STOCK MARKET FORECASTING WITH ARTIFICIAL NEURAL NETWORK MODELS: AN ANALYSIS OF LITERATURE AND AN APPLICATION ON ISE-30 INDEX

Dr. Emin AVCI*

Abstract

Although the artificial neural network models have been increasingly applied to solve variety of real life problems in last few decades, there are still some modeling problems exists in development of these models. This paper intends to provide a comprehensive review of the artificial neural network applications in stock market forecasting. Our goal is to provide a useful and up-to-date analysis of the literature, which will guide the future studies, by placing a special emphasis on the modeling issues. Furthermore, an application of neural network models for predicting the daily returns of ISE-30 index is presented.

Key Words: *Artificial Neural Network Models, Stock Market Forecasting*

Özet

YAPAY SİNİR AĞLARI MODELLERİ İLE HİSSE SENEDİ PİYASASI TAHMİNLERİ: LİTERATÜR ANALİZİ VE İMKB-30 ENDEKSİ ÜZERİNE BİR UYGULAMA

Son yıllarda yapay sinir ağları modelleri gerçek hayata dair pek çok problemin çözümünde yaygın şekilde kullanılmakla beraber, bu modellerin geliştirilmesi aşamasında halen bazı problemler bulunmaktadır. Bu çalışma hisse senedi piyasası tahminlerinde yapay sinir ağları modelleri uygulamalarını kapsamlı şekilde incelemeyi amaçlamıştır. Çalışmanın temel hedefi literatüre geçmiş uygulamaları özellikle modelleme yönünden irdeleyerek, ileride yapılacak araştırmalar için faydalı ve güncel bilgiler sunmaktır. Bunu yanında, IMKB-30 endeksinin günlük getirilerini tahmin etmeye yönelik bir uygulama yapılmıştır.

Anahtar Kelimeler: *Yapay Sinir Ağları Modelleri, Hisse Senedi Piyasa Tahminleri*

---

* Marmara Üniversitesi İngilizce İşletme Bölümü, öğretim görevlisi.

1. INTRODUCTION

For the last decade, neural network applications in finance have an increasing popularity by means of increasing studies on the subject and also by means of applications in different fields of finance. The applications of the neural networks in finance include market forecasting, portfolio selection and diversification, risk scoring, bond rating and business failures prediction.

The reason for the wide spread usage of the neural network models hinders in their unique features like fault tolerance, generalization, adaptability (Medsker, *et.al.*, 1993:11), being data driven model, which did not require a model specification (Desai and Bharati, 1998a: 128), being universal approximators (Hornik, et.al, 1989; Cybenko, 1989; Hornik, *et.al.*, 1990), ability to deal with complex information even the functional form of the data was not known (Hill, *et.al.*, 1994:6), hence nonlinearity (Cao, *et.al.* 2005:2502). Moreover, even if the neural network models are nonlinear in nature, these models can also be used effectively in forecasting linear time series (Zhang, 2001:1199).

However, utilizing the benefits of the neural network models requires valuable modeling considerations. Special emphasis must be placed architectural considerations like number of layers and number of nodes. Although, some rule of thumbs exists in the literature to overcome some modeling issues of neural network models, most of the vital parameters have been determined according to trial and error procedure (Kaastra and Boyd, 1996:217; Zhang, *et.al.* 1998:55).

Several studies have been devoted to guide the researchers on the modeling issues of the neural network models (Hush and Horne, 1993; Bishop, 1994; Kaastra and Boyd, 1996; McMenamin, 1997; Refenes, *et.al.*, 1997; Yao and Tan, 2001). On the other hand, some other studies are trying to help the researchers on this area by addressing the most common modeling procedures utilized in the literature. Among this type of studies, some researchers reviewed the applications that utilized the neural network models as a forecasting tool (Marquez, *et.al.*, 1992, Hill, *et.al.*, 1994; Zhang, *et.al.*, 1998), while some others investigated the neural network applications in variety of business fields including accounting, auditing, finance, management, and marketing (Vellido, *et.al.*, 1999). And, some studies focused on specific application fields of these models, like finance (Fadlalla and Lin, 2001), business failures (O'Leary, 1998; Perez, 2006) and stock market (Thawornwong and Enke, 2003).

This paper intends to provide a comprehensive review of the neural network applications in stock market forecasting. Our goal is to provide a useful and up-to-date analysis of the literature by placing a special emphasis on the modeling issues. Although, there are substantial studies on neural network applications in stock markets, it is decided to limit the literature with the applications in forecasting stock(s) or index(es) values or returns in the last decade in order to present the latest improvements and findings in the area. The literature review is covering total of 63 studies selected from the outstanding articles and conference proceedings, which have been published in generally respected and referred academic journals or conferences during 1996-2007. Furthermore, the forecasting performance of neural network models in predicting ISE-30 index (Istanbul Stock Exchange 30 index) will also be examined.

2. THE DEVELOPMENT OF THE NEURAL NETWORK MODELS

The origin of artificial neural network models dates back to McCulloch and Pitts's study in 1943 (Garson, 1998:1). However, the learning ability of the artificial neural network models was introduced by Donald Hebb in 1949 (Haykin, 1994:37). By 1962, more powerful learning rule was introduced by the work of Rosenblatt, who focused on learning ability in *perceptrons*, or single-layer feedforward networks (Gurney, 1997:209).

One of first significant study on the application of the neural network models for stock market applications was White's study in 1988. This study was questioning the validity of the efficient market hypothesis by examining the forecasting accuracy of the neural network models on IBM stock's daily returns (White, 1988: 451-458).

Following the White's study, several studies were conducted in order to measure the stock market forecasting power of artificial neural network models. While some studies were concentrated on measuring forecasting performance of one type of artificial neural network models (Gencay, 1996; Chandra and Reeb, 1999; Quah and Srinivasan 1999; Walczak, 1999; Eakins and Stansell, 2003; Lam, 2004; Sun *et. al.*, 2005), some other studies were conducted with the aim of comparing the forecasting performances of different artificial neural network models (Kohara, *et. al.* 1997; Saad, et al. 1998; Kim and Chung, 1998). Besides the studies that utilized only the artificial neural network models, there were some other studies, which compared the forecasting performances of different statistical forecasting methods (Dropsy, 1996; Desai and Bharati, 1998a; Lim and McNelis, 1998; Leung, et al., 2000; Maasoumi and Racine, 2002; Olson and Mossman, 2002; Rodriguez, et al., 2005).

Although, some contrary views exist in literature, results of most studies imply that the neural network models are superior to alternative forecasting techniques and the neural network models exhibit promising results for future studies.

3. FINANCIAL APPLICATIONS OF NEURAL NETWORK MODELS IN TURKEY

Although the neural network models have been utilized in variety financial applications (including market forecasting, portfolio selection and diversification, risk scoring, bond rating and business failures), the financial applications of neural network models in Turkey were mainly focused on the stock market forecasting and the financial failure prediction.

Among the studies that considered only the neural network models as a forecasting technique, Diler (2003:65-81) utilized the technical analysis indicators (momentum, MACD etc.) in a backpropagation neural network model (BPNN) in order to estimate the direction of the Istanbul Stock Exchange (ISE) for the following day. After examining several different architectures with changing momentum and learning rates, the model with 10 hidden nodes was found as the best model according to RMSE. The results of the study presented that the direction of the ISE-100 index could be predicted at a rate of 60.81%.

On the other hand, Egeli, *et al.* (2003) utilized the neural network models to forecast the ISE-100 index daily value. They compared the forecasting power of generalized feedforward (GNN) and multi layer perceptron models (BPNN) with the performance of moving averages rule. Total of 6 neural network models were trained with different number of hidden layers (1-2-4). Based on the mean relative percentage errors measure, the neural network models were superior to moving averages rule. When the neural network models compared with each other, the generalized feedforward neural network model was found to be more appropriate for predicting stock market index value.

Furthermore, Avcı (2007:128-42) investigated the forecasting performance of the backpropagation neural network model for ISE-100 index with daily and sessional data. The findings of the study presented that decreasing the data frequency (from daily to sessional data) increases the forecasting performance of neural network models for ISE-100 index. For all the periods under examination the forecasts of sessional analysis were superior to the daily analysis. Moreover, after utilizing the sensitivity analysis, it was also stated that including only the most important variables in the analysis had a positive effect on the forecasting performance of the neural network models.

In an another study, Avcı and Çinko (2007) examined the forecast performance backpropagation neural network (NN) models for daily index returns in six European emerging stock markets (Czech Rep., Hungary, Poland, Romania, Slovenia and Turkey). The neural network models were able to beat the market in 28 periods out of 36 analyzed periods. It was also shown that the statistical performance measures were unable to determine the most profitable neural network models.

Besides the studies, which examined only the neural network forecasting performance, there were some other studies that compared forecasting performance of the neural network models with other statistical forecasting techniques. All of these studies utilized the linear regression model as a benchmark model. Among this type of studies, Altay and Satman (2005:18-33) compared the forecast performances of neural network models with the linear regression for ISE-30 and ISE-All indexes. The forecasting performance of the backpropagation neural network model with different architectures investigated for daily, weekly and monthly data. The findings of the study showed that the forecasting performance of neural network models on the basis of statistical performance measures for daily and monthly data were failed to outperform the liner regression model, however these models were able to predict the direction of the indexes more accurately. Moreover, when the returns calculated from a hypothetical portfolio, neural network models were able to beat the linear regression and buy-and-hold strategy.

In the same manner, Karaatlı *et. al.* (2005:22-48) compared the monthly forecast performance of backpropagation neural network model and linear regression for ISE-100 index. On the basis of root mean square error (RMSE), it was found that the neural network model dominated the linear regression.

In comparing forecasting performances of the neural network model and stepwise regression, Çinko and Avcı (2007:301-307) utilized the ISE-100 index daily and sessional data. For both of the sessional and daily analysis, the backpropagation neural network model was superior to stepwise regression for all the periods based on the normalized mean square error and trend accuracy. However, according to mean square error performance measure, the stepwise regression dominated the neural network model for some periods under investigation.

Table 1: Technical details of the applications

Article	NN Type	Network Architecture (Number of layers and nodes in each layer)	Transfer Function	Performance Measures
Altay et.al. (2005)	BPNN	?-15-7-1 ?-10-10-1 ?-80-1	Tahn	RMSE, MAE, Theil's U, HIT, Return
Avcı (2007)	BPNN	14-7,14,28-1 10-7,14,28-1	Tahn	MSE, NMSE, MAE, Return
Avcı et.al. (2007)	BPNN	5-5,10,12-1	Sig	MSE, NMSE, MAE,HIT, Return
Çinko et.al.(2007)	BPNN	NP	NP	MSE, NMSE, HIT
Diler (2003)	BPNN	7-6,7,8,9,10,11,12,15-1	NP	RMSE, HIT
Egeli et.al. (2003)	BPNN GNN	8-?-1 8-?-?-1 8-?-?-?-?-1	NP	R^2, MRPE
Karatlı et.al. (2004)	BPNN	7-7-3-3-1	NP	RMSE

HIT: The hit rate or the proportion of the correct directional forecasts, positive or negative or overall
NMSE: Normalized Mean-Squared Error
NP: Not provided
MAE: Mean absolute error
MAPE: Mean absolute percentage error
MRPE: Mean Relative Percentage Error
MSE: Mean squared error
RMSE: Root mean squared error
Return: Return calculated for hypothetical investment

Table 1 provides the technical details of the neural network applications for ISE forecasting. As it can be seen from the table, the BPNN model is utilized in all of the studies. Other than the BPNN model, only one study utilized some other neural network model (GNN).

In deciding the optimal architecture of the neural network models, all the reviewed studies followed the trail-and-error procedure. Several different architectures were tested with different number of hidden layers (1, 2, 3 and 4) and different number of hidden nodes. Moreover, the studies, which reported the activation functions, utilized tangent or sigmoid activation functions.

The performances of the neural network models were tested according to several statistical performance measures. HIT, MSE, NMSE and MAE were the most common statistical performance measures. Only three studies (Altay and Satman, 2005; Avcı, 2007; Avcı and Çinko, 2007) analyzed the profitability of the neural network models according to trading strategy. Furthermore, Karaatlı, *et.al.* (2004), Altay and Satman (2005), Çinko and Avcı (2007) were the studies that compare the neural network models' performances with some other statistical forecasting model.

Table 2: Data analysis of the applications

Article	Data Pre-Processing	Data Frequency	Data Sets (Train-Validate-Test)
Altay et.al. (2005)	Logarithmic Difference	Daily-Weekly-Monthly	75% - NA - 25%
Avcı (2007)	Logarithmic Difference	Daily	70% - 20% - 10%
Avcı et.al. (2007)	Logarithmic Difference	Daily	70% - 20% - 10%
Çinko et.al.(2007)	Logarithmic Difference	Daily	70% - 20% - 10%
Diler (2003)	Normalization	Daily	80% - NA - 20%
Egeli et.al. (2003)	NP	Daily	90% - NA- 10%
Karatlı et.al. (2004)	NP	Monthly	80% - NA - 20%

NA: Not applied
NP: Not provided

BPNN: Backpropagation Neural Network Model
GNN : Generalized Feedforward Neural Network Model

Table 2 provides the data related issues in forecasting the ISE. Most of the studies utilized the logarithmic difference in order to transform the price data to return series (Altay and Satman, 2005; Avcı, 2007; Avcı and Çinko, 2007). Only two studies (Diler, 2003; Karaatlı, *et.al.*, 2004) used the price series in the model directly. In accordance with the forecast objective several studies utilized daily data, however in two of the studies weekly and/or monthly data were also considered in the researches (Altay and Satman, 2005; Karaatlı, *et.al.*, 2004). Data sets were divided into two or three subsets in order to train, validate and test the models. In line with the international literature, 70% to 90% of the whole data set was used to train the model and the rest of the data were utilized for validation and/or testing purposes.

Although the core concept of this study is to present the neural network model applications in forecasting stock market values or returns, it is decided to provide some other financial applications of these models in order to provide a comprehensive review of the Turkish market applications.

Other than the market direction or return forecasting efforts, some studies were devoted to forecast the stock market volatility by the use of neural network models. Yümlü, *et. al.* (2003:553-560), Yümlü, *et. al.*(2004), Yümlü, *et. al.* (2005:2093-2103) studied the predictability of Istanbul Stock Exchange (ISE) index and compared the forecasting performances of several statistical methods including mixture of experts (MoE), feed-forward neural networks, multilayer perceptron (MLP), radial basis function networks (RBF), recurrent neural networks (RNN), Jaganathan-Runkle (GJR) and exponential generalized autoregressive conditional heteroskedasticity (EGARCH) models. They stated the superior of the MoE model over the other models from several aspects.

Besides the studies that utilized the neural network models for stock market forecasting, some studies were devoted to utilize neural network models for financial failure prediction and financial strength forecasting. Among first type of studies, Yıldız (2001:51-67) studied on the predictability of financial failure in industrial, commercial and service firms during 1983-1997. Relying on the findings of the study, it was concluded that the neural network models were better predictors for financial failures for publicly traded companies when compared to discriminant analysis. In the same manner, Aktaş, *et.al.* (2003:1-25) compared the forecast performance of neural network models with multiple regression, discriminant and logit analyses. The neural network models found as the best predictor model. In order to compare the forecast performances of neural network models and logistic regression in financial failure prediction, Benli (2002:17-30) studied the industrial firms, which were listed on ISE during the period of 1992-2001. The results showed that the neural network models could attain more correct classification rates. In another study, Benli (2005:31-46) studied 18 banks, which were transferred to Savings Deposits Insurance

Fund, and 21 survival banks during 1997-2001. According to overall correct classification percentages, the findings of the study showed that the neural network models were able to make correct classification at 87%, where the logistic regression could attain 84.2%.

Other than the financial failure prediction, Boyacıoğlu and Kara (2006) compared the performance of discriminant analysis, logistic regression, cluster analysis and the neural network models in forecasting financial strength ratings of 18 banks, which had financial strength rating from Moody's during 2001-2005. Results of the study presented that the neural network models were more effective than other multivariate statistical analyses.

4. NEURAL NETWORK MODEL DESIGN

The design of a neural network model can be expressed by the combination of two terms: the neural network architecture and neurodynamics. The neural network architecture deals with the number of layers, number of nodes in the layers and transfer functions while neurodynamics deals with the properties of each neuron like the transfer functions. The combination of neural network architecture and neurodynamics presents the model for neural network (Kaastra and Boyd, 1996:224).

Variable selection and the data preparation are the other important aspects in developing successful neural network models. Although, data consideration seems like a very easy task at first glance, it is notable that some researches fail as they could never complete the required data set (Deboeck and Cader, 1994:30). Moreover, the researchers should consider three basic factors: data availability, data quality, frequency of the data and cost of the data.

Although, network architecture, neurodynamics and data considerations are the vital elements of successful neural network modeling, there is not a specific method for determining the optimal values of these parameters, and yet neural network modeling has many free parameters to be determined by the researchers (Lim and McNelis, 1998:224). However, some rule of thumbs exists in the literature to overcome modeling issues of neural network models and most of the vital parameters have been determined according to trial and error procedure (Zhang, *et.al.*, 1998:55; Kaastra and Boyd, 1996:217).

The purpose of this section is to present some methods and rule of thumbs in designing neural network models and to address the most common problems in neural network modeling. Moreover, the solutions (if any) or the practices in the literature for these problems will be illustrated.

Table 3: General Review of the Studies

Year	Number of Articles	Number of Conference Proceedings	Total	Year	Number of Articles	Number of Conference Proceedings	Total
1996	5	2	7	2002	2	1	3
1997	2	-	2	2003	7	2	9
1998	7	-	7	2004	2	-	2
1999	6	1	7	2005	6	1	7
2000	4	1	5	2006	4	1	5
2001	2	1	3	2007	3	3	6

Total of 63 studies, which were published during 1996-2007, were reviewed in order to present the practices in neural network modeling especially for stock market forecasting. These studies were selected from the outstanding articles and conference proceedings which have been published in generally respected and referred national or international academic journals or conferences. *Table 3* presents the general overview of the studies reviewed.

4.1. Data

The first step in the neural network model development is data understanding, which is basically composed of variable selection and data preparation. However, before proceeding to the variable selection and data collection phases, the objective of the model should be established. The objective of a neural network model may be short-term or long-term forecasting (forecast horizon) of stock market indexes or stocks. Thus the definition of the forecast objective requires decision on what to forecast and what the forecast horizon will be.

In the literature, most of the studies utilized the neural network models for stock market index forecasting, as it was accepted that the indexes were able to represent all the stocks in the market. Moreover, by the use of index forecasting, some unsystematic risk factors can be avoided (Tharnwonwong and Enke, 2003:51). In the review, it was observed that 53 of the studies forecasted the stock market indexes. While, some studies concentrated on index forecasting of one market (see O'Connor *et.al.*, 2006; Maasoumi, *et.al.*, 2002; Edelman, *et.al.*, 1999), the other studies examined the forecast power of neural network models in more than one market (see Kanas, *et.al.*, 2001; Leungh, *et.al.*, 2000; Patel and Marwall, 2006). On the other hand, only 10 studies placed their forecasting efforts on stocks (see Cao, *et.al.*, 2005; Ng, *et.al.*, 2002; Olson and Mossman, 2003).

Another crucial point is determining the data frequency. The frequency of the data set used in the model should be determined in accordance with the forecast horizon. For example, if the objective is short-term forecasting in stock market then the daily stock market data would be appropriate. On the other hand, researchers with longer forecasting horizon should basically focus on the weekly, monthly or quarterly data (Kaastra and Boyd, 1996:220).

Among the reviewed studies, the number of studies that determined their forecast horizon as intraday (sessional), daily, weekly, monthly and yearly is 2, 43, 4, 16 and 5 respectively. In line with their forecast horizon, some studies utilized intraday (sessional), daily, weekly, monthly and yearly data (See Appendix II for details).

However, there are some exceptional cases. Abdelmouez, *et.al.* (2007), Ghazali *et.al.*(2005) and Walczak (1999) utilized daily data for predicting 5 days ahead. Moreover, Moreno *et.al.* (2007) and Leigh, *et.al.* (2002) made weekly and 20 days ahead forecasts by the use of daily data. On the other hand, Roman and Jameel (1996) considered weekly data for yearly forecasts.

Some studies examined forecasting power of neural network models more than one forecast horizon (sessional, daily, weekly and monthly). While the results of Chenoweth, *et.al.* (1996a) and Altay and Satman (2005) presented the superiority of low frequency data (using monthly data instead of using daily or weekly data) in terms of financial performance measure (return), Avcı (2007) found that the neural network guided trading strategy with sessional data generated highest returns in comparison with daily data. Such conflict may be due to the fact that, each study utilized different neural network models with different architectures and such models were tested on different sampling periods in different market indexes.

4.1.1. Variable Selection

Financial applications of neural network models can consider several information sources including the fundamental data, the technical data and economic data. Fundamental data basically consists of information provided by the financial statements and the financial ratios generated from these statements. On the other hand, technical data consists of market information regarding the intra-day, daily, weekly and monthly index return, volume, high-low price. And moreover, technical indicators (MACD, RSI etc.) can be included under this category. The economic data includes the inflation rate, interest rate, exchange rate, unemployment rate, capacity usage and so on.

Those information sources provide a variety of information (a stock market, for example, opening price, closing price, high-low prices, the volume of trade, several financial ratios, inflation, consumer price index etc.) that can be used as an input variable in the model. However, the question is: which variables will be included in the model?

Several types of input variables were utilized for neural network modeling in the literature and it was clear from Appendix II that there is a sharp disparity between the applications. While some studies solely relied on the technical data like index returns, volume or high-low price (Gencay, 1998; Groşa, *et.al.*, 2005; Kim and Shin, 2007), some other studies utilized only the economic data (Desai, *et.al.*, 1998a,b; Dropsy, 1996; Stansell and Eakins, 2003) or financial ratios (Olson, and Mossman, 2003) or technical analysis indicators (Kim, 2006). Moreover, in some studies the mixture of different data types (including economic and technical data) were utilized together (Lam, 2004; Refenes, *et.al.*, 1997; Chenoweth, *et.al.*, 1996b).

Besides those data types mentioned above, practitioners also used some other data types. For example, Phua, *et.al.* (2001) considered foreign market indexes as input to the model. Kohara *et.al.* (1997) studied effectiveness of qualitative variables by the use of event-knowledge for stock price prediction. Furthermore, Shi, *et.al.* (1999) utilized the forecasts of some other statistical models as an input to a neural network model.

As the above explanations and the appendix II presents, the input variables are often very different than each other, even when the same test is used or same market is being forecasted. In most of the studies, there is not an explanation on why some variables were chosen and why the others were omitted (Thawornwong and Enke, 2003:52). The input variables were mostly subjectively selected and included in the analysis directly without any prior analysis (Gencay, 1998; Gencay and Stengos, 1998; Qi, 1999; Brownstone, 1996).

However, there were some exceptional studies that presented their reasoning of inclusion or omission of some input variables. Among this type of studies, Chen *et.al.* (2003:903-904) and Dropsy (2004:122-123) stated their economic rationale for considering each economic variable for stock market forecasting.

On the other hand, some other studies utilized or advised statistical techniques in order to select the appropriate input variables among several candidate variables. These techniques includes regression analysis (Desai and Bharati, 1998a; Çinko and Avcı, 2007; Altay and Satman, 2005, Singh, 1999), sequential selection techniques (Chenoweth, *et.al.*, 1996a), adjusted R-square (Dropsy, 1996), principle component analysis (Safer, 2001), genetic algorithm (Leigh, *et.al.*, 2002; Phua, *et.al.*, 2001), Akaike's minimum final prediction error (Chen, *et.al.*, 2003) and sensitivity analysis (Safer, 2001; Avcı, 2007; Yao, *et.al.*, 1999).

4.1.2. Data Preparation

After the variables are selected and the data set is collected, the data set should the prepared for modeling purposes. This phase is generally known as the data pre-processing. The need for data pre-processing arise from the statistical problems like seasonality, trends and nonstationary hindered in the price data series (raw data). Although, the usage of the raw data could be observed seldom in the neural network literature (Kaastra and Boyd, 1996:220), these series should not be directly used in their original form for forecasting the stock markets, as the inabilities of neural network models in modeling seasonal and trend variations in the data set was documented (Zhang and Qi, 2005:509-511).

One possible way to overcome the problems in raw data sets is using *transformation*. There are several ways to transform the data: variable differences in the input data $(x_i - x_{i-1})$, relative variable difference $(x_i - x_{i-1}) / q$ (q stands for the standard deviation), natural algorithm $(\ln(x_i / x_{i-1}))$, square root of the selected variables $(x_i)^{1/2}$ and hence trigonometric functional form $(cos(x_i) or\ sin(x_i))$ (Azoff, 1994:22).

The natural logarithm or the logarithmic difference is the most common technique in transforming the raw data set (Kanas, 2003; Rodriguez, 2005; Kim, *et.al.*, 1998; Wang, et.al, 2003; Darrat and Zhong, 2000; Lim, *et.al.*, 1998; Rodriguez, *et.al.*, 2000; Rodriguez, *et.al.*, 2005; Altay and Satman, 2005; and so on). On the other hand, some researchers utilized the relative variable difference transformation for data pre-processing (Sun, 2005; Ghazali *et.al.*, 2006; Abdelmouez *et.al.*, 2007). Moreover, variable difference transformation methodology is also available in the literature (Kohara, *et.al.*, 1997).

Even after the transformation, there are still some problems with the data. Most of the times the magnitudes of the variables in the data set are far different than each other. For example, while some variables have a range between 100 and 500, the others may have a range between 1 and 15. If the variables are used as they are, than the first variable will dominate the other (Callan, 1999:35). Then, the large differences between the ranges of the variables can affect the importance of the other useful inputs (Refenes, 1995a:57).

Moreover, most of the times, the variables are not in the normal operational range of the network. The operational range of the neural networks is defined by the transfer functions used in the network model. The transfer functions generally have an output range of (0,1) or (-1,1). Again, large differences between the operational range of the network and range of the input variables will distort the neural network model forecasting abilities.

The solution to these problems is *scaling* the variables in the data set. Scaling is also called as normalization or standardization in literature. Scaling of the input parameters can clarify which weights are changing as the model proceeds and which input variables the model is using. If the scaling is not used, then understanding and interpretation of neural network models becomes more difficult (Masters, 1993:267).

There are mainly two techniques used in the literature for scaling. The first technique is *statistical scaling* (often-called standardization or z-score normalization) in order to have a variable distribution with zero mean and unit standard deviation. The statistical standardization is achieved by subtracting mean from the actual variable and dividing it by the standard deviation (Refenes, 1995a: 57):

$$x_i^{scl} = \frac{x_i(t) - \bar{x}_i}{\sigma_{x_i}} \tag{1}$$

The second technique is linear normalization, which is used to scale the variable in the operation range of the neural network (Masters, 1993:262):

$$x_i^{scl} = \left[\frac{Max - Min}{x_{Max} - x_{Min}}\right] \times x + \left[Max - \frac{Max - Min}{x_{Max} - x_{Min}} \times x_{Max}\right] \tag{2}$$

Where, X_{Max} and X_{Min} are the maximum and minimum values for the variable and *Max* and *Min* are the maximum and minimum values of the network operating range, and x is the original value of the input variable. The maximum and minimum values for the network operating range are determined by the transfer functions used in the model. If the logistic sigmoid function is considered these values will be in between (0,1). However, it can be taken as (0.1,0.9) or even (0.15,0.85) because the transfer functions are assumed to reach their maximum and minimum levels at infinity (Refenes, 1995a:58; Azoff, 1994:28). On the other hand, Zhang, *et.al.* (1998:50) argues that an output of a node can be small as 0.000045 and large as 0.99995, hence it is not necessary to take the maximum and minimum.

Another approach to linearly normalize the variables in the range of (–1,1) is (Fine, 1999:145):

$$x_i^{scl} = 2\frac{x_i - x_{Max}}{\bar{x}_i - x_{Min}}, \text{ where } \bar{x}_i = \frac{1}{2}(x_{Max} + x_{Min}) \tag{3}$$

Several studies applied the normalization techniques stated above. While some of the studies reported their normalization technique clearly, as z-score normalization (Ng and Lam, 2004; Leigh et.al ., 1998; Leigh et.al ., 2005;) and linear normalization (Yao, *et.al.*,1999; Sun, *et.al.*, 2005; Chen, 2006; Ghazali *et.al.*, 2006; Patel and Marwall, 2006; Diler, 2003); some other studies just stated the usage of normalization in the data pre-processing without providing the normalization technique (Brownstone, 1996; Kim, *et.al.*, 1998; Rodriguez, *et.al.*, 2005; Refenes *et.al.*, 1997). On the other hand, different normalization techniques, such as sigmoidal normalization, also exists in the literature (Schierholt and Dagli, 1996)

It should also be stated that the scaling of the input data is not necessary but it is recommended, as scaling removes effects of outliers, which may exists in the data (Hamid and Iqbal, 2004:1123). Moreover, in constructing a neural network model with linear activation function in the output layer, scaling of the input variables can be discarded. The reason is that there is no defined operational range of the output layer in such models (Hg and Lam, 2000:41-46).

4.2. Network Architecture

One of the most challenging difficulties in neural network modeling is determining the optimum architecture. The architectural design of the neural network models requires special consideration on the number of layers, number of nodes in each layer and the transfer functions to be used in modeling.

By changing the number of input nodes, the number of hidden layers and also the number of nodes in the hidden layers and number of nodes in the output layer, simple neural network architecture can be altered with a more complex architecture. However, in the case of time series forecasting, constructing complex architecture does not guarantee the forecasting accuracy. For example, increasing number of hidden nodes and hidden layers will be harmful in terms of model training time and generalization. Some models may suffer from over-fitting problem due to the complexity as well (Reed, 1993:740).

Several different methodologies for determining the optimum neural network architecture was introduced in the literature. The pruning techniques (skeletonisation, optimal brain damage, optimal brain surgeon), constructive methods (cascade correlation, meiosis networks, dynamic node creation), regularization techniques (weight decay, weight elimination, curvature-driven smoothing) and in-sample model selection criteria (AIC, BIC and their variants) are some examples for these methodologies. Although, these methodologies are quite complex and difficult to implement, neither one of them guarantees the optimum architectural design (Zhang, *et.al.*, 1998:42) and out-of-sample performance (Qi and Zhang, 2001:678). Yet, there is no firm answer on these parameters; however only some heuristics can be used (Desai and Bharati, 1998:132).

It should also be stated that most of the studies utilized trial and error methodology in determining the network architecture. The trial and error methodology requires training of several neural network architectures with different complexity and selection of optimum architecture according to predetermined performance requirements. Although, trial and error process is time consuming, it is unfortunately unavoidable (Ma and Khorasani, 2004:589).

In the following pages, the basic elements of neural network architecture (input nodes, hidden layers, hidden nodes and transfer function) will be illustrated with referring practices in literature. Appendix-I provide the technical details of the neural network architectures in reviewed studies. The network layers are denoted by *Input Layer (number of input nodes) – Hidden Layer (number of hidden nodes) – Output Layer (number of output nodes)*. For example 5,6-7-1 denotes an architecture of 3 layered network (one hidden layer) which have 5 and 6 input nodes in the input layer, 7 hidden nodes in the hidden layer and 1 output node in the output layer. However, it should also be stated that the following methodological considerations are more suitable to the feedforward neural network models.

4.2.1. Determining Number of Input Nodes

Decision on the number of input nodes may be the easiest parameter to determine. The number of input nodes in the input layer represents the number of independent variables in the sample data. So, decision on how many nodes should exist in the input layer clearly depends on the number of independent variables (Zhang, *et.al.*, 1998:45).

As stated in the preceding pages, some researchers applied regression analysis, sensitivity analysis and genetic algorithm to determine the appropriate input variables. Although, there is not a specific method for the determining the number of input nodes (Thawornwong and Enke, 2003:52), it should be stated that too many or too few input nodes would directly affect the forecasting and learning abilities of the model (Zhang, *et.al.*, 1998:45).

From Appendix-I, it is obvious that there is no consistency in terms of number of input nodes used in the models. The number of input nodes utilized in the literature has a range from 1 to 49. Several input node experimentation were handled in order to assess the effects of change in the number of input nodes in forecasting.

4.2.2. Number of Layers

The generalization ability of a neural network model greatly depends on two parameters: the number of hidden layers and the number of hidden nodes. In theory for most of the cases, having only one hidden layer is sufficient for the neural network model (See: Garson, 1998; Zhang, *et.al.*, 1997; Kaastra and Boyd, 1996). However, forecasting ability of the network models with one hidden layer depended on the number of hidden nodes in the model. In order to train the model with one hidden layer, the number of hidden nodes should be increased. On the other hand, if the number of hidden nodes increased, the training time of the network and the generalization ability of the network could be deteriorated (Zhang, *et.al.*, 1998:44).

In addition to one hidden layered network models, two hidden layered network models are also utilized in the literature. Depending of the data structure, two hidden layered network models can provide more desirable results. Moreover, as one layered network models can be overloaded by increasing number of hidden nodes, some researchers preferred to use a neural network model with two layers.

Appendix I provide the number of layers utilized in the literature. Total of 44 (out of 63) studies developed neural network models with one type of structure. Among these studies, 39 studies (out of 44) relied on one hidden layer neural network model (Baba, and Yanjun, 2002; Eakins, *et.al.*, 2002; Safer, 2003; Sitte, *et.al.*, 2000; Kim, *et.al.*, 2007; Leigh, *et.al.*, 2005; and many others) and 5 studies (Armano, *et.al.*, 2005; Chen, *et.al.*,2003; Deasi, *et.al.*, 1998a; Edelman, et.al, 1999; Shi, *et.al.*, 1999) considered only two hidden layer neural network models.

On the other hand, 9 studies (out of 63) considered more than one type of network architecture. Among this type, 6 studies utilized 1 and 2 hidden layer network models (Yao, *et.al.*, 1999; Schierholt, *et.al.*, 1996; Saad, *et.al.*, 1998; Kim, *et.al.*, 1998; Chandra, *et.al.*, 1999; Altay and Satman, 2005), 2 studies utilized 3 hidden layer network models (Roman, *et.al.*, 1996; Lam, 2004) and 1 study utilized 4 hidden layer network model (Egeli, *et.al.*, 2003). Hence, 10 studies did not provide the network architectures.

As literature analysis presents, it is a very rare case to find a neural network model with more than two hidden layers. The reason is that to increase the number of hidden layers also increases the training time of the model and increases the danger of over fitting problem. Moreover, as the neural network model with two hidden layers can perform well in complex decisions, the neural network models with more than two hidden layers are not required (Lippmann, 1987:16).

In considering the optimal number of hidden layers, Yao, *et.al.* (1999:235) found that 2 hidden layer is better than 1 hidden layer in terms of return. In the same manner, Schierholt and Dagli (1996:77) reported that 2 hidden layer is slightly better than 1 layer neural network model. Lam, (2004:575-577) carried out 4 experiments with different neural network architectures. Results presented that the neural network model with 3 hidden layers generated the highest return for 2 experiments while one and two hidden layer networks were the best for the remaining two experiments. However, Lam concluded that the returns generated by 3 hidden layer models are not far better than the 1and 2 hidden layer models. Hence, if the marginal utility and cost of developing complex architecture was taken into account, less complicated model would be the best. Consistent with the Lam (2004), Egeli, *et.al.* (2003) found that 1 hidden layer neural network model was better than 2, 3 and 4 hidden layer models in terms of forecasting accuracy.

The evidence on the superiority of two hidden layered network over one hidden layer network is confusing, in the sense that the superiority can be validated on special types of network architectures. Tamura and Tateishi (1997:251-255) presented the superiority of two hidden layers if the hidden units are selected on the basis of $N-1$ and $(N/2)+3$, for one hidden layered and two hidden layered network. (where N is the number of input-target training pattern). Although, two hidden layered neural network models outperform one hidden layered model in some circumstances, it is more difficult to train the former.

It is advised that whatever the structure of the data is, first construct a model with one hidden layer and if using large number of hidden neurons is not satisfactory then proceed to two hidden layered model by reducing total number of hidden neurons (Masters, 1993; p:176). It is just the trial and error case, as there is no generally accepted technique for deciding the number of hidden layers (Zhang, *et.al.*, 1998:44).

4.2.3. Number of Hidden Nodes

Another crucial point in designing the neural network model design is determining the optimum number of hidden nodes. It is known that neural network models with few hidden nodes have better ability in terms of generalization. However, the forecasting ability of the network model may be limited if the number of hidden nodes is too few. The reason is that the probability of the network model to capture the complex input-output structures is decreasing. On the other hand, if there are too many hidden nodes, at that time the generalization ability of the network model is limited and the over fitting or overtraining problems can be realized (Kwok and Yeung, 1999:1)

Despite the importance of selecting the optimum number of hidden nodes, again, there is no theoretical basis for determining the number of this parameter. However, some rules of thumb exist in literature and they can be used as a starting point. Some researchers advise that the average number of input nodes and output nodes should be used as the initial number of hidden nodes. For example, Callan (1999:134) stated that the number of hidden nodes in a single hidden layer network can be 30-50% of the inputs in the input layer.

On the other hand, some researcher advice heuristic choices to find out the optimum number of hidden nodes to be equal to n, $2n$, $2n\pm1$, $2n\pm1$ and $n/2$, where n is the number of inputs nodes. However, as stated these techniques are the heuristic advises and their validity is case specific (Zhang, *et.al.*, 1998:44).

Moreover, some other approaches are suggested in determining the optimum number of hidden nodes. One approach is the pruning. Pruning approach states that the model must be started with the highest available hidden node to find out the optimum number. In addition, one hidden node with the smallest weight in the model must be dropped after each training. Other approach, growing approach, is the opposite of pruning; it suggests starting with the number of hidden nodes as few as possible; and, if necessary adding one hidden node after each training. In both of these approaches, the selection is made by comparison of which size of hidden nodes is preferable.

In the literature, a few studies reported their methods in determining number of hidden nodes. Among these studies, Brownstone (1996:239) started from a number of nodes, which known to be too few and increased the number until the best result found. And Desai and Bharati (1998b: 415 and 1998a:139) utilized pruning and weight elimination techniques. Furthermore, Abdelmouez (2007) utilized k-fold cross validation for determining number of hidden nodes.

Furthermore, most of the studies relied on the trail error procedure and trained several neural network models with different number of hidden nodes (see Avcı, 2007; O'Connor, et.al, 2006; Diler 2003; Sitte, *et.al.*, 2000; Singh, 1999)

4.2.4. Transfer Functions

Although, the neural network architecture is designed properly, the model may not work properly because the neural network architecture does not deal with the properties of the basic unit of the neural network model: the neuron. As a basic processing element, each neuron plays an important role in the model construction. The main problem about the neuron is determining the transfer function to be used in the model.

The transfer function (sometimes referred to as the transformation function, activation function, squashing or threshold function) determines the relationship between the internal activity level of a neuron and its output. The internal activity of a neuron is composed of summation of the input parameters multiplied by their appropriate weights. However, the firing of the neuron depends solely on the transfer functions used in the model.

Several transfer functions can be considered in neural network modeling. However, in practice a few transfer functions are generally used. The most common transfer functions utilized in neural network models are linear functions (identity, step function, signum, and perceptron function), hyperbolic tangent (tanh) function and logistic function.

The linear functions are suitable to the models where there is no nonlinear relation in the data, and where the model requires small architecture with limited connections to output layer and the output is continuous (Garson, 1998: 97-98). The use of linear transfer function in the hidden layers is not preferable for the multi-layer neural network models, because such a network model will be identical to the single layer nets (Fausett, 1991:17). As the use of linear transfer functions in the hidden layers limits forecasting ability of neural networks for the non-linear data, one of the important advantages of the neural network modeling for forecasting cannot be utilized.

Although, the linear functions are not recommended for the multilayer perceptrons, some other linear functions can be used in the output layer of the model (Masters, 1993:84). Using linear function in the output layer eases the forecasting process. Moreover, no improvement can be realized in forecasting with non-linear functions in the output layer most of the time (McMenamin, 1997:18).

Another set of transfer functions is tanh and logistic functions. These functions are generally referred to as the sigmoid functions. These functions are by far most common forms used in the construction of neural network models (Kaastra and Boyd, 1996:227). The name sigmoid had chosen to state their S-shaped curves (Fausett, 1991:17). They are preferred as they monotonically increase in value and they can be differentiable.

The logistic sigmoid function keeps the output of a neuron between 0 and 1. On the other hand, the hyperbolic tangent transfer function constrains the output value between –1 and 1. In the case of logistic function, only the high values of the inter neuron activities are important in learning as the lower internal activities have a value of 0. On the other hand, in the case of hyperbolic tangent transfer function, higher and lower intra neuron activities affect the learning (Garson, 1998:98). The use of hyperbolic tangent transfer function is found to be more effective in terms of fast convergence of training (Bishop, 1995:127). Moreover, as the hyperbolic tangent transfer function is symmetric, it is recommended to use this function as a starting point (Garson, 1998:98).

The Gaussian function evaluates the inter activity of a neuron according to a bell shaped curve. The output of the neuron increases until a central point value of input and then it starts to decrease (Garson, 1998:98). The Gaussian transfer function is mostly suitable for radial basis function networks but this function can also be used in the multilayer perceptrons with a different activation function in the model (Azoff, 1994:53).

The literature details presented in Appendix I show that 29 studies (out of 63) did not provide the transfer functions utilized in the model design. However, among the studies in which the transfer functions are reported, it is observed that sigmoid transfer functions (including the logistic sigmoid and hyperbolic tangent) are the most preferred functions. 33 of the studies, in which the transfer functions are reported, utilized sigmoid transfer functions in the hidden layers. There is only one study which relied on Gaussian function (Patel and Marwall, 2006). Hence, it can clearly be stated that the use of sigmoid transfer functions in the hidden layers become a standard application in stock market forecasting applications of neural network models.

On the other hand, there are studies which reported the output layer transfer functions. Among these studies only two types of transfer functions are reported: the linear function (Maasoumi, *et.al.*, 2002; Qi, 1999; Wang, *et.al.*, 2003) and sigmoid function (Kohara, *et.al.*, 1997; Refenes, *et.al.*, 1997;).

4.3. Performance Measures

Following the model development, the next step is evaluating the performance of the model. In order to evaluate the success of a neural network model two types of performance measures have been utilized in the literature. The first type of measures is the statistical performance measure and the second type is the financial performance measures.

Statistical measures basically measure the forecasting accuracy of the neural network models beyond the training data. In these statistical methods the forecasting errors are calculated as the difference between the actual value and the forecasted value. Although there are many different statistical methods existing in literature, most common methods are sum of squared errors (SSE), mean absolute error (MAE), mean squared error (MSE), root mean squared error (RMSE), and mean absolute percentage error (MAPE). Moreover, some basic statistical techniques are also utilized like the mean, standard deviation and correlation.

However, the basic objective of the forecasting efforts is to beat the market, or in other words, gaining more returns than the average market return (Refenes, 1995b:70). Although, the statistical methods give a clue about the performance of the neural network models, those methods and techniques do not guarantee the profitability of the trading strategies developed in accordance with the neural network models.

In order to evaluate the profitability of the trading strategies guided by the neural network model, the researchers should consider measuring some profitability of the neural network models. The measures of trading return, return per trade, directional tests, Sterling and Sharpe ratios are basically used to measure the profitability of the neural network models.

As presented in Appendix I, most of the studies relied only on the statistical performance measures. Total of 37 studies utilized the statistical measures for evaluating the model performance (See Brownstone, 1996; Egeli, *et.al.*, 2003; Çinko, *et.al.*, 2007; Kim, *et.al.*, 2007). On the other hand, only 9 studies only on the financial performance measures (See Chandra, *et.al.*, 1999; Lam, 2004; Leigh, *et.al.*, 2005). Moreover, in 15studies, both statistical and financial measures were utilized for evaluation of the model performance (See Rodriguez, *et.al.*, 2005; Ghazali, *et.al.*, 2006).

5. Forecasting ISE-30 Returns with Neural Network Models

The ISE-30 index of the Istanbul Stock Exchange has been calculated with the base value of 976 since December 27, 1996. The ISE-30 index includes the blue-chip stocks of the ISE, and regarded as one of the important economic indicators.

This section will investigate the predictability of daily returns of the ISE-30 index by the use of neural network models. The data and methodological considerations are provided in the following sections.

5.1. Data

The daily data sets used in this study was obtained from the Istanbul Stock Exchange (ISE) through official correspondence. The data sets were including daily closing prices and trading volume information for ISE-30 index between 1 January 2006 and 31 December 2007.

In order to reach a stationary data at some level by removing the long-term trends and reducing the effect of outliers, original data was transformed with logarithm operator. Moreover, transformation could reduce volatility of the data set. The transformation of the price data was achieved by:

$$r_t = \ln\left(y_t \big/ y_{t-1} \right) \qquad (4)$$

where, r_t was denoting the return at time t, and y_t, y_{t-1} was the stock prices for time t and t-1 respectively.

The lagged stock returns, lagged index maximum and minimum values, the moving averages of stock returns, maximum and minimum values of stocks constituted the input variables to the model. These variables were determined in accordance with the applications in literature. The list of input variables provided in *Table 4*.

Table 4: List of input variables

1.	Lagged stock return for 1 day
2.	Lagged stock return for 3 days
3.	Lagged stock return for 5 days
4.	Lagged change in index maximum for 1 day.
5.	Lagged change in index maximum for 3 days.
6.	Lagged change in index maximum for 5 day.
7.	Lagged change in index minimum for 1 day.
8.	Lagged change in index minimum for 3 days.
9.	Lagged change in index minimum for 5 day.
10.	Moving average for stock for 3 days
11.	Moving average for stock for 5 days
12.	Moving average for in index maximum for 3 days
13.	Moving average for in index maximum for 5 days
14.	Moving average for in index minimum for 3 days
15.	Moving average for in index minimum for 5 days

Although, the importance of economic variables on ISE was found in various studies (Ozcam, 1997; Gunes and Saltoglu, 1998), such variables were not included in the study. The reason is that the economic data was mostly available for monthly or quarterly basis, and as a matter of fact it was not suitable for daily analysis (Kim and Chun, 1998).

In order to investigate the performance of neural network models in estimating the ISE-30 index daily returns on monthly basis during 2007, the data set was divided into 12 subsets. Each subset was organized to cover 1 year data. The 1 year period was selected as more current training data was best for optimal forecasting performance (Walczak, 2001). Furthermore, in order to train, validate and test the model, each subset was divided into three: the testing, validation and training data sets. As the objective of the model was assessing monthly performance of the neural network models, the testing data set was composed one month data, which comprised approximately 10% of data for each subset. By following the literature (Deboeck and Cader, 1994; Kaastra and Boyd, 1996; Yao and Tan, 2001), approximately remaining 70% and 20% of data were utilized as training data set and validation data set, respectively.

The moving window approach was adopted in the organization of the data set. When a new data was received, the oldest data from the training data set was dropped and new data was added to the data set. The advantage of the moving window approach was its ability to capture the environmental changes as it utilized more recent data (Morantz, *et.al.*, 2003). Moreover, by utilizing such approach, the forecasting performance of neural network models would be observed on a continuous manner.

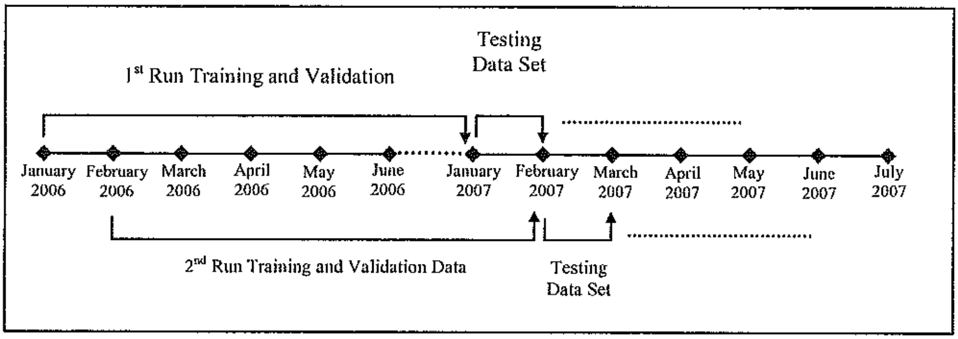

Figure 1: Moving window approach.

Figure 1 illustrates the moving window approach utilized in this study. In the first run to forecast the daily stock returns for January 2007, the data from January 2006 to end of December 2006 would be considered as training (70% of whole subset approximately) and validation (20% of whole subset approximately) data sets. And the model would be tested during the trading days of January 2007. In order to forecast the daily returns for February 2007 in the second run, the January 2006 data would be dropped from data set, but January 2007 data would be included. Although, such data organization would provide continuous forecasts of stock returns, as a result of changing number of trading days in the market, the sizes of training, validation and test data sets would not be equal for each subset. However, such mismatches in the sample periods would not distort overall results of the analysis.

5.2. Methodology

There several neural network models were introduced to the literature, this study utilized three-layer (one hidden layer) multilayer perceptron model with backpropagation algorithm, as these models mathematically proved to be universal approximator for any continuous function (Hornik. *Et al.*, 1989; Cybenko, 1989; Hornik *et al.*, 1990). Moreover, multilayer perceptron model became a standard forecasting tool in neural network literature as over 80% of the studies utilized this model (Gately, 1996; Adya and Collopy, 1998; Thawornwong and Enke 2003).

Four different multilayer perceptron models were trained with different architectures. The first architecture had 3 processing elements (will be denoted as PE-3) in the hidden layer, the second one had 5 processing elements in the hidden layer (will be denoted as PE-5), the third one had 10 processing elements in the hidden layer (will be denoted as PE-10), and the fourth one had 15 processing elements in the hidden layer (will be denoted as PE-15). The number of hidden nodes was determined by following the heuristics advices (Callan, 1999; Zhang, *et.al.*, 1998). Application of more than one heuristic advice was identical to a trial and error procedure. Although, trial and error process was time consuming, it was unfortunately unavoidable (Ma and Khorasani, 2004).

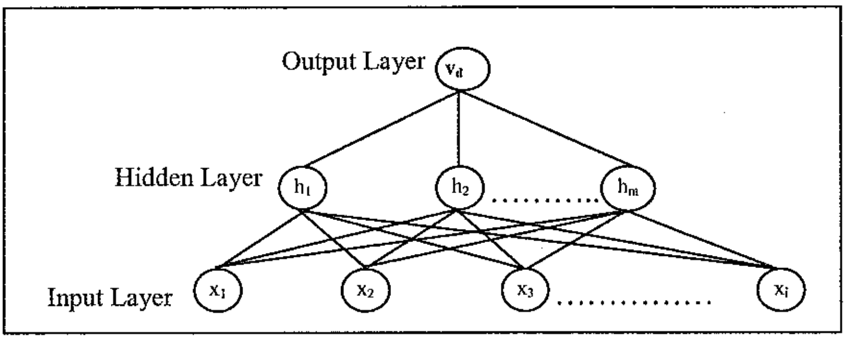

Figure 2: Multilayer perceptron model

Figure 2 presented a simple multilayer perceptron model. The input layer composed of M units of x_i (i = 1,2….N), the hidden layer composed of Z processing entities of h_m (m = 1,2,…..Z), and hence one output layer was denoted by v_d. The output for the model could be presented as:

$$\hat{y} = g\left[\sum_{m=0}^{Z} w_{md}\, g\left(\sum_{i=0}^{M} w_{im} x_i\right)\right]^f \qquad (5)$$

where, w_{im} represented the weights between input and hidden processing units, w_{md} was the weights between hidden processing and the output unit, hence $g(.)$ and $g(.)^f$ was the transformation functions for hidden processing elements and output unit respectively.

The transformation functions for hidden and output layers were hyperbolic tangent, $g(.)$, and linear functions, $g(.)^f$, respectively. The transformation functions are presented mathematically in *Equation 6*:

$$g(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

$$g(x)^f = x \qquad (6)$$

5.3. Performance Measures

Statistical methods basically measure the forecasting accuracy of the neural network models beyond the training data. In these statistical methods the forecasting errors are calculated as the difference between the actual value and the forecasted value. Although there are many different statistical methods existing in literature, the mean absolute error (MAE), mean squared error (MSE), normalized mean squared error and correlation was utilized in this study.

In order to evaluate the profitability of the trading strategies guided by the neural network models, researchers measured the trading return, return per trade, directional tests (HIT), Sterling and Sharpe ratios. In this study the profitability of the neural network models was analyzed by the use of simple trading strategy. The trading strategy was:

$$Trading\ Strategy = \begin{cases} If, y_t \quad > 0; Buy, otherwise\ no\ position \\ If, y_{t+1} > 0; Hold\ position,\ otherwise\ sell \end{cases} \tag{7}$$

Where, y_t was representing the forecasted stock return for trading period t. The profitability of the trading strategy, which was guided by neural network models, was compared to buy-and-hold return of ISE-30 index for each investment period.

5.4. Empirical Findings

The statistical performance measures were utilized to identify the best neural network models in the testing period of the study. *Table 5* presents the finding of the statistical performance measures. Four different performance measures pointed the dominance of the same model for 7 months. These months were January, March, April, June, July, August and November. However, for the remaining 5 months it was unable to identify the best model by relying on the statistical performance measures, as each performance measure presented the dominance of different model. For example, the findings for September presented that the PE-5 model should be the best according to MSE and NMSE performance measure. However, the PE-10 and PE-15 measures should be the best models according to MAE and correlation performance measures, respectively.

While considering the dominance of one model over the others, it is clear from the table that the PE-3 model dominated the other models for 5 months period. The PE-3 model found to be the best model during January, March, July, August and November. For each of these months, four different performance measures pointed out the On the other hand, for April and June, the PE-15 and PE-5 models were identified as the best models by four different performance measures, respectively.

Table 5: Statistical performance measures

Performance Measures	NN Model	January	February	March	April	May	June	July	August	September	October	November	December
MSE	PE-3	**0,00028**	0,00036	**0,00030**	0,00041	0,00021	0,00023	**0,00055**	**0,00092**	0,00041	0,00040	**0,00042**	0,00024
	PE-5	0,00032	**0,00035**	0,00045	0,00032	0,00017	**0,00021**	0,00064	0,00112	**0,00039**	0,00042	0,00048	**0,00021**
	PE-10	0,00035	**0,00035**	0,00060	0,00035	**0,00015**	0,00023	0,00060	0,00127	0,00040	**0,00037**	0,00058	0,00025
	PE-15	0,00029	0,00044	0,00031	**0,00031**	0,00016	0,00022	0,00059	0,00199	0,00041	0,00044	0,00051	0,00028
NMSE	PE-3	**1,09640**	1,03191	**0,92660**	1,29852	1,45454	1,06007	**0,94448**	**1,03997**	1,11286	1,01377	**1,34386**	1,15186
	PE-5	1,25638	**0,99233**	1,38094	0,99963	1,15664	**0,97311**	1,09648	1,26564	**1,04486**	1,05074	1,54322	**1,00090**
	PE-10	1,34829	1,00466	1,84985	1,11410	**1,03147**	1,04891	1,02657	1,43950	1,07696	**0,92725**	1,84218	1,15792
	PE-15	1,11850	1,24746	0,94431	**0,96526**	1,13474	1,00381	1,00691	2,25562	1,09633	1,09082	1,64808	1,32814
MAE	PE-3	**0,01315**	0,01559	**0,01249**	0,01625	**0,00843**	0,01274	0,01900	**0,02513**	0,01251	0,01742	**0,01760**	0,01170
	PE-5	0,01492	0,01500	0,01430	0,01534	0,00972	**0,01246**	**0,01803**	0,02909	0,01299	0,01678	0,01781	**0,01122**
	PE-10	0,01466	**0,01473**	0,01938	0,01588	0,00903	0,01271	0,01871	0,02939	**0,01240**	**0,01620**	0,02043	0,01207
	PE-15	0,01418	0,01630	0,01305	**0,01509**	0,01001	0,01287	0,01813	0,03414	0,01319	0,01801	0,01852	0,01300
r	PE-3	**0,10991**	0,06940	**0,32763**	-0,16813	**0,10177**	-0,06948	**0,29369**	**0,24063**	-0,02723	**0,27356**	**0,35036**	**0,18031**
	PE-5	-0,39387	**0,12973**	0,23446	0,18871	-0,17922	**0,31036**	0,27433	0,09456	0,06608	0,25641	0,26453	0,12743
	PE-10	-0,10737	0,01567	0,12499	-0,25369	0,06363	0,00060	0,28766	-0,22762	0,07830	0,41414	0,04740	-0,01690
	PE-15	-0,09890	-0,02786	0,28968	**0,19462**	0,00123	0,15086	0,23109	-0,28409	**0,28486**	0,26203	0,01025	-0,32564

*** Bold figures represent the best results.**

Table 6 presents the profitability of the neural network guided trading strategies and the market return (ISE -30 index buy and hold return) during the testing periods. At one glance, it is clear that the neural network model guided trading strategies can beat the market or prevent losses for every period under examination. Except for November, the neural network model guided trading strategy dominates the market with a higher positive return. For November, the trading strategy could eliminate the potential losses and limit the losses to -1, 38%; while the market loss was – 6,57%.

Table 6: Profitability of neural network models

	PE-3	PE-5	PE-10	PE-15	Market
January	**7,08**	-4,45	0,96	-1,81	5,90
February	**3,72**	0,20	-3,66	-3,91	0,63
March	**8,82**	7,96	3,62	6,96	4,43
April	-0,01	3,37	1,53	**8,96**	3,78
May	**3,10**	0,58	1,56	1,21	2,77
June	-1,69	**0,76**	-0,24	-0,67	-0,35
July	**15,00**	12,20	6,67	13,40	12,70
August	**4,84**	-1,20	-3,53	-13,62	-5,63
September	2,62	8,66	7,92	**9,23**	8,35
October	7,10	7,79	**16,70**	10,80	6,59
November	-3,98	-3,59	**-1,38**	-7,24	-6,57
December	2,41	**5,37**	1,12	-1,36	2,25
Total return	**49,01**	37,65	31,27	21,95	34,85
Average monthly return	**4,08**	3,14	2,61	1,83	2,90

* **Bold figures represent the best results.**

The PE-3 model can beat the market and other models in 6 month period. These months are January, February, March, May, July and August. On the other hand, PE-5, PE-10 and PE-15 models can dominate the market and the other models in just two month periods. The PE-5 model beat the other models and the market for June and December, where PE-10 model beat the other models and the market for October and November; and hence PE-15 model beat the other models and the market for April and September.

If the performance of the neural network model guided trading strategy was investigated on average monthly return basis, only the PE-3 and PE-5 models could beat the market. The PE-3 model was generated 4,08% return and the PE-5 model was generated 3,14% return on average, while the market average return was 2,90%.

6. CONCLUSION

This study aimed at providing a guideline for stock market applications of neural network models. Total of 63 studies from national and international literature was reviewed to address some modeling issues of the neural network models. Although, there remain several questions to be answered, this study was expected to be a valuable guide for further studies.

Following the analysis of the literature on neural network applications in stock market, this study analyzed the forecasting power of neural network models for ISE-30 index. It was clear that the neural network models were able to beat the market for the testing period under examination.

Although, the PE-3 model dominated the other models in terms of statistical and financial performance measures for most of the testing periods, it was unable to identify one model which constantly beat the other models and the market. Such finding is consistent with the previous literature.

APPENDICES

I. LITERATURE REVIEW: TECHNICAL DETAILS

Article	NN Type	Network Layers (Input-Hidden-Output)	Transfer Functions	Performance Measures
Abdelmouez et.al. (2007)	Mix.	NA	NP	RMSE, MAE, Return, HIT
Altay et.al. (2005)	BPNN	?-15-7-1 ?-10-10-1 ?-80-1	Tahn	RMSE, MAE, Theil's U, HIT, Return
Armano et.al. (2005)	BPNN	10-8-3-3	NP	Return, Sharpe ratio
Avcı (2007)	BPNN	14-7,14,28-1 10-7,14,28-1	Than	MSE, NMSE, MAE, Return
Avcı et.al. (2007)	BPNN	5-5,10,12-1	Sig	MSE, NMSE, MAE,HIT, Return
Baba et.al. (2002)	BPNN	8-15-2	Tanh	Return
Brownstone (1996)	BPNN	49-8 –1	NP	MSE, HIT, RMSE
Cao, et.al. (2005)	BPNN	3-?-1 1-?-1	Log	MAD, MAPE, MSE, STD
Chandra et.al. (1999)	BPNN	?-?-?-1 ?-?-1	NP	Return
Chen, et.al. (2003)	PNN	4, 5, 6-68-2-1	NP	HIT, Return
Chen (2006)	BPNN	NP	NP	MSE, NMSE, MAPE, COR
Chenoweth et.al. (1996a)	BPNN	6-4-1 8-3-1	Tanh	Return, Return per Trade

Chenoweth et.al. (1996b)	BPNN	6-4-1	Tanh	Return, Return per Trade
Çinko et.al.(2007)	BPNN	NP	NP	MSE, NMSE, HIT
Darrat et.al. (2000)	BPNN	?-?-1	Log	MAE, RMSE, Theil
Desai et.al. (1998a)	BPNN	11-15-15-1	Sig	COR, MSE
Desai et.al. (1998b)	BPNN	11-26-1	NP	Return, STD, MSE, COR, MAE, MAPE
Diler (2003)	BPNN	7-6,7,8,9,10,11,12,15-1	NP	RMSE, HIT
Dropsy (1996)	BPNN	8-?-1	NP	HIT, MxAE, RMSE
Eakins et.al. (2003)	BPNN	7-3-1	Tan	Return Sharpe R.
Edelman et.al. (1999)	BPNN	10-5-3-1	Sig-Sig	COR, Sharpe R.
Egeli et.al. (2003)	BPNN GNN	8-?-1 8-?-?-1 8-?-?-?-?-1	NP	R^2, MRPE
Gencay (1996)	BPNN	?-?-?	Sig	MSE
Gencay et.al.(1998)	BPNN	?-?-?	Log	HIT, MSE
Gencay (1998)	BPNN	?-?-?	NP	HIT, Return, Sharpe R.
Ghazali et.al. (2006)	BPNN FLNN PSNN RPNN	NP	NP	NMSE, Return, SNR
Groşa et.al. (2005)	BPNN	4-26-1	Tanh	RMSE, COR, MAP, MAPE
	DBNN	NP	NP	
Kanas et.al (2001)	BPNN	3-6-1	Log	RMSE
Kanas (2003)	BPNN	?-8-1	Log	HIT
Karatlı et.al. (2004)	BPNN	7-7-3-3-1	NP	RMSE

Kim et.al. (1998)	BPNN	5-15,20-1	Sig	HIT
	RNN	5-5-1		
	APNN	5-6-7-1		
Kim (2006)	GANN	12-12-1	Sig	HIT
Kim et.al. (2007)	ATNN TDNN RNN	?-8-1	Log	MSE
	GANN	?-5,8-1		
Kohara et.al.(1997)	BPNN	6-6-1 5-5-1 15-15-1	Sig-Sig	Return, MAE, Variance, COR
	RNN	5,6-20-1		
Lam (2004)	BPNN	16,27,32,48-10-1 16,27,32,48-10-7-1 16,27,32,48-10-7-3-1	NP	Return
Leigh et.al. (2002)	BPNN	22-6-1	NP	NA
Leigh et.al. (2005)	BPNN	3-8-?	NP	Return
Lim et.al. (1998)	BPNN	16,15,10,8-2-1	Sig-Lin	RMSE, MAE, MAPE
Leungh et.al. (2000)	BPNN	6,5-10,6-1	NP	HIT, Return
	PNN	1,2,5-NP		
Maasoumi et.al.(2002)	BPNN	9-8-1	Log-Lin	RMSE, MAE, MAPE, COR, HIT
Moreno et.al. (2007)	GANN	3,5-4,5,8-1	NP	MSE, HIT, Return
Ng et. al. (2000)	BPNN ANN RBF	NP	Sig	Error variance
O'Connor et.al. (2006)	BPNN	6,8,11,26,31-9,10,37,39-1	NP	RMSE, Error points, HIT

Olson et.al. (2003)	BPNN	NP	Tanh	HIT, Return
Patel et.al. (2006)	BPNN RNN	CP-CP-4	Tanh Gaus	CM
Phua et.al. (2001)	BPNN	15-NP	NP	HIT
Qi, (1999)	BPNN	9-8-1	Log-Lin	RMSE, MAE, MAPE, COR, HIT
Refenes et.al. (1997)	BPNN	6-?-1	Sig-Sig	RMSE
Rodriguez et. al. (2000)	BPNN	9-4-1	Log	Return, Ideal Profit, Sharpe Ratio, HIT
Rodriguez et. al.(2005)	BPNN JCN RNN	?-?-?	Tanh	MAE, MAPE, RMSE, Theil, HIT, Return, Sharpe Ratio
Roman et.al. (1996)	BPNN	4-10-5-2-1	NP	HIT, Return
	RNN	4-10-4-1		
Saad et.al. (1998)	RNN	1-8-1	Tanh	HIT
	TDNN	3,4-5,10-1	Tanh	
	PNN	29-?-?-1	NP	
Safer (2001)	BPNN	12-5,9-1	NP	STD
Safer (2003)	BPNN	13-5,9-1	NP	STD
Schierholt et.al. (1996)	BPNN	3-?-3 3-16-8-3	Sig	MSE, HIT
	PNN	NP		
Shi et.al. (1999)	BPNN	?-?-?-1	NP	MSE
Singh (1999)	BPNN	5-10,5,15,15,20,10-1 5-10,20,25,15,20,15-1	NP	RMSE, MAPE, GRMSE, GMRAE, HIT
Sitte et.al.(2000)	TDNN	?-2,4,8,16,32-1	Sig	RMSE

Stansell et.al. (2003)	BPNN	19-?-?	NP	HIT
Sun et.al. (2005)	RBF	3-5,6,16,26,-1	NP	HIT
Walczak (1999)	BPNN	6,12,18,24,30-6,12,18,24,30-1	NP	HIT
Wang et.al. (2003)	TDNN	5,6,8-NP	Sig-Lin	MAPE, HIT
Yao et.al. (1999)	BPNN	5,6-3,4,5-1 5,6-3,4-2,3-1	Tanh	NMSE, HIT

NA: Not applicable
NP: Not provided
CP: Cannot be presented in the table because of page limitations

Neural Network Types	Performance Measures
ANN: ADALINE Neural Network Model	CM: Confusion matrix
APNN: Arrayed Probabilistic N. N. Model	COR: Correlation coefficient
ATNN: Adaptive Time Delay N. N. Model	GMRAE: Geometric Relative Absolute Error
BPNN: Backpropagation Neural Network Model	GRMSE: Geometric Root Mean Square Error
DBNN: Difference Boosting Neural Network	HIT: The hit rate or the proportion of the correct forecasts, positive or negative or overall
JCN: Jump Connection Nets	MAD: Mean absolute deviation
GANN: Genetic Algorithm Optimized N.N. Model	MAE: Mean absolute error
PNN: Probabilistic Neural Network Model	MAP: Maximum absolute percentage error
RNN: Recurrent Neural Network Model	MAPE: Mean absolute percentage error
RBF: Radial Basis Function Neural Network Model	MxAE: Maximum absolute error
TDNN: Time Delay Neural Network Model	MSE: Mean squared error
FLNN: Functional Link Neural Networks	NMSE: Normalized Mean-Squared Error
PSNN: Pi-Sigma Neural Networks	RMSE: Root mean squared error
RPNN: Ridge Polynomial Neural Networks	SNR: Signal to noise ratio
	STD: Standard deviation

II. LITERATURE REVIEW: INPUT VARIABLES

Article	Input Variables	Data Frequency	Forecast Horizon	Forecasted Market	Data Horizon
Abdelmouez et.al. (2007)	Lagged index values	Daily	5 Days	223 US sector indexes	1988-2001
Altay et.al. (2005)	Lagged index returns Lagged percentage change in min. and max value	Daily / Weekly / Monthly	Daily / Weekly / Monthly	ISE-30 / ISE-All	1997-2005
Armano et.al. (2005)	Differences of averages Rate of change Relative strength index Convexity Up trend Down trend Lagged index values	Daily	Daily	COMIT / S&P500	1992-2000 / 1993-2001
Avcı (2007)	Lagged index return Lagged change in volume Moving average for index return and change in volume Moving average for index and volume	Sessional / Daily	Sessional / Daily	ISE – 100	1996-2005
Avcı et.al. (2007)	Lagged index return Moving average for index return	Daily	Daily	ISE -100 / PX50 / BUX / WIG20/ BETI / SBI	2000-2006

Baba et.al. (2002)	Exchange rate (yen-Dollar) TOPIX index Foreign trader transactions (buy-sell) Changes of TOPIX indeks Turnover in Tokyo Stock Exchange Total money used in the TSE PBR in the Tokyo Stock Market	NP	4 weeks	TOPIX – Nikke225	1996-1997
Brownstone (1996)	Total of 54 inputs (details NP)	5 days 25 days	5 days 25 days	FTSE-100	1985-1991
Cao, et.al. (2005)	Market risk factor (Beta) Market Capitalization Book to market value	Daily	Daily	367 SHSE Firms	1999-2002
Chandra et.al. (1999)	Monthly index returns 5 year Treasury security returns	Monthly	Monthly	Wilshire 500	1971-1996
Chen, Et.al. (2003)	Term structure of interest rates Short term interest rates Lagged index returns Government consumption level Private consumption level Gross national product Gross domestic product Consumer price index Industrial production level	Monthly	Monthly	TSEI	1982-1992
Chen (2006)	Lagged index values	Daily	Daily	TAIEX	2001-2003

Chenoweth et.al. (1996a)	Return on 30 year U.S. Gov.Bonds Lagged rate of change in return on U.S. T- Bill Return and lagged return on S&P composite index Return and lagged return on 30 year U.S. Govern. Bonds U.S. Treasury Bill index The Consumer Price Index	Daily / Monthly	Daily / Monthly	S&P -500	1985-1993
Chenoweth et.al. (1996b)	S&P500 index return and lagged returns Lagged U.S. Treasury rate 30 year government bond rate	Daily	Daily	S&P -500	1982-1993
Çinko et.al.(2007)	Lagged index return Lagged change in volume Moving average for index return Moving average for change in volume Moving average for index and volume	Sessional / Daily	Sessional / Daily	ISE -100 /	1996-2005
Darrat et.al. (2000)	Closing index prices	Weekly	Weekly	SHSE / SHZSE	1990-1998
Desai et.al. (1998a)	Yield of 1 month T-bill Yield of 2 -6 month T-bill minus the yield of 1 month T-bill Lag of 1 month holding period return on a 2 month T-bill less return on a 1 month T-bill Yield on Moody's Baa rated bond minus the 1 month T-bill yield The ratio of the real S&P Comp. over its long term average The negative algorithm of the share prices averaged across the last quintile of the NYSE The dividend and earnings yield on S&P Comp. Yield on the long and intermediate-term Gov. Bond	Monthly	Monthly	S&P Com. / Dimensional Fund Adv. Small Comp.	1959-1993

Desai et.al. (1998b)	Same as Desai et.al. (1998a)	Monthly	Monthly	S&P Com.	1959-1990
Diler (2003)	10 days moving average Weighted moving averages Momentum Stochastic BGERSI MACD Daily ISE-100 return	Daily	Daily	ISE-100	1990-2003
Dropsy (1996)	Ratio of government spending to GDP Growth rate of money supply (M2) Short-term interest rates Spread 10 year Gov. Bond yields and short term int. rates Consumer Price Index Ratio of trade balance to GDP Nominal effective rate of depreciation of domestic currency in terms of foreign currency Real oil price inflation rate	Monthly	Monthly	Germany / U.S. / U.K. / Japan stock markets	1971-1990
Eakins et.al. (2003)	Market Capitalization Dividend yield Price/Sales ratio Price/Earnings ratio Price/Book ratio Price to cash flow ratio	Yearly	Yearly	Stocks listed on Compustat data base	1975-1996

Edelman et.al. (1999)	Lagged index returns (1-2-3-4-5 day) High minus low index level of last day Close minus low index level of last day Moving average of past 20 days' returns Standard deviation of the past 20 days' returns 90-day bank bill rate.	Daily	Daily	AOI	1990-1997
Egeli et.al. (2003)	Previous day's index value Previous day's TL/USD exchange rate Previous day's simple interest rate weighted average overnight Dummy variables for days of the week	Daily	Daily	ISE-100	2001-2003
Gencay (1996)	Moving average of raw price data	Daily	Daily	DJIA	1963–1988
Gencay et.al.(1998)	Moving average of raw price data Moving average of volume	Daily	Daily	DJIA	1963–1988
Gencay (1998)	Lagged index returns	Daily	Daily	DJIA	1963–1988
Ghazali et.al. (2006)	Exponential moving average Lagged closing price	Daily	5 days	IBM stock	1961-1962
Groşa et.al. (2005)	Index opening, high, low, close	Daily	Daily	Nasdaq-100 / NIFTY	1995-2002 1998-2001
Kanas et.al (2001)	Lagged Trading volume Stock dividends Lagged stock index returns	Monthly	Monthly	FTSE/ DJIA	1980–2000
Kanas (2003)	Lagged change in dividends	Yearly	Yearly	S&P Composite	1872-1999

Karatlı et.al. (2004)	T-Bill interest rates Gold coin price Consumer price index Industry production index Interest on savings deposits Exchange rate	Monthly	Monthly	ISE-100	1990–2002
Kim et.al. (1998)	Stock Price Index Total Return Index Including Dividends Dividend Yield Trading Volume of the Stock Market Price/Earning Ratio	Daily	Daily	SESPI	1985–1996
Kim (2006)	Stochastic (%K, %D, slow %D) Momentum Rate of change (ROC) Larry William's %R A/D Oscillator Disparity 5 and 10 days Price oscillator Commodity channel index Relative strength index	Daily	Daily	KOSPI	1991-1998
Kim et.al. (2007)	Index value	Daily	Daily	KOSPI-200	1997-1999
Kohara et.al.(1997)	TOPIX close and lagged TOPIX close Dollar/Yen rate and lagged rate 3-month interest rate and lagged rate The price of crude oil and lagged price Dow-Jones 30 industrial stocks and lagged value Event Knowledge	Daily	Daily	TOPIX	1989–1991

Lam (2004)	Total of 16 financial ratio Total of 11 economic data	Yearly	Yearly	364 firms from S&P	1985-1995
Leigh et.al. (2002)	Price window Volume window Price fit (Total of 10 fit values) Volume fit (Total of 10 fit values)	Daily	20 days	NYSE Composite	1981–1996
Leigh et.al. (2005)	Change in prime interest rates Change in NYSE Comp. Volume spike	Daily	Daily	NYSE Composite	1985–1999
Leungh et.al. (2000)	Short – long term interest rates Lagged index returns	Monthly	Monthly	S&P 500 / FTSE 100 / Nikke I225	1967–1995
Lim et.al. (1998)	Lagged index returns	Daily	Daily	ASX All ordinaries	1990–1996
Maasoumi et.al. (2002)	Dividend yield Earnings-price ratio One month T-bill rate Twelve months Treasury bond rate Inflation rate Change in industrial output Growth rate of narrow money stock	Monthly	Monthly	S&P 500	1960-1992
Moreno et.al. (2007)	Lagged index returns	Daily	Daily / Weekly	Total of 49 international indexes	1995-2001
Ng et.al. (2000)	Lagged stock prices Lagged change in NASDAQ index	Daily	Daily	GEM Selected Stocks	Date of listing – 2000

O'Connor et.al. (2006)	Index opening and lagged opening Index moving averages Index gradients Lagged crude oil price Lagged USD/YEN, USD/CAN, USD/GBP exchange rates	Daily	Daily	DJIA	1987-2002
Olson et.al. (2003)	61 accounting ratios Book to market value Earnings-price ratio Price sales ratio	Yearly	Yearly	Canadian stocks	1976-1993
Patel et.al. (2006)	Lagged index closing Moving average of index closing	Daily	Daily	DJIA / JSE All Share / Nikkei 225/ Nasdaq 100	2004-2005
Phua et.al. (2001)	STI Open – High – Low - Close STI trading volume DJIA NASDAQ Heng Seng Index Nikkei 225	Daily	Daily	SESSTI	1998–2000
Qi, (1999)	Dividend yield Earnings-price ratio One month T-bill rate Twelve months Treasury bond rate Inflation rate Change in industrial output Growth rate of narrow money stock	Monthly	Monthly	S&P 500	1954- 1992

Refenes et.al. (1997)	Long term and short term interest rates Earnings per share Dollar/French Franc exchange rate Price to earnings ratio	Daily	Daily	CAC-40	NP
Rodriguez et. al. (2000)	Lagged index returns	Daily	Daily	IGMB	1966–1997
Rodriguez et. al. (2005)	Index daily closing	Daily	Daily	IBEX-35	1989–2000
Roman et.al. (1996)	Weekly index trends	Weekly	Yearly	FTSE-100 / S&P 500 / TOPIX / Hangseng / Toronto 300	1989–1992
Saad et.al. (1998)	Daily closing prices (modified)	Daily	Short term stock trends	Various U.S. stocks	NA
Safer (2001)	Number of new shareholders Eight week ratio of sells/buys Median insider shares bought and sold Av. return from the previous 2 and 4 months to the subsequent 3-6-9 months of insider buy transaction Number of buy and sell transaction Number of shares bought and sold Dollar value of buy and sell transactions	Monthly	Monthly	Various U.S. stocks	1993–1997
Safer (2003)	Same as Safer (2001)	Monthly	Monthly	Various U.S. stocks	1993–1997
Schierholt et.al. (1996)	Lagged index change	Daily	Daily	S&P	1994–1995

Stock Market Forecasting with Artificial Neural Network Models

Shi et.al. (1999)	Historical stock prices Forecasts by other models	Daily	Daily	IBM stock	1961–1961
Singh (1999)	Lagged index returns	Daily	Daily	DAX / FTSE / CAC / SWISS / EOE / S&P	1988–1996
Sitte et.al.(2000)	Index values	Daily	Daily	S&P 500	1973-1994
Stansell et.al. (2003)	19 economic variables	Monthly	Monthly	S&P sector indexes	1994–2000
Sun et.al. (2005)	Lagged price data	Daily	Daily	S&P 500 / CISSE	1993-1995 and 1999 - NP
Walczak (1999)	Lagged index closing value Lagged index volume	Daily	5 days	DBS50 / DJA / Nikkei	1994–1995
Wang et.al. (2003)	Index return Trading volume	Daily	Daily	S6P500 / DJI	1990–2002
Yoa (1999)	Index value Lagged index value 5 -10 days moving average Relative strength index Momentum	Daily	Daily	KLCI	1984–1991

REFERENCES

Abdelmouez, G., Hashem, S.R., Atiya, A.F. and El-Gamal, M.A., (2007), "Neural Network vs. Linear Models for Stock Market Sectors Forecasting", *Proceedings of International Joint Conference on Neural Networks*, Orlando, , USA, August 12-17, 1-5.

Aktaş R., Doğanay, M. and Yıldız, B., (2003), "Mali Başarısızlığın Öngörülmesi: İstatistiksel Yöntemler ve Yapay Sinir Ağı Karşılaştırması", *Ankara Üniversitesi SBF Dergisi*, 58(4), 1-25.

Altay, E. and Satman, M.H., (2005), "Stock Market Forecasting: Artificial Neural Networks and Linear Regression Comparison in an Emerging Market". *Journal of Financial Management and Analysis*, 18(2), 18-33.

Armano, G., Marchesi, M. and Murru, A., (2005), "Hybrid Genetic-Neural Architecture for Stock Indexes Forecasting", *Information Sciences*, 170, 3-33.

Avcı, E., (2007), "Forecasting Daily and Sessional Returns of the ISE-100 Index with Neural Network Models", Doğus Üniversitesi Dergisi, 2007, 8(2), 128-142.

Avcı, E. and Çinko, M., (2007), "Endeks Getirilerinin Yapay Sinir Ağları Modelleri ile Tahmin Edilmesi: Gelişmekte Olan Avrupa Borsaları Uygulaması", *11. Ulusal Finans Sempozyumu*, 17-20 Ekim 2007, Zonguldak.

Azoff, E.M., (1994) **Neural Network Time Series Forecasting of Financial Markets**, Sussex, John Wiley & Sons Inc..

Baba, N. and Yanjun, Y., (2002), "Utilization of Neural Networks & Genetic Algorithms for Constructing a Reliable Decision Support System which Deals Nikkei-225", *SICE 2002*, Aug. 5-7, Osaka, 1603-1608.

Benli, Y.K., (2002), "Finansal Başarısılığın Tahmininde Yapay Sinir Ağı Kullanımı ve İMKB'de Uygulama", *Muhasebe Bilim Dünyası Dergisi*, 4(4), 17-30.

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.

Bishop, C.M., (1994), "Neural Networks and Their Applications", *Review of Scientific Instruments*, 65(6), 1803-1830.

Bishop, C.M., (1995), **Neural Networks for Pattern Recognition**, New York, Clarendon Press,.

Boyacioğlu, M.A. and Kara, Y., (2006), "Türk Bankacılık Sektöründe Finansal Güç Derecelerinin Tahmininde Yapay Sinir Ağları ve Çok Değişkenli İstatistiksel Analiz Tekniklerinin Performanslarının Karşılaştırılması", *10. Ulusal Finans Sempozyumu*, 01-04 Kasım 2006, İzmir.

Brownstone, D., (1996), "Using Percentage Accuracy to Measure Neural Network Predictions in Stock Market Movements", *Neurocomputing*, 10, 237-250.

Callan, R., (1999), **The Essence of Neural Networks**, Essex, Prentice Hall.

Cao, Q., Leggio, K.B. and Schniederjans, M.J., (2005), "A Comparison Between FAMA and French's Model and Artificial Neural Networks in Predicting the Chinese Stock Market", *Computers & Operations Research*, 32, 2499-2512.

Chandra , N. and Reeb, D.M., (1999), "Neural Networks in a Market Efficiency Context", *American Business Review*, 39-44.

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

Chen, K.Y., (2006), "Evolutionary Support Vector Regression Modeling for Taiwan Stock Exchange Market Weighted Index Forecasting", *Journal of American Academy of Business*, 8(1), 241-247.

Chenoweth, T. and Obradovic, Z., (1996a), "A Multi-Component Nonlinear Prediction System for the S&P 500 Index", *Neurocomputing*, 10, 275-290.

Chenoweth, T., Obradovic, Z. and Lee, S.S., (1996b), "Embedding Technical Analysis into Network Based Trading System", *Applied Artificial Intelligence*, 10, 523-541.

Çinko, M. and Avcı, E., (2007), "A Comparison Of Neural Network And Linear Regression Forecasts Of The ISE-100 Index", Marmar*a Üniversitesi Sosyal Bilimler Enstitüsü Öneri Dergisi*, 7(28), 301-307.

Cybenko, G., (1989), "Approximation by Superpositions of a Sigmoidal Function", *Mathematics of Control, Signal and Systems*, 2, 303-314.

Darrat, A.F. and Zhong, M., (2000), "On testing the Random -Walk Hypothesis: A Model Comparison Approach", *The Financial Review*, 35, 105-124.

Deboeck, G.J. and Cader, M., (1994), ***Pre- and Postprocessing of Financial Data***, Deboeck, G.J. (Ed.), **Trading on the Edge**, (27-45), New York, John Wiley & Sons Inc..

Desai, V.S. and Bharati, R., (1998a), "A Comparison of Linear regression and Neural Network Methods for Predicting Excess Returns on Large Stocks", *Annals of Operations Research*, 78, 127-163.

Desai, V.S. and Bharati, R., (1998b), "The Efficiency of Neural Networks in Predicting Returns on Stock and Bond Indices", *Decision Sciences*, 29(2), 405-425.

Diler, A.İ., (2003), "İMKB Ulusal-100 Endeksinin Yönünün Yapay Sinir Ağları Hata Geriye Yayma Yöntemi ile Tahmin Edilmesi, *İMKB Dergisi*, 25-26, 65-81

Dropsy, V., (1996), "Do Macroeconomic Factors Help in Predicting International Equity Risk Primia?: Testing the Out-of-Sample Accuracy of Linear and Nonlinear Forecasts", *Journal of Applied Business Re***search**, 12(3), 120-127.

Eakins, S.G. and Stansell, S.R., (2003), "Can Value Based Stock Selection Yield Superior Risk-Adjusted Returns: An Application of Neural Networks", *International Review of Financial Analysis*, 12, 83-97.

Edelman, D., Davy, P. and Chung, Y.L., (1999), "Using neural Network Prediction to Arbitrage the Australian All-Ordinary Index", *IEEE 3rd International Conference on Knowledge-Based Intelligent Information Systems*, 31st Aug. – 1st Sept. 1999, Adelaide, Australia, 166-169.

Egeli, B., Özturan, M., and Badur, B., (2003)., "Stock market prediction using artificial neural networks", *Proceedings of the 3rd Hawaii International Conference on Business*, Hawai, USA.

Fadlalla, A. and Lin, C.H., (2001), "An Analysis of the Applications of Neural Networks in Finance", **Interfaces**, 31, 112-122.

Fausett, L., (1991), **Fundamentals of Neural Networks: Architectures, Algorithms and Applications**, New Jersey, Prentice Hall.

Fine, T.L., (1999), **Feedforward Neural Network Methodology**, New York, Springer Press.

Garson, G.D., (1998), **Neural Networks: An Introduction Guide to Social Scientists**, London, SAGE Publications.

Gencay, R., (1996), "Non-Linear Prediction of Security Returns with Moving Average Rules", *Journal of Forecasting*, 15, 165-174.

Gencay, R., (1998), "Optimization of Technical Trading Strategies and the Profitability in the Stock Markets", *Economic Letters*, 59, 249-254.

Gencay, R. and Stengos, T., (1998), "Moving Average Rules, Volume and the Predictability of Stock Returns with Feedfoward Networks", *Journal of Forecasting*, 17, 401-414.

Ghazali, R., Hussain, A. and El-Deredy W., (2006), "Application of Ridge Polynomial Neural Networks to Financial Time Series Prediction", *Proceedings of 2006 International Joint Conference on Neural Networks*, Vancouver, Canada, July 16-21, 93-920.

Groşan, C., Abraham, A., Ramos, V. and Han, S.Y., (2005), "Stock Market Prediction Using Multi Expression Programming", Proceedings of Artificial Intelligence Conference, 5-8 December, Portuguese, 73-78.

Gurney, K., (1997), **An Introduction to Neural Networks**, London, UCL Press.

Hamid, S.A. and Iqbal, Z., (2004), "Using Neural Networks for Forecasting Volatility of S&P500 Index Futures Prices", *Journal of Business Research*, 57, 1116-1125.

Haykin, S., (1994), **Neural Networks: A Comprehensive Foundation**, New Jersey, Prentice Hall.

Hg, H.S. and Lam, K.P., (2000), "Modeling of NASDAQ-GEM Stock Price Relationship Using Neural Network, *Proceedings of 2000 IEEE International Conference on Management of Innovation and Technology (ICMIT2000)*, 12-15 Nov., Singapore, 41-46.

Hill, T., Marquez, L., O'Connor, M. and Remus, W., (1994), "Artificial Neural Network Models for Forecasting and Decision Making", *International Journal of Forecasting*, 5, 5-15.

Hornik, K., Stinchcombe, M. and White, H., (1989), "Multilayer Feedforeward Networks Are Universal Approximators", *Neural Networks*, 2, 359-366.

Hornik, K., Stinchcombe, M. and White, H., (1990), "Universal Approximation of an Unknown Mappings and its Derivatives Using Multilayer Feedforward Neural Networks", *Neural Networks*, 3, 551-560.

Hush, D.R. and Horne, B.G., (1993), "Progress in Supervised Neural Networks: What's new since Lippmann?", *IEEE Signal Processing Magazine*, 8-39.

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

Kanas, A. and Yannopoulos, A., (2001), "Comparing Linear and Nonlinear Forecasts for Stock Returns", *Internationals Review of Economics and Finance*, 10, 383-398.

Kanas, A., (2003), "Non-Linear Forecasts of Stock Returns", *Journal of Forecasting*, 22, 299-315.

Karaatlı, M., Güngör, İ., Demir, Y. and Kalaycı, Ş. (2005), "Hisse Senedi Fiyat Hareketlerinin Yapay Sinir Ağları Yöntemi ile Tahmin Edilmesi", *Balıkesir Üniversitesi İİBF Dergisi*, 2(1), 22-48.

Kim, S.H. and Chung, S.H., (1998), "Graded Forecasting Using Array of Bipolar Predictions: Application of Probabilistic Neural Networks to a Stock Market Index", *International Journal of Forecasting*, 14, 323-337.

Kim, K., (2006), "Artificial Neural Networks with Evolutionary Instance Selection for Financial Forecasting", Expert Systems with Applications, 30, 519–526.

Kim, H., and Shin, K., (2007), "A Hybrid Approach Based on Neural Networks and Genetic Algorithms for Detecting Temporal Patterns in Stock Markets", Applied Soft Computing, 7, 569–576

Kohara, K, Ishikawa, T., Fukuhara, Y. and Nakamura, Y., (1997), "Stock Price Prediction Using Prior Knowledge and Neural Networks, *Intelligent Systems in Accounting, Finance and Management*, 6, 11-22.

Kwok, T.Y. and Yeung, D.Y., (1999), "Constructive Algorithms for Structure Learning in Feedforward Neural Networks for Regression Problems", *IEEE Transactions on Neural Networks*, 20, 1-16.

Lam, M., (2004), "Neural network techniques for financial performance prediction: integrating fundamental and technical analysis", *Decision Support Systems*, 37, 567– 581.

Leigh, W., Purvis, R. and Ragusa, J.M., (2002), "Forecasting the NYSE Composite Index with Technical Analysis, Pattern Recognizer, Neural Network, and Genetic Algorithm: A Case Study in Romantic Decision Support", *Decision Support Systems*, 32, 361-377.

Leigh, W., Hightower, R. and Modani, N., (2005), "Forecasting the New York Stock Exchange Composite Index with Past Price and Interest Rate on Condition of Volume Spike", *Expert Systems with Applications*, 28, 1–8.

Lim, G.C. and McNelis, P.D., (1998) "The Effect of the Nikkei and the S&P on the All-Ordinaries : A Comparison of Three Models", *International Journal of Finance and Economics*, 3, 217-228.

Leung, M.T., Daouk, H. and Chen, A., (2000), "Forecasting Stock Indices: A Comparison of Classification and Level Estimation Models", *International Journal of Forecasting*, 16, 173-190.

Lippmann, R.P., (1987), "An Introduction to Computing with Neural Nets", *IEEE ASSP Magazine*, 4-22.

Ma, L. and Khorasani, K., (2004), "New Training Strategies for Constructive Neural Networks with Application to Regression Problems", *Neural Networks*, 17, 361-385.

Maasoumi, E. and Racine, J., (2002), "Entropy and Predictability of Stock Market Returns", *Journal of Econometrics*, 107, 291-312.

Marquez, L., Hill, T., O'Connor, M. and Remus, W., (1992), "Neural Network Models for Forecast: A Review", *IEEE Proceedings 25th Hawaii International Conference on System Sciences*, 494-498.

Masters, T., (1993), **Practical Neural Network Recipes in C++**, San Diego, Academic Press Inc..

McMenamin, J.S., (1997), "A Primer on Neural Networks for Forecasting", *The Journal of Business Forecasting Methods and Systems*, 16(3), 17-23.

Medsker, L., Turban, E. and Trippi, R.R., (1993), ***Neural Network Fundamentals for Financial Analysts***, Trippi, R.R. and Turban, E. (Eds), **Neural Networks in Finance and Investing** (3-27), Probus Pub. Comp..

Moreno, D. and Olmeda, I., (2007), "Is the Predictability of Emerging and Developed Stock Markets Really Exploitable?", *European Journal of Operational Research*, 182, 436–454.

Ng, H.S. and Lam, K.P., (2000), "Modeling of NASDAQ-GEM Stock Price Relationship Using Neural Network", *Proceedings of the 2000 IEEE International Conference on Management of Innovation and Technology (ICMIT 2000)*, 1, 41 – 46.

O'Connor, N. and Madden, G., (2006), "A Neural Network Approach to Predicting Stock Exchange Movements Using External Factors", *Knowledge-Based Systems*, 19, 371–378.

O'Leary, D.E., (1998), "Using Neural Networks to Predict Corporate Failure", *International Journal of Intelligent Systems in Accounting, Finance & Management*, 7, 187-197.

Olson, D. and Mossman, C., (2003), "Neural Network Forecasts of Canadian Stock Returns using Accounting Ratios", *International Journal of Forecasting*, 1, 1-13.

Patel P. B. and Marwall, T., (2006), "Forecasting Closing Price Indices Using Neural Networks", *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*, October 8-11, 2006, Taipei, Taiwan, 2351-2356.

Perez, M., (2006), "Artificial Neural Networks and Bankruptcy Forecasting: A State of the Art", *Neural Computing & Applications*, 15, 154–163.

Phua, P.K.H., Ming, D. and Lin, W., (2001), "Neural Network with Genetically Evolved Algorithms for Stock Prediction", *Asia-Pacific Journal of Operational Research*, 18, 103-107.

Qi, M., (1999), "Nonlinear Predictability of Stock Returns Using Financial and Economic Variables", *Journal of Business and Economics Statistics*, 17(4), 419-429.

Qi, M and Zhang, G.P. (2001), "An investigation of Mode Selection Criteria for Neural Network Time Series Forecasting", European Journal of Operational Research, 132, 666-680.

Reed, R., (1993), "Pruning Algorithms-A Survey", *IEEE Transactions on Neural Networks*, 4(5), 740-747.

Refenes, A.N., Bentz, Y., Bunn, D.W. and Burgess, A.N., (1997), "Financial Time Series Modelling With Discounted Least Squares Backpropagation", *Neurocomputing*, 14, 123-138.

Refenes, A.P., (1995a), ***Data Modelling Considerations***, Refenes, A.P. (Ed.), **Neural Networks in the Capital Markets** (55-67), Sussex, John Wiley &Sons Inc..

Refenes, A.P., (1995b), ***Testing Strategies and Metrics***, Refenes, A.P. (Ed.), **Neural Networks in the Capital Markets** (67-77), Sussex, John Wiley & Sons, Inc..

Refenes, A.P.N., Burgess and N, Bentz, Y., (1997), "Neural Networks in Financial Engineering: A Study in Methodology", *IEEE Transactions on Neural Networks*, 8(6), 1222-1267.

Rodriguez, F.F., Martel, C. and Rivero, S.S., (2000), "On the Profitability of Technical Trading Rules Based on Artificial Neural Networks: Evidence from Madrid Stock Market", *Economic Letters*, 69, 89-94.

Rodriguez, J.V., Torra, S. and Felix, J.A., (2005), "STAR and ANN Models: Forecasting Performance on Spanish Ibex-35 Stock Index", *Journal of Empirical Finance*, 12(3), 490-509.

Roman, J. and Jameel, A., (1996), "Backpropagation and Recurrent Neural Networks in Financial Analysis of Multiple Stock Returns", *Proceedings of the 29th Hawaii International Conference on System Sciences*, 454-460.

Saad, E.W., Prokhorov, D.V. and Wunsch, D.C., (1998), "Comparative Study of Stock Trend Prediction Using Time Delay, Recurrent and Probabilistic Neural Networks", *IEEE Transactions on Neural Networks*, 9(6), 1456-1470.

Safer, A.M., (2001), " Predicting Abnormal Stock Returns with a Nonparametric Nonlinear Method", *IEEE International Joint Conference on Neural Networks (IJCNN'01)*, 15-19 July, 3, 1833-1837.

Safer, A.M., (2003), "A Comparison of two Data Mining Techniques to Predict Abnormal Stock Returns", *Intelligent Data Analysis*, 7(3), 3-13.

Schierholt, K. and Dagli, C.H.,(1996), "Stock Market Prediction Using Different Neural Network Classification Architectures", *Proceedings of IEEE/IAFE Conference on Computational Intelligence for Financial Engineering*, 24-26 March 1996, New York, 72-78.

Shi, S.M., Xu, L.D. and Lui, B., (1999), "Improving the Accuracy of nonlinear Combined Forecasting Using Neural Networks", *Expert Systems with Applications*, 16, 49-54.

Singh, S., (1999), "A Long Memory Pattern Modelling and Recognition System for Financial Time-Series Forecasting", *Pattern Analysis and Applications*, 2, 264-273.

Sitte, R. and Sitte, J., (2000), "Analysis of the Predictive Ability of Tie Delay Neural Networks Applied to S-P500 Time Series", *IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews*, 30(4), 568-572.

Stansell, S.R. and Eakins, S.G., (2003), "Forecasting the Direction of Change In Sector Stock Indexes: An Application of Neural Networks", *Journal of Asset Management*, 5(1), 37-48.

Sun, Y.F., Liang, Y.C., Zhang, W.L., Lee, H.P., Lin, W.Z. and Cao, L.J., (2005), "Optimal Partition Algorithm Of The RBF Neural Network And Its Application To Financial Time Series Forecasting", *Neural Computation & Application*, 14, 36–44.

Tamura, S. and Tateishi, M, (1997), "Capabilities of a Four-Layered Feedforward Neural Network: Four Layers Versus Three", *IEEE Transactions On Neural Networks*, 8(2), 251-255.

Thawornwong, S. and Enke, D., (2003), ***Forecasting Stock Returns with Artificial Neural Networks***, Zhang, P.G. (Ed.), **Neural Networks in Business Forecasting** (47-75), Hershey, Idea Group Inc..

Vellido, A., Lisboa, P.J.G. and Vaughan, J, (1999), "Neural Networks in Business: A Survey of Applications (1992-1998)", *Expert Systems with Applications*, 17, 51-70.

Walczak, S., (2003), ***Forecasting Emerging Market Indexes with Neural Networks***, Zhang, P.G. (Ed.), **Neural Networks in Business Forecasting** (73-101), Hershey, Idea Group Inc..

Walczak, S., (1999), "Gaining Competitive Advantage for Trading in Emerging Capital Markets with Neural Networks", *Journal Of Management Information Systems*, 16(2), 177-192.

Wang, J.H. and Leu, J.Y., (1996), "Stock Market Trend Prediction Using ARIMA-Based Neural Networks" *IEEE International Conference on Neural Networks*, 4, 3-6 June, 2160 – 2165.

White, H., (1988), "Economic Prediction Using Neural Networks: The Case of UBM Daily Stock Returns", *Proceedings of the IEEE International Conference on Neural Networks*, II-451-II-458.

Yao, J., Tan, C.L. and Poh, H.L., (1999), "Neural Networks for Technical Analysis: A Study on KLCI", *International Journal of Theoretical and Applied Finance*, 2(2), 221-241.

Yao, J.T. and Tan, C.L., (2001), "Guidelines for Financial Forecasting with Neural Networks", *Proceedings of International Conference on Neural Information Processing*, Shanghai China, 14-18 November, 1-6.

Yıldız, B. (2001), "Finansal Başarısızlığın Öngörülmesinde Yapay Sinir Ağı Kullanımı ve Halka Açık Şirketlerde Ampirik Bir Uygulama", *İMKB Dergisi*, 17, 51-67.

Yümlü, S., Gürgen S.F. and Okay, N., (2003), "Financial Time Series Prediction Using Mixture Of Experts, *Lecture Notes in Computer Science*, 2869, 553-560.

Yümlü, S., Gürgen S.F. and Okay, N., (2004), "Turkish Stock Market Analysis Using Mixture of Experts", *Engineering in Intelligent Systems (EIS 2004)*, Conference Proceedings.

Yümlü, S., Gürgen S.F. and Okay, N., (2005), "A Comparison Of Global, Recurrent and Smoothed-Piecewise Neural Models for Istanbul Stock Exchange (ISE) Prediction", *Pattern Recognition Letters*, 26(13), 2093-2103

Zhang, G. P. (2001), "An Investigation of Neural Networks for Linear Time-Series Forecasting", *Computers & Operational Research*, 28, 1183-1202.

Zhang, G., Patuwo, B. E. and Hu, M.Y., (1998), "Forecasting with Artificial Neural Networks: The State of the Art", *International Journal of Forecasting*, 14, 35-62.

Zhang, G.P., and Qi, M. (2005), "Neural Network Forecasting for Seasonal and Trend Time Series", *European Journal of Operational Research*,160, 501-514.