UNIVERSITY OF NAIROBI

SCHOOL OF COMPUTING AND INFORMATICS

A Neural Network Model for Predicting Stock Market Prices at the Nairobi Securities Exchange

by

Wanjawa, Barack Wamkaya
P56/73404/2012

A research project submitted in partial fulfillment for the Requirements for the award of Degree of
Masters of Science in Information Systems at the
School of Computing and Informatics
of the University of Nairobi

May 2014

DECLARATION

This research project is my original work and has not been presented for a degree in any other university.

Signed: ___________________________________ Date: _________________________

Wanjawa, Barack Wamkaya
P56/73404/2012

APPROVAL

This research project has been submitted for examination with my approval as university supervisor.

Signed: ___________________________________ Date: _________________________

Lawrence Muchemi
Lecturer
School of Computing and Informatics
University of Nairobi

DEDICATION

This research project is dedicated to the three of the four candidates: Jacqueline, Amos and Stephanie.

ACKNOWLEDGEMENT

I acknowledge the dedication and guidance of my Supervisor, Lawrence Muchemi, who steered me on the right path throughout the project. His valuable insights are greatly appreciated. I am also grateful for the guidance and useful comments from the members of the panel – Dr. Opiyo, Dr. Kahonge and Christine Ronge. I also appreciate the contributions by the other Faculty members at School of Computing and Informatics (SCI) and especially recognize the efforts by Christopher Moturi, the Deputy Director, who ensured that we were enlightened on the requirements of the project. The seminars that he organized also helped a great deal.

I am also grateful for the many interactions and useful discussions that I had with fellow graduate students of SCI. I am appreciative of the support by students who were with me on the same panel for being open in discussions. I cannot forget to appreciate the useful insights by James Indimuli, one of the graduate students at SCI. Finally, I would like to thank my employer, ILRI, for having been accommodative of my absence to attend academic engagements and also allowing me to use their computing facilities.

I finally thank my family for having endured my prolonged working hours both at home and away. This endurance is coming to an end.

However, this research was made possible through the contributions of many other individuals all of whom I may not be able to name, though this is inadvertent. Feel acknowledged for your help, inputs and insights.

ABSTRACT

The Nairobi Securities Exchange (NSE) is a major player in the Kenyan financial sector, dealing with both debt and equity. The entry capital in the equity sector is usually low, hence allowing many ordinary Kenyans to invest in this sector. Investors rely on Stockbrokers to undertake stock trade. Most Stockbrokers use technical, fundamental or time series analysis when advising their clients. However, these methods do not usually guarantee good returns, hence the need to provide Stockbrokers with a substitute predictive tool to guide their decisions. Such a tool can be based on an artificial intelligence (AI) model that trains from available stocks data, gains intelligence and then uses the acquired knowledge for predicting future prices.

The research singled out Artificial Neural Network (ANN) as the basis for the model, after considering various algorithms and their suitability for different problem domains. Through experimentation, the project developed an ANN model, based on feedforward multi-layer perceptron (MLP) with error backpropagation. The final model was of configuration 5:21:21:1 i.e. 5 inputs, with two hidden layers each having 21 neurons and 1 output.

To test the model, the research developed a prototype, based on C# programming language and tested it on data of daily trades at the NSE compiled in the five-year period 2008-2012. The holdout method was used for training and testing, with 80% data for training and the balance 20% for testing. The results showed that the model was able to predict the future trends of three chosen stocks correctly, with a Mean Absolute Percentage Error (MAPE) of between 0.77% and 1.91%. The performance of the model was also validated by comparative tests with other open sources tools (Neuroph and Encog) on the same NSE data, where the comparative Root Mean Square Error (RMSE) were 1.83 (Prototype), 1.94 (Neuroph) and 2.85 (Encog) on one of the test stocks. The model was tested for applicability to other markets by using data from the New York Stock Exchange, where it achieved a MAPE of between 0.71% and 2.77% on three selected stocks in the same test period.

The results showed that ANN-based models can be used to develop low RMSE systems, hence can be used in developing stock market prediction software. This project was done as an ICT for development (ICT4D) initiative.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

2D	Two dimensions
AE	Activation Error
AI	Artificial Intelligence
ANN	Artificial Neural Network
ARIMA	Auto-regression Integrated Moving Average
ART	Adaptive Resonance Theory
ATS	Automated Trading System
B	Billion
CBK	Central Bank of Kenya
CDSC	Central Depository and Settlement Corporation
CSV	Comma Separated Values
DAX	Deutscher Aktien IndeX (German Stock Market Index)
ES	Expert System
GA	Genetic Algorithm
GB	Gigabyte
GDP	Gross Domestic Product
GMM	Gaussian Mixture Model
GOK	Government of Kenya
HIS	Hybrid Intelligent System
HMM	Hidden Markov Model
HTML	Hypertext Markup Language
I/O	Input/Output
IO	Input/Output
ICT	Information and Communications Technology
ICT4D	ICT for Development
IT	Information Technology
JRE	Java Runtime Environment
KNN	K-Nearest Neighbour
KShs.	Kenya Shillings
LM	Levenberg-Marquardt
M	Million
MAPE	Mean Absolute Percentage Error

MLP	Multi-layer Perceptron
MSE	Mean Square Error
N/A	Not Applicable
NN	Neural Network
NSE	Nairobi Securities Exchange
NYSE	New York Stock Exchange
OS	Operating System
PAC	Probably approximately correct
PEAS	Performance measure, Environment, Actuators and Sensors
PLR	Piecewise Linear Representation
RAM	Random Access Memory
RDANN	Robust Design of Artificial Neural Networks
RMS	Root Mean Square
RMSE	Root Mean Square Error
SARSA	State-Action-Reward-State-Action
SOM	Self-Organizing Map
SP500	Standard & Poor’s 500 index i.e. United States Stock Market Index
SQL	Structured Query Language
SVM	Support Vector Machine
US$	United States Dollars

CHAPTER 1
INTRODUCTION

1.1 Background

The Nairobi Securities Exchange (NSE) is a trading institution where shareholders buy and sell stocks in various listed companies. The market generally operates a ‘willing-buyer, willing-seller’ trading, where shares are offered for sale at an indicated price, and willing buyers allowed to acquire them at these prices. If there is no match, then no trade takes place. Such requests that cannot be matched have to await a match during the next trading session or expire after some predetermined time duration. There is also the possibility of trading at ‘market price’, being the acceptance of whatever price the market offers.

The NSE is listed amongst the institutions considered as part of the Economic sector indicators in Kenya, according to the Central Bank of Kenya Statistical bulletin of June 2012 (Central Bank of Kenya, 2012 June). Economic indicators usually gauge how investors are willing to put their money into the economy. The NSE is therefore an important cornerstone of the economy that controls billions of Kenya shillings. The Central Bank of Kenya Monthly Economic review of December 2012 states that the market capitalization or shareholders wealth at the NSE was KShs.1,272 Billion, with equity turnover (shares trades) in that month being KShs.7.582 Billion. On the other hand, the banking sector balance sheet at the same period stood at KShs.2,046 Billion (Central Bank of Kenya, 2012 December). That means that the NSE capitalization was almost half the total balance sheet of Kenyan banks. These figures indicate that the NSE controls a significant portion on the national capital. Developing this sector of the economy is therefore one of the ways of spurring the growth of the overall national economy. This can be achieved by encouraging more investors in the NSE, so that the capitalization can grow progressively.

Part of the investment in the NSE involves buying and selling of shares (equity sector), through the process of stock trading. Stocks, which are also known as shares, are defined as the unit of ownership in a particular publicly listed company. In the process of stock trading, investors acquire wealth by being owners of companies (shareholders) through such company stocks or by getting liquid money from sale of stocks. The stock market is open for sellers to ask for a price for their stock and for buyers to acquire it at the stated costs. Shareholders however do not directly execute the trade, nor is there any meeting between buyers and sellers for sale negotiations. They trade by giving instructions to their Stockbrokers, who in turn execute orders by automatically matching mutually agreeable prices through the NSE trading software. Stockbrokers are the only authorized agencies that can act on behalf of investors in the stock trade business. Stockbrokers do not just execute client orders, they are also charged with the responsibility of advising clients on NSE trades (Government of Kenya, 2009). In the stock trade business, investors, shareholders, traders and clients usually mean the same thing. These are the entities that invest in the stock market by buying and selling stocks through Stockbrokers.

In their advisory role, some Stockbrokers base their advice on their observation of short term price movements (trend). Other Stockbrokers do basic research into the fundamentals of the various stocks or undertake technical analysis before they advise their clients on investment decisions. However, none of these methods have any assurance of profitability and they usually state a caveat as such. While talking about the stock market, Graham (2003) explains technical approaches as those that generally urge investors to buy stocks because they are appreciating in value and sell when their value declines. He says this is the ‘popular’ method. This approach is based on the immediate or short term focus. He further says that many investors acquire common stocks for the mere ‘excitement and temptation’ of the market. This state of investing for the excitement of the market is described as ‘temperament’. Despite the popularity of technical analysis, he states that this approach is unreliable and is akin to ‘simple tossing of a coin’ (Graham, 2003, p. 10)

Since none of the current methods of investment used by NSE Stockbrokers guarantee trade at a profit, investors can be said to be leaving important investment decisions to mere temperament of Stockbrokers. Despite this lack of investment advice that assures investors of profits, this sector controls about half the wealth compared to bank deposits. It is therefore important to have carefully considered investment advice. The likelihood of financial losses, due to inadequate or incorrect advice, is detrimental to investors, especially the low-income citizens who may be investing their livelihoods in the stock exchange, hoping for some profit. Without assurance of some profit, potential investors have been reluctant to consider the NSE as a serious investment sector. Lack of interest in this sector is therefore likely to stagnate the growth of the NSE and hence fail to help the country achieve the desired economic growth.

Stockbrokers need to be empowered to enable them have some capability to provide the best advice to their clients. Such empowerment should not only improve their reputation amongst clients, but is also for their own benefit, since they are paid a commission of each trade. If Stockbrokers do not have tools that can generate good advice, then they are bound to fail in their duty, leading to apathy from the investors. Their current popular methods of technical analysis (using trends for guessing future price movements) and fundamental analysis (buy-and-hold principle for any company that is considered good) have no pointers on the exact price for a future trade. A system that can guide on the most likely next day price is therefore missing and necessary. Such a predictive tool is therefore desirable.

It is for this reason that there is need to formulate an artificial intelligence (AI) model that can be developed into a tool that can be used by Stockbrokers to advise investors on exactly which stock to invest in, for guaranteed profitability. Such a tool should not only show the trend but also the most probable stock prices for next day trade. Additionally, with such a system having generated a shortlist of good stocks, it is possible to further pick the most profitable of these stocks at any given time. Such a system should therefore be able to predict the next day events. Prediction is the foretelling of a future event or outcome, before it occurs. In contrast, trends (technical analysis) would show many stocks moving in a particular direction, without pinpointing the most profitable amongst them. The AI system can be able to provide a longer term investment plan so that the investor knows which stocks to buy and which ones to sell at any particular time in the future (prediction). Such an AI system can be based on neural networks, which are computer algorithms formulated using specific AI rules to learn from data and then be used for tasks such as prediction. Neural networks (NN) and Artificial Neural Networks (ANN) mean the same thing.

Using technical analysis such as trending, as basis of prediction, is a method that has already been employed in some markets, including the NSE, where Ndiritu (2010) developed a support system for trending future prices. Research has also been done on other markets, where stock market price prediction has been attempted. Deng et al. (2011) applied technical analysis for prediction of select stocks at the New York Stock Exchange. Aghababaeyan et al (2011) undertook study on stock market prediction for the Tehran Stock Exchange, where they developed a tool that was used for prediction in that market, achieving an accuracy of 97%. This was a research study tool that was not commercialized. Others such as Khan et al. (2011) conducted a study on the Bangladesh Stock market where they developed a tool with average error of 3.7% & 1.5% on two simulations, while Pan et al. (2005) developed a tool for the Australian Stock market with 80% accuracy in predicting the price direction. None of their tools were developed commercially or targeted for the respective stockbrokers.

1.2 Problem Statement

Trading in shares is big business in many economies including Kenya. The volume of trade is in the range of KShs.500M to KShs.1B daily. This is a large capital in the country's economy. Currently, Stockbrokers who execute trades and advise clients, rely on their experience, technical analysis (price trends) or fundamental analysis (buy and hold) in picking their stocks. These current methods are subjective and are usually short sighted due to their limited capacity to crunch raw numbers. With the value of trade money involved, improper investment could easily mean great losses to investors, especially if they keep making wrong decisions. Lack of guaranteed returns has also led to the reluctance by potential investors to participate in the market. It is therefore desirable to have a tool that can guide on the most likely next day prices (prediction) as a basis of making any investment decision.

Based on the information on their websites, the Kenyan Stockbrokers do not seem to have any artificial intelligence tool that can help them advise clients on which stocks are suitable for any buy or sale trade. The websites provide information that points to use of fundamental and technical analysis methods as being their basis of the predictions of future stock price movement. These tools show a trend on future movement and not the figure of the most likely trade price for any stock in future. It is therefore desirable to have a tool that does not just point at a direction of price movement, but also provides the most likely price value of the stock itself. The Stockbroker is the appropriate target for such a tool, since they execute trades and also advice clients on where to invest. AI methods that can actually analyze the stock prices over time and gain intelligence, then use this intelligence in prediction, can be used to model such a tool. Such a system shall provide guidance on exact price movement, hence allow investors to pick the most profitable investments. This is a better method of deciding on the stocks to trade in than other current non-AI methods, which generally show a trend.

The predictive model shall advise on the exact timing for any sale or buy decision, based on intelligence. This shall enable Stockbrokers to advise their clients to make more strategic decisions. This shall then enable clients to invest their funds in stocks that are bound to appreciate in value. The same tool shall enable investors maximize returns by selling their stocks at the right time. Assured returns shall eliminate uncertainty out of the billion shilling investment industry and hence encourage more people to invest in the NSE.

1.3 Significance of Study

Currently, Kenyan Stockbrokers use other non-AI tools which may not be effective. Additionally, these current methods do not have predictive abilities. These current methods mainly provide trends, but not the most likely next-day stock prices. Despite this lack of appropriate tools, it is still the Stockbrokers who are the interface between clients and the trading platform, hence are in a position of both influential advice to clients and actual execution of trades on the trading platform. By using AI models to develop tools that can help Stockbrokers advise their clients on where to invest (predictive), the investors are likely to make right and sound judgment in their trading. Investors assured of return on investment are therefore more likely to invest in the NSE. This in turn shall enable the NSE to grow and make its expected contribution to national economic growth.

The NSE plays a major role in the economy growth of Kenya. Investment in the stock exchange enables the movement of funds in the various sectors of the economy, since the listed firms represent the full spectrum of the nation’s economy. The funds help in the development of these firms, hence the national economy. Apart from developing these firms, individual citizens benefit by investment through savings, which improves the savings culture. A savings culture enables people to be self-reliant and to ensure their future wellbeing. Stocks also provide the citizens with an avenue to make a profit hence improve their wellbeing, improve their purchasing power, and further enhance their participation in economic activities. It is an investment vehicle like any other.

The project also provides a link between Information and Communications Technology (ICT) systems from theoretical considerations and its implementation on a practical industry problem. The academic community shall be encouraged to relate their theory to practice and hence solve societal problems in our country Kenya – directly applying ICT for development (ICT4D).

1.4 Research Objectives

The research project had the following four specific objectives:

1. Investigate Artificial Intelligence models that are capable of financial market prediction as a basis of designing a model for the Kenyan market
2. Design an AI-based model for use by Kenyan Stockbrokers in predicting prices of NSE stocks
3. Develop a prototype based on the identified model using a suitable programming environment
4. Test and evaluate the performance of the prototype in predicting the share prices of selected NSE stocks, based on NSE data compiled in the five year period, 2008 to 2012

These objectives were achieved through study, model design, data compilation and prototype development, testing and evaluation.

1.5 Assumptions

The research made some assumptions as the basis of study. The first assumption was that trading at the stock market was purely based on supply and demand, with willing buyers and sellers. The second assumption was that there were no external factors (political, economic or social) locally and globally that had a direct effect on stock trading. Thirdly, there was no manipulation of any form, by the Stockbrokers or other players that could affect the stock price movement. The final assumption was that there was no relationship between share price and traded volume. Neither price has an effect on volume and vice versa.

1.6 Limitations

The NSE currently had sixty listed companies whose stock performance could be studied in this research. However, to enable data consistency that facilitated the project, the study was limited to some selected stocks. These stocks were those that were part of the market indicator share price index, to ensure that they had a stake in determining market worth.

Secondly, the stock had to have traded on the market continuously in the five-year period of the study, to ensure data consistency by avoiding gaps in the data stream. Finally, the stock should not have been subjected to any share splits or issuance of bonus shares during the period. This eliminated the usually erratic stock price movement that occurred during such events.

1.7 Structure of the Project

The first chapter introduced the topic of the Kenyan stock market, the trading method, the strategic position of the Stockbroker and the current methods used by Stockbrokers to advise clients on stocks to invest in. The chapter also gave the rationale for the use of artificial intelligence models to build tools that Stockbrokers can use as a basis of advising their clients on investment.

The second chapter provides a detailed review of what is happening in the sectors of stock trade and computer intelligence with a view of linking these two fields. It also provides insights into research done in stock market prediction, the gaps identified and how predictive systems can be constructed. The third chapter provides details on the methodology that was used to design the model, develop the prototype and evaluate the efficacy of the prototype through experimentation. Modalities of data collection and formatting are also presented. The fourth chapter provides the results and then reviews, interprets and analyzes these results. The accuracy of the proposed system in predicting the stock prices is determined and discussed in this chapter. The final chapter summarizes the research findings and suggests areas of further study.

CHAPTER 2
LITERATURE REVIEW

2.1 Overview

To understand how Artificial Intelligence (AI) can be applied in the stock trade business, there was need to study the various players in the field and their contribution to the daily trade. The Capital Markets manages the long term investment sector in Kenya. This is where listed investments (securities), security management, the stock exchange, traders and the trading platforms fall. It was also necessary to review the current tools that Stockbrokers use in advising their clients on the stocks to buy or sell at any given time. Additionally, in order to develop intelligent agents that can be used by Stockbrokers, it was desirable to have insights into the broad field of AI, which attempts to make computers behave like humans by emulating various cognitive skills. From the vast AI field, there was need to study which AI technologies were appropriate to manage the stock trade business and how applications developed by these technologies have performed in other domains. Finally, there was need to identify appropriate AI tools that were suitable for use by Stockbrokers to manage the stock trade in Kenya. This was to facilitate development of a similar tool for the local market.

2.2 Stock Trading Business in Kenya

The Kenyan stock market has different players who trade in various instruments. The instruments are the various securities traded in the market, while the players are the Capital Markets, Central Depository and Settlement Corporation (CDSC), the Nairobi Securities Exchange (NSE) and the traders (Stockbrokers).The first player is the Capital Markets, which deals in mobilizing savings for longer term investments in productive enterprises. The Capital Markets Act defines the constituents of the capital market as ‘long term financial instruments’ that can be traded on a securities exchange or between parties (Government of Kenya, 2009). These investments are generally equity and debt. The Act further defines the structure of the market and modalities of licensing the various players. In its last annual report, titled the ‘Kenya’s financial sector stability report 2011’, the Central Bank of Kenya (2011) lists the various capital market players who were licensed as at December 2011. These are shown on Table 2.1 below.

Table 2.1: *List of players in the Capital Market in Kenya as at December 2011*

Player/Licensee	**No. as at Dec 2011**
Securities Exchange	1
Central Depositories	1
Investment Banks	11
Inv. Banks with Extended Licenses	1
Stockbrokers	10
Investment advisers	18
Fund Managers	19
Collective Investment Schemes	16
Auth. Depositories/Custodians	14
Credit Rating Agencies	1
Venture Capital Companies	1
Dealer	1
Total	**94**

The report further states that the capital markets are expected to be key in financing the Kenya's Vision 2030 projects. The GOK Vision 2030 states that the capital market is expected to be a contributor of upto 50% of the GDP. It is expected to contribute significantly to large capital projects envisaged in the Vision 2030. The plan also aims at increasing the number of listed companies at the NSE to enable more Kenyans build wealth in this sector (Government of Kenya, 2007). The Financial sector stability report by CBK (2011), states that the equity turnover in 2011 was KShs.81.2B while the bond turnover was KShs.445.46B in the same period. The equity sector therefore controlled 15% of the NSE trades. Based on the Vision 2030 projections, the equity part of the Capital markets shall be contributing upto 7.5% of the GDP. This is quite substantial, making the NSE equity (stocks) sector significant and important going forward. Trading in such securities is done at the NSE.

A security is something that has financial value. It is registered so that one can refer to its existence. A security at the NSE is represented by a share or a bond. Shares can also be referred to as equity, while bonds are also called debt instruments. Debt Securities are pre-agreed amounts payable in advance usually for a fixed term. They are payable when a particular waiting period has elapsed. Examples of debt securities are bonds and commercial papers. The NSE trades in bonds, mainly issued by the Government. On the other hand, Equity Securities are company shares, which is a unit of ownership in that particular company. Shareholders can sell their stake in the company by selling off their stocks to other parties (capital gain) or can benefit from interest (dividends) distributed to shareholders. Finally, a derivative is an investment in a future expectation, usually based on some asset.

This type of investment is a derivation of some other instrument, hence investing now for future gain.

The CBK weekly statistical report for week ending March 28, 2013 provides major economic highlights of the country. On the capital markets, CBK states that the market data is measured by two sectors – equity (stocks) and bonds (debt). The equity turnover was KShs.2,449.4M at close of that period, with shareholder wealth of KShs.1,590.62B. On the other hand, the bond market had a turnover of KShs.4,797.75M (Central Bank of Kenya, 2013, March 28). At the close of that week, the ratio of equity to bond turnover was about 1:2. The importance of the equity component at the NSE in particular, and the Kenyan economy in general, is therefore evident. In an earlier period, the 2011 Financial sector stability report by CBK (2011) states that the total equity turnover in 2011 was KShs.81.2B compared to a bond turnover of KShs.445.46B, a ratio of 1:5. However, the debt (bond) market had fewer players transacting big valued deals.

A second player in the Kenyan stock market business is the Central Depository and Settlement Corporation (CDSC), which was created by the Capital Markets Authority to setup a system of handling and settling securities that are listed at the NSE. CDSC started its operations in 2004 (Central Depository and Settlement Corporation Limited (CDSC), 2013). The established system enables investors to register securities in electronic format and update the register whenever trading occurs – sales and purchases. Shareholders therefore need to maintain a register of their NSE portfolios with the CDSC. The two bodies work hand in hand in ensuring that trades (actual sale and purchases at NSE) and the registration of the shareholder stocks (register at CDSC) are synchronized. Before the operationalization of the CDSC, shareholders would get paper certificates for each of their shareholdings. Transfer of shares would involve surrender and update of share certificates to reflect the new ownership.

The third participant in the stock market is the Nairobi Stock Exchange, the predecessor of Nairobi Securities Exchange (NSE). It was formed in 1953, when the London Stock Exchange recognized it as its overseas exchange before Kenya attained independence. The business of stock trading however started in Kenya earlier in the 1920s as informal dealings through gentleman's agreement. At that time, parties would agree to sell stocks and charge commissions. It was then a private club with members from few respectable professions, mainly accountants, auctioneers and lawyers. Thereafter, the NSE admitted stockbrokers from 1954 with minimal private deals. The low level of activity persisted until independence. Even in the period from 1963 to 1988, minimal activities were still recorded at the NSE, as post-independent Kenya was slowly coming to terms with stock trade. Interest in the NSE peaked in 1988 when the Government of Kenya offloaded its 20% stake in the Kenya Commercial Bank ownership by selling these shares to the public through the stock exchange. Thereafter, Kenyans and foreign investors increased their interest in the stock market such that by 1994, the NSE was rated 'the best performing market in the world with a return of 179%' (Nairobi Securities Exchange, 2013c). Other developments saw increase in the number of investors to more than 110,000 shareholders during the Kenya Airways privatization when the Government offloaded its 51% stake. In 2006 the exchange commenced live trading using an Automated Trading System (ATS).

The Nairobi Stock Exchange changed its name to Nairobi Securities Exchange in July 2011. This was in line with its strategic objective of diversifying its range of services to encompass trading in securities i.e. equities, debt, derivatives and associated instruments. Currently, the NSE has a total of nineteen members, who are generally the licensed Stockbrokers and a few other financial institutions (Nairobi Securities Exchange, 2013d). These members can trade at the NSE. The entry level to the bond market is usually high and hence prohibitive to most citizens. It is the corporate sector that usually trade in bonds. The common and easily accessible market is the equity market, where the rules dictate that trades start from a minimum of 100 shares, which on the lowest valued stock, can be as low as total investment of KShs.100. The equity market is therefore more active with many shareholders, hence a segment worthy of further study. It is the common man's market, and the focus of this study. In the equity sector, the NSE has ten different market segments, each with its own listed stocks. **Appendix 1** lists the constituent companies that trade in each of these ten segments. The ten market segments are Agricultural, Commercial and Services, Telecommunication and Technology, Automobiles and accessories, Banking, Insurance, Investment, Manufacturing and allied, Construction and allied and Energy and petroleum segment.

The NSE performance is measured on a daily basis by six key indicators. The first is the '20-share index', which is a measure of the performance of 20 stocks picked from the different sectors of the market. The second is the 'All share index', which measures the performance of all the stocks at the NSE. The third is 'Market capitalization', which provides information on the total value of traded stocks (measured at a particular day of trade), while the forth indicator, 'Share volume' is a measure of the number of shares traded. The final two are 'Equity turnover', the value of stocks trades (usually cumulated over time) and 'Bond turnover' the value of bonds trades (usually cumulated over time).The 20-share index is one of the indices used in measuring the performance of the NSE. This measure is also included in the statistical reports by the CBK. It is therefore an important index. Its component stocks have been carefully chosen to be representative of the various economic sectors, and hence the economy as a whole. Companies whose stocks are part of this index are therefore equally important in any study of the market performance. **Appendix 2** lists the companies that comprise the NSE 20-share index as per press release from the NSE dated 15th October 2012 (Nairobi Securities Exchange, 2013e). An upward movement of any of the indicators points to better gains by investors and generally a good economy. The Central Bank of Kenya tracks these daily movements and includes them in economic indicator reports for Kenya.

Other than the regulator (CMA), the registrar (CDSC) and the trading platform (NSE), there is also a fourth player, called the Stockbroker. The Capital Markets Act, Cap 485A, Laws of Kenya defines a Stockbroker as, "a person who carries on the business of buying or selling of securities as an agent for investors in return for a commission" (Government of Kenya, 2009). A stock purchase transaction at the NSE therefore involves the buyer placing an order with a Stockbroker, who in turn accessed the pool of offers for sale and then buys on behalf of the purchaser. The Stockbroker then updates the CDSC with the number of shares purchased. A stock sale also follows the same procedure, where the seller informs the Stockbroker to sell particular stocks, followed by update of CDSC. A method of providing funds to the Stockbroker and vice versa is usually established in advance e.g. cheque, cash or fund transfer. The Act further states that only authorized securities dealers can access trading and use the securities exchange (NSE) facilities. Stockbrokers are listed amongst the authorized members. Individual or corporate shareholders are therefore not authorized to directly trade in securities in their individual capacities. Table 1 lists the number of authorized Stockbrokers as ten. However, as at June 2013, the Capital Markets Authority (CMA) had licensed eleven stockbrokers to undertake brokerage services in Kenya, including trading at the NSE (Capital Markets Authority, 2013). The list of the current eleven Kenyan Stockbrokers is shown in **Appendix 3**.

The basic principle of trade at the NSE is "willing buyer, willing seller". The sale of shares starts with buyers and sellers giving their offer prices. If a buy and sell offer prices match, then the Stockbroker seals the deal. There is no way of knowing in advance the direction of the price movement for any particular share on a trading day. Its value may go up, remain unchanged or decline. Some level of confidence on market performance is therefore needed to provide advance instructions to Stockbrokers on what to buy or sell. However, the bottom line is usually to try and get a profit for any given transaction. When applying to buy or sell through a Stockbroker, the shareholder gives the instructions on the price they are willing to pay or receive. Alternatively, the shareholder may allow the Stockbroker to trade “at market price”, meaning that the trade is executed at whatever rate dictated by the market, regardless of whether this shall be profitable or not. This latter method is equivalent to disposal, without regard to profitability. Without any inkling on market direction, the trade can be offered at any stated price, though there may be no buyer/seller at such prices. If this happens, then the trade may not be executed for some time or not at all.

For a successful trade, the offer price must be reasonable and attractive, unless there are some underlying speculative reasons for a big variance. According to the Trading Rules of the Nairobi Stock Exchange (2010), there are four types of trades. These various trade methods just match the buyer and seller offers until successful trade takes place or a cancellation after the time lapse. The first type is called ‘Immediate or cancel’, where the offer is valid immediately and is cancelled if there is no match. The second is ‘Good till cancelled’ where the offer is valid for 30-days. The other two are ‘Good till day’ which is valid for 5-days, while ‘Day order’ type is valid for the trading day only.

Instead of “at market price” approach to stock trade i.e. disposal, those who care for a profit usually rely on Stockbroker advice. Stockbrokers in turn mostly rely on intuition, experience or some historical analysis. The historical analysis is usually based on looking at past performance with a view of guessing how the current and future price direction should be (technical). Some stockbrokers just look at the graphical trend of prices and try to extrapolate the curve to decide whether the expected price is up, down or stagnant (time series). A final popular method used by Stockbrokers is fundamental analysis, which urges investors to buy and hold on to good company stocks, regardless of current prices, for as long as possible.

Most Kenyan Stockbrokers claim to employ some market analysis method as a basis of advising their clients. They have a ‘research’ section on their websites that explains their methods. One of the Kenyan stockbrokers, Suntra (2013), indicates that they do research to aid in their investment decisions. They state that they collect data in an ‘organized and systematic’ manner so that they can discern ‘information and trends’ to inform their investment choices. Another stockbroker, Drummond Investment Bank (2013), state that they advise their clients on investment based on research. They aim at gathering market intelligence so that they can understand the market based on internal data (trends and habits of the client) or external data using desk research, to ‘construct a picture of the market’. Kestrel Capital (2013) indicates that they also do research in advising their clients on whether to buy, hold or sell particular stocks. On the sample reports provided, their advice is based on financial analysis of company performance over three year periods. Their data sources are tabulated as information from company’s annual accounts, and the stock price movement over time. An upward trend in share price seems to elicit a ‘buy’ recommendation. Another stockbroker, Genghis Capital (2013), provides various technical analysis reports, as a basis for their buy or sell recommendation, while Apex Africa Capital Ltd (2013) uses technical and time series methods in providing investment advice.

Technical analysis is also favoured as an investment decision method, as advocated by NIC Securities (2013), who claim to use ‘Modern financial valuation modeling techniques’ in their research. Details of how they undertake this evaluation is not provided. Old Mutual Securities (2013) state that they use financial modeling to guide their investment advice, though the reports provided are technical analysis and trends. Kingdom Securities do not have an online presence, but are the agents for Cooperative Bank, who state that Kingdom Securities provide updates and ‘research information’ (Cooperative Bank, 2013). Others on the list of those using technical analysis methods are AIB Capital Ltd (2013) and ABC Capital (2013) who do research with a view of seeing future direction. Other NSE members, such as Africa Alliance (2013) say that their research is meant to devise ideas and strategies to cover African financial markets. Their research is meant to help clients make sound investment decisions and ‘outperform’ the market. On the other hand, Chase Bank (2013) as a financial institution advises that before investing in the market, one needs to ensure adequate research to get information on stocks, movements and trends. They offer these services through their stock broking agency. They however do not elaborate on tools used in the research or trends.

Having reviewed all the eleven licensed Stockbrokers, and some additional members of the NSE, it is evident that none of them use technology, and especially AI, that should deliberately mine useful patterns from market data to give them information, which they can in turn use to advice clients on future investment. They do not seem to have any tool that can specifically advice on the next day prices, in advance of the trade (prediction). This is the gap that this research attempts to address by developing a tool that can be used for prediction instead of their current methods. This tier (Stockbrokers) in the trading chain is also strategic because it is the only entity legally allowed to undertake trading, hence has the ultimate power of final execution of trades at the NSE.

2.3 Stock Market Prediction

Stock market data is probably the most studied time series data (Gavrilov et al., 2000). This is due to its importance in most world economies. This sector therefore remains a major area of research. Predicting the stock market is however not a trivial task. To start with, there is need to model the trend of the stock prices, which is nonlinear, then extract certain features on the input data itself, to make it capable of good use (Zarandi et al., 2012). Others, such as Adebiyi et al. (2012) also agree that stock price prediction is one of the most important topics in finance and business. They state that there are many techniques which are not effective in predicting the market. These include statistical, technical and fundamental analysis. They offer this conclusion because these other techniques cannot provide ‘deeper analysis’ that is needed for the task of prediction.

NeuroAI (2013) states that there are four stock market prediction methods: Technical analysis, Fundamental analysis, Time series analysis and Machine learning. Technical analysis involves the use of technical data such as price, volume and price movement to predict future pricing. The aim of using technical indicators in prediction is to discern trends and patterns, which should then inform a direction of future prices. Huang et al. (2011) states that technical analysis deals with historical price movement to predict a pattern for future investment decisions. The method looks at operations and financial situation of a company in deciding the company’s likely future prospects. Deng et al. (2011) lists some of the popular and useful technical indicators as rate of change (ROC), Moving Average Convergence Divergence (MACD) and bias. ROC shows the difference between the current price and the price some days before, while MACD is a predictor of changes in market tendency, with both a signal and value. Finally, bias checks the difference between stock prices and the moving average. All these indicators must be calculated and their values used for guiding the prediction.

In a previous study, Ndiritu (2010) developed a technical trading support system for the NSE, based on technical analysis. He used technical stock market indicators such as moving averages (simple, exponential), which were stored on a database of trading data. The indicators were then plotted on a graph as a pointer to the market condition. This was meant to guide on investment decisions based on the trend shown by these indicators. The system did not however provide a method of deliberately guiding on which particular stock to invest in or how the trend was expected in the future.

A second stock market prediction method, Fundamental analysis, involves the evaluation of key statistical indicators of the companies themselves, such as balance sheet, profitability, future projects etc. There is a thin line between Technical and Fundamental analysis, since both look at the inherent performance of the company. Nonetheless, Fundamental analysis tend to have an overview of the company from a top level view and considers issues such as political, economic and the business environment of the company. Some of the indicators for these fundamentals may also feature on the list of the Technical considerations. The general requirement of fundamental analysis is to understand the company and decide on its future prospects. The understanding can be based on financial performance such as the balance sheet, profit and loss, strategic plan, competition, return on investment, supply and demand. Chen et al. (2007) states that a primary goal of fundamental analysis is investigating the relationship between financial information and other facts about the company such as inventory or revenue growth. Other measures include price to earning ration, earnings per share, dividend yield, gross sales and return on equity.

While studying the Wall Street, one of the most developed stock markets in the world, Graham (2003) says that it is possible to put energy and study in that market and end up with losses instead of profits, unless the investor thinks critically of the courses of action and measures the chances of success. A short term look at numbers only does not work. He describes any other method that does not follow such sound judgments, including daily investment seen on stock markets as mere ‘speculation’. In another study, Klarman (1991) looked at stock market investment practices and said speculators took most of their time trying to predict or “guess” the stock prices movement, as a basis of their investment decisions. Even use of technical analysis (data of past performance) was still speculation in his view. This is because investment should be based on what the company is all about (fundamentals). We can therefore infer that just looking at numbers alone may not necessarily give meaning or intelligent to what may not be quite apparent.

While analyzing the same US stock market that Graham (2003) had studied, Mizrahi (2008) states that a large percentage of the top 100 money managers holding over US$6.8 trillion in stocks employ flawed stock investment methods that generally have three stages – predicting the future, thinking of how this affects investment, then making a trade – the so called top-down approach. He proposed a bottom-up approach, which looks at company fundamentals (financial reports) as basis of deciding a stock and its price, then paying that price regardless of any other factors (fundamentals). He however cautioned that this could take months or years of patience before getting a reward.

A third prediction method is the Time series method. This uses historical performance to predict on a time series scale. A time series is a sequence of sampled quantities from an observation, out of which discoveries such as periodic distribution can be determined (Zhang et al., 2008). The stock market can be described as a chaotic time series phenomena which can therefore benefit from chaotic time series prediction methods. Other methods in time series prediction are linear regression, auto-regression and Auto-regression Integrated Moving Average (ARIMA). An important characteristic of time series data is the dependence on time, hence a current observation has to depend on a past observation in time (Neto et al., 2009). A typical prediction model in this case requires information external to the particular stock, which can be used to extrapolate the performance of the stock in question. Such information should be having a bearing on the stock of study. They used information such as Sao Paulo stock index, German stock indicator (DAX), United States stock indicator (SP500) and currency exchange rates. They achieved a mean absolute percentage error (MAPE) of 2.08%.

Non AI-based methods of predicting stock prices usually tend to just consider short term figures to view a trend and attempt to predict the future direction, based on this small dataset. However, instead of just relying on the direction pointed out by these figures in a short term, it can be possible to use artificial intelligence (AI) methods to give meaning to the data. This should enable some deliberate analysis of the data to learn from the numbers, gain some intelligence from the data and then use the intelligence to predict how the market direction shall be. Additionally, if we can get a method that can provide short term answers, such as the next day trading prices, then investment and potential gains do not need to wait for a long term. This shall therefore also mitigate the 'long-term' wait period, which is premised by most investment strategies, such as those suggested by Mizrahi (2008).

Based on publicly available information, the current Kenyan Stockbrokers do not claim or reveal use of AI based tools that can help them advise their clients on investment avenues. It is therefore desirable to develop such AI tools for them. This tool shall be based on machine learning. Machine learning prediction methods use AI to examine existing data so as to find out an underlying function that can be used for future predictions. To develop this tool, there is need to understand what AI is all about and how it can be used in such machine learning applications.

2.4 Artificial Intelligence

Tanimoto (1987) states that AI is a field of study that has computational techniques, which are used in the performance of tasks that would ordinarily need human intelligence. AI deals with processing information that includes some aspect of reasoning, learning and perception. Behind the scenes, much of AI concerns design and understanding of knowledge representation. In his definition of AI, Patridge (1998) is of the view that AI definition should deal with behavior. The concern should therefore be about 'what can be done' rather than 'how it does it'. AI problems are therefore ill-defined, being defined by aspects of people's behavior. It is the problem-solving behavior of humans that can directly be seen. The method of solution is not observable. AI systems have a way of matching input and outputs using some general rules that work. Russell et al. (2009) states that the AI field benefitted a great deal from studies and concepts from other areas such as philosophy, mathematics, economics, neuroscience, psychology, computer engineering, control theory, cybernetics and linguistics.

Tanimoto (1987) further observes that in AI, there are three concepts that should be dealt with – data, information and knowledge. Knowledge is 'refined information' or 'information in context'. Knowledge is useful in solving problems and even learning. Knowledge can be represented in methods such as production rules, inclusion hierarchies, mathematical logic, frames, scripts, semantic networks, constraints and relational databases. Other scholars propose two broad ways of defining AI – thought process (thinking) and behavior (acting). AI is therefore concerned with both mimicking of human performance and achieving the ideal called rationality i.e. when the right thing can be done based on a situation (Russell et al., 2009)

It is possible to use an AI agent to solve any given problem. One method is to set a goal for the agent, and let it work towards achieving the goal. The goal is set based on the current situation and the performance measure. Goal formulation is therefore considered as the first step in solving AI problems. The next step is to decide the action and states to consider for the given goal. This is called problem formulation. The problem has five states, being initial state, actions, transition model, goal test and path cost. These guide the agent to look for a sequence of actions to reach its goal (search), using the problem to get a solution. If a solution is possible, the action leading to the solution is taken. Thereafter, a new goal is formulated to repeat the process.

An agent strives to be rational so that it can maximize its performance. It therefore chooses a series of actions that are useful in attaining the goal. This means that the agent needs to learn from what it perceives, since the environment may change over time. However, if the environment is known at all times, then no perception or learning is necessary (lack of autonomy). Practical world problems are based on changing circumstances that call for learning (autonomous agents). Russell et al. (2009) asserts that incorporation of learning can facilitate the design of rational agents that would succeed in a wide range of environments. The task environment is generally the problem. The problem domain contains the Performance measure, Environment itself, Actuators and Sensors (PEAS). The rational agent itself is the solution to this problem.

Most AI systems are built based on one of the following four agent models – simple reflex agent, model-based reflex agent, goal-based agent, utility-based agent. A simple reflex agent performs its action based on rules. The second type of agent, the model-based reflex agent, keeps track of the current state of the world, then acts according to predefined rules. A goal-based agent on the other hand keeps track of the current state of the world and also tracks the goals it should achieve, and then acts according to rules that make it achieve the goal. The final type, called utility-based agent, keeps track of the current state of the world and also measures its preferences of the states using a utility function. It then acts to achieve the best expected utility.

To create state of the art systems, Russell et al. (2009) says that learning agents is the way to go. It allows an agent to become more competent than its initial knowledge could otherwise allow. Learning modifies each component of the agent to bring them to agreement with feedback received. This should improve performance. Machine learning is about techniques that enable systems gather experience from data sets and use the knowledge as a basis of better performance in future applications for similar problems (Oboko, 2012). The components of a learning agent, as depicted in Figure 2.1 below, are: a Learning element for making improvements by adjusting actions of the performance element, a Performance element for selecting external actions (generally the non-autonomous agent, taking percepts and acting) and a Critic to provide feedback about the performance of the agent to the learning element, while using some performance standard e.g. reward or penalty. The final element is the Problem generator, which suggests actions that should lead to new experiences.

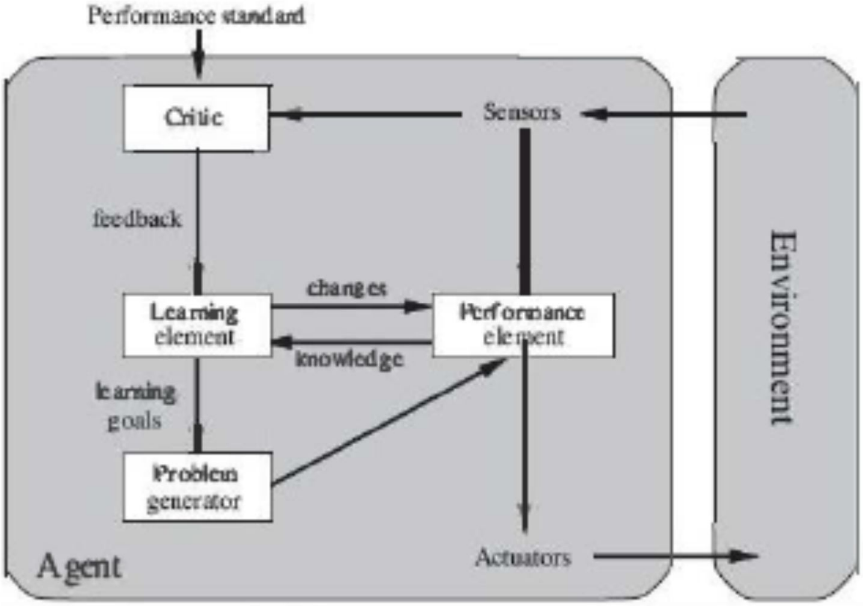


Figure 2.1 – Learning agent (Source Russell et al. (2009))

There are three types of learning, depending on the feedback the agent receives. Unsupervised learning where the agent learns of patterns even without explicit feedback. It is provided with data and it has to formulate the appropriate labels. An application area for such agents is in clustering, which is a popular data mining technique. Algorithms in this domain include Self Organizing Map (SOM) and Adaptive Resonance Theory (ART). The second type of learning is Reinforcement learning, where the agent's knowledge is reinforced by rewards and punishment. They usually rely on some policy e.g. Markov Decision Process. Algorithms that can be employed include Q-learning (for off-policy method) and State-Action-Reward-State-Action (SARSA) for on-policy method. The final learning method is Supervised learning, where the agent is provided with data so that it can observe input and output pairs so as to formulate a function to map such pairs. Popular algorithms include Genetic Algorithm (GA) and Artificial Neural Network (ANN).

In modern AI practice, the distinctions between the learning methods are not all encompassing. For example, it is possible to try and bridge the divide between unsupervised and supervised learning. This new class is called semi-supervised learning, where an agent is given partial examples to help it determine the output from a new set of values not yet seen. Algorithms that employ semi-supervised learning are – Gaussian Mixture Model (GMM) as applied in image classification, Naïve Bayes as applied in text categorization and Hidden Markov Models (HMM) as applied in speech recognition (Zhu, 2013). Oboko (2012) states that learning from an available dataset is called inductive or empirical learning. This means that the supervised learning employs inductive learning to learn from labeled data, while unsupervised learning means inductive learning that gets concepts from unlabeled data. Supervised induction learning can be classified into three, namely, symbolic learning, neural learning and genetic/evolutionary learning. The neural learning class uses ANN as a technique. Algorithms used in ANN include: Adaptive Resonance Theory (ART), Kohonen networks and Feed forward with backward propagation.

The difficulty with learning is to be sure that the agent has enough knowledge to predict correctly. Russell et al. (2009) quotes computational learning theory as providing the answer as, "any hypothesis that is seriously wrong will almost certainly be "found out" with high probability after a small number of examples, because it will make an incorrect prediction. Thus, any hypothesis that is consistent with a sufficiently large set of training examples is unlikely to be seriously wrong: that is, it must be probably approximately correct.". Probably approximately correct (PAC) assumes that future examples shall be drawn from same fixed distribution as the past examples.

Assume random variable ***E*** is a data point before being seen. The probability of generating the same variable can be based on current input/output pairs as defined by Equation 2.1 below. Training sets that enables the agent to be faithful to this equation are adequate enough to enable an accurate prediction.

P(E) = P(X,Y)……………………………………………………….………..2.1

Where:
X is an input value
Y is the corresponding output value in the pair

There are various learning methods that have been developed in AI. The classical methods are supervised, reinforcement and unsupervised. This is based on the data the algorithm shall be exposed to or the type of solution expected of it. Others classify learning algorithms by the structure of the algorithm itself. This results into the following three variant of the classification (Bethard, 2008). The first is Naive Bayes e.g. Bayesian Network and Naive Bayes Network. Bayesian network presents all variable dependence relations, while Naive Bayes network assumes features are conditionally independent for the class variable. Both are trained by finding probability of each class and each feature, given the class. They are useful in classification. Their disadvantage is poor estimation of probability of the network though they have good classification decisions. Wu et al. (2008) lists other applications of Naïve Bayes as text classification and spam filtering. An attempt to modify it for other applications such as data mining, machine learning and pattern recognition has not been as effective as its traditional application in classification.

The second class using this classification method is the K-Nearest neighbor (KNN), where examples are considered as points on an x-y plane, with each plane being a feature of the network. It is used for classifying a new point based on existing points, by calculating how close the new point is to the neighbours using Euclidean distance or Cosine distance. This algorithm basically finds a group of ***k*** objects in the training set which are closest to the test object. It then labels the object according to prevalence of the class in that neighbourhood. This algorithm needs three elements – labeled objects, a distance and the number of nearest neighbours (***k***). One issue that affects performance of KNN is the choice of ***k***. A small ***k*** value makes the results sensitive to noise, while a bigger ***k*** value could result into a neighbourhood with interference from other classes. Combining class labels e.g. majority vote and choice of distance measure (cosine or Euclidean) are also challenges in this method. Apart from classification, KNN can be used for data mining. Wu et al. (2008) lists KNN among the top 10 algorithms in data mining, though classification remains the dominant application. Data mining is however the broad field that incorporates various techniques such as machine learning, pattern recognition, AI, database systems and statistics. KNN can also use unsupervised learning to undertake clustering, which is in the data mining domain.

Finally, advanced algorithms include Support Vector Machines (SVM), which uses support vectors (quadratic programming & heuristics) for classification by maximizing margin of the separating hyperplanes. It uses Kernel methods to classify both linear and non-linear functions in higher dimensional spaces. It requires few training samples and is insensitive to the number of dimensions in the application. Wu et al. (2008) states that SVM aim at finding the most appropriate classification function to distinguish between members of the classes based on training data. The classification should be some geometric function, which then generalizes any new label. Another application, apart from classification is in ranking e.g. ranking documents for information retrieval. One drawback with SVM method is the computational inefficiency (slow training) despite fast classifications. A second advanced algorithm that can be identified is the artificial neural network (ANN), which is a network of computational units (neurons) that employs parallel distributed computing to solve problems. It can be trained using data to be applicable in fields such as classification and regression.

Analysis alone cannot determine if a machine learning algorithm shall work well. Experimental studies are required and should be considered necessary in determining a suitable machine learning algorithm. In comparative analysis, if one algorithm outperforms another, then it is just a matter of how fit it is for the particular application. This therefore cannot be a measure of superiority of the algorithm, but rather its fitness for the problem (Wagacha, 2003). Lim et al. (2000) who conducted a study on the comparison of thirty-three AI algorithms, tend to suggest that there is no significant difference in mean error rates for predication and classification among the major algorithms such as Trees and Rules (Quest, Bayes), Statistical (Discriminants, Logistic regression) and Neural Networks. The data set was however from varied sources and not financial in particular. The study points out the need to know the particular application area in order to single out the algorithms that suits the situation.

A comparison of performance of various machine learning algorithms was also done by Too (2012), who used an open source tool called DTREG to evaluate performance of four algorithms – MLP, SVM, K-Means and Decision Trees. Based on her dataset and application i.e. forecasting weather-based pests, it was K-Means clustering that performed best. However, different algorithms should perform differently on the same task. Suitability for a given task or domain can only be achieved by experimentation (Wagacha, 2003). In another study, Bahrammirzaee (2010) compared ANN, expert systems (ES) and hybrid intelligent systems (HIS), specifically in financial applications. The conclusion was that ANN is better suited for prediction compared to the other methods, though the actual constituents of the HIS could outperform ANN. However, it was difficult to determine the right combination in a HIS, hence just easier to apply the ANN as a method to such problems. ANN is an information processing system that learn and then generalizes, leading to their adaptability in solving various problems including the subjective ones, such as decision making and forecasting.

2.5 Artificial Neural Network

After some lull, the AI field of neural networks, originating from neuroscience, regained prominence from mid-1980s. Russell et al. (2009) says that modern neural network research has laid focus in two areas i.e. making effective network architectures, algorithms and understanding of their mathematical properties and secondly, modeling empirical properties of actual neurons and collections. With time, developments in the AI field of neural networks have enabled these systems to be comparable with corresponding techniques from statistics, pattern recognition and machine learning. AI is now able to solve the problem of expressing knowledge needed by a system by applying learning methods instead of hard coded knowledge.

One reason why ANN is popular is due to their robustness, fault tolerance, ability to learn and generalize, adaptability, universal function approximation and parallel data processing. This enables them to solve complex non-linear and multi input-output (IO) relationship problems (Ortiz-Rodriguez et al., 2013). Gomes et al. (2011) also states that ANNs are useful in practical applications due to their ability to do non-linear mapping, parallel processing methodology, ability to learn from the environment and their subsequent adaptability to the environment. Ghaffari et al. (2006) state that when compared to other methods, ANN have been shown to be superior as modeling technique for data sets with non-linear relationships. This enables them to be applied for data fitting and prediction. In another study, Cerna et al. (2005) stated that the most popular types of ANN were Kohonen network (self-organizing map) and multi-layer perceptron (MLP). The former employs unsupervised learning to cluster data while MLP uses supervised learning to create a classifier. The classifier is then able to match outputs (responses) to inputs (predictors) in the training data. The trained classifier can then predict outputs for new cases. These studies above suggest that ANNs are useful in two AI fields – classification and regression. In classification, the interest is in placing members into distinct groups, while regression involves formulating a real-value target function for any particular input function. The final design of the ANN is generally guided by the application field.

Having established that ANNs are appropriate and applicable to non-linear problems, such as the stock market prediction, it is important to delve into the details of how the ANN works. An ANN is modeled as a representation of the humans own biological neuron and how the neural network functions in a human being. The basic unit is therefore a neuron with the final system being a system of interconnected neurons. Figure 2.2 below depicts the biological neuron.

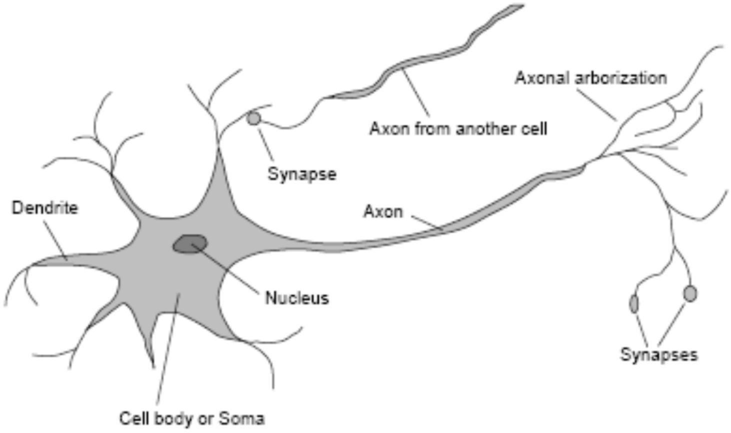


Figure 2.2 – Representation of human neuron (Source: Bethard (2008))

ANN is an AI computation method that is also called connectionism, parallel distributed processing or neural computation. This agent causes an output function when a linear combination of its inputs exceeds a predefined threshold. The Encyclopedia of Mathematics

(2013) states that Neural Networks (NNs) are continuous non-linear functions, which are just represented by a network of computational elements. This type of computational algorithm can be applicable in linear classification, where an on or off (yes or no) is the ultimate result, depending on evaluation of the inputs. ANNs can also be used in regression problems where there is need to obtain a real-valued target function e.g. in a financial system such as the stock market prices. The neural network has different processing units (neurons) that are connected together as a network. These networks have the ability to tolerate noisy inputs, perform distributed computation and can learn to be autonomous. According to Russell et al. (2009), neural networks remain the most popular and effective learning systems in the field of AI.

Figure 2.3 below shows a representation of a seven-input neural network, which processes data to provide a single output, while Figure 2.4 shows a generic neural network with input, processing and output units.

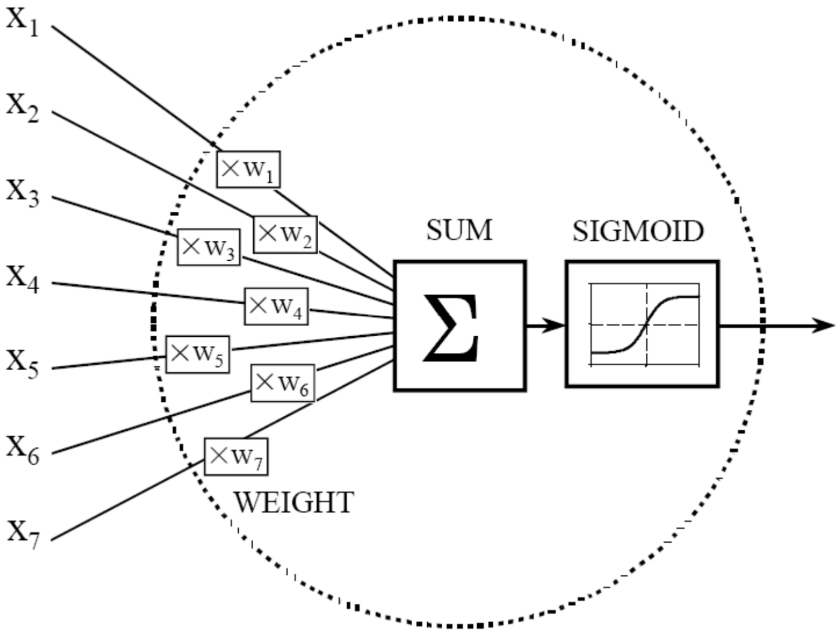


Figure 2.3– Seven-input one-output ANN representation (Source: Smith(1997))

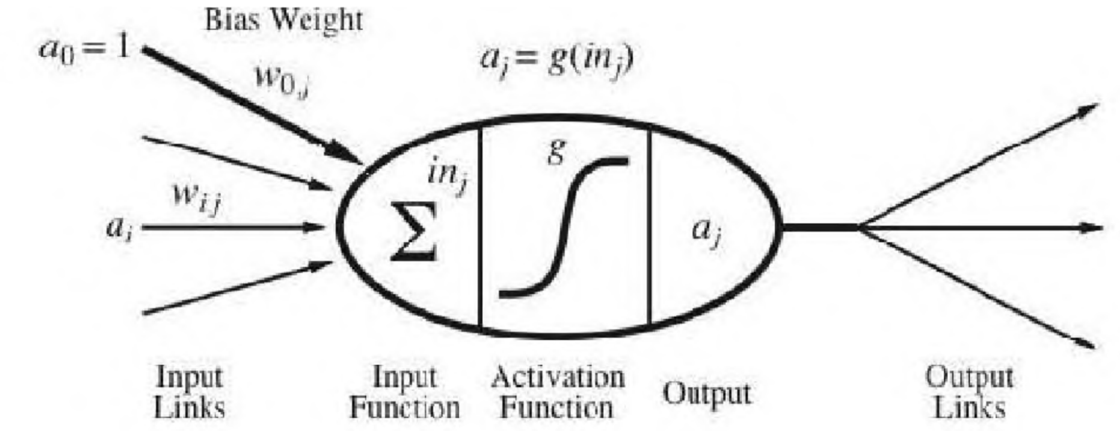


Figure 2.4 – ANN representation (Source: Russell et al. (2009))

From Figure 2.4 above, the descriptions of the various parts of the network are: a basic functional unit called the neuron or node with inputs and outputs. Inputs are processed to obtain an output, which in turn form the input to other neurons. Each link from one neuron to another, ***i*** to ***j***. has its own weight $\boldsymbol{w_{i,j}}$. Additionally, a bias weight (dummy input), $\boldsymbol{a_0}$= 1, with its own weight $\boldsymbol{w_{o,j}}$, also contributes to the summation of neurons output. The final summation is subjected to an activation $\boldsymbol{a_i}$ function to get a final network output. The operational procedure of a typical ANN starts by the first stage of summing up the inputs, where each neuron ***j*** computes a weighted sum of inputs, as shown in Equation 2.2. This involves summation of the inputs and their weights, and also the bias and its weight.

$$in_j = \sum\nolimits_{i=0}^{n} w_{i,j} a_i \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \; 2.2$$

Where:
$\boldsymbol{w_{i,j}}$ is the weight of a link from ***i*** to ***j***
$\boldsymbol{a_i}$ is the bias weight of the node

Thereafter, the output is subjected to an activation function ***g***. Since the NN will usually pass its output to the next layer as input, there is usually need to normalize the outputs to a uniform range of values. This is achieved by applying an activation function to the output. The expected outputs should guide the choice of activation function. Activation function ***g*** can take many forms. Some of the popular activation functions are threshold, logistic function (unipolar sigmoid function), bipolar sigmoid function, hyperbolic tangent function, piecewise linear function and linear function. These functions are described in **Appendix 4**. Choice of activation function is application dependent. Each is suitable for particular problem domains e.g. backpropagation usually need an activation function that has a derivative such as the sigmoid or hyperbolic tangent function. The next stage in the ANN processing involves obtaining a final output as described by Equation 2.3 below:

$$a_j = g(in_j) = g\left(\sum_{i=0}^{n} w_{i,j} a_i\right) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad 2.3$$

Where:
a is the final output
g is the activation function
in is the input
$\boldsymbol{w_{i,j}}$ is the weight of a link from ***i*** to ***j***

After the processing of signal through a neuron, there is usually need to connect to other neurons. From the first neuron, a connection is made to other such similar neurons to form a network. There are two methods of connection - Feedforward network, where the connection is in one direction only, from upstream to downstream without loops or a Recurrent network, where the basic feedforward network is established, then the output of the neurons are also connected back to the input of the neurons.

Though the choice of activation function is application dependent, the logistic regression (sigmoid) is observed to be effective in training neural networks for regression and classification, hence widely used. This is after the realization that simple threshold units do not solve a lot of problems. Feedforward networks, having feedback of the error (backpropagation) are also noted to be useful in real world problems. Recurrent networks tend to be complex to construct and implement. Adebiyi et al. (2012) confirms that the sigmoid function is the most widely used. They say it performs better than others (unit step, piecewise liner, binary transfer and Gaussian functions). Sigmoid functions accept a wide range of input signals (-∞ and +∞) and gives an output between 0 and 1. The output can be further processed and interpreted in relation to desired output. The other feature of ANN in relation to the sigmoid function is the ability to introduce an additional bias to the ANN, with a weight of 1, which is guaranteed to move the function left or right of the central value x=0. The shape of the sigmoid means that most change in weights is done for values near 0.5,since the derivative of the sigmoid is maximum at 0.5. Values near 0 or 1 achieve the least weight change as these are now assumed to be ON or OFF.

There are however many variants of the major activation functions that have been applied to various ANN applications. Gomes et al. (2011) lists some of them as: hyperbolic, polynomial, periodic, sigmoidal, Gaussian, non-polynomial, transfers (rational, Lorentzian). They however state that the sigmoid or hyperbolic tangent type is the most frequently used. In their study, they propose a variant of the sigmoid function which they found suitable for their application. They state that implementation of new activation functions is generally simple, though the role of the activation function is very important on the working of the ANN. Ortiz-Rodriguez et al. (2013) states that multilayer perceptron (MLP) trained with back propagation algorithm is the 'most used ANN' in prediction, classification and modeling. Training can be fully supervised learning as used in classification or semi-supervised learning as used in clustering.

It is possible to apply the various algorithms in implementing ANNs i.e. unsupervised, supervised or reinforcement. It is also possible to have a network that is feed-forward or recurrent. The choice of parameters is application dependent. Training or learning is just the process of 'determining an optimized set of weights based on the statistics of the examples' (Smith, 1997). The purpose of training is therefore to establish a set of network weights, for each node, that can properly map any input data into the desired output. Learning enables us to estimate a set of parameters, say $\boldsymbol{w}$, to define the hypothesis $\boldsymbol{h_w(x)}$. After attaining the hypothesis, the training set can be abandoned, having all been summarized by $\boldsymbol{w}$. This type of learning is called parametric model learning. Learning generally formulates a methodology of adjusting the weight, $\boldsymbol{w}$, to fit solving of problems for the neural network. Additionally, training can be done using either of the following methods - Direct training, by adjusting weights or Indirect training, by adjusting the threshold values

For practical ANNs, there is usually the need to adjust the weights of all neurons automatically, after obtaining the final output. There are different methods of weight adjustment (training). These include simulated annealing, resilient propagation and backpropagation amongst others. Ghaffari et al. (2006) compares various methods used in training ANNs and goes ahead to make comparison about the three broad classes that they identified for their application. These training methods are: Gradient descent backpropagation algorithm (using incremental or batch backpropagation), Quasi-Newton / Levenberg-Marquardt (LM) backpropagaration algorithm and lastly, Genetic algorithm (GA). Their findings ranked the order of predictive abilities, from best, which was similar to the list above. Apart from their research showing that backpropagation was the best performing, they also state that this method is the most commonly used in many applications. Batch Gradient Descent computes weight updates after summing them over all the training examples (also called 'offline' learning). Incremental Gradient Descent on the other hand applies 'online' learning, where the weights are updated after each training example. There are many other variants of ANN training methods developed by different people. Gomes et al. (2011) singles out LM and Fletcher-Reeves conjugate algorithm in their study.

Proponents of genetic algorithm (GA) however suggest that GA is part of the tenets of building AI systems, specifically Intelligent Machines. These tenets are logic-driven search, cultural search (library of knowledge) and evolutionary search. These are attributable to Turing, considered as one of the founders of AI. Koza (2010) lists 77 application areas of GA, terming the results as being 'competitive with human-produced results'. The fields are many, including electrical works, protein identifications, filters, sorting algorithms, computation etc. None of the works however relate to the stock market. Chang et al. (2008) also advocate for inclusion of GA in the training of the ANN. We also have the non-parametric model training, where each hypothesis possesses the characteristics of the particular training set, using the data to predict the next example. Such learning is called instance-based or memory-based learning as applied in table lookup examples. This type of algorithm can be extended to find out the nearest neighbor to the lookup, resulting into the k-nearest neighbor (KNN) algorithm. If the problem does not incline to the table lookup application, then the solution would tend to be a parameter model learning. Nonetheless, Cerna et al. (2005) still contend that ANN is advantageous even in solving non-parametric problems. This enables the ANNs to be useful in many problem areas e.g. data analysis 'irrespective of their statistical properties'.

The above studies agree that backpropagation is a popular ANN training method. The Encyclopedia of Mathematics (2013) explains the backpropagation algorithm as a method of learning, which begins with an initial NN and random weights then computes errors resulting from the classification. Thereafter, small adjustments are made on the weights of the network nodes. The adjustment is by back propagation. The aim of this process is to reduce the overall classification error. The process is repeated severally until an acceptable error rate is attained. Learning or training is the process of adjusting the weights on the network elements to attain an optimum combination of weights that are general enough to solve any new problem. Backpropagation can employ algorithms such as Hebbian learning (change neuron weight depending on pre- and post-neuron strengths) or Widrow-Hoff method (aims at minimizing mean square error).

Having established that backpropagation is a popular algorithm for most current applications, it is therefore desirable to train the ANN using the backpropagation method in a multi-layer network scheme. There is no significant performance gain in hybrids or enhanced training algorithms. A challenge in ANN design is the selection of the optimum number of units that are large enough to fit the purpose but not too large that the ANN fails to generalize the solution (over-fitting).

Russell et al. (2009) recommends that one needs to just try several settings and keep the best. That involves training on a particular setting and testing the efficacy of the settings, then repeating for different settings before choosing the most optimum. One approach, the optimal brain damage, starts by training a fully connected network, then dropping units while observing the performance after retraining. A converse approach is the tiling algorithm, which trains from a single unit and is progressively built up with additional units while monitoring its performance.

There are several issues to consider in the design of ANNs. These issues are: Type of network, Type of training, Proportion of Training and Testing data sets, Number of input and output units, Number and size of hidden layers, Number of repetitions during training (epoch), Choice of activation function and Size of data set.

a) Type of Network:

ANNs can be feedforward or recurrent. There is also need to determine the method of adjusting the neuron weights, which is usually by backpropagation. Backpropagation calculates the value of errors from output towards input neurons, with a view of minimizing the error. Neurons contributing to higher degree of error are also subjected to higher degree of error correction. A variant of backpropagation, called resilient propagation, tries to avoid the possibility of the ANN getting stuck on a local minima by using only the sign of the error gradient as positive, zero or negative. Applications that can use backpropagation are also candidates for Resilient propagation algorithms (Code Project, 2011)

b) Type of Training:

For a case such as stock market data, the historical data is available in totality. This data has values on the respective dates of trade. Training of the ANN is based on this labeled data. Thereafter, testing should confirm the efficacy of the system for the data that the agent has not been exposed to. For this particular application, with this type of data, the learning type is therefore supervised learning. There are three commonly used training protocols or regimes. The first is stochastic, where the training samples are chosen randomly. In the second, batch method, all the data is presented to the algorithm before training. The third method is online, where sample is presented to the classifier only once and no memory is used (Wagacha, 2003).

c) Proportion of Training and Testing sets:

On the determination of proportion of data for use as training set, Russell et al. (2009) suggests that the ANN learning process should be exposed to as much of the available data as possible, instead of relying on a small fraction. In their paper, Ortiz-Rodriguez et al. (2013) state that their methodology of Robust Design of Artificial Neural Networks (RDANN) has enabled them to propose a training set of 80% of the data and the balance 20% for testing. On the subject of selecting training and test data, Microsoft (2013) states that its SQL Server 2012 product has a default of 70% as training data and 30% as testing data when undertaking data mining tasks. In considering data sets and their efficacy, Keogh et al. (2003) suggest that a wide dataset is necessary in testing any algorithms, with one subset for fine tuning an approach, while the other set for actual tests. They also advocate for designs that are free of implementation biases, so that such can be applied to different datasets to realize similar results. These best practices should be applicable to any time series data mining research in areas such as indexing, clustering, classification or segmentation.

The inherent issue of the need for large datasets is also noted by Giles et al. (2001), who state that many models could fit a training set very well, though only few would ultimately provide a good enough generalized solution. High noise on financial data combined with small datasets usually result into over-fitting. Over-fitting is the effect of finding a solution that is particularly suited for the training set, but not general enough to be used on the testing set. The only solution to such a problem is to use a training set with a dataset that is as large as possible. Ben-David et al. (2010) looked at the issue of training and testing sets and the effect of data volumes in these sets. Their work also tried to answer the question on what would happen if the target domain was little known (such as prediction based on testing set) or unknown (such as language processing). They concluded that several mathematical relationships can assist in achieving an optimum level of training sets that guarantees low error rates. Therefore, ultimately, a choice needs to be made on what to use based on the problem and how efficient the solution can be, however, a training set of 70% to 80% is considered optimal. Proportioning methods such as bootstrap, holdout, cross-validation, and bagging can be used, though holdout was the popular method is the above studies i.e. two distinct data partitions, one for training and the other for testing.

d) Input, Hidden and Output units:

The number of units at the input and output layers are application dependent. However, the number of hidden layers and the number of neurons in each hidden layer is a difficult parameter to determine. In proposing designs for evolutionary neural networks, Han et al. (2006) contends that application of NNs to real world problems needs a delicate balance on the topology of the ANN, which is usually problem specific. They argue that repeating 'trial and error' cycles based on previous experiences of similar problems may be the way out. However, their evolutionary neural network (ENN) design tries to automatically configure the topology of the NN. Their most optimal design had 10 inputs, 15 hidden nodes and 2 output nodes. On their part, Ortiz-Rodriguez et al.(2013) used their RDANN method to suggest a configuration of 7:14:31, instead of 'trial and error'.

Others suggest that the rule of thumb for hidden neurons per layer should be 2N +1, where N is the number of input nodes (Neuroph, 2013c). Other considerations for deciding the number of hidden neurons include: A number between number of input neurons and number of output neurons, Sum of output neurons and two-thirds the size of input neurons or Less than two-times the input neurons (Heaton Research, 2013c). Caution should be taken in deciding on number of neurons in the hidden layers. Too few neurons in this layer would result into under-fitting (inability to detect the pattern), while too many would result into over-fitting (inability to generalize based on the training data) and increased training time.

Concerning the number of hidden layers, Heaton Research (2013c) suggests three issues to consider and adopt. One is to have no hidden layer at all, which should be used for linearly separable problems or decisions. A second suggestion is to have 1 hidden layer – should be used for approximating continuous mapping of finite space to another finite space and finally, to try out 2 hidden layers for use in arbitrary boundaries providing approximations with any accuracy. Though it may be assumed that the number of inputs depends on the problem, and hence a trivial decision, it is worth noting that the prediction accuracy increases with increase in number of inputs (Code Project, 2011). That means that it may be desirable to provide a range of inputs (array), even for a single input variable, with a corresponding range of outputs (array). This would be a better strategy that a single input for a single output.

The starting point on ANN design should therefore be one hidden layer, with at least an equal number of neurons as the input layer, before moving on to trial on various network configurations. To help optimize the network topology, methods such as pruning (incremental or selective) can be used to remove neurons that do not contribute much to network performance. Pruning can be implemented as part of the overall ANN design, for dynamic modification, or as a strategy for obtaining optimum topology for practical implementation for particular problems. This however increases the complexity of the programming or the stages of running the algorithm (Heaton, 2008)

e) Number of Repetitions during Training:

Training involves feeding the input pattern to the network in order to generate an output pattern, then comparing the generate output with the actual true output to determine an error value. The next data set is fed to the network and neuron weights adjusted with a view of reducing the error. This process is repeated to the end of the dataset, then back to the start of dataset, until the error is significantly small. A run through a complete dataset is called an epoch. Naeini et al. (2010) suggests a MLP neural network that they use to predict next day stock value of companies based on the Tehran Stock Exchange. Their prediction error on their model was as low as 1.5%. Their model inputs three values of lowest, highest and average stock value and provides one output for the next day stock value. It is a three layer model. They do not state the number of neurons on the hidden layer, but the learning epochs are restricted to 1,000 maximum. Most experiments relating to ANN for prediction, especially financial data, has set the maximum repetitions at 1,000. After that count, the training is restarted with random weight values in case the final error is not acceptable.

f) Choice of Activation Function:

The type of NN being developed should determine the activation function in use. The output expected from the network is the general guide to the choice of activation function. If there is need for a backpropagation algorithm, then the choice of the activation function must be a function that can be differentiated. This is because backpropagation uses the gradient of the error in each repetition, to determine the magnitude of error in the learning process. Functions that can be differentiated includes sigmoid function and hyperbolic tangent function. Even with this decision, the expected output values still determines whether the results of the function are positive numbers only e.g. sigmoid or both positive and negative values e.g. hyperbolic tangent. Stock prediction applications would generally deal with positive stock price values, hence the use of a sigmoid function.

g) Size of dataset:

The size of dataset to use on any ANN experiment is application dependent. For financial applications such as the stock market, any data over 3 years is considered representative. Devi et al. (2011) analyzed values over a three year period, while testing a stock trend analysis system to aid in prediction. In another study, Butler et al. (2009) decided on a 23-year period to test the data through a series of financial cycles. The parameters used for their study were also observable over longer terms e.g. 3-month, 6-month and 1-year Treasury bill rates, which would therefore mean only 23 observations over the study period, in the case of the annual Treasury bill rates. Adhikari et al. (2013) used data sets of between 3 years and 4 years for data containing daily price movements. Wong et al. (2012) considered a four year period in their financial time series application using NNs. These studies suggest that any period over three years seems reasonable, especially for general indices and probably longer for individual stock value analysis. However, in general, NNs work better with a big volume of data sets for better learning and hence good predictions.

h) Other parameters:

Other parameters that can be set on ANNs include a Learning rate and the Momentum. During error backpropagation, the ANN determines both magnitude of change for each node/bias and the rate of improvement for these weights on the network. Based on these settings, the user can determine the modification factor or step size. This is called the learning rate, which is a real value in the range 0 – 1. A value close to 1 would mean faster learning, but this can lead to under-fitting. On the other hand, the momentum allows the adjusted weights of previous iteration to persist on the next iteration. A value greater than 0 means that the effect of the last weight adjustment is manifested in the current weight adjustment. A working strategy based on the two settings is to start training with high learning rate and low momentum and progressively switch the two as training progresses. These settings can be altered programmatically to achieve this objective.

While many studies agree that AI methods, and particularly ANN, can be used to build tools to inform investment decisions at the stock market, none of the studies available on the public domain has applied such methods to the Kenyan market - specifically the NSE. This project therefore aims at designing and developing an ANN-based predictive system specifically suited for the NSE. Since in the Kenyan market it is the Stockbrokers who directly undertake trades, and they do not have such tools, the research shall target them as the beneficiaries of this tool. The particular ANN shall be a feedforward multi-layer perceptron with backpropagation, using supervised learning. It is possible to design ANN models and develop such to practical tools. Development can be done using a programming language. However, it is also possible to use tools that have already been developed. Some of these tools are commercial while others are free or open source (Neural Forecasting, 2013). These tools include the Neuroph framework (Neuroph, 2013b), NICO toolkit (Nico, 2013), Matlab, Java Object Oriented Neural Engine, Joone (Java Object Oriented Neural Engine (Joone), 2013), Encog framework (Heaton Research, 2013b) and Tiberius predictive modeling software (Tiberius, 2013). These can be used to not only test an ANN network, but can also validate results got from a tool that one may develop using a programming language.

2.6 Applications of ANN in Stock Market Prediction

In forecasting noisy time series data, such as the financial market, Giles et al. (2001) recognize the inherent challenges that need to be considered such as non-linearity, smaller samples, high noise and non-stationarity. They state that ANN find applicability in such challenging environments due to their coping mechanisms. Adebiyi et al. (2012) state that ANN technique is a candidate for stock market prediction. The reasons for this assertion is the ANN’s ability to learn and detect relationships among nonlinear variables and deeper analysis of large data sets that tend to fluctuate over short time. They also mine valuable information from mass history of data. Thulasiram et al. (2007) also acknowledge the prevalence on ANN in stock market prediction. They however point out that a longer training time tends to give these methods a disadvantage compared to others such as soft computing e.g. fuzzy or instance based algorithms such as nearest neighbor.

ANNs are good in stock market prediction, though their main drawback is their reliance on historical data, which may not easily reflect the dynamics of the future. It is for this reason that others have proposed a reinforcement learning approach to augment the performance of ANNs. Chen et al. (2007) and Chen et al. (2007b) proposes such a classifier system approach in a study of predicting the Taiwan Index, though in their study they compared the classifier approach to the random purchase model. Other studies have been done to try and compare statistical methods of forecasting financial time series data. Gomes et al. (2011) state that such methods include auto-regression, moving averages, linear or non-linear regression and exponential smoothing models. Others include auto-regression integrated moving average (ARIMA). They state that ANNs seem to be the solution to non-linear time series forecasting, such as energy consumption, weather and financial series. Characteristics of financial time series is data intensity, unstructured, uncertainty and hidden relationships. ANNs seem to deal well with such data characteristics. They can deal with non-linear mapping between input and output.

Kaushik et al. (2009) recognizes that ANN and Support Vector Machines provide better prediction systems for financial markets than other statistical methods. They however, suggest that a hybrid of both could realize better results by combining their various strengths. However, their study indicates that it is easier to predict a pattern (up/down movements) than the actual prices. In a separate study, Zarandi et al. (2012) also recognize the prevalence of ANN, fuzzy logic and GAs as the AI methods popular in dealing with complex engineering problems, such as stock market prediction. They however propose that a hybrid of such methods is capable of better performance than single models. Though hybrids perform better, they are a challenge to design and implement. Chang et al. (2008) suggest augmenting the traditional ANN with a piecewise linear representation as part of the learning algorithm to improve on its performance. In their design, even a GA is used to improve their Piecewise Linear Representation (PLR) algorithm. The design however aimed at determining turning points (peaks and troughs) to inform a buy or sell decision. It was not for prediction of exact stock price performance.

Beating the stock market remains a quest for many computer scientists. Borodin et al. (2004) states that though possible, it is difficult to gain advantage by predicting the prices themselves. They suggest a portfolio selection algorithm that should guarantee a good mix for maximum profit. They suggest that ‘beating’ the market is a matter of algorithm for performance, which is only possible by portfolio selection (knowing the different stocks to have at any given time), than the actual price movement of particular stocks. These studies confirm that ANN is the AI tool that can be used in stock market prediction.

2.7 Conceptual Framework

Review of the literature enabled the research to develop the conceptual framework shown in figure 2.5 below. This framework shows the various relationships between the research variables and targets the particular area identified for the study.

The overview of the research problem was the relationship between the shareholder on one end, and the stock market on the other end, with the shareholder having an objective of making a profitable trade at the stock market. However, there is no direct link between the shareholder and the stock market. This link is only made possible through the stockbroker, who sits at the middle of the chain. The shareholder uses the date, stock and price variables to trade. The stockholder makes a decision (decision guide) and communicates to the Stockbroker. The stockbroker on the other hand, uses some method in deciding what to buy or sell. The current methods rely on trend, while the missing method in the stack is an AI method, that should not only guide on the trend, but also the most likely stock price. At the extreme end is the stock market, where this research targets the ‘equity’ segment, which has a low entry level, making it easily accessible to many Kenyans. Finally, there is feedback on the success of the trade that comes back to the shareholder.

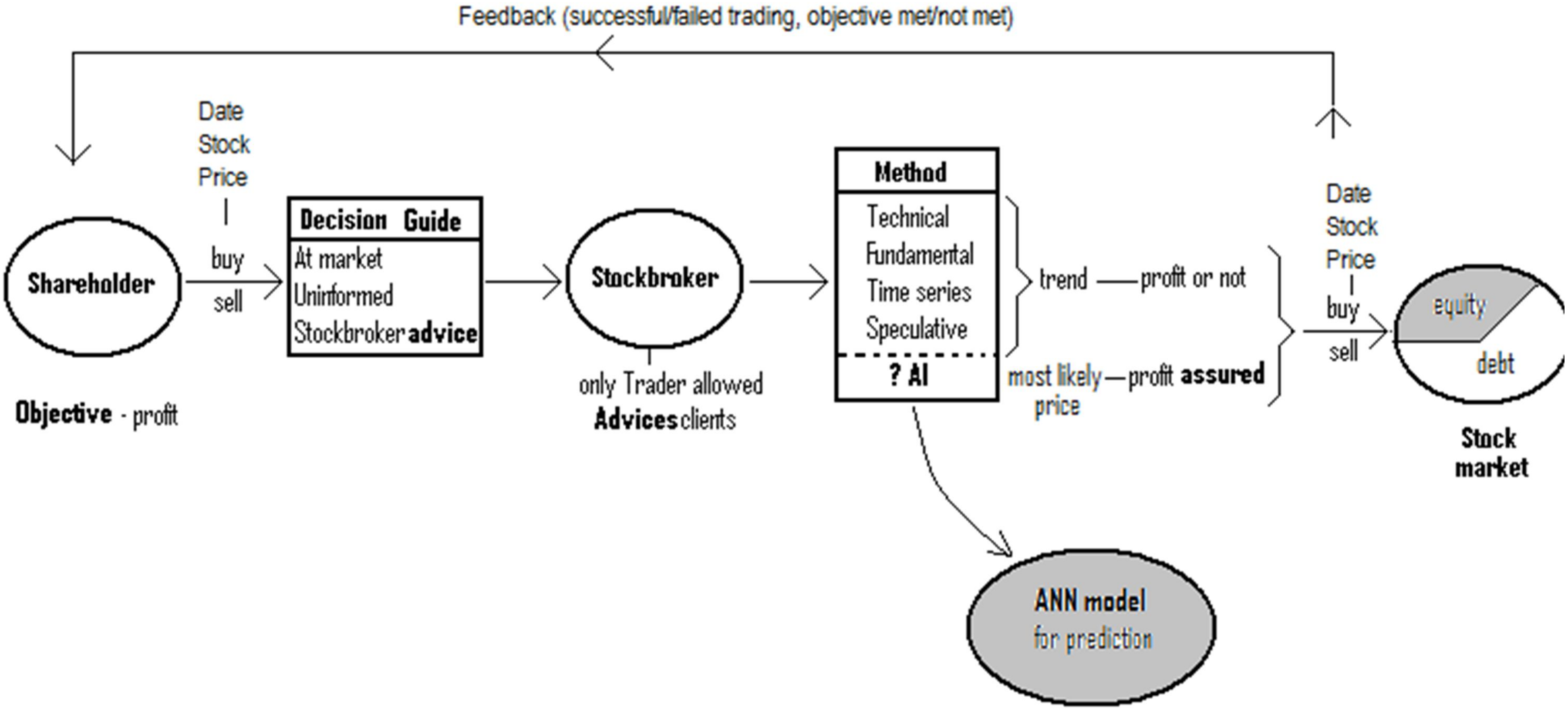


Figure 2.5 – Conceptual Framework (Source: Author)

2.8 Chapter Summary

Chapter 2 presented the operations of the Kenyan stock market and highlighted how stock trading is done. The Stockbroker was singled out as being important in the trade cycle. The stockbroker also acts as an advisor to the client. The chapter confirmed that the Kenyan Stockbrokers did not seem to have any AI tools to guide their investment decisions. The chapter also provided information on AI systems and specifically the ANN agent. It then looked at the theory behind ANN, why it was popular and the various application areas, including stock price prediction. Finally, a case for the need to model an ANN system to predict stock prices at the NSE was stated. Such ANN tools is expected to be helpful to Stockbrokers, who advise their clients on investment decisions at the NSE. The next chapter shall design and develop an ANN model that can predict the stock prices at the NSE, based on data on stock price movement.

CHAPTER 3
RESEARCH METHODOLOGY

3.1 Research Design

This project used applied research to create a specific artificial intelligence (AI) tool based on a model and tested its performance on a practical problem. Chapter 2 indicated the need for such a tool that could be used by Stockbrokers to advice on investment in stocks and highlighted the current lack of such a tool. This research therefore applied a particular underlying AI method to a practical field, in this case, the Kenyan stock market. The direct beneficiary of this application is the Kenyan Stockbroker. Applied research aims at solving a ‘societal or business’ problem (Kothari, 2004).

In this research, an AI model was designed, based on the artificial neural network (ANN) algorithm. This was then developed into a working prototype for purposes of testing. The specific AI agent that was developed was a feedforward ANN with multilayer perceptron using backpropagation and trained using supervised learning, using the holdout method. The programming language environment used was C#. Developing a computer program required a computer with the programming language and necessary utilities for coding and testing the program.

Using ANN for stock market prediction could also be done using existing tools, making it possible to validate the performance of the proposed model-based prototype. This was done by comparing the performance of the prototype with similar tools developed by others. Some of these tools are already available on the public domain, as free, shareware or open source.

A generic ANN prototype could be applicable to any situation where there was need for determination of relationships between variables (matching of input-output pairs). However, practical real life data was desirable for this research, since it related to a real situation, being the work done by Stockbrokers in Kenya. This real life data was more informative than other types of data e.g. auto-generated or random. For that reason, the tool developed was tested on data of daily trades from the Kenyan stock market. The study employed mainly quantitative approach, by using existing raw data as a basis of generating new labels, hence creating relationships between the variables. The results were subjected to analysis such as error rates and percentage accuracy, in numerical terms.

The data used to evaluate the performance of the model-based prototype was the daily closing price of individual stocks over a five year period, 2008 to 2012. The choice of the most recent data enabled the research to relate to the most current data, and apply the prediction ability to the current situation. This provided a direct simulation of usage at the stock exchange as it is. This data was secondary data obtained from the Nairobi Stock Exchange (NSE) or other authorized sources. The use of secondary data was desirable in this study because the research aimed at demonstrating the applicability of a particular tool to a practical business problem. The data about the NSE stocks was already available. By using publicly available data, the research experiments can be conducted independently by anyone accessing this data. The data itself can also be authenticated independently.

The final test data was obtained from carefully chosen stocks of the NSE, which met particular criteria that ensured data consistency. This choice ensured that erratic patterns in the data stream were removed as much as possible. Factors that caused such erratic data movements were determined and addressed. These factors included share splits and bonus issues and also economic, social and political factors. The ANN prototype was subjected to both a training phase and a testing phase. The holdout method was chosen, where a portion of the available data was used for training, while the balance was used for confirmation of prediction accuracy. The research variables were – the date of trade as the independent variable, while the stock price of the day was the dependent variable.

The selection criteria for qualifying research project data generally mimicked the actual behavior of Stockbrokers during trade e.g. what they would do when dealing with stock trade for companies whose shares were split or bonuses issued. In such cases, the Stockbrokers would factor in the split/bonus onto the new prevailing stock price. This factoring does not however show on the published closing stock costs. Instead, the datasheets usually include a footnote to explain the usually erratic price jumps when this happens. This research therefore deliberately avoided these Stockbroker-specific data manipulation decisions and concentrated on an otherwise normal trading situation, for stocks that had not had a split or bonus issue. The stock market, as part of the wider national development sector, could be affected by economic, social and political factors. The effects of such factors on the work done by Stockbrokers were determined and evaluated to determine if there was any adverse effect that they may have had on the normal trade e.g. the prevailing quoted prices.

The performance of the ANN tool was analyzed based on its accuracy in predicting stock prices for a contiguous range of dates beyond the last date of its training. The normal trading dynamics usually ties trades from one day to the next day, and hence it was not desirable to test random dates but rather continuous back-to-back dates e.g. closing numbers of one day forms the opening numbers of the next day trade. Training effectiveness was determined using the root mean square error (RMSE), over the range of training cycles. RMSE was also used to compare the performance of different tools on the same data, where a lower comparative RMSE value would imply a better prediction. However, the testing phase was measured on the basis of mean absolute percentage error (MAPE), to determine exactly how far the actual and predicted values were. Results with lower MAPE were more close to the actual values. In other research, Deng et al. (2011) also based the measurement of the performance of their model on the selected stocks at the New York Stock Exchange on the basis of MAPE and RMSE, while Neto et al. (2009) also measured performance of their model for a Brazilian stock company using MAPE and Mean Square Error (MSE). The formula for RMS error (RMSE) is given by Equation 3.1 while that for MAPE is given by Equation 3.2 below:

$$RMSE \;=\; \sqrt{\frac{\sum_{i=1}^{n}(\hat{y}_{\boldsymbol{i}} - \boldsymbol{y_i})^{\boldsymbol{2}}}{n}} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ 3.1$$

Where:

$\boldsymbol{n}$ = number of observations

$\boldsymbol{\hat{y}_i}$= predicted value

$\boldsymbol{y_i}$= actual value

$$MAPE\ (\%) = \quad \frac{(\sum_{i=1}^{n} |(\hat{y}_{\boldsymbol{i}} - \boldsymbol{y_i})}{n} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ 3.2$$

Where:

$\boldsymbol{n}$ = number of observations

$\boldsymbol{\hat{y}_i}$ = predicted value

$\boldsymbol{y_i}$ = actual value

Though it was possible to test all values on the test dataset, the test dataset was huge (about 1 year), hence the need to test just a subset of the total size. Since the type of data was homogenous, it was possible to use smaller samples (Kothari, 2004). Other aspects that determine sample size include nature of study, number of classes, type of sampling, accuracy, confidence level and population size. Sampling can be classified by representation (probability or non-probability) or by element selection (restricted or non-restricted). Due to the type of data available for testing, the sampling used was unrestricted based on non-probability convenience method.

3.2 Research Data

For purposes of designing and evaluating the model, the research needed data from a typical stock exchange, in this case the NSE. The research therefore compiled data on the performance of all stocks traded at the NSE. In compiling the data, there was need to give some thought on the source, volume and suitability of data.

3.2.1 Sources of Data

Sources of NSE data were many. The NSE itself was the primary source of the data. The data from NSE was available on its website (Nairobi Securities Exchange, 2013b). The data on the website was for the previous day's trade, at any day of access. That meant that one needed daily access to the site in order to get the previous day's data. This data was in hypertext markup language (HTML) format. Such data could be copied from the website to spreadsheet format. The NSE did not provide free access to historical data. This type of data could only be accessed by paid subscription. A screenshot of the NSE website showing the daily trade stock prices is shown on **Appendix 5**. Despite the daily data being available at the NSE website, the NSE had also licensed several data vendors who were allowed to get the data from NSE and relay it to end users. Nairobi Securities Exchange (2013) listed the ten licensed data vendors who could provide the NSE data to the end users. Some of these vendors provided the data free of charge, both for the current day's trade and also historical data.

This research therefore used secondary data available on the public domain from authorized data vendors. Synergy Ltd. was listed as one of the licensed data vendors (Nairobi Securities Exchange, 2013). Synergy Systems Ltd. (2013) provided both current and historical daily trade prices for all NSE listed companies. All historical data used for this research was obtained from the Synergy site. Typical screenshot of the data screen from the Synergy website is shown on **Appendix 6**. This data was available in HTML format and hence had to be copied from the webpage to a spreadsheet.

3.2.2 Volume of Data

The stock market data has patterns that generally manifest over a long term. Clear patterns start to emerge when about 1,000 data items of daily trades are considered. This volume of data implied a period of almost 5 years. In other studies of stock market prediction, other researchers considered periods that varied from 3 year to 23 years. Devi et al. (2011) did a test over a three year period while Adhikari et al. (2013) used data sets of between 3 years and 4 years. On the other hand, Wong et al. (2012) considered a four year period while Butler et al. (2009) used data that was upto 23-years. Due these considerations, the project decided on a 5-year period i.e. January 2008 to December 2012. This five-year period had 1,250 trading days. Some of the stocks however did not trade on some of the dates on this range. The stock market rules however dictate that the last traded price remains prevalent, until the next successful trading date. In this research therefore, the previous day's prices were included for all dates when trade did not take place during the period under review.

3.2.3 Compiling the Data

Data obtained from Synergy was copied from the website to spreadsheet format. The other aspects of the data compilation done were the formatting of the spreadsheet data so that each worksheet contained data for a single daily trade. With about 250 working days in a year, the five year dataset comprised about 1,250 datasets of daily trades, resulting into about 1,250 worksheets. Each spreadsheet had about 70 data rows, for the 60 listed companies and the various headers extracted from the website. The data from the individual daily worksheets were then compiled into one single sheet with about 87,000 data rows. The combined worksheet was reformatted to include a date column and then sorted by company name. The resulting data sheet had each company showing its daily trade for the period January 2008 to December 2012. The data was then sorted by company name and then by date.

3.2.4 Selecting NSE Companies for the Study

The five year period of study was the period between January 2008 and December 2012. In that period, the NSE saw various changes in the stock market. These included the introduction of new companies to the exchange, removal of some companies from the exchange, suspension and later reinstatement of some companies, share splits (which affected trading price by a factor of the split), bonus share issue (which affected trading price by a factor of the bonus) and change in companies that formed part of the NSE 20-share index.

To enable a stable and non-erratic data analysis, the following criteria was used to select companies whose stock data were used in training and testing of the ANN prototype. The stock was to be part of the 20-share index, due to the importance of this index as a barometer of the economic activity in the financial sector. A second consideration was that the stock should have been consistent in its trading activities by being available for trade at the NSE in the 5-year period i.e. not introduced, removed or suspended from trade in the period under review. A final consideration was that the stock should have had no share split or bonus shares issued in the period. This was to remove the erratic change of price that occurred during splits and bonus issues and hence provide a mathematically realistic movement of prices on the stock exchange. Table 3.1 below lists all the constituent counters of the NSE 20-share index and their status based on the criteria above.

Table 3.1: *Analysis of NSE 20-share Index Constituent Stocks*

	Company stock	**5-year trading**	**Split status**	**Bonus Status**
1	Kakuzi Ord.5.00	Y	N	N
2	Sasini Ltd Ord 1.00	Y	Y, 5 to 1 in 2/ 2007 (see 1)	N
3	Barclays Bank Ltd Ord 2.00	Y	Y, 4 for 1 in 2/ 2011 (see 2)	N
4	Equity Bank Ltd Ord 0.50	Y	Y, 1 to 10 in 2/2009 (see 2)	Y 2:1 in 02/ 2007 (see 2)
5	Kenya Commercial Bank Ltd Ord 1.00	Y	Y, 10: 1 in 04/ 2007 (see 2)	N
6	Standard Chartered Bank Ltd Ord 5.00	Y	N	N
7	The Co-operative Bank of Kenya Ltd Ord 1.00	Y	N	Y, 1:5 in 03/ 2012 (see 2)
8	Kenya Airways Ltd Ord 5.00	Y	N	N
9	Nation Media Group Ord. 2.50	Y	Y, 2 to 1 in 03/ 2008 (see 2)	Y, 1:10 in 03/2010 (see 2)
10	Scangroup Ltd Ord 1.00	Y	N	Y, 1:5 in 04/ 2011 (see 2)
11	Uchumi Supermarket Ltd Ord 5.00	N[1]	N/A	N/A
12	Athi River Mining Ord 5.00	Y	Y, 5 for 1 in 05/2012 (see 3)	N
13	Bamburi Cement Ltd Ord 5.00	Y	N	N
14	KenGen Ltd Ord. 2.50	Y	N	N
15	KenolKobil Ltd Ord 0.05	Y	Y, 10 to 1 in 06/2010 (see 2)	N
16	Kenya Power LIGHTING Co Ltd Ord 20.00	Y	Y, 1 to 8 in 10/2010 (see 2)	Y, 1:8 in 10/ 2011 (see 2)
17	British American Tobacco Kenya Ltd Ord 10.00	Y	N	N
18	East African Breweries Ltd Ord 2.00	Y	N	Y, 1:5 in 10/ 2007 (see 2)
19	Mumias Sugar Co. Ltd Ord 2.00	Y	N	Y, 2:1 in 11/ 2007 (see 2)
20	Safaricom Ltd Ord 0.05	N[2]	N	N

N[1] – Suspended from trade, resumed May 2011 (Uchumi, 2013)
N[2] – Initial Public Offer took place in June 2008, leading to public listing on NSE that month (Safaricom)
1 – Source: (African Fine Coffees Association (AFCA), 2013)
2 – Source: (Wazua, 2013)
3 – Source: (Sterling Capital, 2013)

From the above analysis, the stocks that qualified for this study were singled out, as shown in Table 3.2 below. The research could therefore test any of the six qualifying stocks. Their market segment, as shown on the table, represented a sampling of 6 of the 7 sectors that form the NSE 20-share index. This also represented a sampling of 6 of the total 10 sectors in the NSE trading floor.

Table 3.2: *Qualifying Stocks for the Research Project*

	Company stock	5-year trading status	Split status	Bonus Status	Market Segment
1	Kakuzi Ord.5.00	Y	None	None	Agricultural
2	Standard Chartered Bank Ltd Ord 5.00	Y	None	None	Banking
3	Kenya Airways Ltd Ord 5.00	Y	None	None	Commercial
4	Bamburi Cement Ltd Ord 5.00	Y	None	None	Construction
5	KenGen Ltd Ord. 2.50	Y	None	None	Energy
6	British American Tobacco Kenya Ltd Ord 10.00	Y	None	None	Manufacturing

Having determined the final six companies to be studied in the research, new datasets were generated by using the master data as the source data and then extracting data for each of the six companies identified above. This process resulted into six separate worksheets named as Stock01 (Kakuzi), Stock02 (Standard Bank), Stock03 (Kenya Airways), Stock04 (Bamburi Cement), Stock05 (Kengen) and Stock06 (BAT). Each datasheet had only two columns, for the date and stock value. Each of the above spreadsheets was then converted to CSV format, in readiness for feeding onto the ANN program. The CSV data had the same name as the above datasheets, for ease of cross referencing. This conversion resulted into six separate CSV files named appropriately as: Stock01.CSV, Stock02.CSV, Stock03.CSV, Stock04.CSV, Stock05.CSV and Stock06.CSV

3.2.5 Mitigating External Factors That May Affect the Data

It is possible to assume that the trading at the stock exchange operates in an ideal environment, dictated by the needs of sellers and buyers, without emotion or influence. This may however not be true. Several external factors may affect the main attribute of the research, being the price of the stock on a particular day. In section 3.2.4, the study had already mitigated some factors, such as share splits and bonus issues, which mathematically affected the price of any share. External factors however prevail, which may affect the experimental variables. Zarandi et al. (2012) lists the following as external factors that affect stock market performance – Economic factors, Political factors, Social factors and Psychological factors. Economic factors include issues such as the nation's economic growth over the years, and its influence on stock trading prices e.g. whether it affects trade appetite and hence the final prices. The economic growth over the five year period of consideration (Central Bank of Kenya, 2012) is shown on Table 3.3 below:

***Table 3.3:** Kenyan GDP growth from 2008 to 2012*

Year	GDP (%)
2012	4.38
2011	5.76
2010	2.74
2009	1.53
2008	6.99

From the data, the significant erratic pattern in GDP was observed in the 2008-2009 period, when the figure declined by over 5%. Thereafter, the growth/decline between any two successive years was between 1.2% and 3%. In this study, we assumed that the economic performance did not influence the stock market price. This assumption was due to the small change in GDP growth/decline over time (1.2-3%), apart from the 2008-2009 period.

Political factors generally relate to how stable the country is, in terms of a peaceful and free environment for doing business, including stock trading. The Central Bank of Kenya (2009) report states that the decline in economic growth in 2009 compared to 2008 was due to both political and environmental factors. Political reasons were the effects of post-election violence of early 2008, while environmental factors included a drought later that same year. Due to the close relationship between political factors and economic growth, this research assumed that there was no adverse political influence on the performance of the stock market during the period under review, since the only notable factor was the one incidence in 2008.

3.3 Designing the Proposed Model

The design of the model was done in the following general order. The first step required the determination of a prediction period that was practical for the NSE, so that such a period could be used in testing. From information available on the websites of stockbrokers and the

NSE itself, the period of future evaluation, based on other methods, was usually three month periods (quarters) since these were generally the cycles used in the financial reporting in Kenya. All research predictions would therefore be based on three calendar months, if possible, or the next 60 predictions, as determined by the end of training set. The second step was to be the formulation of a baseline model as per the proposals by other researchers. The third step would then be the development of the new proposed model based on the baseline model after change of parameters as determined by experimentation.

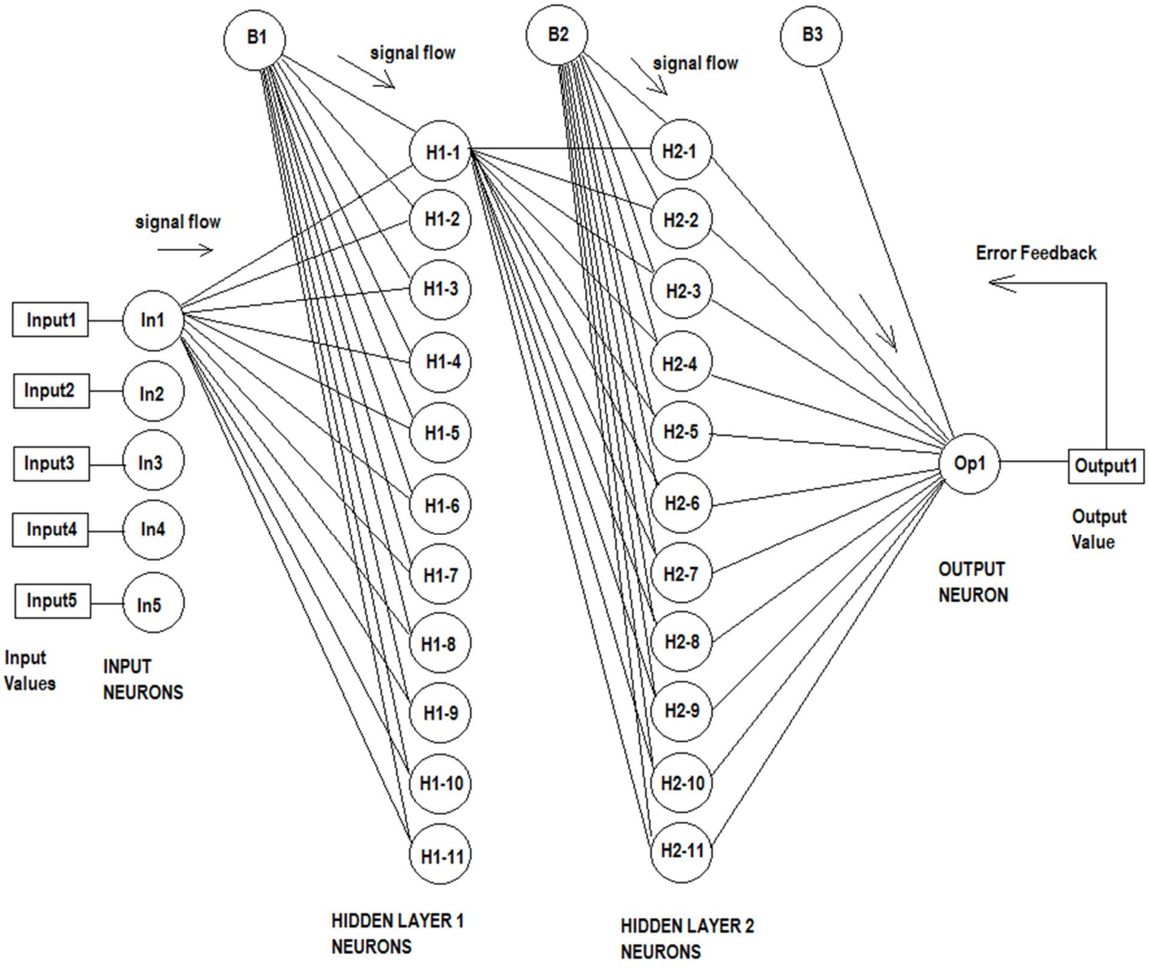
3.3.1 Formulating the Baseline Model

The number of inputs and outputs are determined by the problem domain. In this case, an input of 5 daily stock prices is used with the aim of predicting the 6th price in the series i.e. 5 inputs and 1 output. Based on other research, the hidden neurons should be 2 layers each with 2N+1 neurons, where N is the number of inputs (Han et al., 2006, Heaton Research, 2013b, Neuroph, 2013c).

The baseline model was therefore formulated with the following configuration:

a) Number of inputs = 5
b) Number of hidden layers = 2
c) Number of neurons per hidden layer shall = 11 i.e. 2N+1, when N is the number of inputs
d) Number of outputs = 1
e) Bias per layer = 1

The basic design is shown on Figure 3.1 below:



For clarity, neuron to neuron connection is only shown for In1, H1-1, H2-1, Op1 and biases B1, B2, B3. However, each neuron connects to others in same scheme e.g. In2 connects to H1-1 to H1-11 etc., while H1-2 connects to H2-1, H2-2 etc.

Figure 3.1 – ANN baseline model (Source: Author)

This baseline model was developed into a baseline prototype for purposes of experimenting on various network aspects so as to come up with a new model. The first experiment was done on all the 6 selected stocks. This test was used to determine one single stock, with the best MAPE, to be used in testing other parameters while in developing the new model. The baseline model, with a configuration of 5:11:11:1 was subjected to training using 70% data (02-Jan-2008 to 01-Jul-2011) and the balance for testing. The test period was 3-months (04-Jul-2011 to 30-Sep-2011). The number of training repetitions was set at 150,000 maximum, after which the training was stopped. The results obtained for each of the 6 stocks is shown in **Appendix 7**. The figure 3.2 below shows the graph of MAPE for the six stocks.

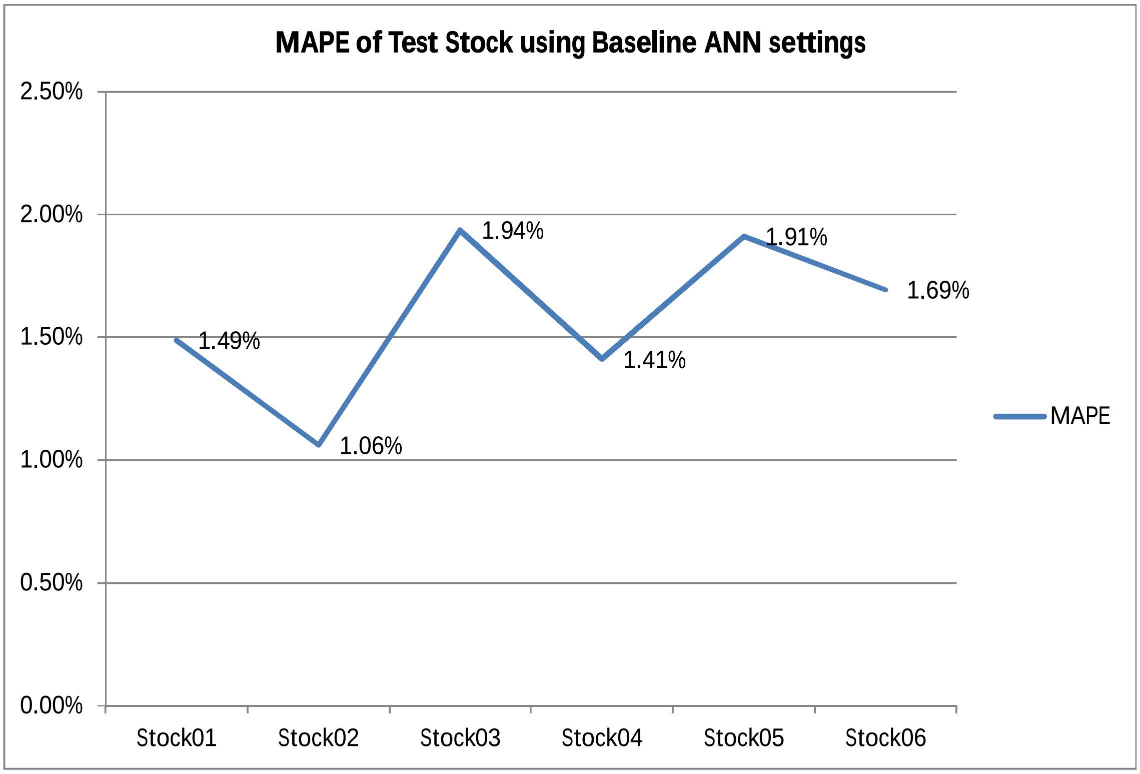


Figure 3.2 – MAPE obtained when testing the 6 stocks on baseline ANN settings

The following observations was made - based on MAPE, the best performing network was that of Stock 02 (Standard bank). Due to this observation, Stock 02 was chosen for conducting the subsequent configuration tests.

3.3.2 Determining Parameters of the New Model

3.3.2.1 Number of Neurons per Hidden Layer

The new model was initiated with a baseline of settings 5:11:11:1, using 70% data for testing at 150,000 training cycles. This baseline model was then tuned, through experiment, to determine an optimum number of neurons per hidden layer. Using Stock02, which achieved the lowest MAPE based on the design time baseline, and setting the number of hidden layers fixed at two, the number of neurons per hidden layer was progressively adjusted, followed by a series of training and testing phases. This test kept the number of hidden layers fixed as 2. The inputs and output were also fixed at 5 and 1 respectively. The data volume was 70% for training (Jan. 2, 2008 to Jul. 1, 2011) and 3 months for testing, out of the 30% test data i.e. Jul. 4, 2011 to Sep. 30, 2011. The results obtained for each setting are shown on Table 3.4 below, based on raw data shown on **Appendix 8**.

Table 3.4: *Stock02 (Standard Bank) Results of varying no. of hidden neurons – using Developed Prototype (NSE)*

Input Neurons	Hidden Layer 1 neurons	Hidden Layer 2 neurons	Output Neurons	MAPE (%)	RMSE
5	1	1	1	1.721	4.49
5	2	2	1	1.041	2.90
5	3	3	1	1.035	3.03
5	4	4	1	1.021	3.02
5	5	5	1	1.022	3.03
5	6	6	1	1.030	3.07
5	7	7	1	1.054	2.99
5	8	8	1	1.046	3.01
5	9	9	1	1.146	3.18
5	10	10	1	1.032	3.06
5	11	11	1	1.116	3.31
5	12	12	1	1.419	5.28
5	13	13	1	1.049	3.01
5	14	14	1	1.093	3.27
5	15	15	1	1.906	9.93
5	16	16	1	1.095	3.31
5	17	17	1	1.091	3.31
5	18	18	1	1.036	3.00
5	19	19	1	1.378	5.12
5	20	20	1	1.220	3.74
5	21	21	1	1.019*	2.99
5	22	22	1	1.060	3.08
5	23	23	1	1.022	2.99
5	24	24	1	1.113	3.35
5	25	25	1	1.128	3.37

*Lowest value

These results are also shown graphically on Figure 3.3 and Figure 3.4 below.

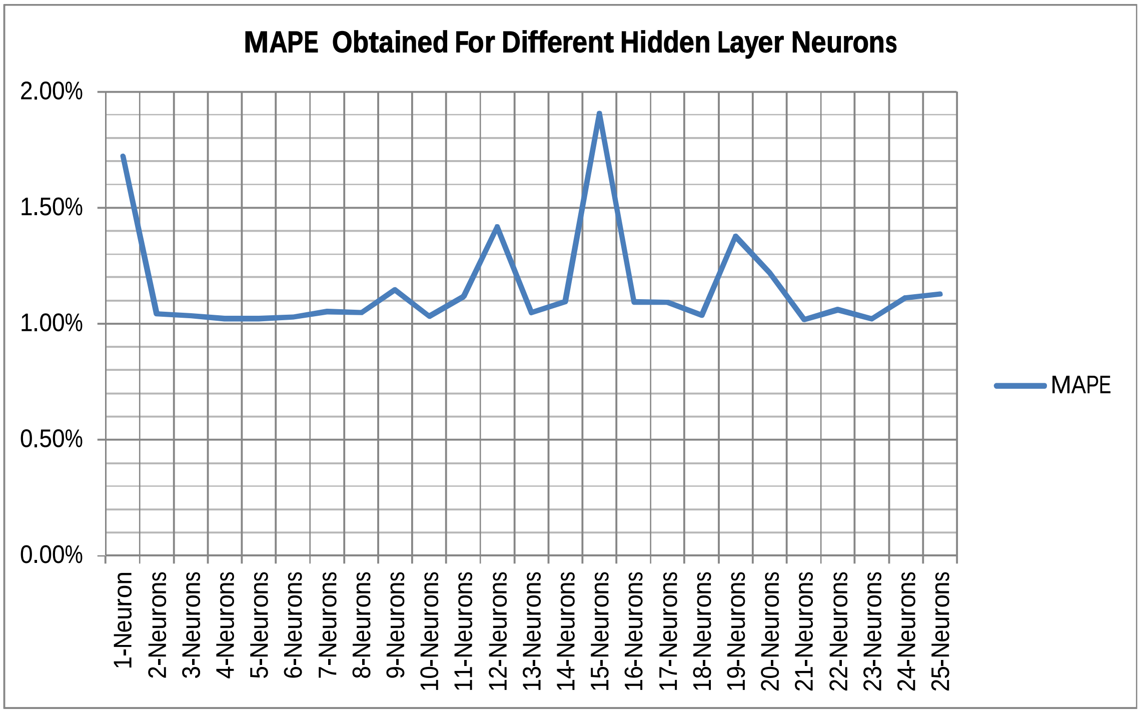


Figure 3.3 – Stock02 (Standard Bank) MAPE for different settings of neurons per each of the 2 hidden layers

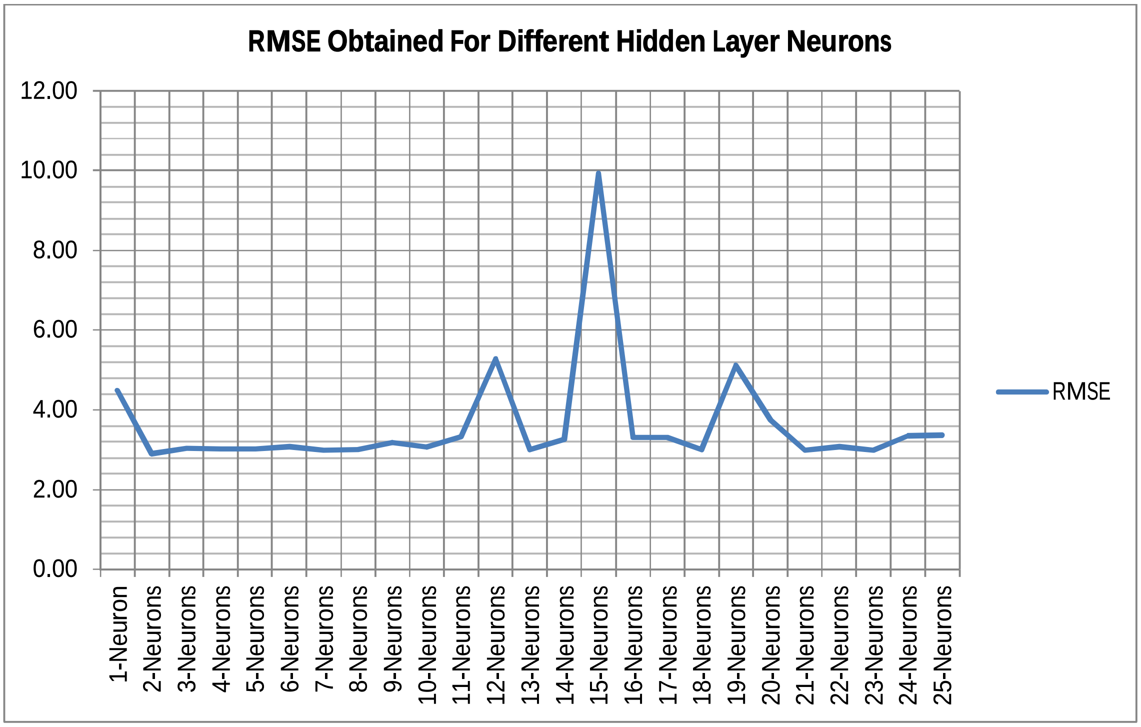


Figure 3.4 – Stock02 (Standard Bank) RMSE for different settings of neurons per each of the 2 hidden layers

The following two observations were made – firstly, the low rates of MAPE was obtained with neurons per hidden layer of 4, 5, 21 and 23. Secondly, based on MAPE, the best configuration was obtained for 21 neurons per hidden layer i.e. 5:21:21:1. Based on this determination, the configuration of 5:21:21:1 was therefore the optimal ANN configuration.

3.3.2.2 Determining the Proportion of Training Set

After obtaining the optimum ANN settings in terms of number of neurons per hidden layer, the network could then be exposed to varying data volumes and used for prediction to determine the effect of data volume on the training. This experiment was also be done based on Stock02, starting with proportions of 10% training data, increasing it to 90% training data. The maximum data available was the 5-year data for the period 2008-2012 (1250 data rows). Starting from the same start date of Jan. 2, 2008, the training data was varied from 10% to 90%. After each data volume had been set, the network was trained and then tested on a 3-month prediction period, standardized as the next 60 predictions after end of training date. The results obtained for each setting are shown on Table 3.5 below. The raw data is show in **Appendix 9**.

Table 3.5: *MAPE and RSME for different Training Data Set Volumes*

Training Data	Training Dates	MAPE	RSME
10%	2-Jan-2008 – 30-Jun-2008	3.41%	8.61
20%	2-Jan-2008 – 31-Dec-2008	5.66%	13.25
30%	2-Jan-2008 – 3-Jul-2009	1.14%	2.33
40%	2-Jan-2008 – 6-Jan-2010	1.33%	2.88
50%	2-Jan-2008 – 5-Jul-2010	5.28%	16.93
60%	2-Jan-2008 – 31-Dec-2010	1.87%	5.76
70%	2-Jan-2008 – 1-Jul-2011	0.94%	2.89
80%	2-Jan-2008 – 30-Dec-2011	0.79%	1.84
90%	2-Jan-2008 – 2-Jul-2012	1.28%	3.99

The results are also shown graphically on Figure 3.5, Figure 3.6 and Figure 3.7 below.

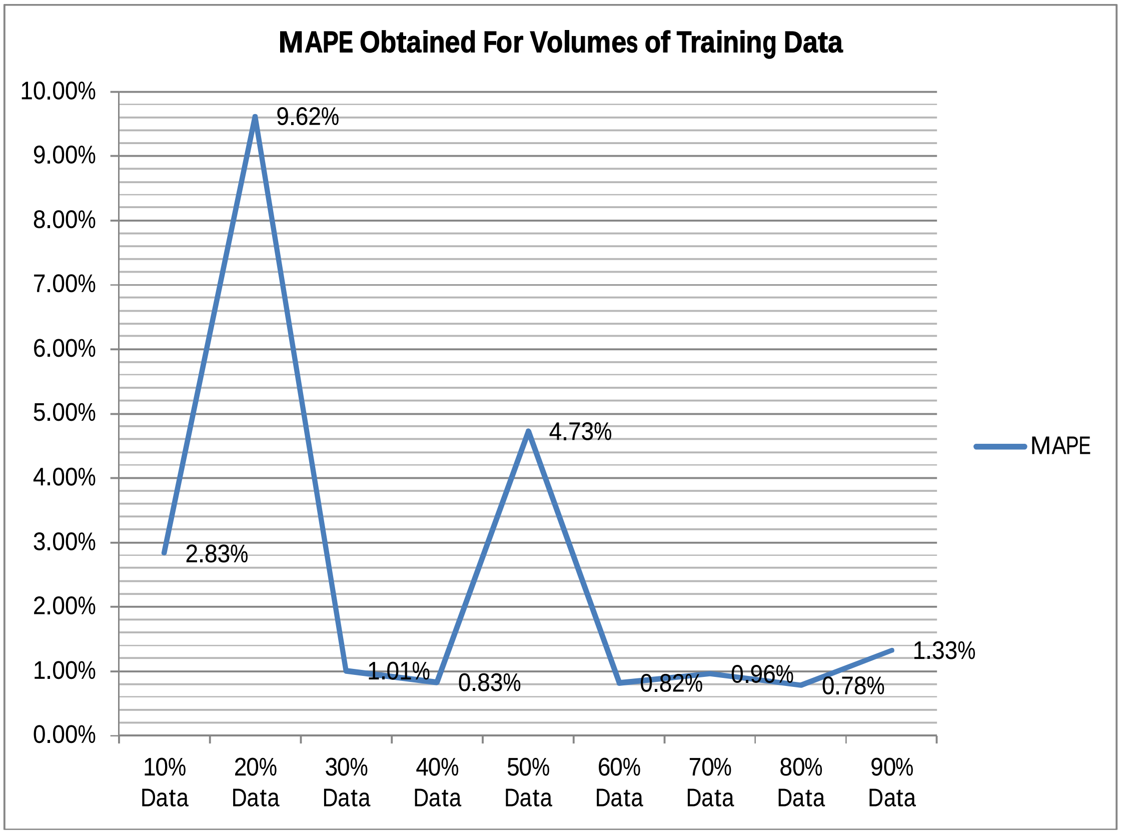


Figure 3.5 – MAPE for different volumes of Training data

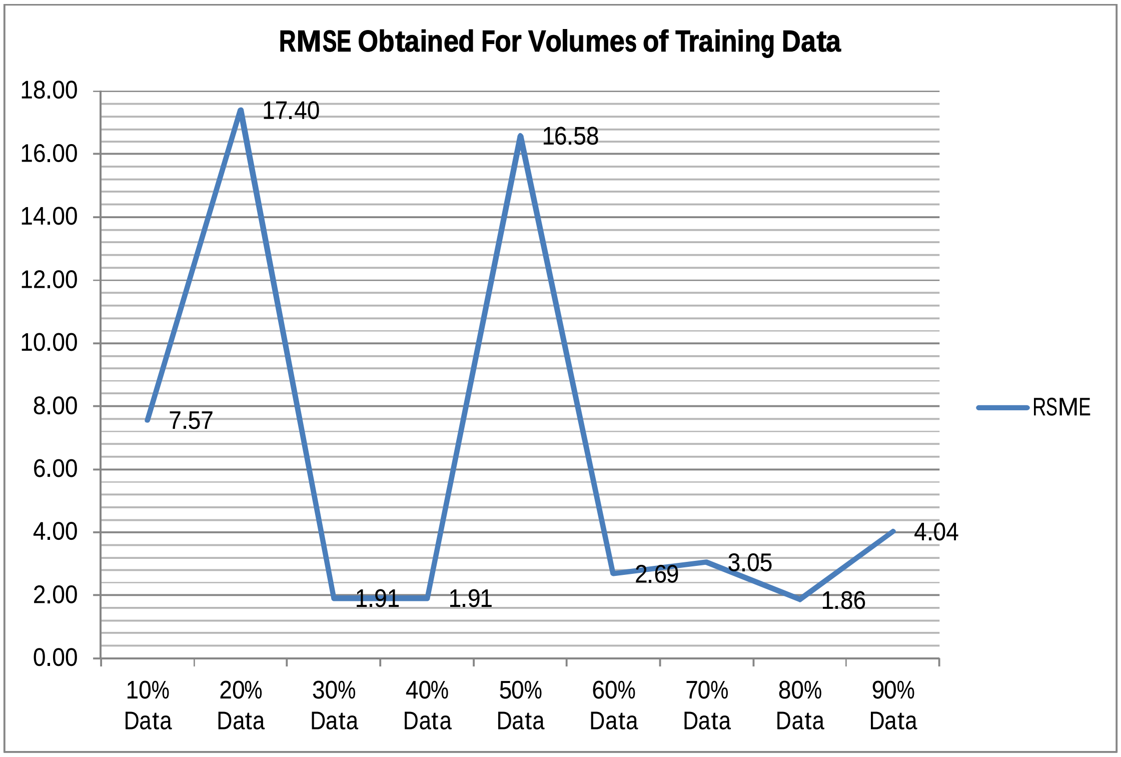


Figure 3.6 – RMSE for different volumes of Training data

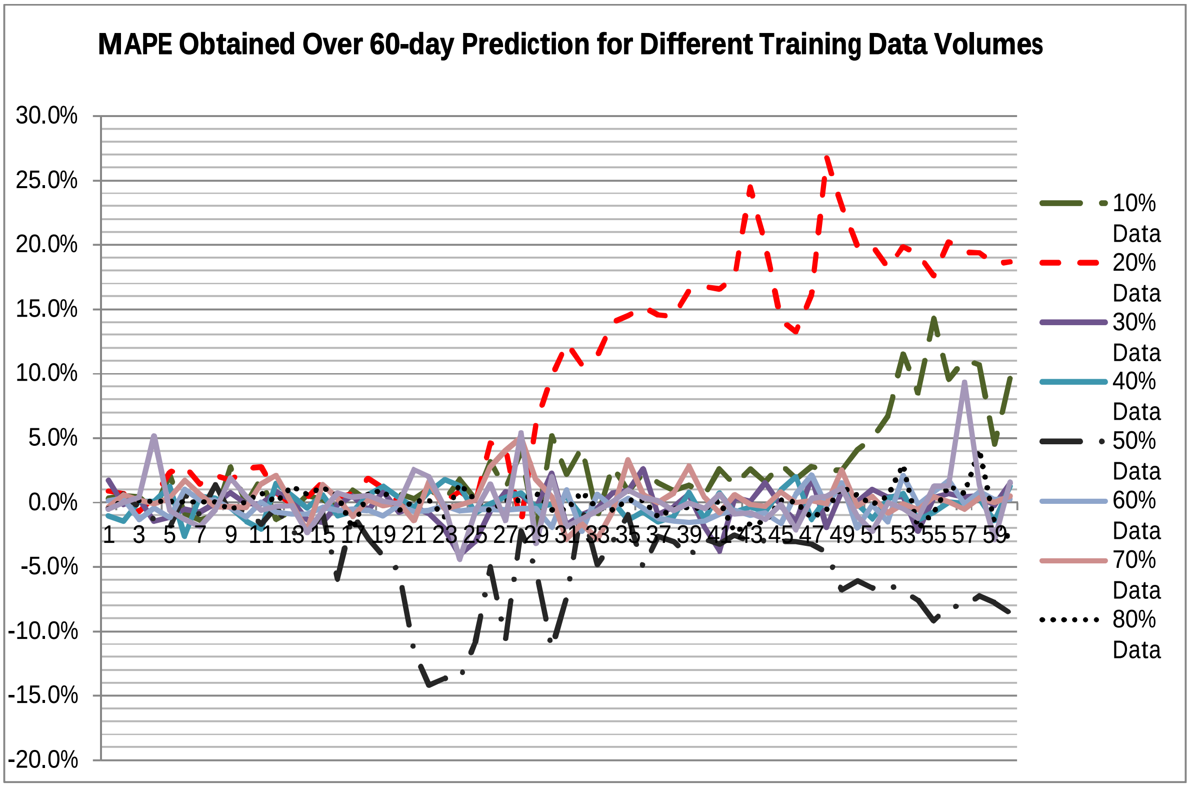


Figure 3.7 – MAPE for different volumes of Training data over 60-day prediction

The following observations were made – the MAPE was low for data volumes of 40% (0.83%), 60% (0.82%), 70% (0.96%) and 80% (0.78%). However, the lowest MAPE was obtained when the training data volume is 80%. This was therefore the most appropriate training data volume. Due to this observation, the optimal setting was therefore the use of 80% of the data for training (1,000 data rows) to enable 60-future predictions

3.3.2.3 Determining the Number of Training Repetitions

Using the tuned network, of configuration 5:21:21:1 with 80% of data for training, the effect of varying the number of training cycles was studied. This experiment was also based on Stock02, starting from one cycle and testing at every 10,000 cycles, upto 180,000 repetitions. A timer was used to record the duration of training for the corresponding training cycles and the corresponding network error. The results obtained for each setting are shown on the graph in figure 3.8 below. The raw data is show in **Appendix 10**.

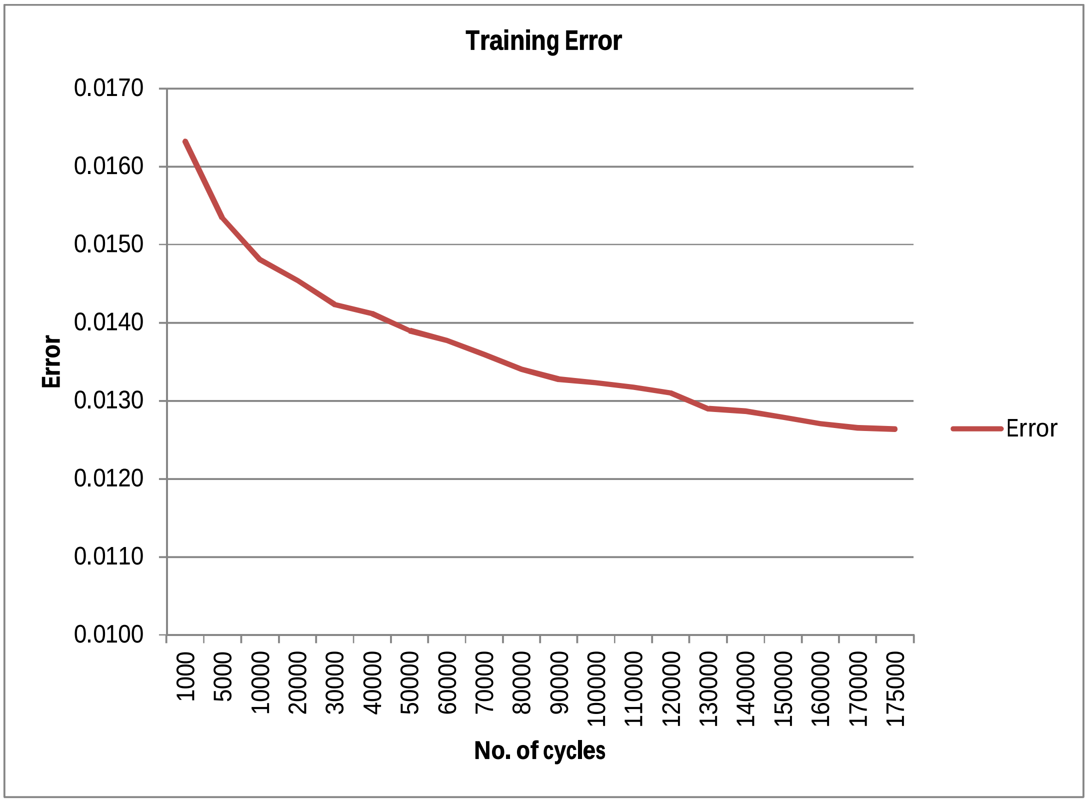


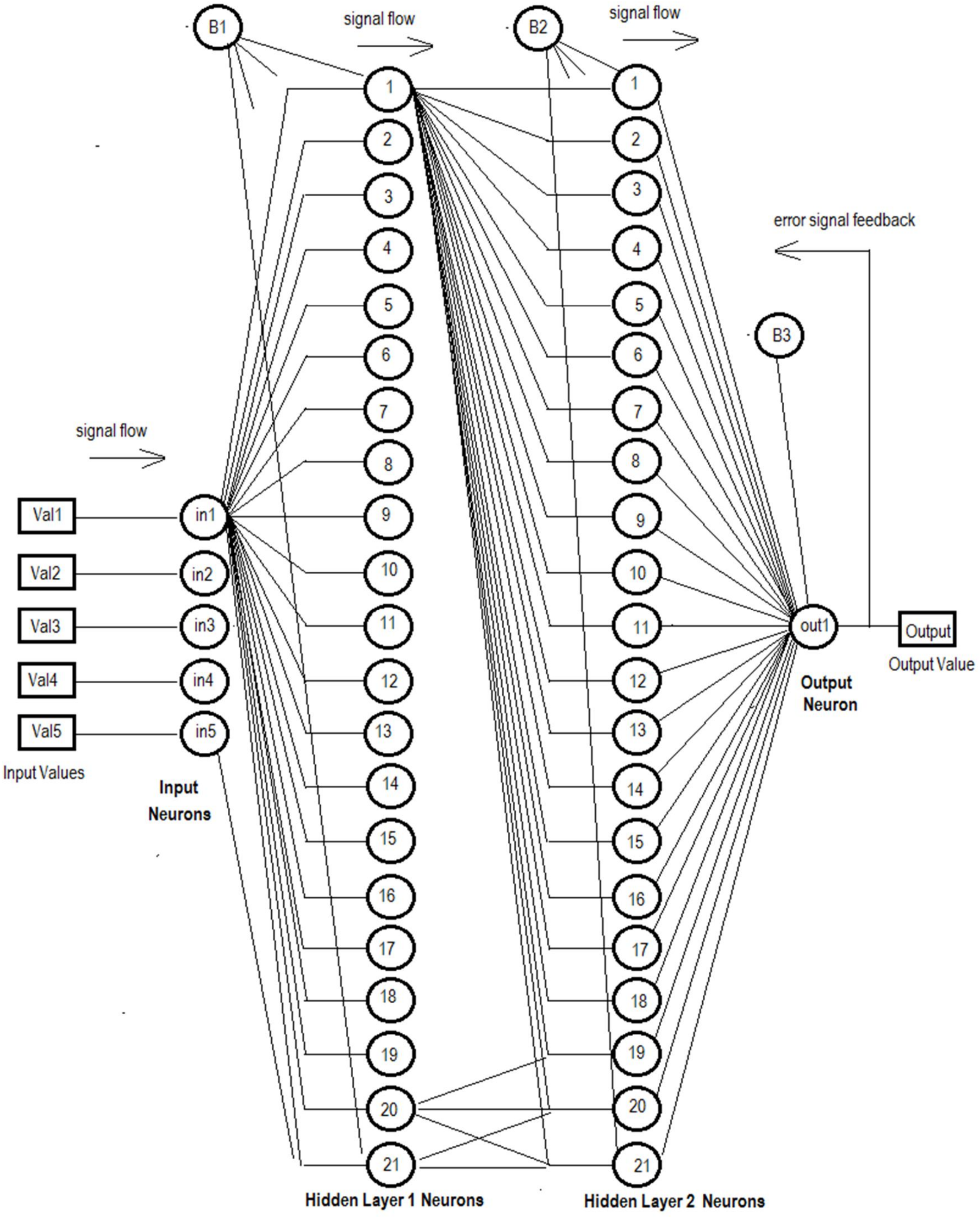
Figure 3.8 – Training error for different cycles of Training data

It was observations that the training error rates started high (1.6%) and reduced as the repetitions over the data was increased. It was also noted that training error tapered off after 130,000 training cycles, with an error rate of 1.30%. By the stop of training at 170,000 cycles, the error rate had only dropped to 1.26%.(2hr 24min). Any repetition after 130,000 training cycles was therefore determined to be acceptable. This took 1hour 40minutes of training. Experimentation also showed that after 12hours of training, the error was 0.01203.

3.3.3 The Final Proposed Model

The final model, as determine by experimentation was of configuration of 5:21:21:1, using 80% of the available data (Jan. 2, 2008 to Dec. 31, 2011) for training, with at least 130,000 repetitions during training (about 1hour 40min). The testing was to be done on a 3 month period i.e. Jan. 2, 2012 to Mar. 31, 2012.

The new developed model is shown in Figure 3.9 below.



For clarity, only typical neuron to neuron connections is shown. In the model, each neuron connects to the next layer e.g. Input Layer neuron1 (in1) connects to each Hidden Layer neuron, same to Input neuron 2 (in2) etc. Hidden Layer 1 neuron 1 connects to Hidden Layer 2 neuron 1 to 21, same to neuron 2 etc.

Figure 3.9 – New ANN model (Source: Author)

3.4 Designing and Developing the Prototype

3.4.1 Designing the Prototype

Designing the ANN prototype required the formulation of the suitable algorithms that implemented the desired goal. The prototype was based on the model of configuration 5:21:21:1, using 80% of the available data (Jan. 2, 2008 to Dec. 31, 2011) for training, with at least 130,000 repetitions. The ANN layout is a multilayer perceptron (input layer, hidden layers, output layer, each with a number of neurons), the network connectivity is Feedforward network (connectivity is between input neurons towards output neurons), weight adjustment is by error backpropagation (resilient) and training is by supervised learning. The design tools that were used were pseudocode and flowcharts.

The pseudocode for creating, training and testing an ANN prototype is shown below:

```
1. Create a network of neurons
   1.1 Set number of inputs = 5
   1.2 Create Input neurons = 5
   1.3 Set number of hidden layers = 2
   1.4 Set number of neurons per hidden layer = 21
   1.5 Create Hidden neurons = 21
   1.6 Set number of outputs = 1
   1.7 Create Output neurons
   1.8 Create Bias neurons = 1 for each hidden layer and 1 for the output (total 3)
   1.9 Set the bias weight = 1

2. Create activation function

3. Initialize the network by Generating random weights for each neuron

4. Prepare the input data file
   4.1 Normalize all data items to be in the range 0 - 1
   4.2 Open the input data file
   4.3 Load contents into memory as a 2D array
   4.4 Determine the size of data file i.e. no. of rows
```

```
5. Train the network
   Repeat
      Load the first data line
      Repeat
         Feed the data line forward through the network
         Calculate the sum of inputs, weights and activation for each layer upto the output neuron
         Obtain final output value (OV)
         Read the desired result (DR) from the data line
         Calculate activation error (AE) = DR – OV
         Change weights feeding output neuron in order to reduce its activation error based on the input data
         Do back propagation i.e. provide an error value back to each hidden neuron as proportional to contribution of AE
         Change weights associated with each hidden neuron, so that their error contribution to input data is reduced
         Load the next data line
      Until end of file
   Until AE is significantly small

6. Test the network
   Repeat
      Ask user for date of prediction
      Use trained network to calculate an output value
      Display the result
   Until user ends program
```

The design of the training algorithm is illustrated by the flowchart in Figure 3.10 below.

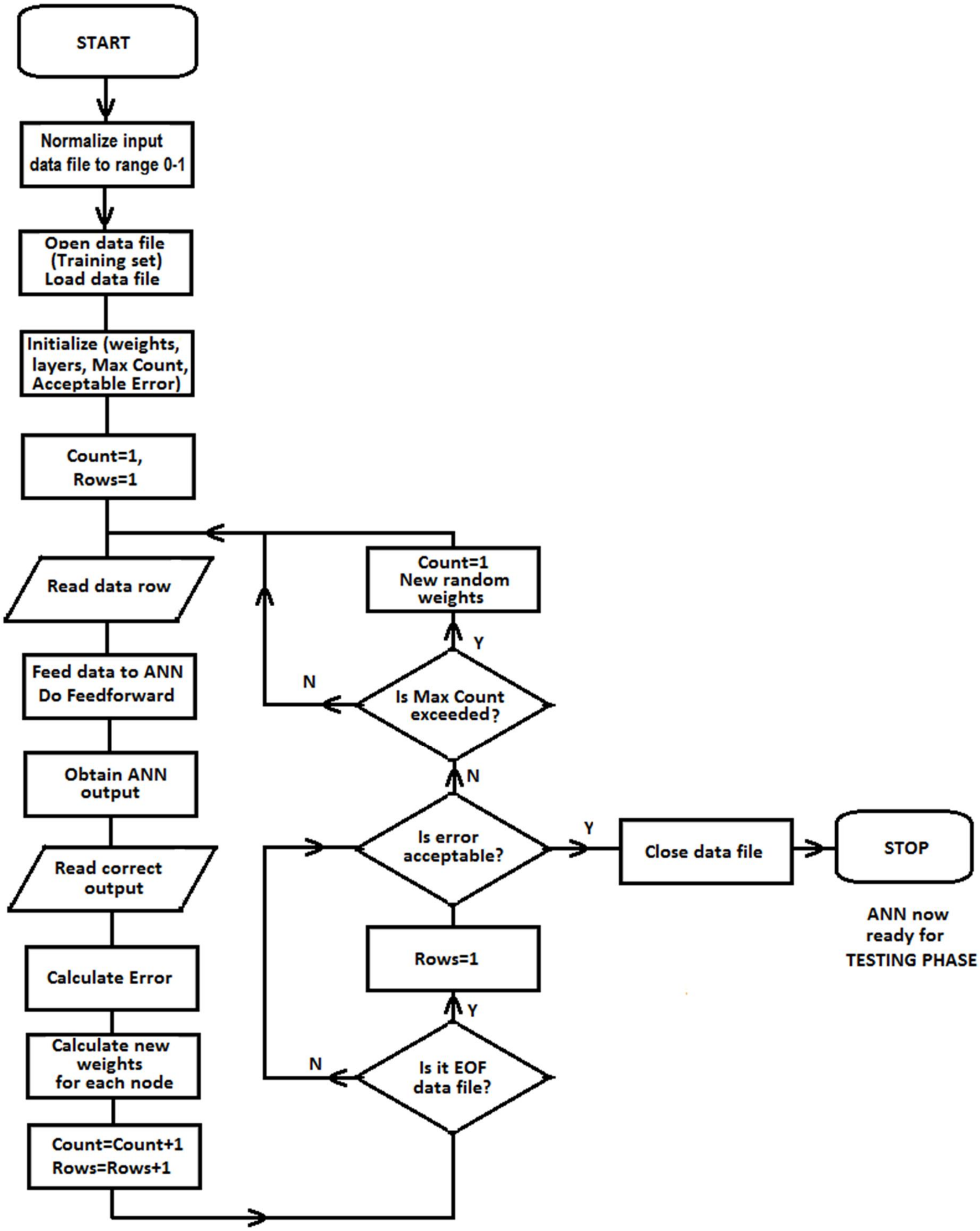

START
Normalize input data file to range 0-1
Open data file (Training set) Load data file
Initialize (weights, layers, Max Count, Acceptable Error)
Count=1, Rows=1
Read data row
Feed data to ANN Do Feedforward
Obtain ANN output
Read correct output
Calculate Error
Calculate new weights for each node
Count=Count+1 Rows=Rows+1
Is it EOF data file?
N
Y
Rows=1
Is error acceptable?
Y
Close data file
STOP
ANN now ready for TESTING PHASE
N
Is Max Count exceeded?
N
Y
Count=1 New random weights

Figure 3.10 – Flowchart for Training ANN (Source: Author)

The design of the testing algorithm is illustrated by the flowchart in Figure 3.11 below.

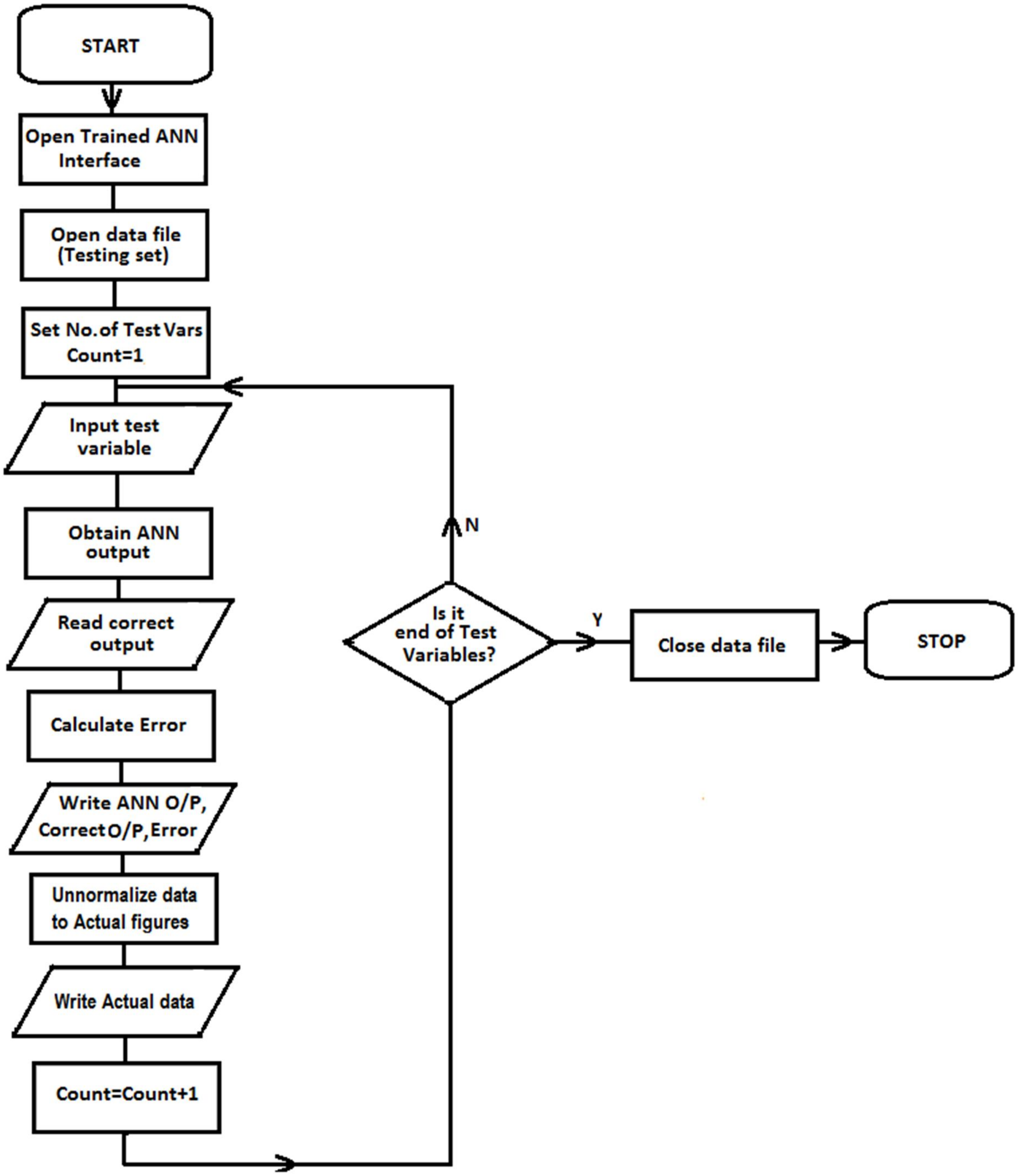

START
Open Trained ANN Interface
Open data file (Testing set)
Set No.of Test Vars Count=1
Input test variable
Obtain ANN output
Read correct output
Calculate Error
Write ANN O/P, Correct O/P, Error
Unnormalize data to Actual figures
Write Actual data
Count=Count+1
Is it end of Test Variables?
N
Y
Close data file
STOP

Figure 3.11 – Flowchart for Testing ANN (Source: Author)

3.4.2 Developing the Prototype

The following materials and equipment were used to enable the development of the prototype:

1. Desktop or laptop computer, with specifications: at least PIV, 1GB RAM, 1GB free disk space
2. Operating System on the computer – Windows 7
3. Programming language on the computer – Visual C# Express 2010
4. Stock market daily price movement data collected and formatted as Comma Separated Values (CSV) format, stored on the computer
5. ANN program developed using Visual C# programming language, installed on the computer
6. Open source program (Neuroph) downloaded, as availed by the developers (Neuroph, 2013) and installed on the computer
7. Open source program (Encog Workbench) downloaded, as availed by the developers (Heaton Research, 2013) and installed on the computer

Development of the prototype was done on a programming platform. This involved writing a computer program using C# programming language. This programming language was chosen because of the ease of designing user interface forms using the inherent Windows-based design tools. Forms improve the ease of interaction and hence ease of use of the system by the end user for the research experiments. The language is also an object-oriented language, which is useful in implementing the algorithm which shall mainly rely on reusable objects. Most components of the ANN engine, especially the neurons are created based on dynamic reusable objects. Finally, C# provides utilities for seamless data I/O with CSV files, which was the main source data file format for this project. This design described a lot of I/O data operations, especially the continuous reading of lines of raw data into the ANN engine.

Available public domain frameworks have already implemented intrinsic NN structures, leaving the programming task to the design of inputs, outputs and activation, without having to code the backpropagation details. Such ANN frameworks include the Encog framework, an open source collection, which is popular with C# programmers (Heaton Research, 2013). The framework is capable of introducing data normalization, if the inputs are of varied magnitudes. This ensures consistency during the training phase. The project prototype was therefore developed using C# based on the design described in 3.4.1. The main components of the C# program in a development environment are shown in **Appendix 17**.

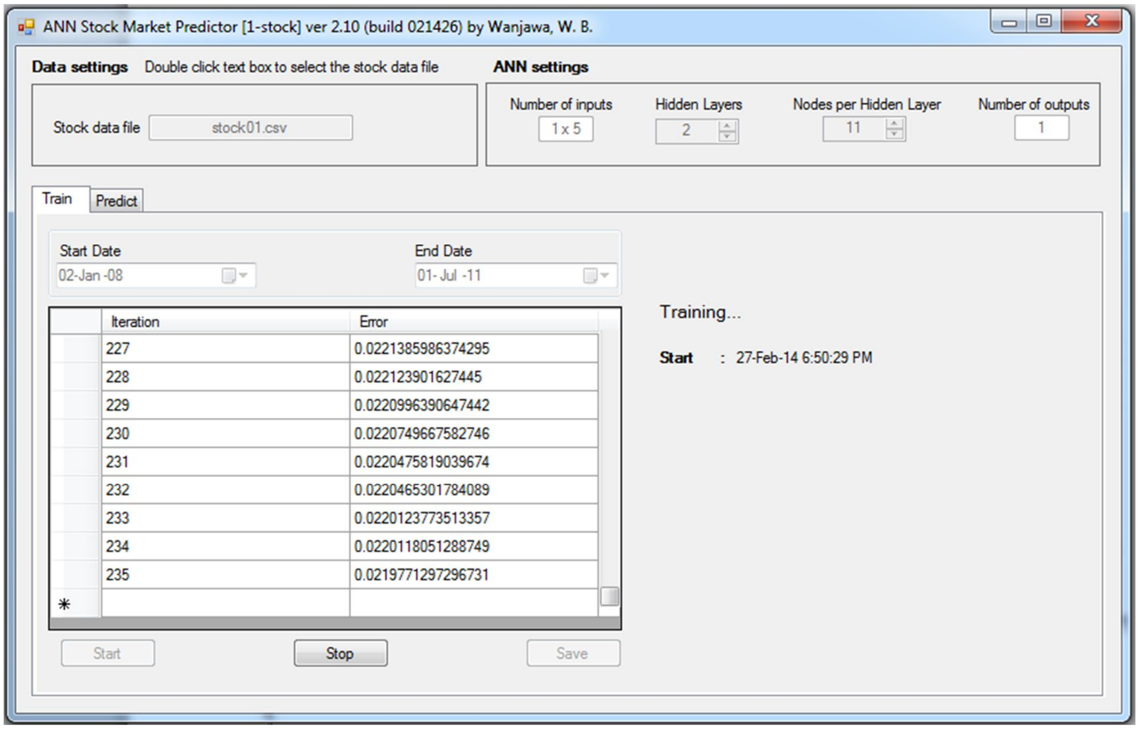
All files with extension .cs are C# program code files that were either auto-generated or programmed. **Appendix 17** also shows the screenshot of the development environment. The main form is ***WinStockMarketPredictor.cs***, which is shown open on that screenshot. The main program is contained on the source code file called ***MainProgram.cs*** and the code running behind the form called ***WinStockMarketPredictor.cs***. The source code of these two programs is shown in **Appendix 18**. The final product was a C# program with an interface that allows for a Training action (Train tab) and a Testing action (Predict tab).

The input layer shall consist of a burst of five continuous stock prices, with the aim of predicting the sixth price. The training aimed at enabling the network to undertake a sixth prediction based on the previous five. The input data was moved through a sliding window of five stocks upto the end of file.

The ***Train tab*** allows the following actions:

1. Loading of the individual data files in csv format. The default data files are already kept on the ‘bin’ folder, but it is possible to pick any other file from any other storage location. The only requirement is that the file must be csv format, with only 2 data columns i.e. date and stock price for that date.
2. Choosing the training start and end dates (typing or using calendar date chooser)
3. Setting up ANN parameters i.e. no. of hidden layers, number of neurons per hidden layer (the default is set to 2 and 21 respectively)
4. Start button – to start training
5. Stop button – to stop training at any time
6. Save button – to save the network weights so that the particular network can be loaded later, and used for prediction.

The prototype with the ‘Train’ tab active is shown on Figure 3.12 below.



ANN Stock Market Predictor [1-stock] ver 2.10 (build 021426) by Wanjawa, W. B.

Data settings Double click text box to select the stock data file

Stock data file: stock01.csv

ANN settings

Number of inputs: 1 x 5

Hidden Layers: 2

Nodes per Hidden Layer: 11

Number of outputs: 1

Start Date: 02-Jan -08

End Date: 01- Jul -11

	Iteration	Error
	227	0.0221385986374295
	228	0.022123901627445
	229	0.0220996390647442
	230	0.0220749667582746
	231	0.0220475819039674
	232	0.0220465301784089
	233	0.0220123773513357
	234	0.0220118051288749
	235	0.0219771297296731
*		

Training...

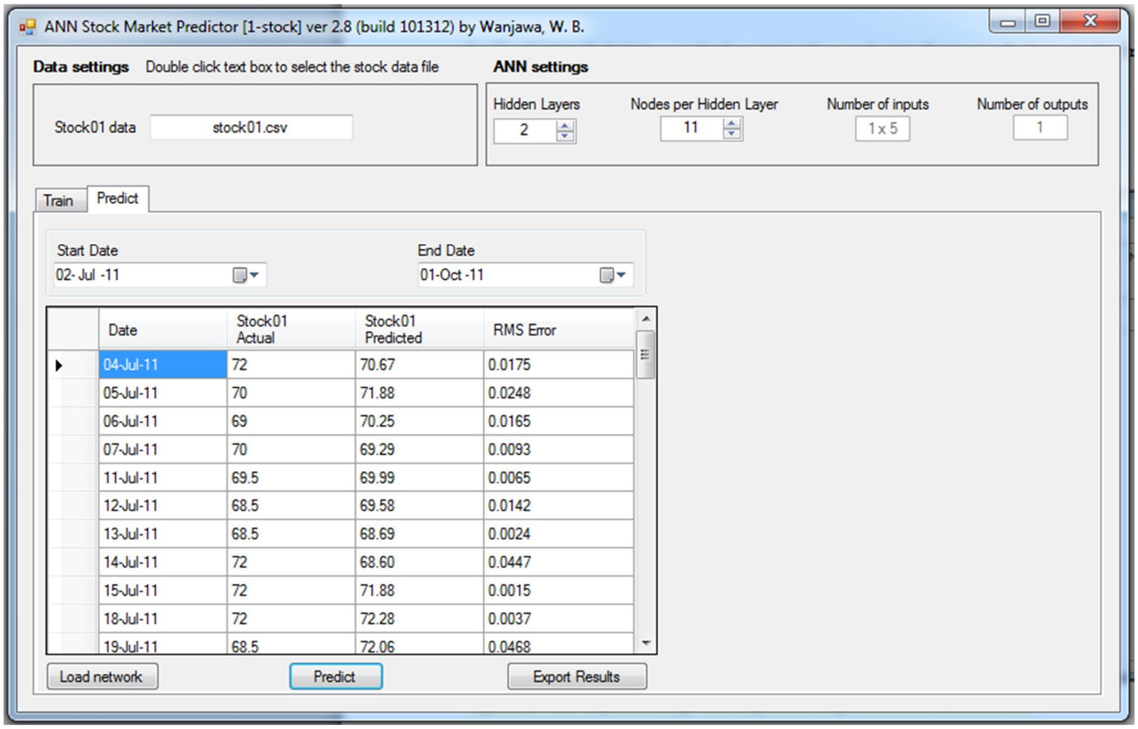
Start : 27-Feb-14 6:50:29 PM

Figure 3.12 – Developed Prototype Showing the Training Interface (Source: Author)

The ***Predict tab*** allows the following actions:

1. Choosing the testing start and end dates (typing or using calendar date chooser)
2. Predict button – to invoke the predict action and populate the blank table with values
3. Load network button – to load a previously saved network (see no. 6 above, under Train button), in readiness for prediction
4. Export results button – to save the results on screen to a file in csv format.

The prototype with the ‘Predict’ tab active is shown on Figure 3.13 below.



ANN Stock Market Predictor [1-stock] ver 2.8 (build 101312) by Wanjawa, W. B.

Data settings Double click text box to select the stock data file

Stock01 data: stock01.csv

ANN settings

Hidden Layers: 2

Nodes per Hidden Layer: 11

Number of inputs: 1 x 5

Number of outputs: 1

Start Date: 02- Jul -11

End Date: 01-Oct -11

Date	Stock01 Actual	Stock01 Predicted	RMS Error
04-Jul-11	72	70.67	0.0175
05-Jul-11	70	71.88	0.0248
06-Jul-11	69	70.25	0.0165
07-Jul-11	70	69.29	0.0093
11-Jul-11	69.5	69.99	0.0065
12-Jul-11	68.5	69.58	0.0142
13-Jul-11	68.5	68.69	0.0024
14-Jul-11	72	68.60	0.0447
15-Jul-11	72	71.88	0.0015
18-Jul-11	72	72.28	0.0037
19-Jul-11	68.5	72.06	0.0468

Figure 3.13 – Developed Prototype Showing the Testing Interface (Source: Author)

The developed prototype was tested to confirm full functionality using tests and utilities such as debugging, functional tests, module tests and watching of variable behavior over program runs. All tests confirmed that the prototype was functional and ready for use.

3.5 Chapter Summary

Chapter three provided the details of research design, choices to be made and justification for such choices. The chapter also formulated a new model then designed the project prototype based on this model. The next chapter provides the results of the evaluation of the prototype developed in this chapter.

CHAPTER 4
EVALUATION OF THE ANN MODEL

4.1 Evaluating the Prototype

Evaluation of the ANN model was done using experimental methods. The tests were done using the developed prototype, based on a configuration of 5:21:21:1,with 80% data for training (Jan. 2, 2008 to Dec. 31, 2011) and a 3-month prediction period (Jan. 2, 2012 to Mar. 31, 2012), with number of repetitions was set to at least 130,000 repetitions. Due to time constraint, the research did final tests on only three of the six qualifying stocks. The price span (lowest to highest closing prices over the period) for the six test stocks were determined as: Stock01 (400%), Stock02 (135%), Stock03 (289%), Stock04 (102%), Stock05 (214%) and Stock06 (134%). Based on these, the most suitable candidates were those with extreme price spans i.e. Stock01 and Stock04. The third candidate was the stock with a median span, in this case, either Stock02 or Stock06. Stock02 was preferred since it had been used in the baseline tests. This selection of three stocks i.e. Stock01, Stock02 and Stock04 enabled the research to test all extreme price jumps on the test data.

4.2 Testing the Prototype on NSE data

This was done using the developed prototype by invoking a training operation, followed by a testing operation.

4.2.1 Results

The results obtained using the prototype on the NSE data for the best performance i.e. Stock02 (Standard Bank) is shown in Tables 4.1 below. The individual results for all the three test stocks are shown in **Appendix 11**. The results table shows the particular date of trade on column one and the corresponding actual closing stock value declared by the NSE analysis report on that date on column two. The third column provided the value that was predicted by the prototype, while the last column shows the error between the actual and the predicted values.

Table 4.1: *Stock02 (Standard Bank) Results– using Developed Prototype on NSE data*

	Stock 02 – Standard Bank		
Test date	Actual	Predicted	Error (%)
03-Jan-12	161.00	160.20	-0.5%
04-Jan-12	161.00	161.13	0.1%
05-Jan-12	161.00	161.23	0.1%
06-Jan-12	161.00	161.15	0.1%
09-Jan-12	161.00	161.12	0.1%
10-Jan-12	161.00	161.10	0.1%
11-Jan-12	161.00	161.10	0.1%
12-Jan-12	161.00	161.10	0.1%
13-Jan-12	162.00	161.10	-0.6%
16-Jan-12	162.00	162.10	0.1%
17-Jan-12	161.00	162.23	0.8%
18-Jan-12	161.00	161.18	0.1%
19-Jan-12	159.00	161.01	1.3%
20-Jan-12	158.00	159.04	0.7%
23-Jan-12	156.00	157.80	1.2%
24-Jan-12	155.00	155.80	0.5%
25-Jan-12	158.00	154.66	-2.1%
26-Jan-12	157.00	157.78	0.5%
27-Jan-12	156.00	157.28	0.8%
30-Jan-12	157.00	156.03	-0.6%
31-Jan-12	157.00	156.92	-0.1%
01-Feb-12	157.00	157.06	0.0%
02-Feb-12	159.00	157.04	-1.2%
03-Feb-12	157.00	159.04	1.3%
06-Feb-12	157.00	157.30	0.2%
07-Feb-12	158.00	156.89	-0.7%
08-Feb-12	158.00	157.96	0.0%
09-Feb-12	158.00	158.11	0.1%
10-Feb-12	157.00	158.09	0.7%
13-Feb-12	158.00	157.07	-0.6%
14-Feb-12	159.00	157.91	-0.7%
15-Feb-12	158.00	159.10	0.7%
16-Feb-12	159.00	158.20	-0.5%
17-Feb-12	160.00	159.00	-0.6%
20-Feb-12	160.00	160.14	0.1%
21-Feb-12	160.00	160.23	0.1%
22-Feb-12	162.00	160.16	-1.1%
23-Feb-12	163.00	162.09	-0.6%
24-Feb-12	164.00	163.32	-0.4%
27-Feb-12	165.00	164.31	-0.4%
28-Feb-12	165.00	165.28	0.2%
29-Feb-12	169.00	165.31	-2.2%
01-Mar-12	172.00	169.02	-1.7%
02-Mar-12	175.00	172.20	-1.6%
05-Mar-12	175.00	175.08	0.0%
06-Mar-12	175.00	175.27	0.2%
07-Mar-12	177.00	175.11	-1.1%
08-Mar-12	178.00	176.84	-0.7%
09-Mar-12	176.00	177.97	1.1%
12-Mar-12	175.00	176.07	0.6%
13-Mar-12	175.00	174.71	-0.2%
14-Mar-12	174.00	174.57	0.3%
15-Mar-12	169.00	173.78	2.8%
16-Mar-12	172.00	168.92	-1.8%
19-Mar-12	173.00	171.45	-0.9%
20-Mar-12	171.00	172.98	1.2%
21-Mar-12	170.00	171.30	0.8%
22-Mar-12	163.00	169.95	4.3%
23-Mar-12	165.00	163.02	-1.2%
26-Mar-12	169.00	164.45	-2.7%
27-Mar-12	169.00	169.02	0.0%
28-Mar-12	172.00	169.60	-1.4%
29-Mar-12	171.00	172.05	0.6%
30-Mar-12	174.00	171.38	-1.5%
MAPE			**0.77%**

The results obtained can be represented by graphs as shown in Figure 4.1 below:

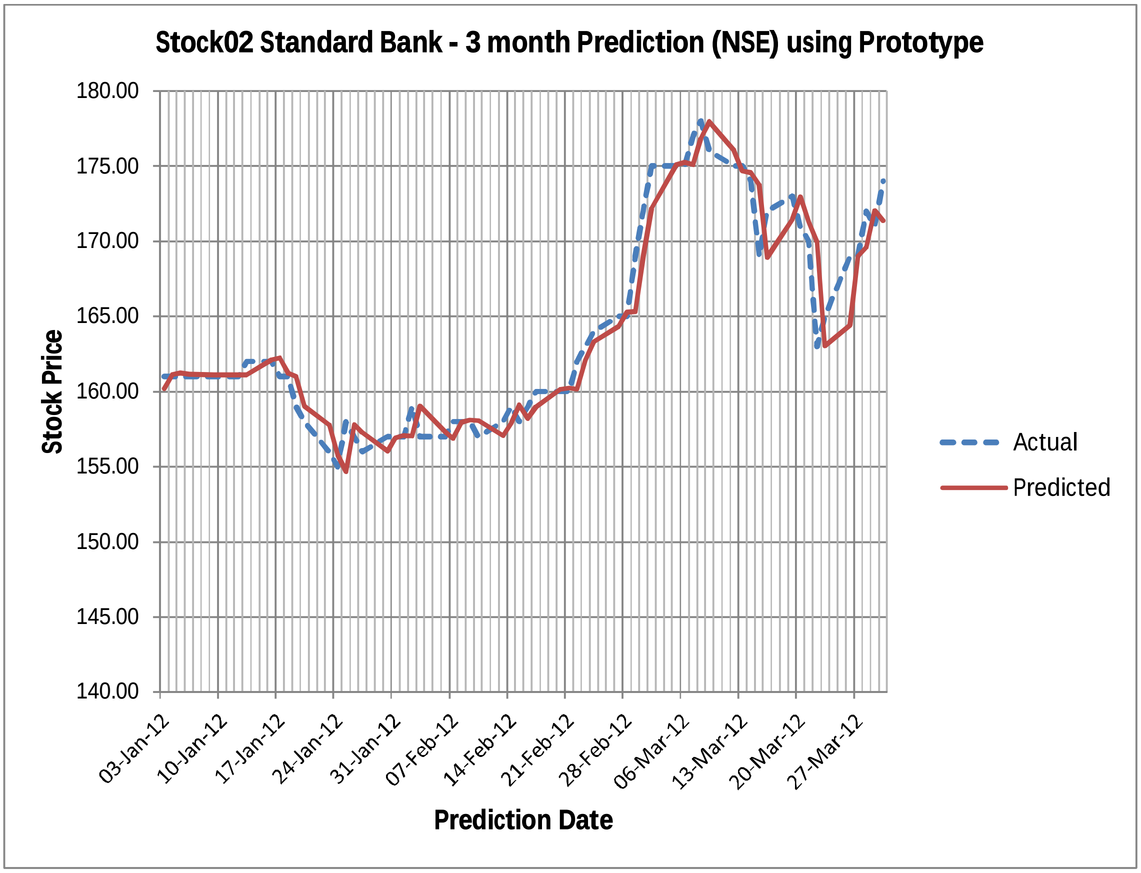


Figure 4.1 – Stock02 Results (Using ANN Prototype) on NSE data – Comparing Actual against Predicted

The comparison of MAPE error for the prototype when applied to the three NSE test stocks is shown on the graph in Figure 4.2 below.

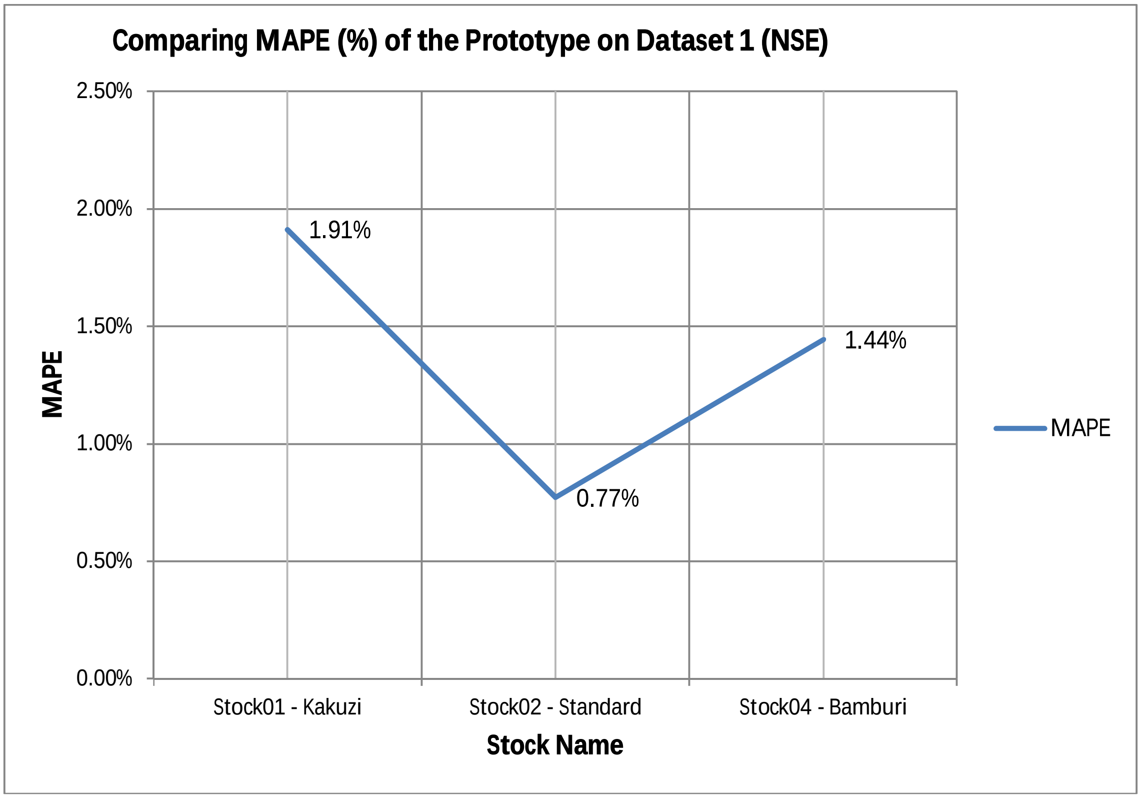


Figure 4.2 – Comparing MAPE of the Prototype on NSE data

4.2.2 Observations and Analysis

It was observed that the error (MAPE) when using the prototype to predict the actuals was 0.77%. When comparing the performance of the protype on the three test stocks, the MAPE was 1.91% for Stock 01 (Kakuzi), 0.77% for Stock02 (Standard bank) and 1.44% for Stock04 (Bamburi). The predicted prices tended to be a little higher than the actual i.e. 36 instances out of 64 observations. It was also noted that none of the predictions was the same as the actual, with 28 predictions being a little lower than the actual on the respective dates. The highest variance on any day of trade in the 64-day period was 4.3% (22-Mar-2012)

4.2.3 Discussions

The lowest MAPE was noted for Stock02 (Standard bank) at 0.77%. Since MAPE is a measure of accuracy in prediction, lower values are usually desirable. This low value therefore meant that the prototype was able to predict prices that were very close to the actual final published prices. Of the three stocks tested, the MAPE for Stock02 was the lowest amongst them. That would mean that the trained network for Stock02 was quite fit for the task. The prediction trend was that of the actual trades for all observations, in terms of up and down movements over time. That meant that is this tool was used for trending only, then it would point to the right up or down movements as per the actual trends on the respective days. ANNs are usually suitable for their task based on their accuracy. However, no trained ANN is known to be ideal, and exhibits some error, whose decree of acceptability varies. This phenomenon is observed on the varied MAPE rates for the three test stocks.

4.3 Comparative Evaluation Using Other Tools

To test the effectiveness of the prototype that has been developed in this project, the same data was used to test two other open source ANN frameworks – Neuroph and Encog Workbench. Each of these tools provides an interface for setting the ANN parameters, including a test data file, training the network and finally testing the network.

4.3.1 Neuroph

Neuroph in an open source ANN framework, based on Java. It can be used to test various ANN configurations. This project used Neuroph version 2.8 (201306052037). Neuroph program requires Java to be preinstalled on the testing system. This test was done using Java Runtime Environment (JRE) version 1.7.0_45. A typical design screen showing a network structure under development in Neuroph is shown in Figure 4.3 below. This tool enables data input, training and testing operations to be done, after which results can be obtained ready for analysis. The procedure for using the Neuroph tool is shown in **Appendix 15**.

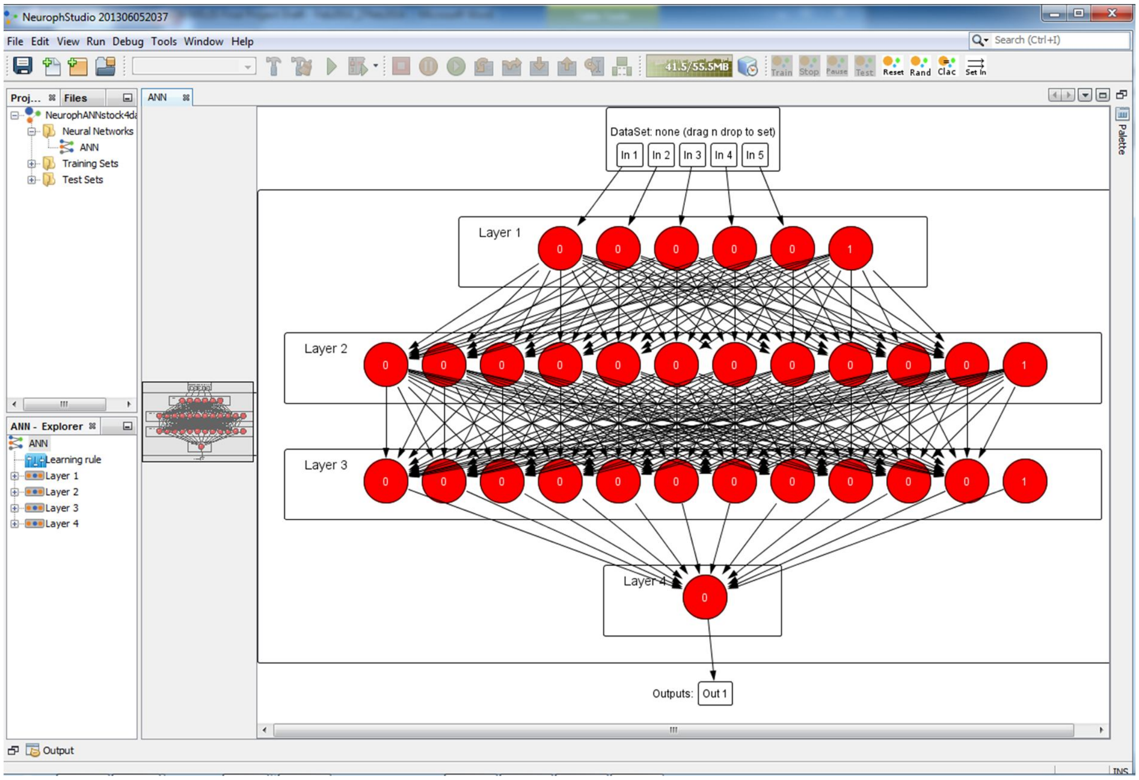
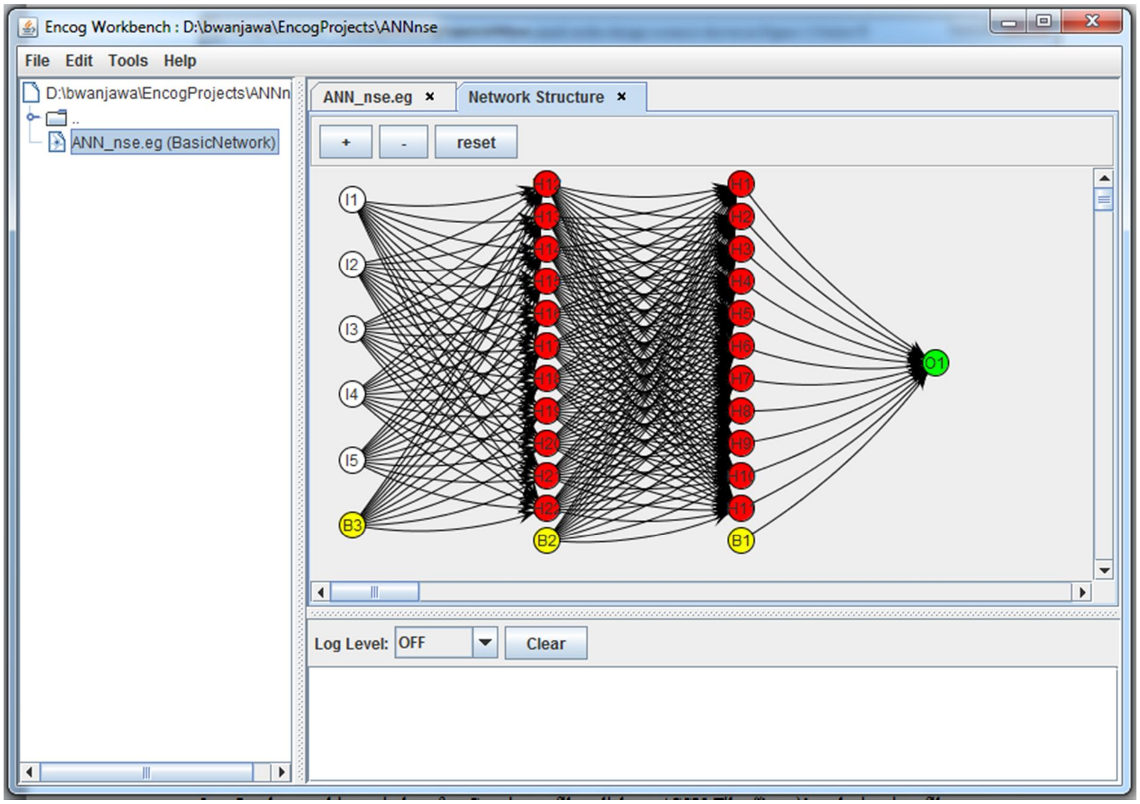


Figure 4.3 – Screenshot of ANN structure on Neuroph screen (Source: Author)

4.3.2 Encog Workbench

Encog Workbench is also an open source ANN framework, based on Java. It comes preset with various ANN parameters that can be selected for experimentation. This project used Encog Workbench version 3.0.1. The program is directly executable without need to install, but it needs Java to be preinstalled on the test machine. This test was done using Java Runtime Environment (JRE) version 1.7.0_45. A typical design screen showing a network structure under development in Encog Workbench is shown in Figure 4.4 below. This tool also enables data input, training and testing operations to be done, after which results can be obtained ready for analysis. The procedure for using the Encog Workbench tool is shown in **Appendix 16**.


Encog Workbench : D:\bwanjawa\EncogProjects\ANNnse
File
Edit
Tools
Help
D:\bwanjawa\EncogProjects\ANNn
..
ANN_nse.eg (BasicNetwork)
ANN_nse.eg ×
Network Structure ×
+
-
reset
I1
I2
I3
I4
I5
B3
B2
H1
H2
H3
H4
H5
H6
H7
H8
H9
H10
H11
B1
O1
Clear

Log Level: OFF

Figure 4.4 – Screenshot of ANN structure on Encog Workbench screen (Source: Author)

4.3.3 Results

The results obtained using Encog Workbench and Neuroph as compared to those obtained using the prototype, based on NSE data for period 2008 to 2012 is shown in Tables 4.2 below for Stock02 (Standard Bank) was the best performing stock based on MAPE. The individual results for all the three test stocks are shown in **Appendix 12** (Neuroph) and **Appendix 13** (Encog Workbench). The results table shows the particular date of trade on column one and the corresponding actual closing stock value declared by the NSE analysis report on that date. The next three sections on the table provided the value that was predicted by the prototype (column 3) and the error (%) between the predicted and the actual (column 4). The results when using Encog (predicted and calculated error) are shown in the second section of the table, while the final section shows the results when using Neuroph i.e. the predicted value and the calculated error compared to the actual value.

Table 4.2: *Stock02 (Standard Bank) Results– Comparing Developed Prototype with Other Tools (NSE)*

Test date	Actual	Prototype		Encog		Neuroph	
		Predicted	Error (%)	Predicted	Error (%)	Predicted	Error (%)
03-Jan-12	161.00	160.20	-0.5%	160.62	-0.2%	160.26	-0.5%
04-Jan-12	161.00	161.13	0.1%	161.57	0.4%	161.06	0.0%
05-Jan-12	161.00	161.23	0.1%	161.60	0.4%	161.30	0.2%
06-Jan-12	161.00	161.15	0.1%	161.49	0.3%	161.14	0.1%
09-Jan-12	161.00	161.12	0.1%	161.50	0.3%	161.14	0.1%
10-Jan-12	161.00	161.10	0.1%	161.50	0.3%	160.97	0.0%
11-Jan-12	161.00	161.10	0.1%	161.50	0.3%	160.97	0.0%
12-Jan-12	161.00	161.10	0.1%	161.50	0.3%	160.97	0.0%
13-Jan-12	162.00	161.10	-0.6%	161.50	-0.3%	160.97	-0.6%
16-Jan-12	162.00	162.10	0.1%	162.53	0.3%	161.74	-0.2%
17-Jan-12	161.00	162.23	0.8%	162.56	1.0%	161.97	0.6%
18-Jan-12	161.00	161.18	0.1%	161.42	0.3%	161.23	0.1%
19-Jan-12	159.00	161.01	1.3%	161.40	1.5%	161.00	1.3%
20-Jan-12	158.00	159.04	0.7%	159.45	0.9%	159.28	0.8%
23-Jan-12	156.00	157.80	1.2%	158.35	1.5%	158.03	1.3%
24-Jan-12	155.00	155.80	0.5%	156.41	0.9%	156.27	0.8%
25-Jan-12	158.00	154.66	-2.1%	155.38	-1.7%	154.80	-2.0%
26-Jan-12	157.00	157.78	0.5%	158.72	1.1%	157.50	0.3%
27-Jan-12	156.00	157.28	0.8%	157.74	1.1%	157.55	1.0%
30-Jan-12	157.00	156.03	-0.6%	156.14	-0.5%	156.91	-0.1%
31-Jan-12	157.00	156.92	-0.1%	157.40	0.3%	157.68	0.4%
01-Feb-12	157.00	157.06	0.0%	157.63	0.4%	157.34	0.2%
02-Feb-12	159.00	157.04	-1.2%	157.45	-1.0%	157.55	-0.9%
03-Feb-12	157.00	159.04	1.3%	159.64	1.7%	159.28	1.5%
06-Feb-12	157.00	157.30	0.2%	157.50	0.3%	158.03	0.7%
07-Feb-12	158.00	156.89	-0.7%	157.22	-0.5%	157.60	-0.3%
08-Feb-12	158.00	157.96	0.0%	158.60	0.4%	158.36	0.2%
09-Feb-12	158.00	158.11	0.1%	158.64	0.4%	158.22	0.1%
10-Feb-12	157.00	158.09	0.7%	158.46	0.9%	158.60	1.0%
13-Feb-12	158.00	157.07	-0.6%	157.38	-0.4%	157.82	-0.1%
14-Feb-12	159.00	157.91	-0.7%	158.49	-0.3%	158.20	-0.5%
15-Feb-12	158.00	159.10	0.7%	159.71	1.1%	159.18	0.7%
16-Feb-12	159.00	158.20	-0.5%	158.51	-0.3%	158.65	-0.2%
17-Feb-12	160.00	159.00	-0.6%	159.45	-0.3%	159.39	-0.4%
20-Feb-12	160.00	160.14	0.1%	160.69	0.4%	160.20	0.1%
21-Feb-12	160.00	160.23	0.1%	160.61	0.4%	160.26	0.2%
22-Feb-12	162.00	160.16	-1.1%	160.49	-0.9%	160.47	-0.9%
23-Feb-12	163.00	162.09	-0.6%	162.61	-0.2%	161.83	-0.7%
24-Feb-12	164.00	163.32	-0.4%	163.70	-0.2%	162.91	-0.7%
27-Feb-12	165.00	164.31	-0.4%	164.55	-0.3%	164.01	-0.6%
28-Feb-12	165.00	165.28	0.2%	165.47	0.3%	165.15	0.1%
29-Feb-12	169.00	165.31	-2.2%	165.34	-2.2%	165.06	-2.3%
01-Mar-12	172.00	169.02	-1.7%	169.03	-1.7%	168.54	-2.0%
02-Mar-12	175.00	172.20	-1.6%	171.86	-1.8%	172.53	-1.4%
05-Mar-12	175.00	175.08	0.0%	174.43	-0.3%	176.45	0.8%
06-Mar-12	175.00	175.27	0.2%	174.14	-0.5%	177.50	1.4%
07-Mar-12	177.00	175.11	-1.1%	173.43	-2.0%	176.67	-0.2%
08-Mar-12	178.00	176.84	-0.7%	174.79	-1.8%	178.07	0.0%
09-Mar-12	176.00	177.97	1.1%	175.57	-0.2%	179.00	1.7%
12-Mar-12	175.00	176.07	0.6%	173.70	-0.7%	177.31	1.3%
13-Mar-12	175.00	174.71	-0.2%	172.59	-1.4%	175.71	0.4%
14-Mar-12	174.00	174.57	0.3%	172.48	-0.9%	174.88	0.5%
15-Mar-12	169.00	173.78	2.8%	171.74	1.6%	173.56	2.7%
16-Mar-12	172.00	168.92	-1.8%	167.73	-2.5%	168.66	-1.9%
19-Mar-12	173.00	171.45	-0.9%	170.12	-1.7%	170.34	-1.5%
20-Mar-12	171.00	172.98	1.2%	171.40	0.2%	172.12	0.7%
21-Mar-12	170.00	171.30	0.8%	169.86	-0.1%	170.65	0.4%
22-Mar-12	163.00	169.95	4.3%	168.99	3.7%	170.24	4.4%
23-Mar-12	165.00	163.02	-1.2%	162.77	-1.3%	162.98	-1.2%
26-Mar-12	169.00	164.45	-2.7%	164.47	-2.7%	162.71	-3.7%
27-Mar-12	169.00	169.02	0.0%	168.56	-0.3%	166.84	-1.3%
28-Mar-12	172.00	169.60	-1.4%	168.87	-1.8%	168.16	-2.2%
29-Mar-12	171.00	172.05	0.6%	171.68	0.4%	172.87	1.1%
30-Mar-12	174.00	171.38	-1.5%	170.55	-2.0%	172.34	-1.0%
MAPE			**0.77%**		**0.88%**		**0.86%**
RMSE			**1.83**		**1.94**		**2.85**

The results obtained are represented by graphs as shown on Figure 4.5 below:

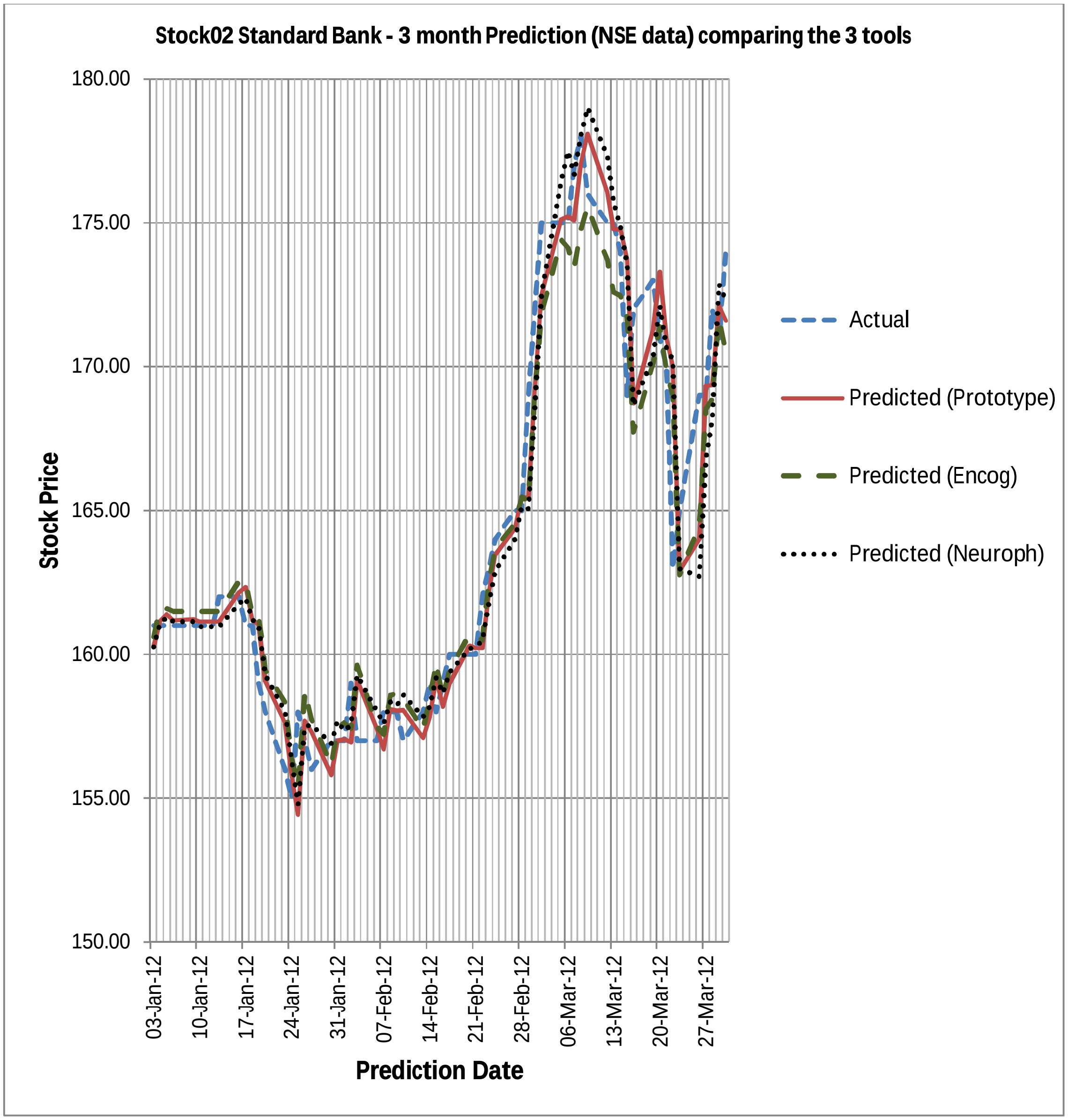


Figure 4.5 – Stock02 (Standard Bank) Results (Comparison of the 3 tools) on NSE data – Comparing Actual against Predicted

The comparison of MAPE and RMSE for three tools (prototype, Neuroph and Encog) on the three test stocks i.e. Stock01 (Kakuzi), Stock02 (Standard bank) and Stock04 (Bamburi) are shown on the graph in Figure 4.6 and Figure 4.7 below.

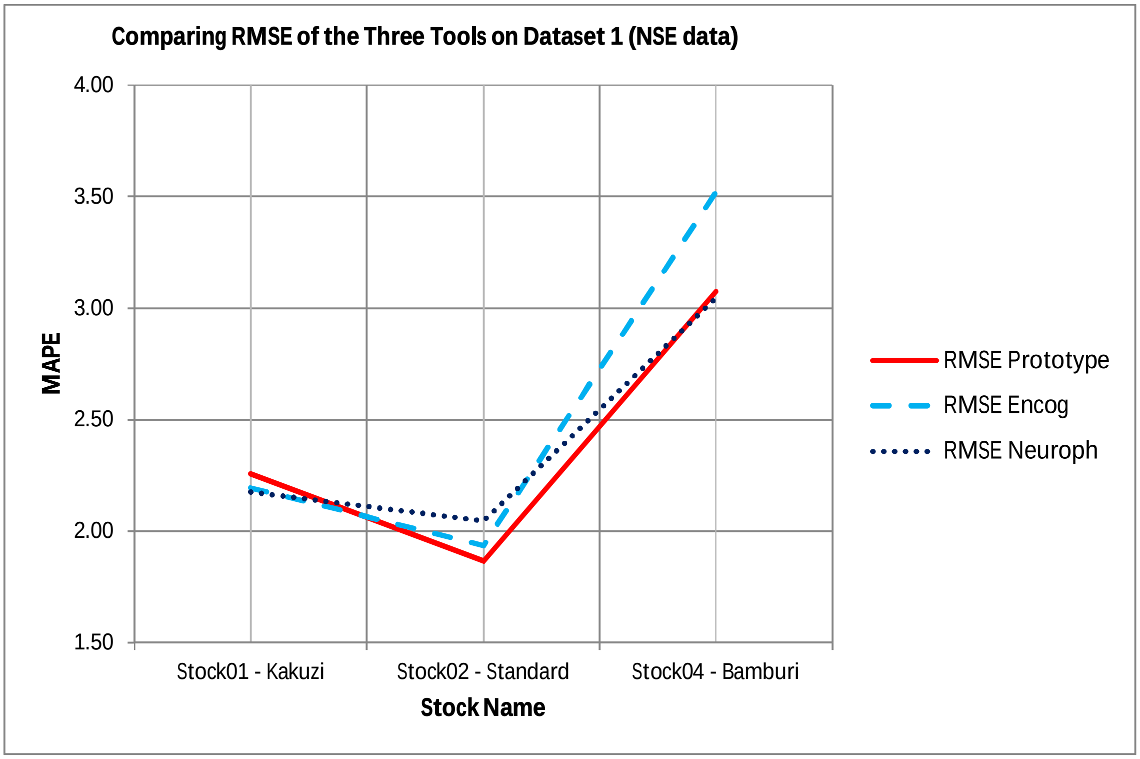


Figure 4.6 – Comparing MAPE of the Three Tools on NSE data

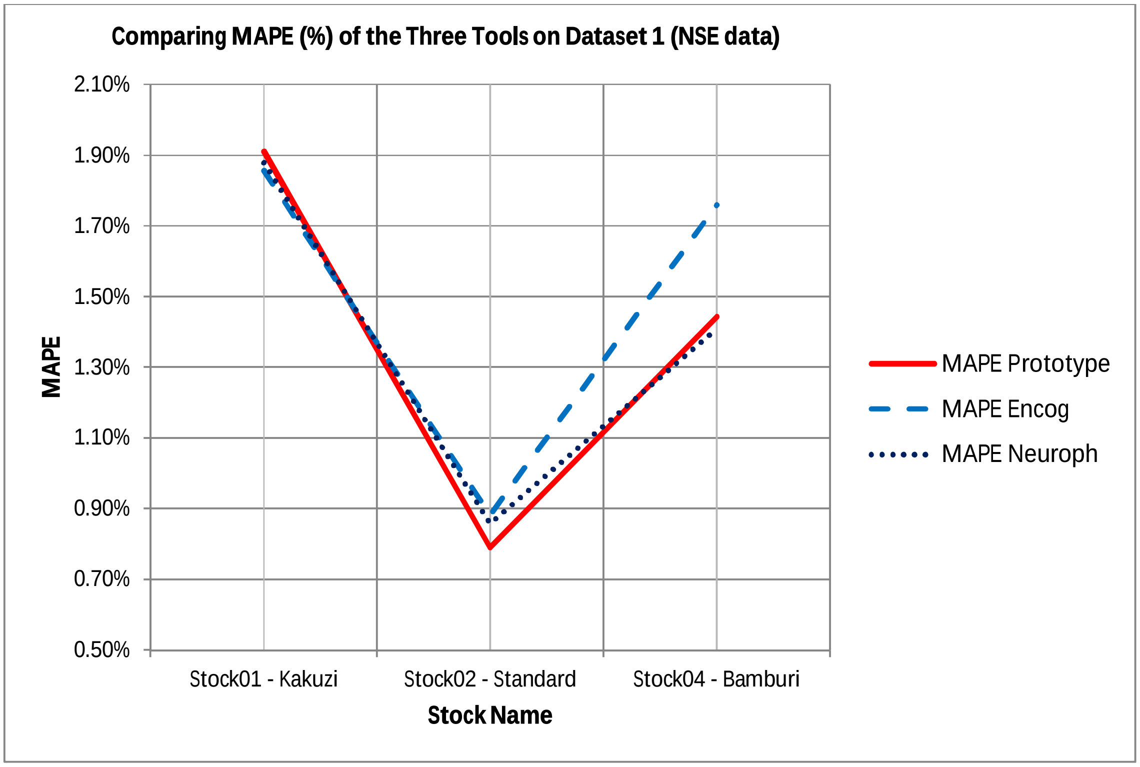


Figure 4.7 – Comparing RMSE of the Three Tools on NSE data

4.3.4 Observations and Analysis

It was observed that the error (MAPE) when using the prototype to predict the actuals was 0.77%, while the MAPE for Neuroph was 0.86% and 0.88% for Encog on the same Sstock02 (Standard bank). The RMSE of the three tools based on Stock02 (Standard bank) was 1.83 for the prototype, 1.94 for Encog and 2.85 for Neuroph. When comparing the performance of the protype on the three test stocks, the MAPE was 1.91% for Stock 01 (Kakuzi), 0.77% for Stock02 (Standard bank) and 1.44% for Stock04 (Bamburi). Out of 64 observations, the three tools had varied number of predictions that were lower than the actual i.e. 26, 33 and 29 for the prototype, Encog and Neuroph respectively. None of the three tools was able to give an exact prediction on any of the 63 observations. The number of predictions above the actual was 38, 31 and 35 for the prototype, Encog and Neuroph respectively. The highest variance on any day of trade in the 64-day period was noted to be 4.4% by Neuroph (22-Mar-2012)

4.3.5 Discussions

An overview of the three tools, based on MAPE and RMSE shows that the three tools performed almost at the same accuracy. Their MAPE was quite close for each of the three tests, though the use of RMSE was able to distinguish the most accurate of the tools for any of the datasets. MAPE is usually used as an error message for gauging how accurate a prediction is, with lower values being highly desirable. On the other hand, RMSE is useful when comparing different tools on the same dataset, with a view of finding out what one is better suited for a particular task. These results show that the prototype performed better than the other tools based on RMSE, for Stock02 (Standard bank). Due to this, the prototype outperformed the other tools in actual predictions on a day to day basis for the same Stock02 (Standard bank), as confirmed by the lower MAPE of 0.77% compared to 0.86% for Neuroph and 0.88% for Encog. An evident relationship between RMSE and MAPE was observed. Low RMSE values resulted to systems with low MAPE and vice versa.

4.4 Evaluating the Prototype Performance in Other Stock Markets

The final evaluation was achieved by applying the developed prototype to data from other stock markets. This test was based on stocks traded at the New York stock exchange, whose data was available in the public domain. Six popular stocks were selected for this test (Dataset 2). The data was for the time period 2008 to 2012. The selected stocks were random, based on popularity of these companies on the world commerce or their reputation. It was assumed that the issue of splits, bonuses and continuity had been addressed in the datasets as obtained. The stocks selected for the test were: Stock01 – Yahoo (Yahoo Financials, 2013f), Stock02 – Microsoft (Yahoo Financials, 2013c), Stock03 – IBM (Yahoo Financials, 2013b), Stock04 – Coca Cola (Yahoo Financials, 2013d), Stock05 – Toyota (Yahoo Financials, 2013e) and Stock06 – Alcatel (Yahoo Financials, 2013)

The data was based on US$ values, and was retained as US$ during training and testing. Due to time constraint, it was not possible to test all the six shortlisted stocks. A representative three were chosen on the basis of the percentage span of the traded prices in the training period. The price span for the stocks were Stock01 (234%), Stock02 (127%), Stock03 (79%), Stock04 (68%), Stock05 (98%) and Stock06 (570%). Based on these, the most suitable candidates were those with extreme price spans i.e. Stock04 and Stock06. The third candidate was the stock with a median span, in this case, either Stock02 or Stock05. Stock02 was selected.

4.4.1 Results

The results obtained for best performing stock, based on MAPE i.e. Stock04 (Coca Cola) is shown in Tables 4.3 below. The individual results for all the three test stocks are shown in **Appendix 14**. The results table shows the particular date of trade on column one and the corresponding actual closing stock value declared by the NSE analysis report on that date on column two. The third column provided the value that was predicted by the prototype, while the last column shows the error between the actual and the predicted values.

Table 4.3: *Stock04 (Coca Cola) Results – using Developed Prototype on NYSE data*

	Stock 04 – Coca Cola		
Test date	Actual	Predicted	Error (%)
03-Jan-12	33.39	33.18	-0.6%
04-Jan-12	33.18	33.26	0.2%
05-Jan-12	33.02	33.09	0.2%
06-Jan-12	32.81	32.95	0.4%
09-Jan-12	32.81	32.76	-0.2%
10-Jan-12	33.01	32.73	-0.8%
11-Jan-12	32.40	32.88	1.5%
12-Jan-12	32.17	32.41	0.7%
13-Jan-12	31.89	32.17	0.9%
17-Jan-12	32.06	31.97	-0.3%
18-Jan-12	32.18	32.00	-0.6%
19-Jan-12	32.11	32.14	0.1%
20-Jan-12	32.42	32.08	-1.0%
23-Jan-12	32.62	32.35	-0.8%
24-Jan-12	32.32	32.54	0.7%
25-Jan-12	32.58	32.32	-0.8%
26-Jan-12	32.38	32.51	0.4%
27-Jan-12	32.11	32.38	0.8%
30-Jan-12	32.12	32.10	-0.1%
31-Jan-12	32.15	32.13	-0.1%
01-Feb-12	32.30	32.13	-0.5%
02-Feb-12	32.29	32.24	-0.2%
03-Feb-12	32.41	32.26	-0.5%
06-Feb-12	32.39	32.36	-0.1%
07-Feb-12	32.63	32.36	-0.8%
08-Feb-12	32.53	32.54	0.0%
09-Feb-12	32.36	32.50	0.4%
10-Feb-12	32.34	32.35	0.0%
13-Feb-12	32.58	32.33	-0.8%
14-Feb-12	32.80	32.50	-0.9%
15-Feb-12	32.56	32.68	0.4%
16-Feb-12	32.78	32.54	-0.7%
17-Feb-12	32.87	32.69	-0.5%
21-Feb-12	32.76	32.78	0.1%
22-Feb-12	32.97	32.71	-0.8%
23-Feb-12	32.93	32.86	-0.2%
24-Feb-12	32.85	32.85	0.0%
27-Feb-12	32.77	32.80	0.1%
28-Feb-12	32.78	32.72	-0.2%
29-Feb-12	33.26	32.71	-1.7%
01-Mar-12	33.13	33.04	-0.3%
02-Mar-12	32.93	33.04	0.3%
05-Mar-12	32.96	32.90	-0.2%
06-Mar-12	32.73	32.88	0.5%
07-Mar-12	32.75	32.69	-0.2%
08-Mar-12	32.98	32.68	-0.9%
09-Mar-12	33.09	32.85	-0.7%
12-Mar-12	33.40	32.96	-1.3%
13-Mar-12	33.69	33.20	-1.5%
14-Mar-12	33.67	33.43	-0.7%
15-Mar-12	33.73	33.50	-0.7%
16-Mar-12	33.65	33.54	-0.3%
19-Mar-12	33.76	33.47	-0.9%
20-Mar-12	33.85	33.54	-0.9%
21-Mar-12	34.11	33.60	-1.5%
22-Mar-12	34.25	33.78	-1.4%
23-Mar-12	34.28	33.90	-1.1%
26-Mar-12	34.48	33.96	-1.5%
27-Mar-12	34.47	34.06	-1.2%
28-Mar-12	34.84	34.08	-2.2%
29-Mar-12	35.40	34.27	-3.2%
30-Mar-12	35.49	34.54	-2.7%
MAPE			**0.71%**

The results obtained are represented by graphs as shown on Figure 4.8 below, while the comparison of MAPE error for the prototype when applied to the three foreign market stocks is shown on the graph in Figure 4.9 below.

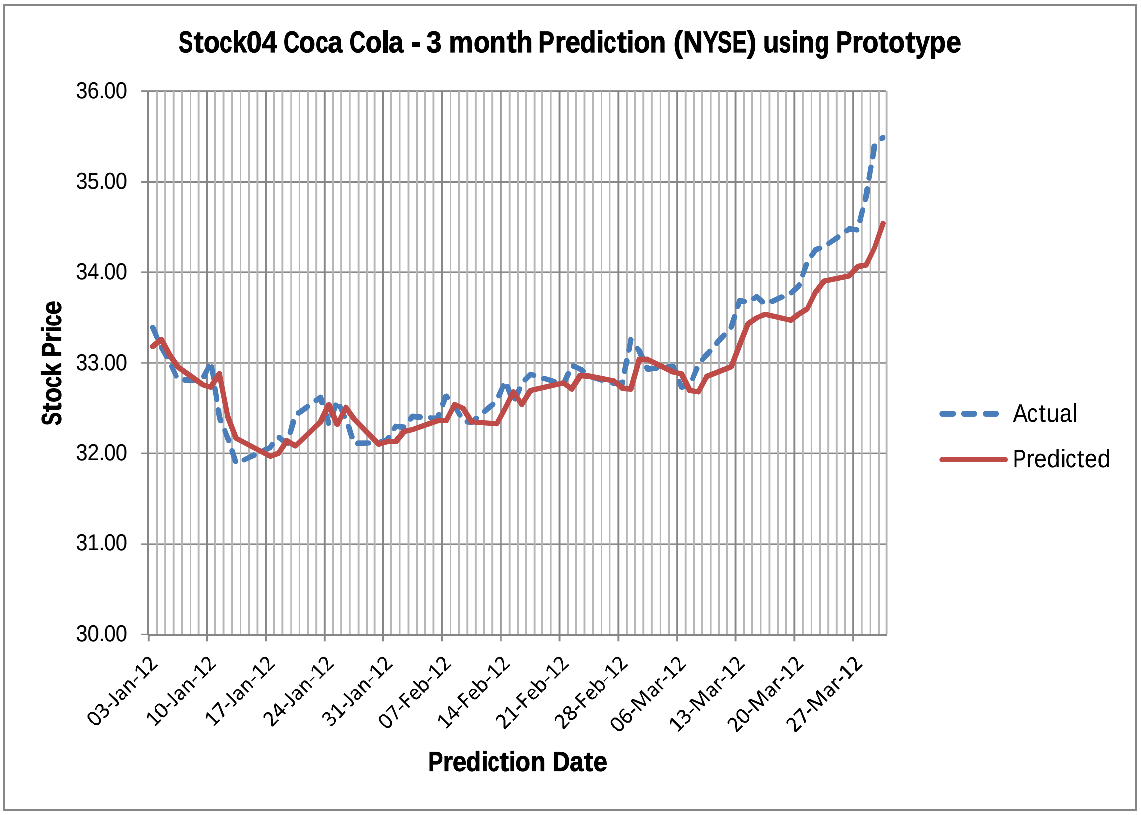


Figure 4.8 – Stock04 (Coca Cola) Results (Using ANN Prototype) on NYSE – Comparing Actual Against Predicted

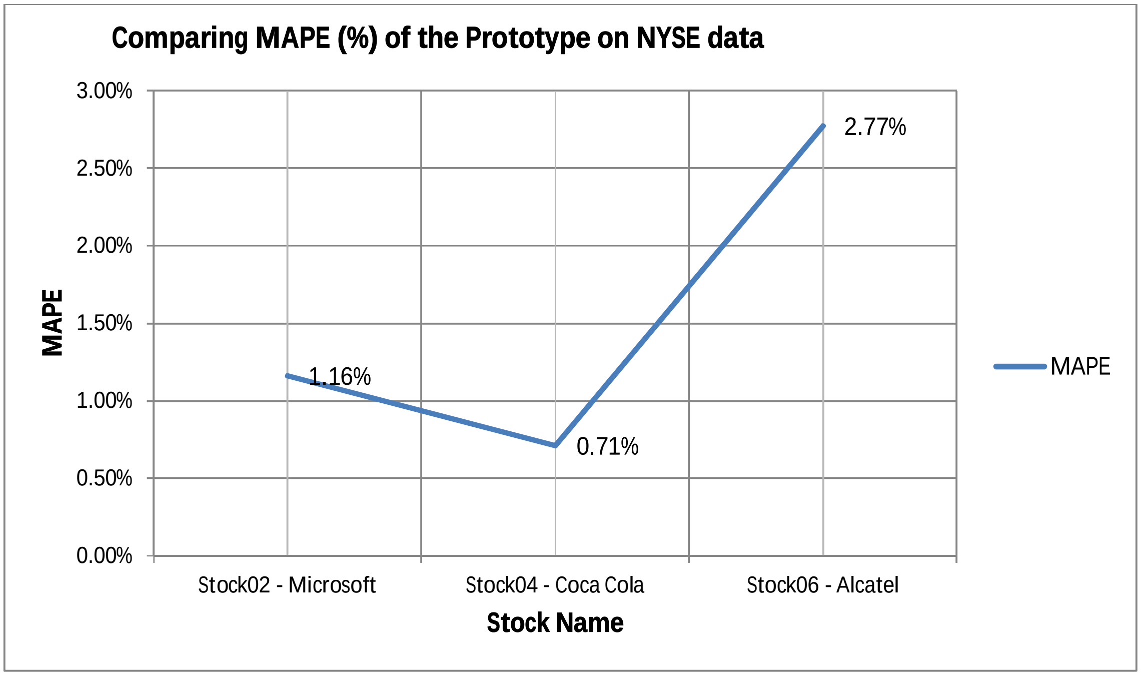


Figure 4.9 – Comparing MAPE of the Prototype on NYSE data

4.4.2 Observations and Analysis

The error (MAPE) when using the prototype to predict the actuals was 0.71%. When comparing the performance of the prototype on the three NYSE test stocks, the MAPE was 1.16% for Stock 02 (Microsoft), 0.71% for Stock04 (Coca Cola) and 2.77% for Stock06 (Alcatel). The predicted prices tended to be a little lower in value than the actual values i.e. 43 instances out of 62 observations. It was noted, as per raw data in Appendix 14, that some of the predictions were the same as the actual on some date of trade e.g. Stock02 (Microsoft) on 18-Jan-2012, 26-Jan-2012, Stock04 (Coca Cola) on 24-Feb-2012 and Stock06 (Alcatel) on 09-Jan-2012, 13-Feb-2012, 15-Feb-2012 and 23-Mar-2012. For Stock04 (Coca Cola), 18 predictions were a little higher than the actual on the respective dates. For this stock, the highest variance on any day of trade in the 62-day period was -3.2% (29-Mar-2012). However, for the three test stocks, the highest variance was on 10-Feb-2012 where the prototype realized an 11.0% MAPE error for Stock06 (Alcatel).

4.4.3 Discussions

The lowest MAPE when using the prototype at the NSE was 0.71% for Stock04 (Coca Cola). With MAPE being a measure of accuracy in prediction, lower values are usually desirable and this meant that the predicted values were generally close to the actual on the range of days that the test was done. This dataset also presented a situation where the prototype predicted the exact stock prices on some particular days. Seven predictions were exact for the three test stocks, with Stock06 (Alcatel) having the highest number of exact predictions being four exact predictions. The raw data indicates that this Stock06 (Alcatel) data generally has one significant digit and two decimal points. The other two stocks had two significant digits and two decimal places. It is safe to point out the relationship between the data range itself and the accuracy in prediction, with high likelihood of exact prediction occurring on data whose values of a smaller span. The Stock06 (Alcatel) values in the 62-day test period were in the range of 1.56 to 2.60.

4.5 Summary

The results obtained indicated that it was possible to develop a model which could be tested as a prototype. This prototype could be used for the prediction of the Kenyan stock market. This model is a multi-layer perceptron (MLP) using a feedforward network, trained using error backpropagation. This developed prototype is ideal for Stockbrokers who interact with the trading platform and execute trades as instructed by their clients. The same Stockbrokers are also expected to advise their clients and such tool can be very handy. The final model was formulated from a baseline model as recommended by other research findings. The baseline model was of configuration 5:11:11:1 with 70% of the available data used for training.

Through experimental evaluation, a new model was developed by first determining an optimum ANN architecture. This was done by conducting a series of experiments where the number of inputs, output and hidden layers were kept constant while varying the number of neurons on each hidden layer. The best performing configuration was a 5:21:21:1. Using this optimized network, the research then tested the effect of varying the training data volume on the prediction accuracy. By keeping the number of required predictions as 60 future values and assuming that there was a maximum of 5-year data, the research varied the volume of training data from 10% of available data to 90% of the data, from the same starting date of Jan. 3, 2008. The findings were that the best performance is achieved with 80% data volume used for training i.e. at least 1,000 training records are necessary for predicting the next 60 days. A final input to the model was the determination of the minimum number of training repetitions that were necessary for optimal results. This was found to be 130,000 repetitions.

The model was used to build a prototype for evaluation. When the model was tested on three selected NSE stocks, it achieved error (MAPE) of 0.77%, 1.44% and 1.91% in order of error values. The model was validated by comparing the prototype with other existing tools, where it was observed that the prototype was capable of desirable performance and performed just as well. Based on RMSE comparison of the three tools (Prototype, Neuroph and Encog), the prototype performed almost at the same level as the other two tools on the three chosen stocks. The respective RMSE for the three tools Prototype, Encog Workbench and Neuroph were 2.26, 2.19 & 2.18 for Stock01 (Kakuzi); 1.83, 1.94 and 2.85 for Stock02 (Standard Bank) and 3.07, 2.81 and 3.04 for Stock04 (Bamburi Cement). These were quite close with an extreme difference of only 0.26. NSE rules dictate that the change of prices for a stock during any particular day should be within a 10% swing of the previous closing average price, apart from during company announcements (NSE, 2010). Based on dataset 1 (NSE data), the prototype was able to meet the requirements of this rule by predicting correctly for all the 3 stocks in the 64-day period (192 predictions).

The prototype was tested for adaptability to other markets using data from a different stock exchange, the New York Stock Exchange, for data of the same period, for three selected stocks. It was observed that the stock price trend predicted followed the trend of the actual prices on the respective dates. Based on the results, where the MAPE was between 0.71% and 2.77%, this model-based prototype could therefore be used on any other stock exchanges, such as the NYSE. The criteria of selection of stocks may however need to be determined, as was done for the NSE, so as to determine qualifying stocks, or put in place measures to mitigate any issues that may be inherent on the raw data but not quite apparent unless additional information is gathered e.g. share splits, stock suspensions etc. It was also observed that the prototype was able to predict some seven instances of exact values in the datasets of the three test stocks from the NYSE.

The results of the experiments confirmed that the project ANN model was able to create low error systems based on the observed MAPE and comparative RMSE values. However, this project selected six stocks from the NSE and finally tested only three. Despite this, the NSE has 60 listed companies where investors trade daily. A model that can provide a prediction on the full range of companies all at once can be desirable, instead of having to carefully select the individual stocks as was done in this project. Developing such a model would require a method of ensuring that the data does not have biases or sources of erratic behavior. The performance of the prototype also varies from test to test. One particular trained network may be better at predicting the performance of one particular stock and not others. Developing a trained network that is well balanced and can predict optimally for all stocks is therefore desirable.

This chapter provided the results of the research made observations based on these results. The chapter finally discussed the results in relation to the project objectives. The next chapter provides the conclusion and recommendations.

CHAPTER 5
CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

Kenyan Stockbrokers are expected to advise their clients on investment choices with the aim of generating good returns for these customers. Only Stockbrokers can access the trading platform and execute stock trades at the Nairobi Securities Exchange. Advice to clients is usually on how stocks are expected to perform in the future, either the immediate future (next day) or long term. The stockbrokers currently use methods such as fundamental, technical and time series analysis in trying to predict future stock prices. These methods generally point to a trend on future performance and not the most likely prices. Without a more accurate prediction method, the stockbrokers may not be able to provide the requisite advice to their clients. There however exist Artificial Intelligence (AI) methods that can be used in developing prediction tools with the capability of indicating the most likely prices for future trades, rather than just the trends.

The research looked at the possibility of using AI to develop these predictive tools. The objectives of the research were to investigate AI models that were capable of financial market prediction so as to develop something similar for the stockbrokers in the local context. The research also aimed at developing an AI-based model and then developing a prototype based on this model. The prototype was to be the basis of testing the model. The final objective was to test model and evaluate its performance. This was made possible by the use of actual stock market data that was compiled from authorized sources for the period 2008 to 2012.

The AI method that was found suitable for developing such predictive models was the Artificial Neural Network (ANN), which exploits parallel computing to gain intelligence from input data as a basis of predicting future values. ANN structures, such as the multi-layer perceptron (MLP) using feedforward network with error backpropagation was found suitable and applicable for such models. One challenge with ANN based systems is the ability to formulate an optimal network, with the right inputs, right number of layers and the right number of neurons at each of the layers. Other considerations such as training cycles, activation function, stopping conditions, data volumes and data partitioning, also need to the formulated or determined. This research used experimental evaluation to determine these parameters, in order to come up with a model that could suit the stock market prediction problem.

Starting from a baseline design based on proposals by other scholars, the research conducted a series of experimental studies to develop a model that was well tuned to the task of stock market prediction. The research determined a configuration of 5:21:21:1 as achieving the highest prediction accuracy. This configuration consists of five input neurons, which correspond to the five day input into a network that aims at predicting the sixth day price. The model also consists of two hidden layers, each with 21 neurons. Finally, the model has one output, being the predicted price based on the five daily inputs.

NSE stock market data was obtained from authorized data vendors for a five year period 2008-2012, guided by the data volume needs for similar prediction applications. Secondary data was used because it was readily available and could also be independently confirmed and authenticated. Though the NSE has 60 counters, the research singled out six stocks for further evaluation based on the criteria of data consistency during the five year period.

Other considerations were taken into account in preparing the data that was suitable for the project. One consideration was the requirement that the stock should be a constituent counter of the stock market index (NSE 20-share index). This is an important index that gauges the national economy. A second consideration was the lack of share splits and bonus share issues during the period under review. Splits and bonus issues would usually affect the normal stock price by a factor of the split or bonus, causing an erratic jump. The research also conducted studies that confirmed that the data and hence the evaluation was free from political or economic influences in the period under review. Six NSE stocks qualified for use in this research i.e. Stock01 (Kakuzi), Stock02 (Standard Bank), Stock03 (Kenya Airways), Stock04 (Bamburi Cement), Stock05 (Kengen) and Stock06 (BAT).

After preparation of the data, experimental evaluation was conducted as a basis of developing the model. The holdout testing method was used and it was determined that at least 1,000 records, which is 80% of the data, was needed for the training set to achieve best results in the prediction of 60 future values (3-months). The number of repetitions needed for optimal training was found to be at least 130,000 repetitions. Such training would last about 1hr 40 minutes.

With the parameters for the model having been determined, the research developed a prototype based on the model, for purposes of evaluation. This phase involved formulating a design using pseudocode and flowcharts, then using these tools to guide the actual development of the prototype using a programming language. Visual C# was chosen and used as the programming environment. This choice was informed by the advantages of using a windows-based environment, the desire to use object-oriented technologies. The language also provides facilities that ease I/O operations on the data files, which was a continuous operation during the training phase, when many rows of data had to be read in bursts of five rows as at time. The prototype program code was tested by functional and debug tests to ensure that it was fit for practical use.

The prototype was then evaluated by experimental methods using NSE data as Dataset 1 (2008 – 2012). 80% of the data, for the period January 2, 2008 to December 31, 2011, was used for training. Out of the one-year data balance, three months period was singled out for testing, based on the normal financial analysis periods, which is usually in quarters. The NSE stocks used for final evaluation were – Stock01 (Kakuzi), Stock02 (Standard Bank) and Stock04 (Bamburi Cement). These stocks represented three of the ten sectors of the NSE, hence a fairly good sampling of the whole bourse. Validation of the prototype was done using two open source tools (Encog Workbench and Neuroph) on the same data, with the same ANN parameter settings.

When evaluated on the basis of the 2008-2012 NSE data, the model-based prototype achieved a Mean Absolute Percentage Error (MAPE) that was as low as 0.77% on one of the three test stock, Stock02 (Standard Bank). The highest MAPE was 1.91% on Stock01 (Kakuzi). The third stock, Stock04 (Bamburi Cement), had a MAPE of 1.44%. The developed prototype also compared quite favorably with other open source tools that were identified for comparative analysis and validation i.e. Neuroph and Encog Workbench. Based on MAPE, all tools performed almost at par. Comparing the Prototype, Neuoroph and Encog Workbench on Stock01 (Kakuzi) showed that the MAPE was 1.91%, 1.88% and 1.86% respectively. For the second stock, Stock02 (Standard Bank), the MAPE was 0.77%, 0.86% and 0.88% respectively, while comparative performance on the third stock, Stock04 (Bamburi Cement), showed a MAPE of 1.44%, 1.41% and 1.32% respectively.

Trading rules of the Nairobi Stock Exchange (NSE) dictate that price swings should be within 10% of the opening average prices. That means that any bid placed in the 10% range has the capacity of being executed. The project results over the test period of three months showed that all predicted values were capable of direct execution as provided by the prototype, since these prices were all within the 10% swing of the opening prices on each of the respective days in the 3-month period. Based on the 2008-2012 data, out of the 64-day period that the 3 stocks were tested i.e. 192 predictions, all predictions were within the 10% swing of the opening prices.

The prototype based on the developed model was observed to be generic enough to find applicability not only to the NSE but to other stock exchanges. Sample test done on three randomly selected stocks from the New York stock exchange (NYSE) showed that the prototype was capable of good prediction based on the same period (2008-2012) with error (MAPE) of 1.16% for Stock02 (Microsoft), 0.71% for Stock04 (Coca Cola) and 2.77% for Stock06 (Actatel). If this market had the same 10% maximum swing rule like the NSE, then only 1 prediction out of the total 186 predictions would be out of range, which is negligible. However, an analysis of the true values on respective trade dates show that swings of more than 10% are allowed on that market. This results therefore confirms that the prototype can be applied to other markets in the same domain.

The research was therefore able to achieve its objectives of developing a model that could be evaluated by a prototype for use at the stock market. The performance evaluation showed that the error (MAPE) while using the prototype both on the local stock market and other foreign markets was quite low, in the range 0.71% to 2.77%. The model also confirmed that it was possible to build AI-based prediction tools that are highly accurate and that can be put to practical use not only at the local bourse but at other similar bourses. The evaluation confirmed that all the predicted prices were actually falling within the allowable tradable stock prices during the next trading period i.e. 10% swing over or under the previous day's price. The predictions were also able to closely follow the trend of the actual for the respective days. That means that the proposed AI-methods could be relied upon for informing both the trend of price movement and also the most likely price in terms of the real-value stock price.

If stockbrokers are to use these AI-based tools, which have high accuracy, and next day prediction capabilities, instead of trends, more people can be encouraged to invest in the stock market since they can be assured of making money by right investments in stocks that sell at a profit or buy at a good bargain. This increased interest should in turn spur growth in this financial sector of the economy. The direct beneficiaries of such tools are the Stockbrokers, in the context of the stock market, since these are the people who act as the link between the investors and the stock market for purposes of trading.

The major contributions of this research are: an ANN model that realizes low MAPE in prediction, a model with comparatively low RMSE when measured against similar tools, a generic model that is applicable at any stock market and a new baseline model that can be subjected to further tweaking by other researchers.

5.2 Recommendations for Further Research

There is need to formulate a model that is generic enough to suit the full range of stocks at the stock market. This project carefully selected six stocks as basis of study. There is also need to explore ANN configurations above the 25 neurons studied in this project and also the effect of increasing the number of hidden layers from the two that were studied in this project. Finally, further research is needed to determine how long a trained ANN system remains valid and effective in prediction before it is found to be in need of retraining.

REFERENCES

ABC Capital. (2013). *Research.* Retrieved September 8, 2013, from http://www.abccapital.co.ke/research.php

Adebiyi, A. A., Ayo, C. K., Adebiyi, M. O., & Otokiti, S. O. (2012). Stock Price Prediction Using Neural Network with Hybridized Market Indicators. *Journal of Emerging Trends in Computing and Information Sciences , 3* (1), 1-9.

Adhikari, R., & Agrawal, R. K. (2013). A Combination of Artificial Neural Network and Random Walk Models for Financial Time Series Forecasting. *Neural Computing and Applications* .

Africa Alliance. (2013). *Research.* Retrieved April 3, 2013, from Africa Alliance: http://www.africanalliance.com/securities/research

African Fine Coffees Association (AFCA). (2013). *Kenya's Sasini in KShs. 349 million profit.* Retrieved July 10, 2013, from http://www.eafca.org/coffee%20news/newsdetail.asp?newsid=334

Aghababaeyan, R., & TamannaSiddiqui, N. (2011). Forecasting the Tehran Stock Market by Artificial Neural Network. *International Journal of Advanced Computer Science and Applications, Special Issue on Artificial Intelligence* .

AIB Capital Ltd. (2013). *Research*. Retrieved September 8, 2013, from AIB Capital Ltd: http://www.afrikainvestmentbank.com/index.php?option=com_content&view=article&id=54&Itemid=55

Apex Africa Capital Ltd. (2013). *Apex Africa Capital Ltd.* Retrieved September 8, 2013, from http://apexafrica.com/?cat=5

Bahrammirzaee, A. (2010). A Comparative Survey of Artificial Intelligence Applications in Finance: Artificial Neural Networks, Expert System and HybridIntelligent Systems. *Neural Computing and Applications , 19* (8), 1165-1195.

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., & Vaughan, J. W. (2010). A theory of learning from different domains. *Machine Learning , 79* (1-2), 151-175.

Bethard, S. (2008). *Statistical Learning Methods – Introduction to Artificial Intelligence.* Retrieved April 12, 2013, from Lecture notes online: http://clear.colorado.edu/~bethard/teaching/csci3202_2008/slides/learning-methods.pdf

Borodin, A., El-Yaniv, R., & Gogan, V. (2004). Can We Learn to Beat the Best Stock. *Journal of Artificial Intelligence Research , 21*, 579-594.

Butler, M., & Daniyal, A. (2009). Multi-objective Optimization with an Evolutionary Artificial Neural Network for Financial Forecasting. *Proceedings of the 11th Annual conference on Genetic and evolutionary computation*, (pp. 1451-1457).

Capital Markets Authority. (2013). *Licensed Stockbrokers*. Retrieved April 12, 2013, from http://www.cma.or.ke/index.php?option=com_content&view=article&id=208&Itemid=283

Central Bank of Kenya. (2009). *Annual Report 2009.* Retrieved April 3, 2013, from http://www.centralbank.go.ke/index.php/cbk-annual-reports

Central Bank of Kenya. (2011). *Financial Sector Stability Report 2011.* Retrieved April 12, 2013, from http://www.centralbank.go.ke/index.php/kenya-financial-sector-stability

Central Bank of Kenya. (2012). *Annual Report 2012.* Retrieved April 10, 2013, from http://www.centralbank.go.ke/index.php/cbk-annual-reports

Central Bank of Kenya. (2012, December). *Monthly Economic Review, December 2012*. Retrieved April 3, 2013, from http://www.centralbank.go.ke/index.php/monthly-economic-review

Central Bank of Kenya. (2012, June). *Statistical bulletin.* Retrieved April 3, 2013, from http://www.centralbank.go.ke/index.php/statistical-bulletin

Central Bank of Kenya. (2013, March 28). *Weekly bulletin.* Retrieved April 12, 2013, from http://www.centralbank.go.ke/index.php/weekly-bulletin

Central Depository and Settlement Corporation Limited (CDSC). (2013). *About Us*. Retrieved April 3, 2013, from http://www.cdsckenya.com/about-us/company-profile/

Cerna, L., & Chytry, M. (2005). Supervised Classification of Plant Communities with Artificial Neural Networks. *Journal of Vegetation Science , 16*, 407-414.

Chang, P., & Fan, C. (2008). A Hybrid System Integrating a Wavelet and TSK Fuzzy Rules for Stock Price Forecasting. *IEEE Transactions on Systems, Man and Cybernetics – Part C: Applications and Reviews , 38* (6), 802-815.

Chase Bank. (2013). *Investing in Kenya*. Retrieved April 3, 2013, from http://www.chasebank.co.ke/product/investing-kenya

Chen, M., Lin, C., & Chen, A. (2007). Constructing a Dynamic Stock Portfolio Decision-Making Assistance Model: Using The Taiwan 50 Index Constituents as an Example. *Soft Computing , 11* (12), 1149-1156.

Chen, Y., & Cheng, C. (2007). Forecasting Revenue Growth Rate Using Fundamental Analysis: A Feature Selection Based Rough Sets Approach. *Fourth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007), 3*, pp. 151-155.

Chen, Y., Wu, Q., & Chen, F. (2007). An IP and GEP Based Dynamic Decision Modelfor Stock Market Forecasting. *ISNN '07 Proceedings of the 4th international symposium on Neural Networks: Advances in Neural Networks*, (pp. 473-479).

Code Project. (2011). *Financial Predictor via Neural Network.* Retrieved August 6, 2013, from http://www.codeproject.com/Articles/175777/Financial-Predictor-via-Neural-Network

Cooperative Bank. (2013). *Invest on the NSE through Kingdom Securities*. Retrieved September 8, 2013, from http://www.co-opbank.co.ke/index.php/invest/diaspora/invest-on-the-nse-thro-kingdom-securities

Deng, S., Mitsubuchi, T., Shioda, K., Shimada, T., & Sakurai, A. (2011). Combining Technical Analysis with Sentiment Analysis for Stock Price Prediction. *2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing,* (pp. 800-807).

Devi, B. U., Sundar, D., & Alli, P. (2011). A Study on Stock Market Analysis for Stock Selection – Naïve Investors' Perspective using Data Mining Technique. *International Journal of Computer Applications , 34* (3), 19-25.

Drummond Investment Bank. (2013). *Portfolio Management*. Retrieved April 3, 2013, from http://www.drummond.co.ke/mambo2/index.php?option=com_content&task=view&id=43&Itemid=159

Encyclopedia of Mathematics. (2013). *Machine Learning*. Retrieved June 18, 2013, from http://www.encyclopediaofmath.org/index.php/Machine_learning

Gavrilov, M., Anguelov, D., Indyk, P., & Motwani, R. (2000). Mining the Stock Market (extended abstract): Which Measure Is Best? *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD'00* (pp. 487-496). ACM.

Genghis Capital. (2013). *Daily Market Notes*. Retrieved September 8, 2013, from http://genghis-capital.com/research/daily-market-notes

Ghaffari, A., Abdollahi, H., Khoshayand, M. R., Bozchalooi, I. S., Dadgar, A., & Refiee-Tehrani, M. (2006). Performance Comparison of Neural Network Training Algorithms in Modeling of Bimodal Drug Delivery. *International Journal of Pharmaceutics , 327* (1-2), 126-138.

Giles, L. C., Lawrence, S., & Tsoi, C. A. (2001). Noisy Time Series Prediction using Recurrent Neural Networks and Grammatical Inference. *Machine Learning , 44* (1-2), 161-183.

Gomes, G. S., Ludermir, T. B., & Lima, L. M. (2011). Comparison of New Activation Functions in Neural Network for Forecasting Financial Time Series. *Neural Computing and Applications , 20* (3), 417-439.

Government of Kenya. (2007). *Kenya Vision 2030 strategy*. Retrieved April 12, 2013, from http://www.planning.go.ke/index.php?option=com_docman&task=cat_view&gid=64&Itemid=69&limitstart=25

Government of Kenya. (2009). *The Capital Markets Act, Laws of Kenya, Cap 485A*. Retrieved April 12, 2013, from Capital Markets Authority: http://cma.or.ke/index.php?option=com_docman&view=docman&Itemid=123

Graham, B. (2003). *The Intelligent Investor* (Revised ed.). New York: HarperCollins Publishers Inc.

Han, S., & Cho, S. (2006). Evolutionary Neural Networks for Anomaly Detection Based on the Behavior of a Program. *IEEE Transactions on Systems, Man and Cybernetics – Part B: Cybernetics , 36* (3), 559-570.

Heaton Research. (2013). *Encog Download*. Retrieved September 12, 2013, from http://www.heatonresearch.com/download

Heaton Research. (2013b). *Encog Machine Learning Framework*. Retrieved August 2, 2013, from http://www.heatonresearch.com/encog

Heaton Research. (2013c). *The Number of Hidden Layers*. Retrieved August 2, 2013, from http://www.sciweavers.org/read/introduction-to-neural-networks-for-c-494

Heaton, J. (2008). *Introduction to Neural Networks for C#* (2nd ed.). Chesterfield: Heaton Research Inc.

Huang, C., Chen, P., & Pan, W. (2011). Using Multi-Stage Data Mining Technique to Build Forecast Model for Taiwan Stocks. *Neural Computing and Applications , 21* (8), 2057-2063.

Java Object Oriented Neural Engine (Joone). (2013). *Introduction*. Retrieved August 2, 2013, from http://www.codertodeveloper.com/

Kaushik, S., & Singhal, N. (2009). Pattern Prediction in Stock Market. *Artificial Intelligence 2009: Advances in Artificial Intelligence. Lecture Notes in Computer Science , 5866*, 81-90.

Keogh, E., & Kasetty, S. (2003). On the Need for Time Series Data Mining Benchmarks - A Survey and Empirical Demonstration. *Data Mining and Knowledge Discovery , 7* (4), 349–371.

Kestrel Capital. (2013). *Sample Research Reports*. Retrieved April 3, 2013, from http://www.kestrelcapital.com/kestrel_capital_sample_research_report.html

Khan, Z. H., Alin, T. S., & Hussain, A. (2011). Price Prediction of Share Market using Artificial Neural Network (ANN). *International Journal of Computer Applications (0975–8887) , 22* (2).

Klarman, S. A. (1991). *Margin of Safety*. USA: HarperCollins.

Kothari, C. R. (2004). *Research Methodology* (2nd Revised ed.). New Delhi: New Age International Publishers.

Koza, J. R. (2010). Human-competitive results produced by genetic programming. *Genetic Programming and Evolvable Machines , 11*, 251-284.

Lim, T., Loh, W., & Shih, Y. (2000). A Comparison of Prediction Accuracy, Complexity, and Training Time of Thirty-Three Old and New Classification Algorithms. *Machine Learning , 40* (3), 203-228.

Microsoft. (2013). *MSDN – Training and Testing Data Sets.* Retrieved April 3, 2013, from http://msdn.microsoft.com/en-us/library/bb895173.aspx

Mizrahi, C. S. (2008). *Getting Started in Value Investing*. New York: John Wiley & Sons Inc.

Naeini, M. P., Taremian, H., & Hashemi, H. B. (2010). Stock Market Value Prediction Using Neural Networks. *2010 International Conference on Computer Information Systems and Industrial Management Applications(CISIM).*

Nairobi Securities Exchange. (2010). *Nairobi Stock Exchange Trading Rules 2010*. Retrieved April 10, 2013, from CFC Stanbic: http://www.cfcstanbic.co.ke/financialservices/downloads/NSETradingRulesAugust12010.pdf

Nairobi Securities Exchange. (2013). *Authorized Data Vendors*. Retrieved April 3, 2013, from https://www.nse.co.ke/resource-center/information-products-and-services/authorized-data-vendors.html

Nairobi Securities Exchange. (2013b). *Equity Daily Statistics.* Retrieved July 19, 2013, from https://www.nse.co.ke/market-statistics/equity-statistics.html

Nairobi Securities Exchange. (2013c). *History of Organization*. Retrieved April 3, 2013, from https://www.nse.co.ke/about-nse/history-of-organisation.html

Nairobi Securities Exchange. (2013d). *List of Member Firms*. Retrieved April 3, 2013, from https://www.nse.co.ke/member-firms/firms.html

Nairobi Securities Exchange. (2013e). *Listed Companies*. Retrieved April 3, 2013, from https://www.nse.co.ke/listed-companies/list.html?limitstart=0

Nairobi Securities Exchange. (2013f). *Press Releases – NSE 20 Share Index Hits the 4,000 mark.* Retrieved April 3, 2013, from https://www.nse.co.ke/media-center/press-release.html?download=5895%3Anse-20-share-index-hits-the-4000-mark&start=20

Ndiritu, J. M. (2010). *Technical Trading Support System (TTSS): A Stock Market Analyst Support System.* MSc. Thesis, University of Nairobi, School of Computing and Informatics, Nairobi.

Neto, M., Calvalcanti, G., & Ren, T. (2009). Financial Time Series Prediction Using Exogenous Series and Combined Neural Networks. *Proceedings of International Joint Conference on Neural Networks June 14-19.* Atlanta, Georgia.

Neural Forecasting. (2013). *Portal onforecasting with artificial neural networks*. Retrieved September 13, 2013, from http://www.neural-forecasting.com/overview.htm

NeuroAI. (2013). *Stock market prediction*. Retrieved April 3, 2013, from http://www.learnartificialneuralnetworks.com/stockmarketprediction.html

Neuroph. (2013). *Neuroph Download*. Retrieved September 13, 2013, from http://neuroph.sourceforge.net/download.html

Neuroph. (2013b). *NeurophRM: Integration of the Neuroph framework into RapidMiner*. Retrieved September 13, 2013, from http://neuroph.sourceforge.net/rapidminer/NeurophRapidMiner.html

Neuroph. (2013c). *Stock Market Prediction Using Neural Networks - An example for time-series prediction*. Retrieved April 12, 2013, from http://neuroph.sourceforge.net/tutorials/StockMarketPredictionTutorial.html

NIC Securities. (2013). *Products.* Retrieved September 8, 2013, from http://nic-securities.com/products/

Nico. (2013). *The NICO Toolkit Overview.* Retrieved September 13, 2013, from http://nico.nikkostrom.com/

Oboko, R. (2012). *Adaptive Learning and Metacognitive Regulation for Ill-Structured Problem Solving Processes in Web-based Learning.* PhD Thesis, University of Nairobi, School of Computing and Informatics.

Old Mutual Securities. (2013). *Research & Reports*. Retrieved September 8, 2013, from http://www.oldmutual.co.ke/corporate/old-mutual-securities/research-and-reports.aspx

Ortiz-Rodriguez, J. M., Martinez-Blanco, M. R., Varamontes, J. M., & Vega-Carrillo, H. R. (2013). Robust Design of Artificial Neural Networks Methodology in Neutron Spectrometry. (K. Prof. Suzuki, Ed.) *Artificial Neural Networks - Architectures and Applications* .

Pan, H., Tilakaratne, C., & Yearwood, J. (2005). Predicting Australian Stock Market Index Using Neural Networks Exploiting Dynamical Swings and Intermarket Influences. *Journal of Research and Practice in Information Technology , 37* (1).

Patridge, D. (1998). *Artificial Intelligence and Software Engineering.* New York: Glenlake Publishing Co. Ltd.

Russell, S., & Norvig, P. (2009). *Artificial Intelligence - A Modern Approach* (3rd ed.). New Jersey: Prentice Hall.

Safaricom. (n.d.). *Annual Reports and Accounts 2009.* Retrieved July 10, 2013, from http://www.safaricom.co.ke/images/Downloads/Annual_Reports/2009_annual_report.pdf

Smith, S. W. (1997). *The Scientist and Engineer's Guide to Digital Signal Processing.* San Diego: California Technical Publishing.

Sterling Capital. (2013). *Company Announcements.* Retrieved July 10, 2013, from http://www.sterlingstocks.com/Research-Centre/Company-Announcements

Suntra. (2013). *Research*. Retrieved April 3, 2013, from Suntra: http://www.suntra.co.ke/?p=144

Synergy Systems Ltd. (2013). *MyStocks – Price list and trading summary for Friday, December 28, 2012*. Retrieved April 10, 2013, from MyStocks: http://live.mystocks.co.ke/price_list/20121228

Tanimoto, S. L. (1987). *The Elements of Artificial Intelligence.* Maryland: Computer SciencePress.

Thulasiram, R. K., & Bamgbade, A. Y. (2007). Application of an Instance Based Learning Algorithm for Predicting the Stock Market Index. *Computational Intelligence in Economics and Finance 2007* , 145-155.

Tiberius. (2013). *Predictive Modeling Software*. Retrieved April 10, 2013, from http://www.tiberius.biz/

Too, E. (2012). *A Comparative Study of Machine Learning Methods for Forecasting Prevalence of Weather-based Pests.* MSc. Thesis, University of Nairobi, School of Computing and Informatics.

Uchumi. (2013). *Investor Information*. Retrieved July 10, 2013, from http://kenya.uchumicorporate.co.ke/investors

Wagacha, P. W. (2003). *Support Vector Machines: A Critical Empirical Evaluation.* PhD Thesis, University of Nairobi, School of Computing and Informatics.

Wazua. (2013). *Investor Forum - Announcements.* Retrieved July 10, 2013, from http://www.wazua.co.ke/investor/corporateevents.aspx

Wong, C., & Versace, M. (2012). CARTMAP: A Neural Network Method for Automated Feature Selection in Financial Time Series Forecasting. *Neural Computing and Applications , 21* (5), 969-977.

Wu, X., Kumar, V., Quinlan, J. R., Ghosh, J., Yang, Q., Motoda, H., et al. (2008). Top 10 Algorithms in Data Mining. *An International Journal of Knowledge and Information Systems , 44* (1), 1-37.

Yahoo Finance. (2013). *Alcatel-Lucent, S.A.(ALU) – NYSE.* Retrieved August 5, 2013, from http://finance.yahoo.com/q/hp?s=ALU

Yahoo Finance. (2013b). *International Business Machines Corporation (IBM) – NYSE.* Retrieved August 5, 2013, from http://finance.yahoo.com/q/hp?s=IBM

Yahoo Finance. (2013c). *Microsoft Corporation (MSFT) – NasdaqGS*. Retrieved August 5, 2013, from http://finance.yahoo.com/q/hp?s=MSFT

Yahoo Finance. (2013d). *The Coca-Cola Company (KO) – NYSE.* Retrieved August 5, 2013

Yahoo Finance. (2013e). *Toyota Motor Corporation (TM) – NYSE*. Retrieved August 5, 2013, from http://finance.yahoo.com/q/hp?s=TM

Yahoo Finance. (2013f). *Yahoo! Inc. (YHOO) – NasdaqGS*. Retrieved August 5, 2013, from http://finance.yahoo.com/q/hp?s=YHOO

Zarandi, M. H., Hadavandi, E., & Turksen, I. B. (2012). A Hybrid Fuzzy Intelligent Agent-BasedSystem for Stock Price Prediction. *International Journal of Intelligent Systems , 27* (11), 947-969.

Zhang, J., Chung, H. S., & Lo, W. (2008). Chaotic Time Series Prediction Using a Neuro-Fuzzy System with Time-Delay Coordinates. *IEEE Transactions on Knowledge and Data Engineering , 20* (7).

Zhu, X. (2013). *Semi-Supervised Learning Tutorial*. Retrieved June 29, 2013, from Lecture notes online: http://www.cs.wisc.edu/~jerryzhu/pub/sslicml07.pdf

APPENDICES

APPENDIX 1 – NSE listed companies

Shown below are the 60 companies that traded at the NSE, as at April 2013:

AGRICULTURAL

1. Eaagads Ltd Ord 1.25
2. Kakuzi Ord.5.00
3. Kapchorua Tea Co. Ltd OrdOrd 5.00
4. Limuru Tea Co. Ltd Ord 20.00
5. Rea Vipingo Plantations Ltd Ord 5.00
6. Sasini Ltd Ord 1.00
7. Williamson Tea Kenya Ltd Ord 5.00

AUTOMOBILES AND ACCESSORIES

1. Car and General (K) Ltd Ord 5.00
2. CMC Holdings Ltd Ord 0.50
3. Marshalls (E.A.) Ltd Ord 5.00
4. Sameer Africa Ltd Ord 5.00

BANKING

1. Barclays Bank Ltd Ord 2.00
2. CFC Stanbic Holdings Ltd ord.5.00
3. Diamond Trust Bank Kenya Ltd Ord 4.00
4. Equity Bank Ltd Ord 0.50
5. Housing Finance Co Ltd Ord 5.00
6. Kenya Commercial Bank Ltd Ord 1.00
7. National Bank of Kenya Ltd Ord 5.00
8. NIC Bank Ltd 0rd 5.00
9. Standard Chartered Bank Ltd Ord 5.00
10. The Co-operative Bank of Kenya Ltd Ord 1.00

COMMERCIAL AND SERVICES

1. Express Ltd Ord 5.00
2. Hutchings Biemer Ltd Ord 5.00
3. Kenya Airways Ltd Ord 5.00
4. Longhorn Kenya Ltd 1.00
5. Nation Media Group Ord. 2.50
6. Scangroup Ltd Ord 1.00
7. Standard Group Ltd Ord 5.00
8. TPS Eastern Africa (Serena) Ltd Ord 1.00
9. Uchumi Supermarket Ltd Ord 5.00

CONSTRUCTION AND ALLIED

1. Athi River Mining Ord 5.00
2. Bamburi Cement Ltd Ord 5.00
3. Crown Berger Ltd 0rd 5.00
4. E.A.Cables Ltd Ord 0.50
5. E.A.Portland Cement Ltd Ord 5.00

ENERGY AND PETROLEUM

1. KenGen Ltd Ord. 2.50
2. KenolKobil Ltd Ord 0.05
3. Kenya Power LIGHTING Co Ltd Ord 20.00
4. Total Kenya Ltd Ord 5.00

INSURANCE

1. British-American Investments Ord 0.10
2. CFC Insurance Holdings
3. CIC Insurance Group Ltd Ord 1.00
4. Jubilee Holdings Ltd Ord 5.00
5. Kenya Re-Insurance Corporation Ltd Ord 2.50
6. Pan Africa Insurance Holdings Ltd 0rd 5.00

INVESTMENT

1. Centum Investment Co Ltd Ord 0.50
2. City Trust Ltd Ord 5.00
3. Olympia Capital Holdings ltd Ord 5.00
4. Trans-Century Ltd Ord 0.50

MANUFACTURING AND ALLIED

1. A.Baumann CO Ltd Ord 5.00
2. B.O.C Kenya Ltd Ord 5.00
3. British American Tobacco Kenya Ltd Ord 10.00
4. Carbacid Investments Ltd Ord 5.00
5. East African Breweries Ltd Ord 2.00
6. Eveready East Africa Ltd Ord.1.00
7. Kenya Orchards Ltd Ord 5.00
8. Mumias Sugar Co. Ltd Ord 2.00
9. Unga Group Ltd Ord 5.00

TELECOMMUNICATION AND TECHNOLOGY

1. AccessKenya Group Ltd Ord. 1.00
2. Safaricom Ltd Ord 0.05

Source: Nairobi Securities Exchange (2013)

APPENDIX 2 – NSE 20-share index companies

Shown below are the twenty companies that form the NSE 20-share index, as at April 2013:

AGRICULTURAL

1 Kakuzi Ord.5.00

2 Sasini Ltd Ord 1.00

AUTOMOBILES AND ACCESSORIES

-

BANKING

3 Barclays Bank Ltd Ord 2.00

4 Equity Bank Ltd Ord 0.50

5 Kenya Commercial Bank Ltd Ord 1.00

6 Standard Chartered Bank Ltd Ord 5.00

7 The Co-operative Bank of Kenya Ltd Ord 1.00

COMMERCIAL AND SERVICES

8 Kenya Airways Ltd Ord 5.00

9 Nation Media Group Ord. 2.50

10 Scangroup Ltd Ord 1.00

11 Uchumi Supermarket Ltd Ord 5.00

CONSTRUCTION AND ALLIED

12 Athi River Mining Ord 5.00

13 Bamburi Cement Ltd Ord 5.00

ENERGY AND PETROLEUM

14 KenGen Ltd Ord. 2.50

15 KenolKobil Ltd Ord 0.05

16 Kenya Power LIGHTING Co Ltd Ord 20.00

INSURANCE

-

INVESTMENT

-

MANUFACTURING AND ALLIED

17 British American Tobacco Kenya Ltd Ord 10.00

18 East African Breweries Ltd Ord 2.00

19 Mumias Sugar Co. Ltd Ord 2.00

TELECOMMUNICATION AND TECHNOLOGY

20 Safaricom Ltd Ord 0.05

Source: Nairobi Securities Exchange (2013)

APPENDIX 3 – List of Kenyan Stockbrokers

Shown below are the eleven licensed stockbrokers, as at April 2013:

	Stockbroker Name	**License No.**
1	ABC Capital Limited	01/12
2	AIB Capital Limited	02/12
3	ApexAfrica Capital Limited	03/12
4	Genghis Capital Limited	04/12
5	Francis Drummond and Company Limited	05/12
6	Kestrel Capital (E.A) Limited	06/12
7	Kingdom Securities Limited	07/12
8	NIC Securities Limited	08/12
9	Old Mutual Securities Limited	09/12
10	Sterling Capital Limited	10/12
11	Suntra Investment Bank Limited	12/12

Source: CMA (2013)

APPENDIX 4 – List of Popular Activation Functions

The listing below describes six popular threshold functions used in ANN designs:

a) **Threshold** – this neuron is called **perceptron**. The output is either 0 or 1. This is expressed in Equation (A4-1) below and represented as shown in Figure A4-1:

$$h_w(x) = \begin{cases} 0, & w.x < 0 \\ 1, & w.x \geq 1 \end{cases} \text{……………………………………………………(A4-1)}$$

Where:
w is the weight
x is the input value

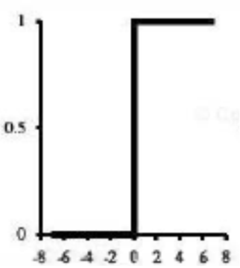


Figure A4-1 – Threshold function (Source: Russell et al. (2009))

b) **Logistic function (Unipolar sigmoid function)** – such ANN is called **sigmoid** perceptron. The output is a value between 0 and 1, as shown on Equation (A4-2) below and represented as shown in Figure A4-2:

$$Logistic(z) = \frac{1}{1+e^{-z}} \text{……………………………………………….. (A4-2)}$$

Where:
z is the input value to the activation function

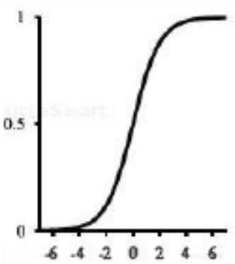


Figure A4-2 – Logistic/sigmoid function (Source: Russell et al. (2009))

c) **Bipolar sigmoid function** – It is also possible to have a variant of the sigmoid, called bipolar sigmoid function, with outputs in the range -1 to 1, as shown in equation A4-3 below.

$$Bipolar(z) = \frac{1-e^{-z}}{1+e^{-z}} \text{……………………………………………………… (A4-3)}$$

Where:
z is the input value to the activation function

d) **Hyperbolic tangent function** – An hyperbolic tangent function shown on equation A4-4 also gives output values in the range -1 to 1, for any given input (z).

$$tanh(z) = \frac{e^{z}-e^{-z}}{e^{z}+e^{-z}} \ldots\ldots\ldots\ldots \text{(A4-4)}$$

Where:
z is the input value to the activation function

e) **Piecewise linear function** – Provides outputs in the range 0 to 1, as described by equation A4-5.

$$h_w(z) = \begin{cases} 1 \mid & z \geq \frac{1}{2} \\ z \mid -\frac{1}{2} > z > \frac{1}{2} \\ 0 \mid & z \leq -\frac{1}{2} \end{cases} \ldots\ldots\ldots\ldots \text{(A4-5)}$$

Where:
z is the input value to the activation function

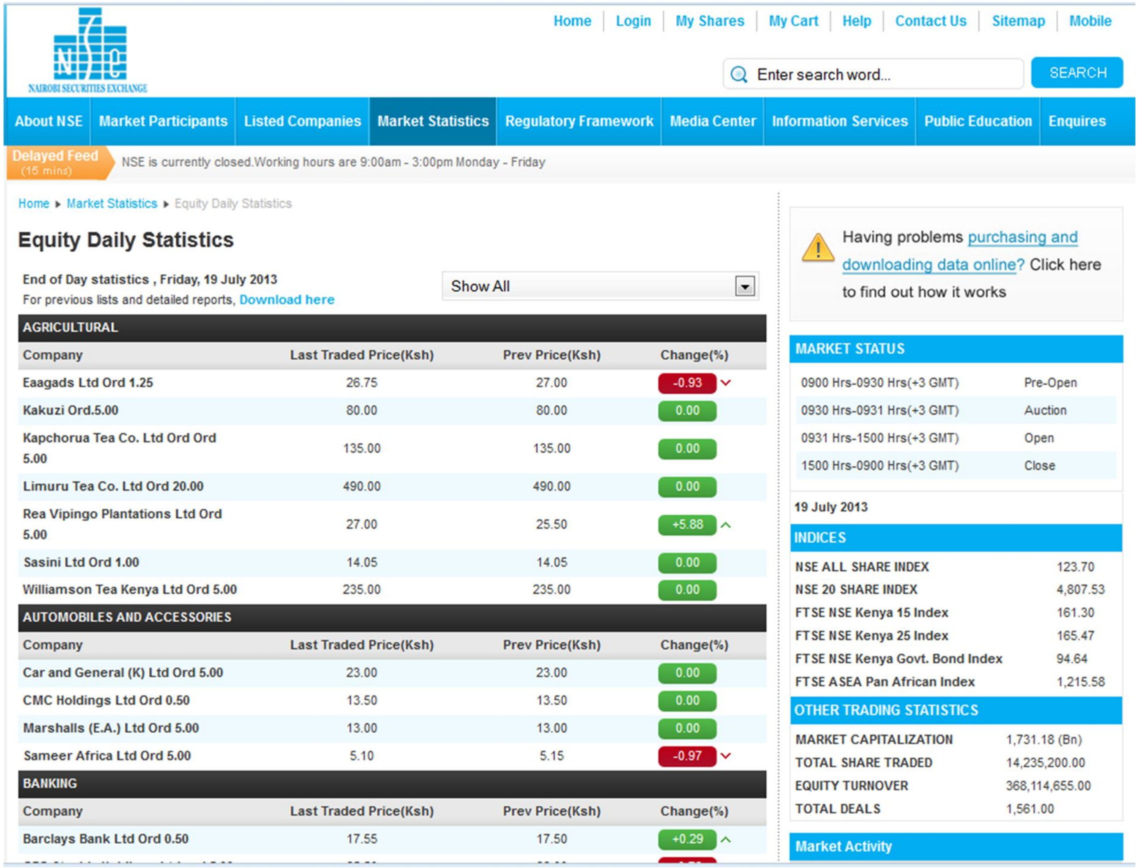
f) **Linear function** – Provides outputs that are same as the inputs, as described by equation A4-6.

$$f(z) = z \ldots\ldots\ldots\ldots \text{(A4-6)}$$

Where:
z is the input value to the activation function

APPENDIX 5 – Screenshot of NSE website

The screenshot below shows the NSE website indicating the stock prices on a typical trading day:


NAIROBI SECURITIES EXCHANGE
Home | Login | My Shares | My Cart | Help | Contact Us | Sitemap | Mobile
Enter search word...
SEARCH
About NSE | Market Participants | Listed Companies | Market Statistics | Regulatory Framework | Media Center | Information Services | Public Education | Enquires
Delayed Feed (15 mins)
Home ▸ Market Statistics ▸ Equity Daily Statistics
Show All

NSE is currently closed.Working hours are 9:00am - 3:00pm Monday - Friday

Equity Daily Statistics

End of Day statistics , Friday, 19 July 2013

For previous lists and detailed reports, Download here

AGRICULTURAL

Company	Last Traded Price(Ksh)	Prev Price(Ksh)	Change(%)
Eaagads Ltd Ord 1.25	26.75	27.00	-0.93 ⌄
Kakuzi Ord.5.00	80.00	80.00	0.00
Kapchorua Tea Co. Ltd Ord Ord 5.00	135.00	135.00	0.00
Limuru Tea Co. Ltd Ord 20.00	490.00	490.00	0.00
Rea Vipingo Plantations Ltd Ord 5.00	27.00	25.50	+5.88 ^
Sasini Ltd Ord 1.00	14.05	14.05	0.00
Williamson Tea Kenya Ltd Ord 5.00	235.00	235.00	0.00

AUTOMOBILES AND ACCESSORIES

Company	Last Traded Price(Ksh)	Prev Price(Ksh)	Change(%)
Car and General (K) Ltd Ord 5.00	23.00	23.00	0.00
CMC Holdings Ltd Ord 0.50	13.50	13.50	0.00
Marshalls (E.A.) Ltd Ord 5.00	13.00	13.00	0.00
Sameer Africa Ltd Ord 5.00	5.10	5.15	-0.97 ⌄

BANKING

Company	Last Traded Price(Ksh)	Prev Price(Ksh)	Change(%)
Barclays Bank Ltd Ord 0.50	17.55	17.50	+0.29 ^

Having problems purchasing and downloading data online? Click here to find out how it works

MARKET STATUS

0900 Hrs-0930 Hrs(+3 GMT)	Pre-Open
0930 Hrs-0931 Hrs(+3 GMT)	Auction
0931 Hrs-1500 Hrs(+3 GMT)	Open
1500 Hrs-0900 Hrs(+3 GMT)	Close

19 July 2013

INDICES

NSE ALL SHARE INDEX	123.70
NSE 20 SHARE INDEX	4,807.53
FTSE NSE Kenya 15 Index	161.30
FTSE NSE Kenya 25 Index	165.47
FTSE NSE Kenya Govt. Bond Index	94.64
FTSE ASEA Pan African Index	1,215.58

OTHER TRADING STATISTICS

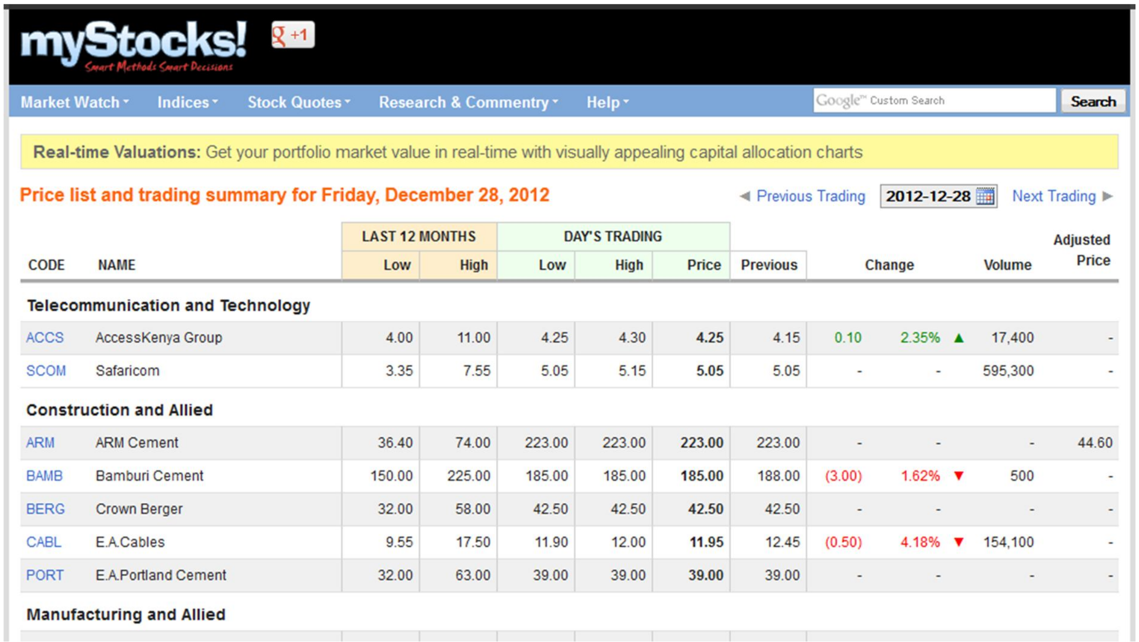
MARKET CAPITALIZATION	1,731.18 (Bn)
TOTAL SHARE TRADED	14,235,200.00
EQUITY TURNOVER	368,114,655.00
TOTAL DEALS	1,561.00

Market Activity

Source: Nairobi Securities Exchange (2013)

APPENDIX 6 – Screenshot of Synergy website

The screenshot below shows the myStocks! website indicating the stock prices on a selected trading day:



Real-time Valuations: Get your portfolio market value in real-time with visually appealing capital allocation charts

Price list and trading summary for Friday, December 28, 2012

◄ Previous Trading 2012-12-28 Next Trading ►

		LAST 12 MONTHS		DAY'S TRADING							Adjusted
CODE	NAME	Low	High	Low	High	Price	Previous	Change		Volume	Price
Telecommunication and Technology											
ACCS	AccessKenya Group	4.00	11.00	4.25	4.30	**4.25**	4.15	0.10	2.35% ▲	17,400	-
SCOM	Safaricom	3.35	7.55	5.05	5.15	**5.05**	5.05	-	-	595,300	-
Construction and Allied											
ARM	ARM Cement	36.40	74.00	223.00	223.00	**223.00**	223.00	-	-	-	44.60
BAMB	Bamburi Cement	150.00	225.00	185.00	185.00	**185.00**	188.00	(3.00)	1.62% ▼	500	-
BERG	Crown Berger	32.00	58.00	42.50	42.50	**42.50**	42.50	-	-	-	-
CABL	E.A.Cables	9.55	17.50	11.90	12.00	**11.95**	12.45	(0.50)	4.18% ▼	154,100	-
PORT	E.A.Portland Cement	32.00	63.00	39.00	39.00	**39.00**	39.00	-	-	-	-
Manufacturing and Allied											

Source: Synergy Systems Ltd. (2013)

APPENDIX 7 – Test Results of Baseline Evaluation on NSE data

These were the test results obtained when using the prototype on NSE data, to test the prediction error on the six test stocks for purposes of selection one for further tuning tests.

Table A7-1: *Baseline results (Stock01 to Stock03) – using Developed Prototype on NSE data*

	Stock01 (Kakuzi)			**Stock02 (Standard Bank)**			**Stock03 (Kenya Airways)**		
Date	Actual	Predicted	Error%	Actual	Predicted	Error%	Actual	Predicted	Error%
4-Jul-11	72.00	69.87	-3.0%	236.00	235.10	-0.4%	37.00	37.22	0.6%
5-Jul-11	70.00	71.67	2.4%	235.00	236.38	0.6%	36.50	36.67	0.5%
6-Jul-11	69.00	70.08	1.6%	235.00	234.99	0.0%	36.25	36.17	-0.2%
7-Jul-11	70.00	68.88	-1.6%	235.00	234.83	-0.1%	35.00	36.01	2.9%
11-Jul-11	69.50	69.84	0.5%	234.00	234.88	0.4%	34.00	34.35	1.0%
12-Jul-11	68.50	70.00	2.2%	230.00	233.77	1.6%	33.75	33.49	-0.8%
13-Jul-11	68.50	68.75	0.4%	228.00	228.98	0.4%	32.75	33.57	2.5%
14-Jul-11	72.00	68.49	-4.9%	228.00	227.02	-0.4%	33.50	32.14	-4.1%
15-Jul-11	72.00	71.34	-0.9%	228.00	227.28	-0.3%	34.25	33.79	-1.3%
18-Jul-11	72.00	72.06	0.1%	229.00	227.86	-0.5%	34.50	34.89	1.1%
19-Jul-11	68.50	71.85	4.9%	226.00	228.98	1.3%	34.75	34.69	-0.2%
20-Jul-11	68.00	68.76	1.1%	221.00	225.15	1.9%	34.50	35.28	2.3%
21-Jul-11	68.00	68.37	0.5%	221.00	220.28	-0.3%	34.50	34.83	1.0%
22-Jul-11	68.00	68.91	1.3%	225.00	220.23	-2.1%	34.25	34.80	1.6%
25-Jul-11	68.00	68.72	1.1%	222.00	225.25	1.5%	33.75	34.41	2.0%
26-Jul-11	68.00	68.18	0.3%	222.00	221.74	-0.1%	33.00	33.65	2.0%
27-Jul-11	68.00	68.11	0.2%	224.00	221.56	-1.1%	32.75	32.76	0.0%
28-Jul-11	68.00	68.11	0.2%	223.00	223.80	0.4%	32.25	32.66	1.3%
29-Jul-11	68.50	68.11	-0.6%	224.00	222.74	-0.6%	32.50	32.03	-1.4%
1-Aug-11	68.50	68.47	0.0%	224.00	223.75	-0.1%	32.75	32.67	-0.2%
2-Aug-11	68.50	68.58	0.1%	227.00	223.54	-1.5%	33.00	33.15	0.5%
3-Aug-11	70.00	68.50	-2.1%	224.00	227.37	1.5%	32.75	33.39	2.0%
4-Aug-11	70.00	69.64	-0.5%	225.00	223.21	-0.8%	33.25	33.03	-0.7%
5-Aug-11	70.00	70.01	0.0%	225.00	224.61	-0.2%	33.00	33.88	2.7%
8-Aug-11	69.00	69.81	1.2%	224.00	224.42	0.2%	32.50	33.28	2.4%
9-Aug-11	68.00	69.01	1.5%	218.00	223.70	2.6%	30.00	32.48	8.3%
10-Aug-11	66.50	68.34	2.8%	209.00	217.19	3.9%	30.00	28.98	-3.4%
11-Aug-11	66.50	67.39	1.3%	203.00	210.46	3.7%	30.00	29.81	-0.6%
12-Aug-11	70.00	67.18	-4.0%	205.00	203.45	-0.8%	30.00	29.83	-0.6%
15-Aug-11	67.00	69.55	3.8%	201.00	203.45	1.2%	30.50	29.71	-2.6%
16-Aug-11	67.00	68.15	1.7%	204.00	200.16	-1.9%	30.50	31.01	1.7%
17-Aug-11	67.00	66.56	-0.7%	207.00	203.49	-1.7%	30.75	30.80	0.2%
18-Aug-11	67.00	67.14	0.2%	213.00	207.53	-2.6%	31.75	31.15	-1.9%
19-Aug-11	71.50	67.71	-5.3%	217.00	214.48	-1.2%	32.25	32.64	1.2%
22-Aug-11	71.50	70.90	-0.8%	212.00	218.39	3.0%	33.25	32.77	-1.4%
23-Aug-11	71.50	71.84	0.5%	210.00	211.61	0.8%	32.75	33.92	3.6%
24-Aug-11	68.50	71.71	4.7%	209.00	210.05	0.5%	31.75	32.97	3.8%
25-Aug-11	70.00	68.44	-2.2%	207.00	208.83	0.9%	31.50	31.59	0.3%
26-Aug-11	70.00	69.67	-0.5%	201.00	207.26	3.1%	31.25	31.58	1.1%
29-Aug-11	73.00	70.43	-3.5%	201.00	201.57	0.3%	31.75	31.12	-2.0%
30-Aug-11	70.00	72.42	3.5%	203.00	200.53	-1.2%	31.50	32.06	1.8%
1-Sep-11	68.00	70.38	3.5%	200.00	202.49	1.2%	30.75	31.67	3.0%
2-Sep-11	68.00	67.94	-0.1%	200.00	199.79	-0.1%	30.00	30.53	1.8%
5-Sep-11	68.00	68.34	0.5%	201.00	200.06	-0.5%	30.25	29.70	-1.8%
6-Sep-11	68.00	68.97	1.4%	199.00	201.03	1.0%	30.50	30.31	-0.6%
7-Sep-11	67.00	68.41	2.1%	199.00	198.84	-0.1%	29.25	30.73	5.1%
8-Sep-11	67.00	67.42	0.6%	199.00	198.92	0.0%	31.00	28.63	-7.6%
9-Sep-11	66.00	67.18	1.8%	199.00	198.89	-0.1%	30.25	32.06	6.0%
12-Sep-11	66.00	66.76	1.2%	194.00	198.69	2.4%	30.75	30.30	-1.5%
13-Sep-11	66.00	66.50	0.8%	195.00	194.19	-0.4%	30.50	30.98	1.6%
14-Sep-11	66.00	66.52	0.8%	194.00	194.41	0.2%	31.00	30.95	-0.2%
15-Sep-11	66.00	66.53	0.8%	195.00	193.22	-0.9%	30.50	31.39	2.9%
16-Sep-11	67.50	66.35	-1.7%	195.00	193.88	-0.6%	31.25	30.63	-2.0%
19-Sep-11	68.00	67.43	-0.8%	196.00	194.65	-0.7%	30.25	31.85	5.3%
20-Sep-11	68.00	68.19	0.3%	195.00	195.48	0.2%	29.50	30.12	2.1%
21-Sep-11	68.00	68.08	0.1%	195.00	194.84	-0.1%	28.50	29.03	1.9%
22-Sep-11	68.50	67.84	-1.0%	196.00	194.60	-0.7%	28.00	27.98	-0.1%
23-Sep-11	68.50	68.40	-0.1%	195.00	195.57	0.3%	28.00	27.45	-2.0%
26-Sep-11	66.00	68.58	3.9%	195.00	194.57	-0.2%	27.75	27.72	-0.1%
27-Sep-11	67.50	66.78	-1.1%	194.00	194.60	0.3%	28.00	27.44	-2.0%
28-Sep-11	67.50	67.17	-0.5%	189.00	193.66	2.5%	27.25	28.04	2.9%
29-Sep-11	67.50	68.11	0.9%	186.00	189.77	2.0%	26.50	26.90	1.5%
30-Sep-11	67.00	67.85	1.3%	178.00	186.40	4.7%	25.75	25.94	0.7%
MAPE			**1.49%**			**1.04%**			**1.94%**

Table A7-2: *Baseline Results (Stock04 to Stock06) – using Developed Prototype on NSE data*

	Stock04 (Bamburi)			**Stock05 (Kengen)**			**Stock06 (BAT)**		
Date	Actual	Predicted	Error%	Actual	Predicted	Error%	Actual	Predicted	Error%
4-Jul-11	180.00	178.67	-0.7%	13.50	13.63	1.0%	260.00	261.73	0.7%
5-Jul-11	179.00	179.34	0.2%	13.15	13.48	2.5%	259.00	259.47	0.2%
6-Jul-11	179.00	179.28	0.2%	13.00	13.01	0.1%	261.00	259.77	-0.5%
7-Jul-11	175.00	178.91	2.2%	12.95	12.88	-0.5%	255.00	260.68	2.2%
11-Jul-11	173.00	175.78	1.6%	12.85	12.87	0.2%	259.00	256.29	-1.0%
12-Jul-11	175.00	172.72	-1.3%	12.75	12.76	0.1%	258.00	257.56	-0.2%
13-Jul-11	175.00	175.81	0.5%	12.45	12.66	1.7%	260.00	259.66	-0.1%
14-Jul-11	175.00	176.97	1.1%	12.25	12.35	0.8%	264.00	259.19	-1.8%
15-Jul-11	178.00	175.38	-1.5%	12.00	12.18	1.5%	267.00	264.22	-1.0%
18-Jul-11	178.00	177.09	-0.5%	11.75	11.93	1.5%	250.00	266.29	6.5%
19-Jul-11	178.00	177.66	-0.2%	12.00	11.69	-2.6%	263.00	251.32	-4.4%
20-Jul-11	175.00	177.31	1.3%	12.35	12.03	-2.6%	263.00	259.25	-1.4%
21-Jul-11	175.00	175.02	0.0%	12.60	12.52	-0.6%	260.00	269.30	3.6%
22-Jul-11	175.00	174.71	-0.2%	12.80	12.64	-1.3%	255.00	259.23	1.7%
25-Jul-11	175.00	176.10	0.6%	12.65	12.80	1.2%	255.00	257.29	0.9%
26-Jul-11	171.00	176.10	3.0%	12.50	12.55	0.4%	256.00	254.54	-0.6%
27-Jul-11	175.00	171.59	-1.9%	12.10	12.43	2.7%	255.00	257.88	1.1%
28-Jul-11	175.00	174.35	-0.4%	11.65	12.06	3.5%	251.00	255.85	1.9%
29-Jul-11	175.00	175.74	0.4%	11.10	11.64	4.9%	263.00	251.23	-4.5%
1-Aug-11	175.00	175.38	0.2%	11.75	11.00	-6.4%	258.00	260.80	1.1%
2-Aug-11	175.00	173.98	-0.6%	12.00	11.88	-1.0%	260.00	260.72	0.3%
3-Aug-11	175.00	175.39	0.2%	11.75	12.14	3.3%	260.00	259.31	-0.3%
4-Aug-11	175.00	175.39	0.2%	12.00	11.72	-2.3%	259.00	262.59	1.4%
5-Aug-11	173.00	175.39	1.4%	12.00	12.05	0.4%	259.00	259.08	0.0%
8-Aug-11	170.00	173.65	2.1%	11.75	12.01	2.2%	255.00	259.40	1.7%
9-Aug-11	170.00	170.29	0.2%	10.70	11.74	9.7%	257.00	255.66	-0.5%
10-Aug-11	172.00	170.05	-1.1%	10.90	10.66	-2.2%	254.00	256.21	0.9%
11-Aug-11	172.00	173.78	1.0%	11.00	10.80	-1.8%	248.00	255.50	3.0%
12-Aug-11	170.00	173.81	2.2%	11.00	10.92	-0.7%	250.00	247.56	-1.0%
15-Aug-11	170.00	170.30	0.2%	10.95	10.99	0.4%	235.00	248.75	5.9%
16-Aug-11	170.00	169.79	-0.1%	10.90	11.00	0.9%	235.00	245.58	4.5%
17-Aug-11	165.00	171.06	3.7%	11.05	10.92	-1.2%	238.00	233.10	-2.1%
18-Aug-11	165.00	165.69	0.4%	11.25	11.12	-1.2%	239.00	244.56	2.3%
19-Aug-11	165.00	165.90	0.5%	11.50	11.36	-1.2%	239.00	249.08	4.2%
22-Aug-11	165.00	165.45	0.3%	11.30	11.61	2.7%	240.00	240.87	0.4%
23-Aug-11	165.00	165.71	0.4%	11.35	11.35	0.0%	247.00	241.75	-2.1%
24-Aug-11	165.00	165.31	0.2%	11.35	11.43	0.7%	240.00	247.83	3.3%
25-Aug-11	155.00	165.31	6.7%	10.95	11.40	4.1%	238.00	242.59	1.9%
26-Aug-11	155.00	156.64	1.1%	10.55	10.92	3.5%	246.00	237.22	-3.6%
29-Aug-11	155.00	157.35	1.5%	10.35	10.49	1.4%	238.00	247.03	3.8%
30-Aug-11	165.00	153.25	-7.1%	10.05	10.16	1.1%	239.00	243.19	1.8%
1-Sep-11	164.00	160.00	-2.4%	10.20	9.84	-3.5%	237.00	238.30	0.5%
2-Sep-11	164.00	166.29	1.4%	10.00	10.02	0.2%	230.00	238.05	3.5%
5-Sep-11	163.00	163.54	0.3%	9.95	9.85	-1.0%	230.00	231.20	0.5%
6-Sep-11	163.00	160.64	-1.4%	10.00	9.90	-1.0%	232.00	226.93	-2.2%
7-Sep-11	163.00	163.33	0.2%	10.00	9.94	-0.6%	234.00	235.17	0.5%
8-Sep-11	160.00	163.01	1.9%	10.30	9.98	-3.1%	234.00	237.56	1.5%
9-Sep-11	160.00	160.52	0.3%	10.30	10.48	1.7%	235.00	236.28	0.5%
12-Sep-11	150.00	160.64	7.1%	10.25	10.52	2.6%	236.00	236.96	0.4%
13-Sep-11	159.00	153.18	-3.7%	10.20	10.37	1.7%	237.00	238.08	0.5%
14-Sep-11	160.00	157.93	-1.3%	10.20	10.20	0.0%	237.00	239.05	0.9%
15-Sep-11	160.00	158.90	-0.7%	10.20	10.15	-0.5%	240.00	238.95	-0.4%
16-Sep-11	160.00	162.56	1.6%	10.00	10.16	1.6%	240.00	241.76	0.7%
19-Sep-11	160.00	157.12	-1.8%	9.95	9.95	0.0%	236.00	242.13	2.6%
20-Sep-11	160.00	159.60	-0.3%	9.95	9.89	-0.6%	236.00	237.05	0.4%
21-Sep-11	160.00	159.80	-0.1%	9.45	9.86	4.3%	239.00	236.04	-1.2%
22-Sep-11	160.00	159.80	-0.1%	9.40	9.53	1.4%	243.00	240.74	-0.9%
23-Sep-11	160.00	159.80	-0.1%	9.50	9.48	-0.2%	239.00	244.90	2.5%
26-Sep-11	156.00	159.80	2.4%	9.40	9.49	1.0%	235.00	241.17	2.6%
27-Sep-11	160.00	156.66	-2.1%	9.30	9.68	4.1%	236.00	234.72	-0.5%
28-Sep-11	150.00	159.84	6.6%	9.35	9.75	4.3%	233.00	235.56	1.1%
29-Sep-11	150.00	153.00	2.0%	9.25	9.69	4.8%	231.00	235.60	2.0%
30-Sep-11	150.00	152.52	1.7%	9.35	9.73	4.1%	230.00	230.58	0.3%
MAPE			**1.41%**			**1.91%**			**1.69%**

APPENDIX 8 – Test Results for Varying ANN Neurons in Hidden Layers using NSE data

Results obtained when varying number of neurons in 2 hidden ANN layers from 1 to 25

Table A8-1: *Varying Neurons per Hidden Layer (1 to 25) – using Developed Prototype (NSE)*

	Neurons Per Hidden Layer									
Test	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Item	Error%	Error%	Error%	Error%	Error%	Error%	Error%	Error%	Error%	Error%
1	1.6%	-0.2%	-0.3%	-0.4%	-0.3%	-0.4%	-0.2%	-0.5%	-0.5%	-0.4%
2	2.5%	0.6%	0.6%	0.4%	0.3%	0.5%	0.7%	0.6%	0.5%	0.5%
3	1.9%	0.0%	0.0%	-0.2%	0.1%	0.1%	0.1%	0.2%	0.6%	0.1%
4	2.0%	0.1%	0.1%	-0.1%	-0.1%	0.0%	0.1%	0.1%	0.8%	0.0%
5	2.4%	0.5%	0.5%	0.3%	0.2%	0.4%	0.6%	0.6%	1.1%	0.4%
6	3.7%	1.7%	1.7%	1.6%	1.8%	1.7%	1.8%	1.8%	2.1%	1.7%
7	2.5%	0.5%	0.7%	0.6%	0.8%	0.7%	0.7%	1.0%	1.5%	0.8%
8	1.6%	-0.3%	0.0%	0.0%	-0.1%	-0.3%	0.1%	-0.2%	0.4%	-0.2%
9	1.7%	-0.2%	0.0%	0.1%	-0.2%	-0.2%	0.2%	-0.4%	-0.6%	-0.3%
10	1.3%	-0.5%	-0.4%	-0.4%	-0.1%	-0.4%	-0.6%	-0.8%	-1.8%	-0.5%
11	3.4%	1.4%	1.4%	1.4%	1.6%	1.3%	1.5%	1.1%	1.1%	1.5%
12	4.0%	1.9%	2.0%	2.0%	2.2%	2.2%	2.1%	2.4%	2.6%	2.3%
13	1.2%	-0.6%	-0.1%	-0.1%	0.0%	-0.2%	-0.2%	-0.2%	-0.4%	0.1%
14	-0.3%	-2.1%	-1.7%	-1.6%	-2.1%	-2.2%	-1.6%	-2.2%	-1.9%	-2.1%
15	3.4%	1.5%	1.5%	1.7%	1.4%	1.2%	1.7%	1.0%	-0.5%	1.2%
16	1.5%	-0.4%	-0.2%	-0.1%	0.5%	0.2%	-0.9%	0.1%	-0.4%	0.3%
17	0.9%	-0.9%	-1.0%	-0.8%	-0.5%	-0.9%	-0.8%	-1.1%	-1.1%	-0.6%
18	2.5%	0.4%	0.3%	0.5%	-0.1%	0.0%	0.3%	0.7%	0.5%	0.4%
19	1.2%	-0.6%	-0.6%	-0.5%	-0.1%	-0.3%	-0.7%	-0.5%	-0.5%	-0.2%
20	2.0%	0.0%	-0.1%	0.1%	0.3%	0.0%	0.2%	-0.2%	0.2%	0.2%
21	0.5%	-1.4%	-1.5%	-1.4%	-1.5%	-1.5%	-1.2%	-1.1%	-1.2%	-1.2%
22	3.6%	1.6%	1.4%	1.5%	1.6%	1.3%	1.7%	1.1%	1.4%	1.6%
23	1.2%	-0.8%	-0.8%	-0.7%	-0.5%	-0.6%	-0.5%	-0.1%	-0.5%	-0.3%
24	2.0%	0.1%	0.0%	0.2%	0.3%	-0.2%	0.1%	-0.3%	0.1%	0.2%
25	2.3%	0.3%	0.2%	0.4%	-0.1%	0.1%	0.5%	0.7%	0.6%	0.4%
26	4.5%	2.6%	2.7%	2.8%	3.2%	2.8%	2.5%	2.5%	2.8%	3.0%
27	5.3%	3.5%	4.1%	4.1%	4.1%	4.0%	4.0%	4.2%	3.8%	4.3%
28	2.5%	1.9%	3.3%	3.1%	2.7%	3.1%	3.1%	3.0%	2.7%	4.1%
29	-2.1%	-1.8%	-0.3%	-0.2%	-1.4%	-0.2%	0.3%	-0.6%	-1.9%	0.4%
30	1.8%	1.7%	2.8%	2.9%	1.9%	3.0%	3.3%	1.6%	4.0%	1.1%
31	-2.5%	-1.9%	-0.9%	-1.2%	-1.1%	-0.6%	-0.5%	-1.8%	-0.2%	-1.8%
32	-1.1%	-1.3%	-1.0%	-0.9%	-0.8%	-0.4%	-1.6%	-0.4%	-1.7%	-1.4%
33	-1.9%	-2.7%	-2.8%	-2.7%	-2.9%	-3.0%	-2.5%	-2.6%	-2.8%	-2.8%
34	0.0%	-1.3%	-1.6%	-1.3%	-1.0%	-1.1%	-1.0%	-1.4%	-1.8%	-1.5%
35	4.7%	2.7%	2.4%	2.7%	2.7%	2.8%	2.6%	3.2%	2.4%	2.7%
36	2.0%	0.6%	0.5%	0.5%	1.0%	0.8%	1.1%	1.6%	1.1%	1.3%
37	1.4%	0.2%	0.3%	0.4%	0.2%	0.3%	-0.2%	0.4%	1.0%	0.4%
38	1.5%	0.6%	0.9%	1.0%	0.2%	0.5%	1.0%	0.6%	1.0%	0.6%
39	3.1%	2.6%	3.3%	3.2%	3.3%	3.5%	2.8%	2.6%	2.6%	2.9%
40	-0.9%	-0.6%	0.2%	0.0%	-0.1%	0.2%	-0.2%	-0.3%	-0.2%	-0.1%
41	-1.4%	-1.2%	-0.6%	-0.6%	-1.1%	-0.4%	-1.0%	-0.5%	-1.5%	-1.3%
42	1.6%	1.5%	1.8%	1.8%	1.4%	1.8%	1.7%	1.3%	1.4%	1.2%
43	-0.6%	-0.2%	0.3%	0.0%	0.4%	0.4%	-0.4%	0.1%	-0.2%	0.1%
44	-0.7%	-0.5%	-0.3%	-0.4%	-0.4%	-0.2%	-0.6%	-0.3%	-0.4%	-0.5%
45	1.0%	1.0%	1.1%	1.0%	0.8%	0.9%	1.2%	0.9%	1.0%	0.9%
46	-0.6%	-0.1%	0.2%	0.0%	0.2%	0.1%	-0.3%	0.0%	-0.1%	0.1%
47	-0.3%	0.0%	0.2%	0.1%	0.1%	0.2%	-0.1%	0.0%	0.0%	-0.1%
48	-0.3%	0.0%	0.1%	0.0%	-0.1%	-0.1%	0.1%	-0.1%	0.0%	-0.1%
49	2.2%	2.6%	2.8%	2.6%	2.8%	2.8%	2.5%	2.6%	2.5%	2.6%
50	-1.9%	-0.7%	-0.5%	-0.7%	-0.7%	-0.8%	-0.9%	-0.8%	-0.4%	-0.6%
51	-0.3%	0.6%	0.7%	0.7%	0.5%	0.9%	0.2%	0.9%	0.2%	0.2%
52	-1.7%	-0.5%	-0.3%	-0.6%	-0.7%	-0.7%	-0.5%	-0.9%	-0.7%	-0.6%
53	-0.8%	0.2%	0.3%	0.2%	0.4%	0.6%	-0.2%	0.4%	-0.3%	0.0%
54	-1.2%	-0.4%	-0.4%	-0.6%	-0.5%	-0.7%	-0.3%	-0.5%	-0.5%	-0.5%
55	0.0%	0.7%	0.7%	0.6%	0.7%	0.7%	0.6%	0.6%	0.5%	0.6%
56	-0.8%	0.1%	0.1%	-0.1%	0.0%	-0.2%	0.1%	0.0%	0.1%	0.0%
57	-1.2%	-0.4%	-0.3%	-0.5%	-0.5%	-0.5%	-0.5%	-0.5%	-0.5%	-0.5%
58	0.0%	0.7%	0.7%	0.5%	0.5%	0.4%	0.7%	0.5%	0.5%	0.5%
59	-0.9%	0.1%	0.2%	-0.1%	0.1%	0.0%	-0.1%	0.1%	0.0%	0.1%
60	-0.2%	0.6%	0.7%	0.5%	0.6%	0.6%	0.5%	0.5%	0.5%	0.5%
61	1.6%	2.7%	2.8%	2.5%	2.5%	2.4%	2.6%	2.5%	2.6%	2.6%
62	-0.5%	1.5%	1.6%	1.5%	1.5%	1.5%	1.2%	1.3%	1.5%	1.3%
63	2.1%	4.4%	4.3%	4.6%	4.3%	4.6%	4.1%	4.7%	4.2%	3.8%
MAPE	**1.721%**	**1.041%**	**1.035%**	**1.021%**	**1.022%**	**1.030%**	**1.054%**	**1.046%**	**1.146%**	**1.032%**
RMSE	4.49	2.90	3.03	3.02	3.03	3.07	2.99	3.01	3.18	3.06

Table A8-1. cont.

Test Item	Neurons Per Hidden Layer									
	11	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**
	Error%	Error%	Error%	Error%	Error%	Error%	Error%	Error%	Error%	Error%
1	-0.4%	-0.5%	-0.3%	-0.5%	-0.5%	-0.2%	-0.4%	-0.3%	-0.7%	-0.3%
2	0.6%	0.4%	0.6%	0.6%	0.4%	0.7%	0.5%	0.3%	0.3%	0.7%
3	0.4%	-0.1%	0.4%	0.3%	0.1%	0.4%	0.0%	-0.1%	0.4%	0.2%
4	0.4%	-0.2%	0.3%	0.1%	0.2%	0.1%	0.0%	-0.1%	0.0%	0.0%
5	0.6%	0.3%	0.6%	0.5%	0.4%	0.6%	0.3%	0.2%	0.2%	0.6%
6	1.9%	1.5%	2.0%	1.9%	1.7%	2.0%	1.6%	1.7%	1.7%	2.0%
7	1.3%	0.8%	1.2%	1.4%	0.9%	0.8%	0.5%	0.8%	0.9%	0.9%
8	0.0%	0.3%	0.0%	0.0%	0.0%	-0.2%	-0.4%	0.2%	-0.4%	-0.1%
9	-0.4%	0.5%	-0.2%	-0.3%	-0.4%	0.0%	-0.5%	0.1%	-0.6%	0.4%
10	-0.7%	-0.4%	-0.5%	-0.5%	-0.9%	-0.3%	-0.6%	-0.6%	-1.0%	-0.1%
11	1.4%	1.3%	1.4%	1.3%	1.2%	1.7%	1.4%	1.5%	1.1%	1.5%
12	2.7%	1.8%	2.4%	2.6%	2.2%	2.4%	1.9%	2.1%	2.2%	2.2%
13	0.5%	-0.2%	-0.1%	0.5%	0.1%	0.8%	-0.2%	0.0%	-0.1%	-0.1%
14	-2.1%	-1.6%	-2.2%	-2.1%	-2.0%	-2.0%	-2.2%	-1.5%	-2.1%	-1.9%
15	0.9%	1.2%	1.0%	1.1%	0.6%	1.3%	1.2%	1.5%	0.8%	1.8%
16	0.1%	-0.4%	0.0%	0.4%	-0.3%	0.2%	-0.4%	-0.8%	0.0%	-1.8%
17	-0.4%	-1.3%	-0.8%	-1.0%	-0.6%	-0.5%	-0.9%	-0.6%	-0.8%	-1.3%
18	0.5%	0.3%	0.4%	0.4%	0.2%	0.4%	0.1%	0.3%	0.2%	0.4%
19	-0.3%	-0.6%	-0.4%	-0.3%	-0.6%	-0.1%	-0.6%	-0.6%	-0.6%	-0.5%
20	0.4%	-0.1%	0.1%	-0.1%	0.1%	0.3%	0.1%	0.3%	0.0%	-0.2%
21	-1.0%	-1.5%	-1.3%	-1.2%	-1.4%	-1.2%	-1.5%	-1.4%	-1.5%	-1.3%
22	1.5%	1.7%	1.5%	1.3%	1.5%	1.8%	1.7%	1.7%	1.3%	1.4%
23	0.2%	-0.9%	-0.3%	0.0%	-0.5%	-0.1%	-0.7%	-0.7%	-0.3%	-0.5%
24	0.4%	-0.2%	0.1%	-0.2%	0.2%	0.3%	0.0%	0.3%	0.0%	-0.5%
25	0.6%	0.3%	0.5%	0.5%	0.2%	0.5%	0.1%	0.4%	0.2%	0.6%
26	2.9%	2.5%	2.8%	2.8%	2.6%	3.2%	2.7%	2.6%	2.5%	2.6%
27	4.9%	4.0%	4.0%	4.8%	4.4%	4.9%	4.1%	4.3%	4.2%	4.2%
28	3.7%	6.1%	2.6%	4.0%	4.2%	3.4%	5.1%	3.3%	4.6%	3.8%
29	-0.5%	14.3%	-1.2%	-0.1%	12.9%	0.4%	-0.6%	0.0%	14.4%	0.3%
30	4.4%	7.8%	1.7%	1.9%	32.5%	2.1%	4.0%	2.7%	6.5%	1.6%
31	-1.5%	-2.3%	-1.2%	-2.1%	12.6%	-3.0%	-0.9%	-1.4%	-1.0%	-8.2%
32	-1.8%	-1.4%	-0.6%	-1.0%	-1.3%	-1.4%	-1.1%	-1.0%	-1.9%	-1.8%
33	-3.0%	-3.7%	-2.9%	-2.3%	-3.1%	-3.3%	-2.9%	-2.8%	-2.5%	-2.6%
34	-1.6%	-0.8%	-1.0%	-0.3%	-1.2%	-2.1%	-0.8%	-0.9%	-1.4%	-1.4%
35	1.8%	2.2%	2.7%	4.0%	2.5%	3.2%	3.2%	2.8%	2.2%	1.5%
36	0.6%	1.2%	0.7%	1.2%	1.2%	0.4%	1.0%	0.9%	1.1%	1.4%
37	-0.7%	-0.6%	0.2%	0.6%	0.8%	-0.2%	0.0%	0.4%	0.9%	0.9%
38	0.7%	1.3%	0.5%	0.7%	0.7%	0.7%	0.7%	1.3%	0.9%	1.2%
39	2.6%	3.1%	3.0%	2.9%	2.8%	3.0%	3.3%	2.6%	2.7%	2.5%
40	0.3%	-0.4%	0.0%	0.3%	0.5%	0.1%	0.8%	-0.3%	0.4%	-0.3%
41	-1.2%	-1.0%	-0.7%	-0.4%	-0.9%	-1.1%	-0.4%	-1.0%	-1.0%	-0.6%
42	0.9%	0.8%	1.7%	1.3%	0.9%	0.9%	1.6%	1.4%	1.1%	0.9%
43	0.0%	-0.4%	0.5%	-0.1%	0.2%	-0.1%	0.2%	-0.5%	-0.6%	-0.8%
44	-0.4%	-0.4%	-0.2%	-0.3%	-0.1%	-0.4%	-0.5%	-0.6%	-0.4%	-0.2%
45	0.8%	0.9%	0.9%	0.9%	0.8%	0.7%	0.8%	1.0%	1.2%	0.9%
46	0.0%	-0.1%	0.3%	-0.1%	0.2%	0.0%	0.2%	-0.4%	-0.4%	-0.3%
47	0.1%	0.2%	0.2%	0.1%	0.2%	0.0%	0.0%	-0.1%	0.0%	0.2%
48	-0.1%	0.0%	0.0%	-0.1%	-0.1%	-0.2%	-0.1%	0.0%	0.0%	-0.1%
49	2.5%	2.6%	2.9%	2.6%	2.7%	2.5%	2.8%	2.4%	2.4%	2.4%
50	-0.1%	-0.7%	-0.6%	-0.5%	-0.1%	-0.5%	-0.4%	-0.6%	-0.6%	-0.6%
51	0.5%	0.4%	0.9%	1.0%	0.5%	0.6%	0.6%	0.3%	0.4%	0.6%
52	-0.7%	-0.9%	-0.6%	-0.8%	-1.0%	-1.0%	-0.3%	-0.6%	-0.8%	-1.0%
53	-0.1%	-0.1%	0.8%	0.3%	0.1%	-0.1%	0.5%	0.0%	-0.5%	-0.4%
54	-0.4%	-0.4%	-0.6%	-0.6%	-0.8%	-0.6%	-0.5%	-0.4%	-0.3%	-0.6%
55	0.6%	0.7%	0.8%	0.6%	0.6%	0.5%	0.7%	0.5%	0.7%	0.5%
56	0.2%	0.2%	-0.1%	-0.1%	0.0%	0.0%	0.0%	0.1%	0.2%	-0.1%
57	-0.4%	-0.3%	-0.4%	-0.4%	-0.4%	-0.5%	-0.5%	-0.5%	-0.4%	-0.6%
58	0.5%	0.5%	0.5%	0.5%	0.3%	0.3%	0.5%	0.5%	0.7%	0.4%
59	0.1%	0.1%	0.1%	-0.1%	0.0%	0.0%	0.1%	-0.1%	0.0%	-0.2%
60	0.7%	0.7%	0.6%	0.6%	0.7%	0.5%	0.6%	0.5%	0.6%	0.5%
61	2.8%	2.8%	2.5%	2.5%	2.5%	2.6%	2.6%	2.6%	2.7%	2.4%
62	2.0%	1.2%	1.6%	1.7%	2.0%	1.5%	1.7%	1.4%	1.7%	1.4%
63	4.6%	3.8%	4.6%	4.9%	4.4%	4.2%	4.3%	4.4%	5.1%	5.3%
MAPE	**1.116%**	**1.419%**	**1.049%**	**1.093%**	**1.906%**	**1.095%**	**1.091%**	**1.036%**	**1.378%**	**1.220%**
RMSE	3.31	5.28	3.01	3.27	9.93	3.31	3.31	3.00	5.12	3.74

Table A8-1 cont.

	Neurons Per Hidden Layer				
Test	**21**	**22**	**23**	**24**	**25**
Item	Error%	Error%	Error%	Error%	Error%
1	-0.4%	-0.3%	-0.5%	-0.6%	-0.4%
2	0.4%	0.4%	0.4%	0.2%	0.3%
3	0.0%	0.2%	0.1%	-0.4%	0.3%
4	-0.1%	0.2%	-0.2%	-0.1%	0.1%
5	0.3%	0.5%	0.3%	0.2%	0.3%
6	1.7%	1.9%	1.8%	1.4%	2.0%
7	0.6%	1.1%	1.0%	0.4%	1.2%
8	-0.2%	0.1%	-0.1%	0.1%	0.0%
9	-0.2%	-0.2%	-0.2%	0.1%	-0.2%
10	-0.4%	-0.5%	-0.4%	-0.7%	-0.4%
11	1.3%	1.4%	1.3%	1.1%	1.3%
12	2.0%	2.4%	2.3%	1.8%	2.6%
13	-0.3%	0.3%	0.1%	-0.3%	0.0%
14	-2.1%	-1.9%	-2.1%	-1.5%	-2.0%
15	1.3%	1.0%	1.2%	1.4%	0.9%
16	0.0%	0.0%	0.3%	-0.4%	0.3%
17	-0.9%	-0.5%	-0.8%	-0.7%	-0.7%
18	0.2%	0.3%	0.1%	0.5%	-0.2%
19	-0.5%	-0.4%	-0.3%	-0.8%	-0.3%
20	0.0%	0.2%	-0.1%	0.0%	0.0%
21	-1.5%	-1.3%	-1.5%	-1.5%	-1.5%
22	1.4%	1.4%	1.3%	1.3%	1.1%
23	-0.7%	-0.3%	-0.5%	-0.8%	-0.2%
24	0.0%	0.3%	-0.1%	0.1%	0.0%
25	0.1%	0.4%	0.2%	0.3%	0.0%
26	2.7%	2.9%	2.9%	2.4%	3.0%
27	3.9%	4.5%	4.4%	3.8%	4.5%
28	2.5%	3.4%	3.2%	3.7%	5.0%
29	-1.8%	-0.9%	-1.3%	2.5%	3.6%
30	-0.2%	1.8%	1.5%	6.4%	1.2%
31	-2.4%	-1.7%	-1.7%	-0.2%	-3.2%
32	-1.7%	-1.2%	-1.3%	-1.4%	-1.4%
33	-2.8%	-3.1%	-2.7%	-2.9%	-2.6%
34	-1.4%	-1.5%	-1.2%	0.9%	-1.4%
35	3.0%	2.6%	2.7%	2.2%	2.9%
36	1.3%	1.5%	1.3%	2.7%	1.4%
37	0.4%	0.9%	0.6%	0.8%	0.3%
38	0.6%	0.6%	0.7%	0.6%	0.8%
39	2.7%	2.8%	2.9%	3.1%	1.5%
40	0.0%	0.0%	-0.1%	-0.3%	-1.7%
41	-1.1%	-1.1%	-1.3%	-0.6%	-1.1%
42	1.4%	1.2%	1.3%	1.1%	1.5%
43	0.1%	0.1%	0.0%	-0.5%	-0.2%
44	-0.4%	-0.4%	-0.4%	-0.3%	-0.6%
45	0.9%	0.8%	0.9%	1.1%	1.0%
46	0.2%	0.1%	0.0%	-0.2%	0.0%
47	0.1%	0.1%	0.0%	0.1%	-0.1%
48	0.0%	-0.1%	-0.1%	0.0%	-0.1%
49	2.7%	2.6%	2.6%	2.6%	2.5%
50	-0.5%	-0.5%	-0.7%	-0.8%	-0.7%
51	0.5%	0.4%	0.3%	0.1%	0.2%
52	-0.6%	-0.8%	-0.8%	-1.2%	-0.8%
53	0.1%	0.2%	-0.1%	-0.4%	-0.3%
54	-0.5%	-0.5%	-0.6%	-0.3%	-0.6%
55	0.6%	0.7%	0.6%	0.7%	0.4%
56	0.0%	0.0%	0.0%	0.2%	-0.1%
57	-0.5%	-0.4%	-0.5%	-0.3%	-0.7%
58	0.5%	0.5%	0.5%	0.7%	0.4%
59	0.1%	0.1%	-0.1%	0.2%	-0.2%
60	0.6%	0.6%	0.5%	0.7%	0.3%
61	2.6%	2.6%	2.5%	2.8%	2.4%
62	1.7%	1.6%	1.3%	0.9%	1.4%
63	4.4%	4.2%	4.0%	3.9%	4.3%
MAPE	**1.019%**	**1.060%**	**1.022%**	**1.113%**	**1.128%**
RMSE	2.99	3.08	2.99	3.35	3.37

APPENDIX 9 – Test Results for Data Volume Tests with the Model using NSE data

These were the test results obtained when using the prototype on NSE data, to test the error obtained when varying the available training data volume from 10% to 80%:

Table A9-1: *Varying Training Data Volume from the Available Pool – using Model (NSE)*

Test Item	**Training Data Volume (starting Jan. 2, 2008)**								
	10%	**20%**	**30%**	**40%**	**50%**	**60%**	**70%**	**80%**	**90%**
	Error%	Error%	Error%	Error%	Error%	Error%	Error%	Error%	Error%
1	0.3%	0.9%	1.7%	-1.0%	-0.5%	0.1%	-0.4%	-0.5%	-0.6%
2	0.5%	0.7%	-0.2%	-1.5%	0.6%	0.3%	0.6%	0.1%	0.0%
3	0.4%	-0.7%	0.3%	0.1%	-0.4%	-1.3%	0.1%	0.1%	0.4%
4	-0.6%	0.3%	-1.5%	0.1%	-1.3%	-0.4%	-0.1%	0.1%	5.2%
5	2.2%	2.3%	-1.1%	1.3%	-2.0%	-1.2%	0.3%	0.1%	-0.6%
6	-0.9%	2.8%	-0.5%	-2.7%	0.7%	1.0%	1.7%	0.0%	-1.3%
7	-1.4%	1.4%	-0.8%	0.4%	-1.0%	0.2%	0.6%	0.0%	-1.9%
8	-1.0%	2.1%	-0.1%	0.3%	1.4%	0.2%	-0.3%	0.0%	-0.6%
9	2.7%	1.7%	0.8%	-0.4%	-0.9%	-0.4%	-0.4%	-0.6%	1.9%
10	-0.1%	2.6%	-0.1%	-1.5%	-0.5%	-1.1%	-0.4%	0.1%	0.5%
11	1.9%	2.7%	-0.1%	-2.1%	-1.7%	0.0%	1.5%	0.7%	-0.6%
12	-1.3%	0.8%	0.8%	1.5%	-0.3%	-0.8%	2.1%	0.0%	-0.3%
13	-0.7%	-0.2%	0.2%	0.5%	-2.3%	-0.9%	-0.1%	1.2%	-0.3%
14	0.6%	0.3%	-1.4%	-0.4%	-2.6%	-0.9%	-2.2%	0.6%	-2.3%
15	0.2%	1.6%	-1.5%	0.6%	-0.8%	-0.4%	1.4%	1.2%	-0.7%
16	-0.4%	0.4%	-0.1%	-1.0%	-5.9%	-0.6%	0.3%	0.6%	0.8%
17	1.0%	0.7%	0.5%	-0.7%	-1.0%	-0.5%	-1.1%	-2.0%	0.4%
18	0.1%	1.8%	-0.6%	0.6%	-2.8%	-0.6%	0.1%	0.7%	0.5%
19	0.1%	1.1%	0.8%	1.3%	-4.2%	-1.1%	-0.3%	0.9%	0.1%
20	0.7%	-0.1%	-0.8%	0.4%	-5.3%	-0.3%	0.0%	-0.6%	-0.1%
21	0.3%	-0.3%	-0.4%	-0.4%	-11.5%	-0.7%	-1.4%	0.1%	2.6%
22	1.1%	1.3%	-0.9%	0.8%	-14.2%	-0.6%	1.6%	0.2%	2.0%
23	0.2%	0.1%	-2.1%	1.8%	-13.7%	-0.2%	-0.5%	-1.2%	-0.3%
24	1.8%	0.9%	-4.1%	1.3%	-13.7%	-0.7%	-0.2%	1.4%	-4.4%
25	0.2%	0.0%	-3.1%	-0.8%	-10.8%	-0.5%	0.2%	0.2%	-0.7%
26	3.1%	4.6%	-0.7%	-0.1%	-5.0%	-0.6%	2.8%	-0.7%	1.4%
27	1.1%	4.2%	0.8%	0.4%	-10.6%	-0.6%	4.0%	0.1%	-1.4%
28	3.9%	-1.6%	0.6%	0.7%	-2.2%	-0.5%	5.0%	0.2%	5.4%
29	-2.2%	6.2%	-0.4%	-0.3%	-5.3%	-0.5%	1.8%	0.7%	-3.2%
30	5.2%	9.8%	2.3%	-0.2%	-11.4%	-1.9%	0.4%	-0.6%	2.0%
31	2.2%	12.3%	-1.7%	0.4%	-7.4%	1.0%	-2.9%	-0.6%	-1.9%
32	4.3%	10.7%	-1.0%	-1.0%	-0.3%	-2.2%	-1.7%	0.8%	-1.3%
33	-1.1%	11.4%	-0.4%	-0.6%	-4.8%	0.6%	-2.8%	-0.5%	-0.7%
34	2.8%	13.9%	0.7%	0.1%	-3.1%	-0.4%	-0.7%	-0.6%	0.1%
35	0.9%	14.5%	0.9%	-1.4%	-1.0%	0.3%	3.3%	0.2%	1.0%
36	2.3%	15.2%	2.6%	-0.7%	-5.1%	-0.4%	0.6%	0.2%	0.3%
37	1.5%	14.5%	-1.0%	-1.5%	-2.6%	-1.2%	0.0%	-1.2%	0.1%
38	0.9%	14.4%	-0.3%	-1.1%	-3.0%	-1.5%	0.7%	-0.5%	-0.6%
39	1.3%	16.4%	0.6%	0.8%	-4.2%	-1.6%	2.8%	-0.4%	-0.1%
40	0.6%	16.8%	-1.9%	-1.3%	-2.9%	-1.5%	0.4%	-0.5%	-0.5%
41	2.6%	16.6%	-3.8%	0.7%	-3.2%	-0.9%	-0.8%	0.1%	0.6%
42	1.3%	17.5%	0.4%	-0.8%	-2.5%	-0.6%	0.6%	-2.3%	-0.9%
43	2.6%	24.5%	0.1%	-0.3%	-3.1%	-1.0%	-0.2%	-1.7%	-0.8%
44	1.6%	20.0%	1.5%	-0.7%	-2.9%	-0.8%	-0.3%	-1.5%	-1.3%
45	3.0%	14.1%	-0.3%	1.0%	-3.0%	-1.6%	0.8%	0.2%	-0.2%
46	1.8%	13.2%	-1.4%	2.0%	-3.1%	0.6%	0.0%	0.0%	-2.1%
47	2.8%	16.1%	1.9%	-1.3%	-3.2%	2.1%	0.1%	-1.2%	0.3%
48	2.5%	26.8%	-2.0%	0.4%	-3.9%	-0.3%	-0.1%	-0.6%	0.5%
49	2.5%	23.0%	0.9%	1.2%	-6.8%	1.5%	2.5%	1.2%	1.1%
50	4.1%	19.9%	0.1%	-0.1%	-6.1%	-2.0%	-0.3%	0.5%	-1.0%
51	5.0%	19.9%	1.0%	-1.3%	-6.7%	-0.2%	0.4%	-0.2%	-2.3%
52	6.7%	18.2%	0.4%	0.3%	-6.4%	-1.5%	-0.8%	0.5%	0.0%
53	11.5%	19.8%	0.5%	0.7%	-6.9%	2.1%	-0.1%	2.9%	-0.5%
54	8.5%	19.3%	-2.3%	-1.4%	-7.6%	-0.2%	-0.6%	-2.0%	-1.2%
55	14.3%	17.6%	0.4%	-0.8%	-9.2%	0.8%	0.4%	-0.8%	1.2%
56	9.6%	20.2%	0.7%	0.0%	-8.1%	1.7%	0.0%	1.3%	1.3%
57	11.0%	19.4%	0.4%	-0.1%	-8.1%	-0.1%	-0.5%	0.7%	9.3%
58	10.7%	19.4%	0.5%	0.3%	-7.3%	0.8%	0.3%	4.1%	1.4%
59	4.5%	18.5%	-0.3%	-1.4%	-7.8%	0.2%	0.0%	-1.6%	-2.9%
60	9.6%	18.7%	1.5%	1.2%	-8.6%	0.3%	0.5%	-2.8%	1.6%
MAPE	**2.833%**	**9.618%**	**1.008%**	**0.831%**	**4.726%**	**0.819%**	**0.961%**	**0.783%**	**1.326%**
RMSE	7.57	17.40	1.91	1.91	16.58	2.69	3.05	1.86	4.04

APPENDIX 10 – Test Results for Training Repetitions with the Model using NSE data

These were the test results obtained when using the prototype on NSE data, to test the error obtained when the training cycles are varied from 0 to 180,000.

Table A10-1: *Varying Training Repetitions – using Stock02 on Developed Model (NSE)*

Test	**Test 1***		**Test 2**		**Test 3**		**Test 4**	
Reps	Error	Time	Error	Time	Error	Time	Error	Time
1,000	0.0163	0:00:24	0.0161	0:00:17	0.0161	0:00:26	0.0152	0:00:23
5,000	0.0153	0:01:43	0.0155	0:01:58	0.0155	0:02:19	0.0151	0:02:12
10,000	0.0148	0:03:56	0.0153	0:04:10	0.0153	0:04:45	0.0149	0:04:33
20,000	0.0145	0:08:22	0.0148	0:09:03	0.0152	0:10:27	0.0140	0:10:24
30,000	0.0142	0:14:00	0.0146	0:14:55	0.0151	0:16:08	0.0144	0:15:23
40,000	0.0141	0:20:24	0.0145	0:21:15	0.0147	0:21:29	0.0143	0:22:19
50,000	0.0139	0:27:53	0.0144	0:28:16	0.0145	0:28:30	0.0139	0:27:43
60,000	0.0138	0:35:56	0.0144	0:36:15	0.0142	0:36:17	0.0136	0:36:28
70,000	0.0136	0:43:16	0.0143	0:43:54	0.0141	0:42:15	0.0134	0:44:37
80,000	0.0134	0:52:04	0.0143	0:52:47	0.0137	0:52:48	0.0133	0:53:28
90,000	0.0133	1:00:10	0.0143	1:02:03	0.0136	1:02:13	0.0132	1:03:53
100,000	0.0132	1:07:49	0.0143	1:14:57	0.0135	1:12:17	0.0132	1:15:12
110,000	0.0132	1:17:16	0.0143	1:21:45	0.0134	1:22:07	0.0131	1:24:18
120,000	0.0131	1:26:33	0.0143	1:32:55	0.0134	1:27:15	0.0131	1:27:46
130,000	0.0129	1:40:22	0.0143	1:44:16	0.0133	1:46:09	0.0131	1:48:13
140,000	0.0129	1:47:03	0.0143	2:01:40	0.0133	1:57:08	0.0130	1:56:30
150,000	0.0128	1:58:15	0.0143	2:08:14	0.0133	2:09:41	0.0130	2:07:26
160,000	0.0127	2:11:58	0.0143	2:21:03	0.0132	2:25:05	0.0130	2:24:10
170,000	0.0127	2:24:13	0.0143	2:35:48	0.0132	2:36:34	0.0129	2:35:32
180,000	0.0126	2:31:40	0.0143	2:44:50	0.0132	2:50:37	0.0128	2:51:40

Note – after 12hours of training (Test 4), the error was 0.01203 in 368,200 reps

APPENDIX 11 – Test Results Using Developed Prototype on NSE data

These were the test results obtained by using the prototype on NSE data (best 3 results):

Table A11-1: *Stock01 (Kakuzi) results – using Developed Prototype on NSE data*

		Test 1		**Test 2***		**Test 3**	
Date	Actual	Predicted	Error%	Predicted	Error%	Predicted	Error%
03-Jan-12	69.50	69.03	-0.7%	68.96	-0.8%	68.97	-0.8%
04-Jan-12	69.50	69.55	0.1%	69.54	0.1%	69.52	0.0%
05-Jan-12	66.00	69.19	4.8%	68.96	4.5%	69.11	4.7%
06-Jan-12	69.00	67.28	-2.5%	67.13	-2.7%	67.16	-2.7%
09-Jan-12	69.00	68.18	-1.2%	68.12	-1.3%	68.21	-1.1%
10-Jan-12	68.00	69.42	2.1%	69.44	2.1%	69.35	2.0%
11-Jan-12	68.00	67.83	-0.3%	67.94	-0.1%	67.92	-0.1%
12-Jan-12	68.00	68.03	0.0%	67.73	-0.4%	67.98	0.0%
13-Jan-12	68.00	68.08	0.1%	68.17	0.3%	68.17	0.3%
16-Jan-12	67.00	67.98	1.5%	68.04	1.6%	68.06	1.6%
17-Jan-12	67.00	67.26	0.4%	67.39	0.6%	67.42	0.6%
18-Jan-12	67.00	67.03	0.0%	67.15	0.2%	67.18	0.3%
19-Jan-12	68.50	67.27	-1.8%	67.32	-1.7%	67.33	-1.7%
20-Jan-12	72.50	68.24	-5.9%	68.16	-6.0%	68.22	-5.9%
23-Jan-12	72.50	71.88	-0.9%	72.06	-0.6%	72.12	-0.5%
24-Jan-12	70.00	72.63	3.8%	72.83	4.0%	72.54	3.6%
25-Jan-12	74.50	70.29	-5.7%	69.46	-6.8%	69.62	-6.6%
26-Jan-12	74.50	73.89	-0.8%	73.02	-2.0%	73.43	-1.4%
27-Jan-12	74.50	73.97	-0.7%	74.74	0.3%	74.36	-0.2%
30-Jan-12	70.00	73.94	5.6%	73.90	5.6%	73.81	5.4%
31-Jan-12	70.00	71.46	2.1%	70.66	0.9%	70.85	1.2%
01-Feb-12	71.00	70.02	-1.4%	70.28	-1.0%	70.26	-1.0%
02-Feb-12	72.00	71.14	-1.2%	71.53	-0.7%	71.22	-1.1%
03-Feb-12	72.00	71.64	-0.5%	71.83	-0.2%	71.72	-0.4%
06-Feb-12	72.00	71.49	-0.7%	71.63	-0.5%	71.60	-0.6%
07-Feb-12	75.00	71.52	-4.6%	71.49	-4.7%	71.55	-4.6%
08-Feb-12	74.50	74.35	-0.2%	74.34	-0.2%	74.31	-0.3%
09-Feb-12	75.00	74.05	-1.3%	74.59	-0.5%	74.20	-1.1%
10-Feb-12	80.00	74.46	-6.9%	74.34	-7.1%	74.31	-7.1%
13-Feb-12	75.00	80.03	6.7%	79.64	6.2%	79.69	6.3%
14-Feb-12	75.00	75.50	0.7%	75.61	0.8%	74.97	0.0%
15-Feb-12	75.00	74.49	-0.7%	74.01	-1.3%	74.24	-1.0%
16-Feb-12	75.00	74.97	0.0%	75.36	0.5%	75.23	0.3%
17-Feb-12	75.00	75.51	0.7%	75.17	0.2%	75.16	0.2%
20-Feb-12	79.50	74.54	-6.2%	75.01	-5.6%	74.73	-6.0%
21-Feb-12	79.50	79.16	-0.4%	79.12	-0.5%	79.03	-0.6%
22-Feb-12	79.50	79.10	-0.5%	79.47	0.0%	79.30	-0.3%
23-Feb-12	79.50	78.92	-0.7%	79.25	-0.3%	78.94	-0.7%
24-Feb-12	80.00	78.82	-1.5%	79.26	-0.9%	79.04	-1.2%
27-Feb-12	84.00	79.95	-4.8%	79.69	-5.1%	79.75	-5.1%
28-Feb-12	84.00	83.10	-1.1%	83.10	-1.1%	83.54	-0.5%
29-Feb-12	78.00	83.58	7.2%	83.26	6.7%	83.61	7.2%
01-Mar-12	82.00	77.71	-5.2%	78.49	-4.3%	77.87	-5.0%
02-Mar-12	74.00	82.00	10.8%	81.71	10.4%	81.39	10.0%
05-Mar-12	74.00	76.54	3.4%	75.10	1.5%	77.05	4.1%
06-Mar-12	74.00	76.20	3.0%	74.79	1.1%	74.87	1.2%
07-Mar-12	74.00	74.44	0.6%	74.50	0.7%	74.85	1.1%
08-Mar-12	74.00	75.38	1.9%	74.17	0.2%	74.48	0.6%
09-Mar-12	74.00	73.46	-0.7%	73.94	-0.1%	73.71	-0.4%
12-Mar-12	74.00	73.46	-0.7%	73.94	-0.1%	73.71	-0.4%
13-Mar-12	76.00	73.46	-3.3%	73.94	-2.7%	73.71	-3.0%
14-Mar-12	76.00	75.48	-0.7%	75.68	-0.4%	75.44	-0.7%
15-Mar-12	76.00	75.48	-0.7%	76.01	0.0%	75.64	-0.5%
16-Mar-12	76.00	75.56	-0.6%	75.75	-0.3%	75.47	-0.7%
19-Mar-12	76.00	75.62	-0.5%	75.92	-0.1%	75.62	-0.5%
20-Mar-12	74.00	75.64	2.2%	76.05	2.8%	75.74	2.4%
21-Mar-12	74.00	73.91	-0.1%	74.46	0.6%	74.22	0.3%
22-Mar-12	74.00	73.72	-0.4%	74.16	0.2%	73.94	-0.1%
23-Mar-12	74.00	73.69	-0.4%	74.32	0.4%	74.07	0.1%
26-Mar-12	75.00	73.65	-1.8%	74.10	-1.2%	73.86	-1.5%
27-Mar-12	76.00	74.42	-2.1%	74.79	-1.6%	74.54	-1.9%
28-Mar-12	76.00	75.45	-0.7%	75.85	-0.2%	75.55	-0.6%
29-Mar-12	79.50	75.53	-5.0%	75.90	-4.5%	75.56	-5.0%
30-Mar-12	77.50	79.31	2.3%	79.14	2.1%	79.09	2.1%
MAPE			**2.13%**		**1.91%**		**2.01%**
RMSE			**2.36**		**2.26**		**2.31**

Table A11-2: *Stock02 (Standard Bank) results – using Developed Prototype on NSE data*

		Test 1		**Test 2**		**Test 3***	
Date	Actual	Predicted	Error%	Predicted	Error%	Predicted	Error%
03-Jan-12	161.00	160.29	-0.4%	160.23	-0.5%	160.20	-0.5%
04-Jan-12	161.00	161.08	0.0%	161.01	0.0%	161.13	0.1%
05-Jan-12	161.00	161.18	0.1%	161.28	0.2%	161.23	0.1%
06-Jan-12	161.00	160.85	-0.1%	161.00	0.0%	161.15	0.1%
09-Jan-12	161.00	160.81	-0.1%	161.07	0.0%	161.12	0.1%
10-Jan-12	161.00	160.79	-0.1%	161.04	0.0%	161.10	0.1%
11-Jan-12	161.00	160.79	-0.1%	161.04	0.0%	161.10	0.1%
12-Jan-12	161.00	160.79	-0.1%	161.04	0.0%	161.10	0.1%
13-Jan-12	162.00	160.79	-0.7%	161.04	-0.6%	161.10	-0.6%
16-Jan-12	162.00	161.96	0.0%	162.07	0.0%	162.10	0.1%
17-Jan-12	161.00	162.11	0.7%	162.26	0.8%	162.23	0.8%
18-Jan-12	161.00	160.64	-0.2%	160.99	0.0%	161.18	0.1%
19-Jan-12	159.00	160.46	0.9%	160.86	1.2%	161.01	1.3%
20-Jan-12	158.00	158.48	0.3%	158.99	0.6%	159.04	0.7%
23-Jan-12	156.00	157.22	0.8%	157.52	1.0%	157.80	1.2%
24-Jan-12	155.00	155.78	0.5%	155.80	0.5%	155.80	0.5%
25-Jan-12	158.00	154.91	-2.0%	154.52	-2.2%	154.66	-2.1%
26-Jan-12	157.00	158.44	0.9%	157.91	0.6%	157.78	0.5%
27-Jan-12	156.00	158.07	1.3%	157.60	1.0%	157.28	0.8%
30-Jan-12	157.00	156.18	-0.5%	155.60	-0.9%	156.03	-0.6%
31-Jan-12	157.00	157.25	0.2%	156.91	-0.1%	156.92	-0.1%
01-Feb-12	157.00	157.66	0.4%	157.21	0.1%	157.06	0.0%
02-Feb-12	159.00	157.43	-1.0%	156.89	-1.3%	157.04	-1.2%
03-Feb-12	157.00	159.55	1.6%	159.04	1.3%	159.04	1.3%
06-Feb-12	157.00	157.67	0.4%	157.38	0.2%	157.30	0.2%
07-Feb-12	158.00	156.86	-0.7%	156.49	-1.0%	156.89	-0.7%
08-Feb-12	158.00	158.40	0.3%	158.15	0.1%	157.96	0.0%
09-Feb-12	158.00	158.59	0.4%	158.15	0.1%	158.11	0.1%
10-Feb-12	157.00	158.30	0.8%	157.94	0.6%	158.09	0.7%
13-Feb-12	158.00	157.23	-0.5%	157.01	-0.6%	157.07	-0.6%
14-Feb-12	159.00	158.12	-0.6%	157.79	-0.8%	157.91	-0.7%
15-Feb-12	158.00	159.65	1.0%	159.28	0.8%	159.10	0.7%
16-Feb-12	159.00	158.42	-0.4%	158.12	-0.6%	158.20	-0.5%
17-Feb-12	160.00	159.01	-0.6%	158.79	-0.8%	159.00	-0.6%
20-Feb-12	160.00	160.51	0.3%	160.32	0.2%	160.14	0.1%
21-Feb-12	160.00	160.33	0.2%	160.16	0.1%	160.23	0.1%
22-Feb-12	162.00	159.97	-1.3%	160.02	-1.2%	160.16	-1.1%
23-Feb-12	163.00	162.18	-0.5%	162.11	-0.5%	162.09	-0.6%
24-Feb-12	164.00	163.62	-0.2%	163.50	-0.3%	163.32	-0.4%
27-Feb-12	165.00	164.33	-0.4%	164.23	-0.5%	164.31	-0.4%
28-Feb-12	165.00	165.29	0.2%	165.32	0.2%	165.28	0.2%
29-Feb-12	169.00	165.20	-2.2%	165.28	-2.2%	165.31	-2.2%
01-Mar-12	172.00	169.33	-1.6%	169.16	-1.7%	169.02	-1.7%
02-Mar-12	175.00	172.38	-1.5%	172.82	-1.2%	172.20	-1.6%
05-Mar-12	175.00	174.72	-0.2%	175.39	0.2%	175.08	0.0%
06-Mar-12	175.00	174.69	-0.2%	175.51	0.3%	175.27	0.2%
07-Mar-12	177.00	174.61	-1.4%	175.07	-1.1%	175.11	-1.1%
08-Mar-12	178.00	176.63	-0.8%	176.99	-0.6%	176.84	-0.7%
09-Mar-12	176.00	177.93	1.1%	178.24	1.3%	177.97	1.1%
12-Mar-12	175.00	175.90	0.5%	176.08	0.6%	176.07	0.6%
13-Mar-12	175.00	174.52	-0.3%	174.62	-0.2%	174.71	-0.2%
14-Mar-12	174.00	174.74	0.4%	174.69	0.4%	174.57	0.3%
15-Mar-12	169.00	174.08	3.0%	173.81	2.8%	173.78	2.8%
16-Mar-12	172.00	168.80	-1.9%	168.47	-2.1%	168.92	-1.8%
19-Mar-12	173.00	171.71	-0.7%	171.08	-1.1%	171.45	-0.9%
20-Mar-12	171.00	173.47	1.4%	173.65	1.5%	172.98	1.2%
21-Mar-12	170.00	171.52	0.9%	171.02	0.6%	171.30	0.8%
22-Mar-12	163.00	169.61	4.1%	169.72	4.1%	169.95	4.3%
23-Mar-12	165.00	162.33	-1.6%	162.47	-1.5%	163.02	-1.2%
26-Mar-12	169.00	164.09	-2.9%	163.52	-3.2%	164.45	-2.7%
27-Mar-12	169.00	169.81	0.5%	169.95	0.6%	169.02	0.0%
28-Mar-12	172.00	170.14	-1.1%	169.77	-1.3%	169.60	-1.4%
29-Mar-12	171.00	171.95	0.6%	172.03	0.6%	172.05	0.6%
30-Mar-12	174.00	171.15	-1.6%	171.65	-1.4%	171.38	-1.5%
MAPE			**0.819%**		**0.815%**		**0.77%**
RMSE			**1.87**		**1.90**		**1.83**

Table A11-3: *Stock04 (Bamburi Cement) results – using Developed Prototype on NSE data*

		Test 1		**Test 2**		**Test 3***	
Date	Actual	Predicted	Error%	Predicted	Error%	Predicted	Error%
03-Jan-12	125.00	125.82	0.7%	122.81	-1.8%	122.52	-2.0%
04-Jan-12	125.00	125.30	0.2%	123.72	-1.0%	124.22	-0.6%
05-Jan-12	125.00	125.96	0.8%	123.98	-0.8%	125.23	0.2%
06-Jan-12	125.00	122.32	-2.1%	124.40	-0.5%	125.55	0.4%
09-Jan-12	125.00	122.32	-2.1%	124.40	-0.5%	125.55	0.4%
10-Jan-12	125.00	122.32	-2.1%	124.40	-0.5%	125.55	0.4%
11-Jan-12	135.00	122.32	-9.4%	124.40	-7.9%	125.55	-7.0%
12-Jan-12	135.00	136.12	0.8%	132.71	-1.7%	135.80	0.6%
13-Jan-12	133.00	136.28	2.5%	134.68	1.3%	136.95	3.0%
16-Jan-12	135.00	132.20	-2.1%	129.43	-4.1%	133.89	-0.8%
17-Jan-12	133.00	150.49	13.2%	135.78	2.1%	133.22	0.2%
18-Jan-12	133.00	131.80	-0.9%	132.65	-0.3%	132.09	-0.7%
19-Jan-12	130.00	131.26	1.0%	132.32	1.8%	132.18	1.7%
20-Jan-12	132.00	127.79	-3.2%	128.90	-2.3%	128.91	-2.3%
23-Jan-12	131.00	130.59	-0.3%	130.83	-0.1%	131.03	0.0%
24-Jan-12	135.00	130.02	-3.7%	130.42	-3.4%	130.44	-3.4%
25-Jan-12	135.00	132.31	-2.0%	135.50	0.4%	135.07	0.1%
26-Jan-12	130.00	136.76	5.2%	136.65	5.1%	135.00	3.8%
27-Jan-12	130.00	128.34	-1.3%	130.72	0.6%	129.88	-0.1%
30-Jan-12	135.00	128.70	-4.7%	128.18	-5.1%	128.33	-4.9%
31-Jan-12	135.00	146.17	8.3%	135.33	0.2%	133.95	-0.8%
01-Feb-12	130.00	133.59	2.8%	136.55	5.0%	135.97	4.6%
02-Feb-12	130.00	129.29	-0.5%	131.14	0.9%	130.03	0.0%
03-Feb-12	130.00	130.74	0.6%	128.27	-1.3%	128.28	-1.3%
06-Feb-12	130.00	128.73	-1.0%	128.60	-1.1%	128.93	-0.8%
07-Feb-12	128.00	127.87	-0.1%	129.29	1.0%	130.15	1.7%
08-Feb-12	130.00	125.33	-3.6%	127.15	-2.2%	128.05	-1.5%
09-Feb-12	130.00	129.51	-0.4%	129.15	-0.7%	129.71	-0.2%
10-Feb-12	129.00	129.82	0.6%	129.94	0.7%	130.09	0.8%
13-Feb-12	125.00	128.26	2.6%	128.78	3.0%	129.48	3.6%
14-Feb-12	125.00	121.19	-3.0%	123.99	-0.8%	124.78	-0.2%
15-Feb-12	130.00	120.95	-7.0%	122.94	-5.4%	124.33	-4.4%
16-Feb-12	130.00	132.76	2.1%	130.61	0.5%	130.08	0.1%
17-Feb-12	127.00	130.41	2.7%	132.03	4.0%	131.51	3.6%
20-Feb-12	127.00	126.97	0.0%	127.03	0.0%	127.94	0.7%
21-Feb-12	130.00	125.36	-3.6%	125.75	-3.3%	126.24	-2.9%
22-Feb-12	130.00	130.65	0.5%	129.77	-0.2%	129.86	-0.1%
23-Feb-12	130.00	130.13	0.1%	130.91	0.7%	130.91	0.7%
24-Feb-12	130.00	130.56	0.4%	130.88	0.7%	130.51	0.4%
27-Feb-12	140.00	130.41	-6.9%	130.04	-7.1%	129.78	-7.3%
28-Feb-12	140.00	137.46	-1.8%	136.91	-2.2%	140.68	0.5%
29-Feb-12	145.00	139.98	-3.5%	140.50	-3.1%	141.74	-2.2%
01-Mar-12	154.00	141.45	-8.1%	135.78	-11.8%	145.98	-5.2%
02-Mar-12	154.00	167.81	9.0%	143.83	-6.6%	153.77	-0.1%
05-Mar-12	155.00	148.26	-4.3%	140.86	-9.1%	154.12	-0.6%
06-Mar-12	155.00	152.28	-1.8%	148.31	-4.3%	154.56	-0.3%
07-Mar-12	157.00	152.23	-3.0%	155.02	-1.3%	154.56	-1.6%
08-Mar-12	157.00	155.89	-0.7%	156.31	-0.4%	156.54	-0.3%
09-Mar-12	157.00	155.89	-0.7%	156.71	-0.2%	156.64	-0.2%
12-Mar-12	153.00	156.48	2.3%	156.84	2.5%	156.65	2.4%
13-Mar-12	150.00	152.33	1.6%	153.74	2.5%	153.52	2.3%
14-Mar-12	150.00	150.20	0.1%	150.16	0.1%	151.03	0.7%
15-Mar-12	155.00	148.96	-3.9%	150.05	-3.2%	150.90	-2.6%
16-Mar-12	155.00	158.25	2.1%	155.60	0.4%	154.78	-0.1%
19-Mar-12	155.00	153.65	-0.9%	154.85	-0.1%	154.83	-0.1%
20-Mar-12	155.00	154.67	-0.2%	155.29	0.2%	154.66	-0.2%
21-Mar-12	155.00	153.09	-1.2%	154.87	-0.1%	154.73	-0.2%
22-Mar-12	155.00	155.10	0.1%	154.74	-0.2%	154.90	-0.1%
23-Mar-12	155.00	155.10	0.1%	154.74	-0.2%	154.90	-0.1%
26-Mar-12	155.00	155.10	0.1%	154.74	-0.2%	154.90	-0.1%
27-Mar-12	153.00	155.10	1.4%	154.74	1.1%	154.90	1.2%
28-Mar-12	153.00	153.33	0.2%	153.17	0.1%	153.28	0.2%
29-Mar-12	150.00	154.22	2.8%	152.65	1.8%	153.17	2.1%
30-Mar-12	150.00	149.57	-0.3%	150.10	0.1%	150.86	0.6%
MAPE			**2.42%**		**2.06%**		**1.44%**
RMSE			**5.01**		**4.58**		**3.07**

APPENDIX 12 – Test Results Using Neuroph on NSE data

These were the test results obtained by using Neuroph on NSE data (best 3 results):

Table A12-1: *Stock01 (Kakuzi) results – using Neuroph on NSE data*

		Test 1		**Test 2***		**Test 3**	
Date	Actual	Predicted	Error%	Predicted	Error%	Predicted	Error%
03-Jan-12	69.50	68.97	-0.8%	69.01	-0.7%	68.72	-1.1%
04-Jan-12	69.50	69.82	0.5%	69.73	0.3%	69.43	-0.1%
05-Jan-12	66.00	69.71	5.6%	69.62	5.5%	69.32	5.0%
06-Jan-12	69.00	66.77	-3.2%	66.75	-3.3%	66.22	-4.0%
09-Jan-12	69.00	68.38	-0.9%	68.43	-0.8%	68.11	-1.3%
10-Jan-12	68.00	69.18	1.7%	69.18	1.7%	68.86	1.3%
11-Jan-12	68.00	68.00	0.0%	68.03	0.0%	67.61	-0.6%
12-Jan-12	68.00	68.27	0.4%	68.20	0.3%	67.81	-0.3%
13-Jan-12	68.00	68.02	0.0%	68.04	0.1%	67.65	-0.5%
16-Jan-12	67.00	67.97	1.4%	68.00	1.5%	67.60	0.9%
17-Jan-12	67.00	67.19	0.3%	67.21	0.3%	66.76	-0.4%
18-Jan-12	67.00	66.98	0.0%	67.02	0.0%	66.57	-0.6%
19-Jan-12	68.50	67.02	-2.2%	67.05	-2.1%	66.61	-2.8%
20-Jan-12	72.50	68.28	-5.8%	68.28	-5.8%	67.93	-6.3%
23-Jan-12	72.50	72.34	-0.2%	72.20	-0.4%	72.12	-0.5%
24-Jan-12	70.00	73.18	4.5%	72.98	4.3%	72.86	4.1%
25-Jan-12	74.50	70.79	-5.0%	70.64	-5.2%	70.35	-5.6%
26-Jan-12	74.50	74.41	-0.1%	74.26	-0.3%	74.29	-0.3%
27-Jan-12	74.50	75.13	0.9%	75.03	0.7%	75.01	0.7%
30-Jan-12	70.00	74.81	6.9%	74.72	6.7%	74.68	6.7%
31-Jan-12	70.00	71.17	1.7%	71.08	1.5%	70.76	1.1%
01-Feb-12	71.00	69.71	-1.8%	69.83	-1.6%	69.52	-2.1%
02-Feb-12	72.00	70.80	-1.7%	70.91	-1.5%	70.68	-1.8%
03-Feb-12	72.00	71.71	-0.4%	71.78	-0.3%	71.61	-0.5%
06-Feb-12	72.00	72.35	0.5%	72.26	0.4%	72.10	0.1%
07-Feb-12	75.00	72.36	-3.5%	72.28	-3.6%	72.11	-3.9%
08-Feb-12	74.50	75.06	0.7%	74.95	0.6%	74.98	0.6%
09-Feb-12	75.00	75.18	0.2%	75.07	0.1%	75.03	0.0%
10-Feb-12	80.00	75.39	-5.8%	75.29	-5.9%	75.28	-5.9%
13-Feb-12	75.00	79.97	6.6%	79.80	6.4%	80.04	6.7%
14-Feb-12	75.00	76.45	1.9%	76.35	1.8%	76.28	1.7%
15-Feb-12	75.00	75.20	0.3%	75.16	0.2%	75.13	0.2%
16-Feb-12	75.00	75.62	0.8%	75.59	0.8%	75.57	0.8%
17-Feb-12	75.00	74.86	-0.2%	74.99	0.0%	74.96	-0.1%
20-Feb-12	79.50	75.36	-5.2%	75.33	-5.2%	75.32	-5.3%
21-Feb-12	79.50	79.27	-0.3%	79.19	-0.4%	79.40	-0.1%
22-Feb-12	79.50	80.17	0.8%	80.04	0.7%	80.20	0.9%
23-Feb-12	79.50	80.01	0.6%	79.87	0.5%	80.03	0.7%
24-Feb-12	80.00	80.21	0.3%	80.07	0.1%	80.23	0.3%
27-Feb-12	84.00	80.21	-4.5%	80.20	-4.5%	80.36	-4.3%
28-Feb-12	84.00	83.23	-0.9%	83.17	-1.0%	83.45	-0.7%
29-Feb-12	78.00	83.85	7.5%	83.75	7.4%	83.99	7.7%
01-Mar-12	82.00	79.35	-3.2%	79.30	-3.3%	79.32	-3.3%
02-Mar-12	74.00	81.46	10.1%	81.45	10.1%	81.70	10.4%
05-Mar-12	74.00	75.46	2.0%	75.62	2.2%	75.45	2.0%
06-Mar-12	74.00	73.17	-1.1%	73.47	-0.7%	73.34	-0.9%
07-Mar-12	74.00	74.33	0.4%	74.42	0.6%	74.35	0.5%
08-Mar-12	74.00	73.51	-0.7%	73.74	-0.4%	73.64	-0.5%
09-Mar-12	74.00	74.32	0.4%	74.29	0.4%	74.22	0.3%
12-Mar-12	74.00	74.32	0.4%	74.29	0.4%	74.22	0.3%
13-Mar-12	76.00	74.32	-2.2%	74.29	-2.2%	74.22	-2.3%
14-Mar-12	76.00	76.13	0.2%	76.07	0.1%	76.13	0.2%
15-Mar-12	76.00	76.57	0.7%	76.49	0.6%	76.52	0.7%
16-Mar-12	76.00	76.49	0.6%	76.40	0.5%	76.43	0.6%
19-Mar-12	76.00	76.59	0.8%	76.50	0.7%	76.53	0.7%
20-Mar-12	74.00	76.39	3.2%	76.36	3.2%	76.40	3.2%
21-Mar-12	74.00	74.59	0.8%	74.59	0.8%	74.50	0.7%
22-Mar-12	74.00	74.14	0.2%	74.17	0.2%	74.09	0.1%
23-Mar-12	74.00	74.22	0.3%	74.25	0.3%	74.18	0.2%
26-Mar-12	75.00	74.12	-1.2%	74.15	-1.1%	74.08	-1.2%
27-Mar-12	76.00	75.22	-1.0%	75.19	-1.1%	75.18	-1.1%
28-Mar-12	76.00	76.35	0.5%	76.28	0.4%	76.33	0.4%
29-Mar-12	79.50	76.52	-3.7%	76.44	-3.8%	76.48	-3.8%
30-Mar-12	77.50	79.53	2.6%	79.40	2.5%	79.61	2.7%
MAPE			**1.93%**		**1.88%**		**1.95%**
RMSE			**2.19**		**2.18**		**2.23**

Table A12-2: *Stock02 (Standard Bank) results – using Neuroph on NSE data*

		Test 1		**Test 2***		**Test 3**	
Date	Actual	Predicted	Error%	Predicted	Error%	Predicted	Error%
03-Jan-12	161.00	159.65	-0.8%	160.26	-0.5%	160.87	-0.1%
04-Jan-12	161.00	160.30	-0.4%	161.06	0.0%	161.69	0.4%
05-Jan-12	161.00	160.66	-0.2%	161.30	0.2%	161.71	0.4%
06-Jan-12	161.00	160.75	-0.2%	161.14	0.1%	161.64	0.4%
09-Jan-12	161.00	160.83	-0.1%	161.14	0.1%	161.64	0.4%
10-Jan-12	161.00	160.68	-0.2%	160.97	0.0%	161.57	0.4%
11-Jan-12	161.00	160.68	-0.2%	160.97	0.0%	161.57	0.4%
12-Jan-12	161.00	160.68	-0.2%	160.97	0.0%	161.57	0.4%
13-Jan-12	162.00	160.68	-0.8%	160.97	-0.6%	161.57	-0.3%
16-Jan-12	162.00	161.14	-0.5%	161.74	-0.2%	162.40	0.2%
17-Jan-12	161.00	161.42	0.3%	161.97	0.6%	162.41	0.9%
18-Jan-12	161.00	161.19	0.1%	161.23	0.1%	161.59	0.4%
19-Jan-12	159.00	161.00	1.3%	161.00	1.3%	161.57	1.6%
20-Jan-12	158.00	159.75	1.1%	159.28	0.8%	159.82	1.2%
23-Jan-12	156.00	158.75	1.8%	158.03	1.3%	158.91	1.9%
24-Jan-12	155.00	157.39	1.5%	156.27	0.8%	157.15	1.4%
25-Jan-12	158.00	156.15	-1.2%	154.80	-2.0%	156.14	-1.2%
26-Jan-12	157.00	156.88	-0.1%	157.50	0.3%	159.06	1.3%
27-Jan-12	156.00	156.96	0.6%	157.55	1.0%	158.30	1.5%
30-Jan-12	157.00	157.13	0.1%	156.91	-0.1%	157.53	0.3%
31-Jan-12	157.00	157.44	0.3%	157.68	0.4%	158.51	1.0%
01-Feb-12	157.00	157.00	0.0%	157.34	0.2%	158.29	0.8%
02-Feb-12	159.00	157.27	-1.1%	157.55	-0.9%	158.37	-0.4%
03-Feb-12	157.00	158.29	0.8%	159.28	1.5%	160.18	2.0%
06-Feb-12	157.00	157.82	0.5%	158.03	0.7%	158.41	0.9%
07-Feb-12	158.00	157.77	-0.1%	157.60	-0.3%	158.37	0.2%
08-Feb-12	158.00	157.91	-0.1%	158.36	0.2%	159.25	0.8%
09-Feb-12	158.00	157.72	-0.2%	158.22	0.1%	159.11	0.7%
10-Feb-12	157.00	158.20	0.8%	158.60	1.0%	159.27	1.4%
13-Feb-12	158.00	157.87	-0.1%	157.82	-0.1%	158.39	0.2%
14-Feb-12	159.00	157.91	-0.7%	158.20	-0.5%	159.18	0.1%
15-Feb-12	158.00	158.34	0.2%	159.18	0.7%	160.06	1.3%
16-Feb-12	159.00	158.30	-0.4%	158.65	-0.2%	159.22	0.1%
17-Feb-12	160.00	158.91	-0.7%	159.39	-0.4%	160.13	0.1%
20-Feb-12	160.00	159.30	-0.4%	160.20	0.1%	160.92	0.6%
21-Feb-12	160.00	159.56	-0.3%	160.26	0.2%	160.87	0.5%
22-Feb-12	162.00	160.01	-1.2%	160.47	-0.9%	160.94	-0.7%
23-Feb-12	163.00	160.83	-1.3%	161.83	-0.7%	162.54	-0.3%
24-Feb-12	164.00	161.69	-1.4%	162.91	-0.7%	163.34	-0.4%
27-Feb-12	165.00	162.98	-1.2%	164.01	-0.6%	164.20	-0.5%
28-Feb-12	165.00	164.26	-0.5%	165.15	0.1%	165.08	0.0%
29-Feb-12	169.00	164.62	-2.6%	165.06	-2.3%	164.94	-2.4%
01-Mar-12	172.00	167.01	-2.9%	168.54	-2.0%	168.52	-2.0%
02-Mar-12	175.00	170.03	-2.8%	172.53	-1.4%	171.50	-2.0%
05-Mar-12	175.00	173.82	-0.7%	176.45	0.8%	174.70	-0.2%
06-Mar-12	175.00	176.12	0.6%	177.50	1.4%	174.76	-0.1%
07-Mar-12	177.00	176.59	-0.2%	176.67	-0.2%	174.37	-1.5%
08-Mar-12	178.00	177.43	-0.3%	178.07	0.0%	176.33	-0.9%
09-Mar-12	176.00	178.07	1.2%	179.00	1.7%	177.22	0.7%
12-Mar-12	175.00	177.89	1.7%	177.31	1.3%	174.97	0.0%
13-Mar-12	175.00	177.22	1.3%	175.71	0.4%	173.82	-0.7%
14-Mar-12	174.00	176.09	1.2%	174.88	0.5%	173.61	-0.2%
15-Mar-12	169.00	174.87	3.5%	173.56	2.7%	172.44	2.0%
16-Mar-12	172.00	171.94	0.0%	168.66	-1.9%	167.63	-2.5%
19-Mar-12	173.00	171.84	-0.7%	170.34	-1.5%	170.52	-1.4%
20-Mar-12	171.00	171.81	0.5%	172.12	0.7%	171.60	0.4%
21-Mar-12	170.00	171.55	0.9%	170.65	0.4%	169.69	-0.2%
22-Mar-12	163.00	171.86	5.4%	170.24	4.4%	169.09	3.7%
23-Mar-12	165.00	166.44	0.9%	162.98	-1.2%	162.59	-1.5%
26-Mar-12	169.00	164.58	-2.6%	162.71	-3.7%	164.08	-2.9%
27-Mar-12	169.00	165.63	-2.0%	166.84	-1.3%	167.85	-0.7%
28-Mar-12	172.00	166.97	-2.9%	168.16	-2.2%	168.02	-2.3%
29-Mar-12	171.00	171.44	0.3%	172.87	1.1%	171.58	0.3%
30-Mar-12	174.00	171.98	-1.2%	172.34	-1.0%	170.45	-2.0%
MAPE			**0.94%**		**0.86%**		**0.92%**
RMSE			**2.28**		**2.05**		**2.03**

Table A12-3: *Stock04 (Bamburi Cement) results – using Neuroph on NSE data*

		Test 1		**Test 2**		**Test 3***	
Date	Actual	Predicted	Error%	Predicted	Error%	Predicted	Error%
03-Jan-12	125.00	125.71	0.6%	125.34	0.3%	126.83	1.5%
04-Jan-12	125.00	125.65	0.5%	125.26	0.2%	126.68	1.3%
05-Jan-12	125.00	125.30	0.2%	124.86	-0.1%	126.15	0.9%
06-Jan-12	125.00	125.18	0.1%	124.73	-0.2%	126.00	0.8%
09-Jan-12	125.00	125.18	0.1%	124.73	-0.2%	126.00	0.8%
10-Jan-12	125.00	125.18	0.1%	124.73	-0.2%	126.00	0.8%
11-Jan-12	135.00	125.18	-7.3%	124.73	-7.6%	126.00	-6.7%
12-Jan-12	135.00	132.85	-1.6%	133.29	-1.3%	133.05	-1.4%
13-Jan-12	133.00	132.94	0.0%	133.44	0.3%	133.42	0.3%
16-Jan-12	135.00	131.14	-2.9%	131.44	-2.6%	132.05	-2.2%
17-Jan-12	133.00	133.34	0.3%	133.90	0.7%	133.82	0.6%
18-Jan-12	133.00	131.66	-1.0%	132.05	-0.7%	132.55	-0.3%
19-Jan-12	130.00	131.56	1.2%	131.93	1.5%	132.41	1.9%
20-Jan-12	132.00	129.15	-2.2%	129.21	-2.1%	130.42	-1.2%
23-Jan-12	131.00	130.68	-0.2%	130.93	-0.1%	131.60	0.5%
24-Jan-12	135.00	129.85	-3.8%	130.00	-3.7%	130.96	-3.0%
25-Jan-12	135.00	133.23	-1.3%	133.76	-0.9%	133.62	-1.0%
26-Jan-12	130.00	133.30	2.5%	133.86	3.0%	133.81	2.9%
27-Jan-12	130.00	129.02	-0.8%	129.05	-0.7%	130.34	0.3%
30-Jan-12	135.00	129.10	-4.4%	129.15	-4.3%	130.31	-3.5%
31-Jan-12	135.00	133.46	-1.1%	134.02	-0.7%	133.80	-0.9%
01-Feb-12	130.00	133.29	2.5%	133.85	3.0%	133.77	2.9%
02-Feb-12	130.00	128.96	-0.8%	128.99	-0.8%	130.27	0.2%
03-Feb-12	130.00	129.09	-0.7%	129.13	-0.7%	130.30	0.2%
06-Feb-12	130.00	129.15	-0.7%	129.20	-0.6%	130.33	0.3%
07-Feb-12	128.00	128.98	0.8%	129.00	0.8%	130.14	1.7%
08-Feb-12	130.00	127.47	-1.9%	127.28	-2.1%	128.74	-1.0%
09-Feb-12	130.00	128.90	-0.8%	128.90	-0.8%	130.01	0.0%
10-Feb-12	129.00	128.92	-0.1%	128.92	-0.1%	130.08	0.8%
13-Feb-12	125.00	128.12	2.5%	128.01	2.4%	129.34	3.5%
14-Feb-12	125.00	125.39	0.3%	124.98	0.0%	126.47	1.2%
15-Feb-12	130.00	125.39	-3.5%	124.96	-3.9%	126.33	-2.8%
16-Feb-12	130.00	128.85	-0.9%	128.82	-0.9%	129.85	-0.1%
17-Feb-12	127.00	128.75	1.4%	128.72	1.4%	129.86	2.3%
20-Feb-12	127.00	126.58	-0.3%	126.26	-0.6%	127.78	0.6%
21-Feb-12	130.00	126.71	-2.5%	126.42	-2.8%	127.87	-1.6%
22-Feb-12	130.00	128.90	-0.8%	128.89	-0.9%	129.97	0.0%
23-Feb-12	130.00	128.82	-0.9%	128.80	-0.9%	129.96	0.0%
24-Feb-12	130.00	128.79	-0.9%	128.77	-0.9%	129.94	0.0%
27-Feb-12	140.00	128.89	-7.9%	128.89	-7.9%	130.05	-7.1%
28-Feb-12	140.00	138.36	-1.2%	139.22	-0.6%	137.59	-1.7%
29-Feb-12	145.00	138.48	-4.5%	139.38	-3.9%	138.04	-4.8%
01-Mar-12	154.00	144.17	-6.4%	144.99	-5.9%	142.93	-7.2%
02-Mar-12	154.00	153.93	0.0%	153.82	-0.1%	152.68	-0.9%
05-Mar-12	155.00	154.15	-0.6%	154.05	-0.6%	153.30	-1.1%
06-Mar-12	155.00	155.19	0.1%	155.01	0.0%	154.52	-0.3%
07-Mar-12	157.00	155.61	-0.9%	155.40	-1.0%	155.08	-1.2%
08-Mar-12	157.00	157.35	0.2%	157.01	0.0%	156.97	0.0%
09-Mar-12	157.00	157.40	0.3%	157.07	0.0%	157.10	0.1%
12-Mar-12	153.00	157.42	2.9%	157.08	2.7%	157.13	2.7%
13-Mar-12	150.00	154.15	2.8%	154.11	2.7%	153.70	2.5%
14-Mar-12	150.00	151.35	0.9%	151.64	1.1%	150.72	0.5%
15-Mar-12	155.00	151.33	-2.4%	151.61	-2.2%	150.54	-2.9%
16-Mar-12	155.00	155.67	0.4%	155.45	0.3%	154.96	0.0%
19-Mar-12	155.00	155.56	0.4%	155.34	0.2%	154.97	0.0%
20-Mar-12	155.00	155.52	0.3%	155.31	0.2%	154.97	0.0%
21-Mar-12	155.00	155.70	0.5%	155.50	0.3%	155.21	0.1%
22-Mar-12	155.00	155.77	0.5%	155.55	0.4%	155.29	0.2%
23-Mar-12	155.00	155.77	0.5%	155.55	0.4%	155.29	0.2%
26-Mar-12	155.00	155.77	0.5%	155.55	0.4%	155.29	0.2%
27-Mar-12	153.00	155.77	1.8%	155.55	1.7%	155.29	1.5%
28-Mar-12	153.00	154.05	0.7%	154.01	0.7%	153.49	0.3%
29-Mar-12	150.00	154.03	2.7%	153.99	2.7%	153.39	2.3%
30-Mar-12	150.00	151.25	0.8%	151.53	1.0%	150.55	0.4%
MAPE			**1.48%**		**1.43%**		**1.14%**
RMSE			**3.14**		**3.09**		**3.04**

APPENDIX 13 – Test Results Using Encog on NSE data

These were the test results obtained by using Encog on NSE data (best 3 results):

Table A13-1: *Stock01 (Kakuzi) results – using Encog on NSE data*

		Test 1		**Test 2***		**Test 3**	
Date	Actual	Predicted	Error%	Predicted	Error%	Predicted	Error%
03-Jan-12	69.50	69.31	-0.3%	69.22	-0.4%	68.89	-0.9%
04-Jan-12	69.50	69.96	0.7%	69.80	0.4%	68.95	-0.8%
05-Jan-12	66.00	69.94	6.0%	69.81	5.8%	68.99	4.5%
06-Jan-12	69.00	65.98	-4.4%	66.79	-3.2%	65.94	-4.4%
09-Jan-12	69.00	68.57	-0.6%	68.68	-0.5%	68.42	-0.8%
10-Jan-12	68.00	69.03	1.5%	69.15	1.7%	68.55	0.8%
11-Jan-12	68.00	67.98	0.0%	68.28	0.4%	67.68	-0.5%
12-Jan-12	68.00	68.10	0.1%	68.33	0.5%	67.55	-0.7%
13-Jan-12	68.00	67.81	-0.3%	68.16	0.2%	67.62	-0.6%
16-Jan-12	67.00	67.84	1.3%	68.16	1.7%	67.61	0.9%
17-Jan-12	67.00	66.82	-0.3%	67.36	0.5%	66.71	-0.4%
18-Jan-12	67.00	66.68	-0.5%	67.20	0.3%	66.65	-0.5%
19-Jan-12	68.50	66.68	-2.7%	67.20	-1.9%	66.64	-2.7%
20-Jan-12	72.50	68.38	-5.7%	68.49	-5.5%	67.94	-6.3%
23-Jan-12	72.50	73.13	0.9%	72.16	-0.5%	71.65	-1.2%
24-Jan-12	70.00	73.67	5.2%	72.77	4.0%	71.92	2.7%
25-Jan-12	74.50	70.88	-4.9%	70.68	-5.1%	69.67	-6.5%
26-Jan-12	74.50	75.14	0.9%	73.91	-0.8%	73.81	-0.9%
27-Jan-12	74.50	75.40	1.2%	74.47	0.0%	74.18	-0.4%
30-Jan-12	70.00	75.42	7.7%	74.40	6.3%	74.23	6.0%
31-Jan-12	70.00	70.71	1.0%	70.96	1.4%	69.96	-0.1%
01-Feb-12	71.00	69.62	-1.9%	70.25	-1.1%	69.81	-1.7%
02-Feb-12	72.00	70.77	-1.7%	71.07	-1.3%	70.67	-1.9%
03-Feb-12	72.00	72.12	0.2%	71.82	-0.2%	71.62	-0.5%
06-Feb-12	72.00	72.69	1.0%	72.11	0.1%	71.57	-0.6%
07-Feb-12	75.00	72.66	-3.1%	72.12	-3.8%	71.59	-4.5%
08-Feb-12	74.50	75.71	1.6%	74.51	0.0%	74.48	0.0%
09-Feb-12	75.00	75.50	0.7%	74.53	-0.6%	74.22	-1.0%
10-Feb-12	80.00	75.93	-5.1%	74.84	-6.4%	74.72	-6.6%
13-Feb-12	75.00	80.51	7.3%	79.22	5.6%	79.55	6.1%
14-Feb-12	75.00	76.32	1.8%	75.69	0.9%	75.18	0.2%
15-Feb-12	75.00	75.67	0.9%	74.96	-0.1%	74.92	-0.1%
16-Feb-12	75.00	75.55	0.7%	75.26	0.3%	74.91	-0.1%
17-Feb-12	75.00	75.21	0.3%	75.07	0.1%	75.02	0.0%
20-Feb-12	79.50	75.66	-4.8%	74.99	-5.7%	74.86	-5.8%
21-Feb-12	79.50	79.87	0.5%	78.69	-1.0%	79.18	-0.4%
22-Feb-12	79.50	80.36	1.1%	79.27	-0.3%	79.46	-0.1%
23-Feb-12	79.50	80.34	1.1%	79.13	-0.5%	79.52	0.0%
24-Feb-12	80.00	80.25	0.3%	79.28	-0.9%	79.56	-0.6%
27-Feb-12	84.00	80.34	-4.4%	79.74	-5.1%	80.15	-4.6%
28-Feb-12	84.00	83.33	-0.8%	83.46	-0.6%	83.66	-0.4%
29-Feb-12	78.00	83.66	7.3%	83.87	7.5%	83.88	7.5%
01-Mar-12	82.00	79.09	-3.5%	78.68	-4.1%	78.63	-4.1%
02-Mar-12	74.00	81.63	10.3%	81.27	9.8%	81.97	10.8%
05-Mar-12	74.00	74.52	0.7%	76.29	3.1%	74.74	1.0%
06-Mar-12	74.00	73.53	-0.6%	74.54	0.7%	74.23	0.3%
07-Mar-12	74.00	74.06	0.1%	74.64	0.9%	73.97	0.0%
08-Mar-12	74.00	73.87	-0.2%	74.17	0.2%	74.03	0.0%
09-Mar-12	74.00	74.61	0.8%	74.03	0.0%	73.79	-0.3%
12-Mar-12	74.00	74.61	0.8%	74.03	0.0%	73.79	-0.3%
13-Mar-12	76.00	74.61	-1.8%	74.03	-2.6%	73.79	-2.9%
14-Mar-12	76.00	76.65	0.8%	75.59	-0.5%	75.71	-0.4%
15-Mar-12	76.00	76.90	1.2%	75.88	-0.2%	75.84	-0.2%
16-Mar-12	76.00	76.89	1.2%	75.86	-0.2%	75.87	-0.2%
19-Mar-12	76.00	76.84	1.1%	75.94	-0.1%	75.89	-0.2%
20-Mar-12	74.00	76.67	3.6%	75.95	2.6%	75.95	2.6%
21-Mar-12	74.00	74.64	0.9%	74.43	0.6%	74.02	0.0%
22-Mar-12	74.00	74.37	0.5%	74.13	0.2%	73.89	-0.1%
23-Mar-12	74.00	74.38	0.5%	74.14	0.2%	73.86	-0.2%
26-Mar-12	75.00	74.43	-0.8%	74.03	-1.3%	73.85	-1.5%
27-Mar-12	76.00	75.64	-0.5%	74.81	-1.6%	74.75	-1.6%
28-Mar-12	76.00	76.77	1.0%	75.74	-0.3%	75.78	-0.3%
29-Mar-12	79.50	76.89	-3.3%	75.87	-4.6%	75.85	-4.6%
30-Mar-12	77.50	80.05	3.3%	78.80	1.7%	79.22	2.2%
MAPE			**2.03%**		**1.86%**		**1.87%**
RMSE			**2.25**		**2.19**		**2.27**

Table A13-2: *Stock02 (Standard Bank) results – using Encog on NSE data*

		Test 1		**Test 2***		**Test 3**	
Date	Actual	Predicted	Error%	Predicted	Error%	Predicted	Error%
03-Jan-12	161.00	161.81	0.5%	160.62	-0.2%	160.99	0.0%
04-Jan-12	161.00	161.81	0.5%	161.57	0.4%	161.72	0.4%
05-Jan-12	161.00	161.81	0.5%	161.60	0.4%	161.80	0.5%
06-Jan-12	161.00	161.81	0.5%	161.49	0.3%	161.73	0.5%
09-Jan-12	161.00	162.70	1.1%	161.50	0.3%	161.72	0.4%
10-Jan-12	161.00	162.82	1.1%	161.50	0.3%	161.65	0.4%
11-Jan-12	161.00	161.93	0.6%	161.50	0.3%	161.65	0.4%
12-Jan-12	161.00	161.79	0.5%	161.50	0.3%	161.65	0.4%
13-Jan-12	162.00	159.80	-1.4%	161.50	-0.3%	161.65	-0.2%
16-Jan-12	162.00	158.50	-2.2%	162.53	0.3%	162.38	0.2%
17-Jan-12	161.00	156.21	-3.0%	162.56	1.0%	162.48	0.9%
18-Jan-12	161.00	154.73	-3.9%	161.42	0.3%	161.74	0.5%
19-Jan-12	159.00	158.28	-0.4%	161.40	1.5%	161.63	1.7%
20-Jan-12	158.00	157.80	-0.1%	159.45	0.9%	160.07	1.3%
23-Jan-12	156.00	156.73	0.5%	158.35	1.5%	159.09	2.0%
24-Jan-12	155.00	157.74	1.8%	156.41	0.9%	157.34	1.5%
25-Jan-12	158.00	157.62	-0.2%	155.38	-1.7%	156.18	-1.2%
26-Jan-12	157.00	157.74	0.5%	158.72	1.1%	158.97	1.3%
27-Jan-12	156.00	159.91	2.5%	157.74	1.1%	158.57	1.6%
30-Jan-12	157.00	158.02	0.6%	156.14	-0.5%	157.75	0.5%
31-Jan-12	157.00	157.70	0.4%	157.40	0.3%	158.53	1.0%
01-Feb-12	157.00	158.75	1.1%	157.63	0.4%	158.44	0.9%
02-Feb-12	159.00	158.75	-0.2%	157.45	-1.0%	158.53	-0.3%
03-Feb-12	157.00	158.94	1.2%	159.64	1.7%	160.18	2.0%
06-Feb-12	157.00	157.84	0.5%	157.50	0.3%	158.72	1.1%
07-Feb-12	158.00	158.67	0.4%	157.22	-0.5%	158.50	0.3%
08-Feb-12	158.00	159.85	1.2%	158.60	0.4%	159.30	0.8%
09-Feb-12	158.00	158.99	0.6%	158.64	0.4%	159.29	0.8%
10-Feb-12	157.00	159.93	1.9%	158.46	0.9%	159.43	1.5%
13-Feb-12	158.00	160.93	1.9%	157.38	-0.4%	158.59	0.4%
14-Feb-12	159.00	161.00	1.3%	158.49	-0.3%	159.24	0.1%
15-Feb-12	158.00	161.06	1.9%	159.71	1.1%	160.13	1.3%
16-Feb-12	159.00	162.76	2.4%	158.51	-0.3%	159.46	0.3%
17-Feb-12	160.00	163.76	2.3%	159.45	-0.3%	160.19	0.1%
20-Feb-12	160.00	164.68	2.9%	160.69	0.4%	160.96	0.6%
21-Feb-12	160.00	165.55	3.5%	160.61	0.4%	161.01	0.6%
22-Feb-12	162.00	165.50	2.2%	160.49	-0.9%	161.05	-0.6%
23-Feb-12	163.00	168.57	3.4%	162.61	-0.2%	162.44	-0.3%
24-Feb-12	164.00	171.47	4.6%	163.70	-0.2%	163.29	-0.4%
27-Feb-12	165.00	174.54	5.8%	164.55	-0.3%	164.11	-0.5%
28-Feb-12	165.00	174.85	6.0%	165.47	0.3%	164.93	0.0%
29-Feb-12	169.00	174.42	3.2%	165.34	-2.2%	164.87	-2.4%
01-Mar-12	172.00	176.07	2.4%	169.03	-1.7%	168.02	-2.3%
02-Mar-12	175.00	177.11	1.2%	171.86	-1.8%	171.17	-2.2%
05-Mar-12	175.00	175.20	0.1%	174.43	-0.3%	174.54	-0.3%
06-Mar-12	175.00	173.91	-0.6%	174.14	-0.5%	174.84	-0.1%
07-Mar-12	177.00	173.59	-1.9%	173.43	-2.0%	174.37	-1.5%
08-Mar-12	178.00	172.62	-3.0%	174.79	-1.8%	176.19	-1.0%
09-Mar-12	176.00	168.40	-4.3%	175.57	-0.2%	177.31	0.7%
12-Mar-12	175.00	170.38	-2.6%	173.70	-0.7%	175.22	0.1%
13-Mar-12	175.00	171.65	-1.9%	172.59	-1.4%	173.81	-0.7%
14-Mar-12	174.00	170.21	-2.2%	172.48	-0.9%	173.48	-0.3%
15-Mar-12	169.00	169.46	0.3%	171.74	1.6%	172.40	2.0%
16-Mar-12	172.00	163.50	-4.9%	167.73	-2.5%	167.78	-2.5%
19-Mar-12	173.00	164.31	-5.0%	170.12	-1.7%	169.97	-1.7%
20-Mar-12	171.00	167.89	-1.8%	171.40	0.2%	171.36	0.2%
21-Mar-12	170.00	168.53	-0.9%	169.86	-0.1%	169.74	-0.2%
22-Mar-12	163.00	171.54	5.2%	168.99	3.7%	168.92	3.6%
23-Mar-12	165.00	170.79	3.5%	162.77	-1.3%	163.01	-1.2%
26-Mar-12	169.00	173.02	2.4%	164.47	-2.7%	163.79	-3.1%
27-Mar-12	169.00	170.66	1.0%	168.56	-0.3%	167.32	-1.0%
28-Mar-12	172.00	172.72	0.4%	168.87	-1.8%	167.96	-2.4%
29-Mar-12	171.00	174.08	1.8%	171.68	0.4%	171.23	0.1%
30-Mar-12	174.00	174.01	0.0%	170.55	-2.0%	170.38	-2.1%
MAPE			**1.85%**		**0.88%**		**0.97%**
RMSE			**3.99**		**1.94**		**2.10**

Table A13-3: *Stock04 (Bamburi Cement) results – using Encog on NSE data*

		Test 1		**Test 2***		**Test 3**	
Date	Actual	Predicted	Error%	Predicted	Error%	Predicted	Error%
03-Jan-12	125.00	123.88	-0.9%	124.58	-0.3%	126.50	1.2%
04-Jan-12	125.00	123.74	-1.0%	124.70	-0.2%	122.90	-1.7%
05-Jan-12	125.00	123.61	-1.1%	125.10	0.1%	122.38	-2.1%
06-Jan-12	125.00	123.33	-1.3%	125.96	0.8%	123.78	-1.0%
09-Jan-12	125.00	123.33	-1.3%	125.96	0.8%	123.78	-1.0%
10-Jan-12	125.00	123.33	-1.3%	125.96	0.8%	123.78	-1.0%
11-Jan-12	135.00	123.33	-8.6%	125.96	-6.7%	123.78	-8.3%
12-Jan-12	135.00	132.26	-2.0%	134.02	-0.7%	126.47	-6.3%
13-Jan-12	133.00	132.91	-0.1%	135.33	1.8%	128.33	-3.5%
16-Jan-12	135.00	131.12	-2.9%	133.28	-1.3%	131.83	-2.3%
17-Jan-12	133.00	133.01	0.0%	134.77	1.3%	133.33	0.2%
18-Jan-12	133.00	131.47	-1.1%	132.29	-0.5%	130.09	-2.2%
19-Jan-12	130.00	131.36	1.0%	132.09	1.6%	130.29	0.2%
20-Jan-12	132.00	128.48	-2.7%	129.78	-1.7%	129.02	-2.3%
23-Jan-12	131.00	130.19	-0.6%	130.94	0.0%	128.52	-1.9%
24-Jan-12	135.00	129.28	-4.2%	130.50	-3.3%	127.78	-5.3%
25-Jan-12	135.00	133.04	-1.5%	133.94	-0.8%	129.37	-4.2%
26-Jan-12	130.00	133.21	2.5%	134.68	3.6%	130.58	0.4%
27-Jan-12	130.00	128.56	-1.1%	130.10	0.1%	130.02	0.0%
30-Jan-12	135.00	128.26	-5.0%	129.61	-4.0%	129.60	-4.0%
31-Jan-12	135.00	133.04	-1.4%	133.55	-1.1%	128.52	-4.8%
01-Feb-12	130.00	133.33	2.6%	134.29	3.3%	129.01	-0.8%
02-Feb-12	130.00	128.50	-1.2%	130.26	0.2%	130.30	0.2%
03-Feb-12	130.00	128.23	-1.4%	129.68	-0.2%	129.79	-0.2%
06-Feb-12	130.00	128.31	-1.3%	129.38	-0.5%	126.95	-2.3%
07-Feb-12	128.00	128.27	0.2%	129.46	1.1%	126.49	-1.2%
08-Feb-12	130.00	126.35	-2.8%	128.31	-1.3%	126.72	-2.5%
09-Feb-12	130.00	128.00	-1.5%	129.59	-0.3%	126.95	-2.3%
10-Feb-12	129.00	128.10	-0.7%	129.80	0.6%	126.60	-1.9%
13-Feb-12	125.00	127.21	1.8%	129.08	3.3%	126.83	1.5%
14-Feb-12	125.00	123.72	-1.0%	126.05	0.8%	125.99	0.8%
15-Feb-12	130.00	123.54	-5.0%	125.45	-3.5%	124.70	-4.1%
16-Feb-12	130.00	127.75	-1.7%	129.34	-0.5%	124.67	-4.1%
17-Feb-12	127.00	127.99	0.8%	129.96	2.3%	125.29	-1.3%
20-Feb-12	127.00	125.31	-1.3%	127.98	0.8%	126.78	-0.2%
21-Feb-12	130.00	125.18	-3.7%	127.62	-1.8%	126.70	-2.5%
22-Feb-12	130.00	127.90	-1.6%	129.51	-0.4%	125.73	-3.3%
23-Feb-12	130.00	128.06	-1.5%	129.85	-0.1%	125.98	-3.1%
24-Feb-12	130.00	128.02	-1.5%	130.04	0.0%	127.54	-1.9%
27-Feb-12	140.00	128.04	-8.5%	129.99	-7.2%	127.83	-8.7%
28-Feb-12	140.00	137.82	-1.6%	139.53	-0.3%	130.64	-6.7%
29-Feb-12	145.00	138.50	-4.5%	141.04	-2.7%	132.90	-8.3%
01-Mar-12	154.00	143.70	-6.7%	146.63	-4.8%	140.26	-8.9%
02-Mar-12	154.00	152.84	-0.8%	155.34	0.9%	147.75	-4.1%
05-Mar-12	155.00	153.76	-0.8%	155.55	0.4%	150.84	-2.7%
06-Mar-12	155.00	154.83	-0.1%	156.26	0.8%	157.81	1.8%
07-Mar-12	157.00	155.11	-1.2%	155.83	-0.7%	158.14	0.7%
08-Mar-12	157.00	157.16	0.1%	156.84	-0.1%	157.24	0.2%
09-Mar-12	157.00	157.29	0.2%	157.05	0.0%	158.02	0.7%
12-Mar-12	153.00	157.34	2.8%	156.96	2.6%	158.97	3.9%
13-Mar-12	150.00	153.75	2.5%	153.53	2.4%	157.24	4.8%
14-Mar-12	150.00	150.76	0.5%	150.23	0.2%	153.85	2.6%
15-Mar-12	155.00	150.51	-2.9%	149.89	-3.3%	150.39	-3.0%
16-Mar-12	155.00	155.08	0.1%	154.43	-0.4%	150.47	-2.9%
19-Mar-12	155.00	155.25	0.2%	155.33	0.2%	152.71	-1.5%
20-Mar-12	155.00	155.23	0.1%	155.52	0.3%	156.70	1.1%
21-Mar-12	155.00	155.27	0.2%	155.43	0.3%	157.50	1.6%
22-Mar-12	155.00	155.41	0.3%	155.06	0.0%	156.08	0.7%
23-Mar-12	155.00	155.41	0.3%	155.06	0.0%	156.08	0.7%
26-Mar-12	155.00	155.41	0.3%	155.06	0.0%	156.08	0.7%
27-Mar-12	153.00	155.41	1.6%	155.06	1.3%	156.08	2.0%
28-Mar-12	153.00	153.59	0.4%	153.36	0.2%	155.04	1.3%
29-Mar-12	150.00	153.46	2.3%	153.13	2.1%	154.41	2.9%
30-Mar-12	150.00	150.62	0.4%	150.46	0.3%	151.59	1.1%
MAPE			**1.76%**		**1.32%**		**2.53%**
RMSE			**3.52**		**2.81**		**4.71**

APPENDIX 14 – Test Results Using Developed Prototype on NYSE data

These were the test results obtained by using the prototype on NYSE data (best 3 results):

Table A14-1: *Stock02 (Microsoft) results – using Project Prototype on NYSE data*

	US$	**Test 1**		**Test 2**		**Test 3***	
Date	Actual	Predicted	Error%	Predicted	Error%	Predicted	Error%
03-Jan-12	25.46	24.69	-3.0%	24.68	-3.1%	24.71	-2.9%
04-Jan-12	26.06	25.41	-2.5%	25.44	-2.4%	25.38	-2.6%
05-Jan-12	26.33	25.91	-1.6%	25.95	-1.4%	25.86	-1.8%
06-Jan-12	26.74	26.20	-2.0%	26.30	-1.6%	26.19	-2.1%
09-Jan-12	26.38	26.62	0.9%	26.66	1.1%	26.65	1.0%
10-Jan-12	26.48	26.43	-0.2%	26.23	-0.9%	26.38	-0.4%
11-Jan-12	26.36	26.51	0.6%	26.55	0.7%	26.48	0.5%
12-Jan-12	26.63	26.41	-0.8%	26.36	-1.0%	26.38	-0.9%
13-Jan-12	26.87	26.60	-1.0%	26.56	-1.2%	26.61	-1.0%
17-Jan-12	26.88	26.82	-0.2%	26.80	-0.3%	26.83	-0.2%
18-Jan-12	26.85	26.86	0.0%	26.86	0.0%	26.85	0.0%
19-Jan-12	26.74	26.88	0.5%	26.85	0.4%	26.85	0.4%
20-Jan-12	28.26	26.77	-5.3%	26.75	-5.3%	26.75	-5.3%
23-Jan-12	28.28	27.86	-1.5%	27.76	-1.8%	27.93	-1.2%
24-Jan-12	27.91	27.83	-0.3%	27.30	-2.2%	27.95	0.1%
25-Jan-12	28.11	27.70	-1.5%	27.91	-0.7%	27.82	-1.0%
26-Jan-12	28.06	28.11	0.2%	28.11	0.2%	28.06	0.0%
27-Jan-12	27.80	27.95	0.5%	27.96	0.6%	27.97	0.6%
30-Jan-12	28.16	27.62	-1.9%	27.81	-1.2%	27.74	-1.5%
31-Jan-12	28.09	28.05	-0.1%	28.09	0.0%	28.05	-0.1%
01-Feb-12	28.43	27.97	-1.6%	27.95	-1.7%	27.98	-1.6%
02-Feb-12	28.49	28.23	-0.9%	28.31	-0.6%	28.24	-0.9%
03-Feb-12	28.76	28.34	-1.5%	28.24	-1.8%	28.31	-1.6%
06-Feb-12	28.72	28.51	-0.7%	28.60	-0.4%	28.50	-0.8%
07-Feb-12	28.87	28.52	-1.2%	28.53	-1.2%	28.50	-1.3%
08-Feb-12	29.16	28.64	-1.8%	28.77	-1.3%	28.60	-1.9%
09-Feb-12	29.27	28.89	-1.3%	28.89	-1.3%	28.80	-1.6%
10-Feb-12	29.01	28.94	-0.2%	29.01	0.0%	28.86	-0.5%
13-Feb-12	29.08	28.83	-0.9%	28.95	-0.4%	28.72	-1.2%
14-Feb-12	28.96	28.98	0.1%	29.06	0.3%	28.78	-0.6%
15-Feb-12	28.77	28.83	0.2%	28.88	0.4%	28.69	-0.3%
16-Feb-12	29.96	28.64	-4.4%	28.77	-4.0%	28.54	-4.7%
17-Feb-12	29.92	29.46	-1.5%	29.39	-1.8%	29.23	-2.3%
21-Feb-12	30.10	29.33	-2.6%	29.03	-3.6%	29.20	-3.0%
22-Feb-12	29.94	29.59	-1.2%	29.82	-0.4%	29.49	-1.5%
23-Feb-12	30.03	29.76	-0.9%	29.58	-1.5%	30.21	0.6%
24-Feb-12	30.14	29.79	-1.2%	29.83	-1.0%	30.28	0.5%
27-Feb-12	30.01	29.82	-0.6%	29.80	-0.7%	30.11	0.3%
28-Feb-12	30.51	29.74	-2.5%	29.79	-2.4%	30.25	-0.9%
29-Feb-12	30.39	30.01	-1.3%	30.03	-1.2%	30.44	0.2%
01-Mar-12	30.91	29.92	-3.2%	29.86	-3.4%	30.41	-1.6%
02-Mar-12	30.71	30.21	-1.6%	30.28	-1.4%	30.56	-0.5%
05-Mar-12	30.44	30.18	-0.9%	30.06	-1.2%	30.53	0.3%
06-Mar-12	30.21	30.20	0.0%	30.23	0.1%	30.46	0.8%
07-Mar-12	30.48	30.18	-1.0%	30.09	-1.3%	30.41	-0.2%
08-Mar-12	30.64	30.15	-1.6%	30.12	-1.7%	30.47	-0.6%
09-Mar-12	30.63	30.09	-1.8%	30.08	-1.8%	30.50	-0.4%
12-Mar-12	30.67	30.12	-1.8%	30.17	-1.6%	30.50	-0.6%
13-Mar-12	31.28	30.23	-3.4%	30.22	-3.4%	30.52	-2.4%
14-Mar-12	31.37	30.41	-3.1%	30.38	-3.2%	30.64	-2.3%
15-Mar-12	31.45	30.41	-3.3%	30.37	-3.4%	30.65	-2.5%
16-Mar-12	31.21	30.55	-2.1%	30.59	-2.0%	30.68	-1.7%
19-Mar-12	30.83	30.60	-0.7%	30.53	-1.0%	30.65	-0.6%
20-Mar-12	30.63	30.53	-0.3%	30.50	-0.4%	30.56	-0.2%
21-Mar-12	30.55	30.45	-0.3%	30.38	-0.6%	30.51	-0.1%
22-Mar-12	30.64	30.31	-1.1%	30.25	-1.3%	30.49	-0.5%
23-Mar-12	30.64	30.24	-1.3%	30.23	-1.3%	30.51	-0.4%
26-Mar-12	31.20	30.20	-3.2%	30.20	-3.2%	30.51	-2.2%
27-Mar-12	31.13	30.38	-2.4%	30.38	-2.4%	30.62	-1.6%
28-Mar-12	30.82	30.35	-1.5%	30.30	-1.7%	30.61	-0.7%
29-Mar-12	30.75	30.40	-1.1%	30.42	-1.1%	30.56	-0.6%
30-Mar-12	30.88	30.46	-1.4%	30.39	-1.6%	30.55	-1.1%
MAPE			**1.42%**		**1.47%**		**1.16%**

Table A14-2: *Stock04 (Coca cola) results – using Project Prototype on NYSE data*

	US$	**Test 1***		**Test 2**		**Test 3**	
Date	Actual	Predicted	Error%	Predicted	Error%	Predicted	Error%
03-Jan-12	33.39	33.18	-0.6%	33.12	-0.8%	33.16	-0.7%
04-Jan-12	33.18	33.26	0.2%	33.14	-0.1%	33.17	0.0%
05-Jan-12	33.02	33.09	0.2%	33.07	0.2%	33.08	0.2%
06-Jan-12	32.81	32.95	0.4%	32.89	0.2%	32.90	0.3%
09-Jan-12	32.81	32.76	-0.2%	32.76	-0.2%	32.76	-0.2%
10-Jan-12	33.01	32.73	-0.8%	32.71	-0.9%	32.72	-0.9%
11-Jan-12	32.40	32.88	1.5%	32.86	1.4%	32.88	1.5%
12-Jan-12	32.17	32.41	0.7%	32.41	0.7%	32.40	0.7%
13-Jan-12	31.89	32.17	0.9%	32.16	0.8%	32.16	0.8%
17-Jan-12	32.06	31.97	-0.3%	32.01	-0.2%	31.99	-0.2%
18-Jan-12	32.18	32.00	-0.6%	32.00	-0.6%	32.01	-0.5%
19-Jan-12	32.11	32.14	0.1%	32.15	0.1%	32.17	0.2%
20-Jan-12	32.42	32.08	-1.0%	32.04	-1.2%	32.08	-1.0%
23-Jan-12	32.62	32.35	-0.8%	32.36	-0.8%	32.37	-0.8%
24-Jan-12	32.32	32.54	0.7%	32.53	0.6%	32.56	0.7%
25-Jan-12	32.58	32.32	-0.8%	32.24	-1.0%	32.30	-0.9%
26-Jan-12	32.38	32.51	0.4%	32.54	0.5%	32.52	0.4%
27-Jan-12	32.11	32.38	0.8%	32.38	0.8%	32.41	0.9%
30-Jan-12	32.12	32.10	-0.1%	32.06	-0.2%	32.08	-0.1%
31-Jan-12	32.15	32.13	-0.1%	32.15	0.0%	32.16	0.0%
01-Feb-12	32.30	32.13	-0.5%	32.13	-0.5%	32.15	-0.5%
02-Feb-12	32.29	32.24	-0.2%	32.21	-0.2%	32.25	-0.1%
03-Feb-12	32.41	32.26	-0.5%	32.23	-0.6%	32.27	-0.4%
06-Feb-12	32.39	32.36	-0.1%	32.32	-0.2%	32.36	-0.1%
07-Feb-12	32.63	32.36	-0.8%	32.33	-0.9%	32.38	-0.8%
08-Feb-12	32.53	32.54	0.0%	32.52	0.0%	32.55	0.1%
09-Feb-12	32.36	32.50	0.4%	32.45	0.3%	32.51	0.5%
10-Feb-12	32.34	32.35	0.0%	32.29	-0.2%	32.33	0.0%
13-Feb-12	32.58	32.33	-0.8%	32.33	-0.8%	32.35	-0.7%
14-Feb-12	32.80	32.50	-0.9%	32.49	-0.9%	32.52	-0.9%
15-Feb-12	32.56	32.68	0.4%	32.65	0.3%	32.69	0.4%
16-Feb-12	32.78	32.54	-0.7%	32.45	-1.0%	32.52	-0.8%
17-Feb-12	32.87	32.69	-0.5%	32.70	-0.5%	32.69	-0.5%
21-Feb-12	32.76	32.78	0.1%	32.78	0.1%	32.81	0.2%
22-Feb-12	32.97	32.71	-0.8%	32.61	-1.1%	32.66	-0.9%
23-Feb-12	32.93	32.86	-0.2%	32.87	-0.2%	32.86	-0.2%
24-Feb-12	32.85	32.85	0.0%	32.81	-0.1%	32.85	0.0%
27-Feb-12	32.77	32.80	0.1%	32.70	-0.2%	32.75	-0.1%
28-Feb-12	32.78	32.72	-0.2%	32.71	-0.2%	32.72	-0.2%
29-Feb-12	33.26	32.71	-1.7%	32.68	-1.7%	32.70	-1.7%
01-Mar-12	33.13	33.04	-0.3%	33.08	-0.2%	33.06	-0.2%
02-Mar-12	32.93	33.04	0.3%	32.93	0.0%	33.00	0.2%
05-Mar-12	32.96	32.90	-0.2%	32.78	-0.5%	32.82	-0.4%
06-Mar-12	32.73	32.88	0.5%	32.94	0.6%	32.90	0.5%
07-Mar-12	32.75	32.69	-0.2%	32.66	-0.3%	32.68	-0.2%
08-Mar-12	32.98	32.68	-0.9%	32.63	-1.1%	32.66	-1.0%
09-Mar-12	33.09	32.85	-0.7%	32.86	-0.7%	32.87	-0.7%
12-Mar-12	33.40	32.96	-1.3%	32.89	-1.5%	32.94	-1.4%
13-Mar-12	33.69	33.20	-1.5%	33.17	-1.5%	33.19	-1.5%
14-Mar-12	33.67	33.43	-0.7%	33.44	-0.7%	33.43	-0.7%
15-Mar-12	33.73	33.50	-0.7%	33.37	-1.1%	33.43	-0.9%
16-Mar-12	33.65	33.54	-0.3%	33.51	-0.4%	33.48	-0.5%
19-Mar-12	33.76	33.47	-0.9%	33.45	-0.9%	33.43	-1.0%
20-Mar-12	33.85	33.54	-0.9%	33.50	-1.0%	33.48	-1.1%
21-Mar-12	34.11	33.60	-1.5%	33.57	-1.6%	33.56	-1.6%
22-Mar-12	34.25	33.78	-1.4%	33.74	-1.5%	33.72	-1.5%
23-Mar-12	34.28	33.90	-1.1%	33.86	-1.2%	33.84	-1.3%
26-Mar-12	34.48	33.96	-1.5%	33.88	-1.7%	33.86	-1.8%
27-Mar-12	34.47	34.06	-1.2%	34.11	-1.0%	34.00	-1.4%
28-Mar-12	34.84	34.08	-2.2%	34.05	-2.3%	34.00	-2.4%
29-Mar-12	35.40	34.27	-3.2%	34.35	-3.0%	34.20	-3.4%
30-Mar-12	35.49	34.54	-2.7%	34.75	-2.1%	34.55	-2.6%
MAPE			**0.713%**		**0.749%**		**0.748%**

Table A14-3: *Stock06 (Alcatel) results – using Project Prototype on NYSE data*

	US$	**Test 1**		**Test 2**		**Test 3***	
Date	Actual	Predicted	Error%	Predicted	Error%	Predicted	Error%
03-Jan-12	1.70	1.57	-7.6%	1.57	-7.6%	1.56	-8.2%
04-Jan-12	1.60	1.68	5.0%	1.68	5.0%	1.70	6.2%
05-Jan-12	1.58	1.60	1.3%	1.60	1.3%	1.60	1.3%
06-Jan-12	1.56	1.58	1.3%	1.57	0.6%	1.57	0.6%
09-Jan-12	1.57	1.57	0.0%	1.56	-0.6%	1.57	0.0%
10-Jan-12	1.62	1.58	-2.5%	1.57	-3.1%	1.57	-3.1%
11-Jan-12	1.75	1.61	-8.0%	1.61	-8.0%	1.62	-7.4%
12-Jan-12	1.80	1.72	-4.4%	1.73	-3.9%	1.75	-2.8%
13-Jan-12	1.74	1.78	2.3%	1.79	2.9%	1.80	3.4%
17-Jan-12	1.78	1.72	-3.4%	1.72	-3.4%	1.73	-2.8%
18-Jan-12	1.88	1.76	-6.4%	1.75	-6.9%	1.78	-5.3%
19-Jan-12	1.99	1.87	-6.0%	1.87	-6.0%	1.89	-5.0%
20-Jan-12	2.00	1.99	-0.5%	2.00	0.0%	1.99	-0.5%
23-Jan-12	2.01	2.00	-0.5%	2.00	-0.5%	2.00	-0.5%
24-Jan-12	1.94	2.01	3.6%	2.00	3.1%	2.01	3.6%
25-Jan-12	1.81	1.94	7.2%	1.93	6.6%	1.95	7.7%
26-Jan-12	1.84	1.81	-1.6%	1.81	-1.6%	1.82	-1.1%
27-Jan-12	1.83	1.84	0.5%	1.83	0.0%	1.85	1.1%
30-Jan-12	1.78	1.83	2.8%	1.84	3.4%	1.84	3.4%
31-Jan-12	1.74	1.78	2.3%	1.78	2.3%	1.78	2.3%
01-Feb-12	1.80	1.73	-3.9%	1.72	-4.4%	1.75	-2.8%
02-Feb-12	1.86	1.79	-3.8%	1.79	-3.8%	1.81	-2.7%
03-Feb-12	1.95	1.86	-4.6%	1.86	-4.6%	1.87	-4.1%
06-Feb-12	1.93	1.95	1.0%	1.95	1.0%	1.95	1.0%
07-Feb-12	1.91	1.93	1.0%	1.93	1.0%	1.93	1.0%
08-Feb-12	1.95	1.91	-2.1%	1.90	-2.6%	1.91	-2.1%
09-Feb-12	1.94	1.95	0.5%	1.94	0.0%	1.96	1.0%
10-Feb-12	2.19	1.95	-11.0%	1.95	-11.0%	1.95	-11.0%
13-Feb-12	2.19	2.20	0.5%	2.19	0.0%	2.19	0.0%
14-Feb-12	2.30	2.21	-3.9%	2.21	-3.9%	2.19	-4.8%
15-Feb-12	2.28	2.30	0.9%	2.29	0.4%	2.28	0.0%
16-Feb-12	2.36	2.30	-2.5%	2.29	-3.0%	2.29	-3.0%
17-Feb-12	2.45	2.36	-3.7%	2.36	-3.7%	2.35	-4.1%
21-Feb-12	2.57	2.46	-4.3%	2.46	-4.3%	2.46	-4.3%
22-Feb-12	2.44	2.56	4.9%	2.56	4.9%	2.56	4.9%
23-Feb-12	2.57	2.45	-4.7%	2.45	-4.7%	2.45	-4.7%
24-Feb-12	2.59	2.56	-1.2%	2.57	-0.8%	2.57	-0.8%
27-Feb-12	2.60	2.60	0.0%	2.61	0.4%	2.61	0.4%
28-Feb-12	2.54	2.58	1.6%	2.58	1.6%	2.59	2.0%
29-Feb-12	2.47	2.55	3.2%	2.55	3.2%	2.55	3.2%
01-Mar-12	2.39	2.47	3.3%	2.48	3.8%	2.48	3.8%
02-Mar-12	2.39	2.40	0.4%	2.41	0.8%	2.40	0.4%
05-Mar-12	2.38	2.40	0.8%	2.40	0.8%	2.40	0.8%
06-Mar-12	2.24	2.39	6.7%	2.40	7.1%	2.39	6.7%
07-Mar-12	2.34	2.26	-3.4%	2.27	-3.0%	2.25	-3.8%
08-Mar-12	2.43	2.34	-3.7%	2.35	-3.3%	2.34	-3.7%
09-Mar-12	2.38	2.44	2.5%	2.45	2.9%	2.44	2.5%
12-Mar-12	2.33	2.39	2.6%	2.39	2.6%	2.37	1.7%
13-Mar-12	2.37	2.34	-1.3%	2.34	-1.3%	2.34	-1.3%
14-Mar-12	2.39	2.38	-0.4%	2.38	-0.4%	2.38	-0.4%
15-Mar-12	2.41	2.40	-0.4%	2.40	-0.4%	2.40	-0.4%
16-Mar-12	2.43	2.41	-0.8%	2.42	-0.4%	2.41	-0.8%
19-Mar-12	2.47	2.44	-1.2%	2.44	-1.2%	2.44	-1.2%
20-Mar-12	2.43	2.47	1.6%	2.48	2.1%	2.47	1.6%
21-Mar-12	2.32	2.44	5.2%	2.44	5.2%	2.44	5.2%
22-Mar-12	2.30	2.33	1.3%	2.34	1.7%	2.33	1.3%
23-Mar-12	2.31	2.31	0.0%	2.32	0.4%	2.31	0.0%
26-Mar-12	2.43	2.32	-4.5%	2.33	-4.1%	2.32	-4.5%
27-Mar-12	2.42	2.43	0.4%	2.43	0.4%	2.43	0.4%
28-Mar-12	2.36	2.43	3.0%	2.43	3.0%	2.42	2.5%
29-Mar-12	2.31	2.37	2.6%	2.37	2.6%	2.36	2.2%
30-Mar-12	2.27	2.32	2.2%	2.33	2.6%	2.32	2.2%
MAPE			**2.82%**		**2.84%**		**2.77%**

APPENDIX 15 – How to Use Neuroph to Setup an ANN for Stock Market Prediction

The procedure for creating and testing an ANN for stock market prediction based on the developed model is described below:

a) Data preparation

1. Reform each stock data (.csv) into data table with 6 columns, representing 6-days of continuous trade. The first 5 columns shall be used for input, while the 6th column shall be used as the output.
2. Normalize the data, by converting the 6 columns into normalized data, using the following formulae in Equation A15-1:

$$Normalized(x) = \frac{Real(x) - Min(x)}{Max(x) - Min(x)} \quad \text{...} \text{ A15-1}$$

3. Use the normalized data as input to the frameworks i.e. Stock01normalized.csv, Stock02normalized.csv,and Stock04normalized.csv.
4. After obtaining the results in normalized form, convert back to real values by using Equation A15-2 below:
*Real(x) = Normalized(x) * (Max(x) – Min(x)) + Min(x)* ………………………………….. A15-2

b) Create a new network

1. Start Neuroph program
2. Click on File menu, and select 'New project'.
3. Pick Neuroph category, accept Neuroph project and click next
4. Provide a project name e.g. ANNstock01, then click Finish
5. A project hierarchy is created with folders – Neural Network, Training Sets and Test Sets
6. Right click the Neural Network folder, click New, click Neural Network
7. On the resulting wizard, provide a name e.g. ANNstock01, on the network type list, select 'Multi Layer Perceptron', then click next.
8. On the resulting screen, setup network parameters as follows:
   Input neurons – 5
   Hidden neurons – 21 21 (meaning 2 layers, with 21 neurons per layer)
   Output neurons – 1
   Transfer function – Sigmoid
   Learning rule – Resilient propagation
   Leave the rest of options default i.e. Use bias neurons checked, Connect input to output neurons unchecked
9. Click Finish to finalize the network, which should then be displayed graphically on screen

c) Creating data files (training & testing sets):

1. On the project hierarchy folders on the left, right click the Training Sets folder, click New, click 'Data Set'
2. On the resulting wizard, provide the required information:
   Training set name e.g. ANNstock01training
   Type – select Supervised
   Number of inputs – 5
   Number of outputs – 1
3. Click on next, to view a blank screen grid of 5 inputs and one output. Click on the 'Load from file' button then 'Choose file' to load the Stock01normalized.csv (ensure that this file has only the 80% training data), navigate through the folder hierarchy to the location of the file and pick the file then choose the value separator as comma and click 'Load'
4. The data is shown on the screen with the headers showing 5 inputs and 1 output
5. Click 'Finish'

6. Repeat 1 to 5 to create the testing set, with a different file name (e.g. ANNstock01testData), based on 3-month data from the 20% of the data (date range January 2, 2012 to March 31, 2012)

d) Training the network:

1. Open the neural network created in (a) above and display the structure on the screen, as shown on Figure 4.3.
2. Drag the dataset created for training from the left panel and drop it on the ‘Dataset’ input part on the network diagram
3. While the network diagram is still on the active screen, click the ‘Train’ icon on the menu and set the parameters for training below:

   Max error – 0.001
   Limit max interactions – 130,000
   Learning rate – 0.2
   Momentum – 0.7
4. Click the ‘Train’ button, to start the training phase which should run until completion

e) Testing:

1. Open the neural network created in (a) above and display the structure on the screen
2. Drag the dataset created for testing from the left panel and drop it on the ‘Dataset’ input part on the network diagram
3. While the network diagram is still on the active screen, click the ‘Test’ icon on the menu bar
4. The Test results are displayed on screen, each entry with both the desired output and an error
5. Copy the data to clipboard, then to an output file (Stock01out.txt)
6. Un-normalized the output data using the formulae in equation (A15-2)

f) Repeat the Training and Testing phases for all the three stocks.

APPENDIX 16 – How to Use Encog to Setup an ANN for Stock Market Prediction

The procedure for creating and testing an ANN for stock market prediction based on the developed model is described below.

When the program starts, the user is asked to create a new project folder and give it a name. Once the project is named, a blank screen is shown. The following procedure was used:

a) Data preparation
   1. Reform each stock data (.csv) into data table with 6 columns, representing 6-days of continuous trade. The first 5 columns shall be used for input, while the 6th column shall be used as the output.
   2. Normalize the data, by converting the 6 columns into normalized data, using the formulae in Equation A15-1 (Appendix 15)
   3. Use the normalized data as input to the frameworks i.e. Stock01normalized.csv, Stock02normalized.csv and Stock04normalized.csv.
   4. After obtaining the results in normalized form, convert back to real values by using Equation A15-2 (Appendix 15)

b) Create the network
   1. Right click the folder icon on the left side of the file hierarchy on the screen, and click 'New file'
   2. On the resulting window for Creating a file, click on 'Machine learning method (*.eg)' and give it a filename e.g. ANNlearning
   3. On the resulting screen, choose a method i.e. Feedforward Neural Network, then OK
   4. On the resulting screen for Creating FeedForward Network, input the following:

      Input neuron count – 5 neurons

      Hidden layer counts – click on Add and input 21 neurons for layer1, and another 21 for layer 2

      Output Neuron count – 1 neuron

      Activation function hidden – default is 'ActivationTANH', click on the change button and pick 'ActivationSigmoid'

      Activation function output – default is 'ActivationTANH', click on the change button and pick 'ActivationSigmoid'

c) Creating data files (training & testing sets):
   1. Right click the folder icon on the left side of the file hierarchy on the screen, and click 'New file'
   2. On the resulting window for Creating a file, click on 'CSV File (*.csv)' and give it a filename e.g. ANNstock01trainData
   3. Double click the created CSV file, which opens as blank file
   4. Open the original data file Stock01normalized.csv (as text file) and copy 80% of the data to the blank CSV file (i.e. from Jan. 2, 2008 to Dec. 31, 2011). Note that Encog has an 'analyst wizard' tool that can normalize data, though this is not used in this research due to time constraint in testing its effectiveness.
   5. Close the file and Save the CSV file in Encog
   6. Right click the created CSV file and click 'Export to training (EGB)', set input count = 5, and ideal count = 1 (no. of columns with correct data), leave the 'significant column present' unchecked then click OK. On the resulting window, leave all options unchecked, apart from 'Generate significant column' which should be checked, then click OK.
   7. You get a message that the 'import is complete'. Click OK to finish the task.

8. Repeat 1 to 7 to create the testing set, with a different file name (e.g. ANNstock01testData), from the 20% of the data (date range January 2, 2012 to March 31, 2012)

d) Training:
   1. Click on Tools menu, then pick the 'Train' option
   2. Confirm that the displayed Training set and Network name are correct, then OK
   3. Choose Training method i.e. Propagation – Resilient (RPROP), then OK
   4. On the resulting screen, leave all values default i.e.

      Maximum error % – 1.0
      Cross validation – 0
      RPROP type – classic
      Max step – 50.0
      Initial update – 0.1

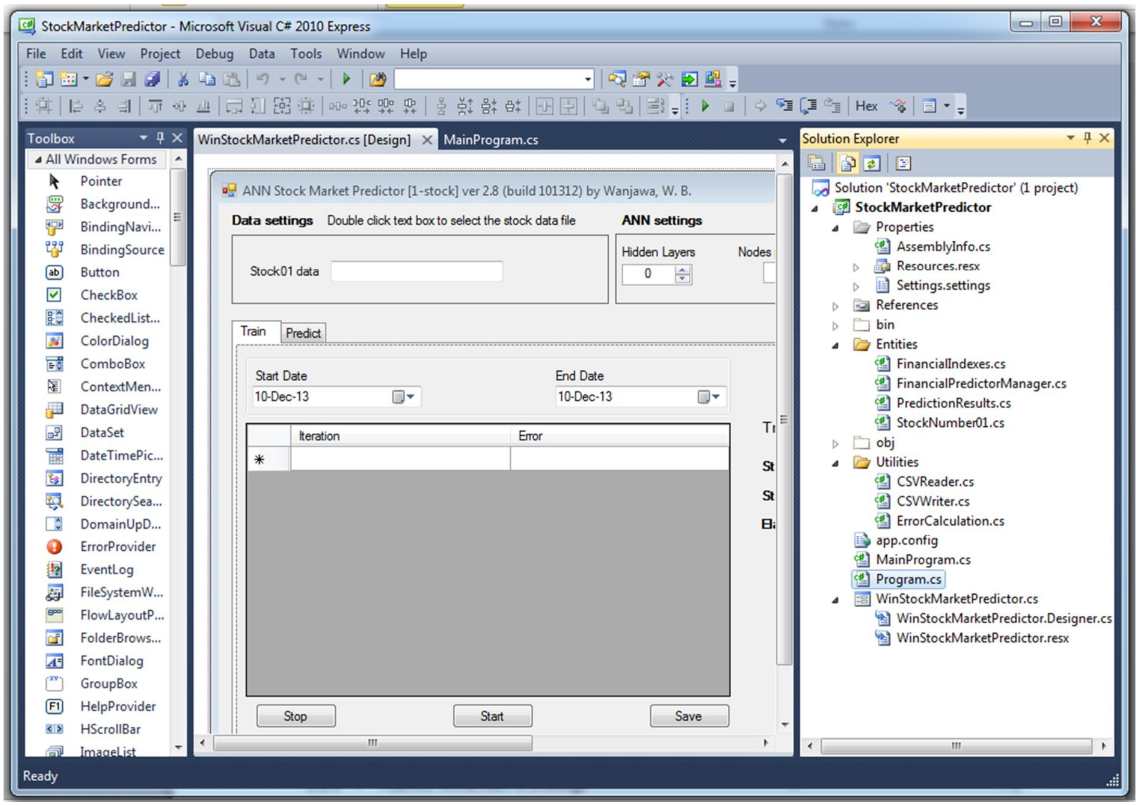
   5. On the resulting graphic, on the bottom side of screen, set log level to 'Info', to get information on network performance
   6. Click 'Start', to start the training
   7. The training runs until finish or until Stopped
   8. In case training runs for less than 5 iterations, click on 'select option' on the lower part of the screen and select 'reset' then 'Start', ensuring the iterations are at least 50

e) Testing:
   1. Click on Tools menu, then pick the Validation chart
   2. Change Training set to the file created from the 20% test data e.g. ANNstock01testData, retain the same Network name, then OK
   3. A graph is displayed showing a comparison between true values (validation in red colour) and what is computed by the ANN (computation in blue colour)
   4. Click the 'Data' tab at the top, to view the raw data for the graph
   5. Select all the text on the data screen and save to Excel
   6. Un-normalize the output data using the formulae in equation A15-2 (Appendix 15)

f) Repeat the Training and Testing phases for all the three stocks.

APPENDIX 17 – Screenshot of C# Development Environment

The screenshot below shows the C# development screen with the various tools used in developing the prototype:



Stock01 data

Hidden Layers 0

Start Date 10-Dec-13

End Date 10-Dec-13

Main components on the design screen:

- Solution: Stock market predictor
- Properties: AssemblyInfo.cs, Resources.resx (Resources.Designer.resx), Settings.settings (Settings.Designer.cs)
- References: (classes being used e.g. encog-core-cs, System, System.Windows.Foms etc.)
- Bin: Debug (data files, dlls), Release
- Entities: FinancialIndexes.cs, FinancialPredictorManager.cs, PredictionResults.cs, Stock01.cs
- Obj: Debug (data files, dlls)
- Utilities: CSVReader.cs, CSVWriter.cs, ErrorCalculation.cs
- App.config
- MainProgram.cs **(*main program)**
- Program.cs
- Stock Market Predictor form.cs (***main form**): Form.Designer.cs, Form.resx

APPENDIX 18 – Source Code of Main Program

Shown below is the source code for the two main components of the C# based prototype:

Source code for *MainProgram.cs*

```
/*
 * Program to Create, Train and Test ANN for Stock Market prediction
 *
 * Author – Wanjawa, W. B.
 * Contact – WanjawaWB@gmail.com
 * Last Revised – February 2014
 * Version – 2.10 (1 input stock at a time)
 *
 */

using System;
using System.Text;
using System.Collections.Generic;
using System.IO;
using System.Threading;
using FinancialMarketPredictor.Entities; //class within the project
using FinancialMarketPredictor.Utilities; //class within the project
using Encog.Neural.Networks;
using Encog.Neural.Networks.Layers;
using Encog.Neural.Networks.Training;
using Encog.Neural.Data.Basic;
using Encog.Neural.Networks.Training.LMA;
using Encog.Neural.Networks.Training.Propagation.Resilient;
using Encog.Neural.Networks.Training.Anneal;
using Encog.Neural.Networks.Training.Strategy;
using Encog.Neural.Activation;
using Encog.Persist.Persistors;

namespace FinancialMarketPredictor
{
publicenumTrainingAlgorithm
  {
    Resilient,
    Annealing,
    Evolutionary
  }

publicdelegatevoidTrainingStatus(int iteration, double error, TrainingAlgorithm algorithm);

publicsealedclassMainProgram// class for prediction
  {
// constants
privateconstint NUMBER_OF_STOCKS = 1;
privateconstint NUMBER_OF_DATA_ROWS = 5;
privateconstint NUMBER_OF_NETWORK_OUTPUTS = 1;
privateconstdouble MAX_ERROR = 0.00005;

// Private members
privateBasicNetwork varNetworkToBeTrained; // Network to be trained
privatedouble[][] varInputMatrix; // Input data Stock01
privatedouble[][] varCorrectValuesMatrix; // Desired output
privateFinancialPredictorManager varFinancialPredictorManager; //Financial market predictor
privateThread varTrainingThread;  // Training tread
privatestring varPathToStock01; // path To Stock01
privateint varSizeOfTrainingData = 1000;// Size of the training data

publicbool Loaded { get; privateset; } // Gets the information about the predictor
publicint HiddenLayers { get; privateset; }
publicint HiddenUnits { get; privateset; }

publicDateTime MaxIndexDate  // Maximum date for training and prediction
    {
```

```
get
      {
return varFinancialPredictorManager == null ? DateTime.MinValue : varFinancialPredictorManager.MaxDate;
      }
    }

publicDateTime MinIndexDate // Minimum date for training and prediction
    {
get
      {
return varFinancialPredictorManager == null ? DateTime.MaxValue : varFinancialPredictorManager.MinDate;
      }
    }

// Constructor for Prediction class
// "hiddenUnits"= Number of hidden units per hidden layer
// "hiddenLayers"= Number of hidden layers
public MainProgram(string varPathToStock01t, int hiddenUnits, int hiddenLayers)
    {
if (!File.Exists(varPathToStock01t))
thrownewArgumentException("Stock01 targets an invalid file");

      varPathToStock01 = varPathToStock01t;

// Create new network
CreateNetwork(hiddenUnits, hiddenLayers);
// Create new financial predictor manager
varFinancialPredictorManager = newFinancialPredictorManager(NUMBER_OF_DATA_ROWS, NUMBER_OF_NETWORK_OUTPUTS);
// Load Stock01
varFinancialPredictorManager.Load(varPathToStock01);
      Loaded = true;
      HiddenLayers = hiddenLayers;
      HiddenUnits = hiddenUnits;
    }

// Reload indexes
publicvoid ReloadFiles(string varPathToStock01t)
    {
if (!File.Exists(varPathToStock01t))
thrownewArgumentException("Path to Stock no. 1 targets an invalid file");

      Loaded = false;
      varPathToStock01 = varPathToStock01t;

// Create new financial predictor manager
varFinancialPredictorManager = newFinancialPredictorManager(NUMBER_OF_DATA_ROWS, NUMBER_OF_NETWORK_OUTPUTS);
// Load Stock01
varFinancialPredictorManager.Load(varPathToStock01);
varCorrectValuesMatrix = varInputMatrix = null;
      Loaded = true;
    }

// Create a new network
privatevoid CreateNetwork(int hiddenUnits, int hiddenLayers)
    {
varNetworkToBeTrained = newBasicNetwork { Name = "ANN Stock Market Predictor [1-stock] ver 2.10 (build 021426) by
Wanjawa, W. B.", Description = "Network for Stock Market prediction analysis" };
varNetworkToBeTrained.AddLayer(newBasicLayer(NUMBER_OF_DATA_ROWS * NUMBER_OF_STOCKS));          /*Input*/
for (int i = 0; i < hiddenLayers; i++)
varNetworkToBeTrained.AddLayer(newBasicLayer(newActivationSigmoid(), true, hiddenUnits));        /*Hidden layer*/
varNetworkToBeTrained.AddLayer(newBasicLayer(newActivationSigmoid(), true, NUMBER_OF_NETWORK_OUTPUTS));
/*Output of the network*/
varNetworkToBeTrained.Structure.FinalizeStructure();                              /*Finalize network structure*/
varNetworkToBeTrained.Reset();                                                   /*Randomize*/
    }

// Create Training sets for the neural network to be trained
publicvoid CreateTrainingSets(DateTime trainFrom, DateTime trainTo)
    {
// find where we are starting from
int startIndex = -1;
int endIndex = -1;
foreach (FinancialIndexes sample in varFinancialPredictorManager.Samples)
      {
if (sample.Date.CompareTo(trainFrom) < 0) startIndex++;
if (sample.Date.CompareTo(trainTo) < 0)   endIndex++;
      }
// create a sample factor across the training area
varSizeOfTrainingData = endIndex - startIndex;
varInputMatrix = newdouble[varSizeOfTrainingData][];
varCorrectValuesMatrix = newdouble[varSizeOfTrainingData][];

// grab the actual training data from that point
for (int i = startIndex; i < endIndex; i++)
      {
varInputMatrix[i - startIndex] = newdouble[NUMBER_OF_DATA_ROWS * NUMBER_OF_STOCKS];
varCorrectValuesMatrix[i - startIndex] = newdouble[NUMBER_OF_NETWORK_OUTPUTS];
varFinancialPredictorManager.GetInputData(i, varInputMatrix[i - startIndex]);
varFinancialPredictorManager.GetOutputData(i, varCorrectValuesMatrix[i - startIndex]);
      }
    }

// Train the network using Backpropagation and Resilient methods
// "status" means Callback function invoked on each _epoch
publicvoid TrainNetworkAsync(DateTime trainFrom, DateTime trainTo, TrainingStatus status)
    {
Action<DateTime, DateTime, TrainingStatus> action = TrainNetwork;
action.BeginInvoke(trainFrom, trainTo, status, action.EndInvoke, action);
    }

// Train network, "status" means Delegate to be invoked
privatevoid TrainNetwork(DateTime trainFrom, DateTime trainTo, TrainingStatus status)
    {
if(varInputMatrix == null || varCorrectValuesMatrix == null)
CreateTrainingSets(trainFrom, trainTo);        /*Create training sets, according to input parameters*/
varTrainingThread = Thread.CurrentThread;
int epoch = 1;
ITrain train = null;
try
      {

var trainSet = newBasicNeuralDataSet(varInputMatrix, varCorrectValuesMatrix);
train = newResilientPropagation(varNetworkToBeTrained, trainSet);
double error;
do
        {
train.Iteration();
error = train.Error;
if (status != null)
status.Invoke(epoch, error, TrainingAlgorithm.Resilient);
epoch++;
        } while (error > MAX_ERROR);
      }
catch (ThreadAbortException) {/*Training aborted*/ varTrainingThread = null; }
finally
      {
train.FinishTraining();
      }
varTrainingThread = null;
    }

// Abort training
publicvoid AbortTraining()
    {
if (varTrainingThread != null) varTrainingThread.Abort();
    }

// Export neural network
    [System.Security.Permissions.FileIOPermission(System.Security.Permissions.SecurityAction.Demand)]
publicvoid ExportNeuralNetwork(string path)
    {
if (varNetworkToBeTrained == null)
```

```
thrownewNullReferenceException("Network reference is set to null. Nothing to export.");
Encog.Util.SerializeObject.Save(path, varNetworkToBeTrained);
    }

// Load neural network, "path" means Path to previously serialized object
publicvoid LoadNeuralNetwork(string path)
    {
varNetworkToBeTrained = (BasicNetwork)Encog.Util.SerializeObject.Load(path);
      HiddenLayers = varNetworkToBeTrained.Structure.Layers.Count - 2 /*1 input, 1 output*/;
      HiddenUnits = varNetworkToBeTrained.Structure.Layers[1].NeuronCount;
    }

// Predict the results, returns List with the prediction results
publicList<PredictionResults> Predict(DateTime predictFrom, DateTime predictTo)
    {
List<PredictionResults> results = newList<PredictionResults>();
double[] present = newdouble[NUMBER_OF_DATA_ROWS * NUMBER_OF_STOCKS];
double[] actualOutput = newdouble[NUMBER_OF_NETWORK_OUTPUTS];
int index = 1;//**changed this to 1 to try resolve issue of dates
foreach (var sample in varFinancialPredictorManager.Samples)
      {
if (sample.Date.CompareTo(predictFrom) > 0 && sample.Date.CompareTo(predictTo) < 0)
        {
PredictionResults result = newPredictionResults();
varFinancialPredictorManager.GetInputData(index - NUMBER_OF_DATA_ROWS, present);
varFinancialPredictorManager.GetOutputData(index - NUMBER_OF_DATA_ROWS, actualOutput);
var data = newBasicNeuralData(present);
var predict = varNetworkToBeTrained.Compute(data);
          result.Stock01actual = actualOutput[0] * (varFinancialPredictorManager.MaxStock01 -
varFinancialPredictorManager.MinStock01) + varFinancialPredictorManager.MinStock01;
          result.Stock01predicted = predict[0] * (varFinancialPredictorManager.MaxStock01 - varFinancialPredictorManager.MinStock01)
+ varFinancialPredictorManager.MinStock01;

          result.Date = sample.Date;
ErrorCalculation error = newErrorCalculation();
error.UpdateError(actualOutput, predict.Data);
          result.Error = error.CalculateRMS();
results.Add(result);
        }
index++;
      }
return results;
    }
  }
}
```

<u>Source code for *WinStockMarketPredictor.cs*</u>
(Code behind the main program interface form, controls and buttons)

```
/*
 * Program to Create, Train and Test ANN for Stock Market prediction
 *
 * Author – Wanjawa, W. B.
 * Contact – WanjawaWB@gmail.com
 * Last Revised – February 2014
 * Version – 2.10 (1 input stock at a time)
 *
 */

using System;
using System.Collections.Generic;
using System.Globalization;
using System.IO;
using System.Windows.Forms;
using FinancialMarketPredictor.Entities;
using FinancialMarketPredictor.Properties;
using FinancialMarketPredictor.Utilities;
using System.Configuration;
using System.Security.Permissions;

namespace FinancialMarketPredictor
{
publicpartialclassWinStockMarketPredictor : Form
   {
       #region Private member fields

// Default paths
privatestring varPathToStock01 = "stock01.csv";

// Predictor
privateMainProgram _predictor;

// Predict daates
privatereadonlyDateTime _predictFrom = CSVReader.ParseDate("2011-12-31");
privatereadonlyDateTime _predictTo = CSVReader.ParseDate("2012-04-01");

// Learn dates
privatereadonlyDateTime _learnFrom = CSVReader.ParseDate("2008-01-02");
privatereadonlyDateTime _learnTo = CSVReader.ParseDate("2011-12-31");

// Max/Min date that can be specified for training and predicting, specified in the AppConfig
privatereadonlyDateTime _maxDate;
privatereadonlyDateTime _minDate;

// Default parameter for number of hidden layers
privateint _hiddenLayers = 2;

// Default parameter for number of neurons in each hidden layer
privateint _hiddenUnits = 21;

// Check if there is a need in reloading the files
privatebool _reloadFiles = false;

// Start and End Time for running application
DateTime dateOne;
DateTime dateTwo;

       #endregion

// Public parameter less constructor
public WinStockMarketPredictor()
       {
InitializeComponent();

           _btnStop.Enabled = false;
```

```
      _btnExport.Enabled = false;
try
      {
        _maxDate = CSVReader.ParseDate(ConfigurationManager.AppSettings["MaxDate"]);
        _minDate = CSVReader.ParseDate(ConfigurationManager.AppSettings["MinDate"]);
      }
catch
      {
        _maxDate = DateTime.Now;                    /*Maximum specified in the csv files*/
        _minDate = CSVReader.ParseDate("2008-01-02");   /*Minimum specified in the csv files*/
      }

/*Set some reasonable default values*/
      _dtpTrainFrom.Value = _learnFrom;
      _dtpTrainUntil.Value = _learnTo;
      _dtpPredictFrom.Value = _predictFrom;
      _dtpPredictTo.Value = _predictTo;

      _dtpTrainFrom.MaxDate = _dtpTrainUntil.MaxDate = _dtpPredictFrom.MaxDate = _dtpPredictTo.MaxDate = _maxDate;
      _dtpTrainFrom.MinDate = _dtpTrainUntil.MinDate = _dtpPredictFrom.MinDate = _dtpPredictTo.MinDate = _minDate;

      _nudHiddenLayers.Value = _hiddenLayers;
      _nudHiddenUnits.Value = _hiddenUnits;
    }

// Load the form
privatevoid WinFinancialMarketPredictorLoad(object sender, EventArgs e)
    {
SetPathsInTextBoxes();  /*Set path in textboxes*/
    }

// Set paths in text boxes
privatevoid SetPathsInTextBoxes()
    {
if (File.Exists(Path.GetFullPath(varPathToStock01)))
        _tbPathToStock01.Text = Path.GetFileName(varPathToStock01);
    }

// Training callback, invoked at each iteration
// "epoch" = Iteration number
// "error" = Current error
// "algorithm" = Training algorithm
privatevoid TrainingCallback(int epoch, double error, TrainingAlgorithm algorithm)
    {
Invoke(addAction, newobject [] {epoch, error, algorithm, _dgvTrainingResults});

    }

// Start training button pressed
privatevoid BtnStartTrainingClick(object sender, EventArgs e)
    {
// Show the start time
      lblElapsedTime.Visible = false;
      lblElapsedLabel.Visible = false;
      lblStopLabel.Visible = false;
      lblCurrentTimeStop.Visible = false;
      lblTraining.Text = "Training...";
      lblTraining.Visible = true;
dateOne = DateTime.Now;
      lblCurrentTimeStart.Text = dateOne.ToString();
      lblCurrentTimeStart.Visible = true;
      lblCurrentTimeStartLabel.Visible = true;

if (_dgvTrainingResults.Rows.Count != 0)
        _dgvTrainingResults.Rows.Clear();

if (_predictor == null)
      {
        _reloadFiles = false;
if (!File.Exists(_tbPathToStock01.Text))
        {
```

```
MessageBox.Show(Resources.InputMissing, Resources.FileMissing, MessageBoxButtons.OK, MessageBoxIcon.Information);
return;
        }
      }

DateTime trainFrom = _dtpTrainFrom.Value;
DateTime trainTo = _dtpTrainUntil.Value;

if (trainFrom > trainTo)
      {
MessageBox.Show(Resources.TrainFromTrainTo, Resources.BadParameters, MessageBoxButtons.OK, MessageBoxIcon.Information);
        _dtpTrainFrom.Focus();
return;
      }
FadeControls(true);
if (_predictor == null)
      {
        varPathToStock01 = _tbPathToStock01.Text;
         Cursor = Cursors.WaitCursor;
        _hiddenLayers = (int)_nudHiddenLayers.Value;
        _hiddenUnits = (int)_nudHiddenUnits.Value;
try
        {
          _predictor = newMainProgram(varPathToStock01, _hiddenUnits, _hiddenLayers);
        }
catch (Exception ex)
        {
MessageBox.Show(ex.Message, Resources.Exception, MessageBoxButtons.OK, MessageBoxIcon.Error);
          _predictor = null;
return;
        }
finally
        {
          Cursor = Cursors.Default;
        }
      }
elseif (_reloadFiles) /*Reload training sets*/
      {
        varPathToStock01 = _tbPathToStock01.Text;

        _predictor.ReloadFiles(varPathToStock01);
        _dtpTrainFrom.MinDate = _predictor.MinIndexDate;
        _dtpTrainUntil.MaxDate = _predictor.MaxIndexDate;
      }
/*Verify if dates do conform with the min/max ranges*/
if (trainFrom < _predictor.MinIndexDate)
        _dtpTrainFrom.MinDate = _dtpTrainFrom.Value = trainFrom = _predictor.MinIndexDate;
if (trainTo > _predictor.MaxIndexDate)
        _dtpTrainUntil.MaxDate = _dtpTrainUntil.Value = trainTo = _predictor.MaxIndexDate;
TrainingStatus callback = TrainingCallback;
      _predictor.TrainNetworkAsync(trainFrom, trainTo, callback);
    }

// Predict the values
privatevoid BtnPredictClick(object sender, EventArgs e)
    {
if (_dgvPredictionResults.Rows.Count != 0)
        _dgvPredictionResults.Rows.Clear();

if (_predictor == null)  /*The network is untrained*/
      {
        _reloadFiles = false;
if (!File.Exists(_tbPathToStock01.Text))
        {
MessageBox.Show(Resources.InputMissing, Resources.FileMissing, MessageBoxButtons.OK, MessageBoxIcon.Information);
return;
        }
switch (MessageBox.Show(Resources.UntrainedPredictorWarning, Resources.NoNetwork, MessageBoxButtons.YesNoCancel,
MessageBoxIcon.Information))
        {
caseDialogResult.Yes:
```

```
break;
caseDialogResult.No:
/*Load the network*/
this.Cursor = Cursors.WaitCursor;
            _hiddenLayers = (int)_nudHiddenLayers.Value;
            _hiddenUnits = (int)_nudHiddenUnits.Value;
try
            {
              _predictor = newMainProgram(varPathToStock01, _hiddenUnits, _hiddenLayers);
            }
catch (Exception ex)
            {
MessageBox.Show(ex.Message, Resources.Exception, MessageBoxButtons.OK, MessageBoxIcon.Error);
              _predictor = null;
return;
            }
finally
            {
this.Cursor = Cursors.Default;
            }
using (OpenFileDialog ofd = newOpenFileDialog() { FileName = "NSEpredictor.ntwrk", Filter = Resources.NtwrkFilter })
            {
if (ofd.ShowDialog() == System.Windows.Forms.DialogResult.OK)
              {
try
                {
                  _predictor.LoadNeuralNetwork(Path.GetFullPath(ofd.FileName));
                }
catch
                {
MessageBox.Show(Resources.ExceptionMessage, Resources.Exception, MessageBoxButtons.OK,
MessageBoxIcon.Error);
return;
                }
              }
            }
break;
caseDialogResult.Cancel:
return;
        }
      }
DateTime predictFrom = _dtpPredictFrom.Value;
DateTime predictTo = _dtpPredictTo.Value;
if (predictFrom > predictTo)
      {
MessageBox.Show(Resources.PredictFromToWarning, Resources.BadParameters, MessageBoxButtons.OK,
MessageBoxIcon.Information);
        _dtpPredictFrom.Focus();
return;
      }

if (_predictor == null)
      {
        varPathToStock01 = _tbPathToStock01.Text;

         Cursor = Cursors.WaitCursor;
        _hiddenLayers = (int)_nudHiddenLayers.Value;
        _hiddenUnits = (int)_nudHiddenUnits.Value;
try
        {
          _predictor = newMainProgram(varPathToStock01, _hiddenUnits, _hiddenLayers);
        }
catch (Exception ex)
        {
MessageBox.Show(ex.Message, Resources.Exception, MessageBoxButtons.OK, MessageBoxIcon.Error);
          _predictor = null;
return;
        }
finally
        {
          Cursor = Cursors.Default;
        }
      }
List<PredictionResults> results = null;
try
      {
if (_reloadFiles)
        {
          varPathToStock01 = _tbPathToStock01.Text;

          _predictor.ReloadFiles(varPathToStock01);
        }
results = _predictor.Predict(predictFrom, predictTo);
      }
catch(Exception ex)
      {
MessageBox.Show(ex.Message, Resources.Exception, MessageBoxButtons.OK, MessageBoxIcon.Error);
return;
      }
foreach (var item in results)
      {
        _dgvPredictionResults.Rows.Add(item.Date.ToShortDateString(),
                        item.Stock01actual, item.Stock01predicted.ToString("F2", CultureInfo.InvariantCulture),
item.Error.ToString("F4", CultureInfo.InvariantCulture));
      }
    }

// Form closing event
privatevoid WinFinancialMarketPredictorFormClosing(object sender, FormClosingEventArgs e)
    {
if(_predictor != null)
        _predictor.AbortTraining();
    }

// Stop training
privatevoid BtnStopClick(object sender, EventArgs e)
    {
      lblTraining.Text = "Training Stopped";
// Show the stop time
dateTwo = DateTime.Now;
      lblCurrentTimeStop.Text = dateTwo.ToString();
      lblStopLabel.Visible = true;
      lblCurrentTimeStop.Visible = true;

// Show the elapsed time
var diff = dateTwo.Subtract(dateOne);
var totalTime = String.Format("{0:00}:{1:00}:{2:00}", diff.Hours, diff.Minutes, diff.Seconds);
      lblElapsedTime.Text = totalTime;
      lblElapsedTime.Visible = true;
      lblElapsedLabel.Visible = true;

FadeControls(false);
      _predictor.AbortTraining();
      _btnExport.Enabled = true;
    }

// New Path to Stock01 is selected
privatevoid TbPathToStock01MouseDoubleClick(object sender, MouseEventArgs e)
    {
OpenFileDialog ofd = newOpenFileDialog() { FileName = "stock01.csv", Filter = Resources.CsvFilter };
if (ofd.ShowDialog() == System.Windows.Forms.DialogResult.OK)
      {
        _tbPathToStock01.Text = Path.GetFullPath(ofd.FileName);
        _reloadFiles = true;
      }
    }

// Fade controls from the main form
//"fade" means If true - fade, otherwise - restore</param>
privatevoid FadeControls(bool fade)
    {
Action<bool> action = (param) =>
```

```
                 {
                   _tbPathToStock01.Enabled = param;
                   _btnStartTraining.Enabled = param;
                   _btnStop.Enabled = !param;
                   _dtpPredictFrom.Enabled = param;
                   _dtpPredictTo.Enabled = param;
                   _dtpTrainFrom.Enabled = param;
                   _dtpTrainUntil.Enabled = param;
                   _nudHiddenLayers.Enabled = param;
                   _nudHiddenUnits.Enabled = param;
                 };
Invoke(action, !fade);
    }

// Save the network, for later reuse
privatevoid BtnExportClick(object sender, EventArgs e)
    {
using(SaveFileDialog sfd = newSaveFileDialog() { FileName = "NSEpredictor.ntwrk", Filter = Resources.NtwrkFilter })
      {
if (sfd.ShowDialog() == DialogResult.OK)
        {
FileIOPermission perm = newFileIOPermission(FileIOPermissionAccess.Write, Path.GetFullPath(sfd.FileName));
try
          {
perm.Demand();
          }
catch (System.Security.SecurityException)
          {
MessageBox.Show(Resources.SecurityExceptionMessage, Resources.SecurityException, MessageBoxButtons.OK,
MessageBoxIcon.Exclamation);
return;
          }
          _predictor.ExportNeuralNetwork(Path.GetFullPath(sfd.FileName));
        }
      }
    }

// Load previously saved network
privatevoid BtnLoadClick(object sender, EventArgs e)
    {
if (!File.Exists(_tbPathToStock01.Text))
      {
MessageBox.Show(Resources.InputMissing, Resources.FileMissing, MessageBoxButtons.OK, MessageBoxIcon.Information);
return;
      }
if (_predictor == null || _predictor.Loaded == false)
      {
/*Load the network*/
this.Cursor = Cursors.WaitCursor;
        _hiddenLayers = (int)_nudHiddenLayers.Value;
        _hiddenUnits = (int)_nudHiddenUnits.Value;
try
        {
          _predictor = newMainProgram(varPathToStock01, _hiddenUnits, _hiddenLayers);
        }
catch (Exception ex)
        {
MessageBox.Show(ex.Message, Resources.Exception, MessageBoxButtons.OK, MessageBoxIcon.Error);
          _predictor = null;
return;
        }
finally
        {
this.Cursor = Cursors.Default;
        }
      }
using (OpenFileDialog ofd = newOpenFileDialog() { FileName = "NSEpredictor.ntwrk", Filter = Resources.NtwrkFilter })
      {
if (ofd.ShowDialog() == DialogResult.OK)
        {
try
```

```
          {
            _predictor.LoadNeuralNetwork(Path.GetFullPath(ofd.FileName));
            _nudHiddenLayers.Value = _predictor.HiddenLayers;
            _nudHiddenUnits.Value = _predictor.HiddenUnits;
          }
catch (System.Security.SecurityException)
          {
MessageBox.Show(Resources.SecurityExceptionFolderLevel, Resources.Exception, MessageBoxButtons.OK,
MessageBoxIcon.Error);
          }
catch
          {
MessageBox.Show(Resources.ExceptionMessage, Resources.Exception, MessageBoxButtons.OK, MessageBoxIcon.Error);
return;
          }
        }
      }
    }

// Save results
privatevoid BtnSaveResultsClick(object sender, EventArgs e)
    {
var dgvResults = _dgvPredictionResults;
SaveFileDialog ofd = newSaveFileDialog {Filter = Resources.CsvFilter, FileName = "NSEresults.csv"};
if (ofd.ShowDialog() == DialogResult.OK)
      {
CSVWriter writer = null;
try
        {
writer = newCSVWriter(ofd.FileName);
        }
catch (System.Security.SecurityException)
        {
MessageBox.Show(Resources.SecurityExceptionFolderLevel, Resources.Exception, MessageBoxButtons.OK,
MessageBoxIcon.Error);
return;
        }
catch(Exception ex)
        {
MessageBox.Show(ex.Message, Resources.Exception, MessageBoxButtons.OK, MessageBoxIcon.Error);
return;
        }
object[,] values = newobject[dgvResults.Rows.Count + 2,dgvResults.Columns.Count];
int rowIndex = 0;
int colIndex = 0;
foreach (DataGridViewColumn col in dgvResults.Columns) /*Writing Column Headers*/
        {
values[rowIndex, colIndex] = col.HeaderText;
colIndex++;
        }
rowIndex++; /*1*/

foreach (DataGridViewRow row in dgvResults.Rows) /*Writing the values*/
        {
colIndex = 0;
foreach (DataGridViewCell cell in row.Cells)
          {
values[rowIndex, colIndex] = cell.Value;
colIndex++;
          }
rowIndex++;
        }

/*Writing the results in the last row*/
writer.Write(values);
      }
    }

// Number of hidden UNITS changed
privatevoid NudHiddenUnitsValueChanged(object sender, EventArgs e)
    {
```

```
if(_predictor != null)
            {
if(MessageBox.Show(Resources.ChangedNetwork, Resources.Warning, MessageBoxButtons.OKCancel, MessageBoxIcon.Warning)
== DialogResult.OK)
                {
                    _predictor = null;
                }
            }
        }

// Number of hidden LAYERS changed
privatevoid NudHiddenLayersValueChanged(object sender, EventArgs e)
        {
if (_predictor != null)
            {
if (MessageBox.Show(Resources.ChangedNetwork, Resources.Warning, MessageBoxButtons.OKCancel,
MessageBoxIcon.Warning) == DialogResult.OK)
                {
                    _predictor = null;
                }
            }
        }

    }
}
```