UNIVERSITY OF LIVERPOOL

SCHOOL OF ENGINEERING
ENGG341 Individual Project

High Performance Computing for High Fidelity Numerical Analysis

Final Report

Muhammad Ali

Aerospace Engineering

April 2016

Supervisor: Dr Edoardo Patelli
Project No. 270

School of Engineering
University of Liverpool
Brownlow Hill, Liverpool, L69 3GH

Abstract

Artificial Neural Networks (ANNs) are highly effective at finding hidden correlations between time-series; this quality makes them particularly suited for use in domain of stock price forecasting. Majority of research papers have used ANNs for technical analysis (studying the price movements independently) but very few have used fundamental analysis (understanding correlation between economic inputs such as Revenue, Earnings and Long term debt to predict price movements of different stocks). This paper applies Artificial Neural Networks to fundamental inputs to forecast prices of different stocks and to evaluate whether such an application has major merits or potential. The findings of this paper suggest that application of ANNs to fundamental analysis for price forecasting has considerable potential but in order to fully uncover the power of this configuration, several limitations and complexities need to be properly understood and taken care of during design phase. During this investigation, ANNs are researched in detail, different configurations are carefully considered and financial domain is thoroughly discussed to get familiar with all the technical knowledge required to construct a suitable experiment. Lastly, the pitfalls, complications and limitations of this configuration (ANNs and fundamental analysis) are discussed with recommendations for improvements.

Table of Contents

Chapter 1: Literature Review 5
1.1 Artificial Neural Networks 5
1.1.1 Introduction to Machine Learning 5
1.1.2 Definition, History and Workings of Artificial Neural Networks 6
1.1.3 Learning in Artificial Neural Networks 10
1.1.4 Data Pre-Processing Techniques for an ANN 13
1.1.5 Artificial Neural Network's Topology 15
1.2 Fundamentals of Stock Price Prediction 16
1.2.2 Types of Financial Analysis 18
1.3 Financial Forecasting with ANNs 21
1.3.1 Financial Data: Challenges, Complications and Solutions. 21
1.3.2 Sources of Fundamental data 22
1.3.3 Testing Methodology in Financial ANNs 23
Chapter 2: Software Selection 24
2.1 TensorFlow 24
2.2 Walkato Enviornment for Knowledge Analysis (Weka) 25
2.3 MATLAB 26
Chapter 3: ANN Development 28
3.1 Primary Assumptions 28
3.2 ANN topology 29
3.3 Input Selection 30
3.3.1 Framework for Stock Selection 30
3.3.2 Stock Portfolio Selection 32
3.3.3 Fundamental Variables Selection 35
3.4 Data Pre-processing 38
3.4.1 The Case for Detrending: Cons and Pros 38
3.4.2 Data Pre-processing in MATLAB 38
3.5 Experimental Studies 39
3.5.1 Prediction Methodology 39
3.5.2 Understanding the Results 41
3.5.3 Experimental Results & Adjustments 42
Chapter 4: Results 44
4.1 Performance Plots 44
Appendix A: Gantt chart 50
Appendix B: Input Selection Graphs 51
Appendix C: Initial Performance Plots of Stocks 69
References 73

Table of Figures

Figure 1: A simple illustration of Machine Learning process 6
Figure 2: A biological neuron (M.Lane & Neidinger, 1994) 7
Figure 3: Basic structure of an ANN (Atsalakis & Valavanis, 2009) 8
Figure 4: Hard limited function (Azoff, 1994) 9
Figure 5: Sigmoid function (Azoff, 1994) 9
Figure 6: Pseudolinear function (Azoff, 1994) 10
Figure 7: Input neurons with specific weights (Atsalakis & Valavanis, 2009) 11
Figure 8: ANN with MLP framework (Azoff, 1994) 12
Figure 9: Apple Inc. data file taken from STOCKPUP 22
Figure 10: Data flow graph (Kartalopoulos, 1995) 24
Figure 11: WEKA GUI (Waikato) 25
Figure 12: WEKA explorer portal (Waikato) 25
Figure 13: Neural Network Time Series Tool (MATHWORKS, n.d.) 26
Figure 14: Architecture of ANN (MATHWORKS, n.d.) 30
Figure 15: Snapshot of Verizon's data file in Microsoft Excel 33
Figure 16: Snapshot of Devon's data file in Microsoft Excel 34
Figure 17: A snapshot of Merck's input file (MerckInputs.xls) 40
Figure 18: Network Architecture Settings (MATHWORKS, n.d.) 40
Figure 19: Performance plot of Merck with default settings 41
Figure 20: Performance plot of Exxon 43
Figure 21: Final Performance plot of Exxon 44
Figure 22: Final Performance plot of Chevron 45
Figure 23: Final Performance plot of Pfizer 46
Figure 24: Final Performance plot of Merck 47

Chapter 1: Literature Review

1.1 Artificial Neural Networks

1.1.1 Introduction to Machine Learning

Machine Learning (ML) is a subfield of computer science that involves study of pattern recognition and computational learning theory in artificial intelligence. Arthur Samuel gave one of the simplest yet apt definitions of this exciting field in 1959; he defined machine learning as the field of study that gives computers the ability to learn without being explicitly programmed. Essentially, machine learning explores the study and construction of algorithms that can learn from and make predictions on data.
Historically, the origins of machine learning can be traced to quest for artificial intelligence (AI). Hebbian theory, proposed by Donald Hebb in 1949, is credited with establishing the thought process for ML. Although it is a theory in neuroscience that proposes an explanation for the adaptation of neurons in brain during the learning process, it has important implications for both supervised and unsupervised machine learning techniques. Generally, machine learning techniques can be classified under two broad domains: Supervised and Unsupervised modes of learning. In supervised mode of learning, computer is presented with inputs and their desired outputs or targets. The singular goal is to learn a general rule that maps inputs to outputs. Speech recognition, handwriting recognition, spam detection, stock price prediction and direct marketing are useful examples for this mode of learning. In unsupervised mode of learning, primary goal of the learning function is to find structures or patterns in the provided unlabelled input. It is very useful mode in feature selection, finding hidden relations between provided inputs. Facebook's intelligent photo tagging mechanism and Google's keyword based image search are very well known examples for this mode of learning.

A more pertinent categorisation to the context of this project involves consideration of desired output of machine-learned system. They are as follows: Classification, Regression, Clustering and Dimensionality reduction. Classification involves division of inputs in to classes where the task of learner is to produce a model that sorts these inputs in to their respective classes. An example for this mode of learning is spam filtering where emails are assigned to either inbox or spam folder. Regression is another example of supervised learning technique where outputs are continuous rather than discrete. Clustering, an example of unsupervised mode of learning, involves division of inputs in to groups. Contrary to sorting mechanism of clustering, the groups are not known. Dimensionality reduction involves simplification of inputs before being fed to the algorithm for the learning process to begin.

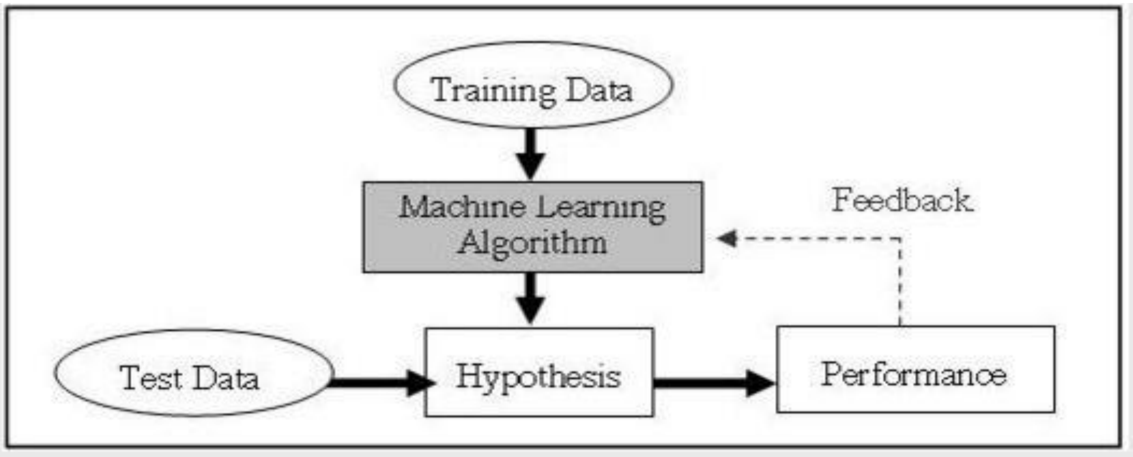


Figure 1: A simple illustration of Machine Learning process

The fundamental process of ML remains same: Training data is inputted to an ML algorithm of user's choice that commences the learning process. After a pre-defined training period where the system has studied the inputs and target outputs, a model is created which forms the basis of hypothesis validation or invalidation. Test data is inputted to the model and degree of error in results justified in conclusion or further corrections made to the model depending on the feedback. The difference, however, lies in the algorithms. They can be *regression* algorithms such as linear, logistic or stepwise, *Decision tree* algorithms such as CHAID, CHART or M5, *Bayesian* algorithms such as Naïve Bayes, Gaussian Naïve Bayes or Bayesian Network and *Artificial Neural Network* algorithms such as Perceptron, Back-propagation or Hopfield Network.

The application for this project, financial forecasting, favours the use of **Artificial Neural Network** (ANN) algorithms for their excellent prediction capacity in non-linear environments. Therefore, ANN will be studied in detail in the chapters to follow.

1.1.2 Definition, History and Workings of Artificial Neural Networks

Artificial Neural Networks (ANN), both in machine learning and cognitive science, consist of group of models which are inspired by biological neural networks (nervous system of animals, centred around brain) that are particularly useful in function approximation that depend on a large and unknown data set or inputs. The conception of ANNs can be traced back to 1943 when W. McCulloch and W. Pitts created world's first mathematical model for a biological neuron. In 1949, Donald Hebb's *The Organisation of Behaviour* highlighted a crucial point: neural networks are strengthened each time they are used. The *perceptron* model was developed next by Rosenblatt, a physiologist, in 1958; the model interconnected perceptrons and employed trial and error to alter the weights in order to initiate learning. Later in 1958, Selfridge introduced the idea of weight space to perceptron. Continuing the research on weights, Widrow and Hoff developed a mathematical method for them in 1960. Minsky and papert's book, *Perceptrons*, in 1969 bought perceptron research to an end by highlighting perceptron's inability to solve non-linear problems; it was not until development of *back propagation* algorithm in 1974 by Werbos that interest in ANN was seen again amongst the academic community and continues to flourish as advances in high performance computing and big data analytics affords researchers to test and create new variants of neural networks and their corresponding algorithms.

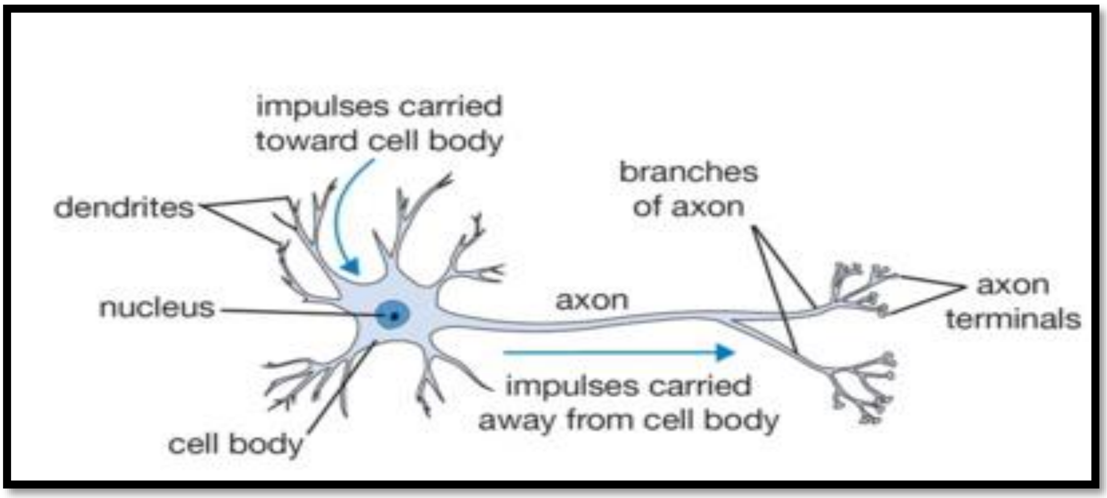


Figure 2: A biological neuron (M.Lane & Neidinger, 1994)

To understand the working of Artificial Neural Networks, it's very important to understand how biological neurons work as ANNs essentially aspire to mimic human brain.

A single neuron in the brain is an incredibly complex machine that even today we don't understand. A single "neuron" in a neural network is an incredibly simple mathematical function that captures a minuscule fraction of the complexity of a biological neuron. So to say neural networks mimic the brain, that is true at the level of loose inspiration, but really artificial neural networks are nothing like what the biological brain does.

-Andrew Ng (Associate professor at Stanford and Chief Scientist at Baidu)

Human brain approximately contains 100 billion neurons in a very dense and well-connected network. As seen in figure 2, a typical neuron contains a **cell body**, outer extensions known as **dendrites** and an **axon**. The bridge or junction that connects dendrites and axons is called **synapse.** It is estimated that a single neuron may have up to 10000 synapses and may be connected with over 1000 neurons. The breakdown of a synapse reveals its three components: *presynaptic terminal* belonging to axon, *cleft* and *postsynaptic terminal* belonging to dendrite. Research has shown that neurons compute a simple threshold calculation that is completed using biochemical and electrical signal processing. Soma or the cell body collects and combines the input signals; if the cumulative signal strength exceeds predetermined threshold then neuron sends an electrical signal along the axon. At the end of axon there are numerous branches, axon terminals as seen in figure 2, called axonic endings that allow connection to other dendrites. Upon reaching the end of axon, signal is converted in to chemical messenger called neurotransmitter that crosses the cleft in the synapses to the dendrite of the next neuron. At the postsynaptic terminal the charge generated by the neurotransmitter is weighted, determined by its relation to a particular input (Kartalopoulos, 1995). This discussion of biological neuron will suffice for future reference as more in-depth review falls outside the domain of this project.

The workings of biological neuron may stand in contrast to that of Artificial Neural Networks on basis of complexity and lack of knowledge but there are structural similarities between the two; ANNs consist of layers of interconnected neurons, each connection has a weighting, set before processing takes place, which determines the bias to certain data. Just like their counterparts, the biological neurons, the bias is set before processing takes place. The signals coming from input terminal are combined in a bias term known as *activation function*; this combined signal is called activation level that is passed on to *transformation function* that determines the input. If the activation level exceeds the set threshold then neuron fires, ensuring that output is bound; addition of this particular function to artificial neuron bounds the output at all times (Luke Biermann, 2006).

A simple model of firing is a step function that is zero (no firing) below a certain value and one (firing) above the value (M.Lane & Neidinger, 1994).
The power of neural computing lies in the threshold concept; it provides a method of transforming complex interrelationships in to simple yes-no situations. When the combination of several factors becomes extremely complex, the neuron model presents a simple yes-no node for sake of simplicity (Shachmurove, 2005)

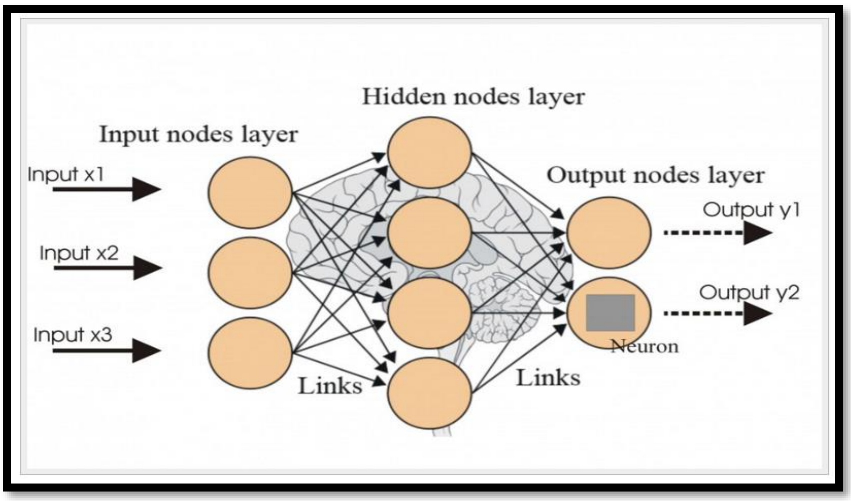


Figure 3: Basic structure of an ANN (Atsalakis & Valavanis, 2009)

The purpose of previously discussed transformation function is not only to determine neuron's output but also to bound it. There are three popular types of transformation functions namely hard limiters, sigmoid and pseudo linear.

Hard limiter function was implemented in early neural networks and is very simple, fulfilling the basic requirements of a transformation function; its value ranges from 0 to 1 that is input dependent.

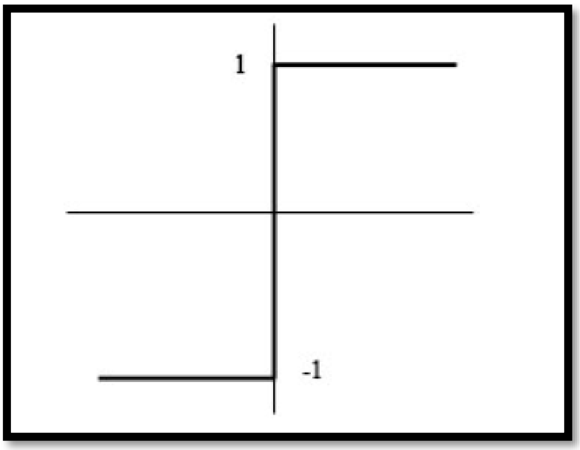


Figure 4: Hard limited function (Azoff, 1994)

Sigmoid functions represent second generation of transformation functions that are relatively better at approximation of biological process. They are used in neural networks to introduce nonlinearity in the model. A neural network element computes a linear combination of its input signals and applies a sigmoid function to the result. A particular reason for its popularity in neural network application is satisfaction of a property between the derivative and itself such that it is computationally easy to perform:

$$\frac{\partial}{\partial t} sig(t) = sig(t)\big(1 - sig(t)\big) \qquad (2.5)$$

(Taskin Kocak, 2007)

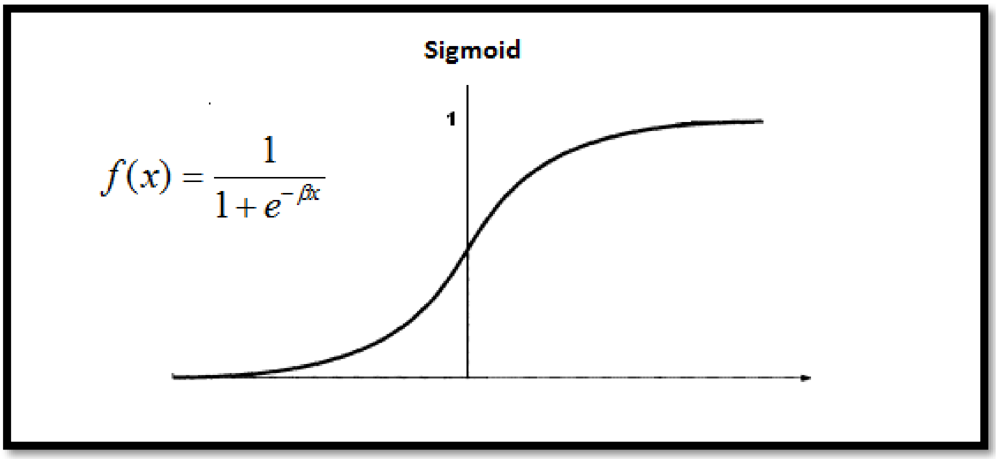


Figure 5: Sigmoid function (Azoff, 1994)

The *pseudolinear* linear function is a bridge between hard limiter and sigmoidal functions as it derives a linear approximation of sigmoidal function. (Shachmurove, 2005)

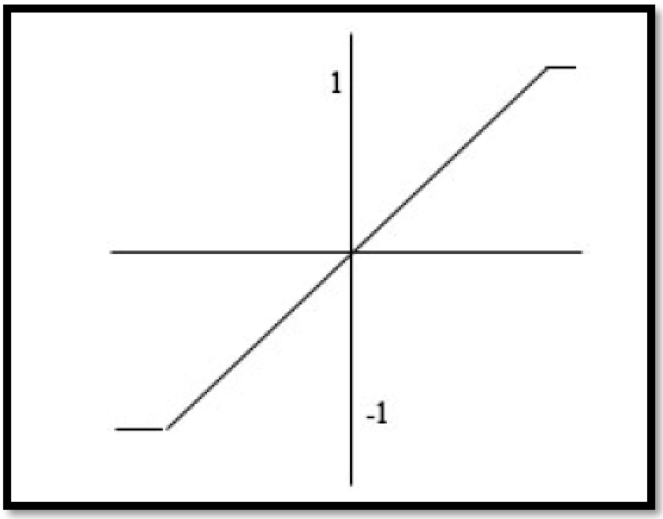


Figure 6: Pseudolinear function (Azoff, 1994)

By far, sigmoidal function is the first choice when it comes to financial applications; whether its technical time series analysis or fundamental analysis, sigmoid function is seen to be employed quite extensively (Kar) (Barber, 2005) (Bhargava & Gupta, 2008) (Zekic, 1998) (Bodis, 2004) (Hui, 2000).
Time constraints limit the scope of testing and pre-experimental studies that can be conducted to test different architectures so if majority of research papers agree with a particular setting, this paper will implement that without further analysis; there is more than sufficient evidence of sigmoidal function being applied successfully, with comparatively superior results, to financial time series problems such as stock price prediction due to its performance in non-linear environments.

So far, a basic structure of ANN has been examined which is too simple and needs more explanation on aspects of learning, training and testing. Further development of ANNs was inspired by works of Warren McCulloch and Walter Pitt, which introduced training algorithms that determine weightings for each neuron.

1.1.3 Learning in Artificial Neural Networks

For an ANN to predict anything, it needs to learn hidden patterns in data for which it must be taught first just like its biological counterpart, variation in learning methodology gives rise to different type of ANNs with their respective training techniques. The learning process starts with an error function that is expressed in terms of thresholds and weights of the neurons in the ANN; prime objective is to attain a minimum error by altering and adjusting weights of inputs at every neuron. At this stage, ANN is said to be in a steady state. Learning algorithm plays a paramount role in prediction ability of an ANN so its selection should be given due consideration with every aspect of domain (individual application) thoroughly analysed; a poorly chosen learning algorithm can have catastrophic results (Shachmurove, 2005).
Learning is largely divided in to two general categories namely *Supervised and Unsupervised learning.*

Supervised Learning

In this mode of learning, briefly discussed in introduction section, ANN's output is compared to target output and adjustments are made to neuron weights to reduce error size; ideally the objective is to reduce the error to zero. Once the network has achieved a relatively low error, understandably after a number of iterations, network training stops. In training phase, same data is inputted and weights adjusted at each iteration, before training commences, all weights of neurons are non-zero and randomly generated.

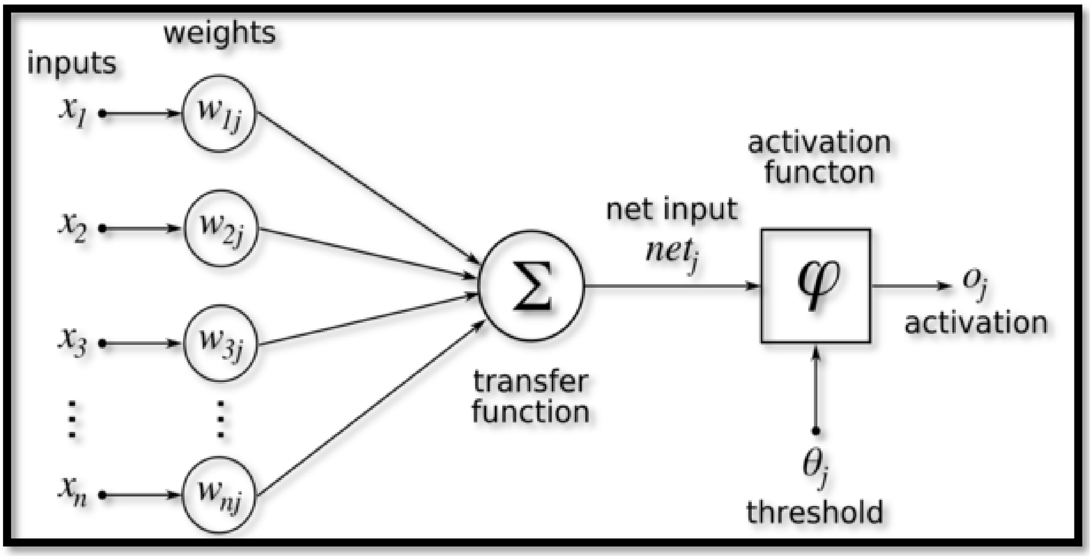


Figure 7: Input neurons with specific weights (Atsalakis & Valavanis, 2009)

Now we will examine supervised learning architecture and algorithms in detail to understand the inner workings of the learning mechanism employed by ANNs.

Perceptron or the neuron (as seen in figure 7) is the building block of the ANN architecture. It was first proposed by Frank Rosenblatt and requires supervised learning in its implementation. The perceptron values, input weights and threshold values, are fixed once training has finished and if any change is made to them, system needs to be retrained as individual customisation of values deems the training invalid; this characteristic of perceptrons gives rise to many variations. This brings us to the discussion of a very popular type of perceptron framework known as *Multi-layer perceptron.*

Multi-layer Perceptron (MLP) MLP is a framework generally implemented in a feedforward artificial neural network model where input data sets are mapped on their appropriate output data sets. An MLP consists of multiple layers of nodes in a directed graph with fully connected layers; there are types of layers called input, hidden and output. The input layer takes the raw data as its input, the hidden layer takes the output from the previous layer as its input and the output receives its data from the hidden layers (workings and input of hidden layers is unknown hence the name hidden). With the exception of input node, every other node is a neuron with a nonlinear activation function. MLP employs *backpropagation*, a supervised learning technique, for training (Hastie, Tibshirani, & Friedman, 2008).

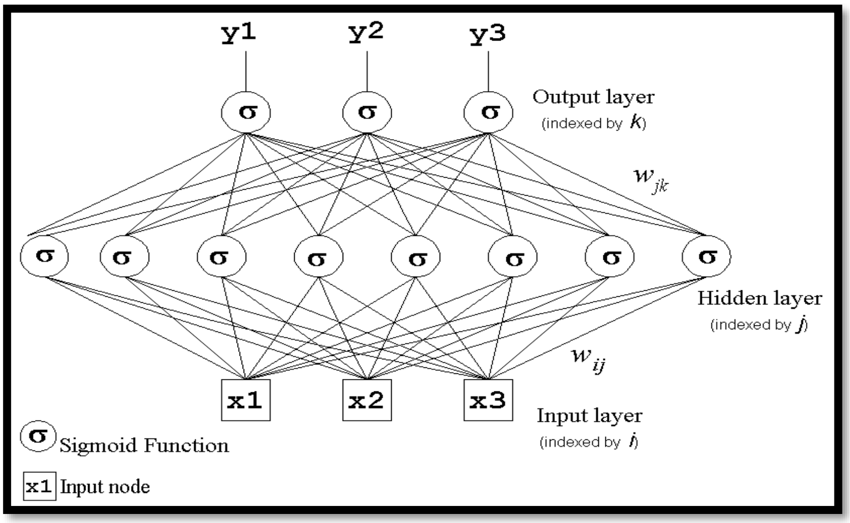


Figure 8: ANN with MLP framework (Azoff, 1994)

A number of learning algorithms have been applied to MLPs but the most suitable and popular choice remains to be Back Propagation algorithm that has been extensively used with feedforward networks (Mia, Biswas, Urmi, & Siddique, 2015) (Kartalopoulos, 1995).

Back Propagation Algorithm

Discovered by Paul Werbos in 1974, Back-propagation algorithm has been successfully applied across a diverse range of problem domains such as finance, medicine, engineering and physics (Statsoft, 2010).

Essentially the method involves weight adjustment starting from the output layer working its way back to the input later hence the name back propagation. The algorithm starts by calculating the error at the output layer relative to the target, this gives a measure of rate of change of error relative to the activation level of output neuron. Next stage involves stepping back one layer and recalculating the weights of output layer with the goal of error minimisation; this process is repeated until the input layer is reached. This process of weight adjustment continues until there is no change in weights anymore or a desired measure of error has been achieved; at this point ANN picks the next available input-target pair and the above mentioned process is repeated (Kartalopoulos, 1995).

When it comes to functionality and performance, Back-Propagation (BP) has faced criticism for its slow training nature owing to requirement of large calculations; one must be extremely cautious when choosing learning rate, as an unusually high rate will result in connection weights getting stuck in local minimum that will yield inaccurate predictions (Wani, 2013).

Irrespective of criticisms, BP has powerful optimisation tools and is known to be an excellent algorithm for generalisation. Furthermore, it has been extensively applied in stock price prediction domain (Patel & Yalamalle, 2014) (Luke Biermann, 2006) (Ayodele, Charles, Marion, & Sunday, 2012).

Unsupervised Learning

Unsupervised learning receives inputs and categorises them according to the hidden patterns it observes in them; if an input pattern can’t be categorised, a new pattern is stored to accommodate it. As discussed earlier in introduction, unsupervised mode of learning doesn't require a teacher however giving it guidelines on how to categorise patterns is imperative in order to derive meaningful results. Most common domain of this mode of learning is classification problems where the ANN employing unsupervised mode of learning needs to classify patterns in to distinct groups or classes based on the underlying characteristics of inputs. Intelligent photo sorting in operating systems of phones from companies like Samsung, Sony and Apple all use classification.

As stock price prediction lies in the domain of regression, a process of estimating relationships among variables, rather than classification, further discussion about unsupervised learning is considered to be out of scope of this project. Furthermore, unsupervised learning networks are inappropriate for financial domain and are not employed successfully in time series forecasting.

In this section, inner workings of learning methodology of an ANN were discussed; role of different components such as perceptron, learning algorithm and MLP was explained in detail. Concepts of Supervised and Unsupervised learning were further highlighted after being briefly introduced in Machine Learning section. At this stage in literature review, two things can be deduced with high certainty; a supervised neural network architecture with a corresponding learning algorithm, most likely Back-propagation (BP), will be highly appropriate for the domain of this paper, stock price prediction using fundamental data. Moreover, BP algorithm's implementation in a variety of financial domain problems, such as credit scoring and stock price forecasting, highlight its importance and functionality.

Now techniques of data pre-processing will be discussed as without the correctly processed data, even the best neural networks with the most advanced architecture have no chance of successful implementation.

1.1.4 Data Pre-Processing Techniques for an ANN

Pre-processing is one of the most crucial and vital stages in any ANN implementation; the objective is to remove noise, resolve inconsistencies and accentuate important relationships in input variables. In this section we will discuss the following three techniques that are generally applied to all time series input data regardless of nature of application.

- Normalisation
- Detrending
- Visual Inspection

In next chapter, investigation of techniques specific to financial domain, stock price forecasting to be precise, will follow to completely understand the steps required for preparing the data in order to build an ANN that is highly optimised and accurate in its predictions.

Normalisation

Stock prices are highly volatile variables that have the tendency to disturb ANNs with their dramatic fluctuations; normalisation is one of the counter measures to prevent such fluctuations. It involves standardising the range of values in a data set.

Moreover, the primary technical reason for such standardising is to make sure that values fall within the range of the transformation function otherwise there are chances that transformation function will drift towards a zero value which will bring training to a halt (known as network paralysis in this context). Another important case for normalisation is that it minimises the effect of magnitude among input variables that facilitates ANN in learning the relevant relationships (Zhang, 2003). Variables are usually normalised to achieve a zero mean and unit standard deviation. An equation for this operation as follows:

$$X_i^{scl}(t) = \frac{X_i(t) - \overline{X_i}}{\sigma_{x_i}}$$

$X_i^{scl}(t)$ stands for the scaled variable, $X_i(t)$ for the original variable and σ_{x_i} for standard deviation of $X_i(t)$, and $\overline{X_\imath}$ the mean value of $X_i(t)$ (Bodis, 2004).

In the book, Neural Networks in Business Forecasting, Peter Zhang states that a normalisation technique where input variables are scaled to fall in a specified range such as between 0 and 1 or -1 and 1 is particularly useful for modelling of ANNs (Zhang, 2003). Furthermore, both (Kartalopoulos, 1995) and (Refenes, 1994) suggest normalisation is quite the norm in financial domain.

Detrending

A process where seasonal and general trends are removed from the data as strong trends in input variables can lead to correlations and regressions that are misleading; ANN learns general features easily in comparison to actual relationships between variables. Stock market is an inherently volatile environment and huge fluctuations in stock prices are observed quite frequently owing to different macro-economic factors hence this problem is very relevant when it comes to constructing financial time series which are heavily dominated by trends. A simple way of achieving detrending in time series is to consider relative values instead of absolute that removes the linear trend:

$$y = x_t - x_{t-1}$$

To remove seasonality, following equation could be applied:

$$y = x_t - x_{t-s}$$

The s stands for time interval where similar pattern is observed.

(Bodis, 2004) employed detrending by actively implementing it whilst (Hui, 2000) (Dorsey & Sexton, 2000) doesn't make it obvious if they implemented detrending or not. A counter argument is presented by (Vanstone, Finnie, & Tan, 2004) where raw input data had better results; he argues that fragile inherent structures in time series are destroyed after data is pre-processed which has serious implications on the results.

Visual inspection

As the name implies, original time series is inspected by the forecaster for any missing values, trends, outliers or other irregularities that can adversely affect the prediction accuracy of the ANN model in visual inspection stage of construction. As such this is practise is very useful (Bodis, 2004) in deciding which pre-processing techniques should be applied to time series. Common outliers in financial domain includes stock splits where stock price is reduced by a factor of split ratio; if the stock splits are not considered in time series, ANN will interpret the dramatic fall in price as a crash when in reality it's just a realignment that should have no real implications.

All three basic data pre-processing techniques have been discussed in sufficient detail now; whilst there has been some doubt in application of detrending in financial domain, visual inspection and normalisation are a must when constructing financial time series as attested by many research journals quotes above and will therefore be applied in this paper.

1.1.5 Artificial Neural Network's Topology

Topology of an ANN involves deciding on the desired configuration of network in terms of neurons, hidden layers and inputs; this is an important step in construction of an ANN as topology directly affects the computational and prediction ability of the system.

There is no such thing as a general optimum network topology as it vastly varies between different applications and is largely paradigm-specific. Both (Bodis, 2004) and (Shachmurove, 2005) reiterate that there is no rule for optimum topology but it remains to be a very important aspect of building an efficient and accurate ANN.

The hidden neurons in the hidden layer (as seen in figure 8) are responsible for computing difficult functions known as non-separable functions. The following is a list of work of ANN researchers on hidden layer selection methodology:

- (Shih, 1994) suggested a network topology with a pyramidal shape: having greatest number of neurons in initial layers and fewer in the later ones. He suggested the number of neurons in each layer to be a number mid-way between the previous and succeeding layers to be twice the number of the preceding layer.
- (Azoff, 1994) presents a rough guideline based on theoretical conditions known as the Vapnik-Chervonenkis dimension[1] that recommends number of training data to be at least ten times the number of weights.
- Lawrence (Lawrence, 1994) has stated the following formula for determining the number of hidden neurons required in a network:
  Number of hidden neurons= training facts × error tolerance.

(M.Lane & Neidinger, 1994) recommends a three layer, fully connected network, with one hidden layer. It might be considered rudimentary and unscientific but trial and error strategy helps a lot at arriving on the system with the desired attributes (prediction threshold). However, the disadvantages include slow and painstakingly laborious process; these two disadvantages are also the reason that ANNs are not widely applied in business practise as partially optimised networks yield inconsistent results.
As part of topology discussion of ANNs, following categories will be examined in-depth:

[1] Azoff refers to an article by (Hush & Hush, 1993)

- *Analytic Estimation:* This consists of techniques that employ statistical or algebraic analysis to find the optimum number of hidden neurons; these techniques are based on study of input vector space in terms of space and dimensionality. It's a good starting point but it's unlikely that optimum configuration could be achieved by its sole application. In-depth discussion of analytic estimation methods is carried by (Refenes, 1994).
- *Constructive techniques:* These involve using trial and error method to reach the optimum number of hidden layers; the process begins with one hidden layer having one hidden neuron then another hidden neuron is added and results compared. The idea behind this addition is to achieve optimum performing ANN by adding neurons until the performance drops from the peak. As can be construed from the procedure, obvious disadvantage is slow process that yields a computationally intensive ANN.
- *Pruning Techniques:* These techniques are particularly useful in large ANNs; essentially pruning employs methodology opposite to constructive techniques. Instead of adding neurons, procedure commences with a large number of hidden neurons and layers that are progressively removed until peak performance has been achieved. According to (Michie, Spiegelhalter, & Taylor, 2009) pruning useless nodes or weights have numerous advantages on performance of ANN; smaller networks require relatively less training time and are easier to interpret.

Time constraints limit the testing phase as mentioned above so decisions have to be made by considering application of techniques in applications that are similar to the domain of this project, stock price prediction using fundamental data. Analytic estimation appears to be rudimentary and time consuming; pruning techniques are primarily employed when dealing with large ANNs that make it irrelevant to our project. Constructive techniques not only appear to be logical starting point for a small ANN that will be used in this project but also are extensively used in financial domain. (Vanstone, Finnie, & Tan, 2004)'s research paper implements constructive strategy: starting with 14 input layers and gradually building the hidden layer.

Another interesting way of investigating optimum hidden layers configuration is to employ genetic algorithms that are derived from a natural selection process which mimics biological evolution. It's basically a feature selection[2] method where irrelevant inputs are removed from system; (Dorsey & Sexton, 2000) use genetic algorithm for topology optimisation in forecasting financial time series. The goal of algorithm is to reduce the redundant connections. Initially there were 61 connections based on ten input nodes, one hidden layer with five hidden neurons and a single output neuron, following the application of genetic algorithm, the connections are substantially reduced without any decrease in predictive ability of the ANN.

1.2 Fundamentals of Stock Price Prediction

In this section, the application of this paper, stock price prediction, will be investigated in detail; traditional stock market theories and prediction methods will be

[2] In machine learning and statistics, *feature selection* is the process of selecting a subset of relevant features in inputs for the use of modal construction.

discussed. Then different forms of input data will be presented which has proved useful in previous research journals for study of trends in stock prices.

1.2.1 The Efficient Market Hypothesis

The Efficient Market Hypothesis or EMH is a well-known and controversial theory in capital markets. It exists in three forms of magnitude: strong, semi-strong and weak.

The strong form asserts that markets follow a random walk[3] pattern that cannot be predicted by following or observing past prices; it believes that stock price reflects all data, both public and private, available regarding the company and the price readjusts itself as the new data is published. This interpretation of Efficient Market Hypothesis or EMH implies that it is impossible to beat the market[4] in the long term.

Semi-strong form of EMH suggests that stock price reflects all publicly released information; therefore, if an investor has access to information that is not known to public, he or she can benefit from it. However, this private information implied is the very definition of insider information[5] that is illegal and severely punished in most countries. Two prevalent forms of analysis, fundamental and technical, use data that is publicly available hence they will fail to ear above average returns on investments (Clarke, Jandik, & Mandelker).

The weak form of EMH argues that prices fully incorporate the information implicit in the sequence of past prices therefore employing technical analysis to beat the market is nearly impossible as it only reflects past price movement in stock market (Dimson & Mussavian, 1998).

There are many examples of both individuals and financial firms that have consistently beaten the market in the long term; these occurrences form the basis for strong criticism of EMH particularly the strong and semi-strong form. The likes of Benjamin Graham, Warren Buffet, George Soros and David Swenson have earned above market returns on investments for decades, all through legal means.

Another interesting and relatively new field of study is behavioural economics; this method applies psychological insights in to human behaviour to explain economic decision-making. In fact, the psychology approaches to stock market trading represent important alternatives to EMH. (Oprean, 2012) challenges the validity of three of the most fundamental premises that constitute EMH: the rationality of the individual, the idea that economic science is globally efficient and the idea that the economic process must be cast away. The prerequisites that lay at the basis of theory of efficient markets are not real. The hypothesis that investors are completely rational and always process instantaneously and correctly all information is surely unrealistic as defining rationality is a matter of individual perspective.

[3] Random walk is a financial term that is used to characterise a price series where all subsequent price changes represent random departures from previous prices (Malkiel, 2003).

[4] Beat the Market is a financial jargon that means earning above average returns on investments in stock market.

[5] Insider information is a non-public fact regarding the plans or condition of a publicly traded company that could provide a financial advantage when used to bull or sell stocks.

1.2.2 Types of Financial Analysis

In the domain of financial forecasting, there are two dominant schools of thought: *Technical* and *Fundamental* analysis. As the domain of this project revolves around prediction based on fundamental analysis, technical analysis will not be discussed in detail as it falls outside the context of this paper.

Technical analysis is a security analysis methodology employed for forecasting the stock prices through study of past market data, primarily price and volume. Most practitioners believe in the 90-10 rule: market is ten per cent logical and ninety per cent psychological. Emphasis is given to psychological factors; forecasting reports are generated by solely considering past price movements and patterns. Technical analysis is primarily used for short term trading as in long term, market prices tend to reflect the fundamentals and realign them accordingly.

Fundamental analysis is the cornerstone of intelligent investing which determines the health of a company by examining core numbers such as income statements, earning releases, balance sheet and other economic measures.

"A stock is not just a ticker symbol or an electronic blip; it is an ownership interest in an actual business, with an underlying value that does not depend on its share price"

-**Benjamin Graham** (The Intelligent Investor)

No discussion of fundamental analysis is complete without mentioning Benjamin Graham, British-born American economist who is considered the father of value investing[6]. Some of Graham's most famous disciples include legendary value investors such as Warren Buffet, Peter Lynch, Mohnish Pabrai and Joel Greenblatt. Graham's "The Intelligent Investor" and "Security Analysis" are considered as the best books ever written on value investment.

Fundamental analysis asserts the importance of treating the stock ownership akin to owing a small proportion of the company hence the investor should be cognizant of all factors which affects share prices; fundamental data includes company earnings, revenue figures, dividends yields, assets, liabilities, long term debt and book value to name a few. Macro-economic data consisting of employment figures, interest rate and inflation can also aid in analysis. Basically the philosophy behind fundamental analysis is opposite to that of technical analysis: Ninety per cent logical and ten per cent psychological (Graham, 1973).

Equity analysts largely consider fundamental data for creating forecasting reports, a fact that provided initial motivation for conducting an exclusive fundamental analysis of stocks for this paper.

Fundamental Analysis: *Types of data*

[6] Value investing involves buying securities (bonds, stocks) that are undervalued by employing fundamental analysis.

A thorough introduction to fundamental data set is imperative to understand the varying influence they have on stock prices. The following is a list[7] of fundamental data that will be used in creating an ANN in the development phase of this project:

1) *Assets*: An asset is a resource with economic value that an individual, corporation or country owns or controls with the expectation that it will provide future benefit.

2) *Liabilities:* A liability comprises of a company's legal debt that arises during the course of business operations. Liabilities include loans, accounts payable, mortgages, deferred revenues and accrued expenses.

3) *Shareholders' equity:* Shareholders equity is a firm's assets minus its total liabilities; it represents the degree to which a company is financed through common and preferred shares.

   Shareholders Equity= Total Assets – Total Liabilities

4) *Goodwill:* Goodwill is an intangible asset that arises as a result of the acquisition of one company by another for a premium value. The value of a company's brand name, customer base, good customer relation, good employer relations and any patents or proprietary technology represents goodwill.

5) *Long-term debt:* Long term debt comprises of loans and financial obligations lasting over one year; it will consist of any financing or leasing obligations that are to come due in a greater than 12-month period.

6) *Revenue:* Revenue is the amount of money that a company actually receives during a specific period.

7) *Earning:* Earnings are the amount of profit that a company produces during a specific period that is usually defined as a quarter (3 months' calendar) or a year.
8) *Earnings per share (EPS):* EPS is defined as the portion of a company's profit allocated to each outstanding share of common stock; it's an indicator of a company's profitability.

9) *Dividend per share:* Dividend per share or DPS is defined as the sum of declared dividends for every ordinary share issued.

10) *Return on Assets (ROA):* ROA is defined as an indicator of how profitable a company is relative to its total assets; it's a measure of how efficient management is at using its assets to generate earnings.

11) *Return on Earnings (ROE):* ROE is defined as the amount of net income returned as a percentage of shareholders' equity; it's a measure of corporation's profitability as it shows how much profit a company generates with the money shareholders have invested.

[7] The definitions are taken from Investopedia (www.investopedia.com).

12) *Book value of equity per share (BVPS):* BVPS is a financial measure that represents a per share assessment of the minimum value of a company's equity.

$$\text{BVPS}=\frac{Value\ of\ Common\ Equity}{Number\ of\ Shares\ Outstanding}$$

13) *Price to book (P/B) ratio:* P/B is a ratio used to compare a stock's market value to its book value; a lower P/B ratio could mean that the stock is undervalued or something is fundamentally wrong with the company.

14) *Price-Earnings (P/E) ratio:* P/E is the ratio for valuing a company that measures its current share price relative to its per share earnings; generally, a high P/E ratio means that investors are anticipating higher growth in the future.

15) *Net margin:* Net margin is defined as the percentage of revenue remaining after all operating expenses such as interest, taxes and preferred stock dividends have been deducted from a company's total revenue.

16) *Asset turnover:* Asset turnover ratio is the ratio of the value of a company's sales revenues generated relative to the value of its assets.

1.3 Financial Forecasting with ANNs

Financial time series are characterized by high degree of noise and volatility that deems the use of traditional computing methods inappropriate. On the contrary, ANNs are known to handle the financial prediction problem rather well relatively (Machiel & Ballini, 2008); they are particularly good at discovering non-linear patterns in time series data and have been extensively applied to technical analysis (Bodis, 2004) (Shachmurove, 2005).

Although majority of ANN applications in financial domain employ technical analysis, there are a number of notable examples of successful application of ANNs to stock forecasting using fundamental analysis; one reason for this disparity is the complexities which arise in application of fundamental data which needs more meticulous consideration rather than just creating a time series based primarily on past price and volumes.

(Emir, Dincer, & Timor, 2012) uses an ANN with 14 fundamental inputs to predict stock prices of ISE[8] 30 companies, (Vanstone, Finnie, & Tan, 2004) uses fundamental inputs such as market price, book value, ROE and dividend pay-out ratio to create an ANN with largely successful results, (Luke Biermann, 2006) follows a similar pattern of using fundamental data inputs (explained previously in section 2.2) to construct an ANN but his literature review and extent of data pre-processing techniques used is very thorough with some very useful insights.

1.3.1 Financial Data: Challenges, Complications and Solutions.

Although ANNs are highly optimised to use with financial problems, they do have some very specific challenges that limit their use in this domain. Existence of *missing, inconsistent and unlikely* values in the time series present serious issues in the training phase and distort results to a large extent. As the historical databases stretch back as far as early 1960s, a cut-off timeline has to be identified that has the least amount of missing and inconsistent values.

Missing values in financial time series can be the replaced by either using a donor value from the existing sample or predicting the value by employing a fairly reliable and accurate statistical model. Both these methods, besides being time consuming and tedious, are prone to affect the prediction of ANN and impact the final result. Financial forecaster has to ensure, for the sake of accuracy of results, that his/her time series is free from such anomalies.

Unlikely values are those that are correct but contextually illogical. So if a time series of a company A has daily stock price fluctuation (rise or fall of price during trading hours) of around 20 to 30 percent, a rise or fall of let’s say 200 percent will be a highly unrealistic scenario which will demand the attention of the forecaster. In such cases, unlikely values should be treated like missing values.

The ultimate aim is to find the time period and respective stocks with no missing, inconsistent and unlikely values to preserve the accuracy of this project.

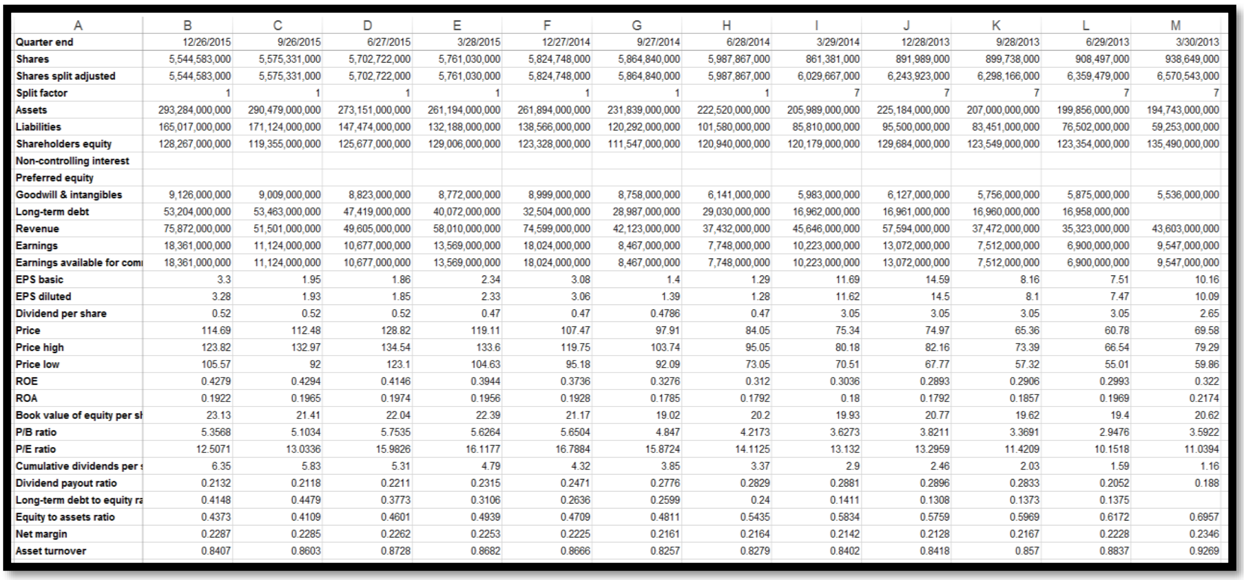
[8] ISE stands for Istanbul Stock Exchange.

1.3.2 Sources of Fundamental data

One of the most important aspects of consideration in constructing an ANN is procurement of accurate data; irrespective of the type of analysis being employed by the forecaster, technical or fundamental, without the right data inputted in to an ANN, chances of getting any meaningful results are impossible.

In financial world, Yahoo and Bloomberg provide reliable and accurate data to both analysts and organisations. Although the vast majority of data available from these websites is both accurate and free but it falls under the technical analysis category; stock price values (both low and high), book value, moving averages, order book, market capitalisation and average daily volume are easily retrievable from MATLAB's data feed toolbox. For a forecaster conducting a technical analysis, these values are more than sufficient to devise a prediction platform.

The very reason why fundamental analysis is avoided is because of scarcity of free fundamental data available online. After researching for more than two weeks, a reliable source of fundamental data with all required values was found. STOCKPUP (stockpup.com) contains 10-Q[9] SEC filings of S&P 100[10] and beyond S&P companies; the data is completely free and available in both raw CSV and Excel format.



A	B	C	D	E	F	G	H	I	J	K	L	M
Quarter end	12/26/2015	9/26/2015	6/27/2015	3/28/2015	12/27/2014	9/27/2014	6/28/2014	3/29/2014	12/28/2013	9/28/2013	6/29/2013	3/30/2013
Shares	5,544,583,000	5,575,331,000	5,702,722,000	5,761,030,000	5,824,748,000	5,864,840,000	5,987,867,000	861,381,000	891,989,000	899,738,000	908,497,000	938,649,000
Shares split adjusted	5,544,583,000	5,575,331,000	5,702,722,000	5,761,030,000	5,824,748,000	5,864,840,000	5,987,867,000	6,029,667,000	6,243,923,000	6,298,166,000	6,359,479,000	6,570,543,000
Split factor	1	1	1	1	1	1	1	7	7	7	7	7
Assets	293,284,000,000	290,479,000,000	273,151,000,000	261,194,000,000	261,894,000,000	231,839,000,000	222,520,000,000	205,989,000,000	225,184,000,000	207,000,000,000	199,856,000,000	194,743,000,000
Liabilities	165,017,000,000	171,124,000,000	147,474,000,000	132,188,000,000	138,566,000,000	120,292,000,000	101,580,000,000	85,810,000,000	95,500,000,000	83,451,000,000	76,502,000,000	59,253,000,000
Shareholders equity	128,267,000,000	119,355,000,000	125,677,000,000	129,006,000,000	123,328,000,000	111,547,000,000	120,940,000,000	120,179,000,000	129,684,000,000	123,549,000,000	123,354,000,000	135,490,000,000
Non-controlling interest												
Preferred equity												
Goodwill & intangibles	9,126,000,000	9,009,000,000	8,823,000,000	8,772,000,000	8,999,000,000	8,758,000,000	6,141,000,000	5,983,000,000	6,127,000,000	5,756,000,000	5,875,000,000	5,536,000,000
Long-term debt	53,204,000,000	53,463,000,000	47,419,000,000	40,072,000,000	32,504,000,000	28,987,000,000	29,030,000,000	16,962,000,000	16,961,000,000	16,960,000,000	16,958,000,000	
Revenue	75,872,000,000	51,501,000,000	49,605,000,000	58,010,000,000	74,599,000,000	42,123,000,000	37,432,000,000	45,646,000,000	57,594,000,000	37,472,000,000	35,323,000,000	43,603,000,000
Earnings	18,361,000,000	11,124,000,000	10,677,000,000	13,569,000,000	18,024,000,000	8,467,000,000	7,748,000,000	10,223,000,000	13,072,000,000	7,512,000,000	6,900,000,000	9,547,000,000
Earnings available for com	18,361,000,000	11,124,000,000	10,677,000,000	13,569,000,000	18,024,000,000	8,467,000,000	7,748,000,000	10,223,000,000	13,072,000,000	7,512,000,000	6,900,000,000	9,547,000,000
EPS basic	3.3	1.95	1.86	2.34	3.08	1.4	1.29	11.69	14.59	8.16	7.51	10.16
EPS diluted	3.28	1.93	1.85	2.33	3.06	1.39	1.28	11.62	14.5	8.1	7.47	10.09
Dividend per share	0.52	0.52	0.52	0.47	0.47	0.4786	0.47	3.05	3.05	3.05	3.05	2.65
Price	114.69	112.48	128.82	119.11	107.47	97.91	84.05	75.34	74.97	65.36	60.78	69.58
Price high	123.82	132.97	134.54	133.6	119.75	103.74	95.05	80.18	82.16	73.39	66.54	79.29
Price low	105.57	92	123.1	104.63	95.18	92.09	73.05	70.51	67.77	57.32	55.01	59.86
ROE	0.4279	0.4294	0.4146	0.3944	0.3736	0.3276	0.312	0.3036	0.2893	0.2906	0.2993	0.322
ROA	0.1922	0.1965	0.1974	0.1956	0.1928	0.1785	0.1792	0.18	0.1792	0.1857	0.1969	0.2174
Book value of equity per sh	23.13	21.41	22.04	22.39	21.17	19.02	20.2	19.93	20.77	19.62	19.4	20.62
P/B ratio	5.3568	5.1034	5.7535	5.6264	5.6504	4.847	4.2173	3.6273	3.8211	3.3691	2.9476	3.5922
P/E ratio	12.5071	13.0336	15.9826	16.1177	16.7884	15.8724	14.1125	13.132	13.2959	11.4209	10.1518	11.0394
Cumulative dividends per s	6.35	5.83	5.31	4.79	4.32	3.85	3.37	2.9	2.46	2.03	1.59	1.16
Dividend payout ratio	0.2132	0.2118	0.2211	0.2315	0.2471	0.2776	0.2829	0.2881	0.2896	0.2833	0.2052	0.188
Long-term debt to equity ra	0.4148	0.4479	0.3773	0.3106	0.2636	0.2599	0.24	0.1411	0.1308	0.1373	0.1375	
Equity to assets ratio	0.4373	0.4109	0.4601	0.4939	0.4709	0.4811	0.5435	0.5834	0.5759	0.5969	0.6172	0.6957
Net margin	0.2287	0.2285	0.2262	0.2253	0.2225	0.2161	0.2164	0.2142	0.2128	0.2167	0.2228	0.2346
Asset turnover	0.8407	0.8603	0.8728	0.8682	0.8666	0.8257	0.8279	0.8402	0.8418	0.857	0.8837	0.9269

Figure 9: Apple Inc. data file taken from STOCKPUP

Each datasheet has 30 variables: **Shares, Split factor, Assets, Liabilities, Shareholders equity, Non-controlling interest, Preferred equity, Goodwill & Intangibles, Long-term debt, Revenue, Earnings, EPS basic, Earnings available for common stockholders, EPS diluted, Dividend per share, Price, Price high, Price low, ROE, ROA, Book value**

[9] Form 10-Q is a quarterly report mandated by United States Federal Securities and Exchange Commission to be filed by publicly traded corporations.

[10] S&P 100 index is a stock market index of United States stocks overseen by Standard & Poor's.

of equity per share, P/B ratio, P/E ratio, Cumulative dividends per share, Dividend payout ratio, Long-term debt to equity ratio, Equity to assets ratio, Net margin, Asset turnover. Most companies have complete data from 1994 to 2015, more than enough period to cover both training and testing phases. It's observed that after 1997, the missing and inconsistent values in the dataset of major companies decreases significantly therefore the data sampling will start from 2000 to 2015 with close inspection given again in the development stage to eliminate the chances of including any missing, inconsistent and unlikely values.

1.3.3 Testing Methodology in Financial ANNs

Before deciding upon a methodology for prediction in this project, it's imperative to study and understand the effectiveness of some of the most successful and widely used methods in stock testing. Supervised learning with Back-propagation algorithm remains to be the leading configuration; price is a known target output and financial inputs such as revenue, earnings, debt and assets understand the trend in its fluctuation. Feedforward networks are most implemented type of neural networks in financial domain (Emir, Dincer, & Timor, 2012). Both in technical and fundamental analysis domain, use of sigmoidal transfer function as the primary error function is widespread and mostly encouraged (Bodis, 2004) (Lawrence, 1994) (Luke Biermann, 2006). Furthermore, use of single hidden layer has known to give excellent prediction performance (Crone, 2004). In terms of input selection for the ANN, there is a unanimous vote towards choosing few yet well-related inputs for superior results; cluttering the input layer with a large number of inputs will decrease the prediction ability of the system.

In this extensive literature review, a thorough overview of every important aspect of an ANN was discussed; financial theories and fundamental inputs were stated and explained. At this point, it's very clear that the learning algorithm of this project will be *backpropagation* along with *sigmoid* function being the activation function. Data will be sourced from Stockpup.com and pre-processing techniques such as *normalisation* and *visual inspection* of time series will be essential to build an accurate ANN. The types of fundamental variables to be employed by this paper will be determined in the development phase with help of *price-variable plots* and visual inspection of data.

The next stage in this investigation is to select a software package to implement and construct an ANN.

Chapter 2: Software Selection

Software selection lies at the core of this project, as without viable software with the desired ANN and its corresponding learning algorithms, making any kind of predictions is impossible. As the aim of this project is to explore the relationships between fundamental data and stock price rather than waste precious time on creating an ANN from the scratch, it is imperative to find a software package that incorporates a variety of algorithms and is relatively easy to implement and modify. The following sections will explore promising software packages and assess their advantages and disadvantages to select the most optimum package for this project.

2.1 TensorFlow

TensorFlow was developed by researchers and engineers working on the Google brain team within Google's Machine Intelligence research organisation, essentially TensorFlow is an open source software library for numerical computation using data flow graphs.

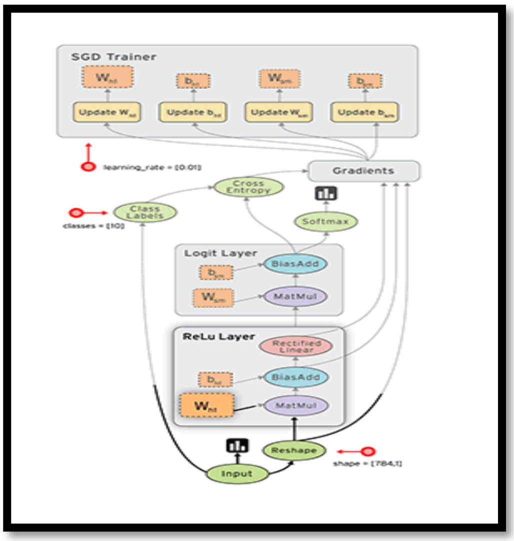


Figure 10: Data flow graph (Kartalopoulos, 1995)

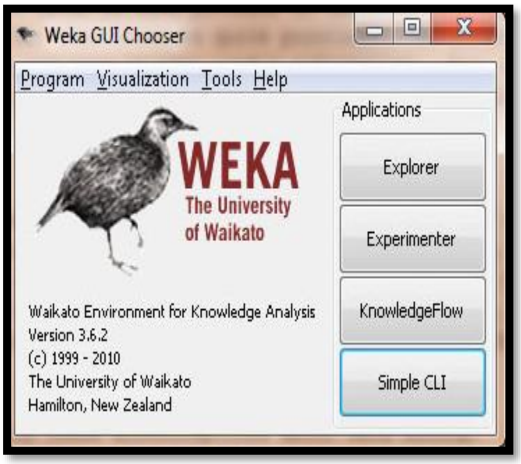
One of the major reasons for including TensorFlow in potential software list is its open source extensive library of APIs, mostly written in C++. Furthermore, in case the ANN requires parallel computing to speed up training time, TensorFlow has a variety of options for assigning compute elements to designated workstations with multiple cores.

Major disadvantage of this software package from Google includes steep learning curve for implementation of C++ to construct an ANN.

2.2 Walkato Environment for Knowledge Analysis (Weka)

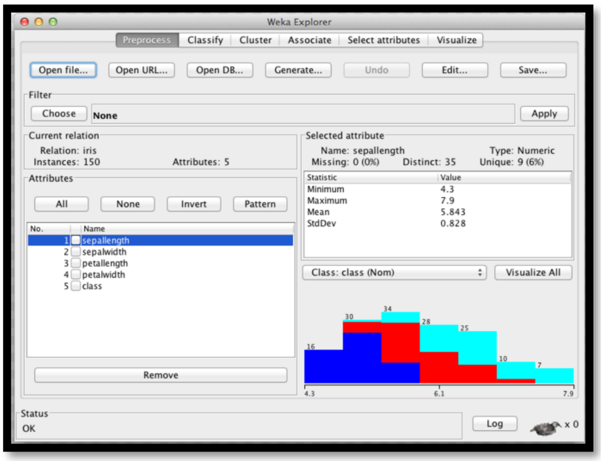
Weka is a popular suite of machine learning software written in Java that was developed at the University of Walkato, New Zealand.

Weka is a workbench that contains a collection of visualisation tools and algorithms for data analysis and predictive modelling, coupled with graphical interfaces for easy access to these functions. Like TensorFlow, it's a free source package that helps in minimising costs associated with this project. Moreover, little to no coding is required as the GUI (Graphical User Interface) is easy to use and learning to work with it takes less than a week. Another major advantage of Weka is its comprehensive collection of pre-processing techniques that save a lot of time in preparing the data for analysis.



Waikato Environment for Knowledge Analysis
Version 3.6.2
(c) 1999 - 2010
The University of Waikato
Hamilton, New Zealand

Figure 11: WEKA GUI (Waikato)



Filter: None

Current relation
Relation: iris
Instances: 150
Attributes: 5

Attributes

No.		Name
1	□	sepallength
2	□	sepalwidth
3	□	petallength
4	□	petalwidth
5	□	class

Selected attribute
Name: sepallength
Type: Numeric
Missing: 0 (0%)
Distinct: 35
Unique: 9 (6%)

Statistic	Value
Minimum	4.3
Maximum	7.9
Mean	5.843
StdDev	0.828

Class: class (Nom)

Status
OK

Figure 12: WEKA explorer portal (Waikato)

To start the analysis, the data[11] needs to be opened in the explorer environment of Weka; depending on types of pre-processing techniques required for a particular data set, algorithms are chosen from a drop down menu in Classify, Cluster, Select Attributes[12], Visualise and Forecast tabs. In terms of productivity and usability, Weka

[11] Weka supports the Attribute-Relation File Format (ARFF)

[12] Attribute is a piece of information which determines the properties of a field or tag in a database.

is an extremely valuable and efficient package as it takes away the coding part and let the user concentrate on investigating the relationships and patterns in data.

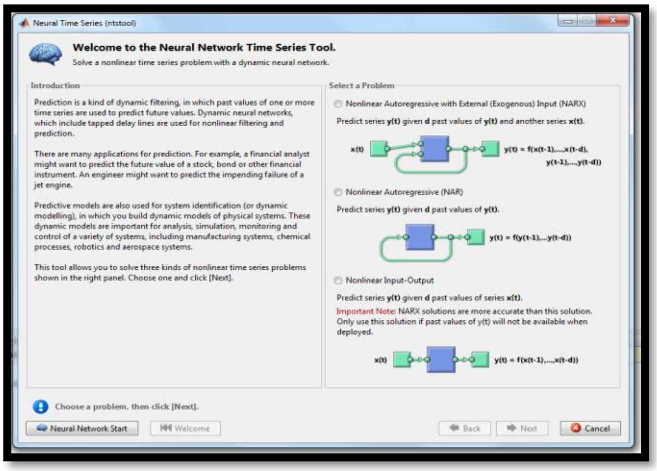
Weka has a dedicated Neural Network GUI that allows structure manipulation of multi-layer perceptions and hidden layers that affects the training of network.

Time series manipulation in Weka environment is of notable importance and has been extensively employed with successful results (Mitchell, 1995) (Gornall, 2013). A dedicated time series portal makes pre-processing and analysing the data relatively easy as well as existence of powerful visualisation tools that considerably help in spotting irregularities such as inconsistent and missing values.

In terms of disadvantages, critics argue that Weka is predominantly a visualisation software that is not optimised for ANN modelling where complex networks are involved that require high level of customisation. As far as this project is concerned, these factors are not relevant to the requirements of this project.

2.3 MATLAB

MATLAB is a multi-paradigm numerical computing environment developed by Mathworks that allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of UI (User Interface) and interaction with programmes written in other languages such as C, C++, Java and Python.



Neural Time Series (ntstool)

Welcome to the Neural Network Time Series Tool.

Solve a nonlinear time series problem with a dynamic neural network.

Introduction

Prediction is a kind of dynamic filtering, in which past values of one or more time series are used to predict future values. Dynamic neural networks, which include tapped delay lines are used for nonlinear filtering and prediction.

There are many applications for prediction. For example, a financial analyst might want to predict the future value of a stock, bond or other financial instrument. An engineer might want to predict the impending failure of a jet engine.

Predictive models are also used for system identification (or dynamic modelling), in which you build dynamic models of physical systems. These dynamic models are important for analysis, simulation, monitoring and control of a variety of systems, including manufacturing systems, chemical processes, robotics and aerospace systems.

This tool allows you to solve three kinds of nonlinear time series problems shown in the right panel. Choose one and click [Next].

Select a Problem

○ Nonlinear Autoregressive with External (Exogenous) Input (NARX)

Predict series **y(t)** given **d** past values of **y(t)** and another series **x(t)**.

○ Nonlinear Autoregressive (NAR)

Predict series **y(t)** given **d** past values of **y(t)**.

○ Nonlinear Input-Output

Predict series **y(t)** given **d** past values of series **x(t)**.

Important Note: NARX solutions are more accurate than this solution. Only use this solution if past values of y(t) will not be available when deployed.

Choose a problem, then click [Next].

Figure 13: Neural Network Time Series Tool (MATHWORKS, n.d.)

In terms of Neural Network functionality, MATLAB offers the required network architecture and the training algorithms specific to the domain of this project; feedforward neural network with backpropagation learning algorithm are part of the toolbox. Furthermore, if time efficient dynamic networks are needed to reduce training time, non-linear autoregressive (NARX), layer recurrent, Elman and Hopfield networks can be selected from the menu (as shown in figure 3.3). Training algorithms such as Levenberg-Marquardt and resilient backpropagation (Rprop) along with sigmoid error function are built in to the toolbox. One of the most significant and unique advantages for MATLAB is integration of data pre-processing techniques such as principal component analysis[13], discretisation and normalisation of data within required range; all these settings can be automatically turned on for all further analysis on time series.

All three software packages are promising but in terms of usability, ease and efficiency, WEKA and MATLAB trump Google's TensorFlow; coding in C++ requires a lot of time and debugging times for an ANN are extremely high for a project of this scale where data manipulation is quite important. WEKA offers a user-friendly environment that is very easy to deploy but lacks the data pre-processing techniques that MATLAB offers. Data manipulation is very essential for any machine learning project so with superior data pre-processing features, MATLAB is appropriately suited for the needs of this paper and will be the main software for constructing the ANN.

[13] Principal Component Analysis is a statistical procedure that involves finding the linear combination of a set of variables that has maximum variance and removing its effect.

Chapter 3: ANN Development

3.1 Primary Assumptions

The assumptions for this project were briefly discussed in literature review section but at this stage it's imperative to state them in detail for future reference. The time constraints for this project makes it impossible for every single configuration to be pre-examined and tested with all input settings. A very thorough literature review was carried specifically to address this pitfall; majority of research journals were carefully examined to decide upon the optimum settings. As discussed in literature review, input selection and data pre-processing are very sensitive and important stages in ANN construction so there will be no compromise in their individual implementation depending on chosen inputs.

The following assumptions are derived from literature review, which will help to reduce the time required to implement an ANN for financial forecasting:

- The type of learning mode (supervised or unsupervised), learning algorithm (back-propagation or Support vector machines) and error function (sigmoidal or hard limited) will be determined by research journals with similar settings and sufficiently high degree of successful results.
- As constructing an ANN from the scratch is beyond the scope of this project, a suitable software package (MATLAB) with all the learning algorithms and error functions will be selected by carefully weighing the advantages and disadvantages of the environment.
- The most successful and time efficient data pre-processing techniques (Normalisation and visual inspection) will be applied to the inputs. Literature review will play an important part in determining the types of pre-processing techniques needed.
- The input selection stage will take in to consideration results of known fundamental analysis forecasting research papers and improve upon the structure by carefully studying the weaknesses and suggestions.

There are two very important reasons to employ these assumptions in our project. Firstly, they will save precious time and allow the completion of this project in the required timeframe. Secondly, they will help to improve the quality of this project by avoiding the mistakes committed by other authors specifically in input selection and data pre-processing sections.

3.2 ANN topology

In this section, each component of the ANN will be stated with reasons given for its selection; type of ANN, learning algorithm, activation function, weights adjustment, and learning rate will be decided to provide an initial framework for experimental studies. **These initial settings are derived from the literature review carried in earlier section**.

- A feedforward neural network with backpropagation learning algorithm will be implemented in MATLAB environment to find the best prediction settings. This combination is a well-known and highly successful foundation for any technical or fundamental analysis project as nearly all the articles, research journals and books reviewed in literature review section has this combination in their initial experimental studies.
- Activation function introduces non-linearity to ANN and in financial domain sigmoidal function is the single most popular choice. Almost 90% of all research journals reviewed had sigmoidal activation function in their ANN.
- It is important to state here that in the basic configuration of MATLAB's time series toolbox, backpropagation algorithm is not included in the training algorithm options; Levenberg-Marquardt, Bayesian Regularisation and Scaled Conjugate Gradient come as standard. Although these algorithms require less training time and are known to be computationally less intensive, their application in financial domain is not clearly established. A free-source code[14] for Backpropagation algorithm will be incorporated to train ANN.
- Choosing a value of learning rate is one of the most crucial steps at this stage, as it will affect both the training time and prediction quality. At this point, an educated guess based on literature review will suffice; training rate that corresponds to optimum performance, after training with chosen inputs, will be used for final settings. Most research papers have used values ranging from 0.15 to 0.45 with acceptable results. For experimental studies, we shall start with a learning rate of **0.3.**
- Using **1000** epochs[15] for training is standard both in MATLAB's Neural Network (NN) settings and research journals have stuck to using between 900 to 1500 epochs for training.
- The selection of hidden layers and their corresponding hidden neurons is purely a trial and error exercise. During literature review, it was found that for financial domain, a single hidden layer (successfully implemented by both (M.Lane & Neidinger, 1994) and (Luke Biermann, 2006)) is well suited to accurately understand structural relationships between inputs and predict prices. This project will implement a single hidden layer but experiment with the number of neurons to optimise performance. Initially MATLAB's standard number of neurons, **10**, will be used. However, if the ANN shows significantly poor prediction results, more hidden layers will be added.

[14] "MLP Neural Network with Backpropagation" is a code contributed by Hesham Eraqi which is available to use by anyone under the BSD license. The code can be found on MATLAB: http://www.mathworks.com/matlabcentral/fileexchange/54076-mlp-neural-network-with-backpropagation

[15] An epoch is a measure of the number of times all of training vectors are used once to update the weights.

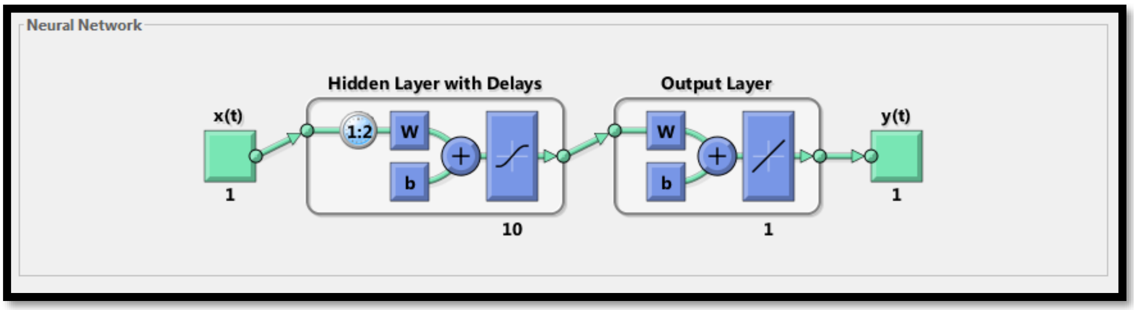


Figure 14: Architecture of ANN (MATHWORKS, n.d.)

3.3 Input Selection

In this section, input selection will be discussed in detail; the selection methodology for this project will be highlighted and different companies analysed to select the right stocks. Furthermore, the most relevant fundamental inputs will be chosen to initiate the experimental studies. As stated earlier, all the input data will be sourced from stockpup.com, one of the few websites providing reliable and free fundamental data.

3.3.1 Framework for Stock Selection

When it comes to stock selection for fundamental analysis, understanding the distinction between asset intensive and non-asset intensive companies is of paramount importance. Asset/Capital intensive industries require relatively large amount of money and financial resources to produce a good or service. Example of such industries includes oil production, transport and telecommunications. Conversely, non-asset/capital intensive industries don't require large amount of financial resources to produce a good or service. Examples include software industry, investment services and Media.

Nearly all of fundamental stock prediction research papers apply the well-known asset intensive vs. non-asset intensive companies' division strategy; a portfolio of asset intensive and non-asset intensive companies is selected with their respective fundamental inputs presented to the ANN.

During the literature review, a number of approaches were identified for input selection. The approaches, briefly discussed, are as follows:

- (Campbell & Shiller, 2001) uses price-earnings and dividend-price ratios to predict price changes in US aggregate annual data from 1871 to 2000. No specific industry is targeted but these two ratios are shown to be of high importance in general price forecasting.
- (Fifield, Power, & Sinclair, 2002) employs division of economic fundamentals in to local and global economic factors such as GDP, inflation, money and industrial production. By using Principal Component Analysis, relevant factors are included in to the final input time series to forecast index returns of 13 ESMs[16] over the period 1987-96. Once again this paper fails to differentiate

[16] ESM stands for Emerging Market Data.

between market or industry segments and indiscriminately presents all the 13 inputs to the pre-selected fundamental data.

- (Vanstone, Finnie, & Tan, 2004) uses Aby filter[17] rule to choose the fundamental economic factors that are then applied to ASX 200 index, Australia's top 200 companies, to forecast future movement.
- (Luke Biermann, 2006) employs asset intensive vs. non-asset intensive method of stock selection followed by pruning algorithms to select the most relevant fundamental inputs for ANN.
- (Emir, Dincer, & Timor, 2012) selects a mix of both fundamental and technical variables to forecast price movements in ISE 30[18] index.

The general pattern in fundamental research methodology can now be easily grasped: tweak the time series by matching highly relevant fundamental factors with an index consisting of companies from all market sectors. Stock selection, the method of choosing a select group of companies, is not given its due importance. In terms of highlighting pitfalls, no one has been as comprehensive as (Luke Biermann, 2006). This paper highlighted a number of very crucial flaws that formed the basis for this paper's stock selection methodology.

1. The correlation between fundamental data and the share price is dependent on the size of share price lag[19].
2. Fundamental data is anything that gives information about the company or the business environment it operates in. As a result, such a set is large, diverse and extremely industry specific. Different data can be applicable or effective at different times.
3. Finding the optimum configuration requires large number of experiments and the results often are not conclusive. Experiments should be designed with computational expense in mind as well as a pre-decided error tolerance threshold.

These shortcomings are further validated by works of Benjamin Graham and Robert J. Shiller. Benjamin's "The Intelligent Investor" is regarded by many as the foremost authority on value investing (investing based on company's fundamentals). Professor Shiller is a Nobel Prize winning economist who teaches economics at Yale University; Professor Shiller's Financial Markets, a 23 lectures online module, was consulted as part of literature review.

Therefore, the following methodology is proposed after reviewing well-known research papers and YouTube lectures:

- **The stocks chosen for this project will take in to account specific industries such as Oil & Gas, Pharmaceuticals and Telecommunications instead of following the basic asset vs. non-asset intensive selection strategy.**

[17] Aby filter rule buys stocks under the following conditions: PE<10; Market Price < Book Value; ROE>12; Dividend Payout Ratio < 25%.

[18] ISE 30 or Istanbul Stock Exchange is an index of Turkey's top 30 companies.

[19] Share price lag is to identify at what point of time in future the price truly reflects the fundamental factors of a company.

- **For each chosen segment, independent studies in form of price graphs will be created to identify the most optimum and custom fundamental factors to construct input time series. A “one size fits all” approach won’t be continued in this paper due to sufficient proof of its failure in dynamic, real-time environment.**
- **No fundamental factors will be chosen based on previous research papers; every variable will be thoroughly investigated before its addition to final input settings.**
- **For share price lag or time delay as called in Neural Network Toolbox, MATLAB’s default settings will be used to initiate experimental studies and adjusted accordingly if required.**
- **Consequently, based on mean square error (MSE), time delay will be altered to get the best results for each market segment.**
- **Companies that have a strong brand name influence such as Facebook, Coca-Cola and Apple will be avoided as most often stock prices are driven by social media hype not by the underlying fundamental factors. Most research papers confirmed this observation.**

Next sub-section will discuss stock selection in terms of companies and their respective industries; final selection of companies will be stated and adequately justified in the end.

3.3.2 Stock Portfolio Selection

As discussed earlier, number one priority during this stage will be to collect data that has no missing values which saves significant time during data pre-processing stage. All data will be sourced from stockpup.com.

The website has fundamental data available on all S&P100[20] companies so the most time consuming part of this stage will be to thoroughly look for missing values such as unexplained fluctuations in revenue, PE and ROE ratios, earnings and assets.

It was observed that the data before 2000 had a lot of missing values so the time frame for data collection, based on visual inspection of major stocks, was 2000-2015.

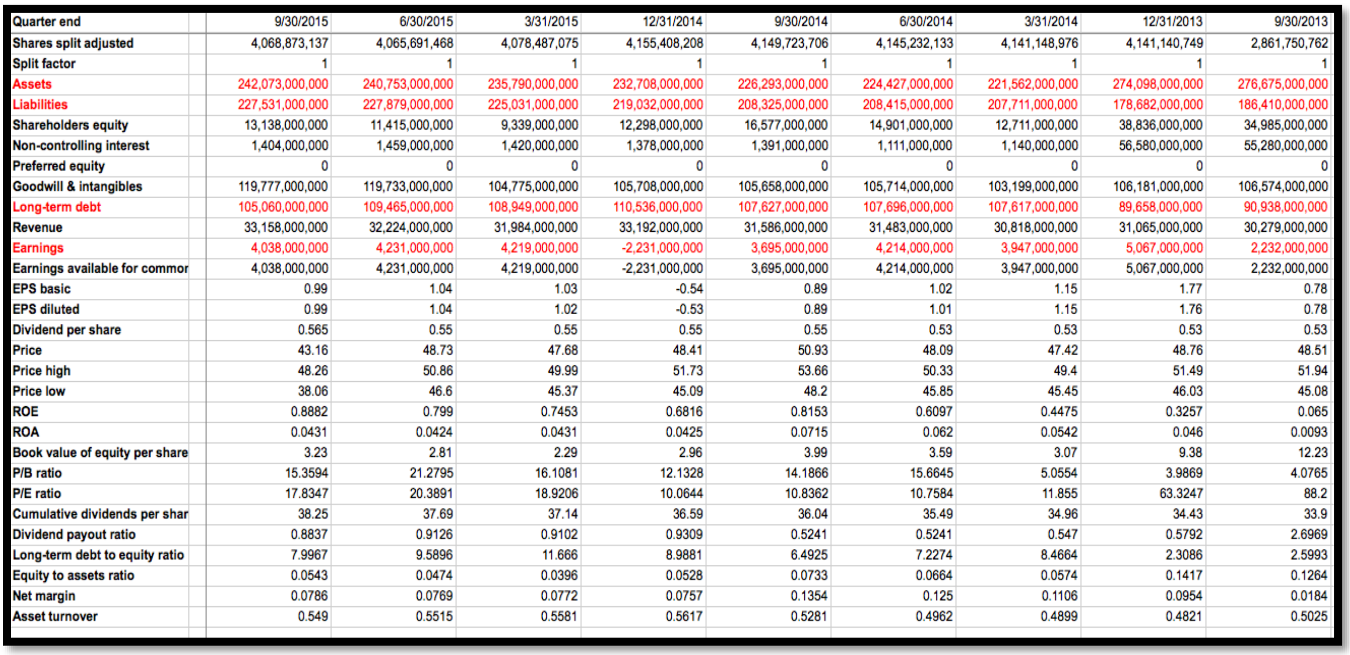
[20] S&P100 is a stock market index of United States stocks maintained by Standard & Poor’s.

The following stocks were found to have no missing values:

- ✓ **Exxon Corporation**
- ✓ **Chevron Corporation**
- ✓ **Verizon Communications Inc.**
- ✓ **Pfizer Inc.**
- ✓ **Merck & Co. Inc.**
- ✓ **Devon Energy Corporation**

In terms of industry, the companies belonged to following two sectors: **Oil & Gas** and **Pharmaceuticals**.

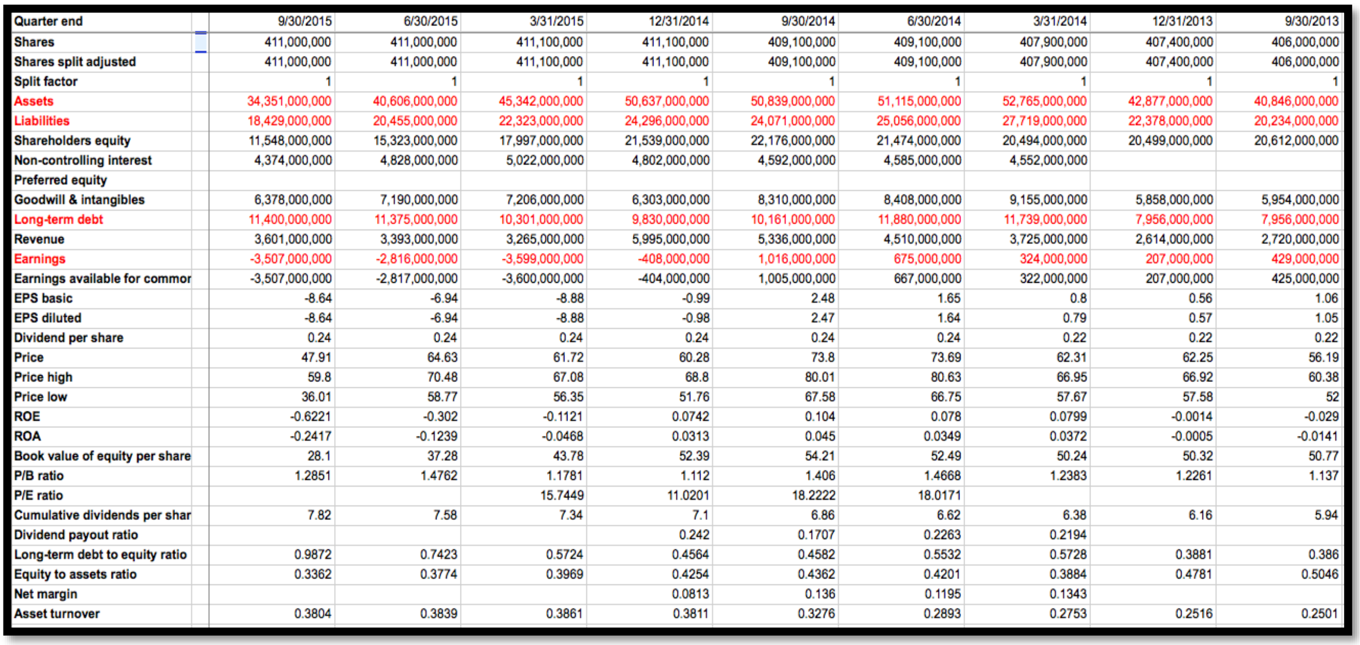
Every company has the following fundamental variables (only the important ones stated for sake of brevity): **Shares, Split adjusted, Split factor, Assets Liabilities, Shareholders equity, Long term debt, Revenue, Earnings, EPS basic, EPS diluted, Price, ROE, ROA, Book value of equity per share, P/B ratio, P/E ratio, Dividend pay-out ratio, Long term debt to equity ratio, Equity to assets ratio, Net margin and Asset turnover.**[21]



Quarter end	9/30/2015	6/30/2015	3/31/2015	12/31/2014	9/30/2014	6/30/2014	3/31/2014	12/31/2013	9/30/2013
Shares split adjusted	4,068,873,137	4,065,691,468	4,078,487,075	4,155,408,208	4,149,723,706	4,145,232,133	4,141,148,976	4,141,140,749	2,861,750,762
Split factor	1	1	1	1	1	1	1	1	1
Assets	242,073,000,000	240,753,000,000	235,790,000,000	232,708,000,000	226,293,000,000	224,427,000,000	221,562,000,000	274,098,000,000	276,675,000,000
Liabilities	227,531,000,000	227,879,000,000	225,031,000,000	219,032,000,000	208,325,000,000	208,415,000,000	207,711,000,000	178,682,000,000	186,410,000,000
Shareholders equity	13,138,000,000	11,415,000,000	9,339,000,000	12,298,000,000	16,577,000,000	14,901,000,000	12,711,000,000	38,836,000,000	34,985,000,000
Non-controlling interest	1,404,000,000	1,459,000,000	1,420,000,000	1,378,000,000	1,391,000,000	1,111,000,000	1,140,000,000	56,580,000,000	55,280,000,000
Preferred equity	0	0	0	0	0	0	0	0	0
Goodwill & intangibles	119,777,000,000	119,733,000,000	104,775,000,000	105,708,000,000	105,658,000,000	105,714,000,000	103,199,000,000	106,181,000,000	106,574,000,000
Long-term debt	105,060,000,000	109,465,000,000	108,949,000,000	110,536,000,000	107,627,000,000	107,696,000,000	107,617,000,000	89,658,000,000	90,938,000,000
Revenue	33,158,000,000	32,224,000,000	31,984,000,000	33,192,000,000	31,586,000,000	31,483,000,000	30,818,000,000	31,065,000,000	30,279,000,000
Earnings	4,038,000,000	4,231,000,000	4,219,000,000	-2,231,000,000	3,695,000,000	4,214,000,000	3,947,000,000	5,067,000,000	2,232,000,000
Earnings available for commor	4,038,000,000	4,231,000,000	4,219,000,000	-2,231,000,000	3,695,000,000	4,214,000,000	3,947,000,000	5,067,000,000	2,232,000,000
EPS basic	0.99	1.04	1.03	-0.54	0.89	1.02	1.15	1.77	0.78
EPS diluted	0.99	1.04	1.02	-0.53	0.89	1.01	1.15	1.76	0.78
Dividend per share	0.565	0.55	0.55	0.55	0.55	0.53	0.53	0.53	0.53
Price	43.16	48.73	47.68	48.41	50.93	48.09	47.42	48.76	48.51
Price high	48.26	50.86	49.99	51.73	53.66	50.33	49.4	51.49	51.94
Price low	38.06	46.6	45.37	45.09	48.2	45.85	45.45	46.03	45.08
ROE	0.8882	0.799	0.7453	0.6816	0.8153	0.6097	0.4475	0.3257	0.065
ROA	0.0431	0.0424	0.0431	0.0425	0.0715	0.062	0.0542	0.046	0.0093
Book value of equity per share	3.23	2.81	2.29	2.96	3.99	3.59	3.07	9.38	12.23
P/B ratio	15.3594	21.2795	16.1081	12.1328	14.1866	15.6645	5.0554	3.9869	4.0765
P/E ratio	17.8347	20.3891	18.9206	10.0644	10.8362	10.7584	11.855	63.3247	88.2
Cumulative dividends per shar	38.25	37.69	37.14	36.59	36.04	35.49	34.96	34.43	33.9
Dividend payout ratio	0.8837	0.9126	0.9102	0.9309	0.5241	0.5241	0.547	0.5792	2.6969
Long-term debt to equity ratio	7.9967	9.5896	11.666	8.9881	6.4925	7.2274	8.4664	2.3086	2.5993
Equity to assets ratio	0.0543	0.0474	0.0396	0.0528	0.0733	0.0664	0.0574	0.1417	0.1264
Net margin	0.0786	0.0769	0.0772	0.0757	0.1354	0.125	0.1106	0.0954	0.0184
Asset turnover	0.549	0.5515	0.5581	0.5617	0.5281	0.4962	0.4899	0.4821	0.5025

Figure 15: Snapshot of Verizon's data file in Microsoft Excel

[21] For review of definitions, refer to section 2.2.2 where they are discussed in detail.



Quarter end	9/30/2015	6/30/2015	3/31/2015	12/31/2014	9/30/2014	6/30/2014	3/31/2014	12/31/2013	9/30/2013
Shares	411,000,000	411,000,000	411,100,000	411,100,000	409,100,000	409,100,000	407,900,000	407,400,000	406,000,000
Shares split adjusted	411,000,000	411,000,000	411,100,000	411,100,000	409,100,000	409,100,000	407,900,000	407,400,000	406,000,000
Split factor	1	1	1	1	1	1	1	1	1
Assets	34,351,000,000	40,606,000,000	45,342,000,000	50,637,000,000	50,839,000,000	51,115,000,000	52,765,000,000	42,877,000,000	40,846,000,000
Liabilities	18,429,000,000	20,455,000,000	22,323,000,000	24,296,000,000	24,071,000,000	25,056,000,000	27,719,000,000	22,378,000,000	20,234,000,000
Shareholders equity	11,548,000,000	15,323,000,000	17,997,000,000	21,539,000,000	22,176,000,000	21,474,000,000	20,494,000,000	20,499,000,000	20,612,000,000
Non-controlling interest	4,374,000,000	4,828,000,000	5,022,000,000	4,802,000,000	4,592,000,000	4,585,000,000	4,552,000,000		
Preferred equity									
Goodwill & intangibles	6,378,000,000	7,190,000,000	7,206,000,000	6,303,000,000	8,310,000,000	8,408,000,000	9,155,000,000	5,858,000,000	5,954,000,000
Long-term debt	11,400,000,000	11,375,000,000	10,301,000,000	9,830,000,000	10,161,000,000	11,880,000,000	11,739,000,000	7,956,000,000	7,956,000,000
Revenue	3,601,000,000	3,393,000,000	3,265,000,000	5,995,000,000	5,336,000,000	4,510,000,000	3,725,000,000	2,614,000,000	2,720,000,000
Earnings	-3,507,000,000	-2,816,000,000	-3,599,000,000	-408,000,000	1,016,000,000	675,000,000	324,000,000	207,000,000	429,000,000
Earnings available for commor	-3,507,000,000	-2,817,000,000	-3,600,000,000	-404,000,000	1,005,000,000	667,000,000	322,000,000	207,000,000	425,000,000
EPS basic	-8.64	-6.94	-8.88	-0.99	2.48	1.65	0.8	0.56	1.06
EPS diluted	-8.64	-6.94	-8.88	-0.98	2.47	1.64	0.79	0.57	1.05
Dividend per share	0.24	0.24	0.24	0.24	0.24	0.24	0.22	0.22	0.22
Price	47.91	64.63	61.72	60.28	73.8	73.69	62.31	62.25	56.19
Price high	59.8	70.48	67.08	68.8	80.01	80.63	66.95	66.92	60.38
Price low	36.01	58.77	56.35	51.76	67.58	66.75	57.67	57.58	52
ROE	-0.6221	-0.302	-0.1121	0.0742	0.104	0.078	0.0799	-0.0014	-0.029
ROA	-0.2417	-0.1239	-0.0468	0.0313	0.045	0.0349	0.0372	-0.0005	-0.0141
Book value of equity per share	28.1	37.28	43.78	52.39	54.21	52.49	50.24	50.32	50.77
P/B ratio	1.2851	1.4762	1.1781	1.112	1.406	1.4668	1.2383	1.2261	1.137
P/E ratio			15.7449	11.0201	18.2222	18.0171			
Cumulative dividends per shar	7.82	7.58	7.34	7.1	6.86	6.62	6.38	6.16	5.94
Dividend payout ratio				0.242	0.1707	0.2263	0.2194		
Long-term debt to equity ratio	0.9872	0.7423	0.5724	0.4564	0.4582	0.5532	0.5728	0.3881	0.386
Equity to assets ratio	0.3362	0.3774	0.3969	0.4254	0.4362	0.4201	0.3884	0.4781	0.5046
Net margin				0.0813	0.136	0.1195	0.1343		
Asset turnover	0.3804	0.3839	0.3861	0.3811	0.3276	0.2893	0.2753	0.2516	0.2501

Figure 16: Snapshot of Devon's data file in Microsoft Excel

Although the stocks of both these companies exhibit no missing values in the required time frame, they both exhibit negative retained earnings[22] throughout the required timeframe. As discussed earlier in literature review, ANNs are quite sensitive to extreme changes and unless carefully programmed can easily misinterpret variables and give out inaccurate predictions. For example, if figure 4.3 is taken in to consideration, the earnings stand at 1,016,000,000 on 9/30/2014 and -408,000,000 on 12/31/2014; in real life, one or few quarters of negative retained earnings makes little difference if the revenue is steady or showing strong growth but in terms of modelling, ANN can misinterpret that as a crash and downgrade its predictions for this stock. Moreover, if these earnings occurred once or twice, they could have been removed, treated as outliers, and the remaining data used as final input but in both these stocks, specifically Devon Energy Corporation, these figures are very common. As such, it is decided to remove both these stocks from further consideration and continue with the investigation of the remaining four companies in the data set.

Oil and Gas:

- ➢ **Exxon Corporation**
- ➢ **Chevron Corporation**

Pharmaceuticals:

- ➢ **Pfizer Inc.**
- ➢ **Merck & Co. Inc.**

[22] **Negative Retained Earnings**, often recorded on the balance sheet as accumulated deficit, means the company has more retained losses over time than accumulated net income.

3.3.3 Fundamental Variables Selection

The next step in stock selection procedure is to see how price responds to different fundamental factors; price vs. fundamental variable graphs will be created in excel to study the relationships in detail. The full set of graphs will be attached in appendix.

Exxon Corporation

- The strongest correlation is seen in **Asset** vs. Share price, **Shareholders equity** vs. Share price, **Liabilities** vs. Share price, **Revenue** vs. Share price, **ROE** vs. Share price, **ROA** vs. Share price and **Net margin** vs. Share price graphs. Surprisingly **P/E ratio** vs. Share price shows inverse relationship: as the P/E ratio drops, share price increases. Recalling from literature review, a high P/E ratio indicates that investors are anticipating high growth that should lead to a direct relationship. This point further highlights the danger of using "one size fits all" approach that distorts results.
- **Long term debt vs. Share price** graph obeys the usual rule in asset intensive companies: as the debt rises, the share price drops. This variable is one of the most important indicators of fundamental workings of any asset intensive company. If large sums of money are being borrowed relative to revenue, the investors dump their share in fear of losing their investment if company goes bankrupt. Although in this case, Exxon being an extremely reputable and valuable firm, it takes approximately 10 billion dollars (refer to Exxon graphs in appendix) of additional long term debt for investors to respond in form of share price drop.
- **Earnings** vs. Share price graph again shows irregular behaviour in terms of price movements. For an asset intensive firm, earnings are very essential and are supposed to fluctuate price movements; a direct relationship is assumed between these two variables. In Exxon's case, for the first 10 years, stock price moves in direct proportion with earnings and partially follows the trend from 2010 to 2015. As ANN explores hidden relationships, only experimental studies can conclusively answer these questions, as there is a limit to what can be understood during this visual inspection stage.
- **Long term debt to equity ratio** vs. Share price and **Asset turnover** vs. Share price graphs don't show any relationships that could justify their inclusion in the dataset.

Chevron Corporation

- The strongest correlations are seen in **Asset** vs. Share price, **Liabilities** vs. Share price, **Long term debt** vs. Share price, **Revenue** vs. Share price, **EPS** vs. Share price, **Book value of equity per share** vs. Share price, **Long term debt to equity ratio** vs. Share price and **Net margin** vs. Share price graphs. **P/E** ratio vs. Share price graph shows the same inverse relation as in the case of Exxon.
- **Earnings** vs. Share price graph shows direct relationship; although the earnings are very volatile with extreme fluctuations observed, the price movement follows the trend close enough to qualify for final selection.
- **ROE** vs. Share price shows extreme irregularity and as such can't be deemed suitable for selection. Similarly, **Equity to assets ratio** vs. Share price and **Asset turnover** vs. Share price graphs fail to exhibit any meaningful relationship.

Pfizer Inc.

- As suspected, there is a significant difference within the asset intensive market segment; both Oil and Gas and Pharmaceutical industries respond to differently to fundamental factors.
- **Asset** vs. Share price, **Liabilities** vs. Share price and **Shareholders equity** vs. Share price graphs exhibit an inversely proportional relationship for most part of the time frame. **Long term debt** vs. Share price graph exhibits a predicted inverse relationship but there is a surprising result in **Revenue** vs. Share price graph: a general drop in share price is observed as revenue increases. **Earnings** vs. Share price graph is too volatile to show any meaningful relationship for consideration. Both **ROE** vs. Share price and **ROA** vs. Share price graphs exhibit a directly proportional relationship.
- **Net margin** graph shows slight correlation; Net margin and price follow a direct relationship for most part of the time frame. **Asset turnover** vs. Share price graph fails to show any meaningful relation that is worthy of further investigation.

Merck & Co. Inc.

- Once again the dynamic and volatile nature of stocks is highlighted in this example; **Assets** vs. Share price, **Liabilities** vs. Share price and **Shareholders equity** vs. Share price graphs show little to no relationship that can ascertain any meaningful trend. In fact, even the most robust indicator, long term debt vs. Share price graph, is showing little change when debt increases or decreases.
- **ROE** vs. Share price, **ROA** vs. Share price, **P/B ratio** vs. Share price, **Long term debt to equity ratio** vs. Share price and **Net margin** vs. Share price graphs show strong correlation.

Final Inputs based on initial analysis:

- ➢ Exxon and Chevron, both part of Oil & Gas sector, have some striking similarities: Values of **assets**, **liabilities**, **revenue**, **P/E ratio**, **Long term debt** and **Net margin** cause significant movement in stock price: both downwards and upwards.
- ➢ The effect of **ROA** figures on Exxon's stock price is more noticeable than that on Chevron but Return on Assets play an important role in any asset intensive company. Furthermore, strong correlation has been noticed in asset vs. Share price graphs of both companies so inclusion of ROA ratio seems to be the logical step.
- ➢ The relationship between **Earnings** and Share price is left to final ANN as at this stage partial correlation is established that is enough to qualify for selection.
- ➢ Pfizer Inc. and Merck & Co. Inc. are part of Pharmaceutical sector that has noticeable differences in terms of responding to different fundamental variables. **ROE** vs. Share price, **ROA** vs. Share, **Long-term debt** vs. Share price, **Net margin** vs. Share price graphs show strong correlation with price.
- ➢ Both **Revenue** vs. Share price and **Earnings** vs. Share price graphs fail to show noticeable correlation. Now the dynamics of pharmaceutical industry need to be considered before plainly discarding both these variables from

input time series; once a new drug is released, spike in earnings is observed as seen in both graphs. Stock price then responds to how the drug is perceived in medical community, in addition to movements in earnings. It can be argued that price is driven by FDA[23] ratings too that doesn't necessarily, at least in the short term, represent a fundamental input. Nonetheless, both these variables, Revenue and Earnings, have huge implications on stock price and shall be added to the input time series. More thorough investigation will be left to the ANN.

Final inputs for Oil & Gas sector

- ✓ **Assets**
- ✓ **Liabilities**
- ✓ **Revenue**
- ✓ **P/E ratio**
- ✓ **Long term debt**
- ✓ **Net margin**
- ✓ **ROA**
- ✓ **Earnings**

Final inputs for Pharmaceutical sector

- ✓ **ROA**
- ✓ **ROE**
- ✓ **Long term debt**
- ✓ **Net margin**
- ✓ **Earnings**
- ✓ **Revenue**

[23] **FDA** stands for Food and Drug Administration. FDA is a federal agency of the United States Department of Health and Human Services.

3.4 Data Pre-processing

3.4.1 The Case for Detrending: Cons and Pros

During the visual inspection stage data was checked for any outliers in terms of missing and unlikely values. Share price vs. Fundamental variable graphs were plotted to establish relationships and more importantly reduce the input set by deleting the variables that showed no relevance.

During the literature review, data pre-processing techniques were discussed for application at a later stage. *Detrending*, a technique of removing general and seasonal trends from data, was briefly explained. In case of Oil & Gas companies, the need for detrending diminishes due to market dynamics of this industry. Oil exploration is one of the few businesses that have a steady and growing demand throughout the year; particularly there is no "season" for oil, modern day economies are heavily dependent on it throughout the year. This fact is reflected in data set of Exxon (attached in appendix), no strong seasonal trends in all chosen input variables are observed.

In case of pharmaceutical companies, seasonal trends are observed due to structural differences. Pharmaceutical sector is characterised by takeovers and mergers; relatively speaking, more share price volatility is observed in graphs of Pfizer and Merck than Exxon and Chevron. If these successive takeovers and mergers, purely characteristic of this particular sector, are to be seen as a seasonal/general irregularity and values are removed to adjust the time series then there is a high risk of compromising results.

A strong case against detrending is found in (Vanstone, Finnie, & Tan, 2004); this paper argues that detrending destroys the fragile inherent structures found in time series. Financial sector is perhaps the most sensitive and difficult environment to model and forecast; it is decided that detrending will not be carried on the data set of this paper.

3.4.2 Data Pre-processing in MATLAB

One of the biggest reasons for MATLAB's selection was its superior data pre-processing techniques that are seamlessly integrated in neural network toolbox. Prediction methodology will be discussed in detail in the next section. The following techniques will be applied to the data:

- Scaling all inputs and targets in the range of -1 to 1.
- Normalising the mean and standard deviation of the training set.
- Automated data division carried by MATLAB's ANN toolbox.

As these functions are built in to NN toolbox, once data set is divided in to training and testing set, it will take a few minutes to execute these actions, saving a lot of time that could be dedicated to other sections.

3.5 Experimental Studies

This is the most important section of this project where initial results will allow the progression towards final settings that are considered the most optimum for chosen inputs. The following bullet points will state what will be discussed in this section:

- The prediction methodology for this project will be discussed in detail
- The data, chosen variables of Exxon, Chevron, Pfizer and Merck, will be divided in to training and testing set
- Initial settings of ANN will be discussed with a variety of diagrams and graphs to explain the whole process
- A framework for understanding the results will be laid out
- Initial results will be discussed
- The ANN settings will be tweaked to arrive upon the best model; this will involve successive training sessions.

3.5.1 Prediction Methodology

Now that we have chosen the initial stocks and their corresponding fundamental inputs, it's time to discuss the method of prediction for this project. Like most other projects of similar nature, the underlying method won't be changed as it works.

As the data was gathered in a different way, in terms of sector rather than crudely separating asset intensive and non-asset intensive firms, two sets of inputs will be presented to the ANN; one belonging to the Oil & Gas sector consisting of Exxon and Chevron, the other belonging to pharmaceutical sector consisting of Pfizer and Merck.

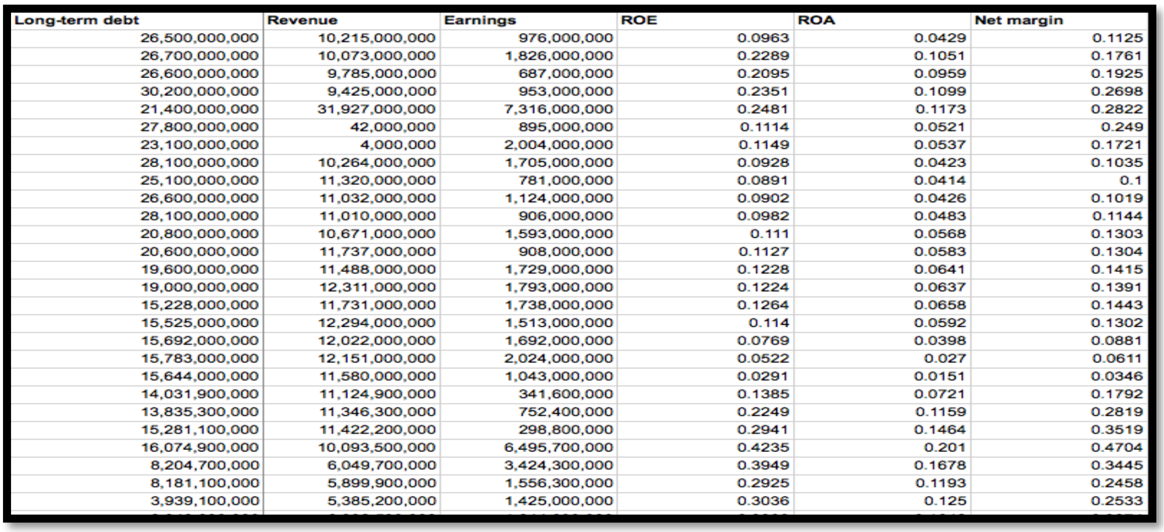
As more than 10 years of data is available for each stock, the minimum threshold for training is sufficiently met so each stock will be individually trained and tested for more accurate analysis.

For each stock, two excel format (.xls) files will be created: inputs file and target file. Input file will contain fundamental variables (stated in section 4.3.3) and target file will contain average price of stock over the chosen time period (2000-2015). Essentially the ANN has to study the hidden patterns in the input file to predict the price in target file.
Data will be randomly segmented according to MATLAB's default settings: 70% for training[24], 15% for validation[25] and 15% for testing[26].

[24] **Training** data is presented to the network during training and the network is adjusted according to its error.

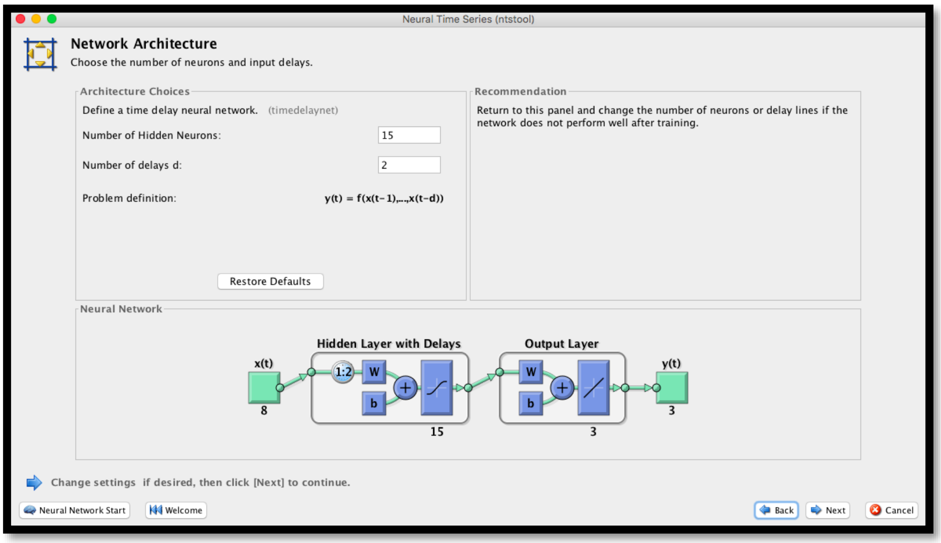
[25] **Validation** data is used to measure network generalisation and halt training when generalisation stops improving.

[26] **Testing** data has no effect on training and so provide an independent measure of network performance during and after training.



Long-term debt	Revenue	Earnings	ROE	ROA	Net margin
26,500,000,000	10,215,000,000	976,000,000	0.0963	0.0429	0.1125
26,700,000,000	10,073,000,000	1,826,000,000	0.2289	0.1051	0.1761
26,600,000,000	9,785,000,000	687,000,000	0.2095	0.0959	0.1925
30,200,000,000	9,425,000,000	953,000,000	0.2351	0.1099	0.2698
21,400,000,000	31,927,000,000	7,316,000,000	0.2481	0.1173	0.2822
27,800,000,000	42,000,000	895,000,000	0.1114	0.0521	0.249
23,100,000,000	4,000,000	2,004,000,000	0.1149	0.0537	0.1721
28,100,000,000	10,264,000,000	1,705,000,000	0.0928	0.0423	0.1035
25,100,000,000	11,320,000,000	781,000,000	0.0891	0.0414	0.1
26,600,000,000	11,032,000,000	1,124,000,000	0.0902	0.0426	0.1019
28,100,000,000	11,010,000,000	906,000,000	0.0982	0.0483	0.1144
20,800,000,000	10,671,000,000	1,593,000,000	0.111	0.0568	0.1303
20,600,000,000	11,737,000,000	908,000,000	0.1127	0.0583	0.1304
19,600,000,000	11,488,000,000	1,729,000,000	0.1228	0.0641	0.1415
19,000,000,000	12,311,000,000	1,793,000,000	0.1224	0.0637	0.1391
15,228,000,000	11,731,000,000	1,738,000,000	0.1264	0.0658	0.1443
15,525,000,000	12,294,000,000	1,513,000,000	0.114	0.0592	0.1302
15,692,000,000	12,022,000,000	1,692,000,000	0.0769	0.0398	0.0881
15,783,000,000	12,151,000,000	2,024,000,000	0.0522	0.027	0.0611
15,644,000,000	11,580,000,000	1,043,000,000	0.0291	0.0151	0.0346
14,031,900,000	11,124,900,000	341,600,000	0.1385	0.0721	0.1792
13,835,300,000	11,346,300,000	752,400,000	0.2249	0.1159	0.2819
15,281,100,000	11,422,200,000	298,800,000	0.2941	0.1464	0.3519
16,074,900,000	10,093,500,000	6,495,700,000	0.4235	0.201	0.4704
8,204,700,000	6,049,700,000	3,424,300,000	0.3949	0.1678	0.3445
8,181,100,000	5,899,900,000	1,556,300,000	0.2925	0.1193	0.2458
3,939,100,000	5,385,200,000	1,425,000,000	0.3036	0.125	0.2533

Figure 17: A snapshot of Merck's input file (MerckInputs.xls)



Neural Time Series (ntstool)

Network Architecture

Choose the number of neurons and input delays.

Architecture Choices

Define a time delay neural network. (timedelaynet)

Number of Hidden Neurons: 15

Number of delays d: 2

Problem definition: y(t) = f(x(t−1),...,x(t−d))

Recommendation

Return to this panel and change the number of neurons or delay lines if the network does not perform well after training.

Neural Network

Change settings if desired, then click [Next] to continue.

Figure 18: Network Architecture Settings (MATHWORKS, n.d.)

Advanced settings page allows user to make further changes such as normalisation, learning rate and learning rate alteration.

After loading both the inputs file and targets file, network parameters such as time delay, number of neurons, normalisation of inputs is specified in order for ANN to initiate training, validation and testing of the given data.

3.5.2 Understanding the Results

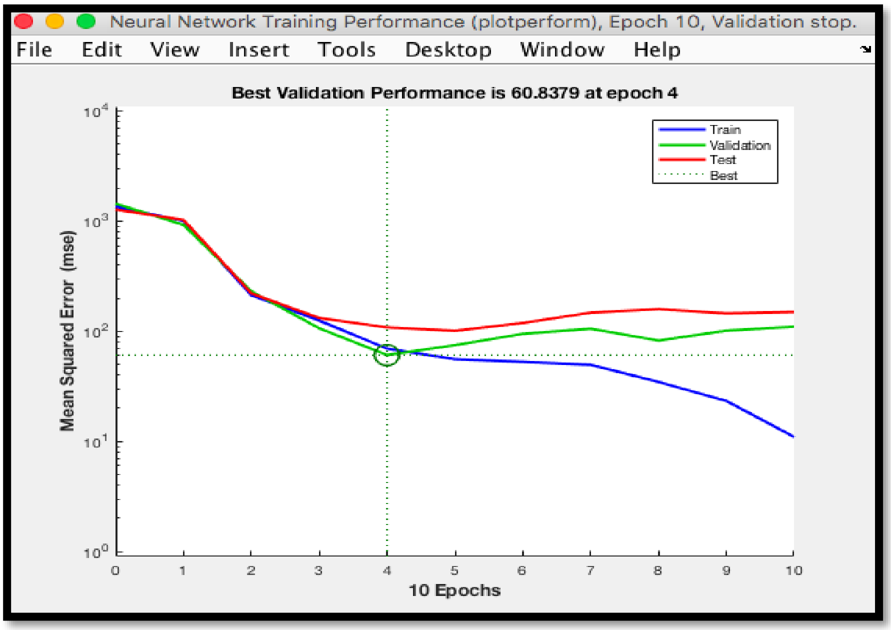


Figure 19: Performance plot of Merck with default settings

Once the network has stopped training, the results are displayed in form of error histograms, performance plots, error cross correlation and regression. For financial domain, performance plots are used to interpret results. Without understanding these plots, results of any financial forecasting project hold no meaning.

A performance plot essentially shows the mean squared error (MSE) of training, validation and testing data; **60.8379** in the graph above is MSE of the data, which is a very high value. The end goal is quite simple: reduce the MSE (calculated error between network's predicted price and actual price of a given stock) to its lowest value possible without compromising the generalisation aspect of the data. For most financial forecasting research papers consulted during literature review, MSE was within the range of 0.05 to 1.2. In the performance plot shown above, the red line represents testing data, blue line represents training data and green line represents validation data. A zero MSE, an ideal situation, indicates that ANN has completely understood the hidden trends in the data and can predict the future stock prices with an accuracy of 100%. This kind of sophistication and accuracy has never been achieved but is the ideal goal of every financial forecaster.

Generalisation, the ability of an ANN to handle unseen data, is a very important aspect of training phase. Although achieving the lowest possible MSE is the end goal but if this is done at the cost of generalisation, results have no significance in real time scenarios. Over-fitting data needs to be avoided at all times; MATLAB was specifically selected for its excellent generalisation features. Training is stopped once the ANN observes that generalisation of data is not improving.

3.5.3 Experimental Results & Adjustments

In this section the pitfalls of experimental results are discussed along with corrections to ensure the final results are as accurate as possible. The performance plots of all 4 stocks can be found in appendix C.

The default settings of MATLAB are certainly not optimised for a financial forecasting problem as can be seen from the performance plot of Merck that gives an extremely high MSE of 60.839.

After tinkering with the settings of the ANN in experimental studies performance plots, the following factors were identified:

- Normalising the data is very essential; initially the data was not normalised so the MSE of most stocks was in excess of 50. As previously discussed in literature review, the activation function lies between -1 to 1 so any value outside of this range is not recognised hence the extremely high MSE.
- One hidden layer is employed for this project but number of neurons in that layer is not rigidly determined; the results of increasing the number of neurons are in line with literature review. Doubling the neurons, if all other factors are kept constant, from 10 to 20 results in a significant decrease in MSE but if the neurons are increased any further (40,60 or even 100), MSE starts increasing and over-fitting occurs. For this particular set of data, optimum performance is achieved if the neurons are only increased in the range of 2 to 18.
- Changing the number of delay from MATLAB's default settings causes the network to give extremely high MSE so this particular setting was set to be constant (2).
- The optimum learning rate lies between 0.14 to 0.23; outside of this range, system is seen to be stuck in local minimum and in rare cases (Pfizer) network paralysis occurs in which case training needs to be reinitiated from the scratch. As such a learning rate of 0.2 was found to be perform significantly well with most stocks (with the exception of Merck).
- A quick comparison between learning algorithms was carried out on Pfizer's data as time constraints didn't allow for a side-by-side analysis with all stocks. Back-propagation was compared with Levenberg-Marquardt algorithm; although Levenberg-Marquardt excelled in memory management and computational time[27], back-propagation was found to be more accurate. Once again, the predictions of excellent relative performance of back-propagation algorithm as stated in literature review were confirmed in this test.
- At default and optimised settings, Chevron's stock was seen to be performing significantly better than others due to its exceptionally low MSE.
- The default segmentation of 70-15-15 between training, validation and testing data respectively was found to perform the best; An 80-10-10

[27] A 2014 MacBook Air with 1.4 GHz Intel Core i5 processor and 4 GB 1600 MHz DDR3 ram was used to carry out the tests.

and 60-20-20 configuration was also tested but yielded relatively inferior results.

- Successively training the data with similar settings was found to decrease the MSE but also increase over-fitting of the data and reduce generalisation. After running the initial settings for 4 times on the Exxon data set, the training MSE can no longer be seen as it merges with validation MSE line (blue line cannot be seen in the diagram below).

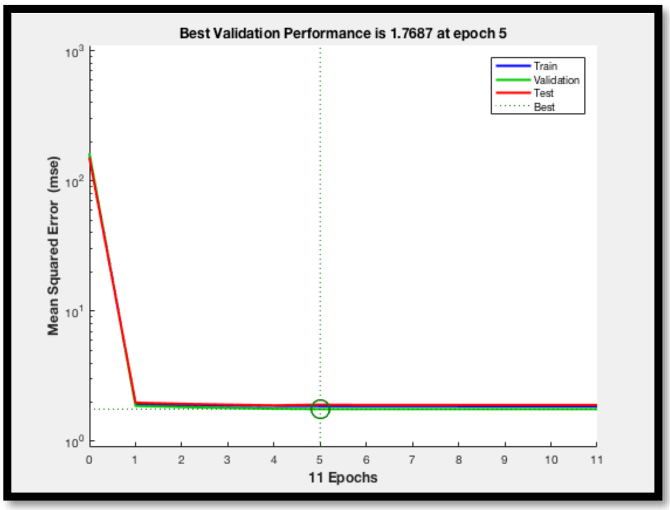


Figure 20: Performance plot of Exxon

Chapter 4: Results

4.1 Performance Plots

The performance plots of Exxon Corporation, Chevron Corporation, Pfizer Inc. and Merck & Co. Inc. are given below with the final optimum settings.

Exxon

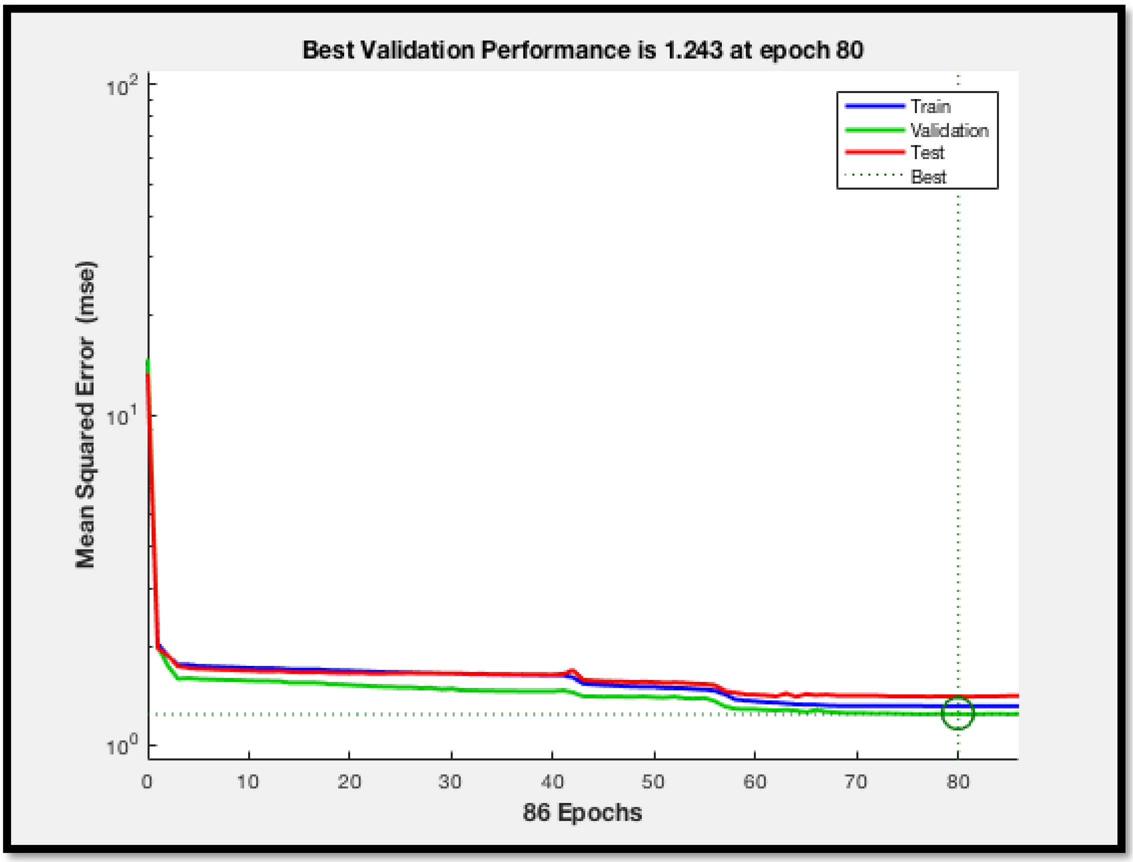


Figure 21: Final Performance plot of Exxon

Learning algorithm: **Backpropagation**
Activation function: **Sigmoid**
Number of delays: **2**
Number of hidden layers: **1**
Number of hidden neurons: **8**
Number of total epochs: **1000**
MSE (Mean Square Error): **1.2423**
Learning rate: **0.20**

Chevron

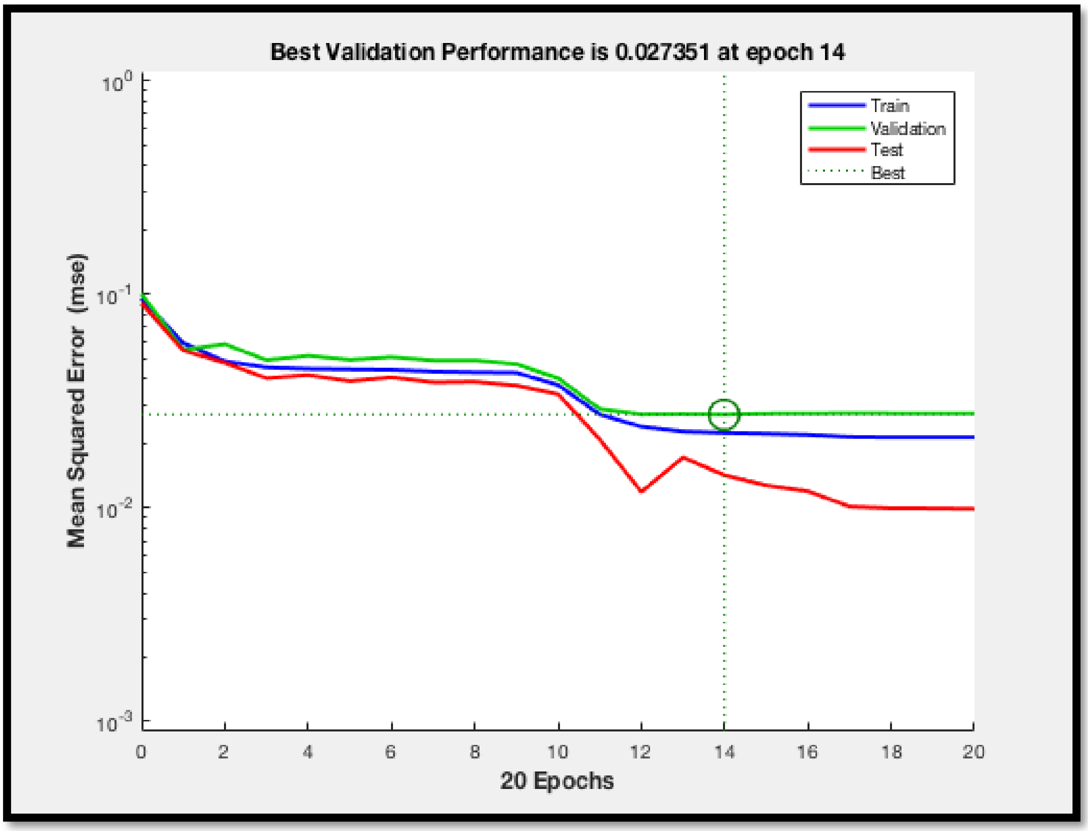


Figure 22: Final Performance plot of Chevron

Learning Algorithm: **Back-propagation**
Activation function: **Sigmoid**
Number of delays: **2**
Number of hidden layers: **1**
Number of hidden neurons: **12**
Number of total epochs: **1000**
MSE (Mean Square Error): **0.027351**
Learning rate: **0.20**

Pfizer

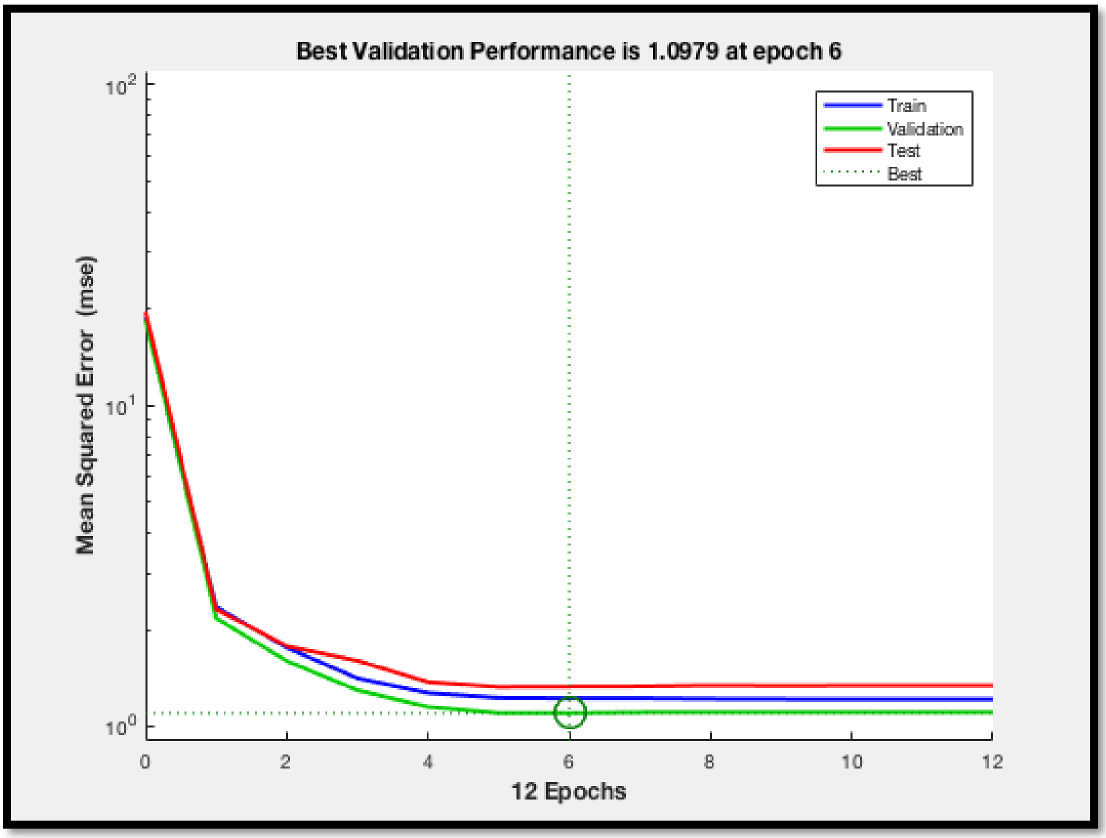


Figure 23: Final Performance plot of Pfizer

Learning Algorithm: **Back-propagation**
Activation function: **Sigmoid**
Number of delays: **2**
Number of hidden layers: **1**
Number of hidden neurons: **4**
Number of total epochs: **1000**
MSE (Mean Square Error): **1.0979**
Learning rate: **0.20**

Merck

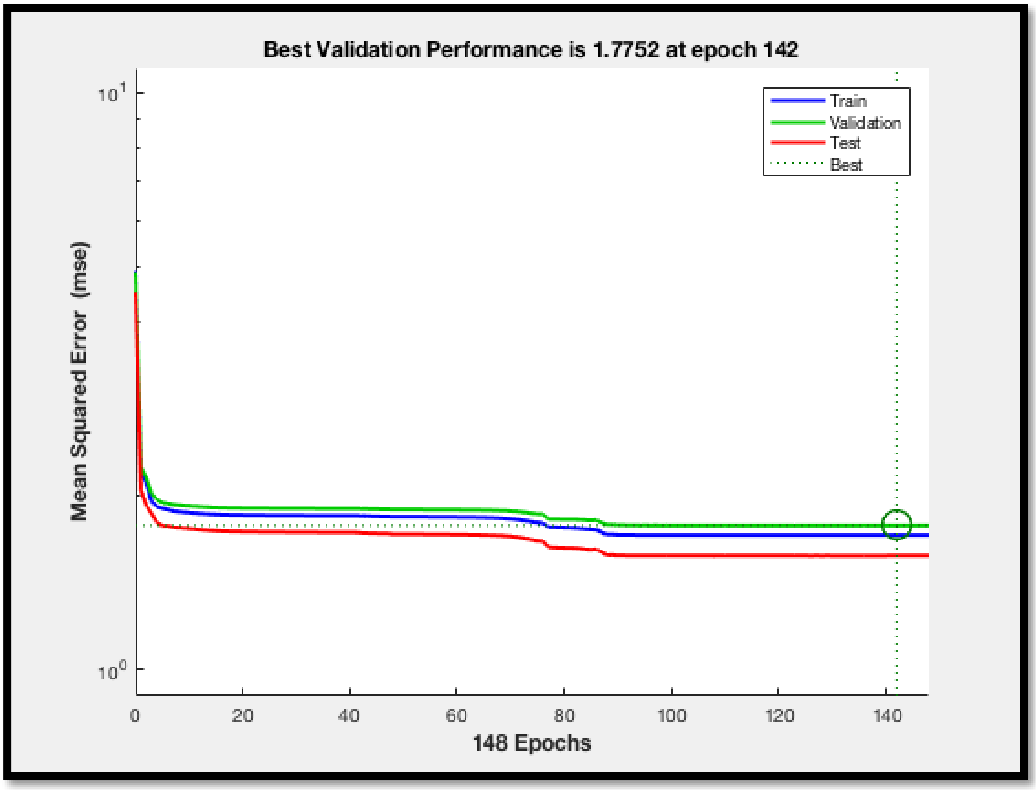


Figure 24: Final Performance plot of Merck

Learning Algorithm: **Back-propagation**
Activation function: **Sigmoid**
Number of delays: **2**
Number of hidden layers: **1**
Number of hidden neurons: **14**
Number of total epochs: **1000**
MSE (Mean Square Error): **1.7752**
Learning rate: **0.20**

Chapter 5: Conclusion

The final results of this project suggest that there is immense potential in utilising Artificial Neural Networks to predict stock prices. However, a number of pitfalls and limitations were observed. The conclusion will essentially be divided in two parts: first part will discuss results in detail and their significance, second will shed light on pitfalls, limitations and suggested improvements to come up with a better and robust model in future.

Although all the MSEs (Exxon: 1.242, Chevron: 0.0274, Pfizer: 1.0979, Merck: 1.7752) obtained in the final results lie in the acceptable range but in real time, multi-dynamic stock market environment, only Chevron's performance holds true significance and meaning in terms of long term strategic trading. On a relative scale, all three stocks (Exxon, Pfizer and Merck) performed in a close range (1.0 to 1.8) but the extremely accurate environment of stock market will not accept this range as feasible; an MSE of 1.242 is the squared difference of ANN's predicted stock price and the real stock price. Over the course of a trading day, where orders placed range from 200 to over a 1000, an MSE of even 1.242 will add up and result in overall loss to the firm. If penny stocks are concerned, this implementation can be partially successful but as an industry wide solution, these results are not sufficiently significant.

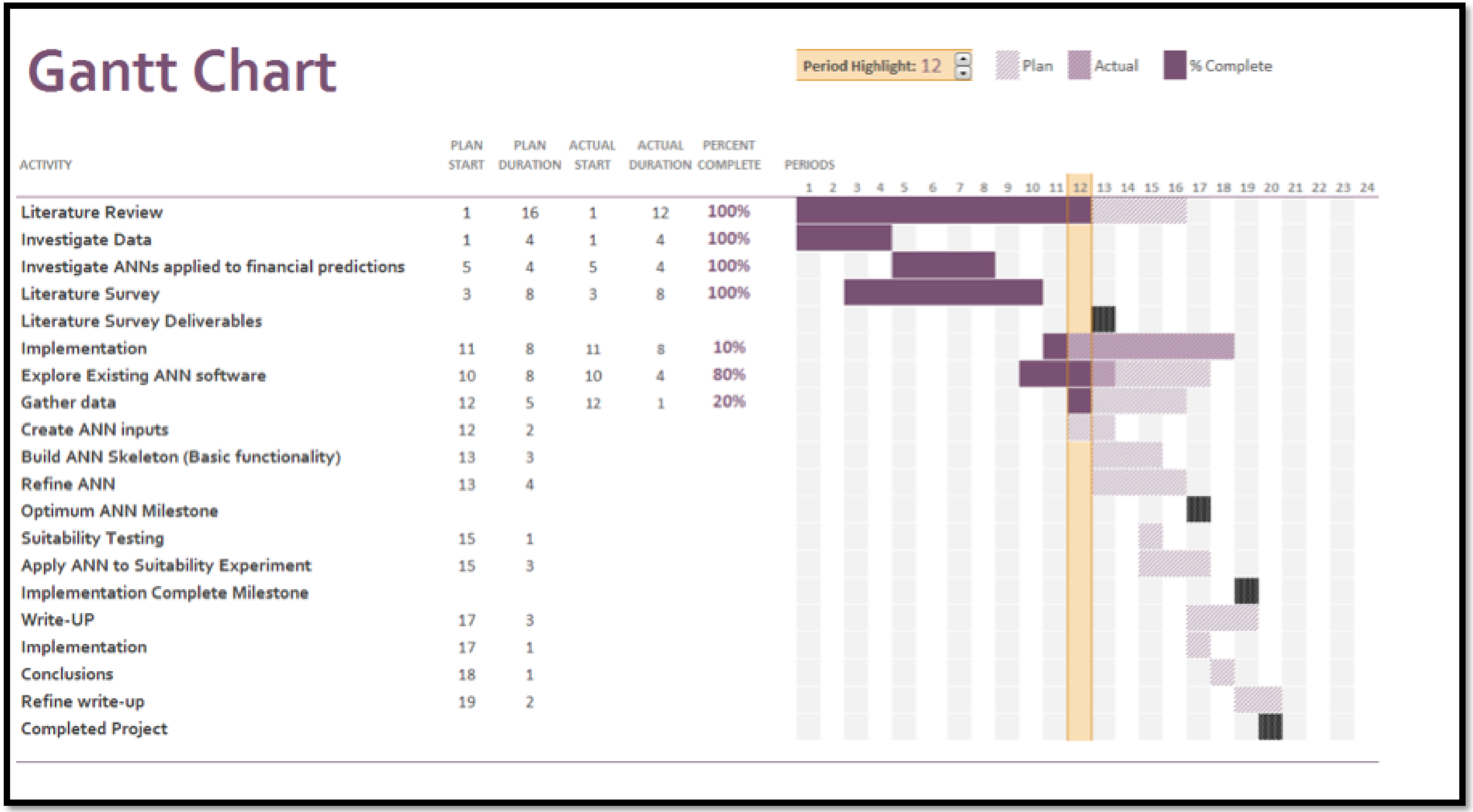
In terms of numbers, Chevron's MSE is 45 times smaller than the three other stocks. At 0.0274, it has the potential to be further improved and integrated with a buy and hold strategy of trading due to its high accuracy. So why did Chevron's stock excelled whilst others lagged? The very answer to this question highlights the complexity and limitations of using ANNs for stock price forecasting. For an asset intensive company, investors are primarily concerned with Assets, Liabilities and Revenue. This is completely logical as in worst-case scenario where a company defaults on its debts, authorities seize its assets and the liabilities are addressed. Before the investors of this misfortunate company are paid, all the liabilities are paid off. No one wants to invest in a company whose liabilities are high relative to its revenue. Chevron's data was seen to show the strongest correlation to all these three factors. The second and most difficult factor is time delay: the time it takes for the fundamentals of a company to reflect or make changes to its stock price. This is one of the factors that need extensive trial and error experiments as determining when the fundamentals of a company will make changes to its price is a paradigm with multiple facets; sentiment analysis will particularly be useful in order to determine this factor. Using cloud computing to gather public sentiments from online news sources (Bloomberg and Yahoo for example) along with social media platforms such as Twitter can help ANN gauge the receptiveness of public to particular changes and with enough training, predict price movements. Essentially it becomes a big data solution where all the relevant channels are part of the input so ANN can understand the hidden patterns with much higher accuracy. The third factor involves the inner workings of an ANN; the analogy of brain is given because no one knows what really happens in the hidden layers. In case of Chevron, the ANN was able to understand and identify the hidden patterns in fundamental variables that enabled it to predict prices with a very low MSE. The last factor lies at the very heart of any forecasting problem: availability of accurate data. One of the hardest parts of literature review was to find accurate and relevant data. Data should never be a limiting factor when devising an experiment; dealing with "available data" instead of "required data" creates a lot of problems and results in an ANN with relatively poor prediction ability.

The findings of this project were both quite revealing and informative. The literature review, data selection, pre-processing and designing the final experiment gave me an invaluable insight of financial forecasting and modelling. If I would have continued with just asset intensive and non-intensive distinction, my final results would have been lacklustre with not even a single stock showing promise of real implementation; the domain of financial prediction requires extreme customisation and attention to detail. No two stocks are similar and hence require tailored settings in order for ANN to yield useful results. Every stock needs to be independently studied and thoroughly investigated; an extremely accurate ANN will even take in to account the type of management that runs the company. An ANN created from the scratch to target a specific industry rather than a wide range of stocks has much better chances of real time application as seen from the results of this paper.

In terms of initial hypothesis of this paper, "understanding and investigating the relationship between fundamental data and price movements", final results of all four stocks indicated a strong correlation; fundamental factors are certainly responsible for price movements and become more prominent in long term.

Moreover, on the technical end of the spectrum and future experiments, an ANN created from the scratch, preferably in C++ because of its responsiveness, employing genetic algorithms to select inputs of a carefully selected industry (Pharmaceuticals, IT or energy) has immense potential. Essentially genetic algorithm or GA is a method for solving optimisation problems based on natural selection problems process; the biggest advantage of GA over Backpropagation algorithm is its effectiveness in avoiding local minima during training.

Appendix A: Gantt chart

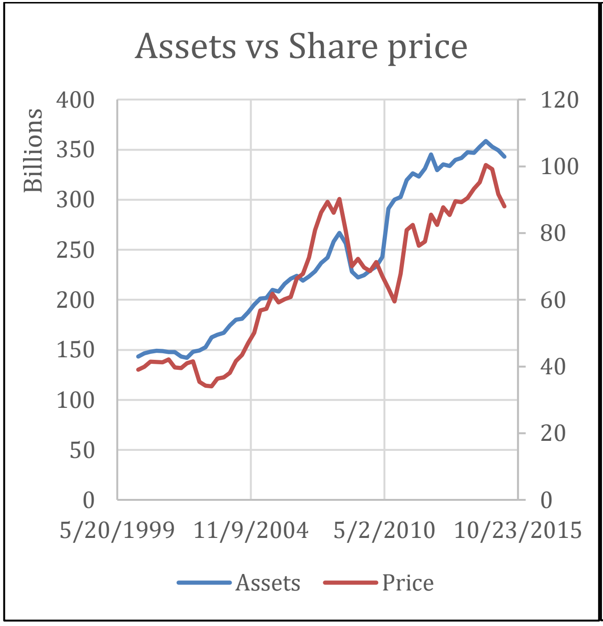


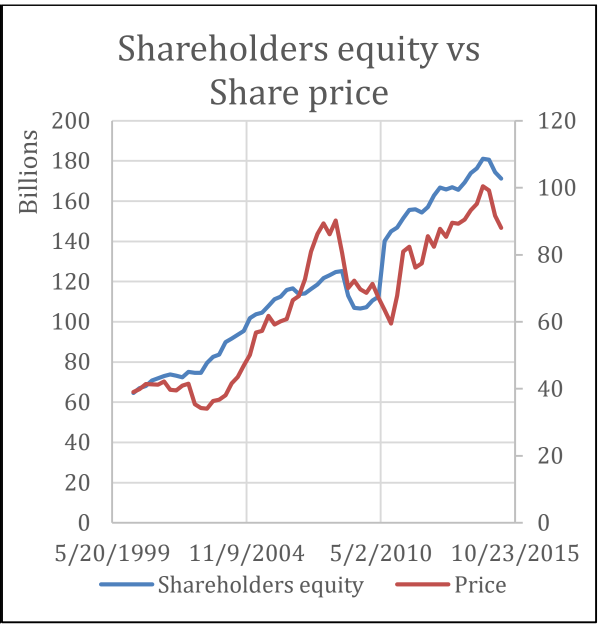
Gantt Chart

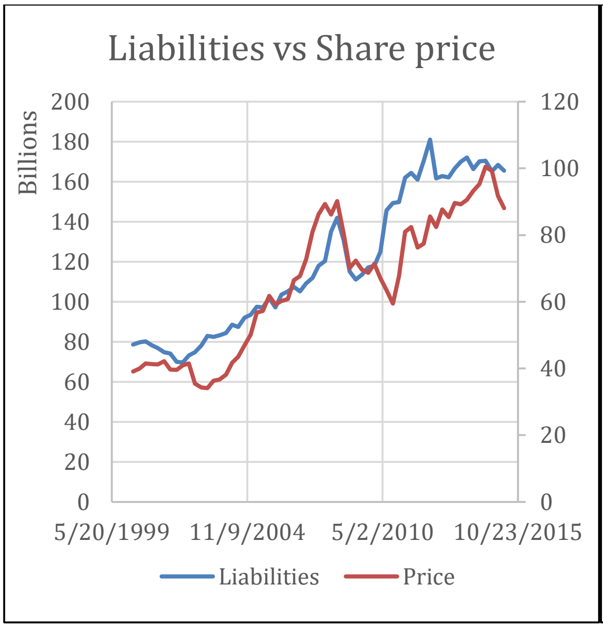
ACTIVITY	PLAN START	PLAN DURATION	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
Literature Review	1	16	1	12	100%
Investigate Data	1	4	1	4	100%
Investigate ANNs applied to financial predictions	5	4	5	4	100%
Literature Survey	3	8	3	8	100%
Literature Survey Deliverables					
Implementation	11	8	11	8	10%
Explore Existing ANN software	10	8	10	4	80%
Gather data	12	5	12	1	20%
Create ANN inputs	12	2			
Build ANN Skeleton (Basic functionality)	13	3			
Refine ANN	13	4			
Optimum ANN Milestone					
Suitability Testing	15	1			
Apply ANN to Suitability Experiment	15	3			
Implementation Complete Milestone					
Write-UP	17	3			
Implementation	17	1			
Conclusions	18	1			
Refine write-up	19	2			
Completed Project					

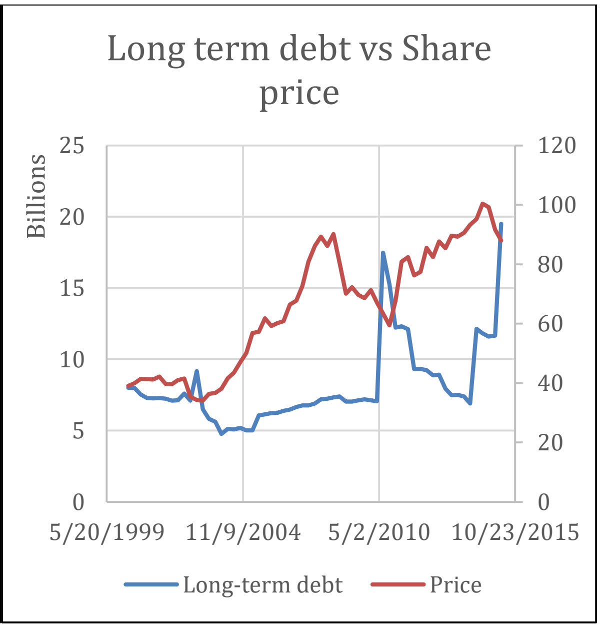
Appendix B: Input Selection Graphs

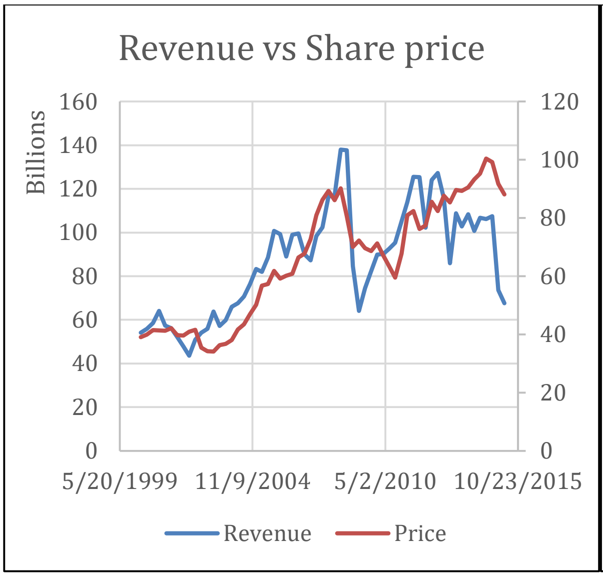
EXXON

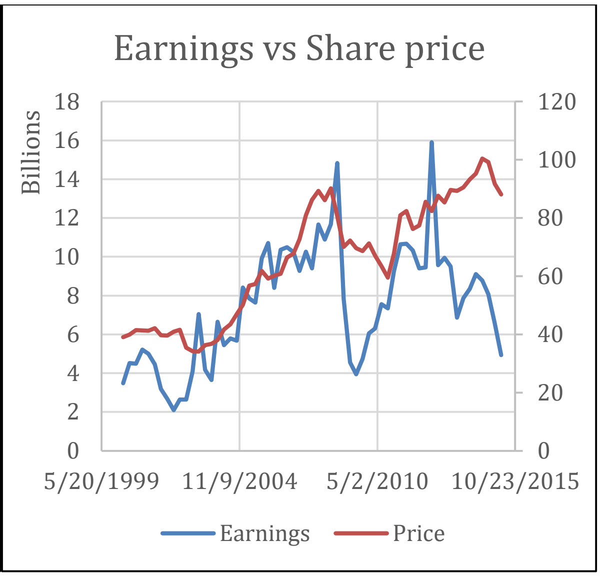

Assets vs Share price

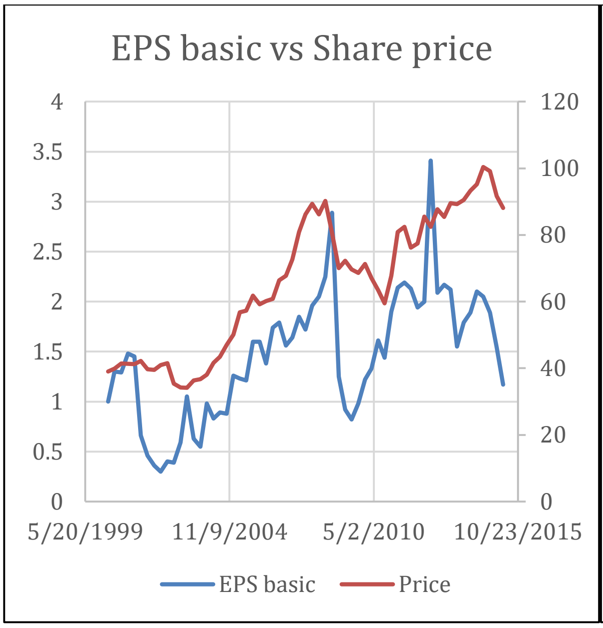

Shareholders equity vs Share price

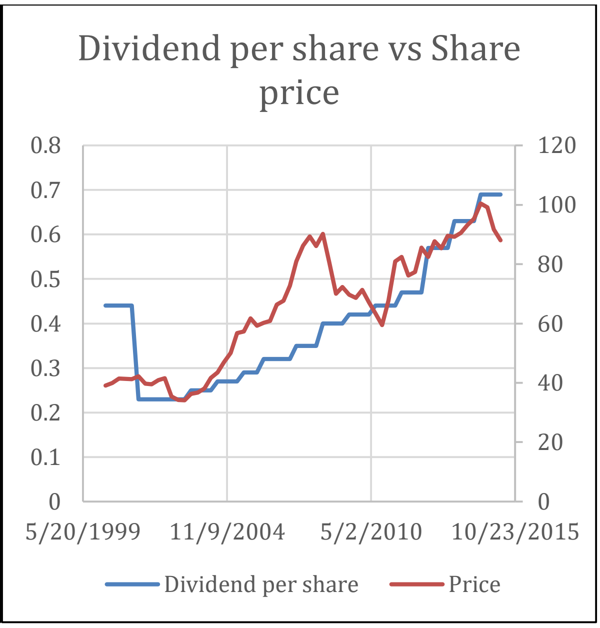

Liabilities vs Share price

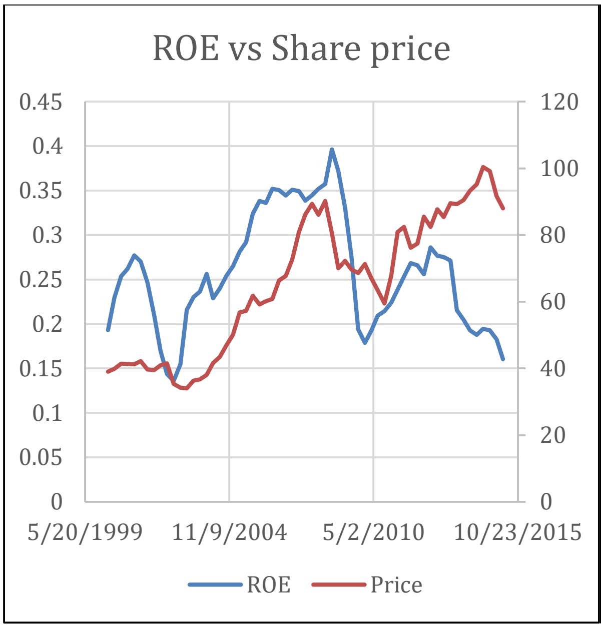

Long term debt vs Share price


Revenue vs Share price
Billions
160
140
120
100
80
60
40
20
0
120
100
80
60
40
20
0
5/20/1999
11/9/2004
5/2/2010
10/23/2015
Revenue
Price

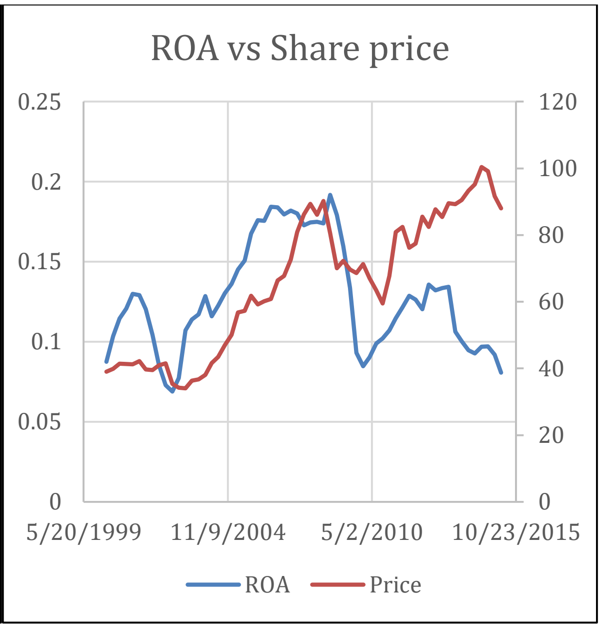

Earnings vs Share price
Billions
18
16
14
12
10
8
6
4
2
0
120
100
80
60
40
20
0
5/20/1999
11/9/2004
5/2/2010
10/23/2015
Earnings
Price


EPS basic vs Share price
4
3.5
3
2.5
2
1.5
1
0.5
0
120
100
80
60
40
20
0
5/20/1999
11/9/2004
5/2/2010
10/23/2015
EPS basic
Price

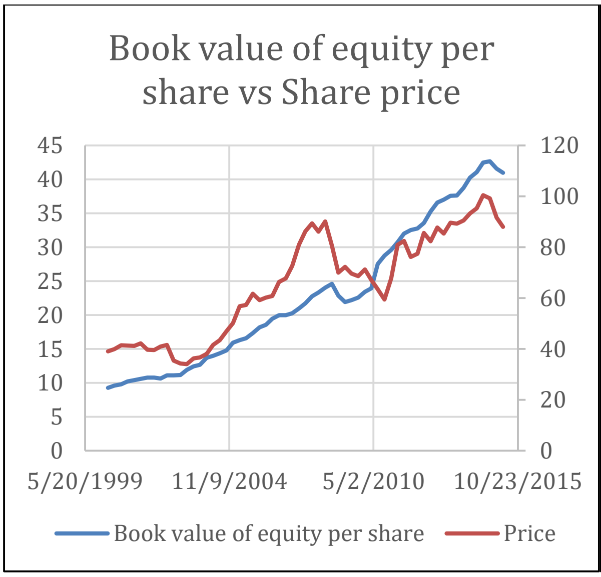

Dividend per share vs Share price
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
120
100
80
60
40
20
0
5/20/1999
11/9/2004
5/2/2010
10/23/2015
Dividend per share
Price



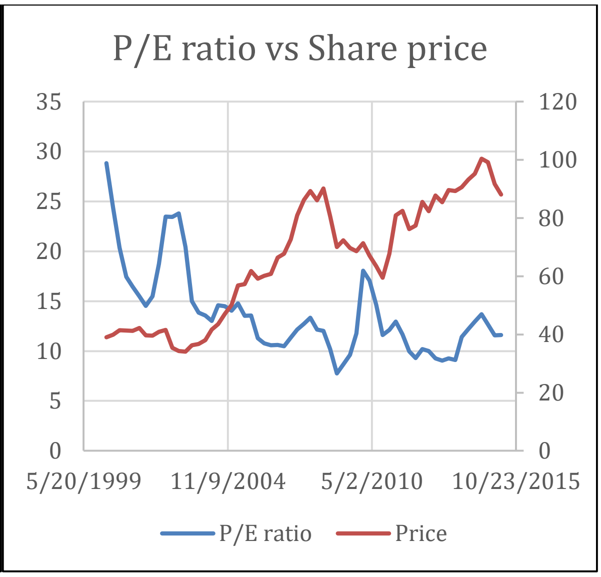
ROE vs Share price



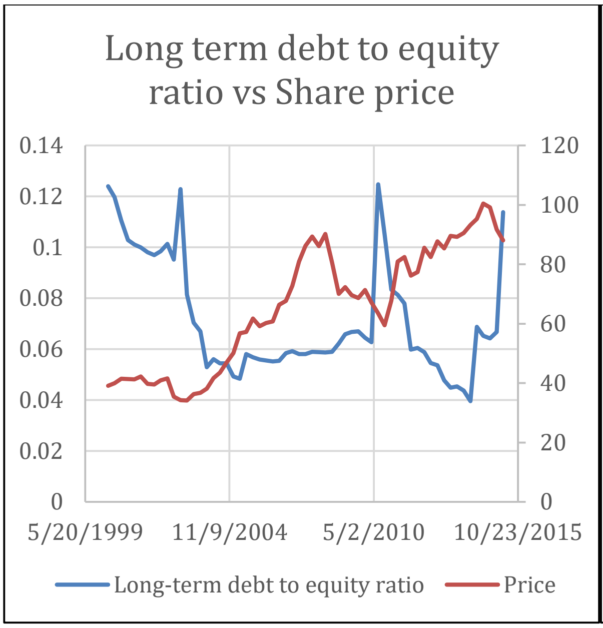
ROA vs Share price

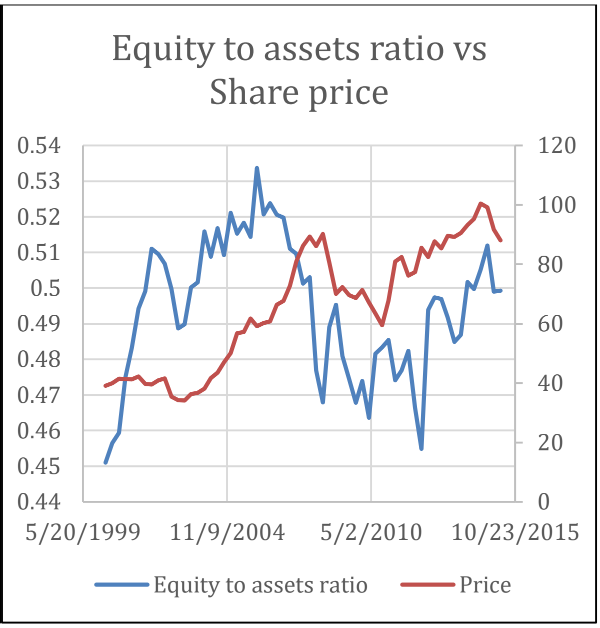


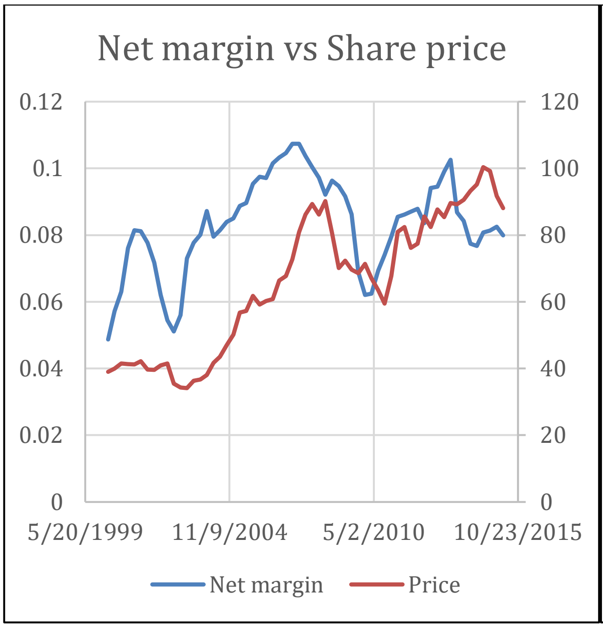
Book value of equity per share vs Share price

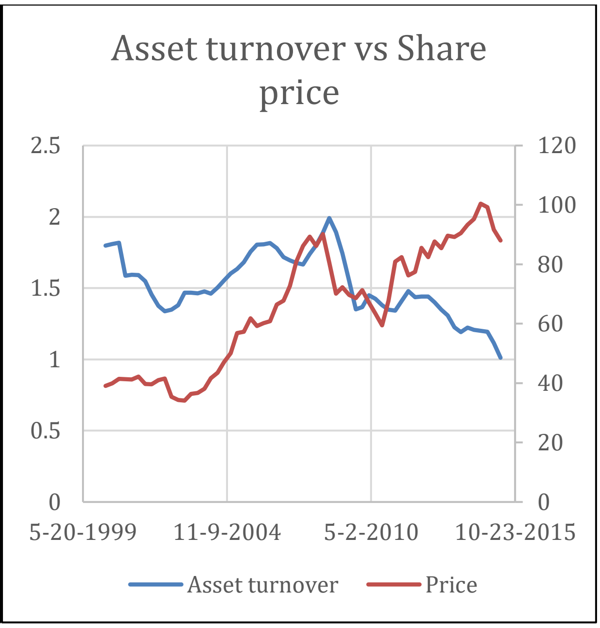


P/E ratio vs Share price

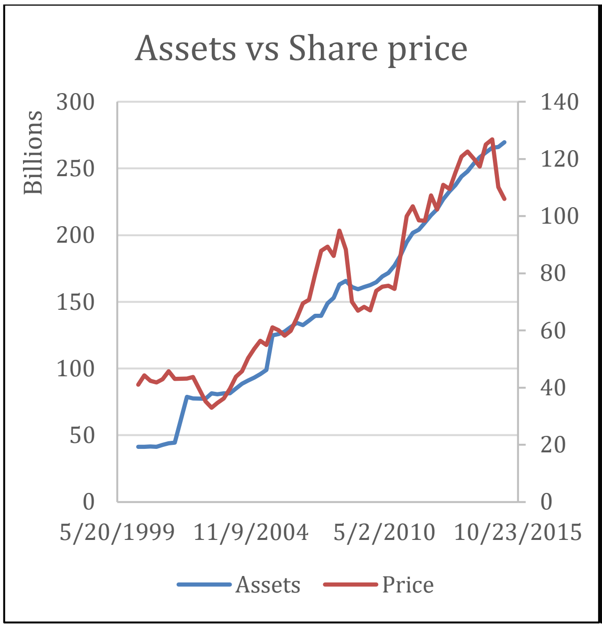

Long term debt to equity ratio vs Share price


Equity to assets ratio vs Share price

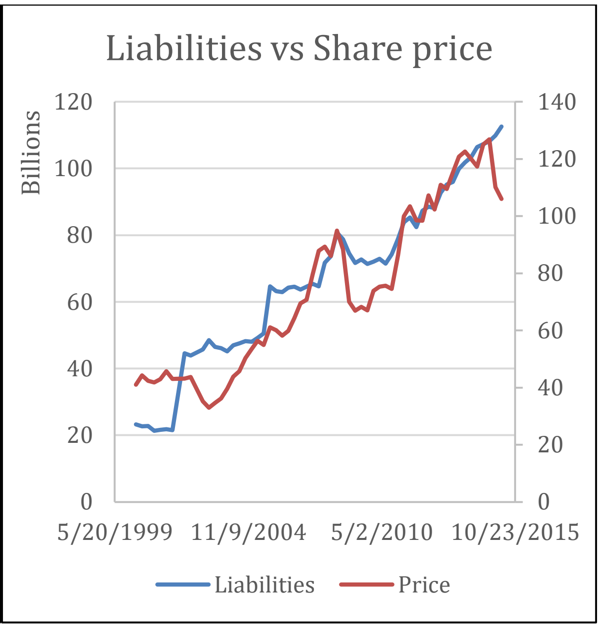

Net margin vs Share price


Asset turnover vs Share price

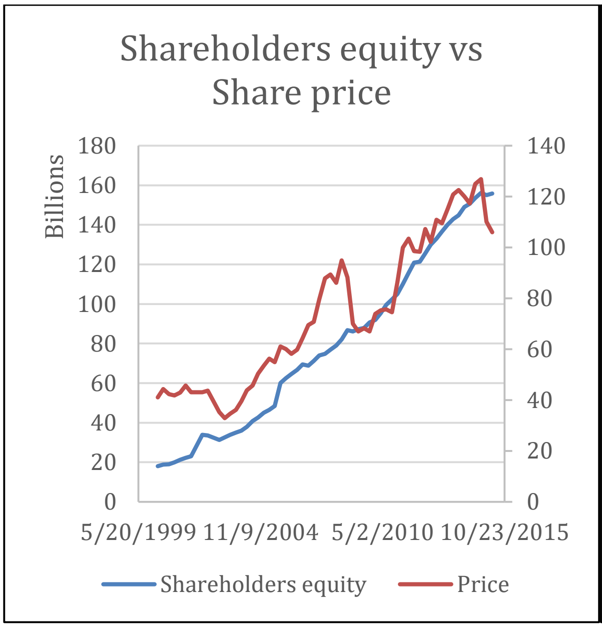
CHEVRON



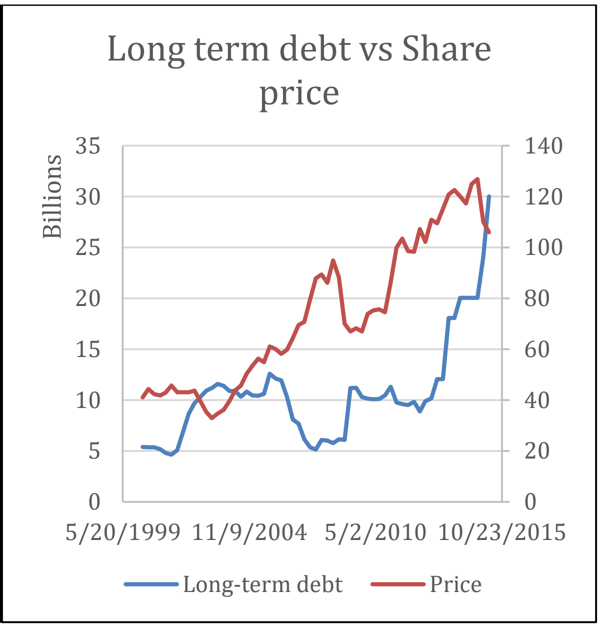
Assets vs Share price
Billions
300
250
200
150
100
50
0
140
120
100
80
60
40
20
0
5/20/1999
11/9/2004
5/2/2010
10/23/2015
Assets
Price



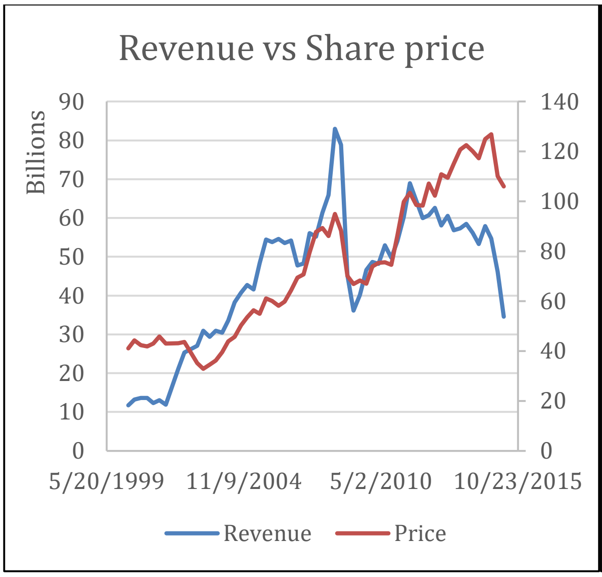
Liabilities vs Share price
Billions
120
100
80
60
40
20
0
140
120
100
80
60
40
20
0
5/20/1999
11/9/2004
5/2/2010
10/23/2015
Liabilities
Price

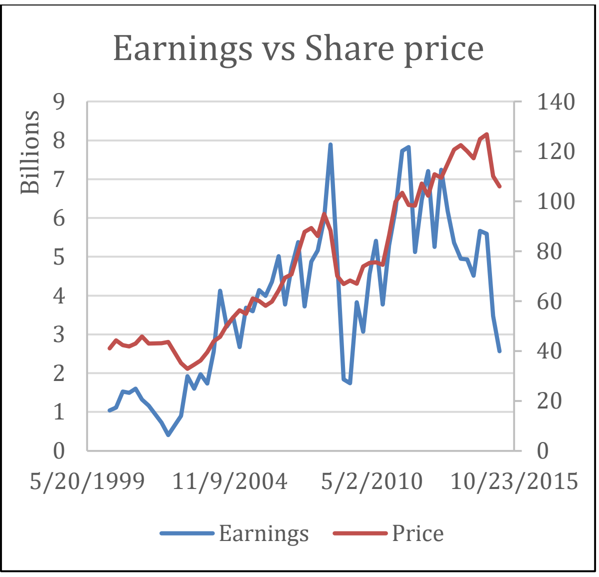


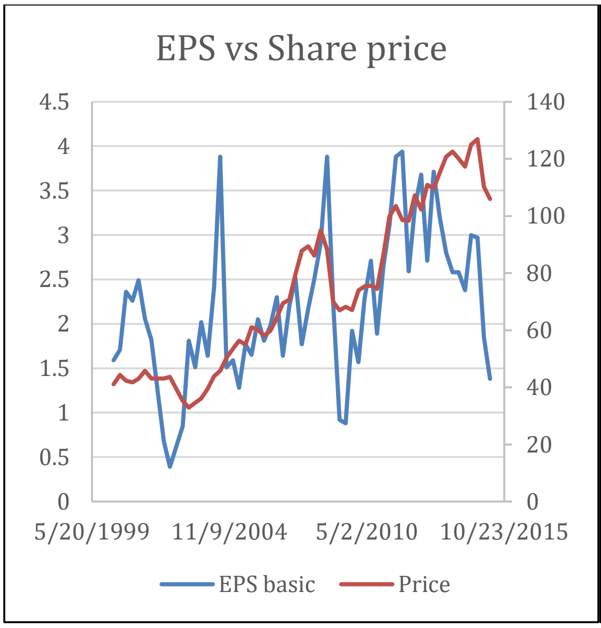
Shareholders equity vs Share price
Billions
180
160
140
120
100
80
60
40
20
0
140
120
100
80
60
40
20
0
5/20/1999
11/9/2004
5/2/2010
10/23/2015
Shareholders equity
Price

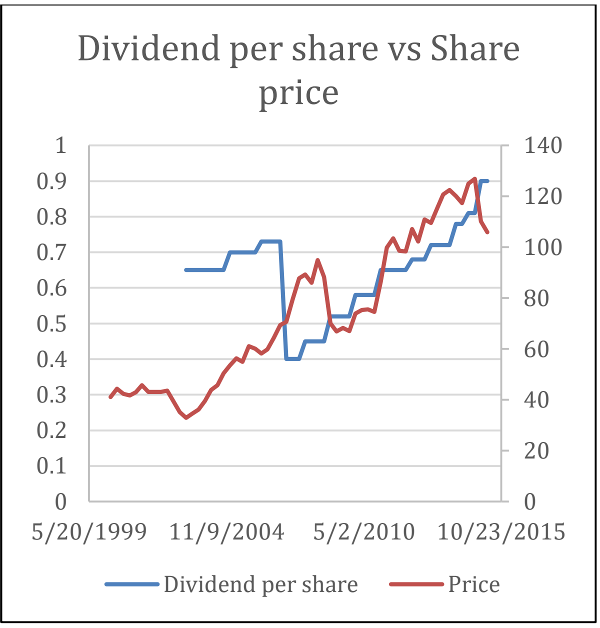


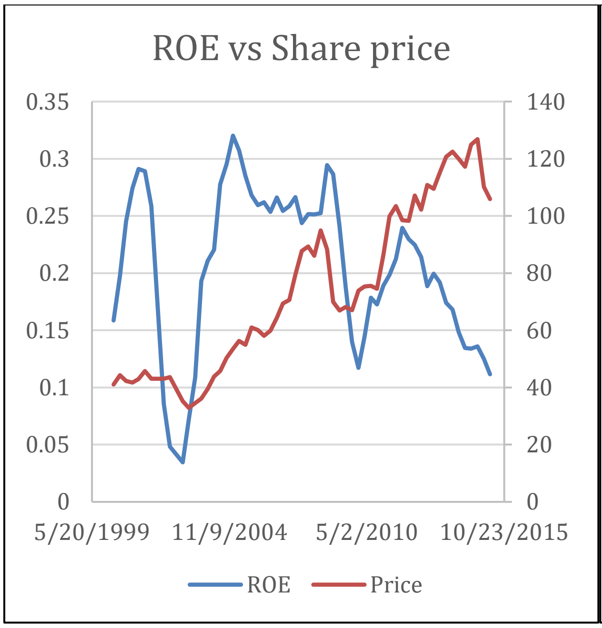
Long term debt vs Share price
Billions
35
30
25
20
15
10
5
0
140
120
100
80
60
40
20
0
5/20/1999
11/9/2004
5/2/2010
10/23/2015
Long-term debt
Price

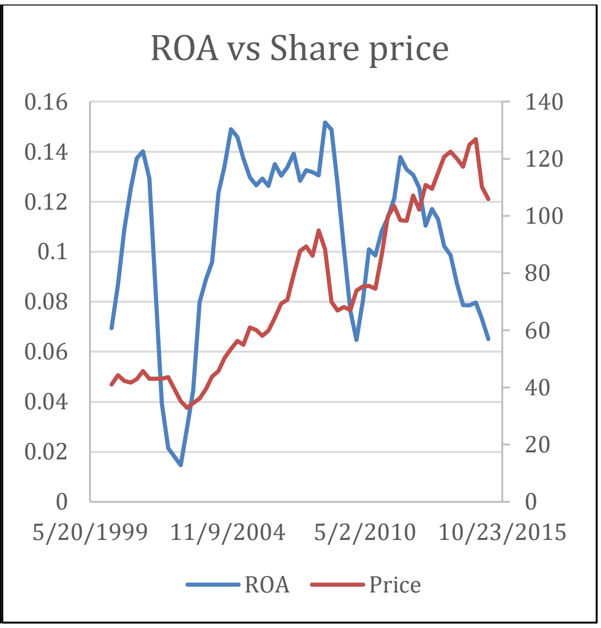

Revenue vs Share price

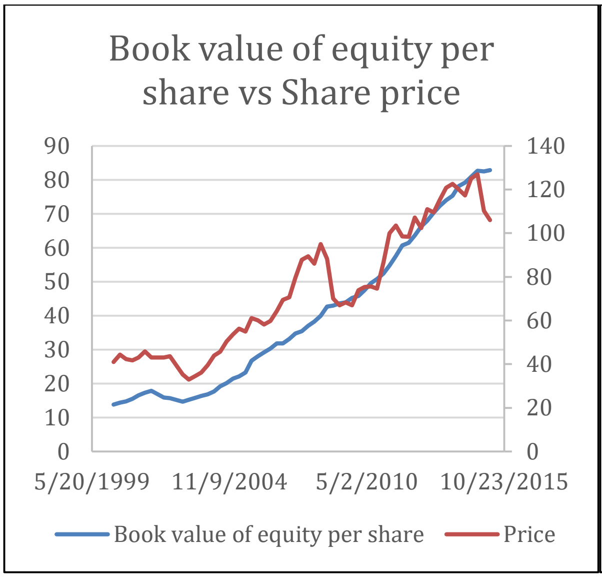

Earnings vs Share price

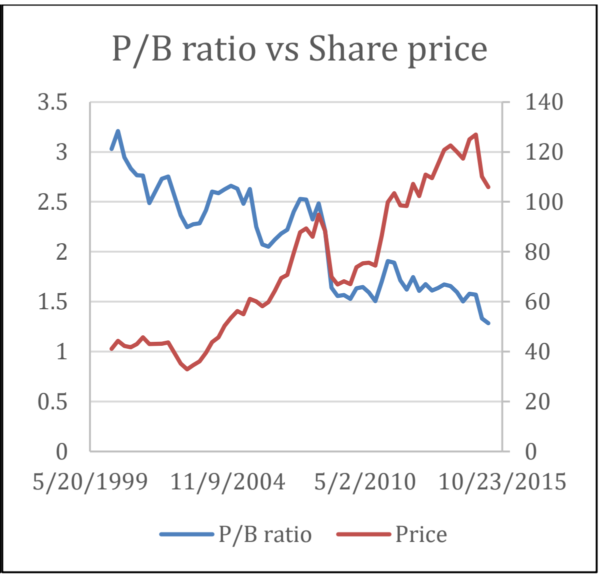

EPS vs Share price

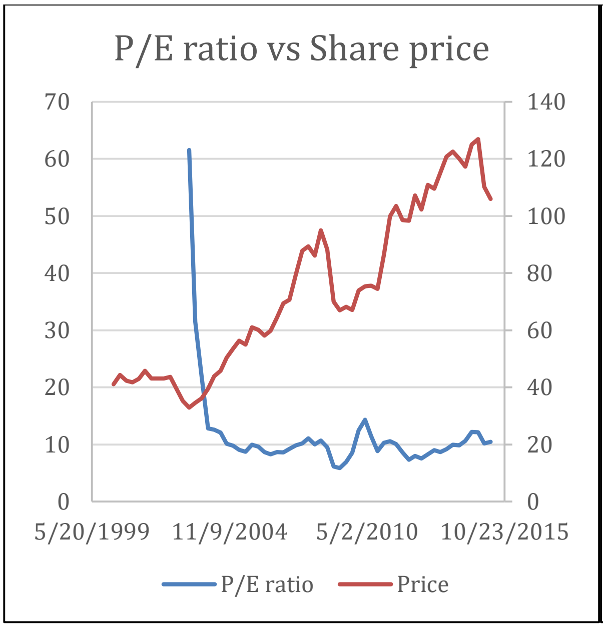

Dividend per share vs Share price

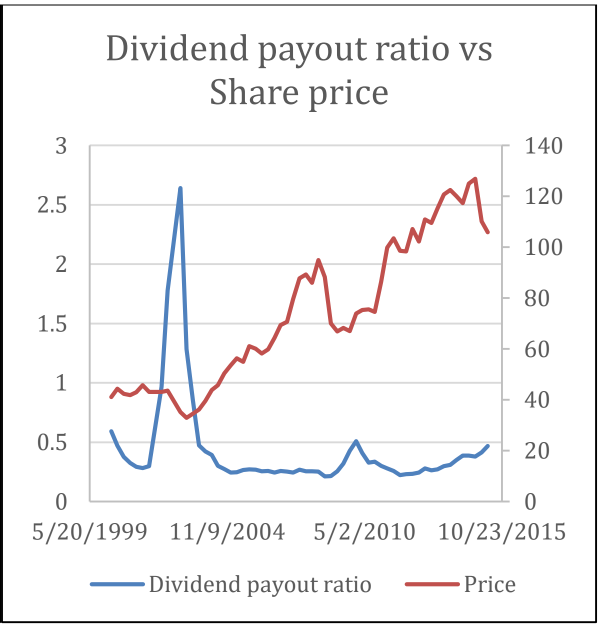

ROE vs Share price

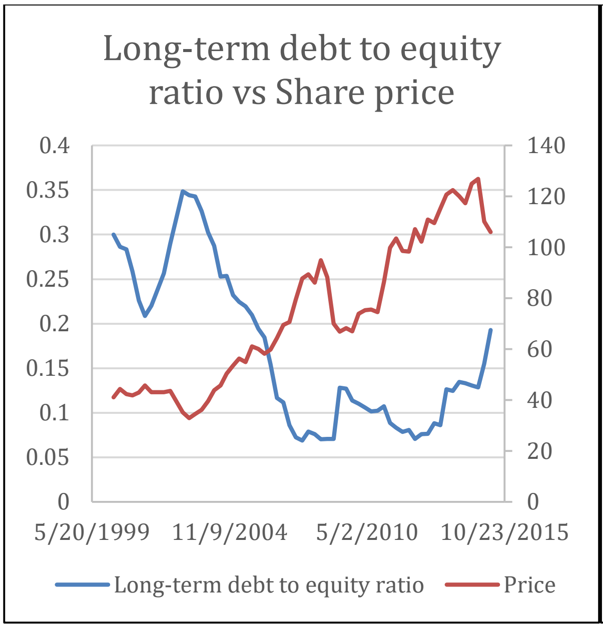

ROA vs Share price


Book value of equity per share vs Share price

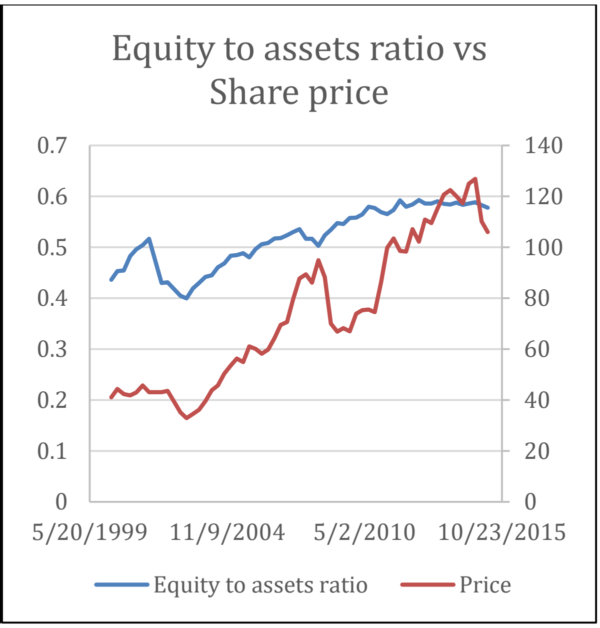

P/B ratio vs Share price


P/E ratio vs Share price

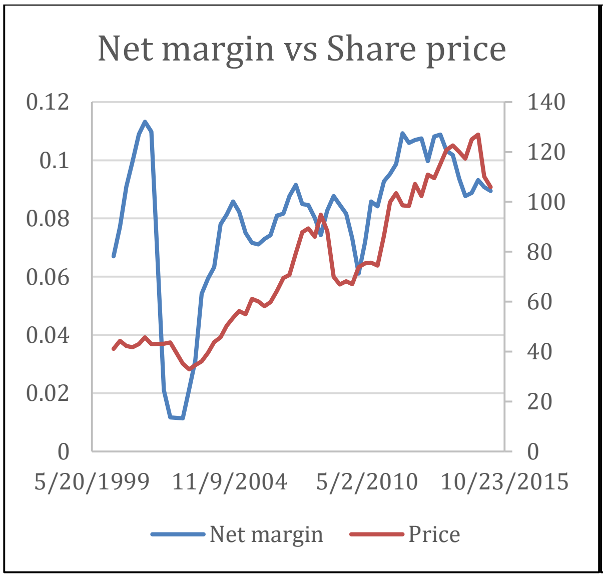

Dividend payout ratio vs Share price



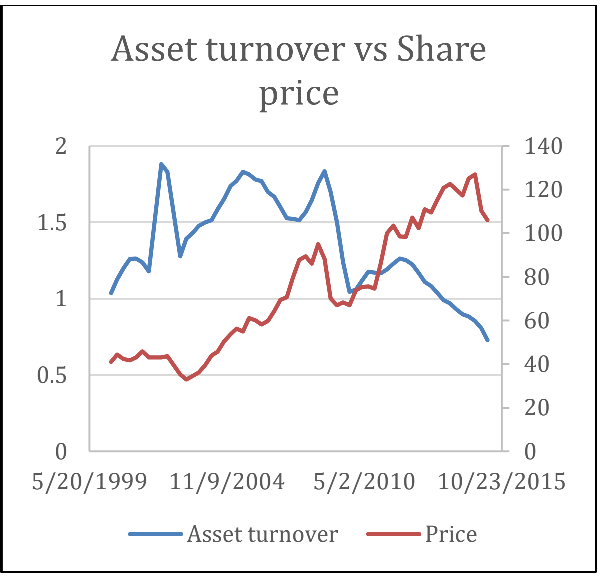
Long-term debt to equity ratio vs Share price



Equity to assets ratio vs Share price

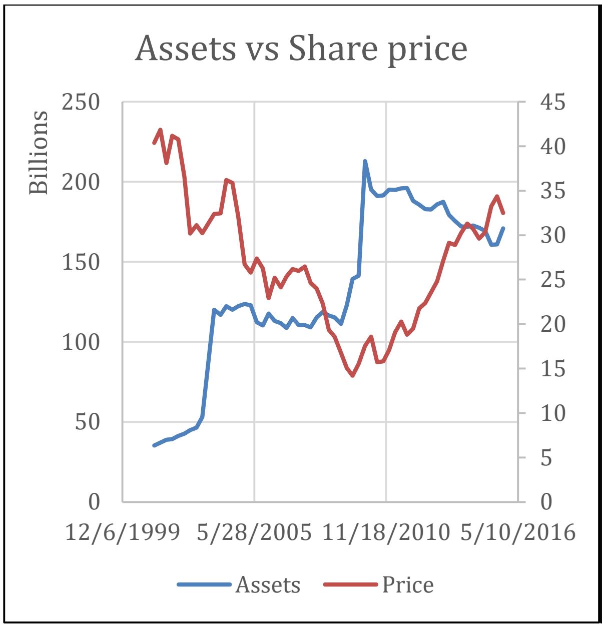


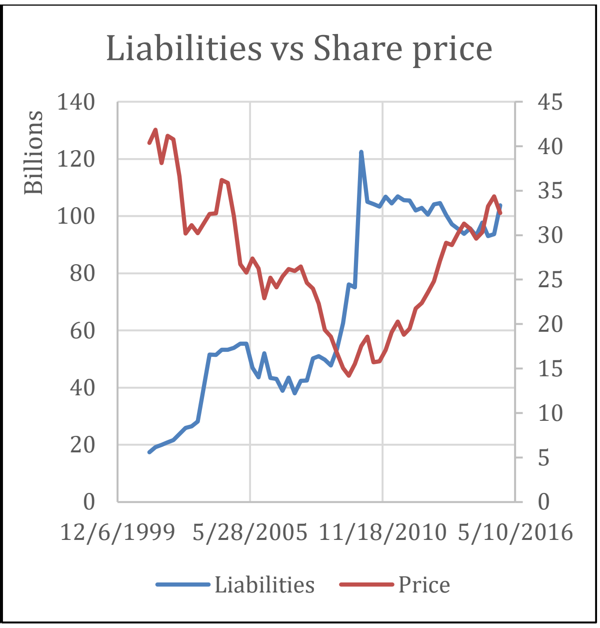
Net margin vs Share price

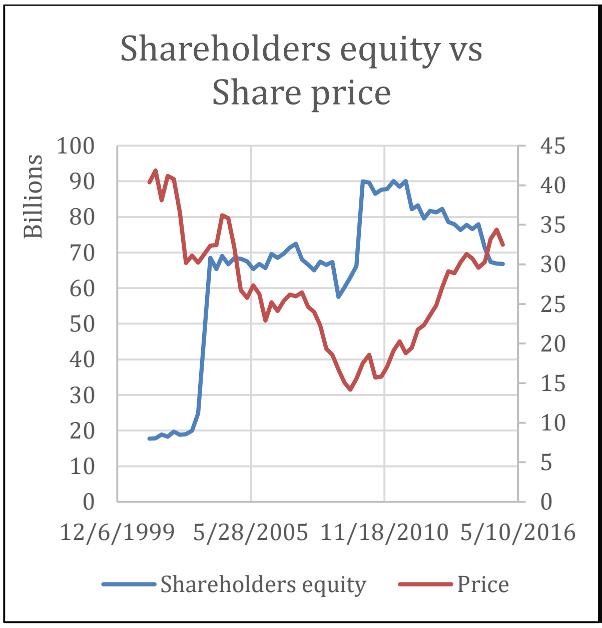


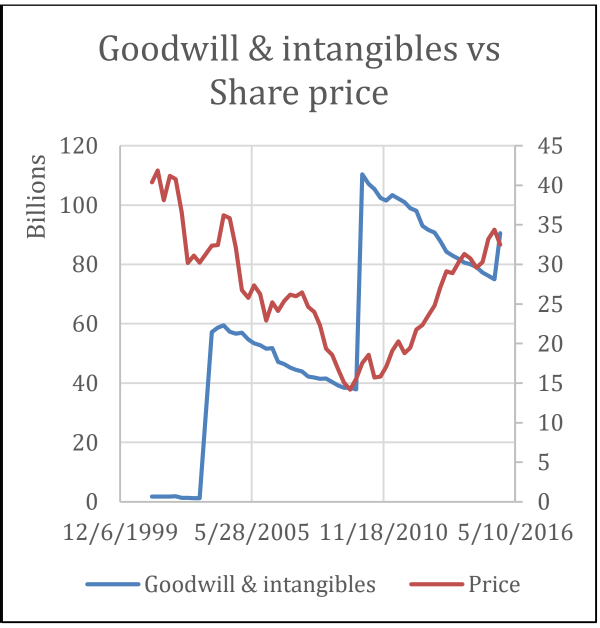
Asset turnover vs Share price

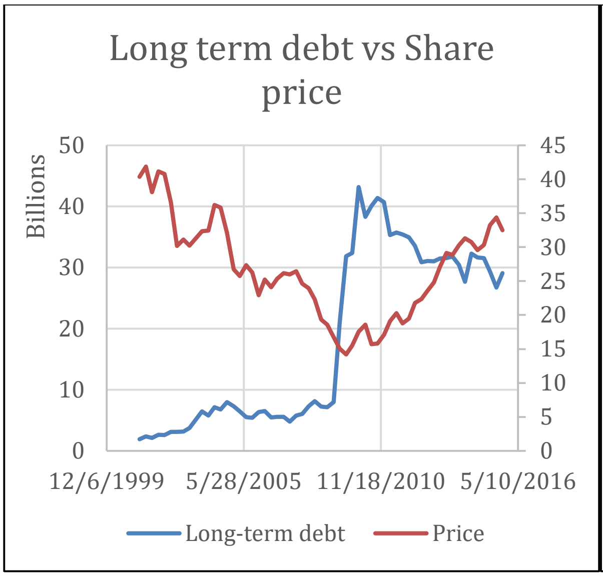
Pfizer Inc.

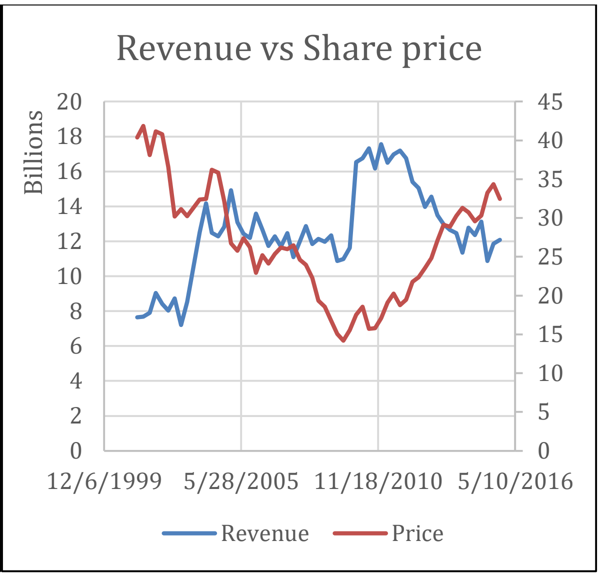

Assets vs Share price
Billions
250
200
150
100
50
0
45
40
35
30
25
20
15
10
5
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
Assets
Price

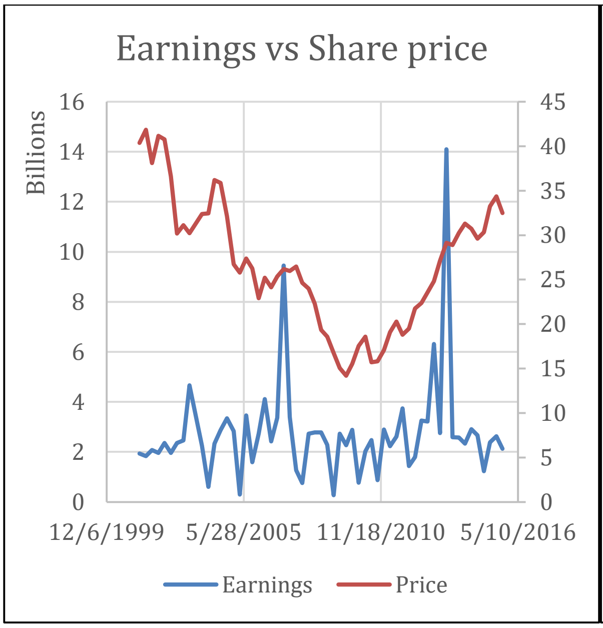

Liabilities vs Share price
Billions
140
120
100
80
60
40
20
0
45
40
35
30
25
20
15
10
5
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
Liabilities
Price

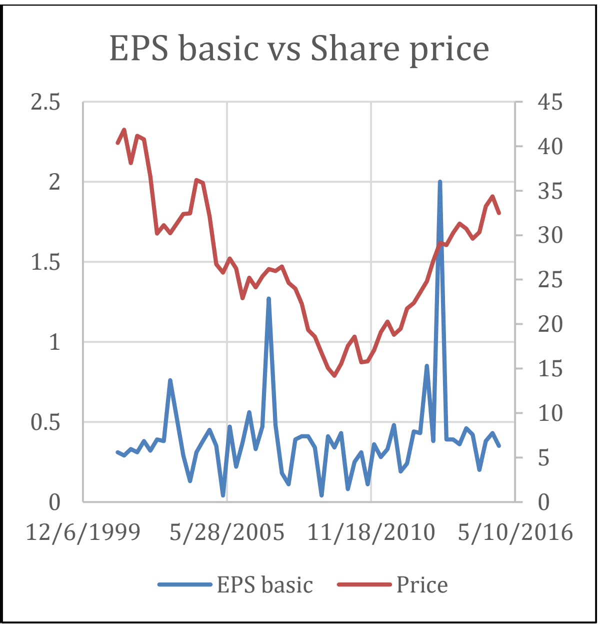

Shareholders equity vs Share price
Billions
100
90
80
70
60
50
40
30
20
10
0
45
40
35
30
25
20
15
10
5
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
Shareholders equity
Price

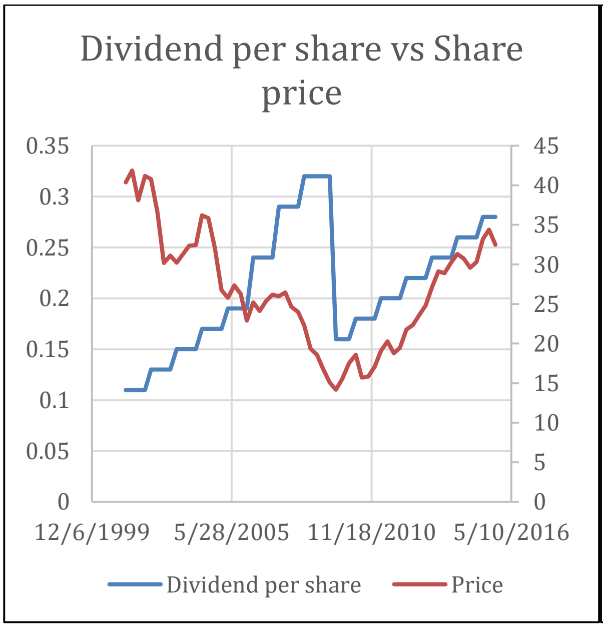

Goodwill & intangibles vs Share price
Billions
120
100
80
60
40
20
0
45
40
35
30
25
20
15
10
5
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
Goodwill & intangibles
Price

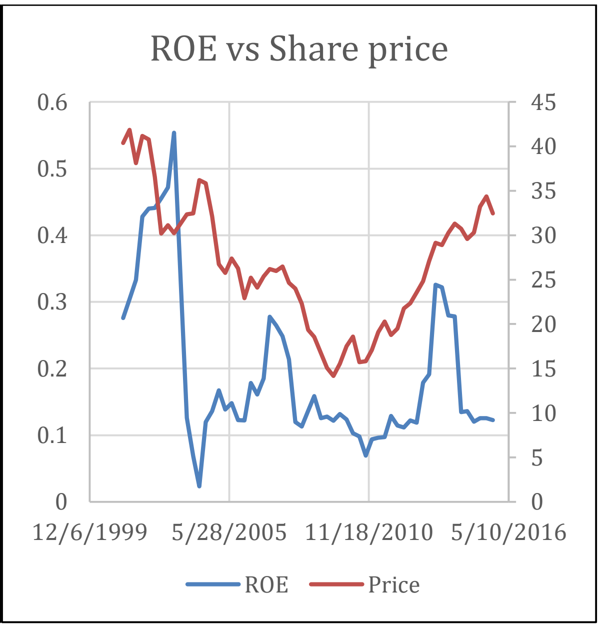

Long term debt vs Share price
Billions
50
40
30
20
10
0
45
40
35
30
25
20
15
10
5
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
Long-term debt
Price

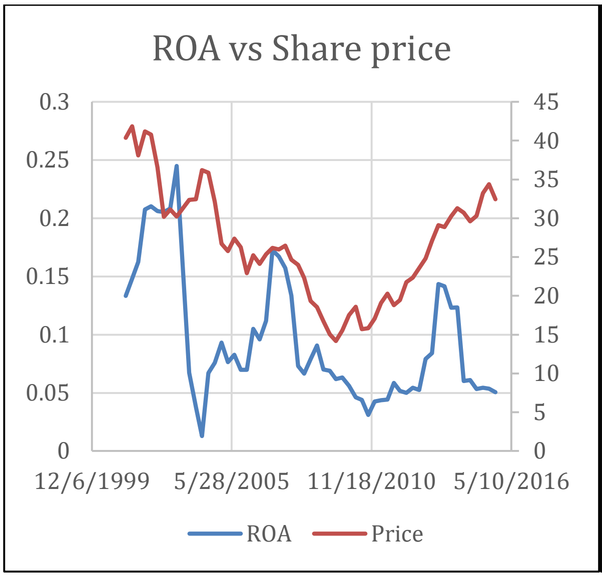

Revenue vs Share price
Billions
20
18
16
14
12
10
8
6
4
2
0
45
40
35
30
25
20
15
10
5
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
Revenue
Price

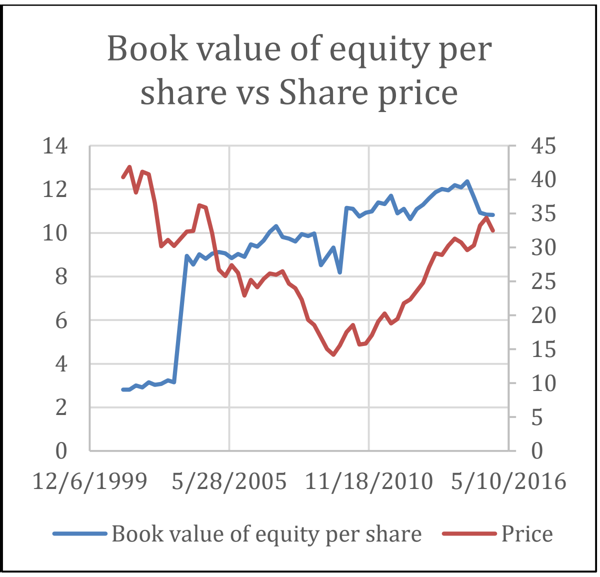

Earnings vs Share price
Billions
16
14
12
10
8
6
4
2
0
45
40
35
30
25
20
15
10
5
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
Earnings
Price

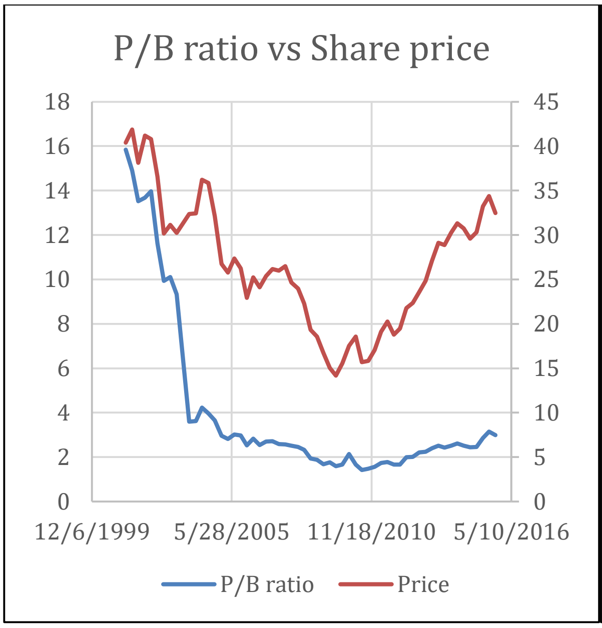

EPS basic vs Share price
2.5
2
1.5
1
0.5
0
45
40
35
30
25
20
15
10
5
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
EPS basic
Price

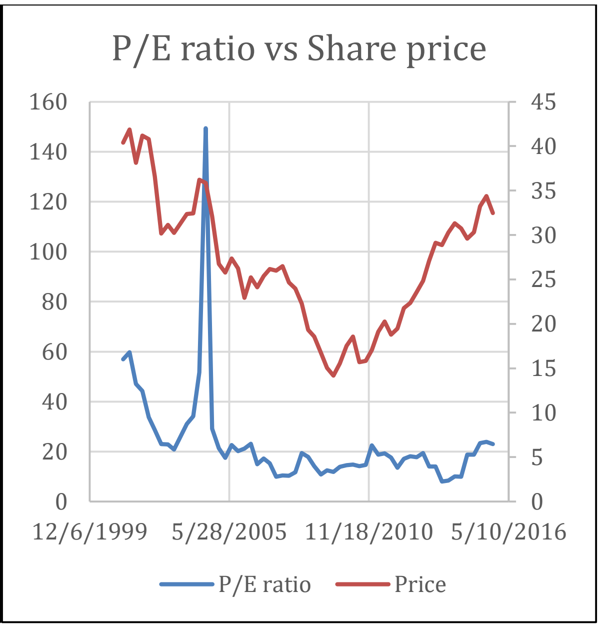

Dividend per share vs Share price
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
45
40
35
30
25
20
15
10
5
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
Dividend per share
Price

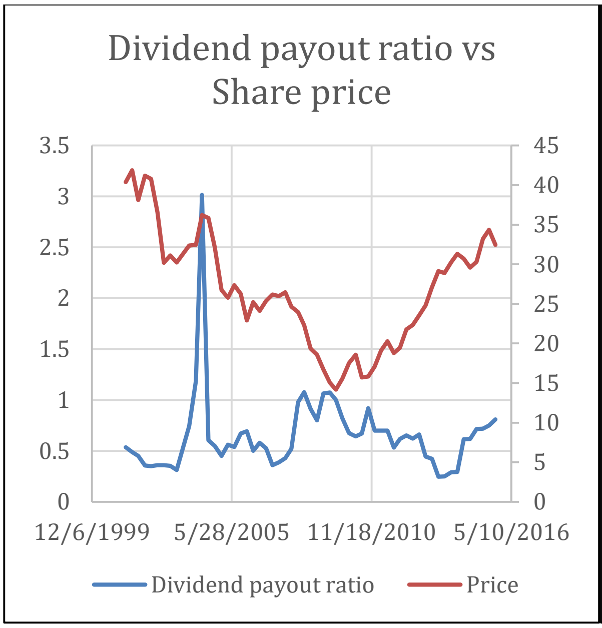

ROE vs Share price
0.6
0.5
0.4
0.3
0.2
0.1
0
45
40
35
30
25
20
15
10
5
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
ROE
Price

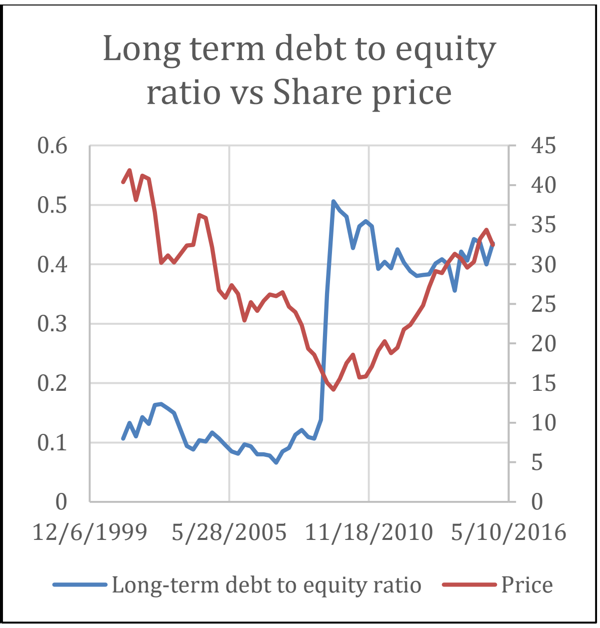

ROA vs Share price
0.3
0.25
0.2
0.15
0.1
0.05
0
45
40
35
30
25
20
15
10
5
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
ROA
Price

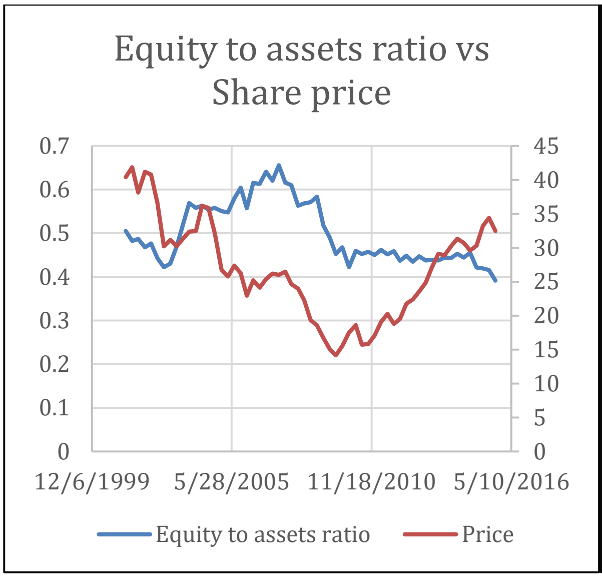

Book value of equity per share vs Share price
14
12
10
8
6
4
2
0
45
40
35
30
25
20
15
10
5
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
Book value of equity per share
Price

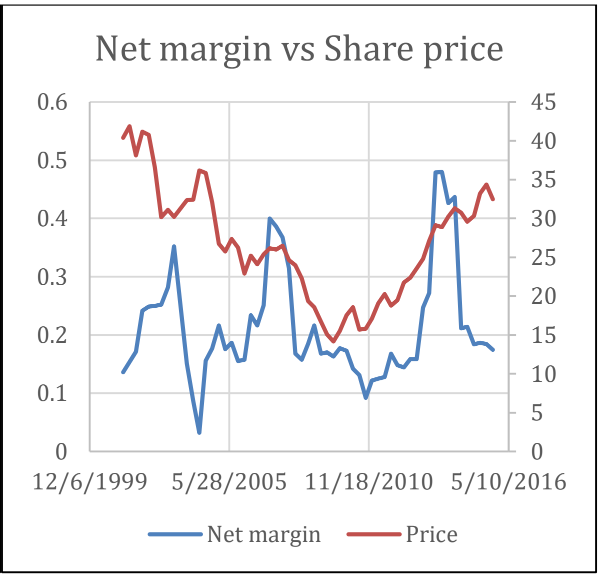

P/B ratio vs Share price
18
16
14
12
10
8
6
4
2
0
45
40
35
30
25
20
15
10
5
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
P/B ratio
Price

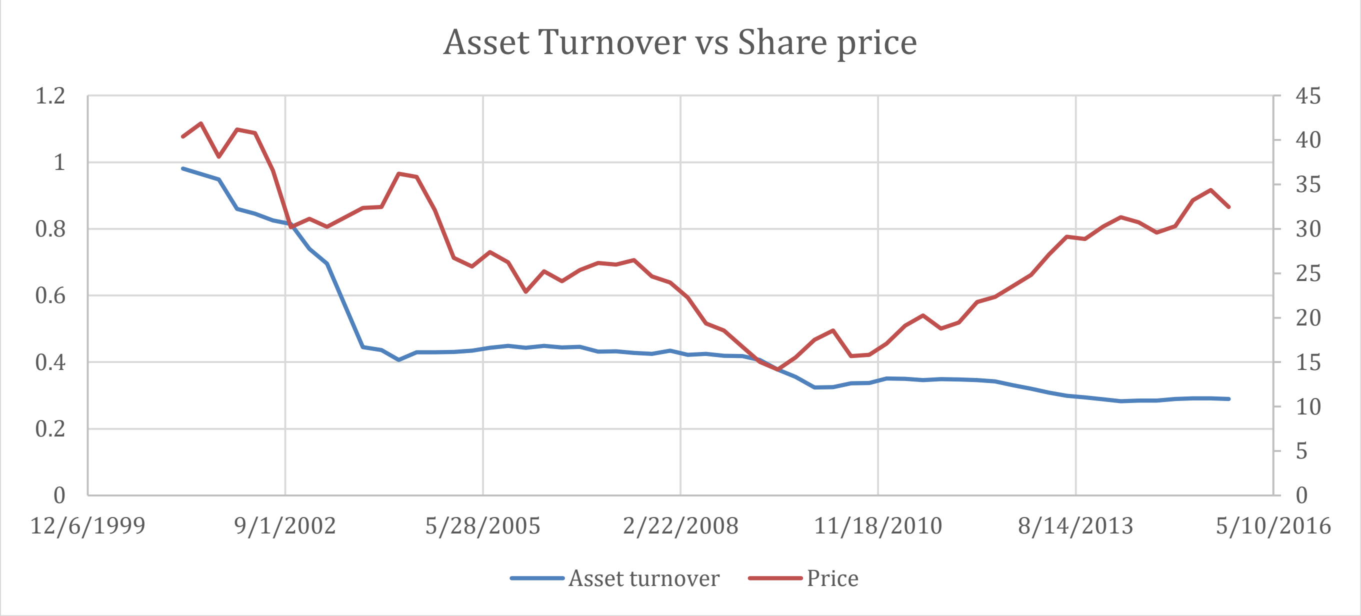

P/E ratio vs Share price
160
140
120
100
80
60
40
20
0
45
40
35
30
25
20
15
10
5
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
P/E ratio
Price


Dividend payout ratio vs Share price


Long term debt to equity ratio vs Share price

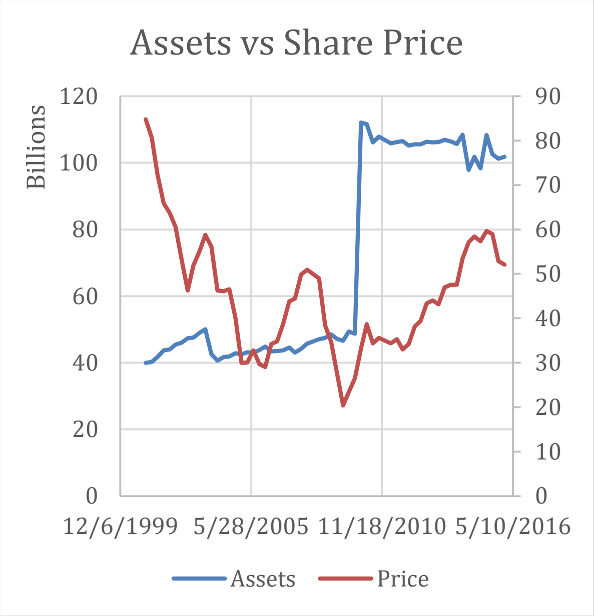

Equity to assets ratio vs Share price

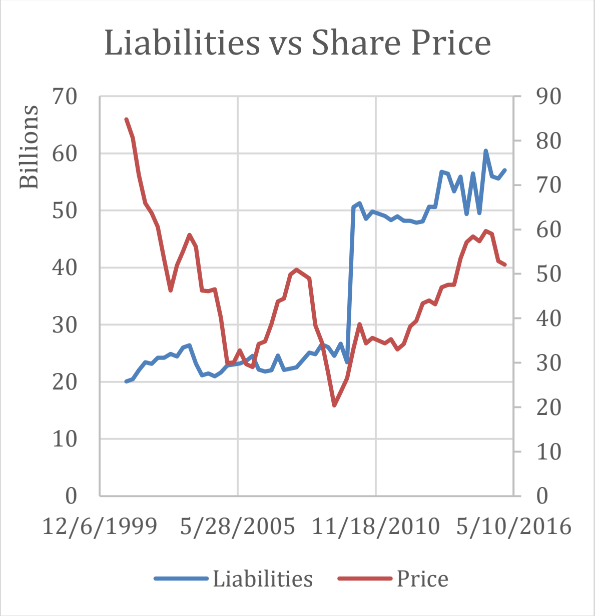

Net margin vs Share price

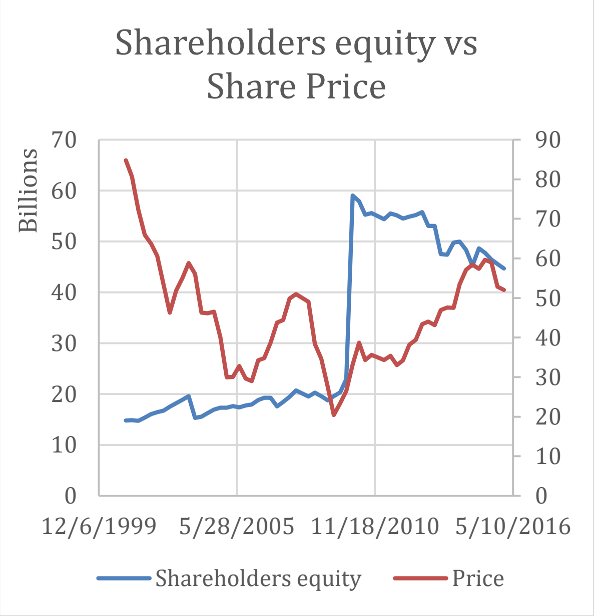


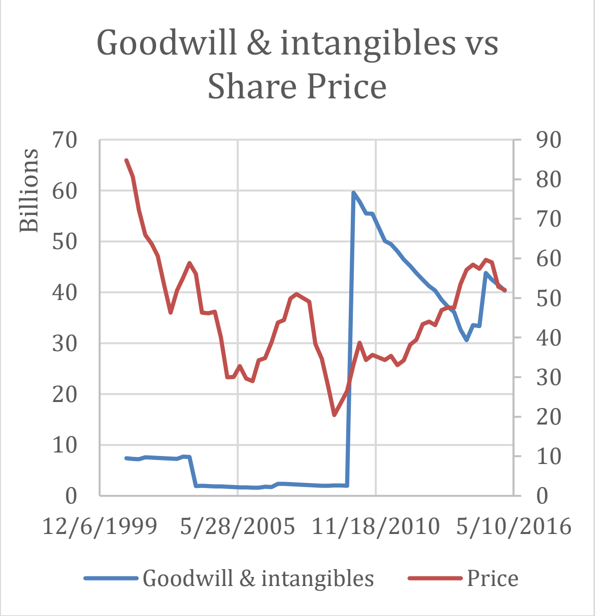
Asset Turnover vs Share price
1.2
1
0.8
0.6
0.4
0.2
0
45
40
35
30
25
20
15
10
5
0
12/6/1999
9/1/2002
5/28/2005
2/22/2008
11/18/2010
8/14/2013
5/10/2016
Asset turnover
Price

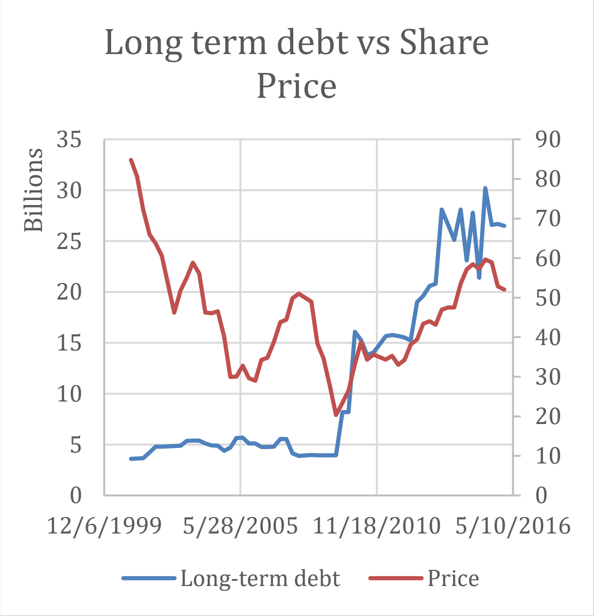
Merck & Co

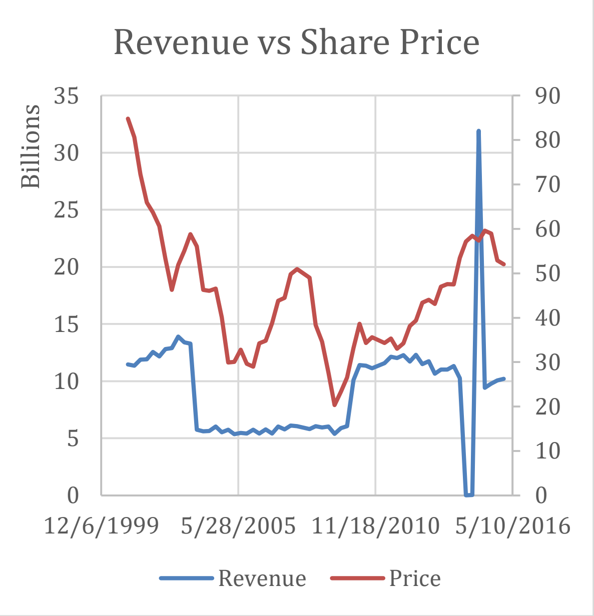

Assets vs Share Price

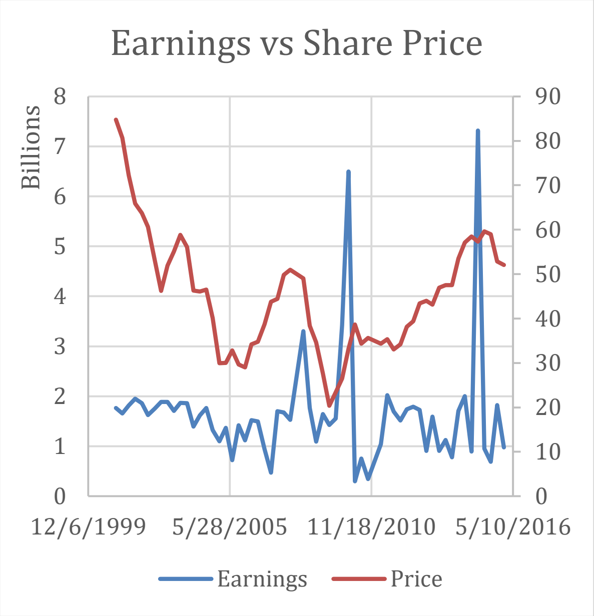

Liabilities vs Share Price

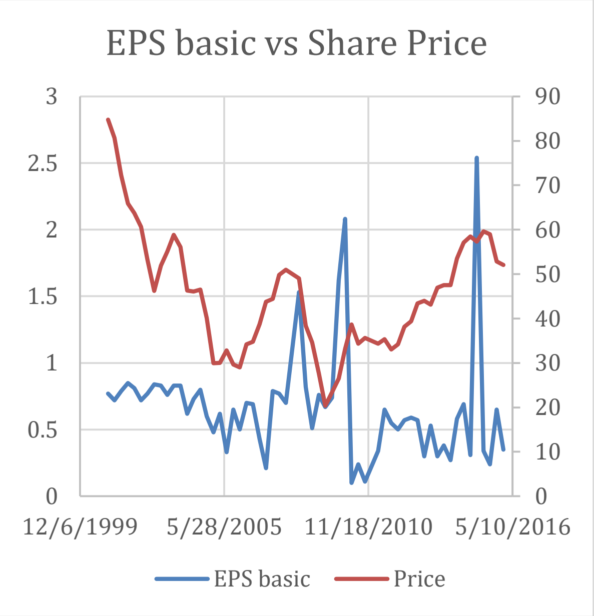

Shareholders equity vs Share Price

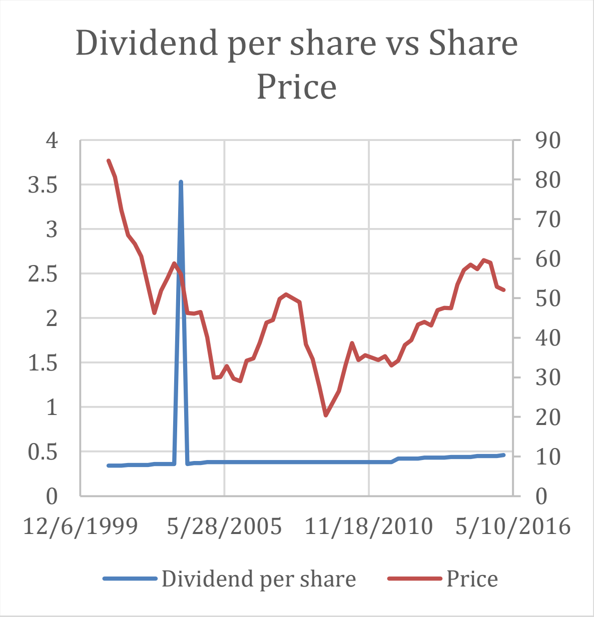

Goodwill & intangibles vs Share Price

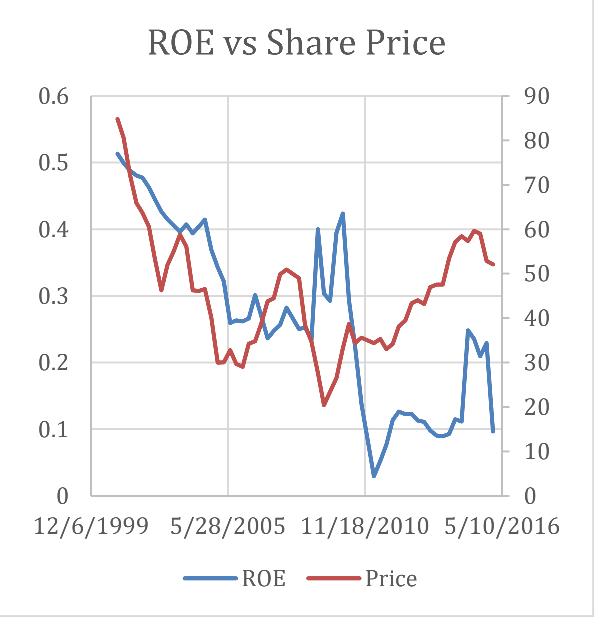

Long term debt vs Share Price
Billions
35
30
25
20
15
10
5
0
90
80
70
60
50
40
30
20
10
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
Long-term debt
Price

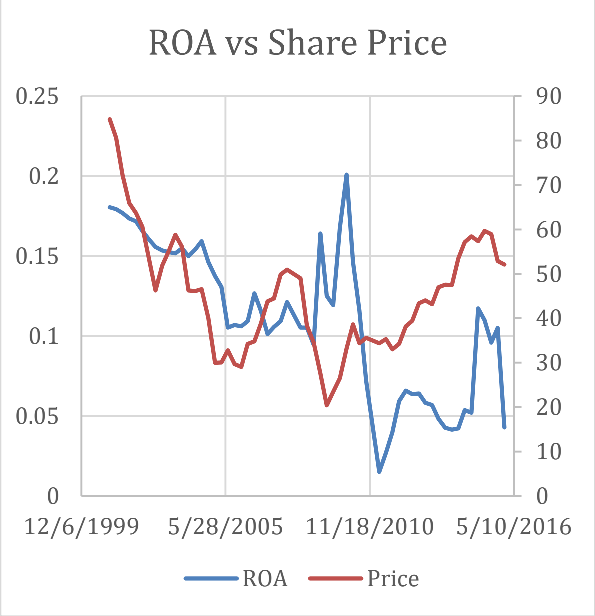

Revenue vs Share Price
Billions
35
30
25
20
15
10
5
0
90
80
70
60
50
40
30
20
10
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
Revenue
Price

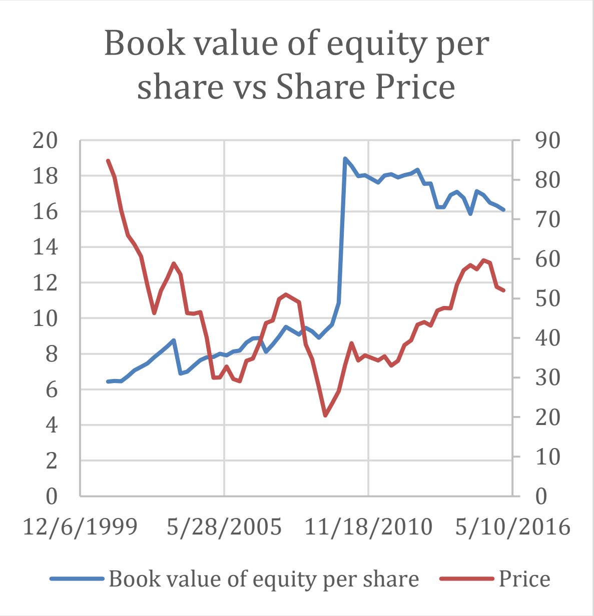

Earnings vs Share Price
Billions
8
7
6
5
4
3
2
1
0
90
80
70
60
50
40
30
20
10
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
Earnings
Price

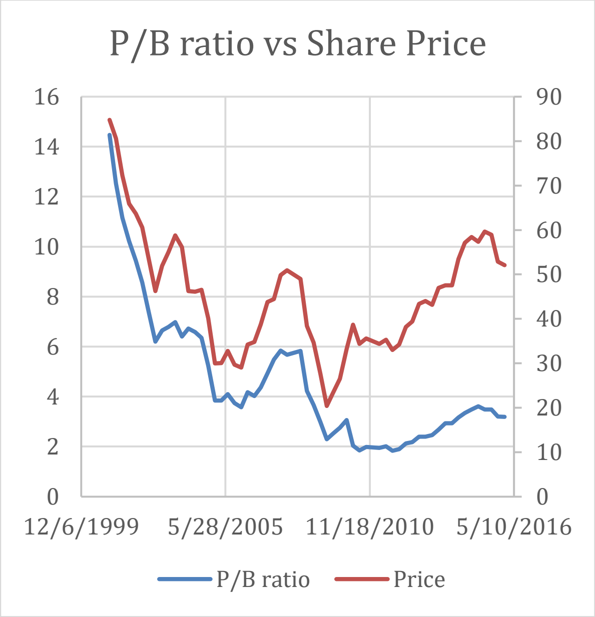

EPS basic vs Share Price
3
2.5
2
1.5
1
0.5
0
90
80
70
60
50
40
30
20
10
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
EPS basic
Price

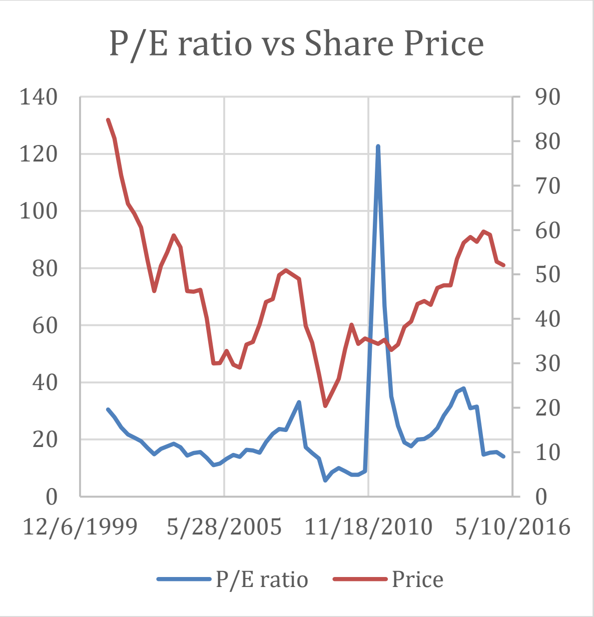

Dividend per share vs Share Price
4
3.5
3
2.5
2
1.5
1
0.5
0
90
80
70
60
50
40
30
20
10
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
Dividend per share
Price

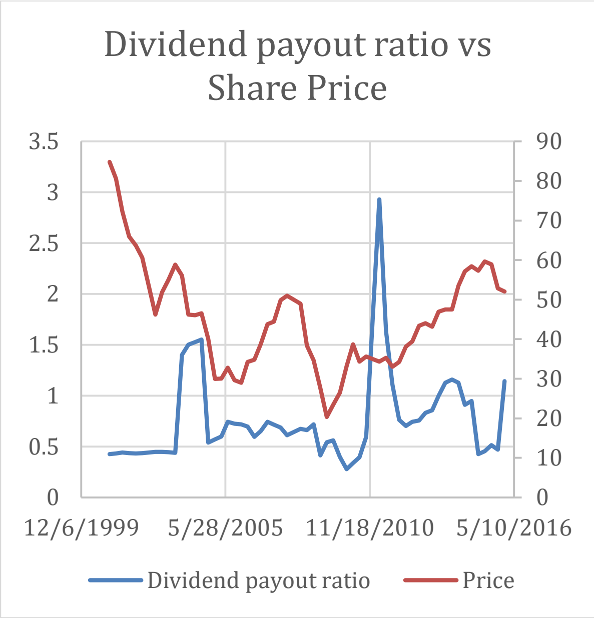

ROE vs Share Price
0.6
0.5
0.4
0.3
0.2
0.1
0
90
80
70
60
50
40
30
20
10
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
ROE
Price

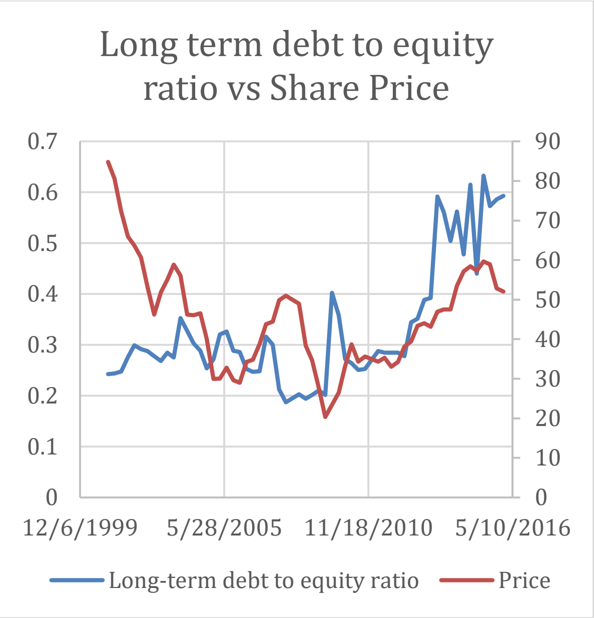

ROA vs Share Price
0.25
0.2
0.15
0.1
0.05
0
90
80
70
60
50
40
30
20
10
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
ROA
Price

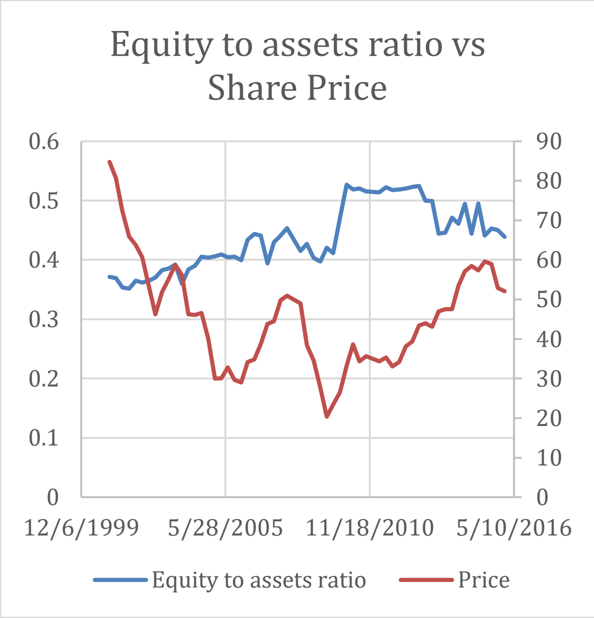

Book value of equity per share vs Share Price
20
18
16
14
12
10
8
6
4
2
0
90
80
70
60
50
40
30
20
10
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
Book value of equity per share
Price

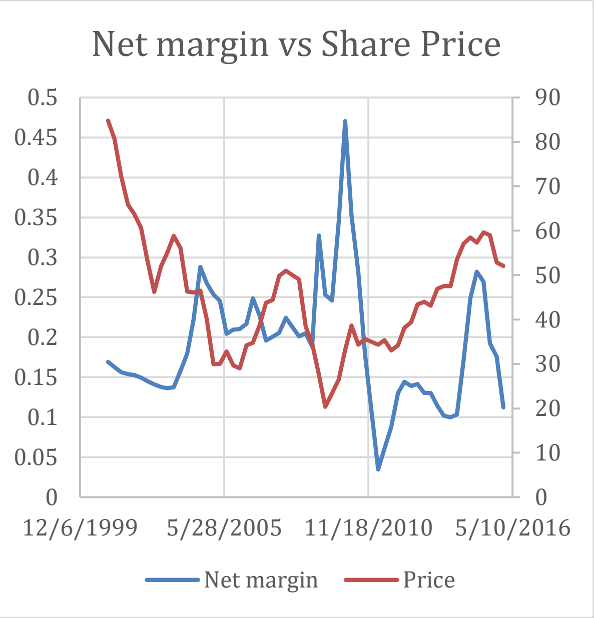

P/B ratio vs Share Price
16
14
12
10
8
6
4
2
0
90
80
70
60
50
40
30
20
10
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
P/B ratio
Price

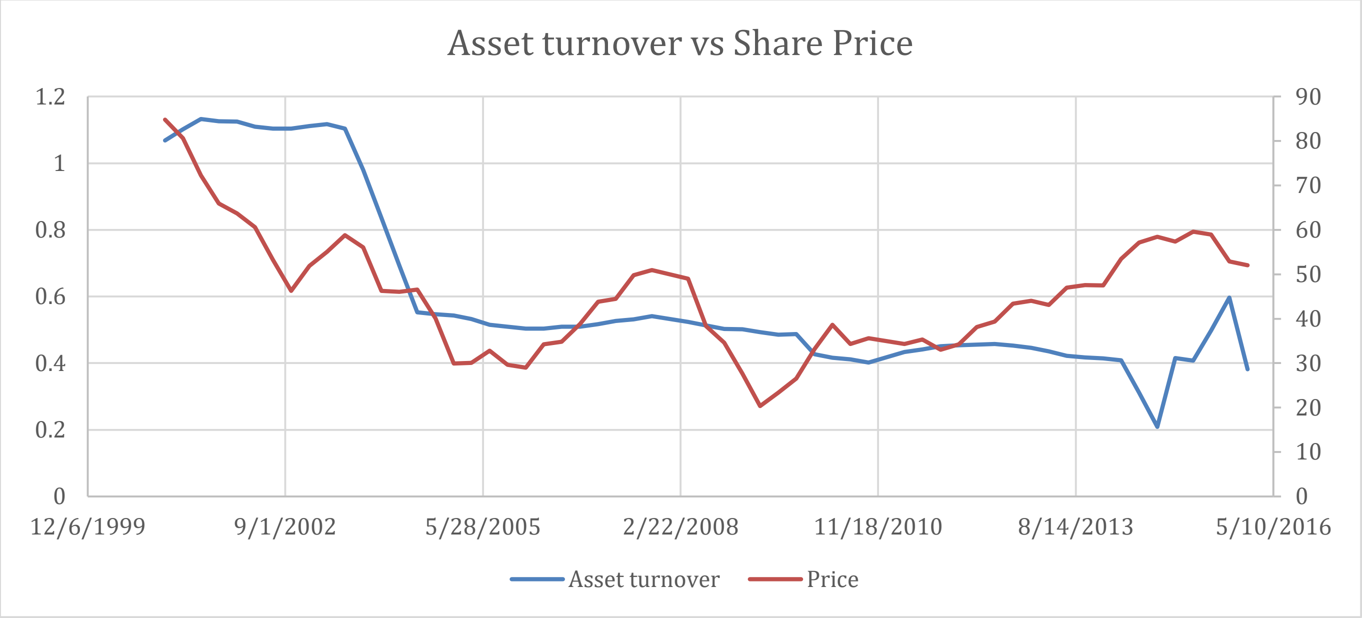

P/E ratio vs Share Price
140
120
100
80
60
40
20
0
90
80
70
60
50
40
30
20
10
0
12/6/1999
5/28/2005
11/18/2010
5/10/2016
P/E ratio
Price


Dividend payout ratio vs Share Price


Long term debt to equity ratio vs Share Price


Equity to assets ratio vs Share Price

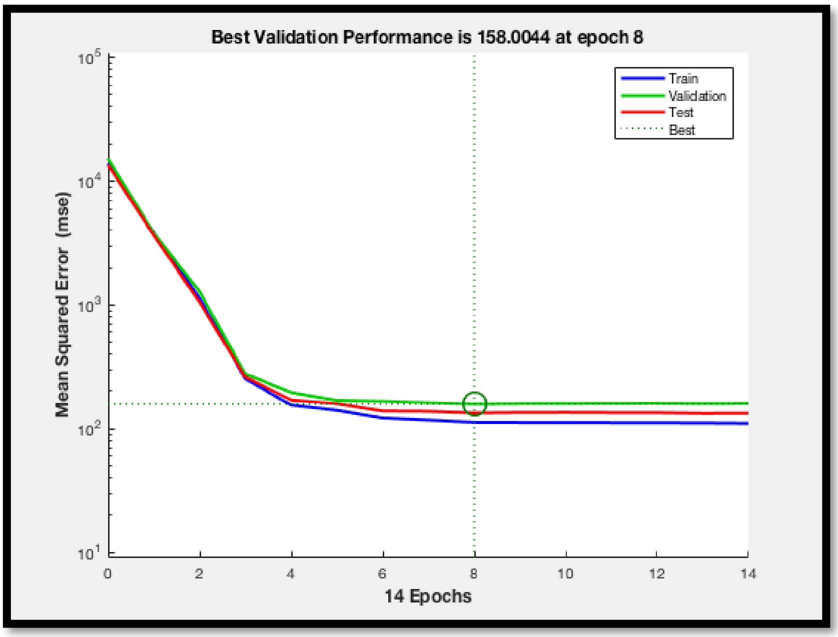

Net margin vs Share Price

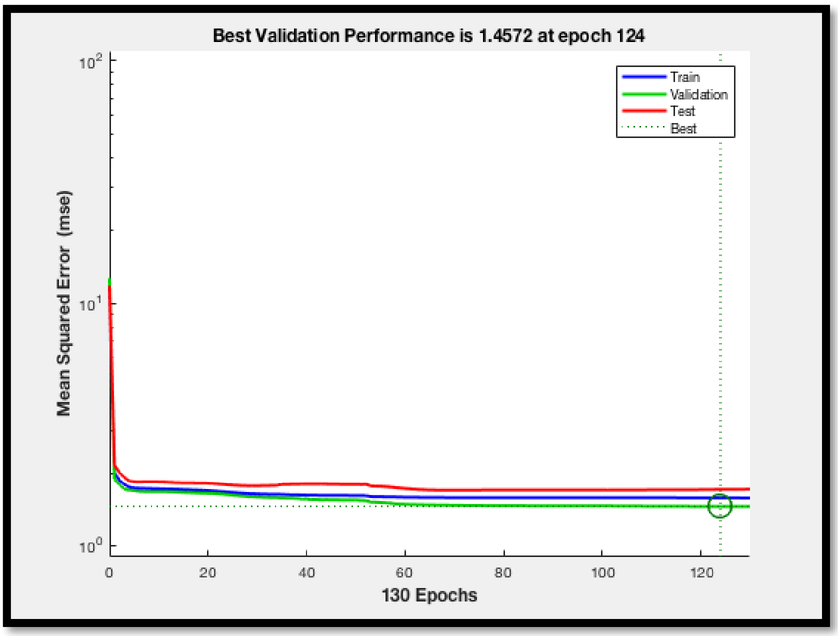

Asset turnover vs Share Price
1.2
1
0.8
0.6
0.4
0.2
0
90
80
70
60
50
40
30
20
10
0
12/6/1999
9/1/2002
5/28/2005
2/22/2008
11/18/2010
8/14/2013
5/10/2016
Asset turnover
Price

Appendix C: Initial Performance Plots of Stocks

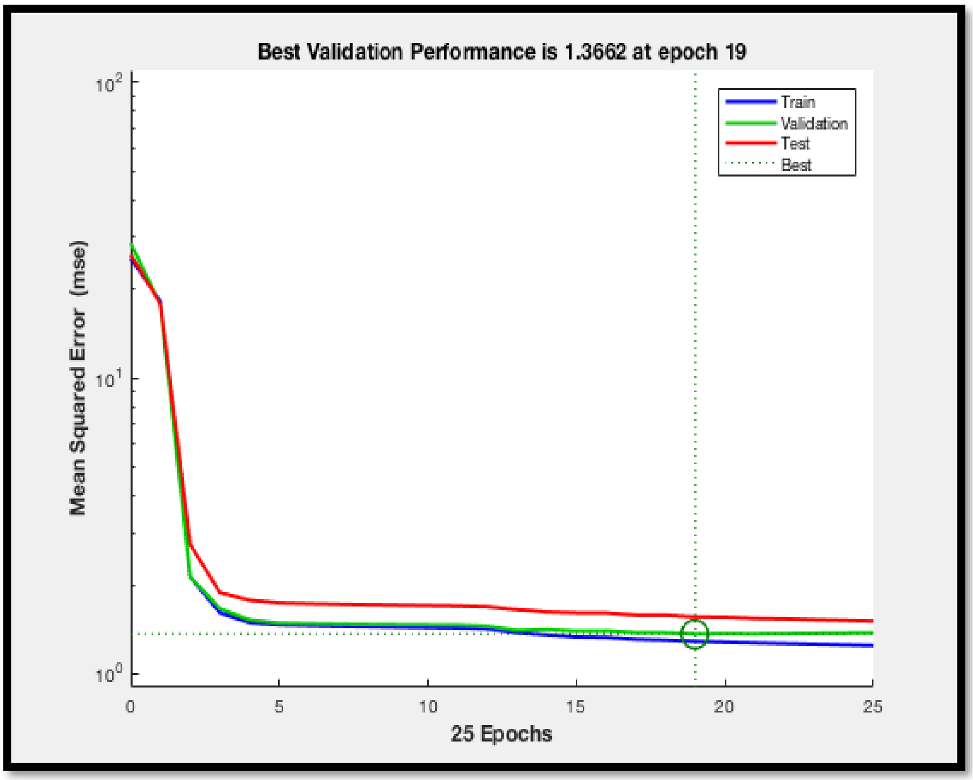
In the experimental stage, understanding how the system works was imperative so countless graphs were created (around 200) with different parameters. Attaching all of them will not only be a useless exercise but also take 30 to 40 pages so in order to give a fair view of capability of a well-trained and optimised ANN, two graphs for each stock will be attached; one graph shows performance plot of a network trained on default settings whilst other shows performance plot of a network trained on carefully optimised settings.

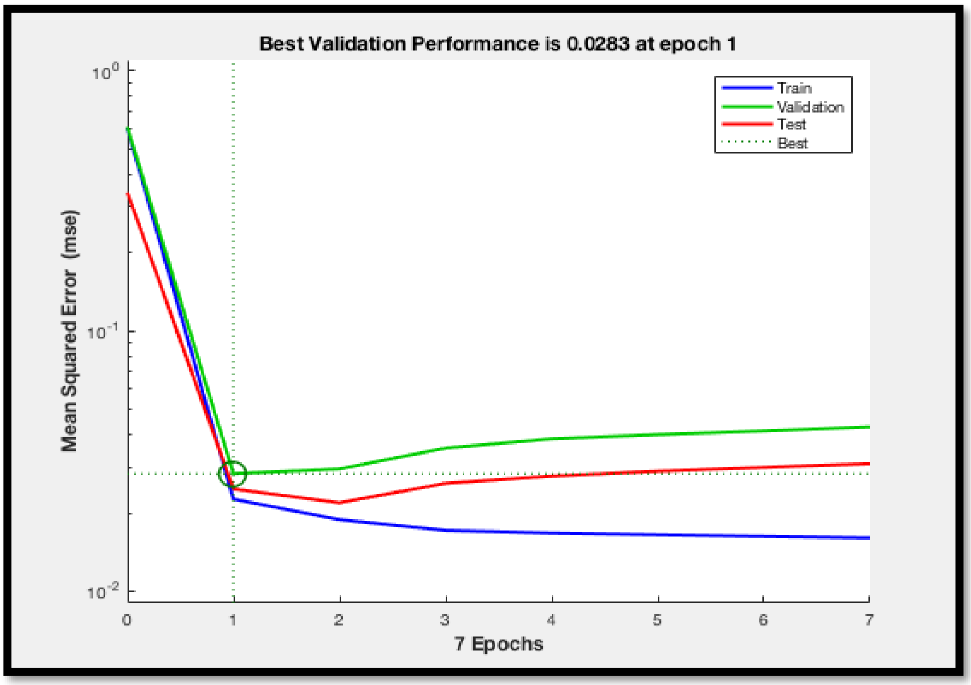
Exxon



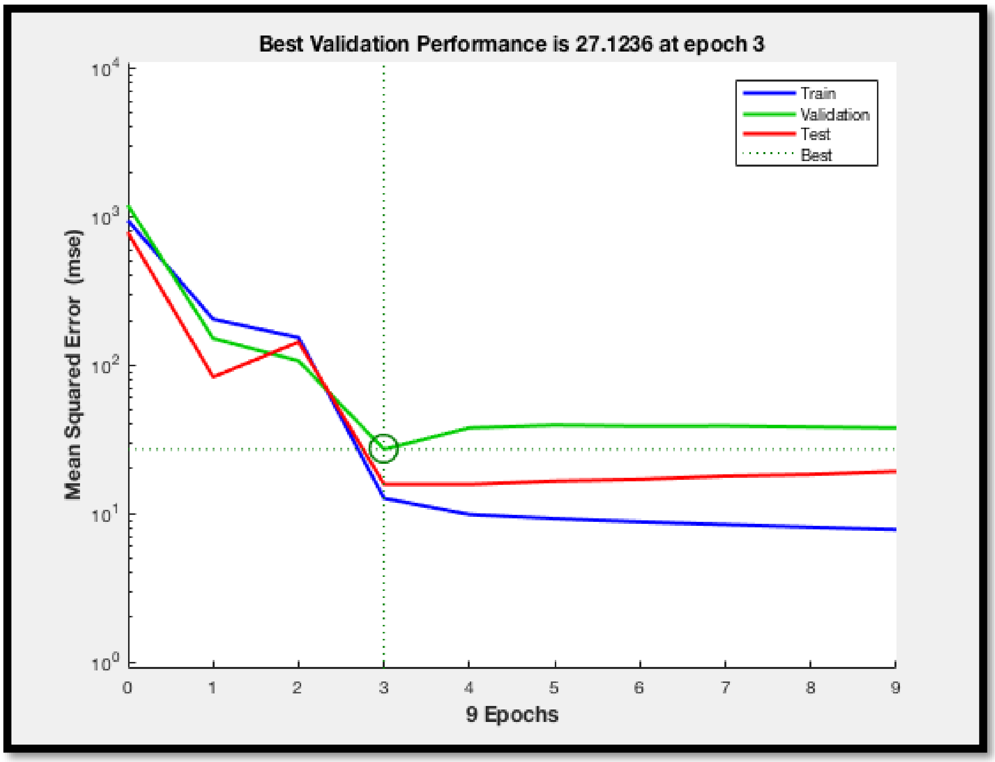


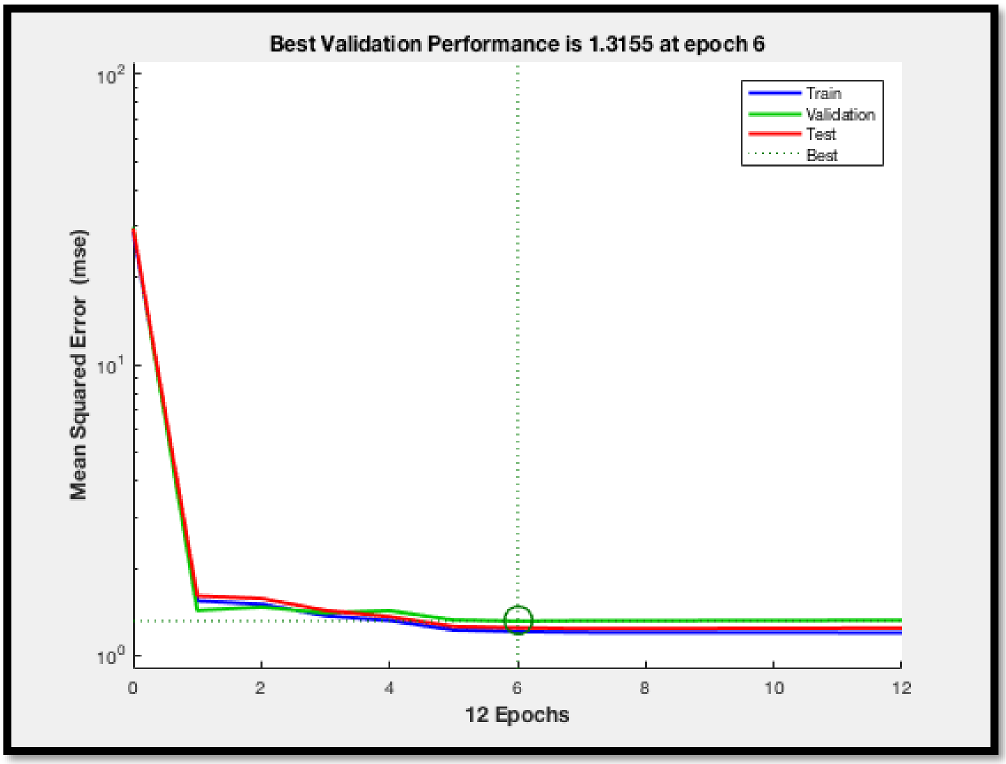
Chevron


Best Validation Performance is 1.3662 at epoch 19
Mean Squared Error (mse)
10^2
10^1
10^0
0
5
10
15
20
25
25 Epochs
Train
Validation
Test
Best

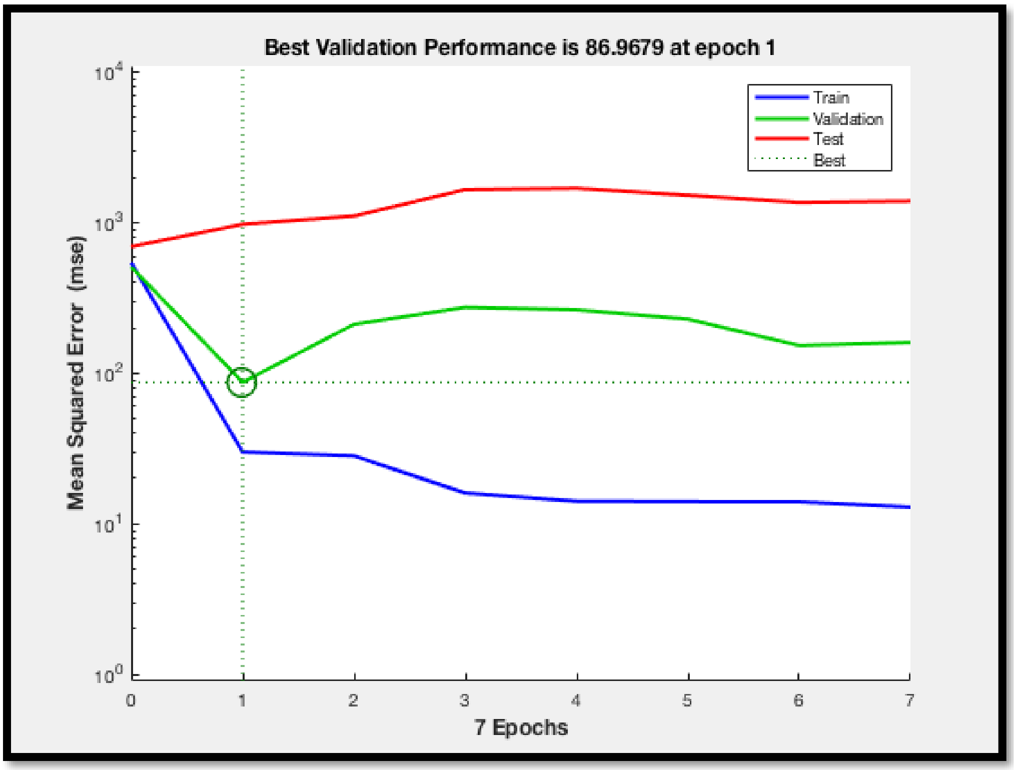

Best Validation Performance is 0.0283 at epoch 1
Mean Squared Error (mse)
10^0
10^{-1}
10^{-2}
0
1
2
3
4
5
6
7
7 Epochs
Train
Validation
Test
Best

Pfizer

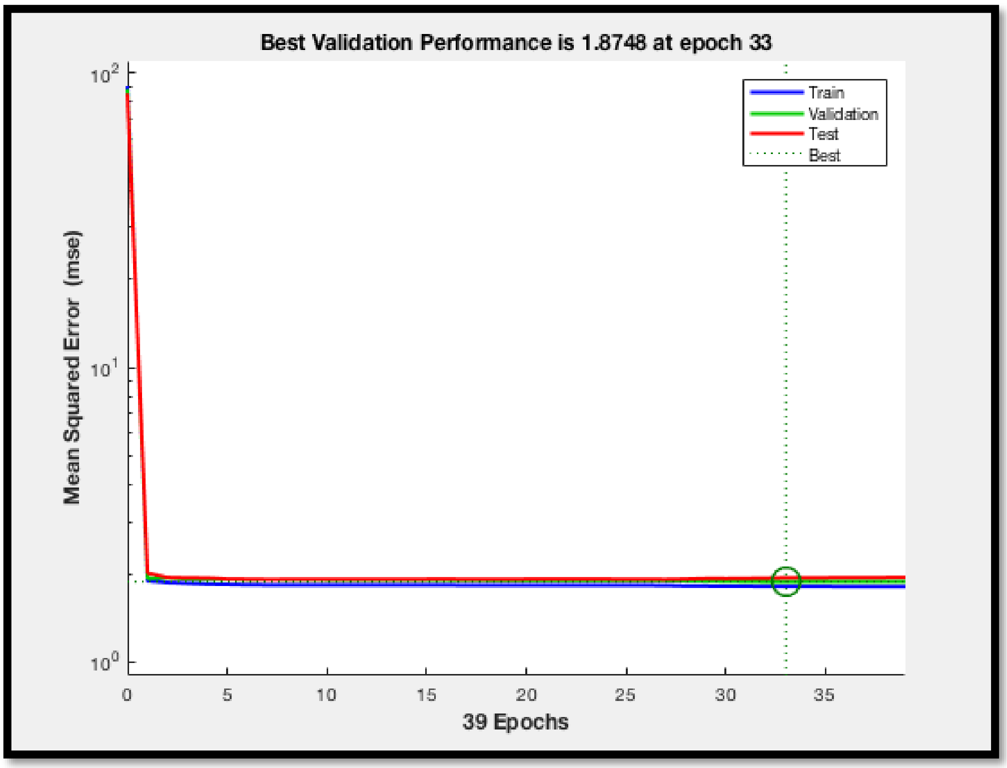

Best Validation Performance is 27.1236 at epoch 3
Mean Squared Error (mse)
10^4
10^3
10^2
10^1
10^0
0
1
2
3
4
5
6
7
8
9
9 Epochs
Train
Validation
Test
Best


Best Validation Performance is 1.3155 at epoch 6
Mean Squared Error (mse)
10^2
10^1
10^0
0
2
4
6
8
10
12
12 Epochs
Train
Validation
Test
Best

Merck



Best Validation Performance is 86.9679 at epoch 1
Mean Squared Error (mse)
10^4
10^3
10^2
10^1
10^0
0
1
2
3
4
5
6
7
7 Epochs
Train
Validation
Test
Best



Best Validation Performance is 1.8748 at epoch 33
Mean Squared Error (mse)
10^2
10^1
10^0
0
5
10
15
20
25
30
35
39 Epochs
Train
Validation
Test
Best

References

Atsalakis, G. S., & Valavanis, K. P. (2009). Surveying stock market forecasting techniques – Part II: Soft computing methods. *Expert Systems with Applications, 36*(3), 5932–5941.

Ayodele, A., Charles, A., Marion, A., & Sunday, O. (2012, January). Stock Price Prediction using Neural Network with Hybridised Market Indicators. *Journal of Emerging Trends In Computing and Information Sciences, 3*(1).

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

Barber, D. (2005). *Learning from Data Layered Neural Networks.* The University of Edinburgh .

Bhargava, N., & Gupta, M. (2008). *Application of Artificial Neural Networks in Business Applications.* IIT Delhi.

Bodis, L. (2004). *Financial Time Series Forecasting Using Artificial Neural Networks .* Babes Bolyai University .

Campbell, J. Y., & Shiller, R. J. (2001). *Valuation Ratios and the Long-Run Stock Market Outlook: An update.* Yale University .

Clarke, J., Jandik, T., & Mandelker, G. (n.d.). *The Efficient Markets Hypothesis.*

Crone, S. F. (2004). Stepwise selection of artificial neural network models for time series prediction. *Journal of Intelligent Systems , 14*(2).

Dimson, E., & Mussavian, M. (1998). A brief history of market efficiency. *European Financial Management Jornal, 4*(1), 91-193.

Dorsey, R., & Sexton, R. (2000). The Use of Parsimonious Neural Networks for Forecasting Financial Time Series. *Finance and Technology*.

Emir, S., Dincer, H., & Timor, M. (2012). A Stock Selection Model Based on Fundamental and Technical Analysis Variables by Using Artificial Neural Networks and Support Vector Machines. *Review of Economics & Finance*.

Fifield, S. G., Power, D. M., & Sinclair, C. D. (2002). Macroeconomic Factors and Share Returns: An Analysis Using Emerging Marke Data. *International Journal of Finance And Economics*, 51-62.

Funahashi, K.-I. (1989). On the approximate realization of continuous mappings by neural networks. *Neural Networks, 2*(3), 183–192.

Gornall, J. F. (2013). *Exloring the potential of a Novel Time Series Prediction Algorithm.* London.

Graham, B. (1973). *The Intelligent Investor* (4th ed.). Harper Collins.

Hastie, T., Tibshirani, R., & Friedman, J. (2008). *The Elements of Statistical Learning* (Second ed.). Springer.

Hui, S. C. (2000). A Hybrid Time Lagged Network for Predicting Stock Prices. *International Journal of Computer Integrated Manufacturing, 8*(3).

Hush, D. R., & Hush, B. G. (1993, january). Progress in Supervised Neural Networks. *IEEE Signal Processing Magazine, 10*(1), 8-39.

Kar, A. (n.d.). *Stock Prediction using Artifificial Neural Networks.* IIT Kanpur, Department of Computer Science and Engineering.

Kartalopoulos, S. V. (1995). *Understanding Neural Networks and Fuzzy logic.*

Lawrence, J. (1994). Introduction to Neural Networks: Design, Theory, and Applications. California Scientific Software Press.

Luke Biermann. (2006). *An Investigation Into Stock Market Predictions Using Neural Networks Applied to Fundamental Financial data.* University of Bath, Bath.

M.Lane, K., & Neidinger, R. D. (1994). *Neural Networks from Idea to Implementation.* Davidson College.

Machiel, L. S., & Ballini, R. (2008). *Design a Neural Network for Time Series Financial Forecasting: Acuuracy and Robustness Analysis.* Sao Paulo.

Malkiel, B. G. (2003). *The Efficient Market Hypothesis and Its Critics.* Princeton.

MATHWORKS. (n.d.). *MATLAB*. Retrieved from MATLAB: http://uk.mathworks.com/products/matlab/

Mia, M. M., Biswas, S. K., Urmi, M. C., & Siddique, A. (2015). An Algorithm For Training Multilayer Perceptron For Image Reconstruction Using Neural Network Without Overfitting. *International Journal of Scientific and Technology Research, 4*(2).

Michie, D., Spiegelhalter, D., & Taylor, C. (2009). *Machine Learning, Neural and Statistical Classification.* Overseas Press.

Mitchell, S. (1995). *The Application of Machine Learning Techniques to Time-Series Data.* Waikato.

Oprean, C. (2012). A Behavioral Finance Perspective Of The Efficient Market Hypothesis. *Annals of Computational Economics, 3*(40).

Patel, M. B., & Yalamalle, S. R. (2014, June). Stock Price Prediction Using Artificial Neural Network. *International Journal of Innovative Research in Science, Engineering and Technology, 3*(6).

Refenes, A. P. (1994). *Neural Networks in Capital Markets.* John Wiley and Sons.

Shachmurove, Y. (2005). Business Aplcations of Emulative Neural Networks. *International Journal of Buisness* (1083-4346).

Shih, Y. (1994). *Neuralyst User's Guide.* Cheshire Engineering Corporation.

Statsoft. (2010). Retrieved October 30, 2010, from www.statsoft.com/textbook/neural-networks/apps

Taskin Kocak. (2007). *Sigmoid Functions and Their Usuage in Artificial Neural Networks.* University of Central Florida, Florida.

Vanstone, B., Finnie, G., & Tan, C. (2004). *Applying fundamental analysis and neural networks in the Australian stock market.* Bond University.

Waikato, U. o. (n.d.). *WEKA.* Univesity of Waikato. Retrieved from WEKA: http://www.cs.waikato.ac.nz/ml/weka/

Wani, M. A. (2013, December). Comparative Study of Back Propagation Learning Algorithms For Neural Networks. *International Journal of Advanced Research in Computer Science and Software Engineering, 3*(12).

Zekic, M. (1998). *Neural Network Applications in Stock Market Predictions.* University of Josip Juraj Strossmayer, Osijek.

Zhang, G. P. (2003). Forecasting Stock Returns With Artificial Neural Networks. In *Neural Networks in Business Forecasting* (p. 350). IGI Global.