Financial Management. *Journal of Systems Management*, *44*(4), 12.

Hsieh, J., Chen, T.-C., & Lin, S.-C. (2019). Credit constraints and growth gains from governance. *Applied Economics*, *51*(11), 1199–1211.

Hu, M. Y., & Tsoukalas, C. (1999). Combining Conditional Volatility Forecasts Using Neural Networks: An Application to The Ems Exchange Rates. *Journal of International Financial Markets, Institutions and Money*, *9*(4), 407–422.

Hull, J. (2007). *Risk Management and Financial Institutions*. Pearson Prentice Hall. https://books.google.com.tr/books?id=G4nCQgAACAAJ

Ilahi, I., Ali, M., & Jamil, R. (2015). Impact of Macroeconomic Variables on Stock Market Returns: A Case of Karachi Stock Exchange. *SSRN Electronic Journal*.

Jing, N., Wu, Z., & Wang, H. (2021). A Hybrid Model Integrating Deep Learning with Investor Sentiment Analysis for Stock Price Prediction. *Expert Systems with Applications*, *178*, 115019.

Kaastra, I., & Boyd, M. (1996). Designing a Neural Network for Forecasting Financial and Economic Time Series. *Neurocomputing*, *10*(3), 215–236.

Kaminsky, G. L., & Reinhart, C. M. (1999). The Twin Crises: The Causes of Banking and Balance-of-Payments Problems. *American Economic Review*, *89*(3), 473–500.

Kamruzzaman, J., Begg, R. K., & Sarker, R. A. (2006a). *Artificial Neural Networks in Finance and Manufacturing*. Idea Group Publishing.

Kamruzzaman, J., Begg, R. K., & Sarker, R. A. (2006b). *Artificial Neural Networks in Finance and Manufacturing*. Idea Group Publishing.

Kantar, L., & Akkaya, M. (2018). Finansal Krizlerin Tahmininde Öncü Göstergelerin Logit-Probit Model ile Analizi: Türkiye Uygulaması. *International Journal of Management Economics and Business*, *14*(3), 575–590.

Kara, Y., Acar Boyacioglu, M., & Baykan, Ö. K. (2011). Predicting Direction of Stock Price Index Movement Using Artificial Neural Networks and Support Vector Machines: The Sample of The Istanbul Stock Exchange. *Expert Systems with Applications*, *38*(5), 5311–5319.

Karaçor, Z., & Alptekin, V. (2006). Finansal Krizlerin Önceden Tahmin Yoluyla Değerlendirilmesi: Türkiye Örneği. *Yönetim ve Ekonomi Dergisi*, *13*(2), Article 2.

Karan, M. B. (2013). *Yatırım Analizi ve Portföy Yönetimi* (4th ed.). Gazi Kitapevi.

Kaya, V., & Yilmaz, O. (2007). *Para Krizleri Öngörüsünde Logit Model ve Sinyal Yaklaşımının Değeri: Türkiye Tecrübesi* (Working Paper No. 2007/1). Discussion Paper.

Kaymaz, V., & Yilmaz, Ö. (2022). Impact of Macro Indicators on Istanbul Stock Exchange During Covid-19 Pandemic. *İzmir İktisat Dergisi*, *37*(1), Article 1.

Kennedy, P. (1998). *A Guide to Econometrics* (Fourth Edition). MIT Press.

Kim, K. (2003). Financial Time Series Forecasting Using Support Vector Machines. *Neurocomputing*, *55*(1), 307–319.

Kim, T. Y., Oh, K. J., Sohn, I., & Hwang, C. (2004). Usefulness of Artificial Neural Networks for Early Warning System of Economic Crisisrisis. *Expert Systems with Applications*, *26*(4), 583–590.

Kocabiyik, T., & Fattah, A. (2020). Makroekonomik Değişkenlerin Borsa Endeksleri Üzerine Etkisi: Türkiye ve ABD Karşılaştırması. *Finansal Araştırmalar ve Çalışmalar Dergisi*, *12*(22), 116–151.

Kohonen, T. (2001). *Self-Organizing Maps*. Springer Science & Business Media.

Kovacova, M., & Kliestik, T. (2017). Logit and Probit Application for The Prediction of Bankruptcy in Slovak Companies. *Equilibrium. Quarterly Journal of Economics and Economic Policy*, *12*(4), 775–791.

Kubat, C. (2019). *MATLAB Yapay Zeka ve Mühendislik Uygulamaları* (4.). Abaküs.

Kutsurelis, J. E. (1998). *Forecasting Financial Markets Using Neural Networks: An Analysis of Methods and Accuracy* [Master's Thesis]. Naval Postgraduate School.

Liang, X., Zhang, H., Xiao, J., & Chen, Y. (2009). Improving Option Price Forecasts with Neural Networks and Support Vector Regressions. *Neurocomputing*, *72*(13–15), 3055–3065.

Lin, C.-C., Chen, C.-S., & Chen, A.-P. (2018). Using intelligent computing and data stream mining for behavioral finance associated with market profile and financial physics. *Applied Soft Computing*, *68*, 756–764.

Lin, C.-S., Khan, H. A., Chang, R.-Y., & Wang, Y.-C. (2008). A New Approach to Modeling Early Warning Systems for Currency Crises: Can a Machine-Learning Fuzzy Expert System Predict the Currency Crises Effectively? *Journal of International Money and Finance*, *27*(7), 1098–1121.

Lintner, J. (1965). Security Prices, Risk, and Maximal Gains From Diversification. *The Journal of Finance*, *20*(4), 587–615.

Liu, M., & Shrestha, K. M. (2008). Analysis of the long-term relationship between macro-economic variables and the Chinese stock market using heteroscedastic cointegration. *Managerial Finance*, *34*(11), 744–755.

Markowitz, H. (1952). Portfolio Selection. *The Journal of Finance*, *7*(1), 77–91.

McCulloch, W. S., & Pitts, W. (1943). A Logical Calculus of the Ideas Immanent in Nervous Activity. *The Bulletin of Mathematical Biophysics*, *5*(4), 115–133.

McNelis, P. D. (2005). *Neural Networks in Finance Gaining: Predictive Edge in the Market*. Elsevier Academic Press.

Metlek, S., & Çetiner, H. (2021). *MATLAB Ortamında Derin Öğrenme Uygulamaları* (1st ed.). İksad Yayınevi.

Mossin, J. (1966). Equilibrium in a Capital Asset Market. *Econometrica*, *34*(4), 768–783.

Mukherjee, T. K., & Naka, A. (1995). Dynamic Relations Between Macroeconomic Variables and The Japanese Stock Market: An Application of a Vector Error Correction Model. *Journal of Financial Research*, *18*(2), 223–237.

Nunes, M., Gerding, E., McGroarty, F., & Niranjan, M. (2019). A Comparison of Multitask and Single Task Learning with Artificial Neural Networks for Yield Curve Forecasting. *Expert Systems with Applications*, *119*, 362–375.

Öğüt, H., Mete Doğanay, M., & Aktaş, R. (2009). Detecting stock-price manipulation in an emerging market: The case of Turkey. *Expert Systems with Applications*, *36*(9), 11944–11949.

Öztemel, E. (2006). *Yapay Sinir Ağları* (2.). Papatya Yayıncılık.

Öztemel, E. (2012). *Yapay Sinir Ağları* (3. Basım). Papatya Yayıncılık.

Panda, C., & Narasimhan, V. (2007). Forecasting Exchange Rate Better with Artificial Neural Network. *Journal of Policy Modeling*, *29*(2), 227–236.

Patil, P. R., Parasar, D., & Charhate, S. (2021). *A Literature Review on Machine Learning Techniques and Strategies Applied to Stock Market Price Prediction*. Springer.

Paul D. McNelis. (2005). *Neural Networks in Finance: Gaining Predictive Edge in the Market*. Academic Press; eBook Subscription Super Collection - Turkey (EBSCOhost).

Peng, Y., Albuquerque, P. H. M., Kimura, H., & Saavedra, C. A. P. B. (2021). Feature Selection and Deep Neural Networks for Stock Price Direction Forecasting Using Technical Analysis Indicators. *Machine Learning with Applications*, *5*, 100060.

Phillips, P. C. B., & Perron, P. (1988). Testing for a unit root in time series regression. *Biometrika*, *75*(2), 335–346.

Quazi, S. (2022). Artificial Intelligence and Machine Learning in Precision and Genomic Medicine. *Medical Oncology*, *39*(8), 120.

Roll, R., & Ross, S. A. (1980). An Empirical Investigation of the Arbitrage Pricing Theory. *The Journal of Finance*, *35*(5), 1073–1103.

Rousseau, P. L., & Sylla, R. (2003). Financial Systems, Economic Growth, and Globalization. In *Globalization in Historical Perspective* (pp. 373–416). University of Chicago Press.

Sarker, I. H. (2021). Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions. *Sn Computer Science*, *2*(6), 420.

Sayilgan, G., & Süslü, C. (2011). Makroekonomik Faktörlerin Hisse Senedi Getirilerine Etkisi: Türkiye Ve Gelişmekte Olan Piyasalar Üzerine Bir İnceleme. *BDDK Bankacılık ve Finansal Piyasalar Dergisi*, *5*(1), Article 1.

Sayılgan, G. (2011). *Soru ve Yanıtlarıyla İşletme Finansmanı* (5.). Turhan Kitapevi.

Schmidhuber, J. (2015). Deep Learning in Neural Networks: An Overview. *Neural Networks*, *61*, 85–117.

Schwert, G. W. (1990). Stock Returns and Real Activity: A Century of Evidence. *The Journal of Finance*, *45*(4), 1237–1257.

Sevüktekin, M., & Nargeleçekenler, M. (2010). *Ekonometik Zaman Serileri Analizi* (3.). Nobel Yayıncılık.

Sharpe, W. F. (1964). Capital Asset Prices: A Theory of Market Equilibrium Under Conditions of Risk. *The Journal of Finance*, *19*(3), 425–442.

Smith, K. A., & Gupta, J. N. D. (2000). Neural Networks in Business: Techniques and Applications for The Operations Researcher. *Computers & Operations Research*, *27*(11–12), 1023–1044.

Staub, S., Karaman, E., Kaya, S., Karapınar, H., & Güven, E. (2015). Artificial Neural Network and Agility. *Procedia - Social and Behavioral Sciences*, *195*, 1477–1485.

Sutskever, I., Martens, J., & Dahl, G. (n.d.). *On The Importance of Initialization and Momentum in Deep Learning*. 9.

Treynor, J. L. (1965). How to Rate Management of Investment Funds. *Harvard Business Review : HBR*, *43*(1), 63–75.

Tsai, C. F., & Wang, S. P. (2009). Stock Price Forecasting by Hybrid Machine Learning Techniques. *Proceedings of the International Multiconference of Engineers and Computer Scientists*, *1*(755), 60.

Tseng, C.-H., Cheng, S.-T., Wang, Y.-H., & Peng, J.-T. (2008). Artificial neural network model of the hybrid EGARCH volatility of the Taiwan stock index option prices. *Physica A: Statistical Mechanics and Its Applications*, *387*(13), 3192–3200.

Turing, A. M. (1950). Computing Machinery and Intelligence. *Mind*, *49*, 433–460.

Uğurlu, M., & Aksoy, H. (2006). Prediction of corporate financial distress in an emerging market: The case of Turkey. *Cross Cultural Management: An International Journal*, *13*(4), 277–295.

Vadlamudi, S. (2017). Stock Market Prediction using Machine Learning: A Systematic Literature Review. *American Journal of Trade and Policy*, *4*(3), 123–128.

Vui, C. S., Soon, G. K., On, C. K., Alfred, R., & Anthony, P. (2013). A review of stock market prediction with Artificial neural network (ANN). *2013 IEEE International Conference on Control System, Computing and Engineering*, 477–482.

Wang, L., Wang, Z., Qu, H., & Liu, S. (2018). Optimal Forecast Combination Based on Neural Networks for Time Series Forecasting. *Applied Soft Computing*, *66*, 1–17.

Wongbangpo, P., & Sharma, S. C. (2002). Stock Market and Macroeconomic Fundamental Dynamic Interactions: ASEAN-5 Countries. *Journal of Asian Economics*, *13*(1), 27–51.

Wooldridge, J. M. (2010). *Econometric Analysis of Cross Section and Panel Data* (The second edition). The MIT Press.

Wu, S.-I., & Lu, R.-P. (1993). Combining Artificial Neural Networks and Statistics for Stock-Market Forecasting. *Proceedings of the 1993 ACM Conference on Computer Science - CSC '93*, 257–264.

Yao, J., Poh, H., & Jasic, T. (1996). *Foreign Exchange Rates Forecasting with Neural Networks*. 754–759.

Yazici, A. C., Öğüş, E., Ankarali, S., Canan, S., Ankaralı, H., & Akkuş, Z. (2007). Artificial Nneural Networks: Review. *Turkiye Klinikleri Journal of Medical Sciences*, *27*, 65–71.

Yildiz, B. (2001). Prediction of Financial Failure With Artificial Neural Network Technology and an Empirical Application on Publicly Held Companies. *Istanbul Stock Exchange Review*, *5*(17), 47–62.

Yolcu, U., Egrioglu, E., & Aladag, C. H. (2013). A New Linear & Nonlinear Artificial Neural Network Model for Time Series Forecasting. *Decision Support Systems*, *54*(3), 1340–1347.

Yu, K., Tresp, V., & Schwaighofer, A. (2005). Learning Gaussian Processes from Multiple Tasks. *Proceedings of the 22nd International Conference on Machine Learning - ICML '05*, 1012–1019.

Zhang, G., Eddy Patuwo, B., & Y. Hu, M. (1998). Forecasting with Artificial Neural Networks: The state of the Art. *International Journal of Forecasting*, *14*(1), 35–62.

Zhang, G. P. (2001). An Investigation of Neural Networks for Linear Time-Series Forecasting. *Computers & Operations Research*, *28*(12), 1183–1202.

Zhang, G. P. (2003). Time Series Forecasting Using a Hybrid ARIMA and Neural Network Model. *Neurocomputing*, *50*, 159–175.

Zhang, G. P. (2004a). Business Forecasting with Artificial Neural Networks: An Overview. *Neural Networks in Business Forecasting*.

Zhang, G. P. (2004b). *Neural Networks in Business Forecasting*. Idea Group Inc (IGI).

Zurada, J. M. (1992). *Introduction to Artificial Neural Systems*. West.