Novel Neural Network Models for Financial Prediction

Asmaa Abdulhussein Mahdi

Supervisor: Dr. Tillman Weyde

School of Mathematics, Computer Science and Engineering

A thesis submitted in partial fulfilment of the requirement for the degree of
Doctor of Philosophy

City, University of London   January 2020

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 65,000 words including appendices, bibliography, footnotes, tables and equations and has fewer than 100 figures.

Asmaa Abdulhussein Mahdi
January 2020

Acknowledgements

All praise and thanks are due to "Allah" the Almighty who granted me the faith and ability to complete this scientific research.

My acknowledgements to my country "The Republic of Iraq" which through the scholarship helped in fulfilling my dream of getting a PhD degree. My thanksgiving to all the martyrs who offered their souls for the right and for saving people's lives.

I would like to express my gratitude to my supervisor "Dr.Tillman Weyde", who supported me with his aspiring guidance, constructive criticism, and friendly advice. Also, my sincere thanks to the "School of Mathematics, Computer Science and Engineering" for providing me with all the necessary facilities.

My thanks go to my father, who deserves special thanks for all his support, his prayers and his help in the difficult times. As well as, to my deceased mother, although she is no longer with me, I am sure that success in achieving my PhD degree would be the best gift I could have offered to her soul.

I would like to express my deepest appreciation and gratitude to my husband "Ali", who is inspired and encouraged me from the first day of getting the PhD scholarship, and all his effort that he provided to help me to complete this research. It won't be justified if I do not pay my thanks to my little angel "Mariam".

I am thankful to my brothers, sisters, friends, colleagues and all those who provided me help and support throughout this research project. I am grateful to all for their prayers and well wishes.

Abstract

Financial markets are an important feature of modern economies, where trading decisions can be critical because of their significant impact on social and economic life. Various models and techniques have been applied to describe and predict financial time series in order to develop effective tools in financial prediction. In particular, neural networks have recently gained significant research interest in financial markets as well as in other domains. As financial time series data show a high degree of non-linearity, neural networks represent an attractive approach in this area.

This thesis introduces a novel neural network model, the FL-SMIA model, as well as several variations and extensions, namely the FL-SMIA*, D-FL-SMIA, MD-FL-SMIA, MD-FL-SMIA-2, M-FL-SMIA, and FL-SMIA-RBM. The FL-SMIA model is a model that uses the principles of the Functional Link Neural Network (FLNN) and the Self-organising Multilayer Neural Network using the Immune Algorithm (SMIA). The FL-SMIA model combines the higher-order inputs , i.e. the products of raw input features, with the self-organising hidden layer (SMIA) that dynamically grows and adapts to the input vectors.

Based on the promising results of the FL-SMIA network in initial experiments, variations and extensions have been developed using deeper architectures (D-FL-SMIA), mixed input representations (M-FL-SMIA), a combination of deep and mixed architectures (MD-FL-SMIA), and of the FL-SMIA with the Restricted Boltzmann Machine in the FL-SMIA-RBM. The proposed models have also been compared with other neural network architectures: FLNN, the Multilayer perceptron (MLP), and SMIA.

All networks have been evaluated for one day and five days ahead prediction using financial and statistical metrics, focusing on the Relative Profit (RP) and Annualised Volatility (AV). Data-sets of three different types have been used: exchange rates (USD/UKP, USD/EUR, JPY/USD), stock price indices (NASDAQ, DJIA), and commodity prices (OIL and GOLD).

In terms of average RP results for the one day ahead prediction, the FL-SMIA was slightly worse than the best model (FLNN) but FL-SMIA model reduced the

investment risk by producing the lowest average AV value. We have also observed notable differences between data types.

For the five days ahead prediction, the M-FL-SMIA model has the highest average RP and the lowest average AV results. Correlation analysis on the residuals has shown differences in behaviour between FLNN model and FL-SMIA model, encouraging further extensions and variations.

Overall, the FL-SMIA model and its extensions will be useful for time series prediction because of their competitive performance and different behaviour to standard neural networks.

Table of contents

List of figures

List of tables

Nomenclature

FLNN Functional Link Neural Network

MLP Multi Layer Perceptron

SMIA Self-organised Multilayer neural network using the Immune Algorithm

FL-SMIA Functional Link - Self organising Multilayer network using the Immune Algorithm

D-FL-SMIA Deeper learning - FL-SMIA network

MD-FL-SMIA Mixed and Deeper - FL-SMIA network

M-FL-SMIA Mixed of FLNN and FL-SMIA networks

FL-SMIA-RBM FL-SMIA with Restricted Boltzmann Machine network

Chapter 1

Introduction

1.1 Financial Prediction

Over the last 20 years, the problem of predicting financial time-series has attracted much interest from both commercial and academic communities, which resulted in a wide range of investigations. Predictive models are contributing to the decisions on economic policies by governments and investments by multinational companies which rely on computer modelling and forecasts [10, 94, 156]. Financial time series are highly non-linear and complex [95] because of many of the risk factors, such as political events, weather conditions, and dynamics of the financial market themselves, which are affecting the stock prices and exchange rates [22]. Artificial Neural Networks (ANNs) as non-linear models have long been seen as promising and have been used extensively in financial time series prediction [59, 154, 12, 110], but they suffer from some problems, particularly over-fitting on smaller data-sets [90, 137]. In 1987, Giles [53] introduced the *Higher Order Neural Network*, which led to [117] introducing the *Functional Link Neural Network* (FLNN). The FLNN was presented to reduce the over-fitting problem by removing the hidden layer from the standard Multilayer Perceptron (MLP) architecture to help reduce the model complexity. Instead, the FLNN uses products of input units to enable the network to perform non-linearly separable classification tasks.

Another approach to improve the MLP model that based on alternative learning methods using to prototypes or clustering, such as Adaptive Resonance Theory [20, 57, 17], or algorithms inspired by immune systems [31] such as the *Self-organised Multilayer neural network using the Immune Algorithm* (SMIA) [101], where the internal representations expand depending on the training data.

1.2 Artificial Neural Networks for Financial Prediction

Artificial Neural Networks (ANNs) have been used extensively for financial prediction because of their adaptability and applicability as non-linear models [137, 74, 10]. Neural network models are capable of learning any non-linear function, as they have been proven to be universal function approximators [72]. The abilities of neural networks in learning complex problems in highly non-linear data have been demonstrated by many applications of neural networks. Neural networks often perform better than traditional statistical methods in financial prediction domain [35].

Many studies in financial prediction have shown good results of neural networks when predicting time series data in different fields and they have generated much interest [95, 141, 142, 154]. The main concern of researchers in the domain of financial prediction with using neural networks is to improve the speed and accuracy of the prediction [141, 40, 52].

When neural networks are employed, two important aspects must be considered:

1. **Architecture Selection:** this point refers to the number of input units, as well as the number of units in the hidden and output layers, and if feedback loops are presented or not.

2. **Learning Algorithm:** different learning algorithms are used to enable the network to achieve the task for which it is being used.

In financial forecasting, many studies tested statistical models and artificial neural networks using various financial time series [152, 64, 154, 35, 75, 116, 36, 81]. They all found that neural network models outperform the traditional statistical models. Furthermore, in a survey on 45 articles published between 1993 and 2004, which analysed the results of forecasting applications when using neural networks in exchange rates [50], most of these studies use gradient descent with the back-propagation learning method. However, since those years many methods have been employed to improve the classical back-propagation algorithm in finding optimised techniques.

1.3 Aims and Objectives

This research focuses on improving the prediction of artificial neural networks by using different self-organising models in addition to the back-propagation algorithm in artificial neural networks.

1.3.1 Aims

The two main aims of this research are the following:

1) Propose novel neural network models for financial prediction.

2) Evaluate all models on different types of financial time series.

1.3.2 Objectives

The specific objectives are:

For aim 1) *Propose novel neural network models for financial prediction* the objectives are the development and implementation of the following:

a) FL-SMIA model: Integration of the principle of the Functional Link Neural Network (FLNN) with the Self-organising Multilayer Neural Network using the Immune Algorithm (SMIA).

b) FL-SMIA* model: Variant of the FL-SMIA network by using a different method to update the connection weights from input units to hidden units.

c) Extensions of the FL-SMIA network using various neural network techniques:

   1. D-FL-SMIA model: developing the FL-SMIA network to a deeper learning network using a number of standard hidden layers.
   2. M-FL-SMIA model: a mixed model that combines the FL-SMIA model and the direct input representations of the FLNN network.
   3. MD-FL-SMIA model: combination model of the mixed architecture of FL-SMIA of input representations and deeper learning using a number of standard hidden layers.

d) FL-SMIA-RBM: this model includes two hidden layers using two unsupervised learning methods. The first layer uses the immune algorithm used with the FL-SMIA model, while the second layer uses the Restricted Boltzmann Machine (RBM) method.

For aim 2) *Evaluate all proposed models on different types of financial time series* the objectives are to perform the following:

a) Evaluate on three types of financial data including exchange rates (USD/UKP, USD/EUR, JPY/USD), stock price indices (NASDAQ, DJIA), and commodity prices (OIL and GOLD), as well as improved objectives.

b) Evaluate on financial (RP, AV, MDD) and statistical (MSE, SNR, CDC, MAE) metrics.

c) Evaluate for different time horizons.

d) Perform correlation analysis and significance tests.

1.4 Methodology

We use the research paradigm of experimental research using predictive models on historical data. The methodology of this research is based on proposing a number of predictive models using historical data in order to predict future values for financial data-sets.

We make assumptions about similarity between past and future. This is not easy to justify, since financial data is highly non-stationary, but since we are working with some success on long time series, there seems to be some reason to believe that it is possible. It is known that financial data are highly non-stationary, and thus financial prediction is considered a challenging task. Therefore, in this research, two methods have been used in order to reduce the non-stationary of the data, as well as using a exploring different learning methods to address the problem.

1.5 Structure of the Thesis

The rest of this thesis has been structured as follows:

Chapter 2 presents the literature review of the neural network models and algorithms. Chapter 3 explains the experimental design and the processing of the data-sets. The proposed FL-SMIA model architecture and learning method are detailed in chapter 4. Chapter 5 includes the experimental results of the proposed model (FL-SMIA) and the comparison with popular models such as FLNN, MLP, and others. In chapter 6, extensions of the FL-SMIA to deeper and mixed models are explained (D-FL-SMIA, MD-FL-SMIA, and M-FL-SMIA). The last proposed model (the FL-SMIA-RBM model) is explained in chapter 7. Chapter 8 contains the experimental results for all models used in this research, as well as the comparison results between the models. In chapter 9, additional evaluation metrics and tests have been presented and discussed. Chapter 10 provides the results of the proposed models (FL-SMIA and M-FL-SMIA) and currents popular models (MLP and FLNN) using an alternative method for more realistic evaluation of financial prediction. Furthermore, comparisons between the models and additional tests are presented and discussed. Finally, chapter 11 presents the conclusions and directions for future work.

Chapter 2

Literature Review

In this chapter, an introduction of the general time series and financial time series have been presented. Also, the aims of analysing the time series have been explained. In addition to that, the definition of financial time series and prediction have been represented in order to show the difficulties of making the prediction in the financial market. Moreover, a literature review of the artificial neural networks and different models of neural network models are represented.

2.1 Neural Networks and Applications

Over the past two decades, significant researchin the field of Artificial Neural Networks (ANN) has led to improvements in applications related to most domains in our life [148, 113, 99, 39, 108]. In general, Artificial Neural Networks are considered as powerful machine learning tools that are used for many tasks including prediction, classification, and clustering [123, 5, 45]. A large number of ANN applications have been proposed, which are aiming to produce more accurate results in financial prediction as well as for other domains. In the financial markets, ANNs are considered a valuable forecasting tool because of their ability to learn and generalise as well as their nonlinear behaviour properties [105, 28, 118, 121].

Due to the inherent noise in patterns of financial series data and nonlinear components, statistical models often lack in the prediction of financial time series compared to the ANN models [32]. Artificial neural networks have been widely used in finance, e.g. in forecasting exchange rates [92, 10, 103], predicting stock values [109], portfolio management [16], credit rating and predicting bankruptcy [87] and inflation and cash forecasting [1].

There are many research works confirming that artificial neural networks are efficient in forecasting financial time series data, e.g. in [140]. Another example of using ANNs is [85], where the authors propose ANN models for recognising patterns when auditing monthly balances in financial accounts and then tests the predictive ability of the proposed ANN models. They used monthly balances data as a time-series, where the target is to recognise the dynamics and the relationships between different accounts. The test results indicate that neural networks are promising tools for recognising the dynamics and the relationships between financial accounts.

In [104] the authors proposed an improvement of FLANN model for one-day prediction, as well as a long term prediction using the stock price of leading stock market indices. The weights of the proposed model have been trained using the least mean square (LMS) and the recursive least square (RLS) algorithms in different experiments. The authors compared the results provided by both the methods and concluded that the application of the FLANN model for the stock market prediction gives results which are comparable to other neural network models. Also, the authors indicated that the RLS-based FLANN model more suitable for online prediction.

McDonald, et. al. [106], used number of financial time series for one day ahead prediction. The researchers studied the effectiveness of using a number of machine learning algorithms and combinations of these algorithms. The investigation results showed that hybrid models which are consisting of a linear statistical model and a non-linear machine learning algorithm, are effective at forecasting the future direction of financial data.

Another approach has been proposed by Li and Sun [93]. The authors constructed a new ensemble method of Case-based reasoning (CBR) that (principal component CBR ensemble (PC-CBR-E)in order to improve the predictive ability of CBR in business failure prediction (BFP) by integrating the feature selection methods in the representation level, a hybrid of principal component analysis with its two classical CBR algorithms at the modeling level and then it weighted majority voting at the ensemble level. The results indicated that PC-CBR-E produced superior predictive performance in Chinese short-term and medium-term BFP.

Basak et al. [9], proposed an ensemble of Decision Trees to improve the accuracy of stock price prediction. The authors used Random Forests and XGBoosted trees in an attempt to speed up the process of growing Gradient Boosted Decision Trees (GBDT). The forecasting problem has been treated as a classification problem, where the classes are an increase or a decrease in the price of a stock with respect to n days back.

2.2 Time Series Analysis and Forecasting

Time series forecasting methods have been applied in various industries and trades, such as energy, electricity, medicine, food, and other industries, in order to recognise trends and make prediction about the future sales. Time series analysis and prediction are generally recognised as an important task. There are many types of time series, which have been used in research in a different domain. In general, the simplest definition for a time series according to Chatfield [22] can be given as:

"A time series is a set of observations measured sequentially through time".

Financial time series such as stock prices, exchange rates, and other financial information are considered as sensitive to changes in economic, political and other social fields [95]. However, in this work external factors are not explicitly modelled.

2.2.1 Financial Time Series and Prediction

Financial time series has attracted high interest in an economic domain, and it realised as one of the important applications for intelligent processing. Financial time series analysis has been defined as [137]: "Financial time series analysis is concerned with the theory and practice of asset valuation over time".

Depending on the activity in the market, financial time series have various time scales (hourly, daily, monthly, seasonally, and other time scales). Examples of financial time series can be found in stock prices, oil prices, exchange rates, and other time series,which related to the economic domain. According to[22], "Time series forecasting is the process of predicting future values using current value".

Financial time series are generally viewed as inherently noisy and non-stationary, [14], [139], and [138]. These properties of financial time series data give the reason of why financial prediction is considered a challenging task.

The non-stationarity and noisiness and the difficulty in predicting financial time series is generally attributed to the principle of market efficiency. It refers to the level with which current prices reflect all relevant information about the actual value of the underlying assets. That means if the markets are efficient then all the relevant information is already integrated into prices, thus there is no way to "beat" the market as there are no undervalued or overvalued securities available. In 1970, this principle has been formulated by Eugene Fama [42]: "A market in which prices always «fully reflect» all available information is called «efficient»". The efficient market hypothesis (EMH) states that capital markets are efficient and Fama concludes that the EMH is supported by most empirical studies. Therefore above-market returns can only be obtained with earlier or exclusive access to relevant information.

Fama [42] discusses three forms of tests for market efficiency:

1. **Weak-form tests:** This form refers to predicting future returns by using past returns. The assumption of this form indicated that in case that no investor can earn excess returns by improving trading rules based on historical price or return information. This assumption leads to the "random walk theory", which states that market prices changes follow a random path up and down, without any influence by historical price movements [38, 145, 44]. In other words, the past movement of market prices or the stock prices cannot be used to predict its future movement. However, excess returns are still possible using fundamental analysis under weak-form market efficiency [97, 133].

2. **Semi-strong form (Public information):** This type assumes that a market considered semi-strong efficient if no investor can earn excess returns from trading rules which depend on any publicly available information such as historical data, financial statements of companies, reports in the financial press, and annual reports [79, 125]. That means using technical analysis and fundamental analysis would not achieve good returns, the reason behind that is using these analyses results in gained information which is already available and incorporated into current prices. Consequently, the only information that is at least temporarily unavailable to the market will be useful to gain an advantage in trading.

3. **Strong form (Private information):**

   This form assumes that if no investor could use any information (publicly available or private information) can earn excess returns [47, 29, 60]. It indicates that market prices reflect allinformation both public and private. Fama [42] himself admits that this assumption is not strictly valid and that there are documented examples of excess returns achieved through monopolistic access to information.

Fama later revisited [43] these categories and included factors like interest rates and dividend yields in the first category, which he renamed 'tests for return predictability'. He renamed the other two categories 'event studies' and 'tests for private information'.

Not all investors and academics have followed the EMH, and pointed to the fact that successful active traders do exist [89]. The random walk theory has been refuted by [97, 98], who showed in their initial paper that there is some predictability in share price series, specifically negative and positive auto-correlation between the weekly index and share returns, respectively.

In order to make the EMH testable, it is necessary to assign a pricing model to risk, for which there are multiple options. Other factors such as market microstructure or – in empirical tests – over-use of data (snooping) can lead to relevant effects that make it seem highly unlikely that the EMH is strictly true in the sense that no price prediction is possible.

Regardless of the extent to which the EMH is demonstrably true or false, financial markets are adversarial in that all participants want to take advantage of all the information they have. This makes it generally difficult for investors to achieve above average returns, but using better analysis than competitors, e.g. by using novel neural network models which could be still lead to an improved price prediction and thus increased profits.

2.2.2 Statistical Models

Financial time series prediction has been traditionally addressed by statistical models and more recently by neural networks, as detailed in the following subsection.

Financial time series have traditionally been modelled by various statistical methods [35][37]

1. **Auto Regressive (AR) Model**

   The auto-regressive method is based on the simple notion, that many time series exhibit a high correlation between new values and some previous values at a given lag. Typically, an auto-regressive (AR) model of order n consists of the past values in the data series, which are used to forecast the next value (A. C. Knowles, 2005). Autoregression represents another example of linear models.

   The AR predictor is defined by:

   $$AR_t = Y_t^* = \sum_{i=1}^{n} R_i Y_{t-i}, \tag{2.1}$$

   where Y_t is the actual value (input) at time t, AR_t and Y^* respectively is the prediction value for a next time period, R_i are regression coefficients.

2. **Moving Average(MA) Model**

   One example of linear models is the Moving Average, which is used widely in financial markets since it is recognised as quick, inexpensive and effective. In this technique, the average value for a set of previous values in the time series is calculated. This average value is used to predict the next time period. In other words, the average of n past values in the time series are used as the basis to forecast the next time period [37], [35].

The moving average is calculated as following:

$$MA_t = Y^*_{t+1} = \frac{(Y_t + Y_{t-1} + Y_{t-2} + \ldots + Y_{t-n+1})}{n} \tag{2.2}$$

where MA_t refers to the moving average at time t, Y^*_{t+1} is the forecasting value for the next time period and n is the number of terms in the moving average.

3. **Auto Regressive Moving Average(ARMA) Model**

   The ARMA model is the integration between the autoregressive model and the moving average model. This model is used for forecasting the next time for financial time series. The equation below determines the ARMA prediction at time t:

$$ARMA_t = Y^*_{t+1} = \sum_{i=1}^{n} R_i Y_{t-i} + \sum_{i=t-n+1}^{q} W_j Q_{t-j} \tag{2.3}$$

   where $ARMA_t$ is the prediction for the next time period, W_j refers to weights that are applied to Q_{t-j}, the previous values of the residuals.

   ARMA models are limited to forecasting stationary signals in time series. Therefore, non-stationary data should be transformed into a stationary time series before it can be predicted using an ARMA model [37].

2.3 Artificial Neural Networks (ANNs)

Artificial Neural Networks (ANNs) are devices that process information (data) in a distributed manner inspired by neurons in the biological nervous system. According to Haykin[63]:

"A neural network is a massively parallel distributed processor that has a natural propensity for storing experiential knowledge and making it available for use".

In general, the structure of neural network models consists of a number of units (artificial neurons), connections, and transfer functions. Neural network applications are divided into categories such as prediction, classification, data association, clustering, pattern completion, and optimisation. Artificial neural networks are capable of solving complex problems, which cannot be solved by traditional computing methods.

2.3.1 Biological Neurons

Biological neurons, e.g. of the human brain, are a specific type of cell. This type of cell has the ability to communicate with up to 200,000 of other neurons [62]. The human mind has a superior ability to solve very complex problems because of the existence of groups of neurons and the strength of their multiple connections.

Biologically, as shown in Figure [1] 2.1, the four basic components of a simple neuron include:

1. **Dendrites**: it collect signals to send it to the cell-body.
2. **Cell-body**: it is responsible for integrates incoming signals and generating the leaving signal then send it to the axon.
3. **Axon**: it receives the signals from cell-body and passes it to the dendrites of another cell
4. **Synapses**: are specialised junctions at which a neural cell communicates with a target cell.

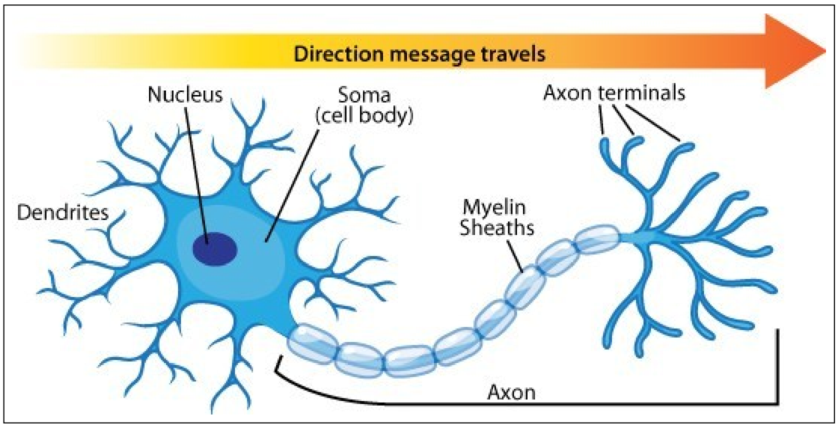


Fig. 2.1 The structure of biological neuron.

An overarching goal of researchers is to understand the ability of human neurons and analyse the behaviour of these neurons in solving complex problems that have not been solved using conventional computing methods. Although most artificial neural networks are only loosely based on biological neural networks, the aim of understanding the behaviour of these neurons is to produce models in the field of artificial neural networks that are nowadays applied in many domains of our lives.

[1]As illustrated by Jason Roell in: https://www.jasonroell.com/2017/06/12/from-fiction-to-reality-a-beginners-guide-to-artificial-neural-networks/

Many types of research have proved that the researchers continuing on improving the methods as well as the models of different neural networks to achieve a higher degree of accuracy and reliability in finding solutions to many complex problems so as to meet the requirements of current and future life [102, 126, 2].

2.3.2 Artificial Neurons (Neuron Model)

Artificial neural networks consist of a number of basic units, it is often called "neuron", "node", "unit" or "processing element". The artificial neuron is designed with a structure similar to the biological neuron in terms of functional performance. As shown in Figure[2] 2.2, an artificial neuron receives several inputs (x_0, x_1,:., x_n) to produces a single output. Each input (data) are associated with different weights ($w_0, w_1, ..., w_z$)

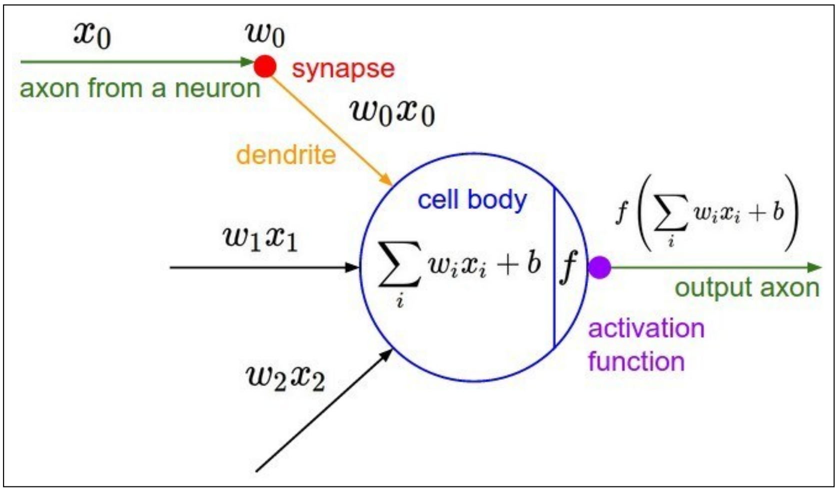


Fig. 2.2 The neuron model (Perceptron)

which are real numbers. In the first ANNs, the neuron's output was either 0 or 1, as it is determined by whether the weighted sum $\sum x_i w_j$ is less than or greater than some threshold value. The threshold it is a real number as it represents a parameter of the neuron.

The input values (x_i) are multiplied with their associated weights (w_j), the product is fed into the summing unit (cell body), which sums the result values ($\sum x_i w_j$). Then the output of the summing unit is passed to a transfer function (often a sigmoid) and turned to an output value.

[2] As by Andrey Karpathy in: https://http://cs231n.github.io/neural-networks-1/

2.3.3 Transfer Functions

There are various non-linear transfer functions, which can be used in neural networks. The idea behind using an activation function in neural networks is to roughly model the way neurons communicate in the brain with each other each one and mathematically, to introduce a non-linearity.

In this research, three types of transfer functions have been used:

1. **The Logistic Sigmoid Function** The sigmoid which is the most popular transfer function. It is called sigmoid because of its 'S' shaped curve as shown in Figure 2.3. The output of the logistic sigmoid function is in the interval [0, 1]. The logistic sigmoid function is calculated as follows:

$$f_{sig}(x) = \frac{1}{1 + exp^{(-x)}} \tag{2.4}$$

where x is the weighted sum of the neuron inputs. The logistic sigmoid function is often used for the hidden nodes in many neural networks such as the multi-layer network [13],[61].

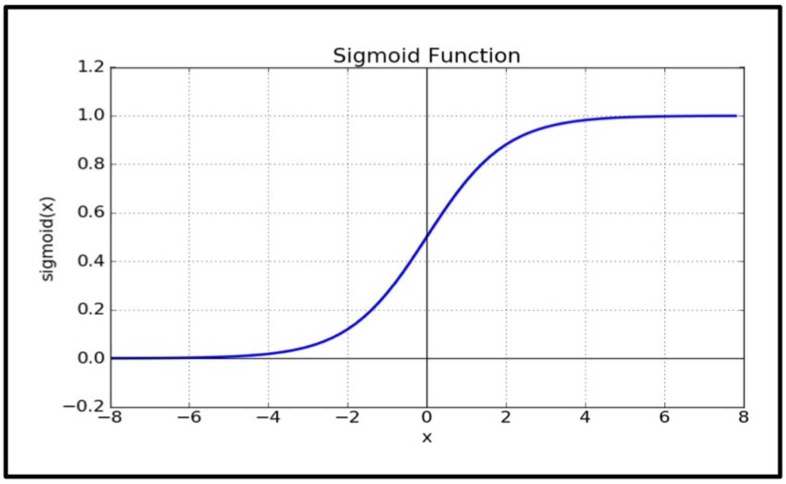


Fig. 2.3 Logistic sigmoid transfer function.

2. **The Hyperbolic Tangent Function** The second type of transfer function used in this work is the hyperbolic tangent function which is another sigmoidal function. This function is a good trade-off for neural networks, where speed is more important than the exact shape of the transfer function [136, 34].

The output value of the hyperbolic tangent transfer function is ranges between -1 and +1, as shown in Figure 2.4.

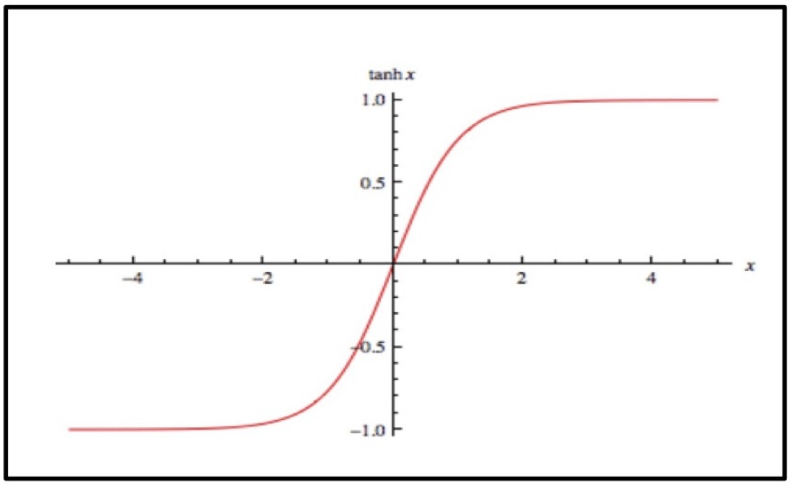


Fig. 2.4 Hyperbolic tangent transfer function.

The hyperbolic tangent transfer function is calculated as follows:

$$f_{hts}(x) = \frac{2}{1 + exp^{(-2x)}} - 1 \tag{2.5}$$

where x is the weighted sum of neuron inputs.

It is worth to note that the hyperbolic tangent transfer function is often recommended for hidden layers to being more balanced. As shown in the figures above, the value 0 for the hyperbolic tangent function appears for input 0 at the greatest gradient, while for logistic sigmoid function 0 is the infimum [147].

3. **The Rectified Linear Function**

   The third type of transfer function which have been used in this research is the Rectified Linear function used in Rectified Linear Units (ReLU).

As shown in Figure 2.5, its a function for x greater than 0. Training of neural networks is considered to be faster with ReLU function, so that it recommended to use with hidden units [86]. The Rectified Linear Units transfer function is calculated as follows:

$$f_{ReLU}(x) = Max(0, x) \tag{2.6}$$

$$f_{ReLU}(x) = \begin{cases} x & \text{for} x \geq 0 \\ 0 & \text{otherwise,} \end{cases} \tag{2.7}$$

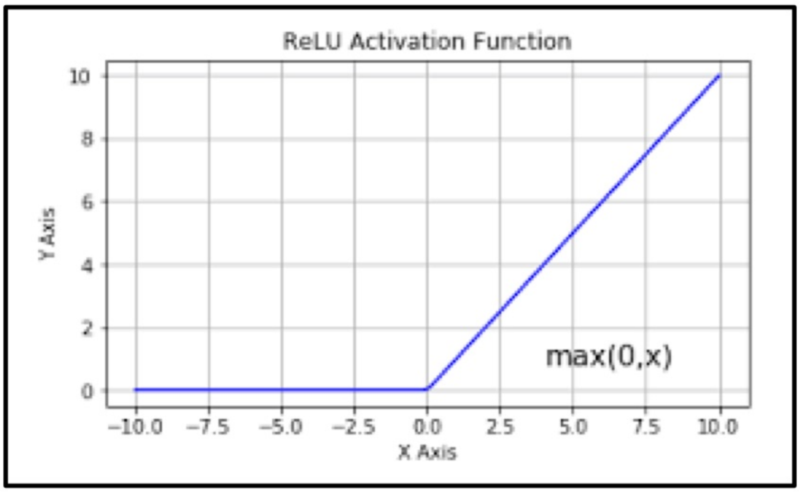


Fig. 2.5 Rectified Linear Units (ReLU) function

where if the input is less or equal to 0, the output is 0. Otherwise, the output is equal to input.

2.3.4 Neural Network Learning

In artificial neural networks there are different approaches that have been used to train the networks such as Supervised Learning, Reinforcement Learning, and Unsupervised Learning. The choice of learning technique depends on the goal of the research or application. In this research Supervised and Unsupervised Learning methods have been used. A summary of learning methods [3] is as follows:

A. **Supervised Learning**

In general, Neural networks are organised into layers, which are input layer, hidden layer/layers and an output layer. The data are fed into the network through the input layer, where the data includes inputs and target output (desired output) associated with each input sample in order to be reproduces by the network. As shown in figure 2.6, in Supervised Learning the network output is compared with the target output, the difference is the error. The network then will be tuned by learning parameters (weight connections) that reduce the error until it can model the training data as accurate as possible.

B. **Unsupervised Learning**

[3] The two diagrams (Supervised Learning, Unsupervised Learning) are inspired by the this video https://www.youtube.com/watch?v=edM0_C4j6R0

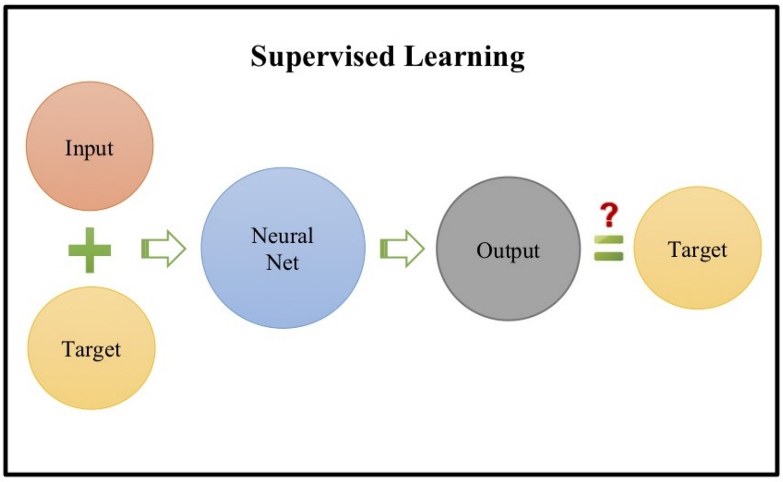


Fig. 2.6 Supervised Learning

Unsupervised Learning is a procedure that is used when there are no output targets associated with the input samples as in figure 2.7. In this case the network receives only input data. That means there are no output targets for comparing with the network output. The objective of using an unsupervised learning algorithm is to build new representations of the data depending on data patterns.

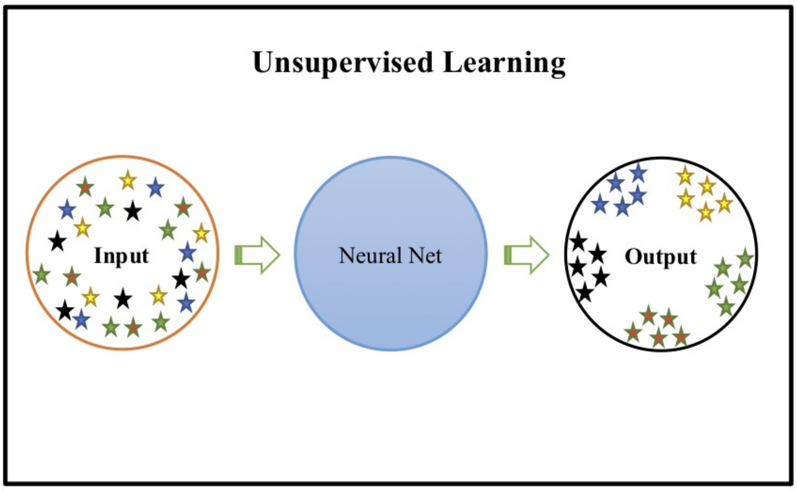


Fig. 2.7 Unsupervised Learning

In this research two techniques of unsupervised learning have been used which are the Immune Algorithm and the Restricted Boltzmann Machine (RBM) in addition to the supervised learning (the back-propagation algorithm).

2.4 Models of Neural Networks

There are different types of neural networks; the simplest structure of an artificial neural network consists of number of units grouped in layers and the layers are connected to each other. As the units can be connected in different ways, neural network models have various forms of interconnection between network layers [46, 153]. The most commonly used neural network architectures are feed forward networks (MLP, HONN) and recurrent networks. There are also other architectures, for instance, self-organizing feature map [63, 83].

Neural networks are typically organised into layers, which are input layer, hidden layer/layers and an output layer, as detailed below. When there are only connections between consecutive layers in the same direction, then the networks is called feed forward neural networks [13, 63]. The input signals are passed to the network in forward direction from the input layer to the hidden layer and then to the output layer. These neural networks can include any number of neurons per layer, as well as it can have any number of hidden layers. Although the main focus of machine learning research has recently has been on deep, recurrent or convolutional neural networks operating on raw data, methods for constructing features and alternative learning algorithms have still potential for improving predictive performance. We focus here on Functional Link Neural Networks and the Immune Algorithm.

2.4.1 The Multilayer Perceptron (MLP)

The simple perceptron refers to a unit with a transfer function and a weight adaptive mechanism (learning) by comparing the actual output and the desired output responses for any input or stimulus[63]. As illustrated in Figure 2.8 the Multilayer Perceptrons (MLP) is a type of feed-forward network, which consists of a number of units grouped in layers, which are:

1. **Input layer** : which sent the input signal (data) to the other layer in a forward direction.

2. **Hidden layer/ layers** : these layers are located between the input layer and output layer, which transmit the signal (data) from the input units to the output units. Hidden layers enables the network to learn complex tasks and solve different problems.

3. **Output layer** : this layer provides the output as the actual response of the network.

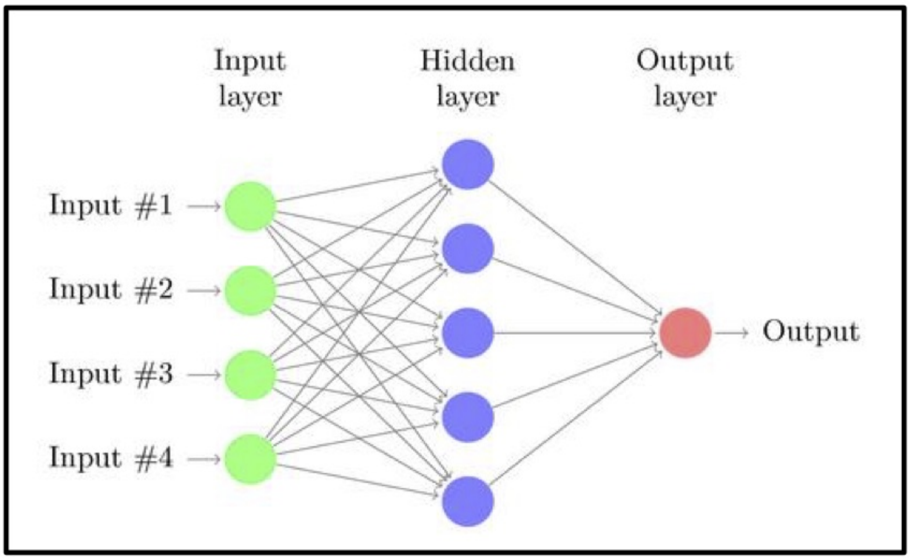


Fig. 2.8 Multilayer feed-forward neural network

In the hidden layers, each hidden unit performs two tasks: firstly, summing the weighted inputs to the hidden unit, secondly, passing the sum through a non-linear activation function. A bias is included in the network as an additional unit, which has a constant value of one. A bias term is usually treated as a weight connection. Thus, the bias term can be learned like other weight connections [40]. The operation of the MLP network can be divided into two phases:-

A. **The training phase** : the MLP network is trained for its specific purpose using training algorithms.

B. **The retrieval phase** : this phase generates output by using the previously trained MLP networks.

MLP network with one hidden layer and a sufficient number of hidden units (having the non-linear transfer function) can produce a feasible function to any desired rate of accuracy [72], MLP networks are training using the supervised algorithm.

The output of an MLP network that uses one hidden layer is calculated as follows:

$$Y_k = \sigma\left(\sum_{j=1}^{J} W_{kj}\sigma\left(\sum_{i=1}^{N} W_{ji}x_i + W_{jb}\right) + W_{kb}\right) \tag{2.8}$$

where σ is a log-sigmoid transfer function, x_i represents the input value, W_{ji} are the weights from the input layer to the hidden layer, W_{jk} are the weights from the hidden layer to the output layer, W_{jb} are the bias weights for the hidden units, and Y denotes the network output. MLP network is considered as a fully connected network since every unit is connected to all units in the next layer.

The Back-Propagation Learning Algorithm

The back-propagation algorithm was introduced in the 1970 [56], however it was not widely appreciated until 1986 when the famous paper was presented by David Rumelhart, Geoffrey Hinton, and Ronald Williams [127]. That paper includes a description of several neural networks where back-propagation works far faster than earlier approaches to learning. Thus, that paper represented the possibility of using neural networks with back-propagation algorithm to solve problems which had previously been insoluble.

For neural networks, supervised learning is an optimization procedure to minimize the error on the data. The most commonly used optimization approach for training neural network is the Gradient Descent, which is realized in the error back-propagation learning algorithm. The learning iteration (epoch) of a neural network can be summarized into two phases[40]:

1. **Feed forward pass** : in this phase, the input vector is applied to the processing units of the network following the direction from the input layer to the output layer. Thus, the actual output is produced. Consequently, all weight connections of the network are fixed through this phase.

2. **Backward propagation** : during this phase, the weight connections of the network are adjusted. Since the actual output (produced by the network) is subtracted from the target output to produce the error signal, this error propagated backward from the output layer to the input layer through the network, so that, this algorithm called " error back-propagation". The reason behind adjusting the connection weights of the network is to reduce the error between the desired output and the actual output. On other words, is to make the actual output of the network become closer to the target output.

2.4.2 The Functional Link Neural Network (FLNN)

The Functional Link Neural Network (FLNN) is a type of Higher Order Neural Network (HONN) that utilizes combination of its inputs [117]. The tensor product model is one type of FLNN where the network input is extended with products of input features. For example, with three inputs features X_1, X_2, X_3 the second order

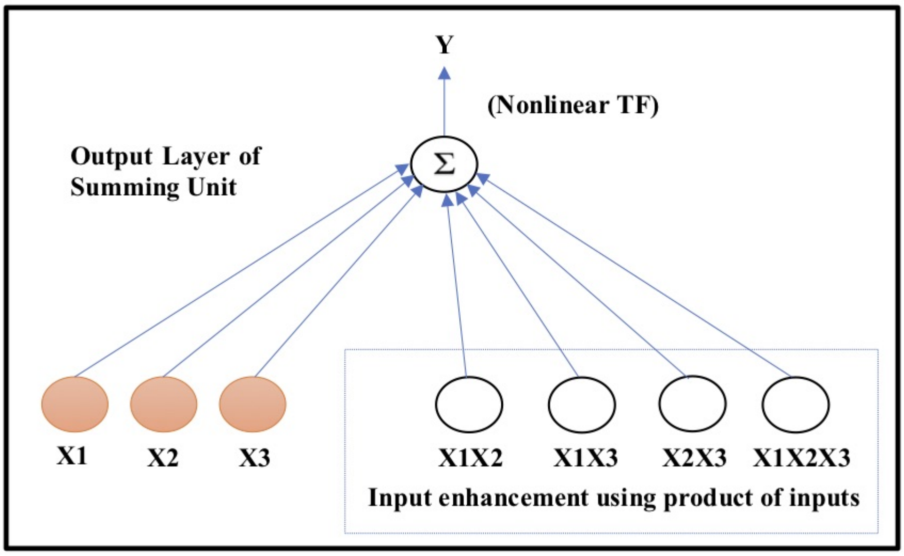


Fig. 2.9 The FLNN-tensor product model following [117].

terms X_1X_2, X_1X_3, X_2X_3 can be added to the input layer and also the third order term $X_1X_2X_3$ as shown in Fig. 2.9. This model utilises the joint activation between the input units to extend the input space without adding any external information.

The FLNN model is trained with the back-propagation method. The FLNN model contains only one connection weight matrix, which leads to faster learning for the FLNN model compared to other neural network models such as the standard Multi-layer Perceptron (MLP)[128]. The idea of using a single weight matrix in the FLNN model is to decrease the number of trainable weights, while the higher-order connections still enhance the network's performance. As a result, the FLNN model could achieve similar performance to the standard multilayer neural networks which are considered a suitable model for solving complex problems [51]. The FLNN model has been used to solve a number of problems, such as classification and regression [115], predicting maintainability [88], identification and prediction [96], and again prediction [58, 100].

Although the architecture of FLNN is simple, the higher order-inputs and the non-linearity leads a network with greater capacity compared to a linear models or models using only input features directly as shown in [53] and [107]. The FLNN architecture can suffer from combinatorial explosion, due to an exponential increase in the number of inputs units. Therefore, only second or third order networks are typically used in practice [74],[142]. The output calculation of the FLNN is as follows:

$$Y = \sigma \left(W_0 + \sum_j W_j X_j + \sum_{i,j} W_{j,k} X_j X_k + \sum_{j,k,l} W_{jkl} X_j X_k X l + .\therefore \right) \tag{2.9}$$

where σ is a non-linear transfer function, W_0 is the threshold, X_j is the $j^t h$ component of input vector X, and W_{jkl} is trainable weight. The FLNN have been successfully used for solving different problems, such as prediction pattern recognition, optimal control and other[52].

The principle of the tensor model has been used in this research to add extra inputs to the proposed network.

2.5 Over-fitting in Machine Learning

Generalisation refers to the ability of a machine learning model to perform well on new data similar to how it performed with training data. The poor generalisation of models is often due to over-fitting [76, 21]. Over-fitting in machine learning refers to the model performing well on training data by learning the details, which are often noise, in the training data, such that the performance of the model is negatively impacted on the new data (testing data) [122]. Over-fitting normally happens because of an excessively complex and flexible model [11]

There is a number of investigations that have studied possible methods to reduce over-fitting to improve the generalisation ability of the neural networks. A basic technique can be used, which is to keep as held-out dataset, or a more sophisticated cross-validation scheme [111, 130] to avoid and measure over-fitting. A validation data-set is a sample of training data that is held from machine during learning to control the learning process. After a model has learned by using the training data-set, the validation data-set is used to evaluate the performance of the model on unseen data. This approach can be used to tune methods to reduce over-fitting such as the regularisation [13, 77].

In [149] the researchers propose different techniques to improve the generalisation of a neural feed-forward neural network. The authors used a self-organised hidden layer and an immune algorithm (SONIA) to predict food quality. The prediction results showed an improvement on generalisation of the proposed SONIA network compared to an MLP network. Another method has been proposed in [119] the authors used a heuristic algorithms approach to prevent the over-fitting problem, as they used an ensemble of classifiers instead of a single classifier. The researchers showed through the results that synchronous minimisation of training error and the ensemble size in the training phase could be help on reducing the amount of the over-fitting significantly.

In [135, 155, 129] the Drop-Out method has been used to address over-fitting. The drop-out method is a regularisation technique that aims at reducing the complexity of the model in order to prevent over-fitting. The "drop-Out" refers to dropping out units including both input and hidden units with their connections from the network or model, but It is not used on the output layer [129]. The drop-Out method can be used during the training phase on supervised learning in a way that some number of layer outputs are randomly ignored or “dropped out”. The drop-out method affects the trained layer in such a way that making it like a layer that is using a different number of units for connectivity to the prior layer. Using the drop-out method result in reducing the over-fitting as well, often it improves the performance of the network comparing to other other regularisation methods. As in [135], researchers indicated that using the drop-out method improves the performance of neural networks on many tasks including document classification, speech recognition, vision, and computational biology.

2.6 Regularisation

Regularisation is one of several techniques which have been used to avoid over-fitting in neural network training [77]. The main objective of neural network training is to build a statistical model of the process underlying the data represented by the weights resulting from the training process. Neural network training does not aim to learn the accurate representation of training data itself. However, the aim is to represent good generalisation to make successful prediction for new input data [13, 61]. Many researchers proved that using regularisation can result in an improvement of the generalisation capability of the networks [46, 101]. In most cases, weight decay, the simplest form of regularisation, has been used.

Regularisation is the technique of adding a penalty term Ω to the error function which can help obtaining a smoother network mappings. It is given by:

$$A^* = E + \lambda\Omega \tag{2.10}$$

where E represents one of the standard error functions such as the sum-of-squares error and the parameter λ controls the strength with which the penalty term Ω can influence the form of the solution. The network training should minimise the total error function A^*. Using this method requires the derivatives of Ω with respect to the strength weights of the network to be determined and the total error can be computed efficiently.

In this research, one form of regularisation technique has been used, which is the Weight decay (L2 loss) [66]. This form based on the sum of the squares of the adaptive parameter in the network. It is calculated as:

$$\Omega = \frac{\lambda}{2} * \sum(W_i^2) \tag{2.11}$$

The popularity of weight decay approach is due to the simplicity of using this method. The idea is that every weight once updated is simply decayed or shrunk as follows:

$$W^{new} = W^{old}(1 - \lambda), 0 \leqslant \lambda \leqslant 1 \tag{2.12}$$

In this case, there are potentially two groups of weights: the first one represents the weights which are not needed for decreasing the error function since this weight values are reduced gradually until they have small values, then they can even be eliminated altogether. The second group of weights that are not decayed because they are required to solve the problem. Thus, using weight decay can result in achieving a good balance between the prediction error and the penalty term in equation 2.11 [139]. The final equation of changing the connection weights is determined according to the following equation:

$$\Delta W_{(i/Wd)} = \Delta W_i - \eta\lambda W_i, Where(i = 1, 2, \therefore, N) \tag{2.13}$$

where $\Delta W_{(i/Wd)}$ is the new updated weights using weight decay term, η is the learning rate, λ is the decay rate, and w_i is the connection weight.

2.7 The Immune Algorithm

Artificial Immune Systems (AIS) have been inspired by the natural immune systems, based on ideas and concepts that originated from immunology. The idea of the Immune Algorithm is based on the behaviour of the antigens and B cells in biological immune systems as initially discussed in [143]. In the immune system, there are recognition balls and antigens. A recognition ball includes a B-cell, a single epitope and many paratopes, the epitope is attached to the B-cell and paratopes are attached to the antigen. [131]. The units that represent B cells, have features such as a certain level of stimulation that is required for a response or a mutation of the B cells which is causing the response to new patterns. These features give an AIS the ability to cluster input patterns in the training data, where the behaviour depends on the parameters in the learning and application processes such as sensitivity to stimulation and mutation behaviour [131]. Immune systems models have been widely applied in various fields. Examples include computer security, function optimisation, pattern recognition, image interpretation, process monitoring, control engineering, data mining [132].

The immune algorithm has some attractions which result in the use of this concept on several kinds of research. An example of using the self-organisation of a neural network model named LVQ (Learning Vector Quantization), that was introduced by Kohonen [84] in 1986. The idea of LVQ is that the input vectors can be combined in several fixed codebook vectors using a supervised learning algorithm in order to make relations between these codebook vectors and pattern categories. Another approach of using the self-organization of neural networks has been introduced by Carpenter and Grossberg in [18] [19], where the authors presented the Adaptive Resonance Theory (ART). ART adaptively clustering the input vectors adaptive, based on a vigilance parameter, such that categories are produced by using an unsupervised learning algorithm.

In 2005, Widyanto et al. [149] introduced the Self-Organised network Inspired by the immune algorithm (SONIA) for the forecasting of sinusoidal signals and time-temperature based quality food data. The simulation results for SONIA in the prediction of sinusoidal signals showed a significant improvement in the approximation error in comparison to the back-propagation network and showed an improvement in the recognition capability for the prediction of time and temperature when using food quality data. Later, this approach has been adapted and applied for financial data prediction [101] and is extended with product terms inputs and other architectural features such as creating the hidden units in this research instead of the clustering method. furthermore, the proposed models in this research are not dealing with the global self-organisation but are just have been used it as a local feature.

Chapter 3

Methodology and Experimental Design

The intention of this chapter is to provide information about the financial time series (data-sets or financial data) that have been used in this research, and the methods of processing financial data, which has been used to make the financial data more symmetrical and closer to a normal distribution.

The rest of this chapter includes; the learning parameters and various financial and statistical metrics to evaluate the performance of the networks.

3.1 Research Approach

The research in this project is computational and data-driven empirical. We develop models, i.e. hypotheses of the processes that determine the prices we are trying to predict. We use machine learning models, specifically connectionist models combined with the immune algorithm. These models carry a large amount of their information in the trainable parameters and components. This research is evaluated by

3.2 Processing Pipeline

This section describes the processing pipeline that has been used in this research. The processing can broadly be divided into three parts as outlined below:

Start with daily price data (one-dimensional).

1. Data Pre-Processing

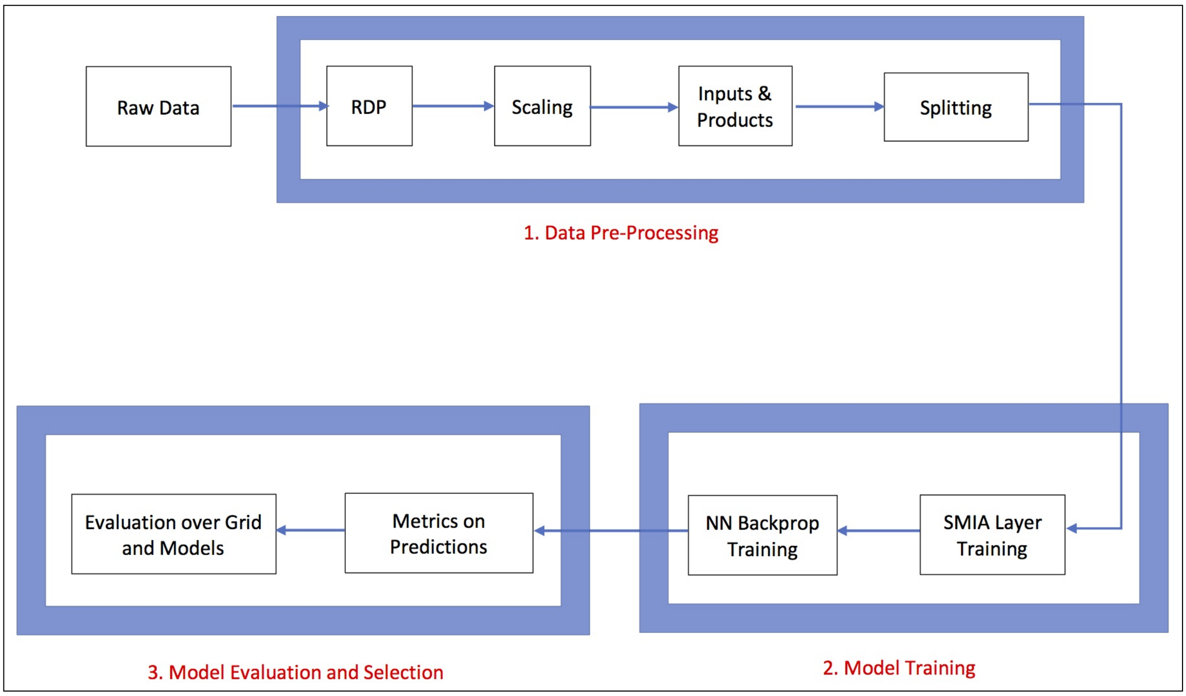


Fig. 3.1 The processing pipeline used in this study.

a. Apply the relative price differences over different time intervals (more details below).

b. Normalise the data.

c. Split the data into training, validation and testing sets.

2. Model Training

a. For the SMIA models, apply the immune algorithm to train the first hidden layer, which is named the SMIA layer (this step is independent of the back-propagation and implemented in Python).

b. Feed the result of the SMIA layer into the next layer in the network. Apply error back-propagation (implemented in TensorFlow). Train over a grid of parameter values.

3. Evaluation and Model Selection

a. Over the parameters grid, calculate financial and statistical metrics.

b. Evaluate and select models based on the metrics.

This process has been applied multiple times in several variations, to establish a suitable set of parameter values in the grid and to add more models as they have been developed.

3.3 Financial Data Used in This Research

As many conclusions have been confirmed that using ANN applications in finance is a useful approach for predicting the exchange rate [151, 4, 8], stock price [6, 80, 54, 114] and sales predictions [55, 112]. Therefore, various types of financial time series have been introduced in this research in order to be used for financial prediction.

The financial data have been used in order to evaluate the proposed FL-SMIA network and all other proposed models (FL-SMIA*, D-FL-SMIA, MD-FL-SMIA, MD-FL-SMIA-2, M-FL-SMIA, and FL-SMIA-RBM), as well as for the comparison to several other neural network architectures (MLP, FLNN, and SMIA). The time series data are available from the Federal Reserve, Board of Governors[1].

The data-sets names acronyms are shown in Table 3.1. Three types of financial time series have been used in this research: exchange rates (USD/UKP, USD/EUR, JPY/USD), stock price indices (NASDAQ, DJIA), and commodity prices (OIL and GOLD). The financial data used in this research which are daily time series covering the period from 1/07/2002 to 11/07/2017.

Table 3.1 Financial time series data-sets.

Financial data-sets	Acronym	Total	Time periods
US dollar to UK pound exchange rate	US/UK	1607	01/07/2002 - 13/11/2008
US dollar to EURO exchange rate	US/EU	1607	01/07/2002 - 13/11/2008
Japanese yen to US dollar exchange rate	JP/US	1607	01/07/2002 - 13/11/2008
NASDAQ composite stock opening price	NQO	1606	01/07/2002 - 12/11/2008
NASDAQ composite stock closing price	NQC	1606	01/07/2002 - 12/11/2008
Dow Jones Industrial average opening stock price	DJO	1605	01/07/2002 - 11/11/2008
Dow Jones Industrial average closing stock price	DJC	1605	01/07/2002 - 11/11/2008
OIL price	OIL	2744	16/08/2006 - 11/07/2017
GOLD price	GOLD	2744	14/09/2006 - 11/07/2017

More detailed information about the data that have been used in this research is listed in Table 3.1, The statistical results on daily returns for all the data-sets are listed in table 3.2. The statistical results include the mean, standard deviation, skew and kurtosis.

[1]http://economagic.com/ecb.htm/fedstl.htm

Table 3.2 Statistics of daily returns for all the time series.

Data-sets	**US/UK**	**US/EU**	**JP/US**	**NQO**	**NQC**	**DJO**	**DJC**	**OIL**	**GOLD**
Mean	-0.00039	0.01645	-0.01189	0.01620	0.01467	0.00457	0.00443	-0.05900	0.13491
Std	0.58407	0.59206	0.64051	1.44244	1.45565	1.19771	1.21405	2.30580	4.17844
Skew	-0.29462	-0.13988	-0.46352	-0.18758	0.18088	0.47989	0.49913	-0.22187	0.37782
Kurtosis	5.44871	1.27146	3.15897	6.78427	6.44277	13.05670	12.86727	3.01283	5.81020

Figures 3.2 to 3.10 provide histograms for all the data sets. The figures have been added in this chapter to give a clear reflection of the statistical results that are shown in table 3.2.

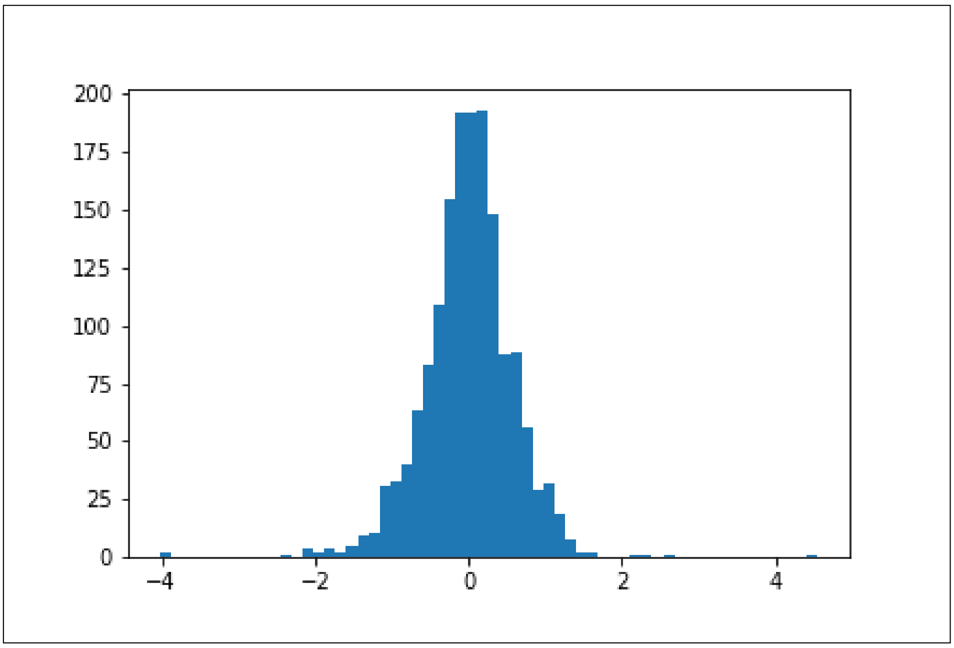


Fig. 3.2 The daily returns on US/UK data.

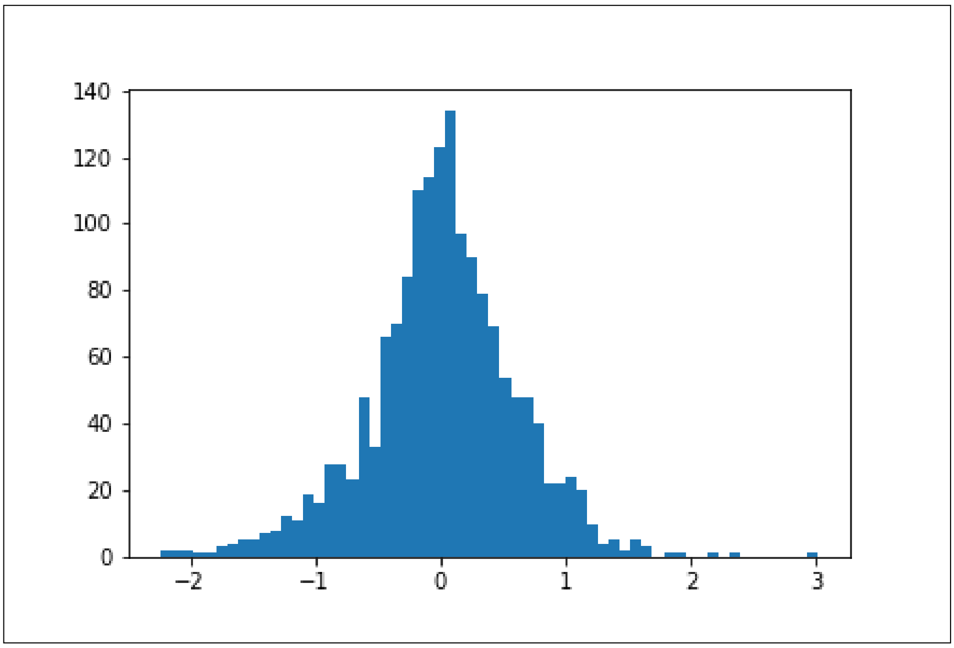


Fig. 3.3 The daily returns on US/EU data.

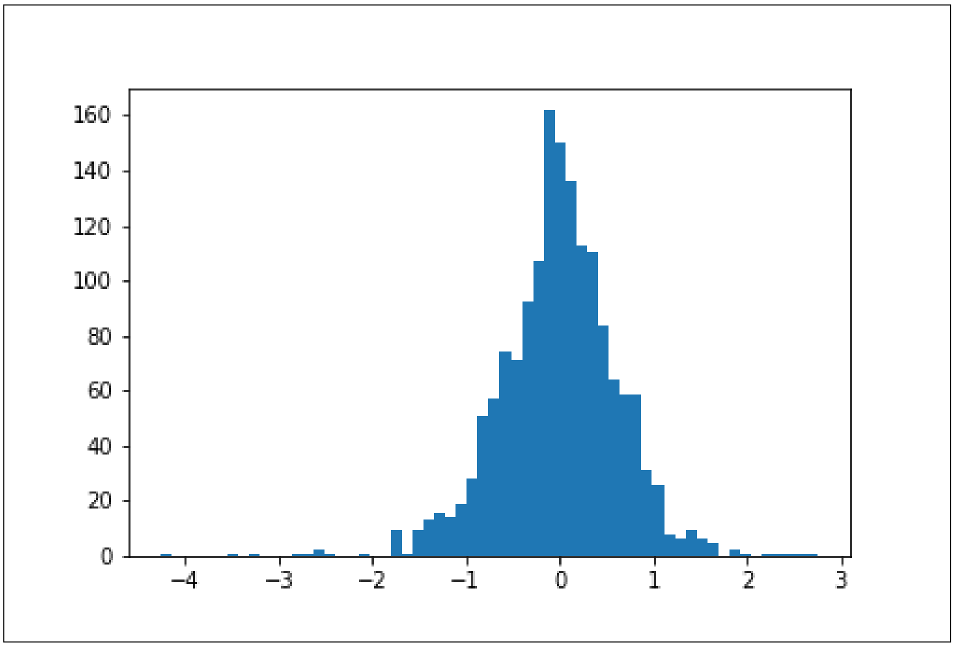


Fig. 3.4 The daily returns on JP/US data.

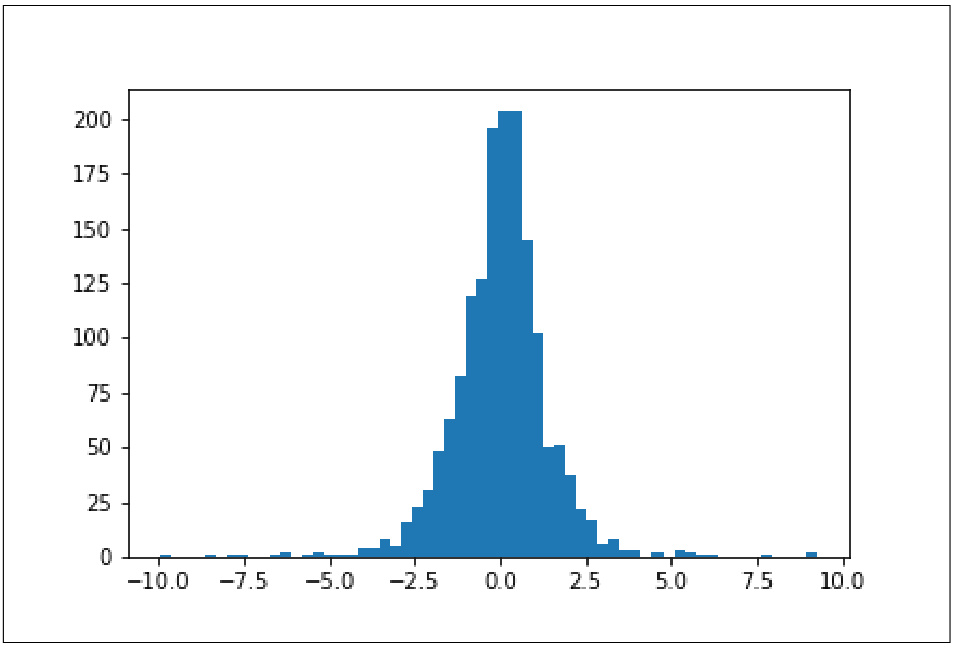


Fig. 3.5 The daily returns on NQO data.

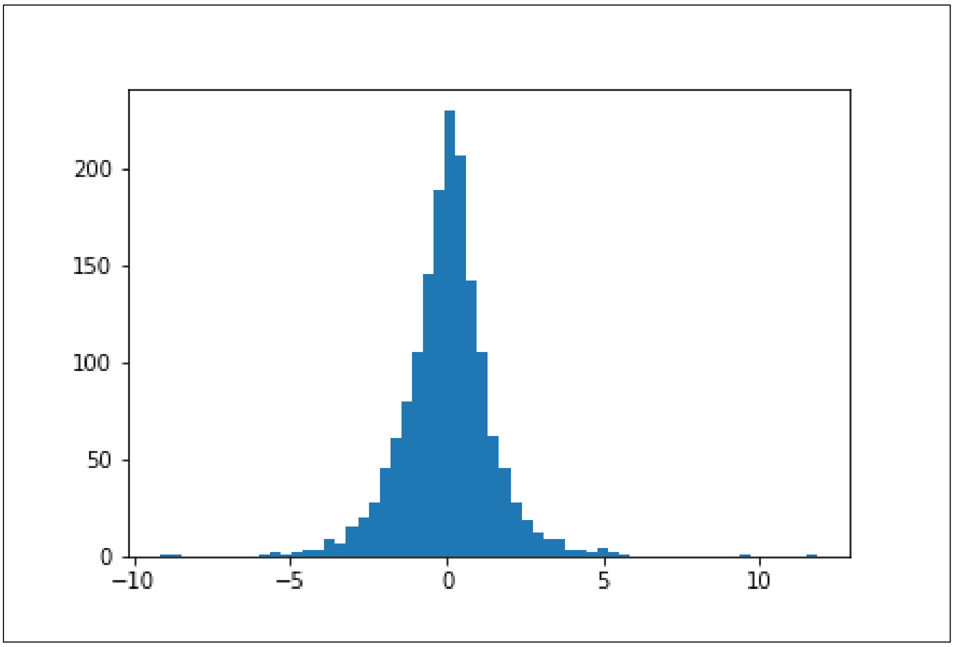


Fig. 3.6 The daily returns on NQC data.

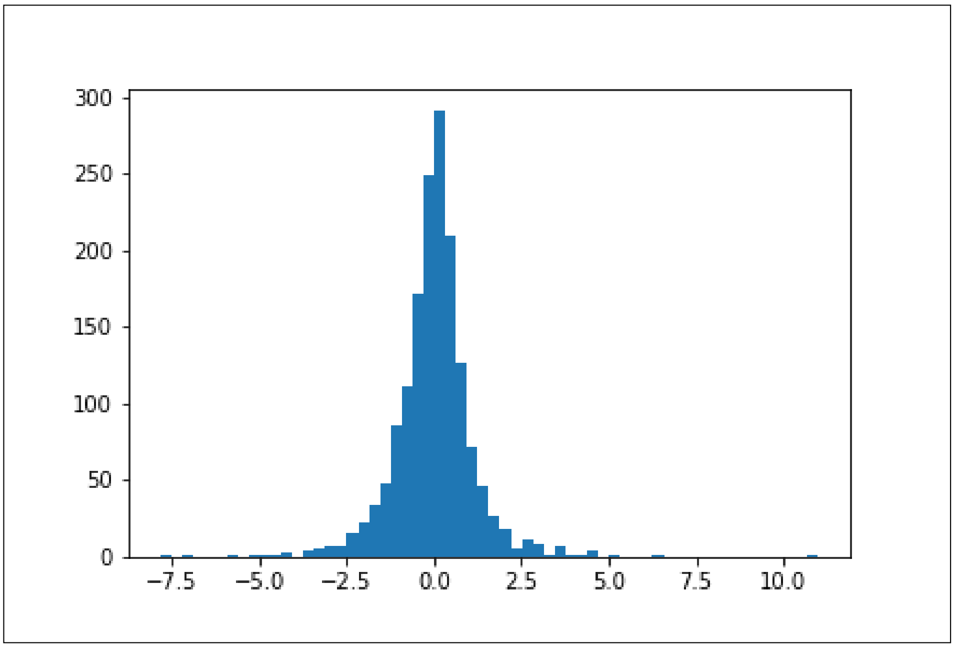


Fig. 3.7 The daily returns on DJO data.

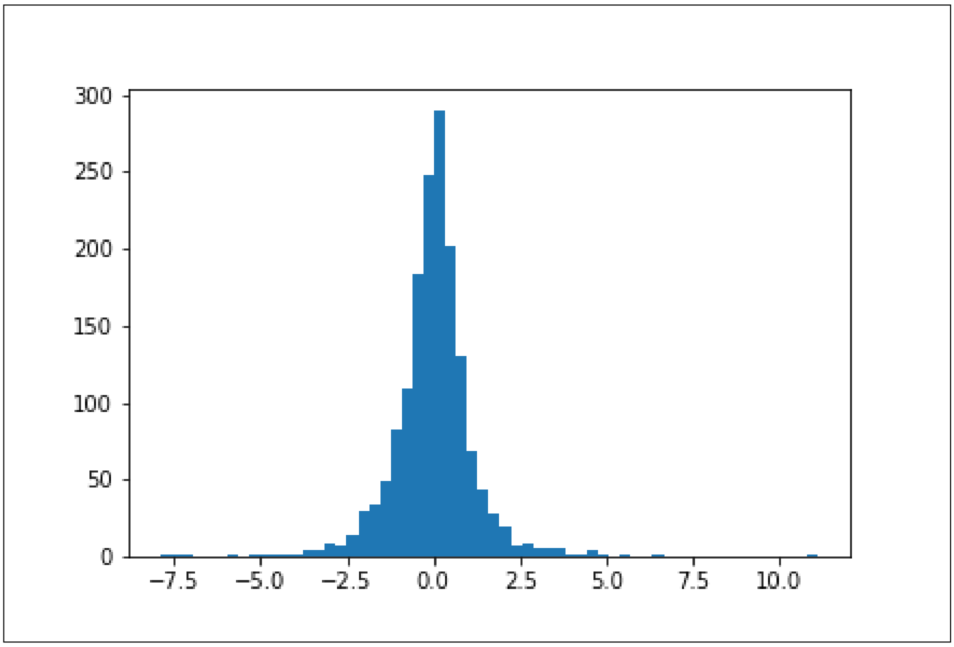


Fig. 3.8 The daily returns on DJC data.

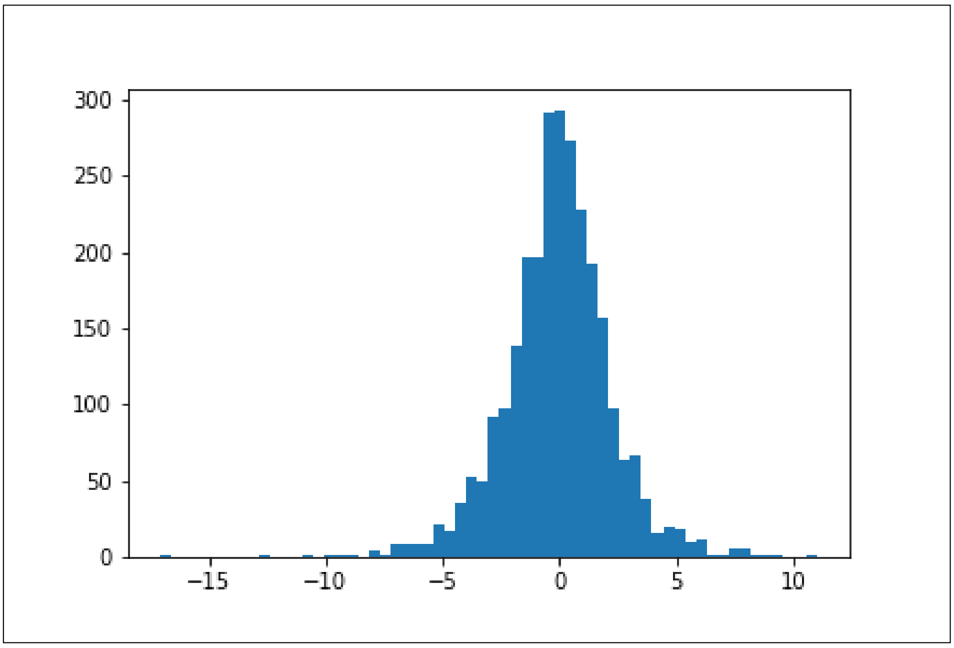


Fig. 3.9 The daily returns on OIL data.

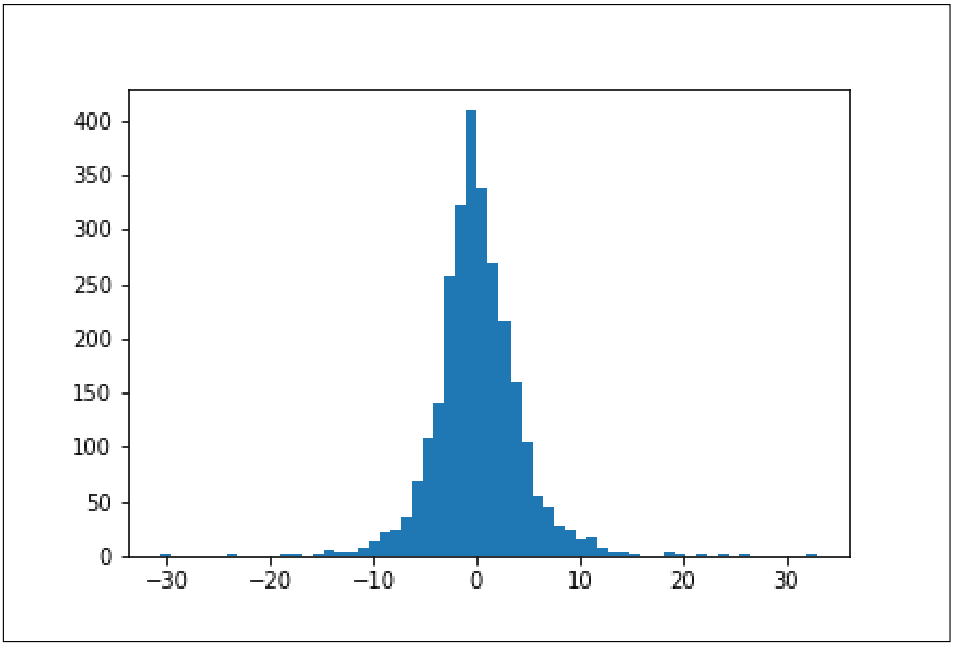


Fig. 3.10 The daily returns on GOLD data.

3.4 Pre-Processing of Financial Data

Financial time series are known as highly noisy, as well as it is non-stationary. The relative difference in the percentage of the price (RDP) has been used in this research in order to reduce the non-stationarity for the data values and transform the data to be nearly stationary data as in [137, 101, 30]. This transformation that we apply to all data-sets makes the distribution of the data more symmetrical and closer to a normal distribution.

As stated in [52] the ideal length of the moving day period should be longer than the prediction horizon. The reason behind using the EMA15 is to retain the useful information contained in the original data, which maybe will be removed by applying the RDP method. Furthermore, it has been shown that using an exponential moving average for the transformation of data (input and output data) can enhance the prediction performance for the neural networks models. The equation of exponential moving average is:

$$EMA_n(i) = \frac{\alpha^0 * P_i + \alpha^1 * P_{i-1} + \alpha^2 * P_{i-2} + \therefore + \alpha^{n-1} * P_{i-n+1}}{\alpha^0 + \alpha^1 + \alpha^2 + \therefore + \alpha^{n-1}} \tag{3.1}$$

where $EMA_n(i)$ is the n-day exponential moving average of the i^{th} day, and α is weighting factor, and P_i is the signal value of the i^{th} day.

The larger the value of α this will result in stronger the impact of older values $\alpha < 1$ In this research, the weighting factor of $\alpha = 0.8$ has been experimentally chosen to use with the exponential moving average equation for the input and output data transformation.

According to [137] the calculations for all the indicators are given in Table 3.3. The input variables consist of the EMA15 which represents the difference between a 15-day exponential moving average and the original signal, the rest of inputs include four RDP values based on five-day periods (RDP-5, RDP-10, RDP-15, and RDP-20). The forecast horizon is 1 day or 5 days. Therefore the output variable presented as a relative difference of price in percent for the next day (RDP+1) or for the five days ahead (RDP+5). As in [137], The output variable was obtained firstly by smoothing the signal with n-day exponential moving average (EMAn(i), where n is less than 5. Therefore, the data transformation can be a 3-day moving average of the data that is centred around the current 5-day values (RDP-5, RDP-10, RDP-15, and RDP-20). Then the smoothed signal presented as a relative difference in the percentage of the price for the next day (RDP+1) or the next five days ahead prediction (RDP+5). On the other hand, the original data series has been transformed and reduced by 20 trading days because the statistical information of the previous 20 trading days was used to define the input vector. The P_i is the signal value of the i^{th} day, and h is the prediction horizon of 1 day or 5 days.

Table 3.3 Input and Output Variables

Indicator	**Calculation**
Input variables	
EMA15	$P_i - EMA_{15}(i)$
RDP-5	$(P_i - P(i-5))/P(i-5) * 100$
RDP-10	$(P_i - P(i-10))/P(i-10) * 100$
RDP-15	$(P_i - P(i-15))/P(i-15) * 100$
RDP-20	$(P_i - P(i-20))/P(i-20) * 100$
Output variable	
RDP+h	$(P(i+h) - P_i)/P_i * 100$ where $P_i = EMA_3(i)$

In this research the trading costs and the interest rates have been ignored when forecasting the data sets. Therefore, the results for profits and other metrics and are not directly comparable with other results that do include these components.

3.5 Scaling Data

All the time series data used in this research has been scaled for the purpose of reducing the range differences. Therefore, all input and output variables were scaled in order to produce a new data range which is more suitable to the transfer functions. The RDP series has been scaled in this research by using the standard minimum and maximum normalisation method, as follows:

$$N_x = (Max_2 - Min_2) * \left(\frac{x - Min_1}{Max_1 - Min_1}\right) + Min_2 \tag{3.2}$$

where N_x is the normalised value of data, Min_1 and Max_1 are the minimum and maximum values in the one set of data, x refer to each value of the data-set values, while Min_2 and Max_2 are minimum and maximum of the desired data (in this research the desired range between (0.2, 0.8)have been used).

3.6 Learning Parameters

Learning parameters are defined as the parameter which can be used for network learning, these parameters can affect the performance of the network as well as it effects on the speed of the learning. So that it is important to select appropriate learning parameters such as the learning rate, the momentum, the initial weights values, and the weight decay values. It is worth noticing that using an approximation initial value for these parameters can lead to decrease the required time for the training phase [51].

In this research, the learning parameters are selected from the following ranges:

1. The range of the weights is selected as a set of random initialise weights from the range [-0.5, 0.5].

2. The values of the momentum term are selected as 0, 0.01, 0.03, 0.1, 0.3, 0.4, 0.5, 0.6 or 1.0.

3. The learning rate values which were utilised are 0.01 0.03, 0.04, 0.1 or 0.4.

4. The decay rate parameters used in the MLP, SMIA, and FL-SMIA networks are 0.0001, 0.0005, 0.001, 0.01, or 0.1.

5. The number of hidden units used with the MLP network is 2, 4, 6, 8, 12.

The combinations of the parameters above have been explored in a grid search for all networks learning that used in this research.

Searching different architectures introduces the potential issues of data snooping. However, we are not making inferences here regarding the data but only regarding the models. Therefore the application of multiple models does not constitute data snooping.

The sequential search for additional models does lead to a potential overstatement of differences between models. Therefore any statements about differences between models apply only to the data-sets used here. Further generalisation could only be applied with additional data, which is partly addressed in chapter 10.

3.7 Evaluation

In machine learning, the goal of the learning is to achieve a good performance of the trained model on new data that comes from the same process as the training data. In order to provide an unbiased estimate for the performance of a model, our models are tested using held-out test data, i.e. data that was not used in the training [7, 3]. This hold-out evaluation method has been used throughout this research.

In the hold-out method, the time series data is commonly divided into three data-sets (Training, Validation and Test set). Each data-set is used for a different task, as follows:

1. Training set: a set of examples that are used to fit the parameters of the model (e.g., the connection weights between neurons in the model) [124]. The training set includes a pair of an input and their corresponding output (target). In the training phase, a supervised learning algorithm (as explained in section 2.3.4) adjusts the parameters of the model based on the outputs of the model [23].

2. Validation set: is a sample of data used to tune other parameters (e.g. the number of hidden units in a neural network), often called hyper-parameters [146]. That means the validation set is used to find the "optimal" hyper-parameters in a selection process, e.g. a grid search, or determine a stopping point for the back-propagation algorithm [11].

3. Test set: is a set of data that uses to estimate the error rate after we have chosen the final model (hyper-parameters and actual weights). The Test set data has to be 'unseen' or 'out-of-sample' i.e. it must have not been used in the training phase or for model selection [120].

In this research, each one of the data-sets has been divided into three sets: the training, the validation, and the testing data with 50%, 25%, and 25%, respectively.

The weights of each model are optimised based on the MSE loss. For the training phase, based on preliminary experiments, the number of epochs has been chosen as 80 epochs, after initial tests with 150 epochs. The epochs number has been reduced to 80 epochs because the error levels remained fairly stable after 50 or 60 epochs. Early stopping has been used for all networks. Every network configuration in the grid search was tested in 50 simulations, to evaluate the performance depending on the random initialisation.

It is good to mention that different methods of optimisation have been used in this research such as Gradient Descent with Momentum Optimiser, RMS-Prop optimiser, and Adam optimiser. The experimental results proved that using the Gradient Descent with Momentum optimiser in this research lead to getting the best results compared with using other optimiser methods.

3.7.1 Problems in Forecasting Financial Times Series

The problems and difficulties in financial prediction are mostly related to the properties of the financial time series data. Most financial data are noisy and non-stationary. Noisiness refers to the unavailability of accurate and complete information of financial time series data from the past to the present day of prediction. Non-stationarity refers to changes in the distribution of the financial time series over time [35, 10]

3.7.2 Evaluation Metrics

In this section, the financial and statistical metrics that have been used to evaluate the performance of all neural networks used in this research will be illustrated.

Statistical metrics such as the Signal to Noise Ratio (SNR), Mean Squared Error (MSE), and the Mean Absolute Error (MAE) have been used in order to evaluate the performance of the neural networks. In addition, statistical metrics could provide useful and accurate information on tracking the signals.

In financial prediction, the main objective of using a financial criterion is to evaluate the ability of the neural network models to generate profits. Therefore, instead of emphasising on the prediction accuracy results, the network's performance focuses on trading profits.

To measure the network's financial predictability, a simple trading strategy has been devised. The trading strategy is simply to *buy* if the network forecasts a positive change and to *sell* if the network predicts a negative change for the next period. This research focusing on:

A) The Relative Profit (RP): evaluate the performance of the network by measuring the percentage of the maximal profitability that could be obtained over a given period of time [101].[2]

B) The Annualised Volatility (AV): to evaluate investment risk of data sample over one year.

The following subsections will provide an explanation of all the metrics used in this research.

[2]This metric was called *Annualised Return* in the [101] and previous publications, but it was deemed that Relative Profit is more descriptive, since the term is independent of time.

3.7.3 Financial Evaluation

1. Relative Profit (RP)

The Relative Profit (RP) measure have been used in this research to estimates the ability of neural networks on automatic trading. The RP measure the total profitability using the strategy of *buy* and *sell* signals [35]. In other words, the RP indicates the total profits that could be gained over a period of time by trading or investing. Relative Profit (RP) is calculated as follows:

A) Calculate the returns (R_i):

$$R_i = \begin{cases} +|y_i| & \text{if}(y_i)(y_i^*) \geq 0 \\ -|y_i| & \text{otherwise} \end{cases} \tag{3.3}$$

where y_i is the target output value (relative difference as defined in table 3.3) and y_i^* represent the predicted output value.

B) Find the sum of profits (cumulative return):

$$CR = \sum_{i=1}^{n}(R_i), \tag{3.4}$$

where n is the total number of data samples.

C) Calculate the annualised profit (annualised return) and annualised maximal possible profit:

$$AnnualisedProfit = \left(\frac{252}{n}\right) * CR, \tag{3.5}$$

$$AnnualisedMaxProfit = \left(\frac{252}{n}\right) * \sum_{i=1}^{n} abs(R_i), \tag{3.6}$$

where n is the total number of the data samples, and 252 is taken as the number of trading days per year.

D) Calculate the Relative Profit (RP), which is expressed as a percentage, the actual profit relative to the maximal profit over all samples:

$$RP = \left(\frac{AnnualisedProfit}{AnnualisedMaxProfit}\right) * 100 \tag{3.7}$$

2. Annualised Volatility (AV)

In order to evaluate the investment risk possibility, the Annualised Volatility (AV) measure has been used in this research to provide information related to the variability of the prices. The AV measure is used to measure the variance of returns over a period of time. The small value of volatility is preferable for financial prediction. Furthermore, in real trading, the volatility measure provides significant information for investment risk which makes it useful for financial analysis.

To calculate the variance of returns, firstly should the daily returns R_i be calculated and use it to calculate the average of the returns R^*. Secondly, calculate the variance of the returns which equal to the average of the square of the difference between the returns and the average of the returns. Then the standard deviations (the square root of the variance of the returns) must be calculated to produce the daily volatility.

$$V_d = \sqrt{\left(\frac{1}{n-1}\right) * \sum_{i=1}^{n} (R_i - R^*)^2} \tag{3.8}$$

where V_d is the daily volatility, n represents the total number of the data sample, R_i illustrate the returns for each time period, and R^* is the average of the returns.

After the daily volatility is calculated, then it will be easy to get the annualised volatility over the year by calculated it as follows:-

$$AV = \sqrt{252} * \sqrt{\left(\frac{1}{n-1}\right) * \sum_{i=1}^{n} (R_i - R^*)^2} \tag{3.9}$$

where 252 is the number of trading days in a year.

3. Maximum Draw-Down (MDD)

The Maximum Draw-Down (MDD) measure has been used to evaluate the trading risk of financial prediction for various network's models that have been used in this research. This MDD measure is defined as " The maximum loss from a peak to trough of a portfolio's value before a new peak is attained" [3].

Simply, MDD is an indicator measure of downside risk over a specified time period. That means MDD only measures the size of the largest loss of a financial trading, without consideration to the frequency of large losses.

[3] as defined in https://www.investopedia.com/terms/m/maximum-drawdown-mdd.asp

A low value of MDD indicates a less risk or small losing, which is desirable on financial prediction. Maximum Drawdown is computed as follows:

$$MDD = min\left(\sum_{i=1}^{n}(CR_i - max(CR_1, \therefore, CR_t))\right) \tag{3.10}$$

$$CR = \sum_{i=1}^{t}(R_i), t = 1, \therefore, n \tag{3.11}$$

$$R_i = \begin{cases} +|y_i| & \text{if}(y_i)(y_i^*) \geq 0 \\ -|y_i| & \text{otherwise} \end{cases} \tag{3.12}$$

where n is the total number of data samples, y_i is the target output value and y_i^* represent the predicted output value.

3.7.4 Signal Processing and Statistical Evaluation Metrics

1. Mean Squared Error (MSE)

The Mean Squared Error (MSE) is the square of the error between the target output and the forecasting output. The MSE is frequently used in the neural networks to measure the prediction error during the training phase and testing phase. The smaller value of the MSE refers to the lowest error of prediction. The Mean Squared Error is calculated as in the follows:

$$MSE = \frac{1}{n} * \sum_{i=1}^{n}(Y_i - Y_i^*)^2 \tag{3.13}$$

where n represents the total number of the data samples, Y_i refers to the target output, and Y_i^* is the forecasting output.

2. Mean Absolute Error (MAE)

The MAE measure the mean absolute error value of the deviation between the actual value (target output) and predicted values. In other words, the average magnitude of the errors for a set of forecasts measures by the MAE, without considering their direction[41]. The smaller value of the MAE, the closer predicted time series values to the target values. MAE is computed as follows:

$$MAE = \frac{1}{n} * \sum_{i=1}^{n}|Y_i - Y_i^*|^2 \tag{3.14}$$

where n represents the total number of the data sample, Y_i refer to the actual of data, and Y^* is the forecasting signals.

3. Signal to Noise Ratio (SNR)

The signal to noise ratio measure is used to compares the amount of significant information provides by the signal with the amount of background noise of the signal (distraction from the signal). The signal to noise ratio is given in decibels (dB), A higher ratio refers to a clearer reading of the signal. The SNR used in many digital processing applications such as image processing and electronic communications. Signal to Noise Ratio is calculated as in follows:

$$SSE = \sum_{i=1}^{n} (Y_i - Yi^*)^2 \tag{3.15}$$

$$m = max(Y_i) \tag{3.16}$$

$$sigma = \frac{m^2 * n}{SSE} \tag{3.17}$$

$$SNR = 10 * log_{10}(sigma) \tag{3.18}$$

where m is the max value of the target data y, n represents the total number of the data sample.

4. Correct Directional Change (CDC)

The aim of using the CDC is to measure the ability of the network models on correctly forecasting the subsequent actual change of a prediction variable. A large value refers to a better predictor. Correct Directional Change computed as in follows:

$$CDC = \frac{1}{n} * \sum_{i=1}^{n} d_i \tag{3.19}$$

$$d_i = \begin{cases} 1 & \text{if} (y_i - y_{i-1})(y_i^* - y_{i-1}^*) \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{3.20}$$

where n represents the total number of the data sample, y_i is the target output value and y_i^* represent the predicted output value.

3.8 Implementation

Initially, the implementation was done in Matlab for the MLP, FLNN, SMIA and FL-SMIA models.

For the extensions and the final evaluation all models have been (re-)implemented in the Python programming language. The TensorFlow library have been used to build for the implementation of the networks structure and the learning methods of all the neural networks used in this research and all evaluation metrics. The SMIA algorithm has been implemented with NumPy as a pre-processing step.

The models have been trained evaluated at City, University of London's data centre, on a server machine with 20 XEON cores, 128GB memory and 2 Nvidia QUADRO M4000 GPUs, as well as a 'white box' with a 8 RYZEN cores, 32 GB of memory and 2 Nvidia GTX 1080 Ti GPUs. It turned out that the GPU is the main factor for processing time.

Chapter 4

The FL-SMIA Neural Network Model

In this chapter, a novel model, the proposed Functional Link-Self-organised Multilayer network using the Immune Algorithm (FL-SMIA) combines aspects of Functional Link Neural Network with the Self-organising Multilayer network using the Immune Algorithm in the structure and the learning algorithm will be introduced.

4.1 Introduction

Multilayer networks trained using the back-propagation algorithm are considered as powerful models for financial time series prediction. However, these models suffer from various problems and weaknesses such as estimating the best weight values and determine the optimal number of hidden units, as the selection of these parameters is considered very important for developing the performance of multilayer networks. In addition, the MLP networks are affected by learning algorithm problems such as over-fitting [139], [52].

The over-fitting problem relates to the capability of the MLP network to update the connections between the layers. In this way, MLP networks can memorise the training data including useless noise. Often the training data are not uniformly distributed in input space, such that there will be not enough training data in a region in input space for the network to correctly approximate the function in this region. In both cases, this will lead to a poor generalization capability.

One method among several, which has been proposed to solve the over-fitting problem and improve the generalization capability of back-propagation neural networks, is the hidden layer self-organised network proposed by Widyanto et al [149] based on the immune algorithm by Timmis [144]. Widyanto et al [149] applied their model to the prediction of sinusoidal signal and time temperature based quality food data. The self-organised hidden layer network inspired by the immune algorithm showed improved experimental results in the prediction of sinusoidal signals, as well as in the prediction of temperature based food quality data. A number of researchers had used immune algorithms for solving several problems as in [150], [143], [131],and [101].

Artificial Immune System (AIS) has been successfully applied for several research areas, such as data mining, computer security, time series prediction, pattern recognition, and process control. AIS is based on the concept of the biological immune system in which the behaviours of the body cells and the antigen are utilised [131].

4.2 Biological Immune System and Immune Algorithm

The concept of the immune algorithm was initially discussed in [144]. The first self-organisation inspired by the immune system appeared in [82], where the researchers used one layer networks combined with the contiguity constraint method for clustering analysis. Later, the networks has been improved by adding the back-propagation output layer and applied for food quality prediction by [149].

Biologically, when the cell is matched with an antigen then this antigen stimulates the cell to duplicate itself and a mutated cell to produce a diverse set of antibodies in order to remove and fight the intruder attacking the body [131]. Thus, the immune system can allow its components to change and learn patterns by changing the strength of connections between individual components.

The Immune Algorithm (IA) belongs to the field of Artificial Immune Systems, which consists of a set of computational methods inspired by the process of the biological immune system. There are many algorithms in this field, such as clonal selection algorithms, immune network algorithms, and negative selection algorithms. These and many other algorithms have been used to solve machine learning problems in several domains with different approaches, including clustering, pattern recognition, classification and optimisation among others.

The Clonal Selection algorithm (CLONALG) is inspired by the Clonal Selection theory of acquired immunity. Clonal Selection theory was proposed to account for the behaviour and capabilities of antibodies in the acquired immune system [15]. The general learning strategy of clonal selection theory involves a population of adaptive information units (each unit representing a problem-solution or component) subjected to a competitive process for selection, these components ultimately improve the adapted of the information units with the environment. Several types of CLONING have been proposed later, such as the B-Cell Algorithm [78], the Multi-objective Immune System Algorithm (MSIRA) [25], the Optimization Immune Algorithm (opt-IA, opt-IMMALG) [26] and the Simple Immunological Algorithm [27].

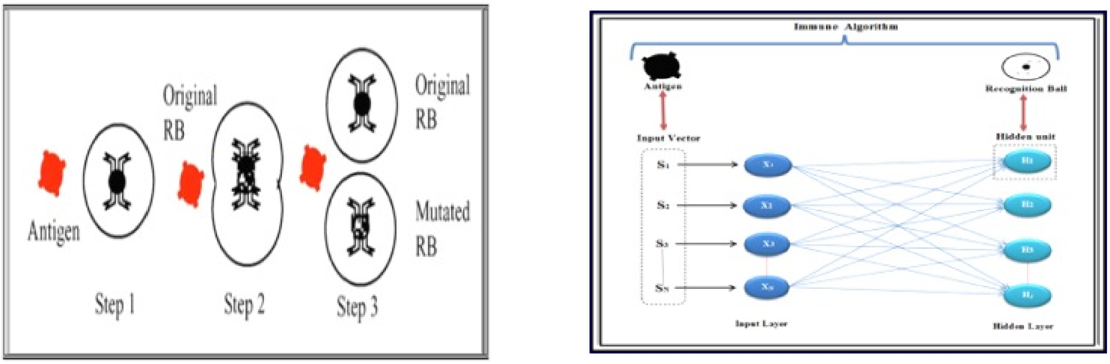
The artificial Immune Network algorithm has been inspired from the Immune Network theory of the acquired immune system. In 1974 an Immune Network Theory (Idiotypic Networks) has proposed by Jerne [73]. The researcher had explained that the immune cells are not at rest in the absence of the pathogen, instead, antibody and immune cells recognise and respond to each other. The objective of the immune network algorithm is to prepare a repertoire of discrete pattern detectors for a given problem domain, in a way that the cells of better performing suppress the low-affinity cells in the proposed model. Later the Immune Network Algorithm has been named the Artificial Immune Network (AIN), that when it had been used for a clustering approach by Timmis in 2000 [143].

In 1994, the negative selection algorithm was proposed by Forrest, et al.[49]. The Negative Selection Algorithm is inspired by the self-non-self recognition behaviours that observed in the mammalian gained immune system. Later (In 1996), the negative selection algorithm has been developed as an algorithm for change detection [33]. The information processing principles of the self-non-self-discrimination process via negative selection is achieved by building a model of changes, anomalies, or unknown (non-normal or non-self) data by generating patterns which do not match an existing corpus of available (self or normal) patterns. The prepared non-normal model is then used to either monitor the existing normal data or streams of new data by seeking matches to the non-normal patterns. An example of the Negative Selection Algorithm is a research work that represented by S. Hofmeyr and S. Forrest in [69][70] as an adaptive framework is the ARTificial Immune System (ARTIS).

The immune algorithm is based on the principle of the natural immune system. It is based on the relationship between its components that consist of antigens and cell which are called recognition balls (RB). The recognition ball in the immune system includes a single epitope and many paratopes where the epitope is attached to the B cell, and paratopes are attached to antigen. The B-cell here will represent several antigens. In neural network, the self-organised layer using immune algorithm serves as a hidden layer in the network (FL-SMIA). The input layer consist of input units represent as antigens and the hidden units in the hidden layer are considered as recognition ball (RB).

The immune algorithm is used to create hidden units. The relation between the antigens and the RB is based on the definition of local pattern relationships between input vectors and hidden nodes. These relationships help this network to easily recognise and define the input data's local characteristics, which increases the network's ability to recognise patterns and improve the generalization ability of the network [82].

In 2009 [101] the Self-organised Multilayer neural network using the Immune Algorithm (SMIA) is proposed as a novel application for financial time series prediction. The simulation result for one step ahead and five steps ahead proved that the SMIA networks produce a better percentage of annualised return than the other multilayer networks. A simple comparison between a biological immune system and immune



(a) Biological immune system as in [149]

(b) SMIA network as in [103]

Fig. 4.1 Biological immune system and immune algorithm

algorithm will shown in table 4.1 bellow:-

Table 4.1 Comparison between a biological immune system and an immune algorithm

	Biological Immune System	**Immune Algorithm**
1	Antigen	Input vector
2	B-cell	Hidden units
3	New antigen (not recognised)	New input vector (not matching)
4	New antigen stimulates a B-cell to replicate	A value not matching a hidden unit
5	A mutated B-cell will created	A new hidden unit will created

4.3 The Structure of the FL-SMIA Network

The architecture of the proposed FL-SMIA network consists of the input layer, which comprises a number of input units $X_1, X_2, \ldots, X_Z$, the self-organising hidden layer with units $H_1, H_2, \ldots, H_N$, and the output layer consisting of one output unit as shown in Fig. 4.2. Here Z, and N refers to the number of units in each layer. This research

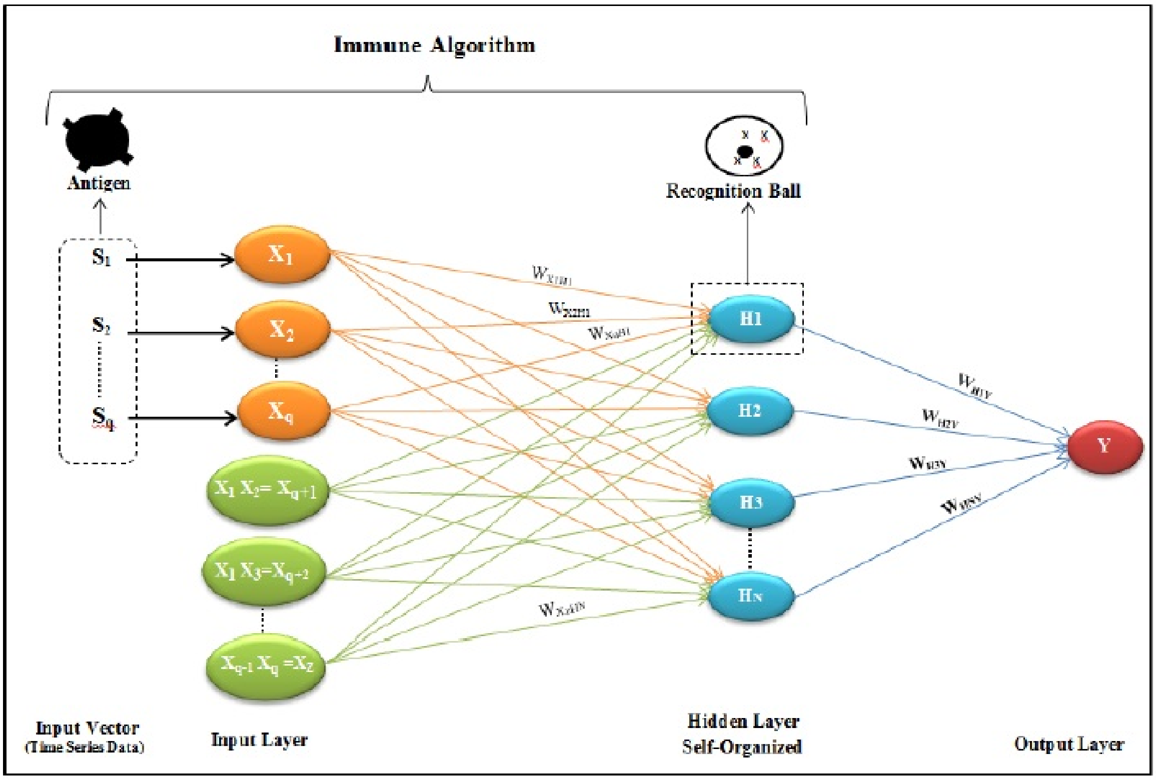


Fig. 4.2 The proposed FL-SMIA architecture (Functional Link Self-organised Multilayer network using the Immune Algorithm).

focuses on adding second order terms to the input units. In our example below the network has five input features $X_1, \ldots, X_5$. Adding the second order term to the inputs results in 10 additional inputs $(X_1X_2, X_1X_3, X_1X_4, \ldots, X_4X_5)$, leading to fifteen input units in total, five with input features and ten with products of inputs features as represented in Fig. 4.2. The FL-SMIA network uses a hidden layer which operates like in [101] and [149]. The design of the hidden units is inspired by B cells recognising pathogens in immune systems.

The output of the hidden units is determined using the Euclidean distance between the input units (X_i) and the connection weights between the input units and the hidden units (W_{Hij}). The advantage of using the Euclidean distance is to make the network capable of exploiting local information of the input data. The output of a hidden unit H_j is calculated as:

$$H_j = f_{hts}\left(\sqrt{\sum_{i=1}^{Z}(W_{Hij} - X_i)^2}\right) \tag{4.1}$$

where W_{Hij} represents the weight of the connection from the i^{th} input unit to the j^{th} hidden unit, and f_{hts} is the hyperbolic tangent sigmoid function.

The number of hidden units is determined from the data by learning with the Immune Algorithm as described in the next section.

The outputs of the hidden units are aggregated in a standard layer with the network output given by:

$$Y = f_{ls}\left(\sum_{j=1}^{N} W_{Hjy}.H_j + B_y\right) \tag{4.2}$$

where W_{Hjy} represent the strength of the connection weights between the j^{th} hidden unit and the output unit, B_y is the bias of the output unit Y, and f_{ls} is the logistic sigmoid function.

In general, the FL-SMIA model proposes to use the immune algorithm into train the first sets of connection weights, after that the second sets of weights use the standard back-propagation algorithm, in order to produce the network's output which then uses on evaluating and selecting models based on the metrics.

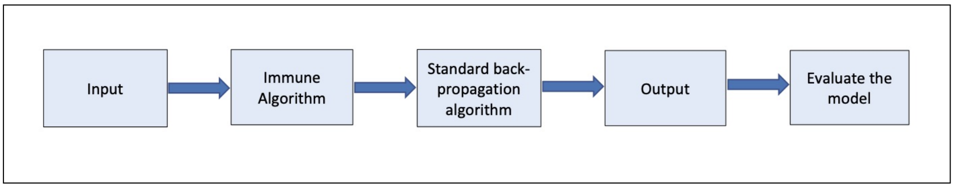


Fig. 4.3 The theoretical proposed framework for FL-SMIA network.

4.4 Learning in the FL-SMIA Network

The FL-SMIA as described above has two weight matrices, the first between the input layer and the hidden layer, the second between the hidden layer and the output layer. The first set of weights and the structure of the hidden layer are trained using the Immune Algorithm [149]. As indicated in figure 4.4, the immune learning algorithm starts with representing the input vector with weights to the hidden units. If the input vector values matches with the hidden unit, this means the input has found its corresponding hidden unit, and then the connection weights values should be updated. Otherwise, the input vector values are not matching with any hidden units, in this case a new hidden unit will be created. The process is repeated until new hidden units stop created.

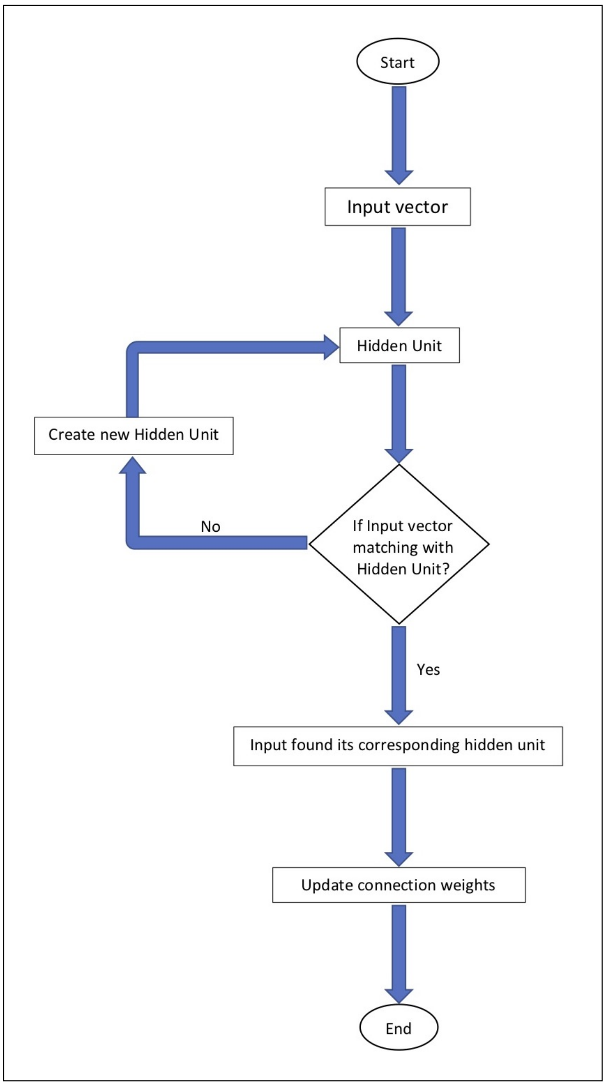


Fig. 4.4 Immune algorithm framework.

For more explanation about training the first set of weights using the Immune Algorithm and the structure of the hidden layer, it could be summarised in the follows. In the Immune Algorithm a hidden unit corresponds to a recognition ball (RB) in the immune system. Each hidden unit represents one or more input vectors with the weights of the connections from the input layer to the hidden unit The hidden unit Hj is represented by (P_j, W_{Hj}), where P_j is the number of input vectors associated with Hj, and W_{Hj} is the vector of weights from the input layer to Hj hidden units.

We start with one hidden unit ($N = 1$) and the first hidden unit is created with $P_1 = 1$ and $W_{H1} = X_1$. The Immune Algorithm then performs the following steps to create and update the hidden units until all inputs of the network have found their corresponding hidden unit.

1) For $m = 1, \ldots, M$ perform the following:

   a) For $j = 1, \ldots, N$, calculate the Euclidean distance between the m-th input and the weight vector of the j^{th} hidden unit:

$$dist_{mj} = \sqrt{\sum_{i=1}^{Z} (x_{mi} - w_{Hji})^2} \tag{4.3}$$

   where x_{mi} is the i^{th} element of input vector x_m, and w_{Hji} is the i^{th} component of vector w_{Hj}, i.e. the weight of the connection from input m to hidden unit j.

   b) Determine the closest unit c, i.e. the unit with the shortest distance to x_m:

$$dist_{mc} = min_j(dist_{mj}) \tag{4.4}$$

   c) If the shortest distance $dist_{mc}$ is below the stimulation level Sq (where Sq is selected between 0 and 1), then the input has found its corresponding hidden unit. In this case the weight vector w_{Hc} of the hidden unit closest to x_m will be updated as following:

$$w_{Hc_{new}} = w_{Hc} + \eta * dist_{mc} \tag{4.5}$$

   where $\eta_i \in (0, 1)$ is the learning rate for the Immune Algorithm, w_{Hc} is the weight vector of the hidden unit closest to x_m. P_c will be incremented by 1. Otherwise, the shortest distance $dist_{mc}$ is greater than the stimulation level Sq. This means that no matching hidden unit was found for the input and we create a new hidden unit (P_N, W_N) with $P_N = 1$ and $W_{HN} = X_m$. Then we update the following:

$$N = N + 1 \tag{4.6}$$

2) Repeat from step 1 as long as new hidden units have been created.

The second weight matrix is trained using the standard back-propagation algorithm [127] with regularisation to penalise large weights [13] in batch mode.

In our case with a single output neuron the weight change is calculated as:

$$\Delta W_{Hjk} = -\eta_b \frac{\partial J}{\partial W_{Hj}} - \lambda W_{Hjk} \tag{4.7}$$

where W_{Hjk} is the weight of the connection from hidden units Hj to the output unit, $\eta_b \in [0,1]$ is the learning rate, and J the mean squared error on the training set. The second term on the right-hand side effects the regularisation, which is controlled by the parameter λ. The bias is adapted in the same way but without regularisation.

The SMIA and SMIA* algorithms only create and never merge hidden neurons. Therefore, the algorithm terminates at the latest when there one hidden neuron is assigned to each data point. With the stimulation level that we use (0.45) we get typically around 40 hidden neurons with FL-SMIA and somewhat fewer with FL-SMIA*.

4.5 The FL-SMIA* Network: an Alternative Learning Method

As an alternative to the SMIA method, we introduced a second method to update the connection weights from input units to hidden units after we found the shortest distance $dist_{mc}$. In this network we assign each input vector the same importance. This method calculates the new vector for the selected hidden unit as the average of all input vectors assigned to the unit by replacing equation 4.5 with the following equation:

$$w_{Hc_{new}} = \frac{w_{Hc}(P_{Hc} - 1) + x_m}{P_{Hc}} \tag{4.8}$$

where P_{Hc} is the number of input vectors associated with the hidden unit Hc. We refer to this second method as FL-SMIA*. By applying the Immune Algorithm in either variant, a hidden layer representation is created that reflects the variety of vectors in the input data, which helps avoid over- and under-fitting problems because the hidden layer expands with the size and diversity of the training data. The units in the hidden layer contain explicit patterns that the network uses for prediction, which can be examined and interpreted by financial domain experts.

Chapter 5

Experimental Results of FL-SMIA Network

This chapter includes the simulation results of the proposed network Functional Link-Self-organised Multilayer network using the Immune Algorithm (FL-SMIA), which is supported by the regularization technique.

As the main objective of this research is to improve the prediction ability of non-linear networks (e.g. MLP network), therefore, the results of FL-SMIA and FL-SMIA* networks have been compared with the results of the Multilayer Perceptrons network (MLP), the Self-organising Multilayer network using the Immune Algorithm (SMIA). We also evaluated the Functional Link Neural Network (FLNN) because the FLNN network has been successfully used on financial prediction domain and we use the concept of the FLNN inputs with the FL-SMIA network.

All the networks have been tested on all data-sets that are listed in Table 3.1 (the financial time series data) using the metrics which have been explained previously in section 3.7.2.

In this research, we evaluate all combinations of learning parameters in a grid search as listed in section 3.6 on each of the nine data-sets. For each point in the grid (i.e. each parameter combination), 50 simulations have been run with each neural network model on each of the financial time series. Each simulation includes 80 epochs. The results have been selected based on the best RP value that produced from each model that was used in this research.

The results for the FLNN model have been produced using the second-order terms (inputs and their products). The MLP model produced best predictions by using 8 hidden units for all data-sets except GOLD, where only 4 hidden units lead to the best predictions.

The number of hidden units that have been generated by the models (SMIA, FL-SMIA, and FL-SMIA*) varied according to each of the data sets as listed in Table 5.1

Table 5.1 shows that the number of hidden units which were created by the SMIA network is between 10 and 24 hidden units. While the FL-SMIA model had generated more hidden units numbers ranges between 40 to 86 hidden units. For the FL-SMIA* model, the number of hidden units is from 11 to 32 hidden units.

When comparing the number of hidden units in Table 5.1, it is clear that the FL-SMIA model has generated larger numbers of hidden units than what has been generated by SMIA and FL-SMIA* models.

Table 5.1 The number of hidden units that have been generated by different versions of SMIA models for different data-sets.

Network	USD/UKP	USD/EUR	JPY/USD	NQO	NQC	DJO	DJC	OIL	GOLD
SMIA	12	19	24	18	21	19	19	10	11
FL-SMIA	40	59	71	43	49	47	47	86	56
FL-SMIA*	16	32	25	16	20	11	17	42	31

For the rest of this chapter, the first section of this chapter includes the results and the comparison between the results for one day ahead prediction. In the second section of this chapter, the results for five days ahead prediction will be presented and compared.

5.1 One Day Ahead Prediction

In this section, the results of 50 simulations using the data-sets in Table 3.1 (the financial time series data) are presented for comparison and analysis of networks performance. The results of the comparison are based on the values of the profit and the values of the investment risk that produced by each network that has been used in this research. This means the network that has achieved a higher percentage of the Relative Profit (RP) and a lower value of the Annualised Volatility (AV) is considered the best model could be used on the financial prediction domain.

Table 5.2 illustrates the results of Relative Profit (RP) for all networks used in this research for the prediction of one day ahead and the average RP for all data.

The results of Relative Profit (RP) show that the FLNN network predicted the highest RP values than all other networks for six out of the nine data-sets. However, the FL-SMIA, and FL-SMIA* networks outperformed the FLNN network on the prediction of data DJC, OIL, and GOLD.

The forecasting results for the FL-SMIA network proved that using the immune algorithm improves the performance of this network compared with the other multilayer networks. The FL-SMIA network results are lower but still compititive.

The comparison results for the RP average for one day ahead prediction proved that on one hand the average of Relative Profit of FL-SMIA network is lower than the average of Relative Profit the FLNN by 2.074. On the other hand, the proposed network FL-SMIA produced the highest value (73.268) of average RP than all other multilayer networks. Thus, the FL-SMIA network could be considered as an improved model for a multilayer networks financial prediction domain.

Table 5.2 The results of the Relative Profit (RP) for all networks for one day ahead prediction and the average RP for all data.

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD	Average
FLNN	**81.107**	**80.037**	**79.716**	**72.297**	**68.910**	**72.477**	69.370	77.878	76.285	**75.342**
MLP	72.108	73.238	74.500	68.314	66.152	67.774	64.756	78.911	76.050	71.311
SMIA	73.715	78.126	75.646	71.889	67.112	62.450	62.585	70.192	70.615	71.142
FL-SMIA	76.225	76.648	75.044	70.415	66.695	68.687	**70.276**	75.970	**79.451**	**73.268**
FL-SMIA*	73.111	72.807	73.569	69.369	66.396	67.959	65.618	**81.201**	67.949	70.887

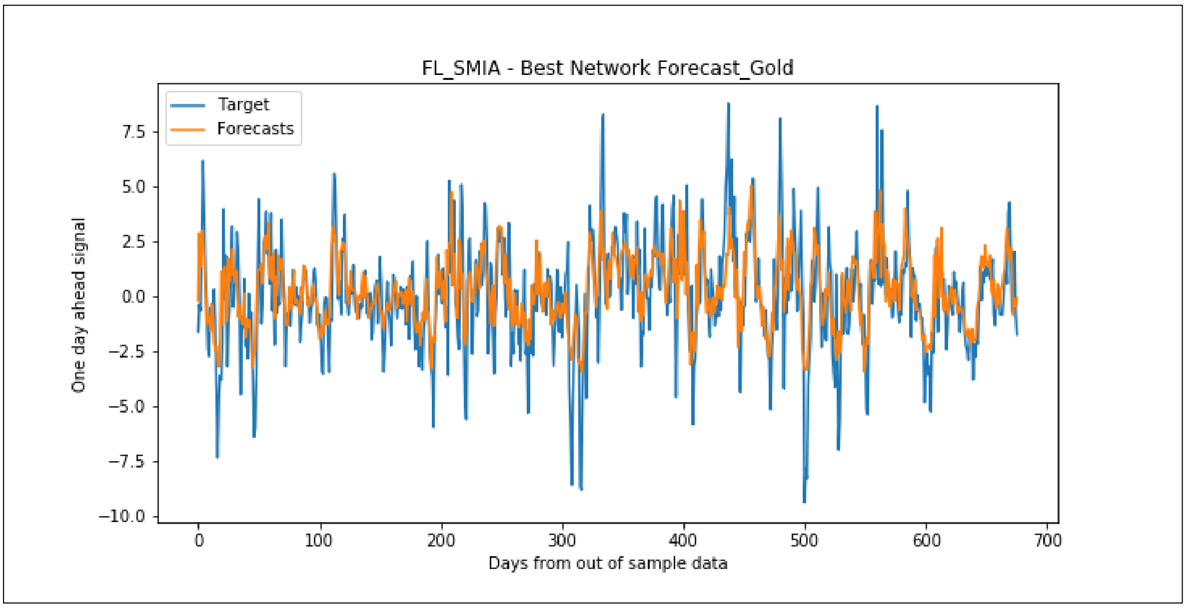


Fig. 5.1 The best forecasting for test data for the prediction for one day ahead using the FL-SMIA network.

Table 5.3 shows for the investment risk as measures by Annualised Volatility (AV, lower is better). The comparison of AV results indicates that the FLNN model reduces the trading risk by producing lower AV results than all other networks when using

Table 5.3 The results of Annualised Volatility (AV) for all networks for one day ahead prediction and the average AV for all data.

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD	Average
FLNN	**4.1191**	**4.3898**	**5.3227**	**12.6817**	**12.2235**	**10.5783**	10.7969	20.3893	32.3896	12.5434
MLP	4.3763	4.5828	5.4748	12.9159	12.3757	10.8177	11.0213	20.2407	32.4344	12.6933
SMIA	4.3328	4.4428	5.4412	12.7031	12.3184	11.0668	11.1218	16.7822	15.7535	**10.4403**
FL-SMIA	4.6260	4.4865	5.4590	12.7885	12.3434	10.7724	**10.7459**	**16.6670**	**15.5541**	**10.3825**
FL-SMIA*	4.3487	4.5900	5.4700	12.8467	12.3604	10.8080	10.9742	26.6794	38.4523	14.0589

the data of US/UK, US/EU, JP/US, NQO, NQC, and DJO. However, the FL-SMIA network outperformed all other networks when predicting the AV value for DJC stock price data. For the commodity price data (OIL, and GOLD), the FL-SMIA network had lower AV values than all other networks.

The results for average Annualised Volatility (AV) in table 5.3 proved that FL-SMIA model reduced the investment risk by produced the lowest average of (AV) value than all other networks including the FLNN model.

Figure 5.1 illustrates an example of FL-SMIA network performance when using the GOLD data. As can be seen, the target signal (data), and the forecasted signal are in mostly closer to each other, which indicate that the FL-SMIA network has the ability to learn the behaviour of financial data.

Table 5.4 The results for the MSE-Testing for one day ahead prediction and the average of MSE-Testing for all data.

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD	Average
FLNN	0.00992	0.00841	0.01247	0.00401	0.00573	0.00505	0.00395	0.00712	0.005891	0.00695
MLP	0.00384	0.00294	**0.00238**	0.00215	**0.00296**	0.00327	0.00314	0.00420	0.004450	0.00326
SMIA	0.00382	**0.00291**	0.00240	0.00220	0.00320	0.00240	**0.00229**	0.00030	0.000071	**0.00218**
FL-SMIA	**0.00381**	0.00340	0.00260	0.00230	0.00330	0.00241	0.00230	**0.00025**	**0.000067**	**0.00227**
FL-SMIA*	0.00442	0.00348	0.00256	**0.00210**	0.00306	**0.00215**	0.00240	0.00221	0.002950	0.00283

As clearly seen in table 5.4, that the results for MSE-Testing phase indicated all networks produced low values for the MSE-Testing measure when forecasting all the data. However, the comparison of average results demonstrates that the SMIA and FL-SMIA networks produced the lower average of MSE- Testing values than all other networks which have been used in this research.

For the multilayer networks, the average results of MSE refer that the SMIA network is outperforming all multi-layer networks, while the average value of MSE for FL-SMIA network is close to the average value of MSE for SMIA network.

Furthermore, the MSE average for the FL-SMIA network (0.00227) outperformed the FLNN network which produced the average of MSE (0.00695). It is good to notice

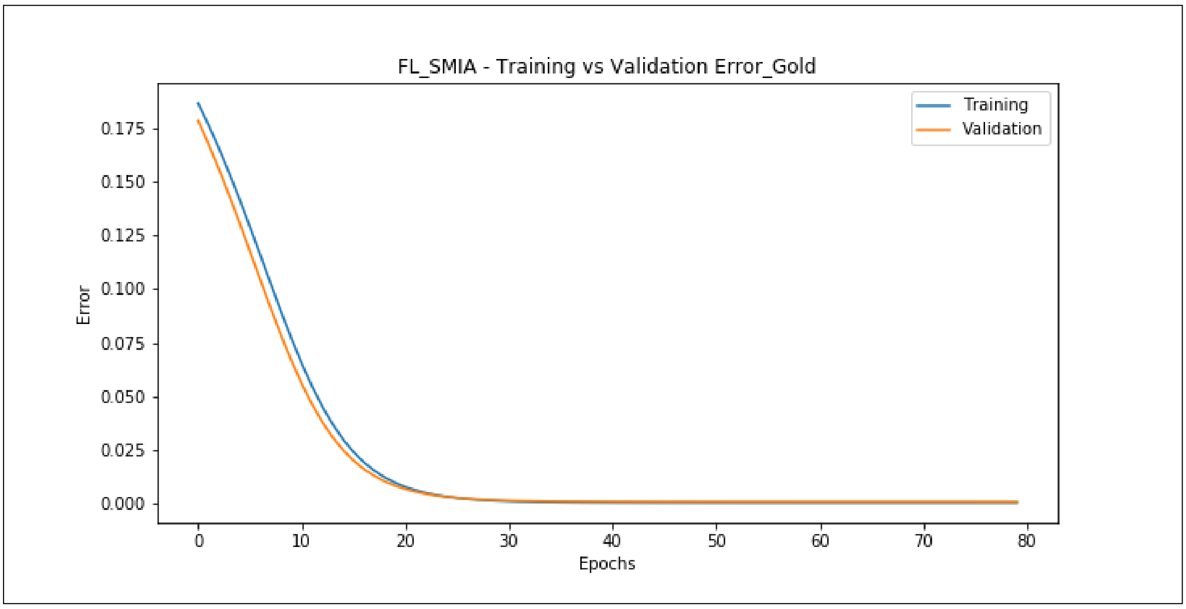


Fig. 5.2 The training error and validation error for the GOLD data for one day ahead prediction using FL-SMIA network.

that FL-SMIA network has successfully down the MSE value than what the lowest value than all other MSE values of all networks when predicted the GOLD data.

Figure 5.2 illustrates the errors of training and validation for the FL-SMIA network when using the GOLD data. The errors of training and validation are looking closer to each other and they are near to zero value. consequently, the results of one day ahead prediction proved that the proposed FL-SMIA network improved the multi-layer network's performance on financial prediction domain.

5.2 Five Days Ahead Prediction

This section is illustrated the experimental results for the prediction for five days ahead using financial time series data. The main objective of this section is to appraisal the predictive ability of the networks (FLNN, MLP, SMIA, FL-SMIA, and FL-SMIA*). The network which acquires the higher ratio of profits is considered as the best model for the financial prediction.

The networks have been tested using all data-sets from Table 3.1 (the financial time series data) using the metrics which have been explained previously in the section 3.7.2. As well as, the results of FL-SMIA network will be compared with all networks that have been used in this research in order to investigate if the proposed networks FL-SMIA or FL-SMIA* could outperform the other networks for five days ahead prediction.

Table 5.5 The results of the Relative Profit (RP) for all networks for five days ahead prediction and the average RP for all data.

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD	Average
FLNN	91.529	92.452	86.985	**88.165**	86.397	**89.522**	89.564	89.814	**90.009**	**89.382**
MLP	91.185	91.501	83.338	83.214	80.772	80.399	84.377	89.723	88.962	85.941
SMIA	88.403	91.927	86.237	86.566	87.257	84.469	86.468	**90.892**	89.595	87.979
FL-SMIA	**92.021**	**92.495**	**87.492**	86.626	**87.341**	86.625	87.454	87.629	89.497	**88.575**
FL-SMIA*	90.765	92.475	87.054	87.279	87.016	87.995	**89.723**	86.236	83.758	88.033

As could be seen through table 5.5, the average results of the Relative Profit (RP) for five days ahead prediction for all networks that used in this research showed that the FLNN network outperforms the proposed network (FL-SMIA) with the results (89.382 vs 88.575) respectively. However, the FL-SMIA network attained the highest average value of RP than all the multi-layer networks. This result proved that the proposed network (FL-SMIA) improved the performance of multi-layer networks for financial prediction.

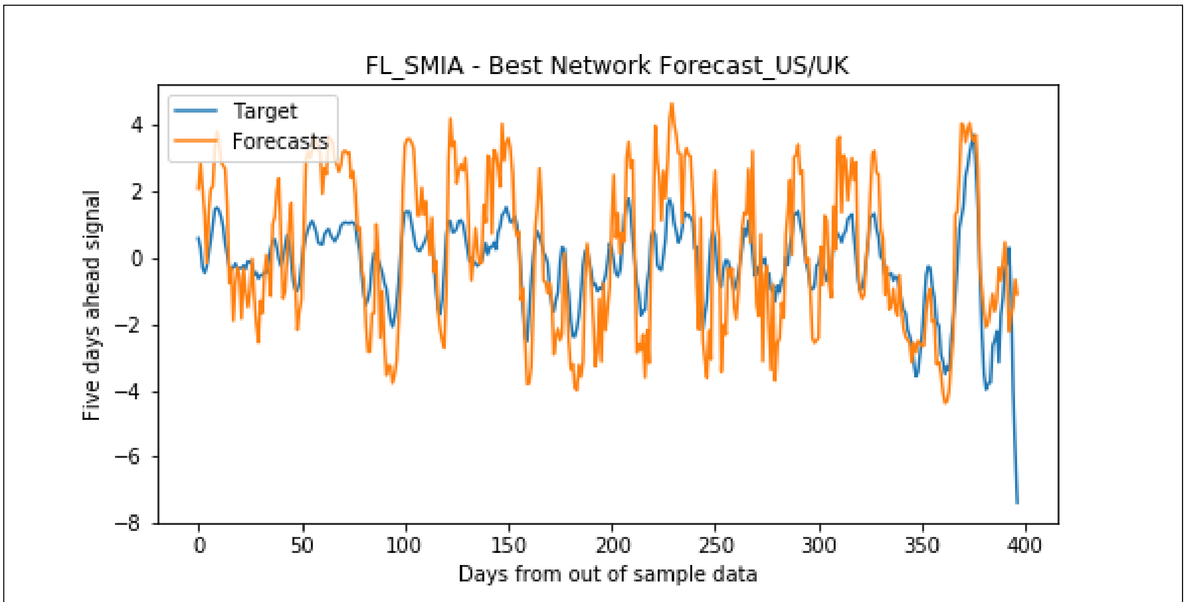


Fig. 5.3 The best forecasting on test data for the prediction of five steps ahead using FL-SMIA network.

An example of the FL-SMIA network prediction for the US/UK data has been illustrated in Figure 5.3, the forecasted signal looks mostly followed to the target signal, which means that the FL-SMIA network has the ability to learn the behaviour of financial data.

Table 5.6 The results of the Annualised Volatility (AV) for all networks for five days ahead prediction and the average AV for all data.

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD	Average
FLNN	**15.47144**	15.56260	16.81314	**35.16634**	35.75308	**30.72444**	30.75543	61.52982	**101.34412**	**38.12449**
MLP	15.49770	15.68840	17.29550	36.42510	37.12470	32.65290	31.90650	61.59024	102.29560	38.94185
SMIA	15.85660	15.62360	16.89820	35.57702	**35.47790**	31.82670	31.4531	**60.80959**	101.72290	39.22406
FL-SMIA	15.65030	**15.53630**	**16.71910**	35.56130	35.73694	31.36340	31.23310	62.93981	101.81230	**38.50584**
FL-SMIA*	15.55320	15.53940	16.78210	35.38960	35.54270	31.05930	**30.71120**	63.74985	106.65943	38.99853

The average AV results as in table 5.6 indicated that FLNN model reduced the investment risk by produced the lowest average AV value than all other networks followed by the average AV value of FL-SMIA model. Furthermore, comparing the average AV results between the multi-layer networks proved that the FL-SMIA model outperforms all other multi-layer networks by decrease the average AV value to (38.50584). In

Table 5.7 The results for the MSE-Testing for five days ahead prediction and the average of MSE-Testing for all data.

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD	Average
FLNN	0.00739	0.01046	0.01358	0.00468	0.00555	0.00327	0.00375	0.00732	0.01222	0.00758
MLP	0.00264	**0.00173**	0.00261	0.00305	0.00385	0.00130	0.00230	0.00626	0.00594	0.00330
SMIA	**0.00158**	0.00264	**0.00236**	0.00141	**0.00177**	0.00136	0.00155	0.00186	0.00954	0.00268
FL-SMIA	0.00204	0.00319	0.00254	0.00150	0.00190	0.00165	0.00167	0.00279	0.00271	**0.00222**
FL-SMIA*	0.00220	0.00310	0.00313	**0.00140**	0.00199	**0.00122**	**0.00120**	**0.00178**	**0.00227**	**0.00203**

table 5.7, the results of the MSE-Testing and the average for all networks for the prediction of five days ahead have been shown. Although all networks reach low error values of MSE-Testing for all the data, the comparison between the average results for the networks demonstrates that the proposed networks (FL-SMIA and FL-SMIA*) have the lowest average of MSE-Testing values than all other networks which have been used in this research.

Figure 5.4 demonstrate an example of the training and validation errors for FL-SMIA network when using the US/UK data. The figure presents the MSE results for the training data error and validation data error for the prediction of five days ahead. The training error and the validation error are closer to each other and are reduce the errors by near to zero value.

Through all the tables illustrated in this chapter which presented the performance of all the networks that have used in this research. The average results for the one day ahead prediction and for five days ahead prediction proved that the proposed network (FL-SMIA) outperformed all the multi-layer networks that used in this research.

These good results considered the proposed network (FL-SMIA) as the promising multi-layer model on financial prediction domain, as well as, these results encourages

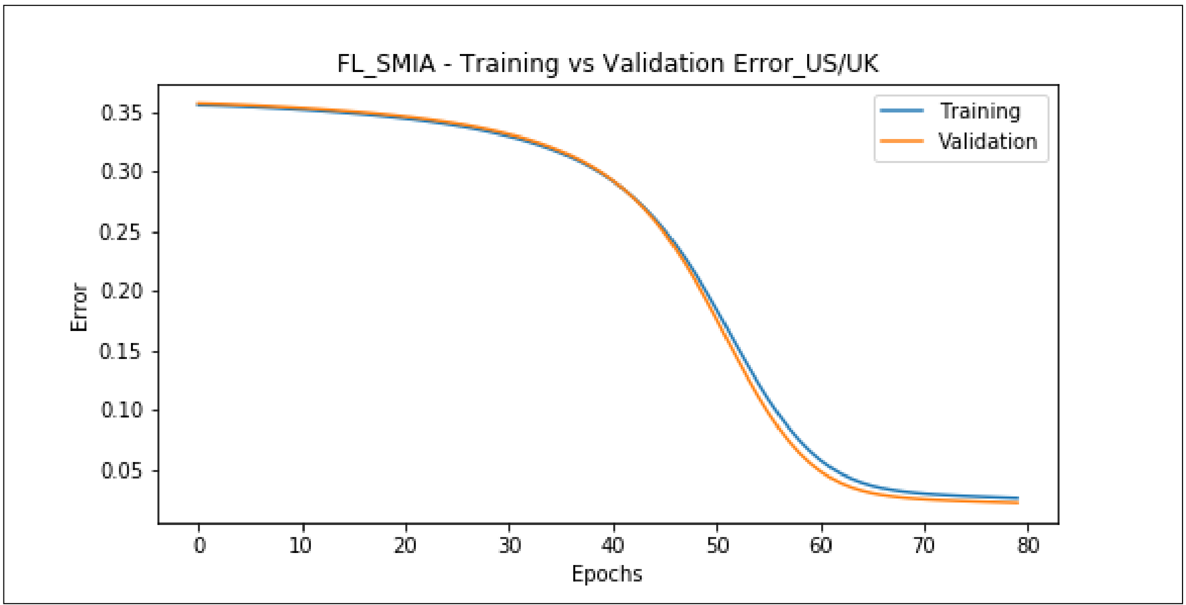


Fig. 5.4 The development of the MSE for the prediction of five steps ahead using FL-SMIA network.

the continuation of this research towards improving and developing the proposed network (FL-SMIA) in order to achieve more accurate results for financial data.

All information about average performance and the statistical significance of these results and of the differences between architectures for all models that have been used in this research will be included in chapter 8.

Chapter 6

Extensions 1: Deeper and Mixed Networks: D-FL-SMIA, MD-FL-SMIA, MD-FL-SMIA-2, M-FL-SMIA

As the main focus of machine learning research recently has been on deep, recurrent or convolution neural networks operating on raw data, methods for constructing features and alternative learning algorithms have still potential for improving predictive performance.

In this chapter, the developing of the proposed network (FL-SMIA) to deeper learning network using the Immune Algorithm (D-FL-SMIA) will be explained. As well as, the improved networks using the idea of mixed data (MD-FL-SMIA and M-FL-SMIA networks) will be explained.

The proposed networks (D-FL-SMIA, MD-FL-SMIA, MD-FL-SMIA-2, and M-FL-SMIA) aims to improve the performance of the Multilayer networks. It is good to note that a regularization technique has been used with all proposed networks in order to improve the generalization of the proposed network.

In this research, the number of hidden units in the standard hidden layers have been determined based on the error results for the networks without using an algorithm to determine the number of hidden units for each layer except the first layer (using the Immune Algorithm to generate the hidden units with each set of data).

Experimentally, the results with the lower error for the D-FL-SMIA model have been produced by using three standard hidden layers, each layer includes a different number of hidden units (150, 100, and 50, respectively). Each model of (MD-FL-SMIA and MD-FL-SMIA-2) contains two standard hidden layers with a different number of hidden units (100 and 50 hidden units respectively).

6.1 The D-FL-SMIA Network Structure

The structure of the proposed network (FL-SMIA) have been changed by adding three extra hidden layers to this network, thus, the D-FL-SMIA network includes four hidden layers: the first hidden layer which is the self-organised layer using the immune learning algorithm (unsupervised learning algorithm), and the three extra hidden layers are the standard hidden layers, which is used the Back Propagation Algorithm (supervised learning algorithm).

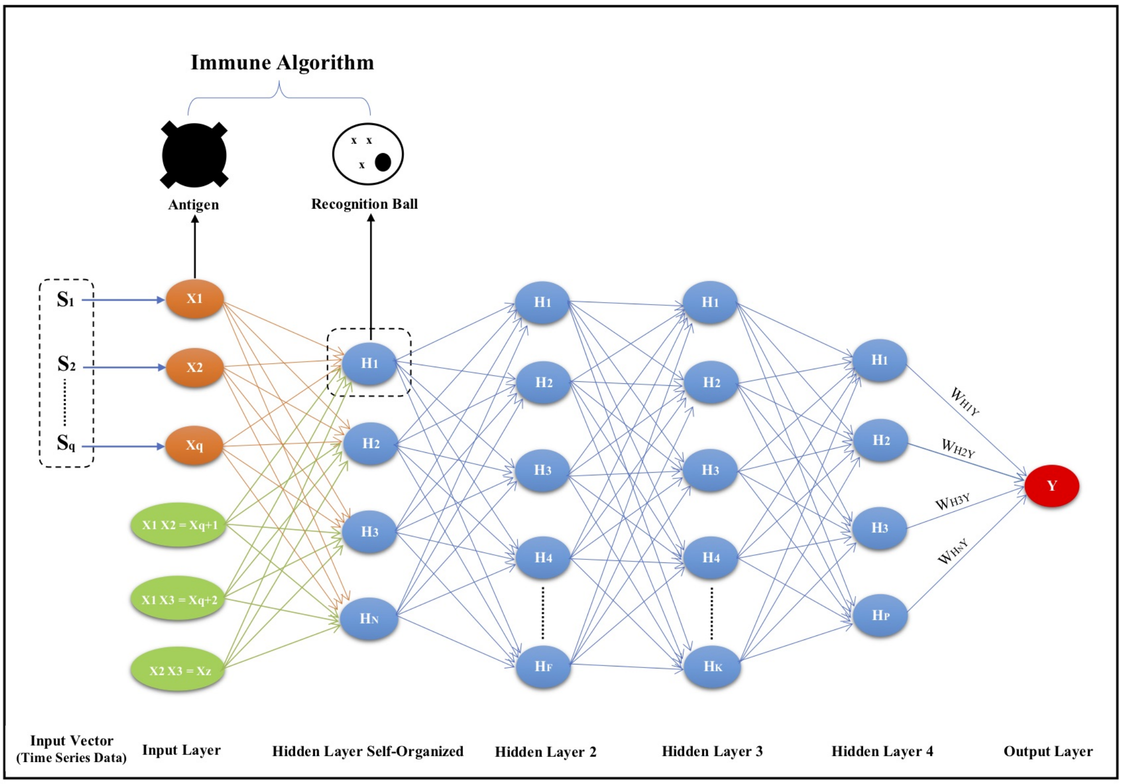


Fig. 6.1 The proposed D-FL-SMIA architecture (Deep Functional Link Self-organised Multilayer using the Immune Algorithm).

As shown in fig 6.1, the structure of the proposed D-FL-SMIA network consists of the following:

1. The input layer includes a number of input units $(X_1, X_2, ...X_Z)$).

2. The self-organising hidden layer (SMIA layer) with units $(H_1, H_2, ...H_N)$, where the number of hidden units in the SMIA layer depends on the type of data and the immune algorithm.

3. Three standard hidden layers, each layer includes a different number of hidden units $(H_F, H_K, andH_P)$, where F = 150, K = 100, and P = 50, respectively, refers to the number of hidden units in each layer.

4. The output layer includes only one output unit (Y).

Depends on layers number that described previously, the D-FL-SMIA network has five sets of weights matrices, the first set between the input layer and the self-organised hidden layer (SMIA layer), the second set of connection weights are between the self-organised hidden layer and the standard hidden layer, then the third and fourth weight matrix is between the rest of standard hidden layers, the last weight matrix is located between the fourth hidden layer and the output layer.

The data are passed in from the input layer to the first hidden layer, the output of the first hidden layer considered as inputs to the next hidden layer and so on, while the output of the last hidden layer sends to the output unit (Y) which is the network output.

6.2 The D-FL-SMIA Network Learning

As explained in chapter (4) of this thesis, the input layer receive external data (normalised time series data) i.e., $(S_1, S_2, ..., S_q)$, where $S \in [0, 1]$, and thus $X_q \in [0, 1]$. The number of actual input units (X) equal to the number of external input data (S). Thus, the input units are $X_1, X_2, ...X_Z$, where Z represents the number of input units to the network includes the actual inputs $(X_1, X_2, ..., X_q)$ and their products $(X_1X_2 = X_{q+1}, X_1X_3 = X_{q+2}, ..., X_{q-1}X_q = X_Z)$.

For the first hidden layer, the proposed network use the immune algorithm as an unsupervised learning method using the financial data as in table 3.1. The hidden units are designed as an implementation of the B cell recognition algorithm that inspired from the immune system as in [149], and Mahdi et al. [101]. Each hidden unit receives fifteen inputs, which means there are fifteen connection weights between each hidden unit and input units.

The output of the hidden units is determined using the Euclidean distance between the input units (X_i) and (W_{Hij}) (the connection weights between the input units and the hidden unit) as in equation 4.1.

The number of hidden units is determined based on the type of data and the use of the Immune learning Algorithm (explained in chapter 4 section 4.4). Thus, the number of hidden units for the first hidden layer will be vary depending on the data-sets. Then the outputs of the first hidden layer transform to the second hidden layer.

The second hidden layer, which is a standard hidden layer use a number of hidden units equal to 150 units, while the third hidden layer (standard hidden layer) used 100 hidden units, and the fourth hidden layer (standard hidden layer) with 50 units, all standard layers are training by using the back-propagation algorithm (supervised learning).

During the training phase, the main goal is to minimise the error between the target values and the network output values. In order to reach this goal, the weight vectors must be updated.

The output of each hidden layer considered as an input to the next hidden layer and so on. The outputs of the standard hidden layers are calculated as follows:

$$H_k = f_{ls}\left(\sum_{j=1}^{N} W_{jk}H_j + B_k\right), k = 1, 2, ..., V \tag{6.1}$$

where W_{jk} represent the connection weights from j^{th} hidden unit to k^{th} the hidden units in the next hidden layer, B_k is the bias associated with the hidden units for each hidden layer, and f_{ls} is the logistic sigmoid function.

The outputs of the last hidden layer are aggregated in a standard layer. The network output is given by:

$$Y = f_{ls}\left(\sum_{j=1}^{N} W_{Hjy}H_j + B_y\right) \tag{6.2}$$

where W_{Hjy} represent the strength of the connection weights between the j^{th} hidden units and the output unit, B_y is the bias of the output unit Y, and f_{ls} is the logistic sigmoid function.

6.3 The MD-FL-SMIA Networks

In this section, the proposed MD-FL-SMIA network (Mixed and Deeper Functional Link Self-organised Multilayer using the Immune Algorithm) will be presented. The aim of proposing this version of the network is to investigate the effect of the mixture of raw data with trained data (using immune algorithm) on the performance of the neural network on the field of financial prediction.

Two versions of MD-FL-SMIA network have been proposed in this research using the method of mixed and Deep network for prediction, the two versions have similar structures but the difference between these models is based on the dimensionality of the inputs of the data, as it will be explained in following:

A) The first version titled the MD-FL-SMIA network, which uses inputs of **Data (inputs)**= $(X_1, X_2, ...X_q)$ mixed with the data which trained using immune algorithm in the first hidden layer.

B) The second version titled MD-FL-SMIA-2 network, which uses inputs of **FL-Data (inputs + higher order inputs)** = $(X_1, X_2, ...X_Z)$ mixed with Data that trained by using the immune algorithm in the first hidden layer.

These two versions of networks (MD-FL-SMIA, and MD-FL-SMIA-2) have been proposed in order to investigate the prediction performance of each network by using a different type of mixed data.

6.3.1 The MD-FL-SMIA Network Structure

The proposed MD-FL-SMIA network as shown in figure 6.2 consists of the following:

1. **Input layer:** includes number of input units $X_1, X_2, ...X_Z$ (inputs and their products).

2. **Self-organising hidden layer:** is the first hidden layer with hidden units $(H_1, H_2, ...H_N)$, where the number of hidden units in this layer depends on the data and the immune algorithm. The number of hidden units in this layer = **Data (inputs =** $X_1, X_2, ...X_q$**))** + hidden units created using immune learning algorithm. (e.g. US/UK = 40 hidden units + 5 hidden units, total hidden units = 45).

3. **Hidden layer 2:** is a standard hidden layer, it includes number of hidden units $(H_1, H_2, ...H_F)$, where F = 100.

4. **Hidden layer 3:** is a standard hidden layer, which includes $(H_1, H_2, ...H_K)$ number of hidden units, where K = 50.

5. **Output layer:** this layer consist of only one output unit (Y).

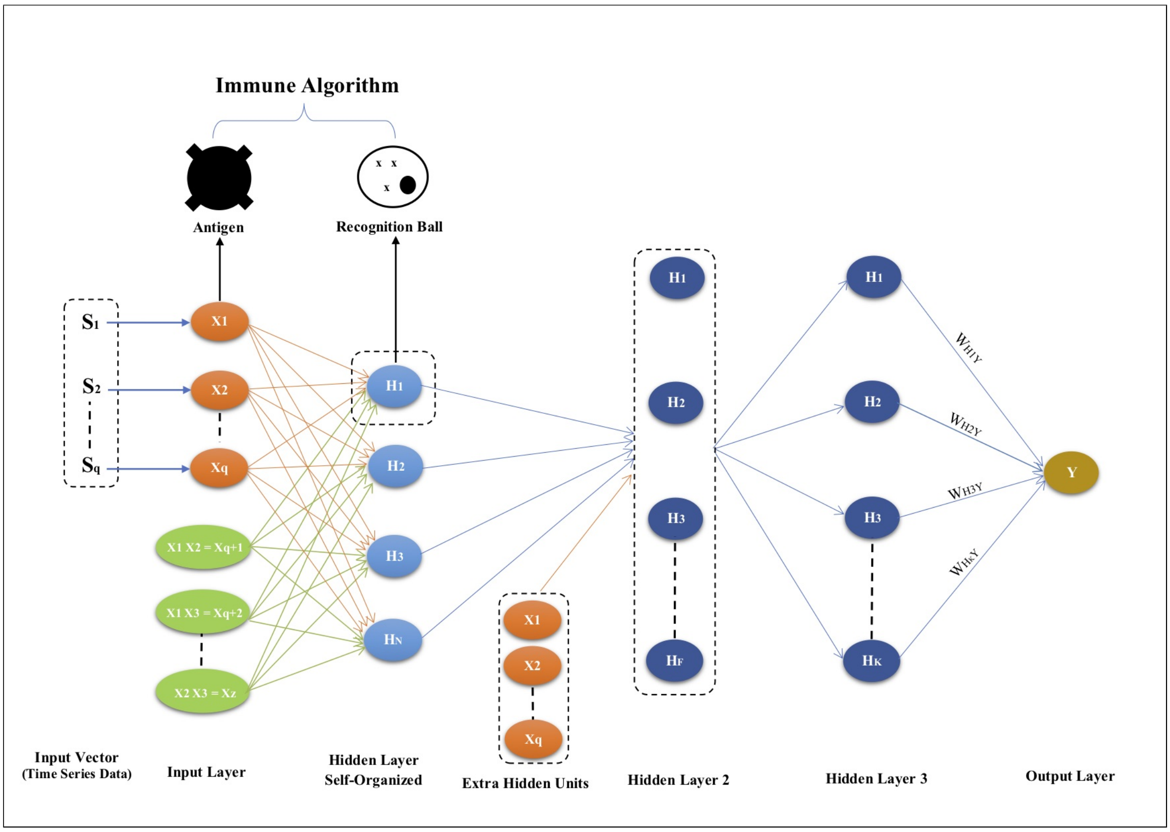


Fig. 6.2 The proposed MD-FL-SMIA architecture (Mixed and Deeper Functional Link Self-organised Multilayer using the Immune Algorithm).

6.3.2 The MD-FL-SMIA-2 Network Structure

As shown in figure 6.3, the proposed MD-FL-SMIA-2 model structure includes the following:

1. **Input layer:** includes number of input units $X_1, X_2, ...X_Z$ (inputs and their products).

2. **Self-organising hidden layer:** is the first hidden layer with hidden units $(H_1, H_2, ...H_N)$. The number of hidden units in this layer = the number of hidden units that equal to **FL-Data (inputs + higher order inputs =** $X_1, X_2, ...X_Z$**)** + hidden units created using immune learning algorithm. (e.g. US/UK = 40 hidden units + 15 hidden units, total hidden units = 55).

3. **Hidden layer 2:** is a standard hidden layer, it includes number of hidden units $(H_1, H_2, ...H_F)$, where F = 100.

4. **Hidden layer 3:** is a standard hidden layer, which includes $(H_1, H_2, ...H_K)$ number of hidden units, where K = 50.

5. **Output layer:** this layer consist of only one output unit (Y).

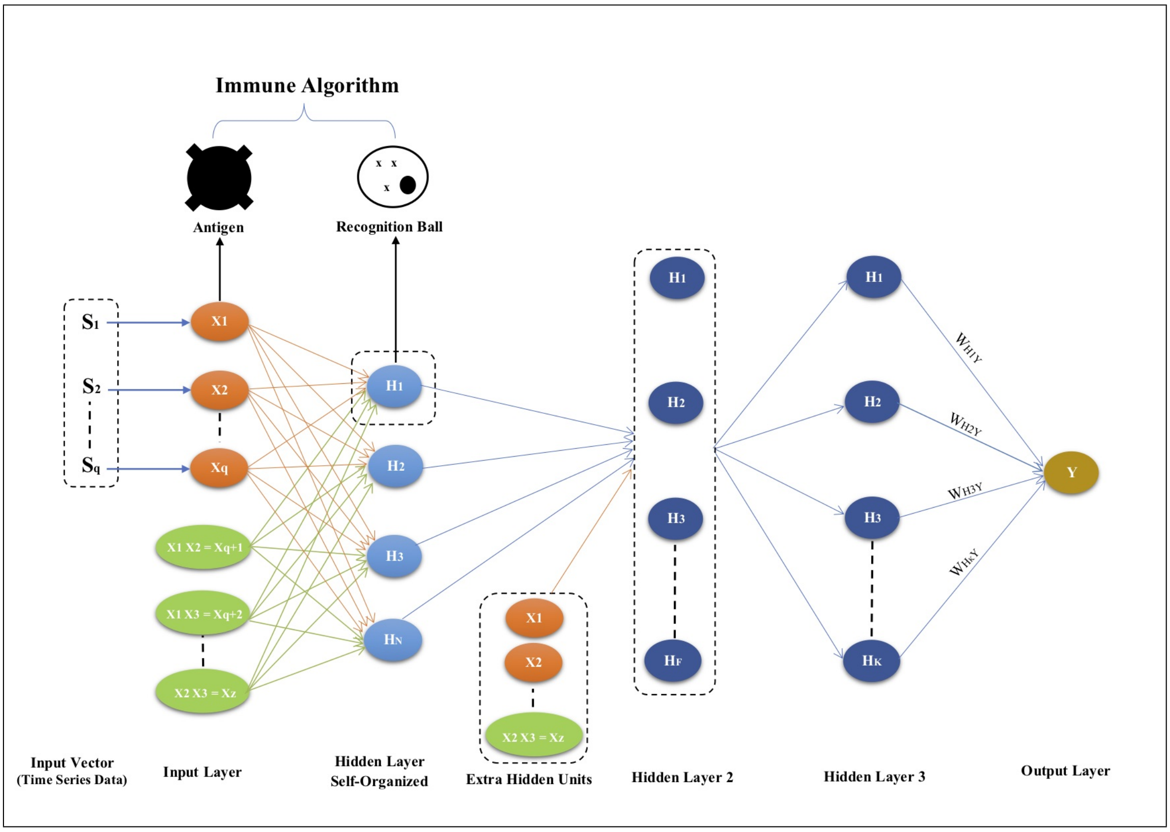


Fig. 6.3 The proposed MD-FL-SMIA-2 architecture (Mixed and Deeper Functional Link Self-organised Multilayer using the Immune Algorithm).

As presented in figure 6.2 and figure 6.3, the input units receive external data (normalised time series data) i.e., $(S_1, S_2, ..., S_q)$, where $S \in [0, 1]$, thus $X_q \in [0, 1]$.

6.4 The MD-FL-SMIA and MD-FL-SMIA-2 Networks Learning

The MD-FL-SMIA network and MD-FL-SMIA-2 network have four weight matrices, the first weight matrix between the input layer and the first hidden layer (Self-organised layer), the second weight matrix between the first hidden layer and the second hidden layer,and so on until the weight matrix between the last hidden layer and the output layer.

As explained in chapter 4, the first set of weights are trained using the Immune Algorithm [149].

The first hidden layer (SMIA layer) received the FL-Data from input layer and trained the FL-Data using unsupervised learning method (immune algorithm). As mentioned previously in section 4.4, that in this layer the number of hidden units is varied as it depends on the type of the data-set and the immune learning algorithm which is used to create the hidden units. After the FL-Data is passed to the network, the first weight matrix between the input layer and the first layer (Self-organised layer or SMIA layer) are trained.

Then the output of the (SMIA layer) will be transferred to the second hidden layer. The dimensions of the second weights matrix between the first hidden layer and the second hidden layer will be equal to the number of hidden units that generated by (SMIA layer) + number of hidden units which are equal to the number of inputs (**Data**, or **FL-Data**). In other words, the hidden layer 2 will receive mixed values of inputs depending on the type of network as follows:

1) The MD-FL-SMIA Network: the second weights matrix which is trained by using a number of weights that equal to the mix of input **Data** + the output of the **FL-Data** which is trained using the immune algorithm.

2) The MD-FL-SMIA-2 Network: the second weights matrix which trained by using a number of weights that equal to the mix of input **FL-Data** + the output of the **FL-Data** which is trained using the immune algorithm.

The output of the second hidden layer will feed into the next layer (Hidden Layer 3) which consists of 50 hidden units, thus the output of the third hidden layer feed into the output layer which uses only one output unit as shown in fig 6.2.

The output of the hidden units in the first hidden layer is determined using the Euclidean distance between the input units (X_i) and the connection weights between the input units and the hidden units (W_{Hij}) as in equation 4.1.

The output of a hidden unit H_j is calculated as:

$$H_j = f_{hts}\left(\sqrt{\sum_{i=1}^{Z}(W_{Hij} - X_i)^2}\right) \tag{6.3}$$

where W_{Hij} represents the weight of the connection from the i^{th} input unit to the j^{th} hidden unit, and f_{hts} is the hyperbolic tangent function. The outputs of the first hidden layer are represents as inputs to the second hidden layer The output of the second hidden layer (Hidden Layer 2) is calculated as:

$$H_F = f_{ls}\left(\sum_{jf}^{F} W_{jf}H_j + B_f\right) \tag{6.4}$$

where H_F denotes the output of the second hidden layer, f_{ls} is a log-sigmoid transfer function, H_j represents the output of the first layer to the second hidden layer, W_{jf} is the weights connection between the self-organised hidden layer and the hidden layer 2, and B_f is the bias.

The network output is given by:

$$Y = f_{ls}\left(\sum_{k=1}^{K} W_{Hky}H_k + B_y\right) \tag{6.5}$$

where W_{Hky} represents the connection weights between the k^{th} hidden units and the output unit, B_y is the bias of the output unit Y, and f_{ls} is the logistic sigmoid function.

For the third and fourth weights matrix which are updated using the standard back-propagation algorithm (supervised learning algorithm)[127] with regularisation to penalise large weights [13]. In our case with a single output unit, the weights change is calculated as:

$$\Delta W_{HFK} = -\eta_b \frac{\partial J}{\partial W_{Hf}} - \lambda W_{HFK} \tag{6.6}$$

where W_{HFk} is the weight of the connection from hidden units fk to the output unit, $\eta_b \in [0, 1]$ is the learning rate, and J the mean squared error on the training set. The second term on the right-hand side effects the regularisation, which is controlled by the parameter λ. The bias is adapted in the same way but without regularisation.

6.5 The M-FL-SMIA Network

This section introduces a novel neural network architecture (M-FL-SMIA model) inspired by immune system. The M-FL-SMIA model combines the higher-order inputs (the products of raw input features) of the FLNN model with the Functional Link Self-organising Multilayer Neural Network using an Immune Algorithm (FL-SMIA).

The M-FL-SMIA Network structure consists of the following two components:

1) **The FLNN Network**

   The FLNN network is defined as a generalised linear model consisting of the input layer which including number of the actual inputs $(X_1, X_2, ..., X_q)$ and their products ($X_1X_2 = X_{q+1}, X_1X_3 = X_{q+2}, ..., X_{q-1}X_q = X_Z$) represented a vector of z features. $w = (w_1, w_2, ..., w_z)$ are the model parameters and b is the bias, Y refers to the prediction value (the network without hidden layer). The output calculation of the FLNN is calculated as following :

$$Y_{FL} = f_{ls}\left(W_0 + \sum_j W_j X_j + \sum_{j,k} W_{jk} X_j X_k\right) \tag{6.7}$$

where Y_{FL} denotes the FLNN network output, f_{ls} is a log-sigmoid transfer function, W_0 is the bias term, X represents the input values, and W is the weights from the input layer to the output layer.

It is good to mention that, the interactions between the features add non-linearity to the generalised linear model.

2) **The FL-SMIA Network**

As explained in chapter 4, that the FL-SMIA Network consists of the input layer, one hidden layer (SMIA layer), and output layer which includes one hidden unit.

The FL-SMIA model has two weight matrices. The first matrix is trained by using an unsupervised learning method (immune algorithm), while the second matrix trained using supervised learning (Back-Propagation algorithm). The output of the FL-SMIA network is given by:

$$Y = f_{ls}\left(\sum_{j=1}^{N} W_{Hjy} H_j + B_y\right) \tag{6.8}$$

where W_{Hjy} represents the connection weights between the j^{th} hidden units and the output unit, B_y is the bias of the output unit Y, and f_{ls} is the logistic sigmoid function.

6.6 Combining the Training of FLNN and FL-SMIA (M-FL-SMIA Network)

The M-FL-SMIA Network learn using a combination of FLNN and the FL-SMIA algorithms. These two models are combined using a weighted sum of their outputs on the output layer.

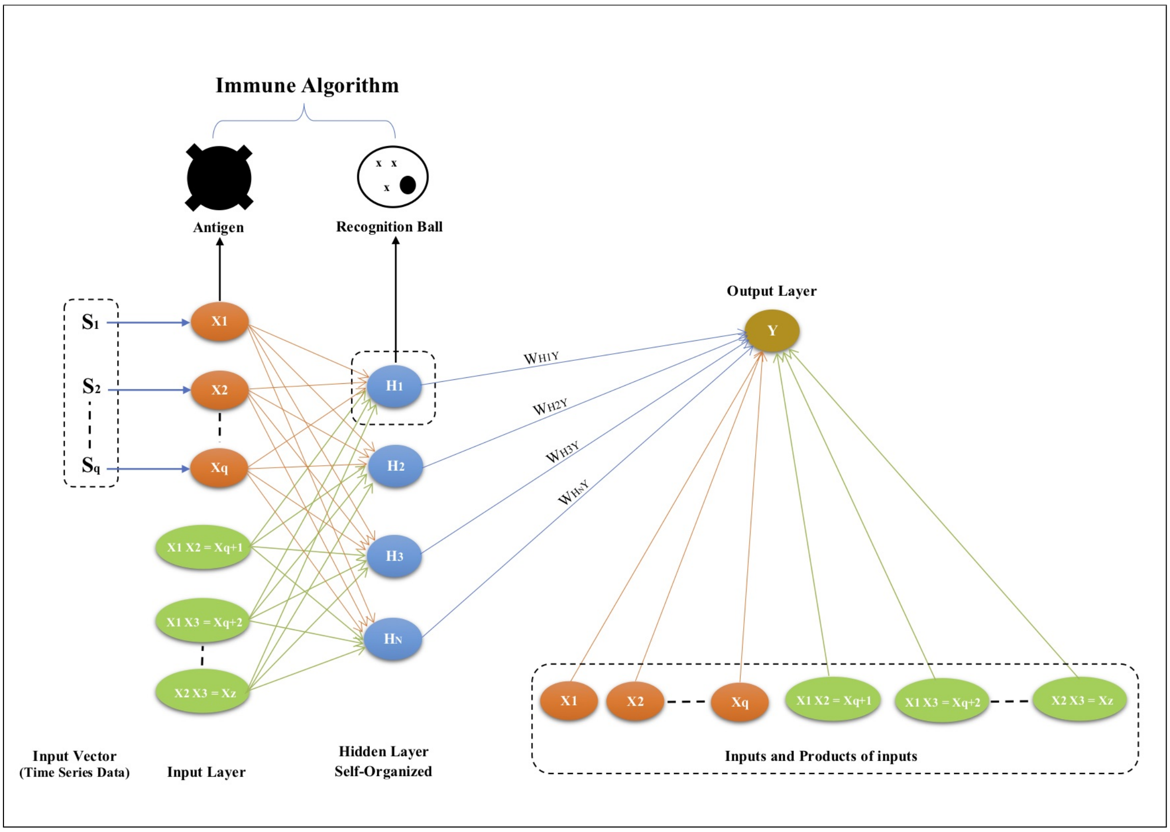


Fig. 6.4 The proposed M-FL-SMIA network architecture (Mixed of FLNN and FL-SMIA networks).

As illustrated in figure 6.4, the hidden layer in M-FL-SMIA network received the FL-Data $= X_1, X_2, ..., X_Z$. where Z represents the total number of input units (FL-Data) includes the input data and its products. The FL-Data is trained using unsupervised learning method (immune algorithm). The output of the hidden units in the hidden layer is calculated using the Euclidean distance between the input units (X_i) and the connection weights between the input units and the hidden units (W_{Hij}) as in equation 4.1. The outputs of the hidden layer then will be transfers to the output layer.

In addition to that, the output layer received inputs of FLNN network directly from the input layer in order to train using the back-propagate leaning algorithm. Thus, the output unit in the output layer produces the prediction results for the data-set using the mixed of network's data.

The combined model's prediction is:

$$Y = f_{ls}\left(W_{FLNN}[X_i...X_z] + (W_{FL-SMIA}) + B\right) \tag{6.9}$$

where Y is the network output, f_{lg} is the sigmoid function, $X_i ... x_z$ are the inputs and their product transformations of the original features x, and B is the bias term, W_{FLNN} is the vector of all FLNN model weights and $W_{FL-SMIA}$ are the weights between the SMIA layer and t, f_{hts} the hyperbolic tangent activation function.

The prediction results for all the proposed networks will be presented and compared with other networks in chapter 8 of this thesis.

Chapter 7

Extensions 2: FL-SMIA with a Restricted Boltzmann Machine (FL-SMIA-RBM)

In this chapter, a novel model based on learning using a Functional Link-Self-organised network of the Immune Algorithm (FL-SMIA) and the Restricted Boltzmann Machines (FL-SMIA-RBM) is proposed.

The Restricted Boltzmann Machine (RBM) is trained with an unsupervised learning method. The FL-SMIA model also uses an unsupervised learning method (an immune algorithm). Here, the aim is to combine the FL-SMIA and RBM in one model that uses unsupervised learning to improve the performance of a multi-layer network, inspired by Deep Belief Networks[67]. The Restricted Boltzmann Machine uses an unsupervised learning method, that is used as pre-training to render learning Deep Belief Networks more effective. The Restricted Boltzmann Machine in the proposed network (FL-SMIA-RBM) is used to reduce the dimensions of the hidden layer of the FL-SMIA network with the aim of improving the generalisation of the network.

7.1 The Structure of FL-SMIA-RBM Network

This section introduces the structure of the proposed FL-SMIA-RBM model. The FL-SMIA-RBM is a multilayer neural network model which including the following:

1. The input layer consists of the raw inputs and their products, i.e. second order inputs (15 input units in our case: 5 raw inputs and 10 second order inputs).

2. The first hidden layer (SMIA) is the self-organised hidden layer using the immune learning algorithm (unsupervised learning). In this layer, the number of hidden units depends on the data-set.

3. The second hidden layer is a standard hidden layer including 10 units. Combined with the previous hidden layer, it is trained as an RBM.

4. The output layer with only one unit. This layer is trained with the Back-Propagation Algorithm (supervised learning).

In this model, only 10 hidden units have been used in the second hidden layer. This number of hidden units has been determined during preliminary tests by monitoring the reconstructions loss when using the unsupervised learning method (RBM) and the final error results on the supervised task. Experimentally, the lowest results for the network's error have been produced with 10 hidden units in the second hidden layer of the FL-SMIA-RBM model. The 10 hidden units of the second hidden layer have been used for each of the data-sets.

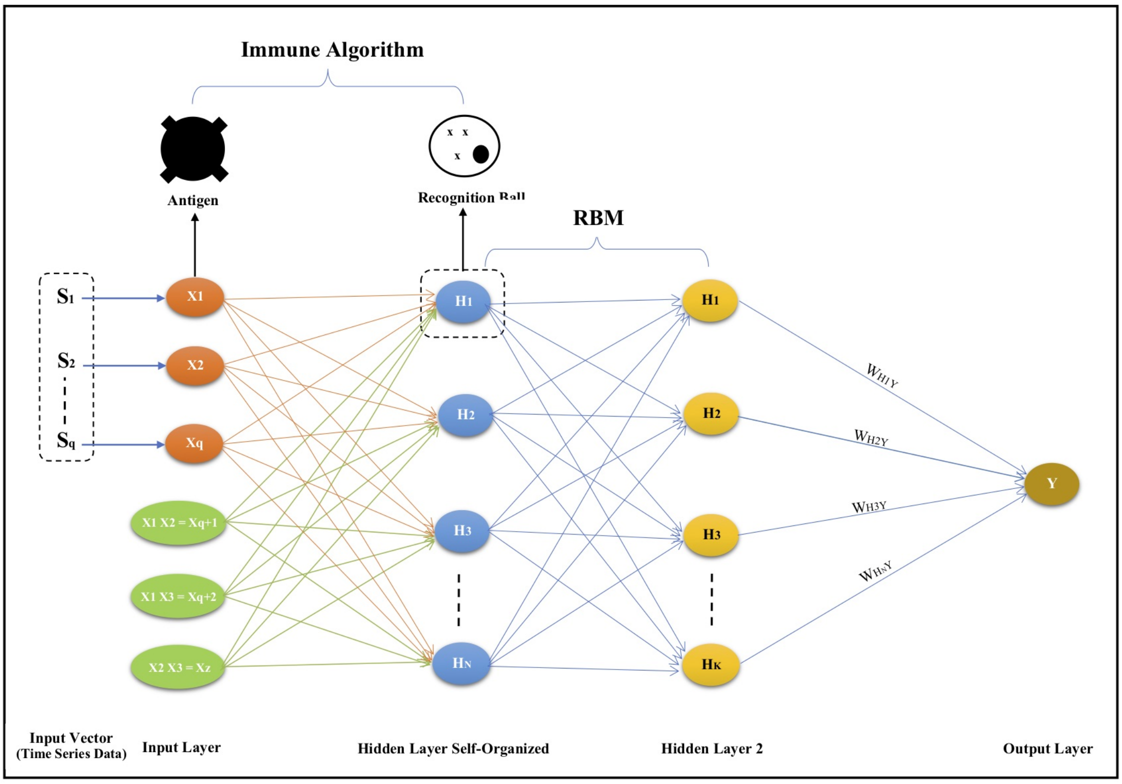


Fig. 7.1 The proposed network FL-SMIA-RBM

7.1.1 The FL-SMIA Network

As explained previously on chapter 4 and [102], the architecture of the proposed FL-SMIA network consists of the input layer, which includes a number of input units $(X_1, X_2, .\therefore X_Z)$, and the self-organising hidden layer with units $(H_1, H_2, .\therefore H_N)$, and the output layer consisting of one output unit. The FL-SMIA network uses a hidden layer as in [101] and [149]. The design of the hidden units is inspired by B cells recognising pathogens in immune systems.

The output of the hidden units is produced by using the Euclidean distance between the input units (X_i) and the connection weights between the input units and the hidden units (W_{Hij}) as in equation 4.1. The number of hidden units is determined from the data by learning with the Immune Algorithm as described in chapter 4 section 4.2. The outputs of the hidden units (SMIA layer) are represents as inputs to the second hidden layer (Hidden layer 2).

7.1.2 The Restricted Boltzmann Machine (RBM)

The Restricted Boltzmann Machine is an unsupervised learning model that has been introduced in [134]. RBMs have been used in several applications including classification[89], feature learning [24], dimensionality reduction [68], and as pre-training for representation learning in deep architectures [91]. The Restricted Boltzmann Machines (RBM) is defined as a network which has only two layers: the visible and the hidden layer, which is often used as output for further processing.

The visible neurons are connected to the hidden neurons in a stochastic way and without connections between each of visible neurons in the input layer or hidden neurons in the hidden layer. In other words, in an RBM there are no direct connections between units in the same layer, which is why it is called restricted [68, 65].

The RBM learns to extract features from the data by reconstructing the inputs. The learning process in the RBM aims to improve the reconstruction of the data (binary activation values in the visible layer) by adjusting the weights matrix between the visible and hidden layer (also binary activation) in the following steps [1][68] :

1. **Forward pass:** in this step, the data passed via the visible inputs forward to the hidden units as in figure 7.2, where each input is associated with individual weight and overall bias, which result in activating or deactivate the hidden units.

[1]The following two diagrams are inspired by the video https://www.youtube.com/watch?v=FsAvo0E5Pmw

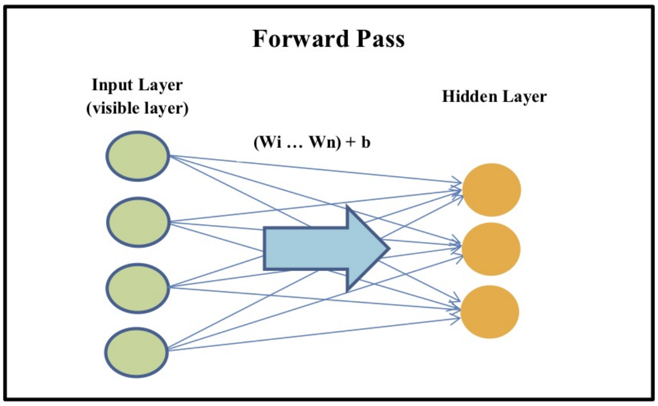


Fig. 7.2 The RBM-forward pass step

2. **Backward pass (reconstruction):** as shown in figure 7.3, during this step, the activation of the hidden units will be sent to the input units, which will be activated based on the individual weights and bias.

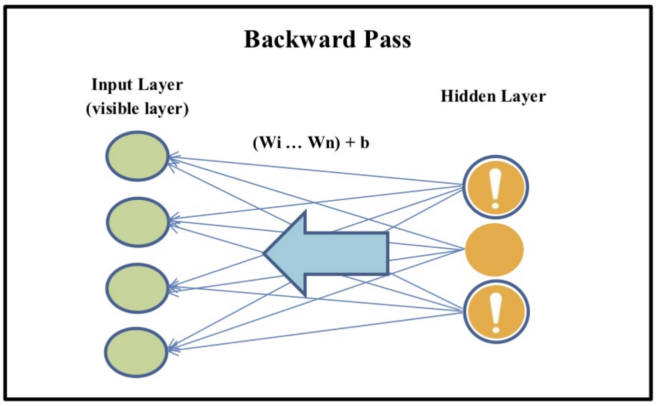


Fig. 7.3 The RBM-backward pass step

3. **Quality assessment (step 3):** in this step, the quality of the reconstruction is assessed by comparing the reconstructed from step 2 with the original input. We calculate and attempt to reduce the error by adjusting the weights and bias values.

7.1.3 Learning in the RBM

In this section the explanation of RBM learning with equations will discussed according to [71] and [66].

An RBM is a network of two layers, the first layer includes a number of inputs or visible units (v) as binary inputs, and the second layer with a number of hidden units (h), every hidden unit has a binary state of 0 or 1, therefore the input and hidden units are the random variables (v, h), the visible units are combined with biases a_i and the hidden units are associated with biases b_j.

The RBM is given by the following:

$$F(v,h) = -\sum_{i}^{N}\sum_{j}^{L} W_{ij} v_i h_j - \sum_{i}^{N} a_i v_i - \sum_{j}^{L} b_j h_j, \tag{7.1}$$

where v_i is the binary state of visible unit i, h_j is the binary state of hidden unit j, a_i and b_j are their biases, W_{ij} is the weight between the visible unit i and hidden unit j, N refers to the number of visible units, and L is the number of hidden units.

The probability of the network for a possible pair of visible and hidden variables is:

$$p(v,h) = \frac{1}{G} * e^{-F(v,h)} \tag{7.2}$$

where G is the partition function for normalisation, which is given by:

$$G = \sum_{i}^{N}\sum_{j}^{L} e^{-F(v,h)} \tag{7.3}$$

The probability of a visible vector v is determined by summing over all possible hidden vectors:

$$p(v) = \frac{1}{G}\sum_{j}^{L} e^{-F(v,h)} \tag{7.4}$$

In an RBM network there are no direct connections between hidden units which are simply lead to getting a sample of data. The binary state h_j, of each hidden unit j is set to 1 with probability:

$$P\left(h_j = 1|v\right) = f_{sig}\left(b_j + \sum_{i=1}^{N} W_{ij} v_i\right) \tag{7.5}$$

where f_{sig} is the sigmoid activation function $f_{sig} = 1/(1 + exp^{(} - x))$.

Also as there are no direct connections between visible units in RBM, the conditional distribution of a visible unit is defined as:

$$P(v_i = 1|h) = f_{sig}\left(a_i + \sum_{j=1}^{L} W_{ij}h_j\right) \tag{7.6}$$

where f_{sig} is the sigmoid transfer function,W_{Hij} represents the weight of the connection from the i^{th} input unit to the j^{th} hidden unit.

The derivative of the log probability of a training vector v with respect to a weight is:

$$\frac{\partial \log p(v)}{\partial w_{ij}} = \langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{model} \tag{7.7}$$

The $\langle \cdot \rangle$ indicates the expected value under the distribution in the index. For the model case we estimate the expectation with the a single reconstruction cycle and get a simple update rule [66]:

$$\Delta w_{ij} = \lambda \left((v_i h_j)_{data} - (v_i h_j)_{model} \right) \tag{7.8}$$

where λ is a learning rate. The update for the biases is analogous.

After determining the binary states of the hidden units using equation 7.5 , one binary state for hidden units are chosen, and by setting each v_i to 1 with a probability in equation 7.6 , a restriction is produced and the updated in weights is given by:

$$\Delta w_{ij} = \lambda \left((v_i h_j)_{data} - (v_i h_j)_{recon} \right) \tag{7.9}$$

After the second weight matrix of the FL-SMIA-RBM model (in figure 7.1) is updated, the values of the second hidden layer are aggregated as a standard hidden layer H_k with the network output given by:

$$Y = f_{ls}\left(\sum_{k=1}^{K} W_{ky}H_k + B_y\right) \tag{7.10}$$

where f_{ls} is the sigmoid activation function, W_{ky} represent the connection weights between the k^{th} hidden unit (second layer) and the output unit, H_k is the output of the second hidden layer, and B_y is the bias of the output unit Y.

The prediction results and all performance details for the proposed networks FL-SMIA-RBM will be presented and compared with other networks in chapter 8 of this research.

Chapter 8

The Experimental Results

This chapter includes the Experimental results of the proposed model (FL-SMIA, FL-SMIA*) and all extended networks of FL-SMIA model that have been proposed in this research, the extended networks include the Deeper of FL-SMIA network (D-FL-SMIA), the Mixed data networks (MD-FL-SMIA) and (MD-FL-SMIA2), the Mixed network of FLNN with FL-SMIA (M-FL-SMIA), and the FL-SMIA and Restricted Boltzmann Machines (FL-SMIA-RBM).

The prediction results of the proposed FL-SMIA network and the neural network models (FLNN, MLP, SMIA, and FL-SMIA*) has been shown in chapter 5. The average results of the RP, AV, and MSE-Testing metrics for one day ahead prediction and five days ahead prediction proved that the proposed FL-SMIA network outperformed all other multilayer networks. However, the FLNN network outperformed the proposed FL-SMIA network on the average of RP measure for one day ahead prediction and on the average values of RP, and AV for five days ahead prediction. While for the average of MSE-Testing, the proposed FL-SMIA network outperformed all other multilayer networks and the FLNN network for one day ahead prediction and five days ahead prediction.

In this chapter, the results of the FL-SMIA model and all proposed models have been compared with the results of other neural network models including the Multilayer Perceptrons network (MLP), the Self-organising Multilayer network using the Immune Algorithm (SMIA), and the Functional Link Neural network (FLNN). All networks in this research have been tested using all data-sets presented in Table 3.1.

8.1 The Prediction for One Day Ahead

The results of the prediction for one day ahead have been presented in this section through the tables below which are starting from table 8.1 to table 8.9.

The results of 50 simulations generated from the nine sets of financial time series using all neural network models in this research are presented. Each simulation includes 80 epoch, while each epoch use a combination of parameters in a grid search as listed in section 3.6. The results have been selected based on the best RP value that produced from each model used in this research

Each table includes the number of hidden units that has been used with each model structure for each data-sets. while for the FLNN model, the second-order has been used, which refers to the probability of inputs products. As well as, for all proposed models the second order have used in addition to the number of hidden units that created by the immune algorithm. In addition to the RP, AV, and MAE, the MSE results including MSE Training results and MSE-Testing results.

The best results for all networks have been illustrated in this section with focusing on the profit values; the network which generates a higher value of Relative Profit (RP) and lower values of AV and MSE-Testing is considered to be the best model for financial prediction.

The details regarding the comparison results, as well as the analysis for the prediction results for one day ahead have been included in the following subsection 8.1.1

Table 8.1 The best one day ahead prediction result for US/UK

Networks	Hidden No. or Order	RP	AV	MSE		MAE
				Training	Testing	
FLNN	2	**81.1074**	**4.1191**	0.00966	0.00992	0.0775
MLP	8	72.1075	4.3763	0.00404	0.00384	0.0483
SMIA	12	73.7150	4.3328	**0.00240**	0.00382	0.0479
FL-SMIA	40	76.2248	4.6260	0.00400	**0.00381**	**0.0471**
FL-SMIA*	16	73.1111	4.3487	0.00423	0.00442	0.0504
D-FL-SMIA	40	77.1528	4.2403	0.02410	0.02504	0.1256
MD-FL-SMIA	45	73.5375	4.3431	0.03119	0.03701	0.1587
MD-FL-SMIA-2	55	74.0055	4.3302	0.02169	0.02092	0.1165
M-FL-SMIA	55	75.6951	4.2827	0.03950	0.03508	0.1512
FL-SMIA-RBM	40	59.4492	4.6794	0.00843	0.00833	0.0704

Table 8.2 The best one day ahead prediction result for US/EU

Networks	Hidden No. or Order	RP	AV	MSE Training	MSE Testing	MAE
FLNN	2	**80.0367**	**4.3898**	0.00880	0.00841	0.0701
MLP	8	73.2384	4.5828	0.00365	0.00294	0.0405
SMIA	19	78.1255	4.4428	0.00330	**0.00291**	**0.0388**
FL-SMIA	59	76.6475	4.4865	**0.00320**	0.00340	0.0407
FL-SMIA*	32	72.8073	4.5900	0.00336	0.00348	0.0418
D-FL-SMIA	59	70.7636	4.6546	0.02304	0.01737	0.1015
MD-FL-SMIA	64	75.0903	4.5370	0.04052	0.03238	0.1460
MD-FL-SMIA-2	74	75.6655	4.5206	0.01026	0.01215	0.0817
M-FL-SMIA	74	75.0340	4.5386	0.02561	0.02407	0.1260
FL-SMIA-RBM	59	60.6847	4.6537	0.00775	0.00795	0.0684

Table 8.3 The best one day ahead prediction result for JP/US

Networks	Hidden No. or Order	RP	AV	MSE Training	MSE Testing	MAE
FLNN	2	**79.7162**	**5.3227**	0.00991	0.01247	0.0878
MLP	8	74.5002	5.4748	0.00174	**0.00238**	0.0369
SMIA	24	75.6464	5.4412	0.00172	0.00240	**0.0361**
FL-SMIA	71	75.0439	5.4590	**0.00170**	0.00260	0.0370
FL-SMIA*	25	73.5690	5.4700	0.00182	0.00256	0.0372
D-FL-SMIA	71	71.6386	5.5627	0.01227	0.01432	0.0967
MD-FL-SMIA	76	73.7225	5.5042	0.02953	0.03165	0.1408
MD-FL-SMIA-2	86	73.2885	5.5165	0.01420	0.01576	0.0985
M-FL-SMIA	86	76.6836	5.4170	0.03453	0.03633	0.1578
FL-SMIA-RBM	71	59.6911	5.8563	0.00395	0.00554	0.0565

Table 8.4 The best one day ahead prediction result for NQO

Networks	Hidden No. or Order	RP	AV	MSE Training	MSE Testing	MAE
FLNN	2	**72.2967**	**12.6817**	0.00324	0.00401	0.0455
MLP	8	68.3140	12.9159	0.00151	0.00215	0.0309
SMIA	18	71.8887	12.7031	0.00130	0.00220	0.0298
FL-SMIA	43	70.4152	12.7885	0.00140	0.00230	0.0304
FL-SMIA*	16	69.3693	12.8467	**0.00120**	**0.00210**	**0.0295**
D-FL-SMIA	43	66.1349	13.0263	0.00522	0.00693	0.0626
MD-FL-SMIA	48	65.5493	13.0570	0.01948	0.03018	0.1299
MD-FL-SMIA-2	58	67.9004	12.9316	0.02164	0.01746	0.1043
M-FL-SMIA	58	71.1504	12.7488	0.04152	0.04898	0.1803
FL-SMIA-RBM	43	56.4982	13.4893	0.00270	0.00354	0.0415

Table 8.5 The best one day ahead prediction result for NQC

Networks	Hidden No. or Order	RP	AV	MSE Training	MSE Testing	MAE
FLNN	2	**68.9098**	**12.2235**	0.00494	0.00573	0.0569
MLP	8	66.1522	12.3757	0.00237	**0.00296**	0.0381
SMIA	21	67.1120	12.3184	**0.00210**	0.00320	0.0379
R-SMIA FL-SMIA	49	66.6952	12.3434	0.00212	0.00330	0.0388
FL-SMIA*	20	66.3961	12.3604	0.00214	0.00306	**0.0373**
D-FL-SMIA	49	65.0047	12.4586	0.01301	0.01228	0.0840
MD-FL-SMIA	54	66.3828	12.3777	0.00880	0.00798	0.0692
MD-FL-SMIA2	64	66.3654	12.3787	0.00799	0.01035	0.0790
M-FL-SMIA	64	66.6946	12.3591	0.01906	0.02178	0.1120
FL-SMIA-RBM	49	53.4892	13.0521	0.00435	0.00524	0.0541

Table 8.6 The best one day ahead prediction result for DJO

Networks	Hidden No. or Order	RP	AV	MSE Training	MSE Testing	MAE
FLNN	2	**72.4772**	**10.5783**	0.00344	0.00505	0.0524
MLP	8	67.7744	10.8177	0.00215	0.00327	0.0409
SMIA	19	62.4502	11.0668	0.00150	0.00240	0.0332
FL-SMIA	47	68.6866	10.7724	0.00124	0.00241	0.0317
FL-SMIA*	11	67.9594	10.8080	**0.00120**	**0.00215**	**0.0314**
D-FL-SMIA	47	66.7245	10.8722	0.01848	0.02419	0.1257
MD-FL-SMIA	52	70.3455	10.6910	0.00397	0.00608	0.0571
MD-FL-SMIA2	62	67.1006	10.8540	0.00812	0.01126	0.0797
M-FL-SMIA	62	69.5321	10.7328	0.01925	0.02887	0.1319
FL-SMIA-RBM	47	55.2311	11.3684	0.00227	0.00349	0.0425

Table 8.7 The best one day ahead prediction result for DJC

Networks	Hidden No. or Order	RP	AV	MSE Training	MSE Testing	MAE
FLNN	2	69.3698	10.7969	0.00272	0.00395	0.0462
MLP	8	64.7555	11.0213	0.00213	0.00314	0.0403
SMIA	19	62.5851	11.1218	0.00130	**0.00229**	0.0323
FL-SMIA	47	**70.2760**	**10.7459**	0.00120	0.00230	**0.0312**
FL-SMIA*	17	65.6177	10.9742	**0.00111**	0.00240	0.0317
D-FL-SMIA	47	67.5880	10.8873	0.00674	0.00833	0.0703
MD-FL-SMIA	52	65.9437	10.9680	0.00688	0.01235	0.0856
MD-FL-SMIA2	62	69.7390	10.7778	0.00814	0.01099	0.0814
M-FL-SMIA	62	68.2698	10.8531	0.01116	0.01322	0.0927
FL-SMIA-RBM	47	56.3057	11.3912	0.00214	0.00332	0.0416

Table 8.8 The best one day ahead prediction result for OIL

Networks	Hidden No. or Order	RP	AV	MSE Training	MSE Testing	MAE
FLNN	2	77.8776	20.3893	0.00633	0.00712	0.0651
MLP	8	78.9114	20.2407	0.00340	0.00420	0.0497
SMIA	10	70.1922	16.7822	0.00035	0.00030	0.0120
FL-SMIA	86	75.9697	16.6670	**0.00029**	**0.00025**	**0.0103**
FL-SMIA*	42	**81.2012**	26.6794	0.00189	0.00221	0.0366
D-FL-SMIA	86	70.0492	**15.6057**	0.03973	0.03811	0.1450
MD-FL-SMIA	91	73.2164	21.0230	0.00748	0.00716	0.0664
MD-FL-SMIA2	101	74.4490	20.8611	0.01330	0.01271	0.0917
M-FL-SMIA	101	77.3883	20.4586	0.03283	0.03895	0.1665
FL-SMIA-RBM	86	60.9979	22.4291	0.00302	0.00373	0.0464

Table 8.9 The best one day ahead prediction result for GOLD

Networks	Hidden No. or Order	RP	AV	MSE Training	MSE Testing	MAE
FLNN	2	76.2846	32.3896	0.00565	0.00589	0.0585
MLP	4	76.0497	32.4344	0.00418	0.00445	0.0504
SMIA	11	70.6151	15.7535	**0.00011**	0.00007	0.0060
FL-SMIA	56	**79.4514**	**15.5541**	0.00012	**0.00006**	**0.0058**
FL-SMIA*	31	67.9489	38.4523	0.00260	0.00295	0.0404
D-FL-SMIA	56	74.4163	20.8654	0.01011	0.01221	0.0886
MD-FL-SMIA	61	74.5336	32.7184	0.02071	0.02406	0.1215
MD-FL-SMIA2	71	73.1978	32.9620	0.01785	0.01937	0.1004
M-FL-SMIA	71	74.2437	32.7718	0.01499	0.01424	0.0961
FL-SMIA-RBM	56	61.0485	34.9100	0.00352	0.00362	0.0439

Table 8.10 to table 8.12 including the heuristic parameters for each model on each data set to produce the results for each model.

Table 8.10 Table of Learning rate values that have been used to predict results for one day ahead prediction

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD
FLNN	0.1	0.1	0.1	0.4	0.4	0.4	0.03	0.1	0.1
MLP	0.1	0.1	0.1	0.1	0.1	0.4	0.4	0.4	0.4
SMIA	0.4	0.4	0.1	0.4	0.01	0.4	0.4	0.1	0.4
FL-SMIA	0.4	0.1	0.03	0.1	0.1	0.1	0.1	0.1	0.1
FL-SMIA*	0.1	0.03	0.01	0.4	0.01	0.4	0.1	0.1	0.03
D-FL-SMIA	0.03	0.1	0.03	0.01	0.03	0.01	0.1	0.01	0.4
MD-FL-SMIA	0.01	0.01	0.01	0.03	0.03	0.03	0.03	0.03	0.01
MD-FL-SMIA-2	0.04	0.03	0.01	0.03	0.1	0.04	0.04	0.01	0.01
M-FL-SMIA	0.1	0.03	0.04	0.04	0.04	0.03	0.03	0.04	0.4
FL-SMIA-RBM	0.1	0.03	0.1	0.1	0.1	0.03	0.4	0.1	0.4

Table 8.11 Table of Momentum values that have been used to predict results for one day ahead prediction

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD
FLNN	0.6	0.01	0.01	0	0.01	0.4	0.4	0.4	0.4
MLP	0.1	0.03	0.01	0.01	0.01	0.01	0.01	0.4	0.3
SMIA	0.1	0.01	0.4	0	0.1	0.01	0.03	1	0.1
FL-SMIA	0.1	0.6	0.6	0.6	0.6	0.01	0.4	0.4	0.03
FL-SMIA*	0.1	0.3	0.6	0.01	0.1	0.01	0.03	0.4	0.03
D-FL-SMIA	0.01	0.1	0.01	0.4	0.1	0.01	0.4	0.01	0.01
MD-FL-SMIA	0.01	0.4	0.01	0.4	0.1	0.4	0.03	0.4	0.6
MD-FL-SMIA-2	0.01	0.6	0.03	0.1	0.03	0.01	0.6	0.6	0.6
M-FL-SMIA	0.01	0.4	0.1	0.4	0.1	0.4	0.4	0.6	0.03
FL-SMIA-RBM	0.6	0.6	0.4	0.1	0.01	0.4	0.4	0.6	0.01

Table 8.12 Table of decay rate values that have been used to predict results for one day ahead prediction

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD
FLNN	0.0001	0.01	0.001	0.01	0.01	0.0001	0.1	0.001	0.001
MLP	0.0001	0.1	0.0001	0.1	0.1	0.01	0.0001	0.01	0.0001
SMIA	0.01	0.0001	0.001	0.001	0.0001	0.0001	0.0001	0.0001	0.001
FL-SMIA	0.01	0.001	0.1	0.0001	0.0001	0.01	0.01	0.0001	0.1
FL-SMIA*	0.0001	0.0001	0.001	0.0001	0.0001	0.0001	0.0001	0.0001	0.001
D-FL-SMIA	0.0001	0.0001	0.01	0.1	0.01	0.01	0.1	0.0001	0.01
MD-FL-SMIA	0.0001	0.01	0.1	0.0001	0.1	0.001	0.0001	0.01	0.01
MD-FL-SMIA-2	0.01	0.01	0.1	0.01	0.0001	0.001	0.001	0.0001	0.0001
M-FL-SMIA	0.01	0.1	0.0005	0.0005	0.001	0.001	0.1	0.001	0.01
FL-SMIA-RBM	0.01	0.001	0.01	0.1	0.01	0.1	0.001	0.1	0.1

Table 8.13 Training times for running a search with 10 models and 9 data-sets over a grid with ($9 \times 5 \times 5 \times 5$) parameter combinations (not all apply to all models, see section 2.3.4), with 50 simulations of 80 epochs each.

Data	FLNN	MLP	SMIA	FL-SMIA	FL-SMIA*	D-FL-SMIA	MD-FL-SMIA	MD-FL-SMIA-2	M-FLSMIA	FL-SMIA-RBM
US/UK	2:19:35	3:25:19	2:17:18	3:19:29	3:15:19	3:56:49	3:50:42	3:56:48	3:49:55	5:39:51
US/EU	2:20:14	3:29:02	2:18:13	3:20:07	3:22:17	3:59:47	3:49:42	3:54:47	3:48:56	5:28:51
JP/US	2:20:29	3:28:47	2:18:39	3:20:14	3:17:24	3:59:12	3:39:42	3:53:32	3:58:59	5:45:23
NQC	2:19:45	3:30:29	2:18:18	3:19:43	3:15:48	3:49:44	3:43:54	3:54:59	3:26:58	4:26:43
DJC	2:19:50	3:29:26	2:18:49	3:19:40	3:12:41	3:46:43	3:36:46	3:48:47	3:26:02	5:20:34
OIL	3:24:10	4:48:35	3:42:11	4:26:35	4:22:45	4:49:31	4:43:39	4:54:46	3:35:00	5:55:46
GOLD	3:23:15	4:49:20	3:52:17	4:25:28	4:20:24	4:45:38	4:38:40	4:46:48	3:31:56	5:53:45

Table 8.13 illustrated the time that took to train each model when using each one of the data-sets for 50 simulations (each simulation runs 80 epochs), the time is also based on the values range for all parameters which have been listed in chapter 3 section 3.6. The training time that listed in table 8.13 is approximately similar for predicting one day and five days ahead. The FLNN model required less training time comparing to all other models. While the FL-SMIA-RBM model needs more time than required for training any other model.

8.1.1 The Comparison and Analysis of the Prediction Results for One Day Ahead

This section focuses on comparing and analysing the results that have been produced by all the networks used in this research using the financial data which has illustrated in Table 3.1.

The comparison between the results for all the networks, as well as the analysis results, have been listed with more details in the following:

1. **Hidden units:**

   The prediction results that illustrated in the tables 8.1 to 8.9 indicated that the FLNN with using only second order has outperformed all the networks for six data-sets. The MLP network , which used a single hidden layer are produced best prediction results for the RP by using 8 hidden units for the prediction of all data-sets except with GOLD data as it used only 4 hidden units.

   It is clearly shown in table 8.14 that the number of hidden units which created by SMIA networks are between 10 to 24 hidden units. While the FL-SMIA model created a number of hidden units which are from 40 to 86 hidden units. When comparing the number of hidden units for the FL-SMIA model with the other networks that used the immune algorithm (SMIA, and FL-SMIA*), the FL-SMIA

Table 8.14 The number of hidden units that generated by different versions of SMIA models for different data-sets.

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD
SMIA	12	19	24	18	21	19	19	10	11
FL-SMIA	40	59	71	43	49	47	47	86	56
FL-SMIA*	16	32	25	16	20	11	17	42	31
D-FL-SMIA	40	59	71	43	49	47	47	86	56
MD-FL-SMIA	45	64	76	48	54	52	52	91	61
MD-FL-SMIA-2	55	74	86	58	64	62	62	101	71
M-FL-SMIA	55	74	86	58	64	62	62	101	71
FL-SMIA-RBM	40	59	71	43	49	47	47	86	56

network created the largest number of hidden units than the hidden units created by SMIA and FL-SMIA* networks.

For the other networks that started from FL-SMIA ended with FL-SMIA-RBM, each network uses one hidden layer but creates different numbers of hidden units because of using the immune algorithm and different type of the data-set.

For the deeper FL-SMIA network (D-FL-SMIA) hidden units number, on the first hidden layer the D-FL-SMIA network created the same number of hidden units that created by FL-SMIA model then it used a fixed number of hidden units for each hidden layers as explained in section 6.1.

While with the MD-FL-SMIA network, the number of hidden units increased with five hidden units than the number that created with the FL-SMIA model for all the data-sets due to the use of the extra hidden units equal to 5 units. after creating the hidden units in the first hidden layer (SMIA layer). While using the 15 hidden units with the hidden units that created in the first hidden layer lead to raise the number of hidden units to reach the highest number of hidden units used in this research when performing the MD-FL-SMIA2 and M-FL-SMIA networks. The FL-SMIA-RBM network predicted using the same hidden units number of FL-SMIA network in the first hidden layer then it reduced to 10 hidden units in the second hidden layer.

2. **Relative Profit (RP)**

   The Relative Profit (RP) results indicated that the FLNN network predicted highest RP values than all other networks for most data-sets. However, the FL-SMIA and the FL-SMIA* networks outperformed the FLNN network on the prediction of data DJC, OIL, and GOLD.

The comparison results between the proposed networks and the MLP networks proved that for exchange rate data-sets, the D-FL-SMIA network outperformed the MLP network and all other multilayer networks, followed by the value (RP) that produced by FL-SMIA network when using the US/UK exchange rate data. Despite that the proposed FL-SMIA model outperformed the MLP network and all the other proposed networks when used US/EU data, the prediction results of the SMIA network still indicated highest profits than all other multilayer networks for US/EU data. The proposed network (M-FL-SMIA) network improved the performance of FL-SMIA and the multilayer networks, the M-FL-SMIA network reached the highest profits than all other multilayer networks for JP/US exchange rate.

For the prediction of stock prices data-sets, the RP results indicated that the highest RP values are predicted by the networks which used the immune learning algorithm. The proposed network (M-FL-SMIA) outperforms the FL-SMIA and all other multilayer networks except the SMIA for the RP value of NQO. while the stock prices data NQC, the SMIA network followed by the FL-SMIA model outperforming all multilayer networks on RP values. The MD-FL-SMIA reached the highest RP value than all multilayer networks to improve the performance of multilayer networks when predicted the DJO. While the MD-FL-SMIA2 produced the second highest RP value after the FL-SMIA network with the DJC data.

The rest of the data (OIL, and GOLD) have been predicted successfully to reach the highest RP values than all other networks when using the networks FL-SMIA*, and FL-SMIA respectively.

The FL-SMIA-RBM network which represents the last proposed version of the FL-SMIA network predicted unexpected results for financial data, as the FL-SMIA-RBM network produced the lowest RP values than all other networks for all financial data-sets.

3. **Annualised Volatility (AV):**

   The risk of investment is measured by the Annualised Volatility (AV), the desired value is the lowest predicted AV value for financial data. As shown in tables 8.1 to 8.9, the AV results are unsteady as it depends on the data type.

   For the exchange rate data, the FLNN reduced the trading risk by produced the lowest AV values than all other networks when used the data of US/UK, US/EU, and JP/US. The comparison between the multi-layer networks and the proposed network illustrated that the D-FL-SMIA network outperformed the M-FL-SMIA and all multilayer networks by reduced the annualised volatility value when forecasts the data of US/UK. While for US/EU data, the SMIA network followed by the FL-SMIA model produced the lower AV values than all other multi-layer networks. The M-FL-SMIA network decreased the AV values of the proposed multilayer networks when predicted the JP/US data. Consequently, the results of AV means that use the immune algorithm improves the performance of the multi-layer network on reducing the investment risk.

Regarding the stock price data, the comparison results of annualised volatility (AV) indicated that the FLNN network outperformed all other networks when predicted the data of NQO, NQC, and DJO. However, the FLNN network competed with immune multi-layer networks. The FL-SMIA network outperformed all other networks when predicted the AV value for DJC stock prices data.

For the commodity prices data (OIL, and GOLD), the D-FL-SMIA, and FL-SMIA networks predicted the lowest values of AV than all other networks with 15.6057, and 15.5541 respectively.

4. **MSE-Training:**

   The results of MSE for the training phase and testing phase, which listed in tables 8.1 to 8.9, indicated that most of the networks decreased the error values for all the data-sets. Nevertheless, the proposed FL-SMIA network and other networks outperformed the proposed networks (D-FL-SMIA, MD-FL-SMIA, MD-FL-SMIA2, M-FL-SMIA, and FL-SMIA-RBM) for the results of the MSE measure.

   For the MSE-Training, the comparison results between the multi-layer networks and the FLNN network illustrated that the SMIA network outperforming the FLNN network and all other networks when predicting the US/UK, NQC, and GOLD data. While the FL-SMIA model reduced the error values of MSE-Training by producing the lower MSE-Training values than all other networks when using the US/EU, JP/US, and OIL data-sets. Furthermore, the FL-SMIA* model outperformed all other networks with error values of the MSE-Training phase when predicting the NQO, DJO, and DJC data-sets with producing the low values of error (0.00121, 0.00120, 0.00111) respectively.

5. **MSE-Testing:**

   The MSE-Testing results indicated that although the results demonstrate that the proposed FL-SMIA network predicted the lowest error value than all other networks when used the US/UK, OIL, and GOLD data-sets, the MLP network still produced the lowest error than all other networks when predicting the JP/US and NQC data-sets. It is good to notice that in the testing phase the proposed FL-SMIA network produced the lowest MSE-Testing value (0.000067) when predicting the GOLD data, which considered as the lowest error value has been predicted in this research.

The other observation is the use of the immune learning algorithm helped the multi-layer networks to predict lower error (MSE) than the FLNN network. The SMIA network outperformed all other networks for the MSE-Testing results when using US/EU and DJC data-sets. Also, the results indicated that using the FL-SMIA* network on the prediction of the NQO and DJO data-sets improved the network performance to reducing the MSE values for the training and testing phases. When comparing the MSE-Testing results for the proposed network FL-SMIA with the other proposed networks (D-FL-SMIA, MD-FL-SMIA, MD-FL-SMIA2, M-FL-SMIA, and FL-SMIA-RBM), it could be seen that in most cases of the MSE-Testing results that the FL-SMIA network and the FL-SMIA-RBM network outperformed all other proposed networks.

6. **Mean Absolute Error (MAE):**

   Regarding the Mean Absolute Error (MAE) results, on one hand, the FL-SMIA network outperformed all other networks when forecasted low MAE values (0.0471,0.0312, 0.0103, and 0.0058) using the data of US/UK, DJC, OIL, and GOLD respectively. On the other hand, the FL-SMIA* network produced the lowest MAE values than all other networks when using the NQO, NQC, and DJO data-sets. The SMIA network has also outperformed all other networks for the results of MAE when using the exchange rate data (US/EU, and JP/US).

   For the proposed networks (D-FL-SMIA, MD-FL-SMIA, MD-FL-SMIA2, M-FL-SMIA, and FL-SMIA-RBM), Although the results proved that the FL-SMIA-RBM network produced better results for MAE than the performance of other proposed networks. However, the FL-SMIA-RBM network still on jostle with the D-FL-SMIA network.

8.1.2 The Comparison of the Average Results

The average results of the Relative Profit (RP) for one day ahead prediction for all networks are illustrated in table 8.15. The comparison between all the networks indicated that the RP average results of the proposed networks FL-SMIA and M-FL-SMIA is lower than the RP average results of the FLNN network. While the comparison results between the multi-layer networks showed that the proposed networks FL-SMIA, and M-FL-SMIA outperformed all other multilayer networks. The proposed networks FL-SMIA and M-FL-SMIA produced the higher values of average RP with the values (73.268, and 72.743) respectively.

Table 8.15 The best results of Relative Profit (RP) and the average for all networks for the prediction for one day ahead.

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD	Average
FLNN	**81.107**	**80.037**	**79.716**	**72.297**	**68.910**	**72.477**	69.370	77.878	76.285	**75.342**
MLP	72.108	73.238	74.500	68.314	66.152	67.774	64.756	78.911	76.050	71.311
SMIA	73.715	78.126	75.646	71.889	67.112	62.450	62.585	70.192	70.615	70.055
FL-SMIA	76.225	76.648	75.044	70.415	66.695	68.687	**70.276**	75.970	**79.451**	**73.268**
FL-SMIA*	73.111	72.807	73.569	69.369	66.396	67.959	65.618	**81.201**	67.949	70.887
D-FL-SMIA	77.153	70.764	71.639	66.135	65.005	66.724	67.588	70.049	74.416	69.941
MD-FL-SMIA	73.538	75.090	73.722	65.549	66.383	70.346	65.944	73.216	74.534	70.925
MD-FL-SMIA-2	74.006	75.666	73.289	67.900	66.365	67.101	69.739	74.449	73.198	71.301
M-FL-SMIA	75.695	75.034	76.684	71.150	66.695	69.532	68.270	77.388	74.244	72.743
FL-SMIA-RBM	59.449	60.685	59.691	56.498	53.489	55.231	56.306	60.998	61.049	58.155

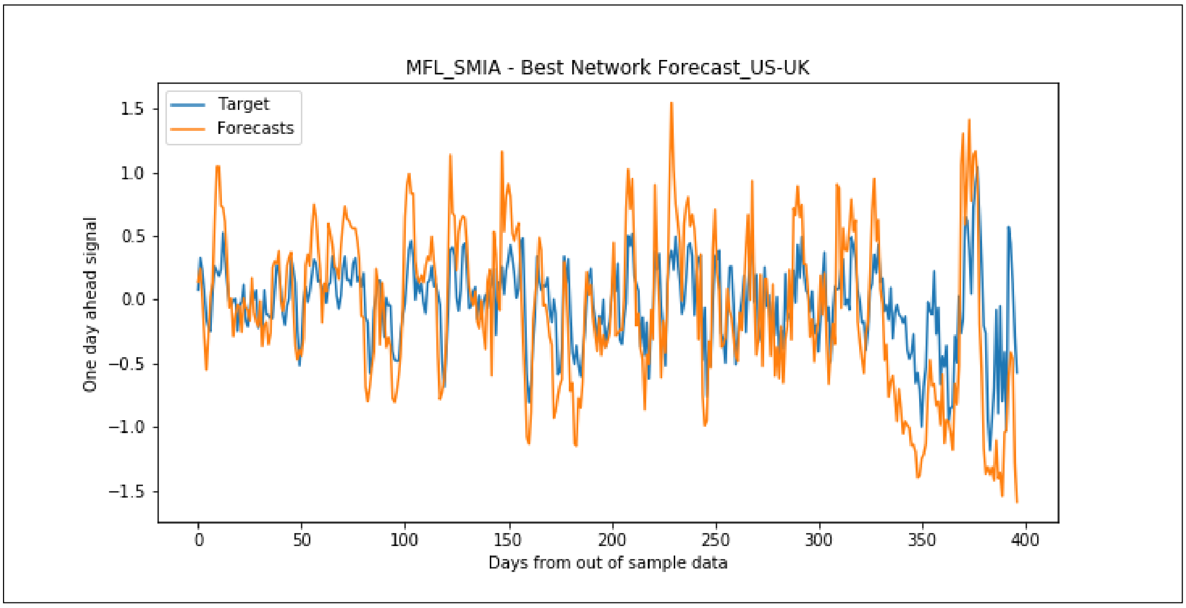


Fig. 8.1 The forecasting for one day ahead prediction using the M-FL-SMIA network.

Figure 8.1 illustrated an example for proposed network prediction (M-FL-SMIA) for the US-UK data, the figure also shows that in the most the forecasted signal followed the target signal. In other words, it proves the ability of the M-FL-SMIA network to learn the behaviour of financial data, M-FL-SMIA network could be considered as a promising model on financial prediction area.

Table 8.16 The best results for the Annualised Volatility (AV) and the average for all networks for the prediction for one day ahead.

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD	Average
FLNN	**4.1191**	**4.3898**	**5.3227**	**12.6817**	**12.2235**	**10.5783**	10.7969	20.3893	32.3896	12.5434
MLP	4.3763	4.5828	5.4748	12.9159	12.3757	10.8177	11.0213	20.2407	32.4344	12.6933
SMIA	4.3328	4.4428	5.4412	12.7031	12.3184	11.0668	11.1218	16.7822	15.7535	**10.4403**
FL-SMIA	4.6260	4.4865	5.4590	12.7885	12.3434	10.7724	**10.7459**	16.6670	**15.5541**	**10.3825**
FL-SMIA*	4.3487	4.5900	5.4700	12.8467	12.3604	10.8080	10.9742	26.6794	38.4523	14.0589
D-FL-SMIA	4.2403	4.6546	5.5627	13.0263	12.4586	10.8722	10.8873	**15.6057**	20.8654	10.9081
MD-FL-SMIA	4.3431	4.5370	5.5042	13.0570	12.3777	10.6910	10.9680	21.0230	32.7184	12.8022
MD-FL-SMIA-2	4.3302	4.5206	5.5165	12.9316	12.3787	10.8540	10.7778	20.8611	32.9620	12.7925
M-FL-SMIA	4.2827	4.5386	5.4170	12.7488	12.3591	10.7328	10.8531	20.4586	32.7718	12.6847
FL-SMIA-RBM	4.6794	4.6537	5.8563	13.4893	13.0521	11.3684	11.3912	22.4291	34.9100	13.5366

As shown in table 8.16, that for the investment risk, the results of average Annualised Volatility (AV) proved that the FL-SMIA model reduced the investment risk by produced the lowest AV value than all other networks including the FLNN model.

Table 8.17 The best results for the MSE-Testing and the average for all networks for one day ahead prediction.

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD	Average
FLNN	0.00992	0.00841	0.01247	0.00401	0.00573	0.00505	0.00395	0.00712	0.005891	0.00695
MLP	0.00384	0.00294	**0.00238**	0.00215	**0.00296**	0.00327	0.00314	0.00420	0.004453	0.00326
SMIA	0.00382	**0.00291**	0.00240	0.00220	0.00320	0.00240	**0.00229**	0.00030	0.000071	**0.00218**
FL-SMIA	**0.00381**	0.00340	0.00260	0.00230	0.00330	0.00241	0.00230	**0.00025**	**0.000067**	**0.00227**
FL-SMIA*	0.00442	0.00348	0.00256	**0.00210**	0.00306	**0.00215**	0.00240	0.00221	0.002950	0.00283
D-FL-SMIA	0.02504	0.01737	0.01432	0.00693	0.01228	0.02419	0.00833	0.03811	0.012208	0.01764
MD-FL-SMIA	0.03701	0.03238	0.03165	0.03018	0.00798	0.00608	0.01235	0.00716	0.024060	0.02098
MD-FL-SMIA-2	0.02092	0.01215	0.01576	0.01746	0.01035	0.01126	0.01099	0.01271	0.019368	0.01455
M-FL-SMIA	0.03508	0.02407	0.03633	0.04898	0.02178	0.02887	0.01322	0.03895	0.014237	0.02906
FL-SMIA-RBM	0.00833	0.00795	0.00554	0.00354	0.00524	0.00349	0.00332	0.00373	0.003617	0.00497

Table 8.17 includes the average results of MSE-Testing for one day ahead prediction, the results showed that all networks reduced the values of MSE-Testing when forecasted all the data-sets. However, the comparison between the average results of the networks demonstrates that the SMIA network and the proposed models (FL-SMIA and FL-SMIA*) produced the lowest average of MSE-Testing values than all other networks which have been used in this research including the FLNN network.

The average results of MSE-Testing proved that the SMIA network is outperforming all networks with the value 0.00218, followed by the proposed networks FL-SMIA, and FL-SMIA* which outperformed all other networks with the average of the MSE-Testing results (0.00227, and 0.00283) respectively. Focusing on the average of the MSE-Testing results it could be noticing that the proposed network FL-SMIA-RBM outperformed

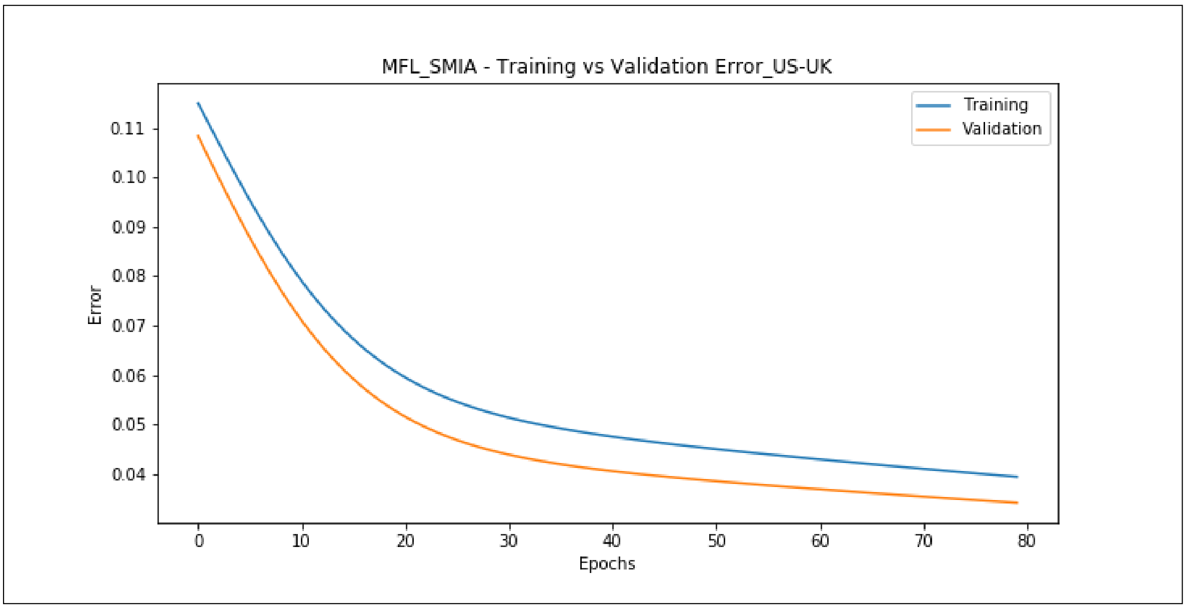


Fig. 8.2 The training error and validation error for one day ahead prediction using M-FL-SMIA network.

the FLNN withe the average of MSE-Testing value (0.00497 vs 0.00695). Also, the proposed network FL-SMIA-RBM produced the lowest average of the MSE-Testing value than the proposed networks (D-FL-SMIA, MD-FL-SMIA, and MD-FL-SMIA2).

Figure 8.2 demonstrates the prediction error of the M-FL-SMIA network when using the US-UK data. The figure shows that the training error and the validation error are closer to each other and they are near to the zero value. All figures that relevant to the prediction error for the networks have been included in Appendix A.

The results and figures proved clear information about the ability of the proposed M-FL-SMIA network for financial prediction.

8.2 The Prediction for Five Days Ahead

In this section, the results for the prediction for five days ahead will be shown through table 8.18 to 8.26.

The experimental results for five days ahead prediction using the proposed network (FL-SMIA) and several versions of developing FL-SMIA have proved better results than the results of one day ahead prediction.

To investigate and analyse the prediction results for all networks that proposed in this research, the experimental results for all networks that have been used in this research that used all financial time series that showed in Table 3.1, will be included in this section.

The results of 50 simulations produced by the neural network models using the nine data-sets have been listed in this research. Each simulation includes 80 epochs, while each epoch uses a combination of parameters in a grid search as listed in section 3.6. The results have been selected based on the best RP values produced from each model that has been used in this research.

Each table below including the number of hidden units that have been used with each model structure for each data-sets. For the FLNN model, the second-order of inputs have been used, which refers to the probability of inputs products to produce the results. While the proposed models used the second order of inputs and the hidden units that created by the immune algorithm. In addition to the RP, AV, and MAE, the MSE results including MSE Training results and MSE-Testing results.

Table 8.27 to table 8.29 are including the heuristic parameters for each model on each data set to find the best results for each model.

The details related to all results in tables 8.18 to 8.26 including the comparison between the results and the analysis for the prediction results for five days ahead have been illustrated in the following subsection 8.2.1

Table 8.18 The best results for five days ahead prediction US/UK

Networks	Hidden No. or Order	RP	AV	MSE		MAE
				Training	Testing	
FLNN	2	91.52893	15.47144	0.00606	0.00739	0.06512
MLP	8	91.18500	15.49770	0.00213	0.00264	0.03770
SMIA	12	88.40330	15.85660	0.00106	**0.00158**	0.02832
FL-SMIA	40	**92.02060**	15.65030	**0.00091**	0.00204	**0.02830**
FL-SMIA*	16	90.76450	15.55320	0.00100	0.00220	0.02930
D-FL-SMIA	40	90.01430	15.67080	0.01387	0.01324	0.09359
MD-FL-SMIA	45	91.47115	15.47915	0.04681	0.04281	0.17791
MD-FL-SMIA2	55	91.74532	**15.44247**	0.03580	0.02818	0.13846
M-FL-SMIA	55	90.80453	15.56753	0.00387	0.00514	0.05233
FL-SMIA-RBM	40	86.89928	16.06299	0.00393	0.00517	0.05231

Table 8.19 The best results for five days ahead prediction US/EU

Networks	Hidden No. or Order	RP	AV	MSE Training	MSE Testing	MAE
FLNN	2	92.45180	15.56260	0.00870	0.01046	0.07740
MLP	8	91.50050	15.68840	0.00224	**0.00173**	**0.03120**
SMIA	19	91.92650	15.62360	0.00184	0.00264	0.03280
FL-SMIA	59	92.49530	15.53630	**0.00164**	0.00319	0.03420
FL-SMIA*	32	92.47520	15.53940	0.00172	0.00310	0.03370
D-FL-SMIA	59	93.34943	15.42257	0.06314	0.05695	0.19397
MD-FL-SMIA	64	91.66700	15.68291	0.03708	0.03349	0.15307
MD-FL-SMIA2	74	92.58482	15.54201	0.03861	0.03501	0.15903
M-FL-SMIA	74	**93.74462**	**15.36008**	0.04496	0.03838	0.15676
FL-SMIA-RBM	59	81.37685	17.09618	0.00662	0.00871	0.06974

Table 8.20 The best results for five days ahead prediction JP/US

Networks	Hidden No. or Order	RP	AV	MSE Training	MSE Testing	MAE
FLNN	2	86.98499	16.81314	0.01027	0.01358	0.09241
MLP	8	83.33750	17.29550	0.00199	0.00261	0.03902
SMIA	24	86.23720	16.89820	0.00153	**0.00236**	**0.03640**
FL-SMIA	71	**87.49160**	**16.71910**	**0.00146**	0.00254	0.03670
FL-SMIA*	25	87.05390	16.78210	0.00160	0.00313	0.03990
D-FL-SMIA	71	83.30714	17.32141	0.03381	0.03558	0.15845
MD-FL-SMIA	76	84.06701	17.21941	0.00636	0.00925	0.07435
MD-FL-SMIA2	86	85.94410	16.96078	0.02922	0.03194	0.15038
M-FL-SMIA	86	85.84617	16.97451	0.02789	0.03256	0.15153
FL-SMIA-RBM	71	77.80853	18.01588	0.09355	0.09822	0.28542

Table 8.21 The best results for five days ahead prediction NQO

Networks	Hidden No. or Order	RP	AV	MSE Training	MSE Testing	MAE
FLNN	2	88.16550	35.16634	0.00422	0.00468	0.05060
MLP	8	83.21360	36.42510	0.00296	0.00305	0.03980
SMIA	18	86.56590	35.57702	**0.00077**	0.00141	0.02410
FL-SMIA	43	86.62570	35.56130	0.00098	0.00150	0.02470
FL-SMIA*	16	87.27870	35.38960	0.00079	**0.00140**	**0.02400**
D-FL-SMIA	43	83.01807	36.48495	0.01106	0.01448	0.08870
MD-FL-SMIA	48	81.66096	36.81188	0.01200	0.01313	0.08704
MD-FL-SMIA2	58	83.98674	36.24646	0.00737	0.00769	0.06766
M-FL-SMIA	58	**88.28911**	**35.13308**	0.00936	0.00752	0.06721
FL-SMIA-RBM	43	70.99768	39.11082	0.00335	0.00375	0.04604

Table 8.22 The best results for five days ahead prediction NQC

Networks	Hidden No. or Order	RP	AV	MSE Training	MSE Testing	MAE
FLNN	2	86.39704	35.75308	0.00527	0.00555	0.05542
MLP	8	80.77150	37.12470	0.00360	0.00385	0.04490
SMIA	21	87.25701	35.47790	0.00140	**0.00177**	0.02770
FL-SMIA	49	87.34070	35.73694	0.00119	0.00190	**0.02751**
FL-SMIA*	20	87.01610	35.54270	**0.00115**	0.00199	0.02820
D-FL-SMIA	49	85.04999	36.10657	0.02183	0.02163	0.12274
MD-FL-SMIA	54	86.30407	35.77776	0.02341	0.02320	0.12535
MD-FL-SMIA2	64	85.53857	35.97940	0.01854	0.01838	0.11259
M-FL-SMIA	64	**87.56939**	**35.43797**	0.03705	0.02956	0.14181
FL-SMIA-RBM	49	73.30163	38.83697	0.00361	0.00397	0.04556

Table 8.23 The best results for five days ahead prediction DJO

Networks	Hidden No. or Order	RP	AV	MSE Training	MSE Testing	MAE
FLNN	2	89.52173	30.72444	0.00244	0.00327	0.04085
MLP	8	80.39930	32.65290	0.00115	0.00130	0.02660
SMIA	19	84.46940	31.82670	0.00075	0.00136	0.02369
FL-SMIA	47	86.62530	31.36340	**0.00065**	0.00165	0.02370
FL-SMIA*	11	87.99470	31.05930	0.00069	**0.00122**	**0.02300**
D-FL-SMIA	47	88.68378	30.91655	0.00807	0.01059	0.07874
MD-FL-SMIA	52	89.46520	30.73750	0.00428	0.00533	0.05537
MD-FL-SMIA2	62	90.01112	30.61085	0.01362	0.01854	0.10823
M-FL-SMIA	62	**90.76172**	**30.43459**	0.01279	0.01682	0.10088
FL-SMIA-RBM	47	73.24815	33.97119	0.00535	0.00758	0.07005

Table 8.24 The best results for five days ahead prediction DJC

Networks	Hidden No. or Order	RP	AV	MSE Training	MSE Testing	MAE
FLNN	2	89.56439	30.75543	0.00275	0.00375	0.04438
MLP	8	84.37650	31.90650	0.00177	0.00230	0.03430
SMIA	19	86.46770	31.45310	**0.00006**	0.00155	0.02370
FL-SMIA	47	87.45370	31.23310	0.00066	0.00167	0.02350
FL-SMIA*	17	89.72310	30.71120	0.00064	**0.00120**	**0.02230**
D-FL-SMIA	47	90.22519	30.59969	0.02796	0.03073	0.14588
MD-FL-SMIA	52	88.10474	31.09269	0.00974	0.01435	0.09235
MD-FL-SMIA2	62	89.48640	30.77369	0.01063	0.01253	0.08979
M-FL-SMIA	62	**90.63058**	**30.50318**	0.00839	0.01295	0.08809
FL-SMIA-RBM	47	74.65478	33.80575	0.00450	0.00786	0.06611

Table 8.25 The best results for five days ahead prediction OIL

Networks	Hidden No. or Order	RP	AV	MSE		MAE
				Training	Testing	
FLNN	2	89.81445	61.52982	0.00628	0.00732	0.06718
MLP	8	89.72291	61.59024	0.00537	0.00626	0.06206
SMIA	10	**90.89164**	**60.80959**	0.00149	0.00186	0.01099
FL-SMIA	86	87.62945	62.93981	0.00210	0.00279	**0.01360**
FL-SMIA*	42	86.23575	63.74985	**0.00128**	**0.00178**	0.03220
D-FL-SMIA	86	86.24819	63.79760	0.02460	0.02487	0.12466
MD-FL-SMIA	91	86.35109	63.73456	0.01710	0.02033	0.11668
MD-FL-SMIA2	101	88.62042	62.30858	0.00943	0.01034	0.08039
M-FL-SMIA	101	90.33990	61.18063	0.01440	0.01752	0.10561
FL-SMIA-RBM	86	80.49572	67.11419	0.00642	0.00745	0.06812

Table 8.26 The best results for five days ahead prediction GOLD

Networks	Hidden No. or Order	RP	AV	MSE		MAE
				Training	Testing	
FLNN	2	**90.00925**	**101.34412**	0.01062	0.01222	0.08603
MLP	8	88.96170	102.29560	0.00536	0.00594	0.05755
SMIA	11	89.59479	101.72290	0.00860	0.00954	0.07595
FL-SMIA	56	89.49655	101.81230	0.00240	0.00271	**0.01334**
FL-SMIA*	31	83.75752	106.65943	**0.00210**	**0.00227**	0.03100
D-FL-SMIA	56	88.80650	102.43480	0.02717	0.02973	0.13522
MD-FL-SMIA	61	89.77235	101.56104	0.01201	0.01305	0.08991
MD-FL-SMIA2	71	88.78457	102.45450	0.01850	0.02417	0.12214
M-FL-SMIA	71	89.52910	101.78271	0.03879	0.04517	0.17751
FL-SMIA-RBM	56	84.64353	106.01466	0.00873	0.00998	0.07522

Table 8.27 Table of Learning rate values that have been used to predict results for five days ahead prediction.

Network	**US/UK**	**US/EU**	**JP/US**	**NQO**	**NQC**	**DJO**	**DJC**	**OIL**	**GOLD**
FLNN	0.03	0.03	0.1	0.4	0.1	0.4	0.4	0.01	0.1
MLP	0.1	0.1	0.1	0.1	0.1	0.4	0.4	0.4	0.4
SMIA	0.4	0.4	0.1	0.4	0.01	0.4	0.4	0.1	0.4
FL-SMIA	0.1	0.1	0.4	0.4	0.3	0.4	0.4	0.1	0.1
FL-SMIA*	0.1	0.03	0.01	0.4	0.01	0.4	0.1	0.1	0.03
D-FL-SMIA	0.01	0.03	0.01	0.03	0.01	0.01	0.01	0.03	0.01
MD-FL-SMIA	0.01	0.01	0.1	0.01	0.01	0.1	0.1	0.03	0.01
MD-FL-SMIA-2	0.03	0.01	0.01	0.03	0.03	0.01	0.03	0.1	0.01
M-FL-SMIA	0.3	0.1	0.03	0.1	0.03	0.04	0.1	0.1	0.03
FL-SMIA-RBM	0.1	0.4	0.03	0.03	0.1	0.1	0.1	0.1	0.1

Table 8.28 Table of momentum values that have been used to predict results for five days ahead prediction

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD
FLNN	0.6	0.4	0.1	0.1	0.1	0.4	0.01	0.4	0.6
MLP	0.1	0.03	0.01	0.01	0.01	0.01	0.01	0.4	0.3
SMIA	0.1	0.01	0.4	0	0.1	0.01	0.03	1	0.1
FL-SMIA	0.1	0.6	0.6	0.6	0.6	0.01	0.4	0.4	0.03
FL-SMIA*	0.1	0.3	0.6	0.01	0.1	0.01	0.03	0.4	0.03
D-FL-SMIA	0.1	0.01	0.03	0.4	0.03	0.01	0.4	0.01	0.04
MD-FL-SMIA	0.1	0.6	0.01	0.03	0.4	0.03	0.1	0.1	0.6
MD-FL-SMIA-2	0.1	0.4	0.03	0.1	0.03	0.03	0.01	0.01	0.6
M-FL-SMIA	0.01	0.4	0.4	0.1	0.6	0.6	0.4	0.1	0.1
FL-SMIA-RBM	0.6	0.1	0.4	0.6	0.4	0.4	0.01	0.4	0.1

Table 8.29 Table of decay rate values that have been used to predict results for five days ahead prediction

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD
FLNN	0.0001	0.0001	0.01	0.0001	0.0001	0.0001	0.001	0.0001	0.001
MLP	0.0001	0.1	0.0001	0.1	0.1	0.01	0.0001	0.01	0.0001
SMIA	0.01	0.0001	0.001	0.001	0.0001	0.0001	0.0001	0.0001	0.001
FL-SMIA	0.01	0.001	0.1	0.0001	0.0001	0.01	0.01	0.0001	0.1
FL-SMIA*	0.0001	0.0001	0.001	0.0001	0.0001	0.0001	0.0001	0.0001	0.001
D-FL-SMIA	0.001	0.0001	0.01	0.01	0.001	0.1	0.001	0.001	0.1
MD-FL-SMIA	0.0001	0.1	0.1	0.0001	0.01	0.1	0.0001	0.01	0.001
MD-FL-SMIA-2	0.01	0.0001	0.01	0.0001	0.0001	0.1	0.0005	0.1	0.01
M-FL-SMIA	0.1	0.0005	0.0005	0.0001	0.0005	0.01	0.0001	0.1	0.0005
FL-SMIA-RBM	0.001	0.001	0.001	0.01	0.1	0.001	0.0001	0.0001	0.01

8.2.1 Comparison and Analysis of Results of Five Days Ahead Prediction

In this section, the best prediction results for all financial time series that shown in Table 3.1 presented in order to compare and analysis the networks performance. The comparison between the results for all the networks, as well as the analysis results, have been listed with more details in the following:

1. **Hidden units:**

   The number of hidden units which created by SMIA networks for five days ahead prediction is the same number of hidden units which created by SMIA networks for one day ahead prediction 8.14. as also could be seen in tables 8.18 to 8.26. en The results indicated that the FLNN network has been reached a high RP values with a product of inputs using the same network's order that has been used for prediction of one day ahead to performed all the data-set, as well as the same numbers of hidden units have been used for prediction of five days ahead with multilayer networks (As explained in subsection 8.1.1).

2. **Relative Profit (RP):**

   In general, the Relative Profit (RP) results proved that using the unsupervised immune learning algorithm improves the performances of the multi-layer networks and leads to producing the highest RP values than all other networks including the FLNN network for all data-sets in Table 3.1 except the stock prices of GOLD. However, the last proposed version of the FL-SMIA network (FL-SMIA-RBM) network predicted the lowest results for all financial data comparing to all other networks results. AS shown in tables 8.18 to 8.26, that the proposed networks FL-SMIA and M-FL-SMIA outperformed all other networks to reach the highest results than all other networks in financial prediction domain.

   Comparing the RP results of the exchange rate data-sets for the multi-layer networks indicated that, the all networks which used the immune algorithm outperformed the MLP network when forecasted the US/EU data, as M-FL-SMIA reached the highest RP value (93.74462) than all other RP values that have been predicted in this research. For US/UK exchange rate data, the prediction RP results improved using FL-SMIA, MD-FL-SMIA, and MD-FL-SMIA2 networks to reach the results (92.02, 91.47, and 91.74) respectively. While when using the JP/US data the prediction for all multilayer immune networks except D-FL-SMIA and MD-FL-SMIA improved the performance of the MLP network.

For the prediction of stock price data-sets, the RP results proved that all networks which used the immune learning algorithm have improved the performance of the MLP network. as these networks produced the highest values of RP than the MLP network for the NQC, DJO, DJC data-sets. However, the MLP network outperformed the proposed networks (D-FL-SMIA and MD-FL-SMIA) when predicting the NQO data. The highest RP results for OIL data have been predicted by the SMIA, M-FL-SMIA networks which improve the performance of the MLP network. However, all other multilayer immune networks compete with the MLP network. The proposed network (MD-FL-SMIA) has outperformed all other multilayer networks when predicting the GOLD stock prices with the RP value (89.5291).

3. **Annualised Volatility (AV):**

   The Annualised Volatility (AV) values represent the risk of investments, the results of AV have been listed in the tables 8.18 to 8.26. As the desirable value is the lowest predicted AV value, the results AV for the exchange rate data, proved that the proposed networks MD-FL-SMIA2, M-FL-SMIA, and FL-SMIA successfully outperformed all other networks when predicted a lowest AV values when used the data of US/UK, US/EU, and JP/US respectively. consequently, using the proposed networks (MD-FL-SMIA2, M-FL-SMIA, and FL-SMIA) resulted in reducing the risk of investment when predicting for five days ahead with using exchange rate data. The comparison of AV results for the stock price data indicated that the M-FL-SMIA network has minimised investment risk as it produced the lowest values of AV than all other networks when predicted the data of NQO, NQC, DJO, and DJC. While the SMIA network outperforms all other networks when forecasted the OIL data. The AV values for the GOLD data showed that the FLNN network produced the lowest AV values than all other networks. This result leads to conclude that the proposed networks (M-FL-SMIA, D-FL-SMIA, and MD-FL-SMIA) outperformed all the multi-layer networks when predicted the AV values of DJC and GOLD data.

4. **MSE-Training:**

   For the results of MSE-Training, as the results in tables 8.18 to 8.26 presented that the comparison results between the multi-layer networks and the FLNN network showed that the multi-layer networks FL-SMIA, SMIA, and FL-SMIA* produced the lowest error values for MSE-Training than all other networks. The proposed network FL-SMIA outperformed all other networks using the US/UK, US/EU, JP/US, and DJO data-sets. While the SMIA network outperformed all other networks when predicted the NQO, and DJC data-sets. As well as, the proposed network FL-SMIA* produced the lowest values of MSE-Training than all other networks when using NQC, OIL, and GOLD data-sets.

Comparing the MSE-Training results between the proposed networks and the other networks showed that in general the networks which used the immune algorithm performed better than other multi-layer networks. However, the proposed networks (D-FL-SMIA, MD-FL-SMIA, MD-FL-SMIA2, and M-FL-SMIA) produced higher MSE-Training values than other networks.

5. **MSE-Testing:**

   The MSE-Testing results indicated that the proposed network FL-SMIA* produced the lowest errors than all other networks for five data-sets including NQO, DJO, DJC, OIL, and GOLD. While, SMIA network outperformed all networks when predicted the US/UK, JP/UK, and NQC data. Also, the MLP network produced the lowest MSE-Testing value than all other networks with US/EU exchange rate data. The Comparing of the MSE-Testing results between the proposed networks (D-FL-SMIA, MD-FL-SMIA, MD-FL-SMIA2, M-FL-SMIA, and FL-SMIA-RBM) showed that the FL-SMIA-RBM network outperformed the mentioned networks. However, the proposed networks (FL-SMIA, and FL-SMIA*) produced the lowest MSE-Testing values than the FL-SMIA-RBM network.

6. **Mean Absolute Error (MAE):**

   For the Mean Absolute Error (MAE) results, the proposed network FL-SMIA produced the lowest values of MAE than all other networks when using the US/UK, NQC, OIL, and GOLD data-sets. As well as, the proposed network FL-SMIA* outperformed all other networks when forecasted lower MAE values for NQO, DJO, and DJC data-sets. While the SMIA and MLP networks produced the lowest results than all other networks only when using the JP/US and US/EU data-sets respectively. The comparison results for the extended of FL-SMIA network (D-FL-SMIA, MD-FL-SMIA, MD-FL-SMIA2, M-FL-SMIA, and FL-SMIA-RBM), showed that the FL-SMIA-RBM network outperformed the other networks. However, the FL-SMIA-RBM network still competes with the FL-SMIA network.

8.2.2 The Comparison of the Average Results

The average results of the Relative Profit (RP) for the prediction of five days ahead for all networks have been listed in table 8.30. The results proved that the proposed network M-FL-SMIA outperformed all other networks includes the FLNN network (89.724 vs 89.383). As well as, the proposed networks FL-SMIA, FL-SMIA*, MD-FL-SMIA2, and M-FL-SMIA produced the highest values of average RP than MLP networks. Therefore, the proposed networks FL-SMIA, FL-SMIA*, MD-FL-SMIA2, and M-FL-SMIA could be considered as promising models in the financial prediction domain for five days ahead prediction.

Table 8.30 The best results of the Relative Profit (RP) for the five days ahead prediction with the average for all data-seta.

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD	Average
FLNN	91.529	92.452	86.985	88.165	86.397	89.522	89.564	89.814	**90.009**	**89.382**
MLP	91.185	91.501	83.338	83.214	80.772	80.399	84.377	89.723	88.962	85.941
SMIA	88.403	91.927	86.237	86.566	87.257	84.469	86.468	**90.892**	89.595	87.979
FL-SMIA	**92.021**	92.495	**87.492**	86.626	87.341	86.625	87.454	87.629	89.497	88.575
FL-SMIA*	90.765	92.475	87.054	87.279	87.016	87.995	89.723	86.236	83.758	88.033
D-FL-SMIA	90.014	93.349	83.307	83.018	85.050	88.684	90.225	86.248	88.807	87.634
MD-FL-SMIA	91.471	91.667	84.067	81.661	86.304	89.465	88.105	86.351	89.772	87.652
MD-FL-SMIA2	91.745	92.585	85.944	83.987	85.539	90.011	89.486	88.620	88.785	88.522
M-FL-SMIA	90.805	**93.745**	85.846	**88.289**	**87.569**	**90.762**	**90.631**	90.340	89.529	**89.724**
FL-SMIA-RBM	86.899	81.377	77.809	70.998	73.302	73.248	74.655	80.496	84.644	78.158

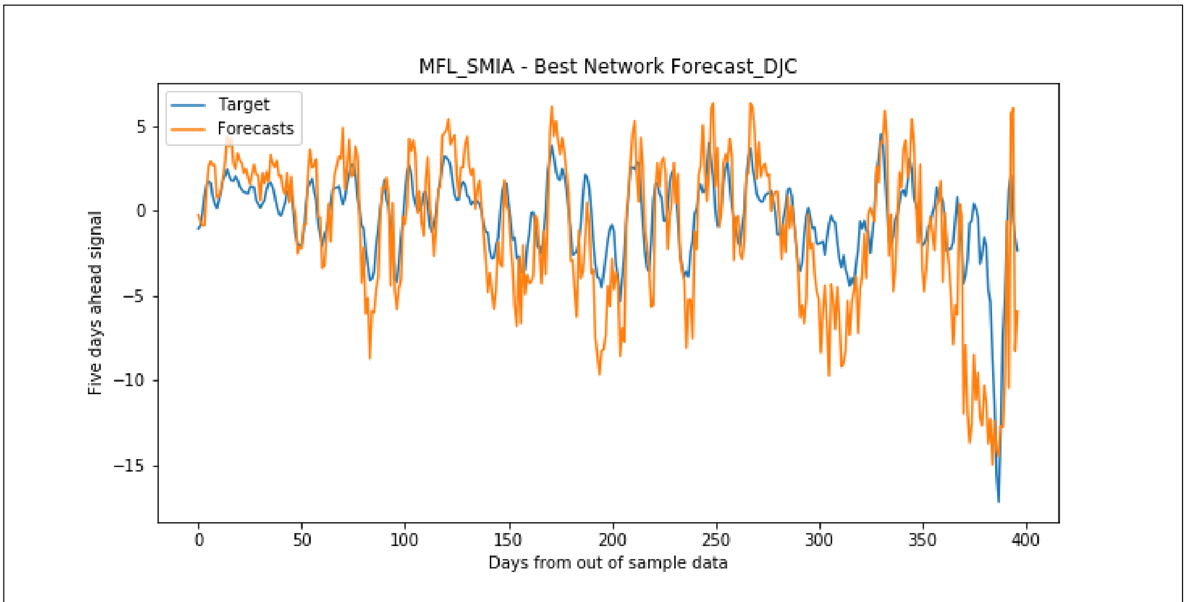


Fig. 8.3 The forecasting for five days ahead prediction using the M-FL-SMIA network.

As showed in figure 8.3, the prediction of the proposed network (M-FL-SMIA) when using the DJC stock price data, the figure illustrated that the forecasted signal followed the target signal. However, in some points, the signals look not close to each other. in general, the prediction results and the behaviour of the signals indicated that the M-FL-SMIA network looks a promising model for financial prediction for five days ahead.

Table 8.31 The best results of the Annualised Volatility (AV) for the prediction of five days ahead with the average for all networks.

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD	Average
FLNN	15.47144	15.56260	16.81314	35.16634	35.75308	30.72444	30.75543	61.52982	**101.34412**	**38.12449**
MLP	15.49770	15.68840	17.29550	36.42510	37.12470	32.65290	31.90650	61.59024	102.29560	38.94185
SMIA	15.85660	15.62360	16.89820	35.57702	35.47790	31.82670	31.4531	**60.80959**	101.72290	39.22406
FL-SMIA	15.65030	15.53630	**16.71910**	35.56130	35.73694	31.36340	31.23310	62.93981	101.81230	38.50584
FL-SMIA*	15.55320	15.53940	16.78210	35.38960	35.54270	31.05930	30.71120	63.74985	106.65943	38.99853
D-FL-SMIA	15.67080	15.42257	17.32141	36.48495	36.10657	30.91655	30.59969	63.79760	102.43480	38.75055
MD-FL-SMIA	15.47915	15.68291	17.21941	36.81188	35.77776	30.73750	31.09269	63.73456	101.56104	38.67743
MD-FL-SMIA-2	**15.44247**	15.54201	16.96078	36.24646	35.97940	30.61085	30.77369	62.30858	102.45450	38.47986
M-FL-SMIA	15.56753	**15.36008**	16.97451	**35.13308**	**35.43797**	**30.43459**	**30.50318**	61.18063	101.78271	**38.04159**
FL-SMIA-RBM	16.06299	17.09618	18.01588	39.11082	38.83697	33.97119	33.80575	67.11419	106.01466	41.11429

Table 8.32 The best results for the MSE-Testing for the prediction of five days ahead with the average for all networks.

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD	Average
FLNN	0.00739	0.01046	0.01358	0.00468	0.00555	0.00327	0.00375	0.00732	0.01222	0.00758
MLP	0.00264	**0.00173**	0.00261	0.00305	0.00385	0.00130	0.00230	0.00626	0.00594	0.00330
SMIA	**0.00158**	0.00264	**0.00236**	0.00141	**0.00177**	0.00136	0.00155	0.00186	0.00954	0.00268
FL-SMIA	0.00204	0.00319	0.00254	0.00150	0.00190	0.00165	0.00167	0.00279	0.00271	**0.00222**
FL-SMIA*	0.00220	0.00310	0.00313	**0.00140**	0.00199	**0.00122**	**0.00120**	**0.00178**	**0.00227**	**0.00203**
D-FL-SMIA	0.01324	0.05695	0.03558	0.01448	0.02163	0.01059	0.03073	0.02487	0.02973	0.02642
MD-FL-SMIA	0.04281	0.03349	0.00925	0.01313	0.02320	0.00533	0.01435	0.02033	0.01305	0.01944
MD-FL-SMIA-2	0.02818	0.03501	0.03194	0.00769	0.01838	0.01854	0.01253	0.01034	0.02417	0.02075
M-FL-SMIA	0.00514	0.03838	0.03256	0.00752	0.02956	0.01682	0.01295	0.01752	0.04517	0.02285
FL-SMIA-RBM	0.00517	0.00871	0.09822	0.00375	0.00397	0.00758	0.00786	0.00745	0.00998	0.01697

The average AV results showed that the M-FL-SMIA network and FLNN network produced the lowest values than all other networks as in table 8.31. However, the results proved that the M-FL-SMIA network produced a lower average AV value than the FLNN network (38.04150 vs 38.12449). The average results of MSE-Testing for five days ahead prediction have been listed in table 8.32. The results illustrated that, although all networks have decreased the values of MSE-Testing using all the data-sets, the average comparison between the proposed networks and the FLNN network proved that the proposed networks FL-SMIA*, and FL-SMIA produced lower average values of MSE-Testing the all other networks.

For the comparison between all networks that used in this research, The MSE-Testing average results showed that the multi-layer networks which have been used the immune learning algorithm (SMIA, FL-SMIA, and FL-SMIA*) outperforming all other networks with the average of values (0.00268, 0.00203, and 0.00222) respectively. This result emphases that the using algorithm of immune learning gives the reason behind the improvements in prediction ability of the multi-layer networks.

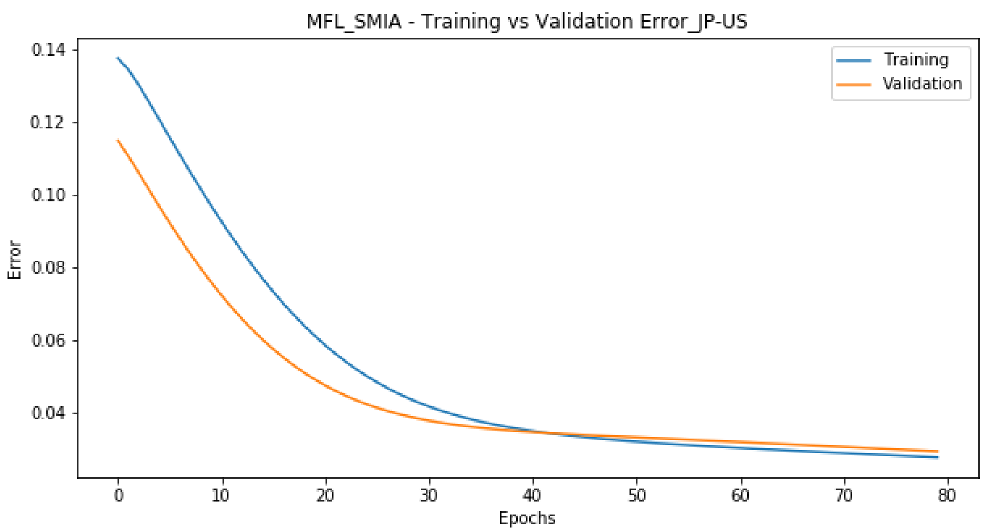


Fig. 8.4 The training error and validation error for five days ahead prediction using M-FL-SMIA network.

The prediction errors for the proposed network M-FL-SMIA using the JP/US data have been illustrated in figure 8.4 as an example of the convergence between the training error and validation error, the figure shown that the signals are near to zero value. All relevant figures have been listed in Appendix B.

The M-FL-SMIA network results and figures proved that the proposed network M-FL-SMIA have been successfully predicted the financial data, as well as it improving the prediction ability of the multi-layer networks and the proposed network FL-SMIA network for five days ahead prediction.

Chapter 9

Additional Evaluation and Discussion

In this chapter, the results for several metrics will be presented and discussed in order to evaluate the prediction for all neural networks that have been used in this research. The metrics results include the results for the Maximum Draw-Down(MDD), Correct Directional Change (CDC), and Signal to Noise Ratio (SNR).

In addition, two statistical tests have been utilised to test the difference between the proposed networks (FL-SMIA, FL-SMIA*, D-FL-SMIA, MD-FL-SMIA, and M-FL-SMIA) and other networks (FLNN, MLP, and SMIA).

9.1 Maximum Draw-Down (MDD)

As explained in chapter 3 3, the MDD is an indicator measure of downside risk over a specified time period. That means MDD only measures the size of the largest loss of financial trading, without consideration to the frequency of large losses.

In this section, the MDD results for all data-sets and the average results for all networks have been illustrated to measure the ability of the proposed networks in financial prediction domain.

The desirable MDD value for the financial prediction is the lower value which indicates less risk or less losing.

The results of maximum Draw-Down have been listed in table 9.1, the average results of MDD for one day ahead prediction demonstrate that the SMIA network and FL-SMIA network outperformed all other networks as they produced lower average MDD values than all other networks.

Table 9.1 The best results for the MDD for the one day ahead prediction with the average for all networks.

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD	Average
FLNN	**-0.59050**	-0.98186	-1.06498	-3.94011	-4.76773	-2.15584	-4.39138	-3.54513	-7.10301	-3.17117
MLP	-1.14592	-1.34810	-1.37053	-4.19710	-5.31520	**-2.04130**	**-2.07280**	-3.54401	-7.43470	-3.16330
SMIA	-1.14585	-0.70860	-1.05898	**-3.94010**	-5.30600	-2.28780	-3.27870	-4.69830	-1.55892	**-2.66481**
FL-SMIA	-1.14582	-0.98176	-1.37050	-4.17920	-5.31100	-4.53540	-4.39550	-3.98901	**-1.31332**	**-3.02461**
FL-SMIA*	-1.14591	-1.67883	-1.74721	-4.18505	-4.10674	-2.83302	-6.23950	-3.13635	-14.20200	-4.36384
D-FL-SMIA	-1.14605	-1.34812	-1.37047	-4.81193	**-3.36531**	-2.83132	-6.82699	**-2.39031**	-7.94212	-3.55918
MD-FL-SMIA	-1.14587	**-0.88375**	**-0.89781**	-5.80070	-3.36621	-2.83122	-2.75674	-6.13785	-6.63816	-3.38426
MD-FL-SMIA-2	-1.14586	-0.99930	-1.11824	-3.94020	-3.36701	-2.83204	-2.75676	-7.94212	-7.47440	-3.50844
M-FL-SMIA	-1.14583	-1.34810	-1.49962	-4.19713	-3.36611	-2.83212	-2.75685	-4.16641	-8.41764	-3.30331
FL-SMIA-RBM	-3.66902	-1.97400	-1.94909	-4.88751	-4.54845	-3.05554	-3.63258	-6.01979	-16.73931	-5.16392

In terms of measuring the results of maximum Draw-Down (MDD) for five days ahead prediction, it can be noticed from the results in table 9.2 that the FLNN network and the M-FL-SMIA network produced the lowest values for MDD when compared to all other networks.

Table 9.2 The best results for the MDD for the prediction of five days ahead with the average for all networks.

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD	Average
FLNN	-1.33533	-2.16558	-2.72277	-6.10270	**-3.90036**	-3.82648	-3.69906	-7.61349	-13.24989	**-4.95730**
MLP	-1.86650	-1.32580	-3.76240	-6.82220	-9.87550	-6.62070	-5.57790	**-6.73280**	-13.25989	-6.20485
SMIA	-1.64940	-1.35280	-2.72280	-4.45810	-6.94850	-3.85601	-8.39410	-7.61259	-13.24990	-5.58269
FL-SMIA	-2.18570	-1.35820	-2.72275	**-4.10180**	-6.10180	-8.27690	-8.17740	-8.46331	**-13.24949**	-6.07082
FL-SMIA*	**-1.33530**	-1.35841	-2.72281	-6.10180	-7.08090	-6.62070	-3. 69905	-13.67533	-24.88985	-7.49824
D-FL-SMIA	-1.42010	**-1.32576**	-3.76235	-6.10199	-7.08089	-4.52117	-3.69816	-17.50750	-21.56010	-7.44200
MD-FL-SMIA	-1.40679	-2.09530	-2.72307	-6.10210	-4.27926	-3.82650	-4.08933	-22.27483	-21.56013	-7.59526
MD-FL-SMIA-2	-1.36718	-2.09620	-2.72249	-6.10179	-6.28381	**-3.82558**	-3.69915	-14.74310	-15.13290	-6.21913
M-FL-SMIA	-1.66014	-1.35824	**-2.72187**	-5.11686	-7.08080	-3.82639	**-3.07051**	-8.27316	-14.24857	**-5.26184**
FL-SMIA-RBM	-2.38445	-18.94282	-5.49508	-13.45247	-14.49167	-8.40934	-8.50935	-19.93960	-25.72228	-13.03856

9.2 Correct Directional Change (CDC)

The Correct Directional Change (CDC) has been used in this research to measure the ability of the networks on correctly forecasting the subsequent actual change of a prediction variable. A large value refers to a better predictor.

As higher values are preferable for the correct Directional change (CDC) measure, it appears in table 9.3 that all networks obtained high CDC results for most data-sets.

While the highest value of the CDC is 71.596 which has been achieved by using the proposed network FL-SMIA on forecasting the GOLD data.

The average results for the CDC measure proved that the SMIA and FL-SMIA networks reached the highest values than all other networks that used in this research for the prediction for one day ahead.

Table 9.3 The best results for the CDC for the prediction of one day ahead with the average for all networks.

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD	Average
FLNN	65.657	61.364	62.879	66.414	64.646	65.152	64.141	62.647	59.406	63.590
MLP	**68.690**	63.380	63.890	66.330	**65.150**	63.540	64.560	62.790	57.930	64.029
SMIA	67.680	62.880	62.370	67.850	62.120	62.780	63.790	**71.430**	70.591	**65.721**
FL-SMIA	64.890	61.870	62.630	68.101	61.110	62.530	62.780	69.580	**71.596**	**65.010**
FL-SMIA*	67.010	61.700	62.940	66.080	63.860	63.930	63.700	62.353	59.610	63.465
D-FL-SMIA	65.400	56.818	59.848	62.879	62.373	64.646	62.626	58.080	62.500	61.686
MD-FL-SMIA	63.636	59.849	**64.141**	63.131	60.354	**65.153**	64.394	62.354	58.074	62.343
MD-FL-SMIA-2	60.606	62.121	58.838	67.172	62.121	60.859	63.636	62.941	58.370	61.852
M-FL-SMIA	66.162	59.848	60.606	**68.434**	62.374	63.384	**66.162**	62.500	59.407	63.209
FL-SMIA-RBM	62.626	**66.919**	60.604	64.394	60.606	63.131	63.384	61.324	61.037	62.669

Table 9.4 The best results for the CDC for the prediction of five days ahead with the average for all networks.

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD	Average
FLNN	66.667	65.404	64.899	62.374	57.828	**65.404**	62.626	63.529	58.815	**63.061**
MLP	68.670	65.660	62.120	60.000	59.850	62.531	**63.540**	64.265	58.519	62.795
SMIA	67.170	64.890	63.130	60.510	59.090	61.772	62.540	**65.000**	**61.481**	62.843
FL-SMIA	**68.920**	**68.640**	**65.150**	61.120	57.830	62.533	62.530	63.971	60.148	**63.427**
FL-SMIA*	62.370	63.380	61.360	62.530	**61.870**	61.774	63.290	63.120	58.990	62.076
D-FL-SMIA	63.380	63.889	58.333	**63.131**	60.859	61.111	62.375	55.441	60.000	60.947
MD-FL-SMIA	64.899	65.909	63.636	60.859	60.354	64.394	62.374	62.794	54.519	62.193
MD-FL-SMIA-2	64.646	60.110	55.808	62.121	60.353	62.374	62.879	63.530	60.444	61.363
M-FL-SMIA	63.889	67.677	57.828	60.354	58.333	63.131	62.121	64.559	58.963	61.873
FL-SMIA-RBM	60.606	60.101	57.576	61.111	57.576	59.596	60.354	61.618	60.296	59.870

For the five days ahead prediction, as in table 9.4 the correct directional change (CDC) results indicates that the proposed network FL-SMIA produced the higher CDC value (68.920) than all other networks when forecasting the US/UK exchange rate data.

The average results of the CDC showed that proposed network FL-SMIA outperformed all other networks with the CDC value (63.427). However, FL-SMIA network followed by FLNN network.

9.3 Signal to Noise Ratio (SNR)

The aim of using the Signal to Noise Ratio (SNR) measure is to compare the amount of significant information provides by the signal (data) with the amount of background noise of the signal (distraction from the signal).

For one day ahead prediction, the SNR results which have been illustrated in table 9.5. As the higher value of the SNR indicating a clearer reading of the signal, the highest value of the SNR over all the data-sets have predicted by the FL-SMIA network when used to forecast the GOLD exchange rate data with an SNR value of 37.689, which indicates a clearer reading of the signal.

The average results of SNR indicated that the highest average of SNR values than all other networks produced by the SMIA and FL-SMIA networks which are 26.327 and 26.233 respectively.

Table 9.5 The best results for the SNR for the prediction of one day ahead with the average for all networks.

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD	Average
FLNN	**22.270**	**23.660**	20.996	**24.707**	23.013	**25.163**	**24.564**	25.514	24.669	23.840
MLP	22.220	23.370	**23.390**	24.630	**23.310**	22.660	22.7300	24.740	24.390	23.493
SMIA	22.221	23.501	23.380	24.550	22.960	24.040	24.090	34.670	37.534	**26.327**
FL-SMIA	22.250	22.760	23.070	24.420	22.790	23.980	24.050	**35.084**	**37.689**	**26.233**
FL-SMIA*	21.630	22.652	23.100	24.540	23.160	24.480	23.840	24.269	23.570	23.471
D-FL-SMIA	19.360	19.399	18.690	22.849	20.560	16.200	21.380	13.090	21.051	19.175
MD-FL-SMIA	15.823	16.164	14.493	14.986	22.465	24.188	20.013	24.109	16.361	18.734
MD-FL-SMIA-2	19.439	22.650	18.824	18.074	21.654	20.369	20.834	20.753	16.929	19.947
M-FL-SMIA	16.816	18.679	13.945	12.572	17.749	15.160	19.593	14.393	19.349	16.473
FL-SMIA-RBM	19.918	19.435	21.789	23.160	21.857	23.229	22.702	22.680	21.697	21.829

Table 9.6 The best results for the SNR for the prediction of five days ahead with the average for all networks.

Network	US/UK	US/EU	JP/US	NQO	NQC	DJO	DJC	OIL	GOLD	Average
FLNN	**27.481**	**24.544**	23.052	**26.217**	**25.068**	23.923	**25.778**	24.339	26.476	**25.209**
MLP	23.833	25.660	22.860	21.320	20.550	25.510	23.020	22.713	21.030	22.944
SMIA	26.070	23.850	**23.290**	24.670	23.920	25.310	24.740	21.599	25.925	24.375
FL-SMIA	24.970	23.020	22.980	24.340	23.610	24.460	24.410	**27.790**	**28.887**	**24.941**
FL-SMIA*	24.640	23.190	22.060	24.680	23.420	**25.770**	26.010	25.230	24.360	24.373
D-FL-SMIA	21.750	13.156	15.425	19.076	16.626	20.923	14.075	17.677	17.069	17.308
MD-FL-SMIA	14.621	16.857	21.396	19.689	16.434	24.463	19.397	20.172	22.738	19.529
MD-FL-SMIA-2	17.166	17.191	15.677	23.806	17.467	17.570	20.162	24.729	19.436	19.245
M-FL-SMIA	22.541	16.034	16.242	23.325	14.856	18.558	20.024	21.961	15.140	18.742
FL-SMIA-RBM	22.154	20.956	8.862	23.318	20.585	21.793	21.920	23.356	22.159	20.567

The SNR results for five days ahead the prediction have been listed in table 9.6, the results indicated that the FL-SMIA network produced the higher value of SNR than all networks when forecasting the GOLD data with the value of 28.887.

For the average results, the FLNN network outperformed all other networks followed by the FL-SMIA network (25.209 vs 24.941). While the comparison between the multi-layer networks showed that the FL-SMIA* network competes with the networks SMIA to reach the highest average for SNR values. The average SNR results for five days ahead prediction also showed that the comparison between the proposed networks, indicated that the FL-SMIA, FL-SMIA*, and FL-SMIA-RBM networks outperformed all other proposed networks in this research.

9.4 Significance of Differences in RP Results

The Wilcoxon signed rank test have been used in this research for paired samples to determine differences in the overall RP performance of the models overall used the data-sets.

9.4.1 The Wilcoxon Signed Rank Test for One Day Ahead Prediction

The Wilcoxon signed rank test for one day ahead the prediction showed that the FL-SMIA produces significantly better RP values than D-FL-SMIA, MD-FL-SMIA-2 and FL-SMIA-RBM networks ($p < 0.01$ and $p < 0.05$). While the results of RP produced by FLNN, MLP, SMIA, FL-SMIA*, MD-FL-SMIA, and M-FL-SMIA showed no significant differences with the FL-SMIA results.

For the other proposed networks, the Wilcoxon test results indicated that FL-SMIA* network produces only significantly better RP values than FL-SMIA-RBM network but in same time FL-SMIA* network appreciably lower RP values than FLNN. The Wilcoxon test results for the D-FL-SMIA, MD-FL-SMIA and MD-FL-SMIA-2 networks showed significantly lower RP values than the FLNN models and the FL-SMIA model and significantly better RP values than FL-SMIA-RBM network. While the D-FL-SMIA, MD-FL-SMIA and MD-FL-SMIA-2 networks, the test results illustrated no significant differences with the other networks.

For the proposed M-FL-SMIA network, the test results denoted that the M-FL-SMIA model produces significantly better RP values than MLP and FL-SMIA-RBM networks but significantly lower RP values than FLNN model. While no significant differences with the other networks. The test results also illustrated that the FL-SMIA-RBM network produces significant differences of RP values compared with all networks.

9.4.2 The Wilcoxon Signed Rank Test for Five Days Ahead Prediction

For five days ahead prediction, the Wilcoxon signed rank test results proved that the FL-SMIA network produces significantly better RP values than FL-SMIA-RBM network ($p < 0.01$ and $p < 0.05$), While the differences of RP values between FL-SMIA network and all other networks are not significant. The FL-SMIA* results are in the middle of the range, only significantly better than MLP and significantly worse than FL-SMIA network and MFL-SMIA network.

The extended of FL-SMIA networks (D-FL-SMIA, MD-FL-SMIA-2), the test results indicated that D-FL-SMIA and MD-FL-SMIA-2 networks produces no significant differences of RP values compared with all networks except the M-FL-SMIA model as it produces significantly better RP values than the D-FL-SMIA and MD-FL-SMIA-2 models. While the FL-SMIA-RBM networks produces significantly lower RP values than the D-FL-SMIA and MD-FL-SMIA-2 models.

9.5 Similarity Between Residuals

To measure the similarity between models behaviour over the data-sets, the correlation coefficients have been applied to the residuals for each model that has been used in this research.

9.5.1 Correlation Analysis

The correlation between the FL-SMIA network and the other proposed networks as well as the correlation between the FL-SMIA network and the existing models (FLNN, MLP, and SMIA) have been tested in this research in order to understand whether the predictions are qualitatively different. To this end, the correlation coefficients have been calculated to investigate the correlation between all the networks used in this research.

The correlation coefficient is a measure that determines the degree to which two variables are associated. The correlation coefficient is considered a useful measure in the financial domain. For example, it can be used to determine an investment behaves in relation to another fund or asset class.

The range of values for the correlation coefficient is between -1.0 and 1.0. The correlation coefficient between two variables is calculated as follows:

$$P_{xy} = Corr(x, y) = \frac{Cov(x, y)}{\sigma_x \sigma_y}, -1 \leqslant P_{xy} \leqslant 1 \tag{9.1}$$

where $Cov(x, y)$ represent the co-variance between the two variables x and y, the $\sigma_x \sigma_y$ represent the standard deviations of x and y respectively.

A value of exactly 1.0 means there is a perfect positive relationship between the two variables. While a value of exactly -1.0 means there is a perfect negative relationship between the two variables. Negative correlation means that the variables move in opposite directions; for a positive increase in one variable, there is a decrease in the second variable [48].

The correlation coefficients between all networks architectures that have been used in this research for prediction of the NQC data-set are shown in table 9.7. We can see that the correlation between FL-SMIA and SMIA network is higher than between FL-SMIA and the MLP and other proposed models. Interestingly, the correlation value between FL-SMIA and FLNN is even lower than between FL-SMIA and the MLP model. The FL-SMIA model, therefore, seems to provide some new characteristics that could make it useful in an ensemble model, e.g. together with the MLP and the FLNN.

Table 9.7 also shows that the correlation between FL-SMIA model and the proposed networks (FL-SMIA*, D-FL-SMIA, and M-FL-SMIA) is higher than that between FL-SMIA and the other proposed models. This result can be interpreted such that although all proposed networks use the same learning algorithm, they have different architectures which result in lower values of correlation between the networks.

Table 9.7 The correlation coefficients for NQO (one day ahead prediction)

Network	FLNN	MLP	SMIA	FL-SMIA	FL-SMIA*	D-FL-SMIA	MD-FL-SMIA	MD-FL-SMIA-2	M-FL-SMIA	FL-SMIA-RBM
FLNN	**1**	0.6847	0.6807	0.6671	0.4994	0.8144	0.8074	0.8932	0.8724	0.6752
MLP	0.6847	**1**	0.8547	0.8431	0.7262	0.7354	0.7619	0.5680	0.7664	0.6815
SMIA	0.6807	0.8547	**1**	0.9889	0.8203	0.6780	0.6444	0.6840	0.7255	0.6447
FL-SMIA	0.6671	0.8431	0.9889	**1**	0.8311	0.7042	0.6691	0.6950	0.7452	0.6599
FL-SMIA*	0.4994	0.7262	0.8203	0.8311	**1**	0.5741	0.5118	0.5838	0.6037	0.8179
D-FL-SMIA	0.8144	0.7354	0.6780	0.7042	0.5741	**1**	0.8853	0.7880	0.9316	0.6856
MD-FL-SMIA	0.8074	0.7619	0.6444	0.6691	0.5118	0.8853	**1**	0.7117	0.9459	0.6813
MD-FL-SMIA-2	0.8932	0.5680	0.6840	0.6950	0.5838	0.7880	0.7117	**1**	0.8477	0.6813
M-FL-SMIA	0.8724	0.7664	0.7255	0.7452	0.6037	0.9316	0.9459	0.8477	**1**	0.7408
FL-SMIA-RBM	0.6752	0.6815	0.6447	0.6599	0.8179	0.6856	0.6813	0.6813	0.7408	**1**

For the correlation between FL-SMIA* network and other networks, the results in table 9.7 demonstrated that the correlation between FL-SMIA* network and the SMIA, FL-SMIA, and FL-SMIA-RBM networks produced higher values than between FL-SMIA* and other networks. While the correlation results of other proposed networks showed that the the networks D-FL-SMIA, MD-FL-SMIA, MD-FL-SMIA-2, and M-FL-SMIA have performed more similarly with FLNN model, as well as they performed more similarly with each other than with FL-SMIA model and other networks. The correlation between FL-SMIA-RBM network and the proposed networks (FL-SMIA* and M-FL-SMIA) predicted the highest results than all other networks, but it shows a lower correlation value than all other networks.

Table 9.8 The correlation coefficients for US/EU (five days ahead prediction).

Network	FLNN	MLP	SMIA	FL-SMIA	FL-SMIA*	D-FL-SMIA	MD-FL-SMIA	MD-FL-SMIA-2	M-FL-SMIA	FL-SMIA-RBM
FLNN	**1**	0.9461	0.9316	0.8773	0.8151	0.8560	0.8562	0.8705	0.9008	0.7415
MLP	0.9461	**1**	0.9143	0.8509	0.7803	0.8356	0.8342	0.8848	0.8519	0.7891
SMIA	0.9316	0.9143	**1**	0.9837	0.9376	0.9383	0.9609	0.9564	0.9401	0.8405
FL-SMIA	0.8773	0.8509	0.9837	**1**	0.9391	0.9153	0.9546	0.9619	0.9101	0.7935
FL-SMIA*	0.8151	0.7803	0.9376	0.9391	**1**	0.9406	0.9578	0.8985	0.9417	0.8419
D-FL-SMIA	0.8560	0.8356	0.9383	0.9153	0.9406	**1**	0.9568	0.9263	0.9591	0.9000
MD-FL-SMIA	0.8562	0.8342	0.9609	0.9546	0.9578	0.9568	**1**	0.9400	0.9509	0.8873
MD-FL-SMIA-2	0.8705	0.8848	0.9564	0.9619	0.8985	0.9263	0.9400	**1**	0.9067	0.8095
M-FL-SMIA	0.9008	0.8519	0.9401	0.9101	0.9417	0.9591	0.9509	0.9067	**1**	0.8397
FL-SMIA-RBM	0.7415	0.7891	0.8405	0.7935	0.8419	0.9000	0.8873	0.8095	0.9067	**1**

The correlation coefficients for five days ahead prediction between all network architectures used in this research with the US/EU data-set are listed in table 9.8.

The results showed on one hand that the proposed FL-SMIA model performs more similarly to SMIA (0.9837) than to FLNN, and MLP (0.8773, and 0.8509). On the other hand, the correlations between FL-SMIA and the other proposed networks are higher than the correlations between FL-SMIA model and the networks (FLNN, MLP and FL-SMIA-RBM). The results indicate that some characteristics of FLNN and SMIA components are retained in the proposed extended models.

As represented in table 9.8, in general, the correlation between all proposed networks and other networks demonstrated that the proposed networks performed more similarly to each other and to the SMIA network than to the FLNN and MLP networks. Also, it is observed that the results of the correlation coefficient for five days ahead prediction are higher than the correlation coefficient results for the prediction for one day ahead.

9.6 Discussion on the Comparison of RP Results

This section discusses the comparison results between all the networks which have been used in this research. This section will discuss the following:

1. Discusses the effect of using the inputs and their products and the use of immune algorithm with a multilayer network for improving the prediction ability of the multi-layer networks in the field of financial prediction. Therefore, the performances of the FL-SMIA network and other proposed networks have been compared with the multilayer networks (MLP, and SMIA).

2. Summarise the performances of different architectures models which have been used the inputs and their products in the domain of financial prediction. So that,

the performances of the FL-SMIA network and the other proposed networks have been compared with the performance of the FLNN model.

3. The comparison between the extended networks of FL-SMIA (FL-SMIA*, D-FL-SMIA, MD-FL-SMIA, MD-FL-SMIA-2, M-FL-SMIA, and FL-SMIA-RBM) and the existing networks that used in this research (FLNN, MLP, and SMIA) will be discussed through this section.

9.6.1 Comparison Between the FL-SMIA Network and the Multi-layer Networks

In this section, the prediction results of the FL-SMIA network compared with the multi-layer networks (MLP, and SMIA) will be explained. The discussion will be based on RP results which have been presented in chapters 8.

The results for one day ahead prediction indicated that the FL-SMIA network outperformed the MLP network when produced the higher Relative Profit (RP) values for all data-sets except the OIL data.

While FL-SMIA network outperformed the SMIA on five from nine of data-sets. However, the FL-SMIA network competes with the SMIA network for forecasting the US/EU, JP/US, NQO, and NQC data-sets. Overall, the FL-SMIA network predicted the highest RP average results than the MLP, and the SMIA networks for one day ahead prediction.

For the five days ahead prediction, on one hand, the results indicated that the FL-SMIA network outperformed MLP network for all data-sets except the OIL data. However, the RP average results showed that the FL-SMIA network produced the higher RP average value (88.575) than the RP average value of the MLP network. On the other hand, the comparison of the RP results between FL-SMIA network and the SMIA network proved that the FL-SMIA network results for all data-sets look closer to the results for SMIA network. However, the average results of RP showed that the FL-SMIA network reached higher average PR than the SMIA network.

Consequently, for five days ahead prediction, the FL-SMIA network outperformed the multi-layer networks (MLP, and SMIA) that have been used in this research. Therefore, FL-SMIA network could be considered as a promising model for financial prediction.

9.6.2 Comparison Between the FL-SMIA Network and the FLNN Network

The prediction for one day ahead, the comparison of RP results between the FL-SMIA network and the FLNN network indicated that the FLNN network achieved higher RP on all data-sets except the stock prices data (DJC and GOLD). Also, the average RP of the FLNN network was higher the FL-SMIA network (75.34 vs 73.26). However, the AV result of the FL-SMIA (and SMIA) was significantly better than that of the FLNN (10.38 vs 12.54).

For the five days ahead prediction, the comparison for RP results between the FL-SMIA model and the FLNN network represented that, although the FL-SMIA model outperforms the FLNN network for all exchange rate data-sets and one of stock prices data (NQC), the average value of RP for FLNN network is higher than the average RP for the FL-SMIA model (89.38 vs 88.57). The average AV is value slightly worse (higher) for FL-SMIA.

9.6.3 Discussing the Results of the FL-SMIA Network Extensions

This section discusses and compares the results for the extended FL-SMIA networks, which are FL-SMIA*, D-FL-SMIA, MD-FL-SMIA, MD-FL-SMIA-2, M-FL-SMIA, and FL-SMIA-RBM.

For one day ahead prediction, the FL-SMIA network outperformed all the other proposed models. However, in most cases the proposed networks produced results similar to the FL-SMIA network except the FL-SMIA-RBM. The average RP results showed that the FL-SMIA network reached a higher value (73.268) than all the other proposed networks. The most common second performance to the FL-SMIA Network is the M-FL-SMIA as it produced the second highest average AR (72.743).

While the financial prediction for five days ahead indicated an improvement for all proposed networks (better prediction results than with one day ahead). The results also showed that in some cases the proposed networks outperformed the FL-SMIA Network (as discussed in chapter 8).

Consequently, the FL-SMIA Network is compared with the other proposed networks. In the average RP results, the results proved that the M-FL-SMIA network outperformed the FL-SMIA Network with a higher average RP than the FL-SMIA Network (89.724 vs 88.575). Moreover, the M-FL-SMIA network outperformed the FLNN Network for the average RP results (89.724 vs 89.382). Also, the M-FL-SMIA network outperformed all other networks used in this research when comparing the average RP results for five days ahead prediction.

The last proposed network (FL-SMIA-RBM) produced lower results of RP than all the other proposed networks. This may be due to use two methods of unsupervised learning. However, the unsupervised learning help on reducing the MSE-Testing results to produced lower errors than all proposed networks except the FL-SMIA and FL-SMIA* networks. Thus, the FL-SMIA-RBM may perform better if used with another domain.

Chapter 10

Alternative Evaluation and Model Selection

In this chapter, an alternative evaluation method for the neural network models has been used. In the previous chapter, we focused on the models performing best on the test set of moving averages of prices of exchange rate index and commodity data, as introduced in table 3.1 and table 3.3 in chapter 3.

In this chapter we adapt the data and evaluation to focus on reflecting actual application situation where we buy and sell tradable assets on current prices and don't know the future returns. The additional experiments focus on two proposed models FL-SMIA and M-FL-SMIA, which were most promising in the previous sections, and for comparison the FLNN and MLP have been included in this chapter.

The data-sets used in additional experiments include all the data that are listed in table 3.1 except the opening prices (NQO and DJO). Instead, we include two new data-sets, which are a Crude Oil WTI Futures (C-OIL) from 1/10/1999 to 29/05/20019 and SPDR 500 ETF (SPY) from 1/10/1999 to 27/04/20019, as more tradeable alternatives to commodities and indices. The statistics of daily return percentages ($(P_t - P_{t-1}) \times 100/P_{t-1}$) for SPY and C-OIL data are listed in table 10.1. Figure 10.1 shows histograms of daily returns for SPY and C-OIL. As expected, the values of these statistics for C-OIL are in the same range as for the oil price (OIL), and the histograms are visually similar. The SPY values and histogram are broadly similar to the DJ and NASDAQ indices. We wanted to make use of the longer time series available for these datasets, therefore the values for OIL here are not directly comparable to those for C-OIL.

Table 10.1 The statistics of daily returns for the two additional time series.

Data-sets	Mean	Std	Skew	Kurtosis
SPY	-0.00934	1.19939	0.28396	10.05643
C-OIL	0.01029	2.36144	0.33737	4.11287

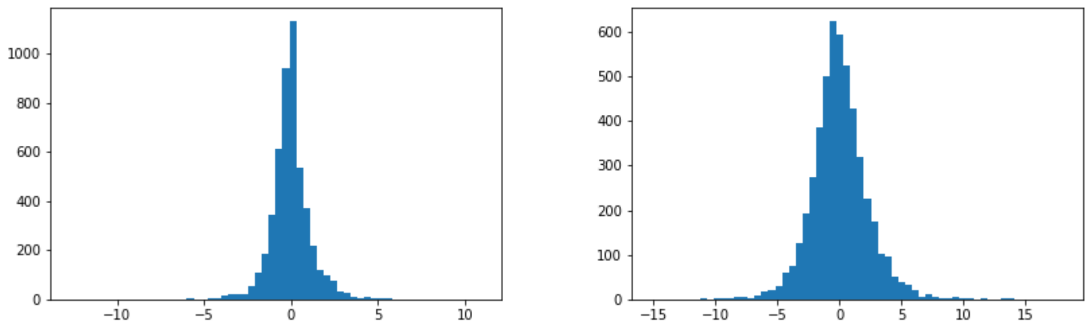


Fig. 10.1 Daily return histograms for SPY (left) and C-OIL (right).

10.1 Training and Evaluation

We report the results of 20 simulations to investigate the performance of the neural network models. The reported values are for the combination of hyperparameters that produced the best average validation result in 20 simulations a grid search as listed in section 3.6. Each simulation includes 80 epoch of training with early stopping as described in the previous chapters.

10.1.1 Statistical Comparisons

For comparing models across datasets, a Wilcoxon signed-rank test is used, as in the previous chapters, as the distribution of the metrics is rarely normally distributed as we found when testing the data. The Wilcoxon test tests for different medians between the compared distributions. It is applied to the average values per dataset, as the values over simulations can not be meaningfully paired for a Wilcoxon test.

In addition, we compare selected pairs of models on individual datasets using a Mann-Whitney U test. The Mann-Whitney U test is a non-parametric test whether a random pair of values from two samples is likely to have a greater value from one sample. We also considered using a t-test, a parametric test which finds different distribution means. However, Shapiro tests showed that there were few normally distributed samples among the simulation results. Thus there would have been very low coverage, and it was decided not report t-test results.

10.2 One Day Ahead Prediction

In this section the mean results for one day ahead prediction have been listed in tables 10.3, 10.6 and 10.9, showing the minimum, maximum, average, and the standard deviation for each model.

The parameters found in the grid search are listed in table 10.2. Here, LR is the learning rate, MOM is the momentum, DR is the decay rate, and H-U refers to the number of hidden units in the hidden layer. The FLNN, FL-SMIA, and M-FL-SMIA are the second order models.

Table 10.2 The parameters values that have been used to predict average results for one day ahead prediction (20 simulations)

Data	FLNN			MLP				FL-SMIA				M-FL-SMIA			
sets	LR	MOM	DR	LR	MOM	DR	H-U	LR	MOM	DR	H-U	LR	MOM	DR	H-U
US/UK	0.4	0.03	0.01	0.4	0	0.1	12	0.4	0.03	0.01	40	0.03	0.4	0.0005	55
US/EU	0.4	0.01	0.0005	0.4	0	0.0005	4	0.03	0	0.001	59	0.01	0.03	0.001	74
JP/US	0.4	0	0	0.4	0	0.01	4	0.1	0.4	0.0005	71	0.1	0.4	0.001	86
NQC	0.1	0.1	0.1	0.4	0.01	0	12	0.4	0.01	0.001	49	0.1	0.03	0	64
DJC	0.1	0.01	0.0005	0.1	0.03	0.1	12	0.1	0.01	0.1	47	0.01	0.01	0.0001	62
OIL	0.1	0.03	0.1	0.4	0	0.001	12	0.4	0.03	0.1	86	0.04	0.1	0.01	101
GOLD	0.4	0	0.001	0.4	0.01	0.0001	6	0.4	0.4	0	56	0.03	0	0.1	55
SPY	0.1	0.4	0.001	0.4	0	0.01	12	0.1	0.01	0.001	45	0.4	0.03	0.1	60
C-OIL	0.4	0.03	0.0001	0.1	0.03	0.0005	8	0.1	0.4	0.0005	47	0.4	0.03	0.01	62

10.2.1 Relative Profit (RP)

The average Relative Profit it highest for the MLP model for 5 out the 9 datasets and in the average over the datasets. The FL-SMIA has the best average RP for 2 datasets and in most cases the lowest SD. Interestingly, although the FL-SMIA has the lower average and SD than MLP, it has a higher maximum on five datasets and on average. The M-FL-SMIA produces surprisingly low results, given the promising results in the previous chapters.

The standard deviation of all models is on average higher than the values, showing wide variation across simulations. The significance tests in table 10.4 show that the differences between models across datasets are mostly not significant, except for the M-FL-SMIA model versus the MLP.

For the two datasets, where the FL-SMIA model produces higher average RP results, we used the Mann-Whitney U test (table 10.5). However, the differences are

Table 10.3 The results for the RP over 20 simulations for one day ahead prediction.

Data	FLNN				MLP				FL-SMIA				M-FL-SMIA			
sets	Min	Max	Avg	Sd	Min	Max	Avg	Sd	Min	Max	Avg	Sd	Min	Max	Avg	Sd
US/UK	-35.59	36.48	7.84	22.79	-24.44	47.61	18.13	21.98	-21.32	50.08	**19.39**	20.29	-28.46	38.60	8.56	18.69
US/EU	-31.22	43.63	16.47	24.49	-35.75	50.74	**17.63**	25.83	-38.66	52.50	11.08	23.85	-35.06	42.64	8.76	17.77
JP/US	-27.18	40.18	**13.08**	22.09	-25.75	36.68	11.09	17.49	-26.50	33.95	9.56	17.64	-24.70	32.17	7.30	15.52
NQC	-18.55	26.04	6.96	14.34	-13.73	28.65	**12.70**	13.08	-19.70	28.83	9.26	12.00	-26.35	27.55	10.76	15.12
DJC	-22.71	28.15	5.60	13.82	-12.41	31.08	8.96	12.20	-26.88	32.82	**10.89**	18.29	-10.47	24.78	7.05	10.49
OIL	-38.92	40.59	17.08	20.61	-32.36	46.79	**19.89**	23.51	2.96	46.34	15.16	9.45	-35.14	36.35	12.82	21.76
GOLD	-25.65	28.07	10.90	18.74	-15.30	31.37	**11.03**	15.11	-18.81	26.06	8.18	14.87	-15.28	32.14	9.78	14.66
SPY	-18.88	38.00	**16.77**	16.43	-27.31	36.91	11.83	19.26	-14.32	38.40	10.52	11.56	-31.40	37.31	11.22	16.21
C-OIL	-36.41	39.04	12.20	20.99	-11.69	34.76	**16.47**	14.74	-25.81	38.45	10.43	19.19	-39.21	38.81	7.95	19.78
Average	-28.35	35.58	11.88	19.37	-22.08	38.29	**14.19**	18.13	-21.01	38.60	11.61	16.35	-27.34	34.48	9.36	16.67

Table 10.4 Wilcoxon signed rank test of models differences on average RP for one day ahead prediction. Level of significance $\alpha = 0.05$.

Comparison	R+	R-	Mean Difference	z-value	W-value	result
FL-SMIA versus FLNN	18	27	-4.86	-0.5331	18	not significant at $p < .05$
FL-SMIA versus MLP	5	40	-6.02	-2.0732	5	not significant at $p < .05$
FL-SMIA versus M-FL-SMIA	39	6	2.85	-1.9548	6	not significant at $p < .05$
M-FL-SMIA versus FLNN	8	37	-7.11	-1.7178	8	not significant at $p < .05$
M-FL-SMIA versus MLP	0	45	-8.27	-2.6656	0	**significant at $p < .05$**
FLNN versus MLP	10	35	-5.75	-1.4809	10	not significant at $p < .05$

clearly not significant. Thus, we don't have evidence that the SMIA models outperform the MLP or FLNN on average on any dataset.

10.2.2 Annualised Volatility (AV)

Annualised Volatility (AV) is a measure of investment risk and low AV values are desirable. Table 10.6 presents the AV results for each model when using all the datasets.

The results show wide variation over the datasets. The standard deviation over the simulations is comparatively very low, as is the variation between models. The only exception is the M-FL-SMIA on GOLD, which is dramatically lower than the other models (9 vs 70+). Even though these values look unusual, double checking revealed nothing suspicious. Therefore it seems that the M-FL-SMIA provides exceptionally consistent predictions for the GLOD dataset. It seems, however, advisable to run further experiments to confirm this results and and conclusions before using this model

Table 10.5 Mann-Whitney U Test for 20 average RP values of data-set, level of significance $\alpha = 0.05$

Comparison	U-value	Z-score	P-value	results
FL-SMIA versus MLP (US/UK)	199	0.01353	0.99202	not significant at $p < .05$
FL-SMIA versus MLP (DJC)	163.5	0.9738	0.33204	not significant at $p < .05$

in practice. There was unfortunately not enough time to perform more in-depth investigation of this phenomenon as part of this PhD project.

The comparison between models over the datasets showed no significant differences in Wilcoxon tests in table 10.7. This is despite the AV value of the M-FL-SMIA averaged over the datasets being the lowest by some margin (last row in table10.6), but that is only because of the single results for GOLD, and the non-parametric test is not sensitive for such a singular deviation.

The tests on individual datasets showed that the difference between the M-FL-SMIA and the second-best model MLP is highly significant in a Mann-Whitney U test (table 10.8). The FL-SMIA is producing significantly lower AV than MLP on DJC, but this difference is less significant.

Table 10.6 AV results over 20 simulations for one-day ahead prediction.

Data sets	**FLNN**				**MLP**				**FL-SMIA**				**M-FL-SMIA**			
	Min	**Max**	**Avg**	**Sd**	**Min**	**Max**	**Avg**	**Sd**	**Min**	**Max**	**Avg**	**Sd**	**Min**	**Max**	**Avg**	**Sd**
US/UK	8.74	9.07	8.93	0.12	8.50	9.07	8.87	0.19	8.05	9.07	**8.79**	0.29	8.70	9.07	8.97	0.11
US/EU	8.53	9.00	8.79	0.15	8.36	9.00	**8.76**	0.21	8.50	9.00	8.83	0.17	8.55	9.00	8.90	0.13
JP/US	11.32	11.85	11.64	0.16	11.41	11.85	**11.71**	0.12	11.48	11.85	11.73	0.13	11.52	11.85	11.76	0.10
NQC	27.53	27.98	27.81	0.16	27.44	27.98	27.76	0.17	27.43	27.98	27.83	0.14	27.48	27.97	**27.75**	0.17
DJC	24.33	24.75	24.64	0.12	24.24	24.76	24.63	0.14	24.18	24.76	**24.52**	0.15	24.43	24.75	24.67	0.10
OIL	41.15	43.22	42.34	0.63	40.44	43.16	**42.05**	0.71	42.04	43.23	42.93	0.31	41.57	43.24	42.44	0.58
GOLD	71.50	73.12	72.18	0.47	71.07	73.19	72.44	0.75	71.73	73.14	72.60	0.39	8.81	9.07	**9.00**	0.09
SPY	20.17	21.04	**20.71**	0.27	20.22	21.04	20.74	0.25	20.49	21.04	20.92	0.15	20.21	21.04	20.81	0.27
C-OIL	38.20	39.94	**39.28**	0.59	38.57	39.95	39.40	0.48	38.25	39.96	39.43	0.50	38.18	39.96	39.44	0.51
Average	27.94	28.89	28.48	0.30	27.80	28.89	28.48	0.34	28.02	28.89	28.62	0.25	21.05	21.77	**21.53**	0.23

Table 10.7 Wilcoxon signed rank test of models differences on average AV for one day ahead prediction. Level of significance $\alpha = 0.05$.

Comparison	**R+**	**R-**	**Mean Difference**	**z-value**	**W-value**	**result**
FL-SMIA versus FLNN	36	9	19.83	-1.5993	9	not significant at $p < .05$
FL-SMIA versus MLP	34	11	19.86	-1.3624	11	not significant at $p < .05$
FL-SMIA versus M-FL-SMIA	26	19	19.72	-0.4146	19	not significant at $p < .05$
M-FL-SMIA versus FLNN	33	12	12.74	-1.2439	12	not significant at $p < .05$
M-FL-SMIA versus MLP	35	10	12.77	-1.4809	10	not significant at $p < .05$
FLNN versus MLP	21.5	23.5	19.72	-0.1185	21.5	not significant at $p < .05$

Table 10.8 Mann-Whitney U tests for AV values over 20 simulations.

Comparison	**U-value**	**Z-score**	**P-value**	**results**
FL-SMIA versus MLP (US/UK)	173	-0.71683	0.47152	not significant at $p < .05$
FL-SMIA versus MLP (DJC)	119	-2.17753	0.02926	**significant at $p < .05$.**
M-FL-SMIA versus MLP (NQC)	192	-0.20288	0.84148	not significant at $p < .05$
M-FL-SMIA versus MLP (GOLD)	0	-5.39649	0.00001	**significant at $p < .05$.**

10.2.3 Mean Square Error (MSE)

For the MSE, the comparison of results in table 10.9 illustrates that the MLP produces the lowest average MSE results for five data sets and the average over the datasets. The variation between datasets for the same model is mostly within an order of magnitude and the variation between models is small, with the exception of the M-FL-SMIA which produces consistently higher AV values than the other models. The standard deviations tended to be smaller than the values, indicating some consistency in the MSE performance.

Consequently, the Wilcoxon test for model differences over datasets shows significance for all comparisons involving M-FL-SMIA in table 10.10.

Out of the cases where the SMIA models performed best on individual datasets, only FL-SMIA on DJC and the M-FL-SMIA on SPY were significantly better than the MLP.

Table 10.9 The average results for the mean MSE-Testing over 20 simulations for one day ahead prediction

Data	FLNN				MLP				FL-SMIA				M-FL-SMIA			
sets	Min	Max	Avg	Sd	Min	Max	Avg	Sd	Min	Max	Avg	Sd	Min	Max	Avg	Sd
US/UK	0.0053	0.0071	0.0059	0.0005	0.0052	0.0068	0.0056	0.0004	0.0010	0.0085	**0.0048**	0.0028	0.0087	0.2386	0.0673	0.0696
US/EU	0.0064	0.0116	0.0078	0.0011	0.0060	0.0066	0.0061	0.0002	0.0038	0.0069	**0.0056**	0.0009	0.0212	0.2521	0.1378	0.0701
JP/US	0.0044	0.0194	0.0070	0.0033	0.0041	0.0051	**0.0043**	0.0003	0.0050	0.0099	0.0077	0.0018	0.0085	0.1849	0.0489	0.0376
NQC	0.0026	0.0064	0.0032	0.0008	0.0025	0.0032	**0.0027**	0.0002	0.0040	0.0378	0.0086	0.0086	0.0049	0.2134	0.0447	0.0581
DJC	0.0027	0.0647	0.0069	0.0133	0.0025	0.1386	0.0095	0.0296	0.0031	0.0098	**0.0053**	0.0018	0.0107	0.3083	0.1202	0.0841
OIL	0.0035	0.0073	0.0043	0.0009	0.0034	0.0056	**0.0041**	0.0007	0.0034	0.0034	0.0034	0.0000	0.0167	0.3123	0.0780	0.0758
GOLD	0.0021	0.0056	0.0031	0.0009	0.0019	0.0020	**0.0020**	0.0000	0.0029	0.0096	0.0066	0.0021	0.0068	0.2177	0.0639	0.0606
SPY	0.0013	0.0042	0.0024	0.0009	0.0012	0.0016	0.0013	0.0001	0.0018	0.0090	0.0057	0.0021	0.0011	0.0013	**0.0012**	0.0000
C-OIL	0.0022	0.0040	0.0027	0.0005	0.0021	0.0104	**0.0026**	0.0018	0.0031	0.0100	0.0070	0.0020	0.0026	0.0576	0.0092	0.0116
Average	0.0034	0.0145	0.0048	0.0025	0.0032	0.0200	**0.0042**	0.0037	0.0031	0.0117	0.0061	0.0024	0.00090	0.1985	0.0635	0.0519

Table 10.10 Wilcoxon signed rank test of model differences on MSE for one day ahead prediction.

Comparison	R+	R-	Mean Difference	z-value	W-value	result
FL-SMIA versus FLNN	31	14	0	-1.007	14	not significant at $p < .05$
FL-SMIA versus MLP	34	11	0	-1.3624	11	not significant at $p < .05$
FL-SMIA versus M-FL-SMIA	2	43	-0.13	-2.4286	2	**significant at $p < .05$**
M-FL-SMIA versus FLNN	44	1	0.06	-2.5471	1	**significant at $p < .05$**
M-FL-SMIA versus MLP	44	1	0.06	-2.5471	1	**significant at $p < .05$**
FLNN versus MLP	37	8	0	-1.7178	8	not significant at $p < .05$

10.2.4 Overall Comparison

Overall, the MLP performs best or tied on average on most datasets by RP, AV and MES. However, the FL-SMIA has the best results for US/UK exchange rates and for DJC across all three metrics. Although the difference betwen the FL-SMIA and MLP

Table 10.11 Mann-Whitney U test for MSE values for one day ahead.

Comparison	U-value	Z-score	P-value	results
FL-SMIA versus MLP (US/UK)	185	0.39223	0.69654	not significant at $p < .05$
FL-SMIA versus MLP (US/EU)	135	-1.74473	0.08186	not significant at $p < .05$
FL-SMIA versus MLP (DJC)	23	4.77434	0.00001	**significant at $p < .05$.**
M-FL-SMIA versus MLP (SPY)	57	-3.85464	0.00012	**significant at $p < .05$.**

is not alwast significant, there is some evidence that the FL-SMIA offers and advantage for these time series. Further tests would be advisable before using these models to confirm or reject this results for practical use.

10.3 Five Days Ahead Prediction

In this section, the results for five days ahead prediction are listed and discussed.

The parameters resulting from the grid search that produced the highest RP on average on the validation set are shown in table 10.12. Again, LR is the learning rate, MOM is the momentum, DR is the decay rate, and H-U refers to the number of hidden units in the hidden layer. FLNN, FL-SMIA, and M-FL-SMIA are second order models.

Table 10.12 The parameters values that have been used to predict average results for five days ahead prediction

Data-sets	FLNN			MLP				FL-SMIA				M-FL-SMIA			
	LR	MOM	DR	LR	MOM	DR	H-U	LR	MOM	DR	H-U	LR	MOM	DR	H-U
US/UK	0.4	0	0	0.4	0.01	0.01	6	0.4	0.01	0.001	40	0.04	0.4	0.0005	55
US/EU	0.1	0.4	0.1	0.4	0	0.0001	8	0.03	0.4	0.001	59	0.03	0	0.0001	74
JP/US	0.1	0.4	0	0.4	0.01	0.0001	12	0.03	0.03	0.01	71	0.01	0.01	0.0005	86
NQC	0.4	0.03	0.01	0.1	0.03	0.01	12	0.01	0.01	0.01	49	0.03	0.03	0.0005	64
DJC	0.4	0.03	0.0005	0.4	0.01	0	12	0.4	0.01	0.001	47	0.01	0.4	0.001	62
OIL	0.4	0.03	0.0005	0.4	0	0.1	8	0.03	0.4	0.001	86	0.03	0.1	0.0005	101
GOLD	0.4	0.03	0.01	0.4	0	0.0005	12	0.4	0.03	0	56	0.1	0.01	0.0001	71
SPY	0.4	0.01	0.001	0.4	0	0.01	6	0.4	0.4	0.1	45	0.4	0.03	0.001	60
C-OIL	0.1	0.4	0.001	0.4	0	0.0001	6	0.4	0	0.01	47	0.1	0.1	0.0005	62

10.3.1 Relative Profit (RP)

The comparison results for the mean RP results between the proposed networks (FL-SMIA, and M-FL-SMIA) and the existing networks (FLNN and MLP) showed that MLP produces the highest RP across datsets, with some margin over FLNN, FL-SMIA and M-FL-SMIA.

There is much variation betweeen simulations as shown by the SD values that are higher than the RP values on average (RP can be negative). Variation between models and datasets is comparatively high with no clear trends. Accordingly, the tests in table 10.14 show not significant differences between models.

For the two datasets where SMIA models had the highest RP values, Mann-Whitney U tests (table 10.15) showed not significant difference to the MLP model, although FL-SMIA on SPY produces higher RP by some margin (however, less than the SD).

Table 10.13 The average results for the mean RP over 20 simulations for five days ahead prediction

Data sets	FLNN				MLP				FL-SMIA				M-FL-SMIA			
	Min	Max	Avg	Sd	Min	Max	Avg	Sd	Min	Max	Avg	Sd	Min	Max	Avg	Sd
US/UK	-63.29	79.31	**38.65**	35.69	-36.79	77.52	30.33	37.76	-3.26	79.01	29.07	23.32	-61.46	75.39	14.88	40.59
US/EU	-37.39	70.14	**31.01**	28.64	-47.86	72.07	26.53	40.49	-44.02	68.12	21.70	31.84	-60.04	74.43	20.54	36.40
JP/US	-41.11	70.40	31.07	34.41	-56.20	70.67	**33.19**	38.32	-62.66	69.82	17.37	40.88	-20.77	75.05	28.96	30.58
NQC	-44.89	63.33	**21.67**	31.80	-36.09	60.61	21.26	33.02	-55.13	62.38	15.67	32.00	-61.45	63.52	15.76	37.22
DJC	-64.59	65.01	**25.83**	44.44	-64.87	64.55	24.62	39.34	-60.87	58.26	12.09	37.44	-35.45	61.04	21.50	26.17
OIL	-67.27	71.14	23.89	45.17	-69.41	73.71	**33.96**	43.62	-53.37	70.60	30.55	36.17	-70.46	71.31	25.29	41.04
GOLD	-55.41	73.44	22.04	45.13	-48.56	74.75	**30.88**	40.03	-65.45	74.38	23.77	41.03	-69.88	75.34	21.40	43.55
SPY	-68.00	67.62	15.04	40.84	-53.28	65.53	24.72	35.51	-8.58	66.40	**35.09**	15.71	-54.51	64.28	29.17	35.92
C-OIL	-61.63	57.63	10.50	34.47	-68.56	65.66	21.92	39.02	-36.52	69.92	25.63	32.51	-67.26	68.11	**25.88**	39.73
Average	-55.96	68.67	24.41	37.84	-53.51	69.45	**27.49**	38.57	-43.32	68.77	23.44	32.32	-55.70	69.83	22.60	36.80

Table 10.14 Wilcoxon signed rank test for avg RP for five days ahead prediction. Level of significance $\alpha = 0.05$.

Comparison	R+	R-	Mean Difference	z-value	W-value	result
FL-SMIA versus FLNN	21	24	-7.57	-0.1777	21	not significant at p < .05
FL-SMIA versus MLP	10	35	-3.09	-1.4809	10	not significant at p < .05
FL-SMIA versus M-FL-SMIA	27	18	2.9	-0.5331	18	not significant at p < .05
M-FL-SMIA versus FLNN	21	24	-7.57	-0.1777	21	not significant at p < .05
M-FL-SMIA versus MLP	10	35	-1.4809	-3.09	10	not significant at p < .05
FLNN versus MLP	12	33	-2.12	-2.2439	12	not significant at p < .05

Table 10.15 The Mann-Whitney U Test for 20 average RP values of data-sets, level of significance $\alpha = 0.05$

Comparison	U-value	Z-score	P-value	results
FL-SMIA versus MLP (SPY)	185	0.39223	0.69654	not significant at p < .05
M-FL-SMIA versus MLP (C-OIL)	175	0.66273	0.50926	not significant at p < .05

10.3.2 Annualised Volatility (AV)

The picture for AV in five days ahead prediction in table 10.16 is one of low variation between datasets and models. Overall, MLP has the lowest AV and FL-SMIA the highest, but within 2% difference on average.

Therefore is it not surprising that only the FL-SMIA vs MLP comparison over datasets leads to a significant difference in the Wilcoxon tests in table 10.17. The Mann-Whitney U tests show no significant differences in table 10.18.

Table 10.16 The average results for the mean AV over 20 simulations for five days ahead prediction

Data sets	FLNN				MLP				FL-SMIA				M-FL-SMIA			
	Min	Max	Avg	Sd	Min	Max	Avg	Sd	Min	Max	Avg	Sd	Min	Max	Avg	Sd
US/UK	19.11	23.25	**21.50**	1.31	19.31	23.25	21.78	1.23	20.95	23.27	22.59	0.72	20.06	23.27	22.16	1.12
US/EU	20.13	23.58	22.39	1.11	19.91	23.62	**22.00**	1.12	20.34	23.62	22.60	1.10	19.64	23.62	22.41	1.31
JP/US	21.53	25.23	23.66	1.35	21.49	25.23	**23.35**	1.25	21.59	25.20	23.79	1.27	21.75	25.23	24.03	1.18
NQC	46.14	51.86	49.80	1.79	46.65	51.85	49.71	1.83	46.32	51.66	50.10	1.63	46.11	51.85	**49.56**	2.27
DJC	39.99	44.87	**41.92**	1.47	40.01	44.54	42.51	1.55	40.64	44.99	43.23	1.38	40.62	44.97	43.70	1.30
OIL	79.36	95.14	87.17	5.65	78.07	94.77	**85.80**	5.31	79.62	95.14	88.30	5.98	79.27	94.06	88.08	5.48
GOLD	126.75	151.56	140.28	8.98	125.76	151.67	**140.13**	8.94	126.05	151.66	141.56	8.60	127.56	151.59	141.57	9.23
SPY	38.04	44.23	41.76	2.06	38.51	44.19	41.81	1.77	40.08	44.22	42.35	0.96	38.74	44.14	**41.45**	1.68
C-OIL	78.71	88.98	85.56	3.07	76.07	88.93	83.62	4.16	75.50	88.98	84.42	3.91	76.25	88.98	**82.91**	4.93
Average	52.19	60.97	57.12	2.98	51.76	60.89	**56.75**	3.02	52.34	60.97	57.66	2.84	52.22	60.86	57.32	3.17

Table 10.17 Wilcoxon signed rank test for avg AV for five days ahead prediction. Level of significance $\alpha = 0.05$.

Comparison	R+	R-	Mean Difference	z-value	W-value	result
FL-SMIA versus FLNN	38	7	35.27	-1.8363	7	not significant at $p < .05$
FL-SMIA versus MLP	45	0	35.66	-2.6656	0	**significant at p < .05**
FL-SMIA versus M-FL-SMIA	34	11	35.25	-1.3624	11	not significant at $p < .05$
M-FL-SMIA versus FLNN	31	14	34.93	-1.007	14	not significant at $p < .05$
M-FL-SMIA versus MLP	36	9	35.32	-1.5993	9	not significant at $p < .05$
FLNN versus MLP	33	12	35.12	-1.2439	12	not significant at $p < .05$

Table 10.18 The Mann-Whitney U Test for 20 average AV values of data-sets, level of significance $\alpha = 0.05$

Comparison	U-value	Z-score	P-value	results
M-FL-SMIA versus MLP (NQC)	194	0.14878	0.88076	not significant at $p < .05$
M-FL-SMIA versus MLP (SPY)	173	-0.71683	0.47152	not significant at $p < .05$
M-FL-SMIA versus MLP (C-OIL)	187	-0.33813	0.72786	not significant at $p < .05$

10.3.3 Mean Square Error (MSE)

Table 10.9 shows that the MLP network produces the lowest MSE on most data-sets. However on NQC, its MSE is much higher that that of FLNN, which therefore has the lowest average MSE over the data-sets. The MSE of the SMIA models is around an order or magnitude higher that that of FLNN and MLP.

Accordingly the overall comparisons show in table 10.20 that both SMIA models are significantly worse than MLP and FL-SMIA is significantly worse than FLNN.

On individual datasets, FL-SMIA has a good MSE on SPY, which is sginficantly lower than MLP, although the absolute difference is small (.0017 vs .0018).

10.3.4 Overall Comparison

The picture for five day ahead prediction is even less clear than that for one day ahead prediction. The MLP model seems to perform best on RP, FLNN best on AV and

Table 10.19 The average results for the mean MSE-Testing over 20 simulations for five days ahead prediction

Data sets	FLNN				MLP				FL-SMIA				M-FL-SMIA			
	Min	Max	Avg	Sd	Min	Max	Avg	Sd	Min	Max	Avg	Sd	Min	Max	Avg	Sd
US/UK	0.0051	0.0083	0.0063	0.0009	0.0048	0.0054	**0.0051**	0.0002	0.0068	0.0261	0.0122	0.0047	0.0092	0.3378	0.0659	0.0957
US/EU	0.0077	0.0090	0.0081	0.0003	0.0078	0.0083	**0.0079**	0.0001	0.0163	0.2028	0.0879	0.0568	0.0160	0.3118	0.1029	0.0935
JP/US	0.0072	0.0203	0.0103	0.0039	0.0069	0.0076	**0.0071**	0.0002	0.0085	0.3210	0.1151	0.1113	0.0166	0.3084	0.1160	0.0856
NQC	0.0044	0.0072	**0.0051**	0.0008	0.0042	0.2692	0.0271	0.0682	0.0125	0.2660	0.0998	0.0804	0.0084	0.2510	0.0707	0.0785
DJC	0.0032	0.0057	0.0041	0.0007	0.0030	0.0036	**0.0032**	0.0001	0.0049	0.0481	0.0136	0.0098	0.0106	0.2993	0.1055	0.0992
OIL	0.0051	0.0141	0.0072	0.0026	0.0047	0.0057	**0.0050**	0.0003	0.0066	0.3063	0.0812	0.0783	0.0179	0.3248	0.1304	0.0953
GOLD	0.0052	0.0076	0.0060	0.0006	0.0052	0.0079	0.0058	0.0008	0.0072	0.2971	**0.0035**	0.0621	0.0089	0.3062	0.0573	0.0775
SPY	0.0018	0.0042	0.0024	0.0006	0.0017	0.0022	0.0018	0.0001	0.0017	0.0018	**0.0017**	0.0000	0.0022	0.0224	0.0094	0.0057
C-OIL	0.0034	0.0345	0.0057	0.0067	0.0031	0.0038	**0.0033**	0.0002	0.0036	0.0189	0.0079	0.0041	0.0045	0.1915	0.0367	0.0532
Average	0.0048	0.0123	**0.0061**	0.0019	0.0046	0.0349	0.0074	0.0078	0.0076	0.1653	0.0470	0.0453	0.0105	0.2615	0.0772	0.0760

Table 10.20 Wilcoxon signed rank test for avg MSE for five days ahead prediction. Level of significance $\alpha = 0.05$.

Comparison	R+	R-	Mean Difference	z-value	W-value	result
FL-SMIA versus FLNN	41	4	0.04	-2.1917	4	not significant at p < .05
FL-SMIA versus MLP	42	3	0.04	-2.3102	3	**significant at p < .05**
FL-SMIA versus M-FL-SMIA	5	40	-0.06	-2.0732	5	not significant at p < .05
M-FL-SMIA versus FLNN	45	0	0.07	-2.6656	0	**significant at p < .05**
M-FL-SMIA versus MLP	45	0	0.07	-2.6656	0	**significant at p < .05**
FLNN versus MLP	36	9	0	-1.5993	9	not significant at p < .05

Table 10.21 The Mann-Whitney U Test for 20 average MSE values of data-sets, level of significance $\alpha = 0.05$

Comparison	U-value	Z-score	P-value	results
FL-SMIA versus MLP (GOLD)	165	0.93323	0.35238	not significant at p < .05
FL-SMIA versus MLP (SPY)	101	-2.66443	0.00782	**significant at p < .05.**

MSE, but few differences are significant, with the exception of MSE, where the SMIA models perform markedly worse on average.

The only data-set where using a SMIA model could be of benefit is SPY, where the RP is markedly higher than other models, MSE is lower, and AV is similar to the best model. However, the RP difference is not significant (see row 2 in table 10.15), so that additional experiments would be needed to confirm.

10.3.5 Training Times

The training timed for each model for five days ahead are listed in table 10.22. The training time for one day ahead is very similar.

Overall, the training times for the SMIA models are longer, with the M-FL-SMIA using even more time, which is as expected since it combines a SMIA with an FLNN structure.

Table 10.22 Training times for running a search with 4 models and 9 data-sets over a grid with $(9 \times 5 \times 5 \times 5)$ parameter combinations (not all apply to all models, see section 2.3.4), with 20 simulations of 80 epochs each.

Data sets	FLNN	MLP	FL-SMIA	M-FL-SMIA
US/UK	1:35:19	2:15:42	2:09:20	2:30:44
US/EU	1:48:10	2:29:32	2:24:12	2:49:42
JP/US	1:46:53	2:30:14	2:25:17	2:36:58
NQC	1:28:58	2:32:47	2:19:49	2:24:38
DJC	1:45:49	2:26:20	2:10:40	2:36:12
OIL	1:52:11	2:52:31	2:47:32	2:55:56
GOLD	1:22:57	2:49:30	2:45:23	2:58:59
SPY	2:37:17	3:14:38	3:13:28	3:42:55
C-OIL	2:32:49	3:29:13	3:25:10	3:38:53

10.4 Discussion

In this chapter, an alternative method for prediction (single values), hyper-parameter selection and evaluation (based on average performances) and data from more tradeable assets have been used to provide as assessment of the models that is more related to real life scenarios.

The results on the whole do not replicate the positive results for the SMIA models that we found in the previous chapters. It seems that MLP is superior in most metrics. Since MLP is also a simpler and less computationally expensive model, it would in most cases be preferable for applications.

There are a few data-sets where FL-SMIA models show better results than existing models (1 day US/UK and DJC and 5 days SPY). These differences would need to be confirmed in further test and their benefit assessed against additional cost through longer training, but the performance gain may be worth it.

10.5 Example of a Simple Trading Strategy

In this section, a Simple Trading Strategy will be discussed as well as, the simulation results for four models (FLNN, MLP, FL-SMIA, and M-FL-SMIA) using three data-sets (SPY, GOLD, and US/UK) will be listed. The example aim to explain a simple trading strategy that used to start an investment with 100 dollars as in the following:

1. If we start with 100 dollars, at every time step, we need to calculate the expected change by using a neural network model(e.g. FL-SMIA model)

2. If the expected change is negative, that means a ‘sell’ signal. While if the expected change is positive, that represents a ‘buy’ signal. Based on the expected changes in the stock prices then we could buy or sell accordingly.

3. If we have already bought the stock and then get another buy signal, we hold the stock and take not action. Analogously, if we have already sold the stock and get a sell signal, we take no action.

4. At the next time step, if we hold the stock we adjust the value according to the new price. Otherwise, we keep the money, i.e. our account remains unchanged.

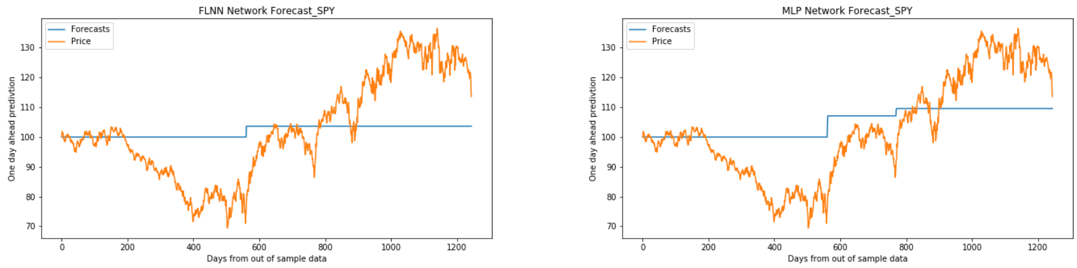
Table 10.23 Simulation results (100 dollars) based on the models

Data-sets	FLNN				MLP				FL-SMIA				M-FL-SMIA			
	Profit	MDD	SNR	CDC	Profit	MDD	SNR	CDC	Profit	MDD	SNR	CDC	Profit	MDD	SNR	CDC
SPY	103.65	-67.51	18.27	49.68	109.59	-56.21	21.02	50.00	123.61	-36.39	10.41	49.76	131.02	-33.61	20.76	51.85
GOLD	104.69	-203.52	17.38	53.63	149.23	-129.95	23.76	45.48	153.77	-119.64	24.83	58.22	132.82	-39.91	10.75	46.37
US/UK	109.40	-4.32	20.95	50.51	103.39	-5.93	20.74	54.80	101.34	-5.94	8.67	55.81	106.06	-7.02	12.16	52.78

The results in table 10.23 include the profits, Maximum Draw Down (MDD), Signal to Nose Ratio (SNR), and Correct Directional Change (CDC) values. In general, the results indicated that the proposed models (FL-SMIA and M-FL-SMIA produced the highest profits for SPY and GOLD data. For MDD results, which refer to the maximal trading loss, the M-FL-SMIA produces better results than all other networks with the SPY and GOLD data-sets, while FLNN produced the best MDD on the US/UK data.

Although the results for other metrics of the models are varied, in some cases it reflects useful information regarding the performance of the models. The MLP model produces consistently good SNR results, i.e. a clearer reading of the signal. However it is outperformed by the Fl-SMIA and FLNN models when using the GOLD and US/UK data-sets respectively. The Correct Directional Change (CDC) results prove that FL-SMIA outperformed the other models when using GOLD and US/UK data-sets, while M-FL-SMIA model produced the highest CDC value than the other models when using the SPY data.

Figures 10.5 to 10.7 illustrate the simulation results for three datasets (SPY, GOLD, US-UK). The simulation line moves when the price forecast increases (positive signals) and keeps horizontal, which means holding the stock, when a price reduction is forecast (negative signal).



for FLNN and MLP.]SPY data (orange) and trading simulation (blue) for the FLNN model (left) and the MLP model (right).

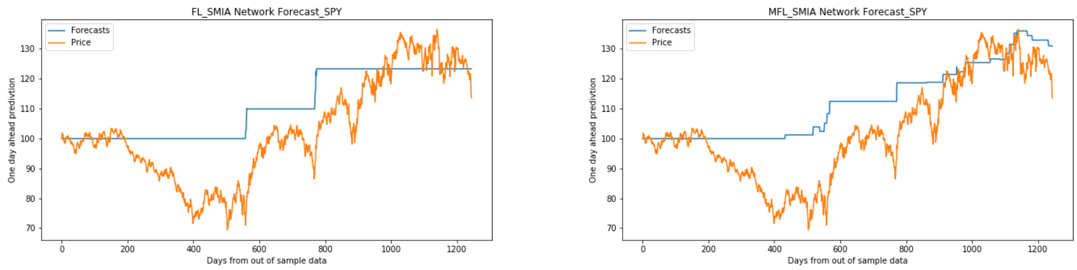


Fig. 10.3 SPY data (orange) and trading simulation (blue) for the FL-SMIA model (left) and the M-FL-SMIA model (right).

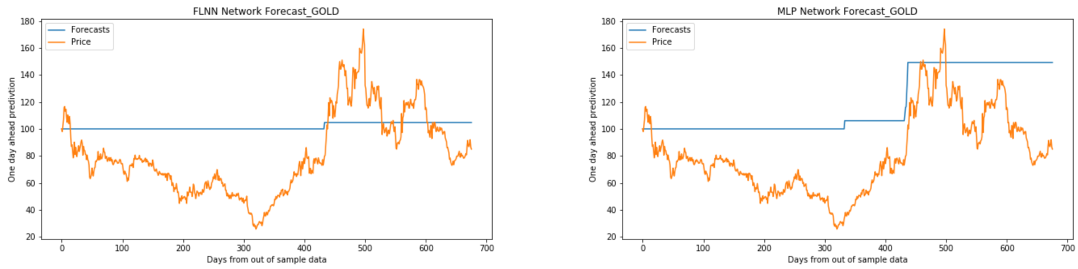


Fig. 10.4 GOLD data (orange) and trading simulation (blue) for the FLNN model (left) and the MLP model (right).

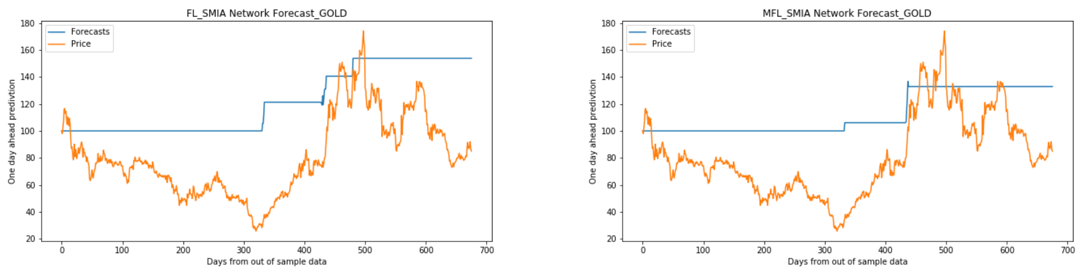


Fig. 10.5 GOLD data (orange) and trading simulation (blue) for the FL-SMIA model (left) and the M-FL-SMIA model (right).

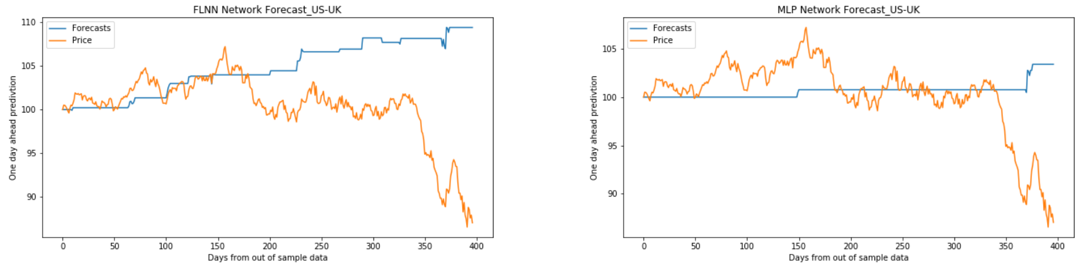


Fig. 10.6 US-UK data (orange) and trading simulation (blue) for the FLNN model (left) and the MLP model (right).

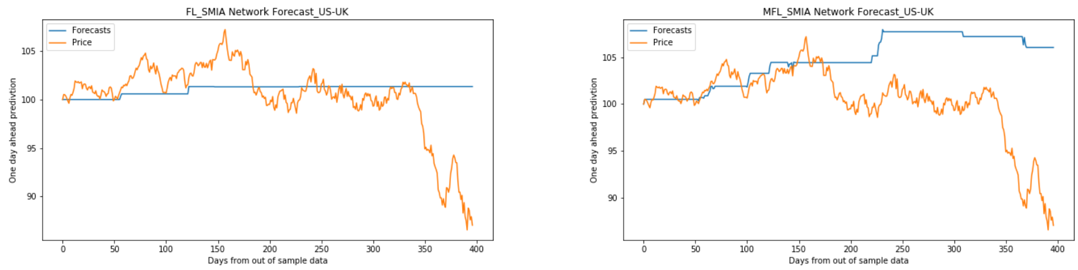


Fig. 10.7 US-UK data (orange) and trading simulation (blue) for the FL-SMIA model (left) and the M-FL-SMIA model (right).

Chapter 11

Conclusions and Future Work

In this chapter, all prediction results which have been illustrated in the previous chapters for all networks that have been proposed in this research will be summarised and directions for future work will listed.

In this research, the novel FL-SMIA model (Functional Link with Self-organised Multilayer neural network using the Immune Algorithm) is proposed. Further, as extensions of FL-SMIA network the FL-SMIA*, D-FL-SMIA, MD-FL-SMIA, MD-FL-SMIA-2, M-FL-SMIA, and FL-SMIA-RBM have been proposed for financial prediction.

Three different types of financial data have been used in order to investigate the ability of the proposed networks for the prediction. The data-sets including: exchange rates (USD/UKP, USD/EUR, JPY/USD), stock price indices (NASDAQ, DJIA), and commodity prices (OIL and GOLD) as well as more tradeable alternatives for indices and commodities (SPY and C-OIL).

The prediction ability for all proposed networks have been tested for one day ahead prediction and five days ahead prediction. The prediction results have also been compared with multi-layer neural networks (MLP, and SMIA) and one model of Higher order neural network (FLNN).

The Wilcoxon signed rank test and Mann Whitney U tests have been used to test the significance differences between models in this research. Additionally, the correlation coefficient was applied in order to determine the similarity between the networks behaviour for financial prediction.

The conclusions for all results will be provided in the following sections.

11.1 Conclusions for the Prediction for One Day Ahead

This section discusses the results for one day ahead prediction focusing on financial prediction results, Relative Profit (RP) and Annualised Volatility (AV).

The RP results indicated that the FLNN model outperforms all other models for six data-sets from nine data-sets. however, the proposed models and the MLP model still on it competes with the FLNN model to predict the highest profits. The proposed models in some cases produce the highest RP than FLNN and MLP models such as the prediction results when used DJC, OIL and GOLD data.

The AV results showed that the FLNN model reduces the AV values for six data-sets, however, it is on competing with the proposed models. The proposed models (FL-SMIA, and D-FL-SMIA) produced the lowest AV results for the rest three data-sets (DJC, GOLD, and OIL) respectively.

As most of the results for the metrics of the models are varied, the prediction results that have been presented in chapter 8 and 9 could be concluded based on the metrics and the average results for the data-set as in the following:

1. **Relative Profit (RP)**

   The prediction results showed that the proposed models which use the Immune Algorithm and the method of the inputs and their products both contribute to improving the networks' performance for financial time series prediction compared to other multi-layer networks (MLP and SMIA).

   The average RP results for one day ahead prediction indicated that the FL-SMIA network and their extensions (FL-SMIA*, D-FL-SMIA, MD-FL-SMIA, MD-FL-SMIA-2, M-FL-SMIA, and FL-SMIA-RBM) generally outperform the MLP network.

   Consequently, the average results proved that the use of inputs and their products with the Immune learning Algorithm improves the performance of multi-layer networks.

2. **Annualised Volatility (AV)**

   The average results of Annualised Volatility (AV) indicated that the FL-SMIA network produced the lowest value of average AV of all networks. In other words, using the proposed model (FL-SMIA) for financial prediction help in reducing the investment risk more than all other models that used in this research.

3. **Mean Squared Error (MSE)**

   The average results of MSE-Testing showed that although the SMIA network outperformed all networks by reducing the MSE-Testing when produced the average MSE-Testing value (0.00218), the proposed networks FL-SMIA, FL-SMIA* outperformed all other networks with the average of the MSE-Testing (0.00227, and 0.00283) respectively. While the FLNN network produced the average of MSE-Testing (0.00695). When focusing on the average of the MSE-Testing results, could be noticed that the proposed network FL-SMIA-RBM outperformed the FLNN model with the average of MSE-Testing value (0.00497 vs 0.00695). In addition, the proposed network FL-SMIA-RBM produced the lowest average of the MSE-Testing value than the proposed networks (D-FL-SMIA, MD-FL-SMIA, MD-FL-SMIA2, and M-FL-SMIA).

4. **Mean Absolute Error (MAE)**

   The results of MAE indicated that the FL-SMIA network competes with the SMIA and FL-SMIA* networks on reducing the MAE results. However, the FL-SMIA network outperformed all networks when predicting four data-sets.

   The average results of MAE show that on one hand, the FL-SMIA network outperformed all other networks with a decrease in the average value of MAE to (0.03033). On the other hand, the comparison results between the proposed networks indicated that FL-SMIA, FL-SMIA*, and FL-SMIA-RBM networks reduced the average results of MAE more than all other proposed networks.

5. **Maximum Draw-Down (MDD)**

   The average results of Maximum Draw-Down (maximum of trading loss) showed that the SMIA and FL-SMIA models produced lower values of average results of MDD than all other networks. The results proved that the maximum of trading loss could be reached by using the SMIA and FL-SMIA models for financial prediction.

6. **Correct Directional Change (CDC)**

   The average results of the CDC indicated that the SMIA and FL-SMIA networks reached the highest values (65.721, and 65.010) than all other networks that used in this research (The higher value of Correct Directional Change (CDC) is the target).

7. **Signal to Noise Ratio (SNR)**

As the higher ratio value of SNR refers to a clearer reading of the signal, the results proved that the higher values of average SNR were produced by using the SMIA and FL-SMIA networks(26.327 and 26.233) respectively.

The results in chapter 10 illustrated a different picture of the proposed models (FL-SMIA and M-FL-SMIA) than the results which have been listed in chapter 8. The reason behind that is predicting the prices in chapter 10 instead of using the relative difference in the percentage of the price (RDP) method for the prediction in chapter 8.

The prediction results in chapter 10 indicated that the MLP model produced the highest average results for RP for five data-sets from nine the nine data-sets. While the FL-SMIA model produced the highest RP average values than all other models when used the US/UK and DJC data. Also, the FLNN model outperformed all other models when used two data-sets (JP/US and SPY).

For the AV and MSE, the results showed that each model could reduce the AV and MSE values for two data-sets, however, the MLP model produced better results than the other models.

11.2 Conclusions for the Prediction for Five Days Ahead

In this section, the prediction results for five days ahead prediction with focusing on financial prediction results, Relative Profit (RP) and Annualised Volatility (AV) will be discussed.

The Relative Profit (RP) results for five days ahead prediction proved that using the immune learning algorithm help on improving the performances of the proposed models and lead to producing the highest RP values than all other networks including the FLNN network for all data-sets except the GOLD data. The proposed network (M-FL-SMIA) predicted the highest RP results for five data-sets from nine comparing to all other network results.

For the Annualised Volatility (AV), the results indicated that the proposed models (FL-SMIA, MD-FL-SMIA-2, and M-FL-SMIA) outperformed all other networks when reduced the AV values for the data-sets.

As most metrics results for the models are varied depend on the data type, the prediction results for five days ahead which presented in chapter 8 and 9 could be concluded based on the metrics and the average results for the data-set as in the following:

1. **Relative Profit (RP)**

   The average RP results for five days ahead prediction proved that all the proposed network (FL-SMIA, FL-SMIA*, D-FL-SMIA, MD-FL-SMIA, MD-FL-SMIA-2, and M-FL-SMIA except FL-SMIA-RBM network) outperformed the MLP network.

   The FLNN, FL-SMIA and M-FL-SMIA models produced better RP results than other networks by using the product terms (inputs and their products), while the FL-SMIA-RBM does not outperform other networks.

2. **Annualised Volatility (AV)**

   The average results of Annualised Volatility (AV) proved that FLNN network and M-FL-SMIA network minimize investment risk as it produced the lowest values of average AV than all other networks.

3. **Mean Squared Error (MSE)**

   The MSE-Testing average results showed that the multi-layer networks which use the immune learning algorithm (FL-SMIA*, FL-SMIA, SMIA) outperformed all other networks with the average of values (0.00203, 0.00222, and 0.00268) respectively. While the FLNN network produced the average of MSE-Testing values (0.00758).

   As the average results of the MSE-Testing of the proposed networks (FL-SMIA*, FL-SMIA) and the SMIA network proved an improvement of the performance for the multi-layer networks, this result emphases that using the immune learning algorithm give the reason behind the improvements of prediction ability of the multi-layer networks.

4. **Mean Absolute Error (MAE)**

   The average results of MAE illustrated that the FL-SMIA network outperformed all other networks with the average value of MAE equal to 0.02506.

   For the other proposed networks, the average results of MAE showed that only FL-SMIA* network compete with the FL-SMIA network.

5. **Maximum Draw-Down (MDD)**

The average results of maximum Draw-Down (MDD) proved that the FLNN network followed by the M-FL-SMIA network, produced the lower values of MDD when compared to all other networks. However, the FL-SMIA network produced a lower average value of MDD than the MLP network and all proposed networks except the M-FL-SMIA network.

6. **Correct Directional Change (CDC)**

   The conclusion of the average results of the CDC showed that the FL-SMIA network outperformed all other networks with the CDC value (63.427).

7. **Signal to Noise Ratio (SNR)**

   The average results for SNR proved that the FLNN network outperformed all other networks. While the comparison between the multi-layer networks showed that the FL-SMIA network produced a higher value of average SNR than all the multi-layer networks. The FL-SMIA* network competes with the networks SMIA network to reach the second highest average for SNR values between the multi-layer networks.

The prediction results in chapter 10 showed that the FLNN model outperformed the other models by producing the highest average results for RP for four data-sets from the nine data-sets. Also, the MLP model outperformed all other models for three data-sets. While each of the proposed models (FL-SMIA and M-FL-SMIA) produced the highest RP average values than all other models only when used the SPY and C-OIL data respectively.

For the Annualised Volatility (AV), the results showed that each model could reduce the AV values, however, the MLP model outperformed all other models. The MLP model produced the lowest AV average results than the other models for four data-sets. The proposed model M-FL-SMIA also outperformed the other models by produced the lowest average AV values for the other three data-sets.

The results for Mean Squared Error (MSE) in chapter 10 illustrated the ability of the models to reduce the prediction error. Although the proposed models could not reduce the MSE average values such as MLP and FLNN model, the proposed FL-SMIA model outperformed all other models when reduced the average MSE values for GOLD and SPY data-sets. consequently, the prediction results for five days ahead in chapter 10 showed that the MLP model performed better than the proposed models as well as better than the FLNN model.

11.3 Overall Conclusions

The overall prediction conclusions for all networks that have been used in this research will be discussed in this section.

The performance of the proposed FL-SMIA model and other proposed networks have been evaluated using nine financial data-sets. The conclusions have based on the average results of financial prediction.

Overall, this research concluding the results for the forecasting financial data using different methods based on the results listed in chapter 8 and chapter 10. The prediction results in chapter 8 for one day ahead prediction concluded that the average Relative Profit (RP) results illustrated that although the proposed FL-SMIA model improved the prediction abilities of the multi-layer networks and outperformed the multi-layer network (MLP), the FLNN models produced a higher value of average RP than all other networks. Regarding the investment risk, the results of average Annualised Volatility (AV) proved that the FL-SMIA model reduced the investment risk by produced the lowest (AV) value than all other networks including the FLNN models. Additionally, the average of maximum Draw-Down (MDD) results showed that the lowest values of loses have been produced by the SMIA network followed by the FL-SMIA network. The average results for the MSE-Testing denoted that for one day ahead prediction, the SMIA network followed by the FL-SMIA network produced lower results than all other networks.

Based on the results for one day ahead prediction in this research, it can be concluded that among the multi-layer networks for financial data for one day ahead prediction, the proposed FL-SMIA model is a good choice, as it produced higher profit (average RP) value than all other multi-layer networks and lower loss (average AV) value than all other networks. Although the FLNN model produced a somewhat higher average RP value than all other networks, the FLNN model performed worse than the FL-SMIA model with the average values of AV and MSE for one-day prediction.

Also, the SMIA network and the proposed FL-SMIA model produced lower average values for MSE than all other networks including the FLNN model, which means that it is a good choice for one day ahead prediction to predict a financial data to use the SMIA network or the proposed FL-SMIA model.

The conclusions for the prediction for five days ahead is that on one hand, the average results of RP proved that the M-FL-SMIA model outperformed all other networks (followed by FLNN network). On the other hand, the average AV results denoted that M-FL-SMIA and FLNN networks produced the lowest values of AV than all other networks. While the average MDD showed that the lowest values have been produced by the FLNN model followed by the M-FL-SMIA model to represent the maximum loss of trading.

It is clear from these results, that when deciding to use a network for financial prediction for five days ahead, it is good to use the proposed M-FL-SMIA model because in most cases the M-FL-SMIA model produced higher profits, lower losses and less volatility than all other networks.

For the other domains, the decision on which network should be chosen depends on the MSE results. Therefore, when using non-financial data for five days ahead prediction, the good decision is to use the proposed FL-SMIA* network or the proposed FL-SMIA network, as these networks have produced lower average MSE values than all other networks.

The results in chapter 10 indicated that predicting the prices instead of using the relative difference in the percentage of the price (RDP) method leads to produce results which are not replicate the positive results for the FL-SMIA and other proposed versions which have been illustrated in chapter 8. The results in chapter 10 proved that the MLP model outperforms all other models in most cases when compared to the proposed models. However, in some cases, the FL-SMIA produces better results for RP, AV, and MSE than the MLP and FLNN models. The results for one day ahead prediction showed that the FL-SMIA model outperformed all other models when used two data-sets (US/UK and DJC).

For the five days ahead prediction, the results in chapter 10 indicated that the FL-SMIA model could only predict better results than all models for one data set which is the SPY. while MLP model performed better than the proposed models as well as better than the FLNN model.

Consequently, the differences between the results in chapter 8 and chapter 10 would require more tests and analysis to provide more information which could explain the reasons for the instability of the results.

A general limitation of the models proposed here is that they have only be evaluated on a single output variable. There is, however, nothing that prevents multiple output units in our current models, so that multidimensional models are a natural extension. The results and the needed parametrizations may, however, be different from the one-dimensional case studied here.

Finally, the overall conclusions of this research, resulting in thinking about many ideas related to applying the novel models (FL-SMIA and the M-FL-SMIA)in the future, as these models could be useful for applications in finance and other domains.

11.4 Future Research Directions

The proposed FL-SMIA model and its extensions which were presented and discussed in this research may be used in the future in the domain of financial prediction as well as for other neural network applications.

The proposed networks could be developed and use in further for different domains of science. Based on the conclusions drawn from this research, the following proposals may be worth pursuing for future work:

* Improving the FL-SMIA network by using recurrent connections in the self-organised network and an immune learning algorithm. The use of recurrent connections can support the network with a long memory. Also, Using the recurrent connections with the proposed networks may improve the performance of the proposed networks (FL-SMIA, M-FL-SMIA) for the financial prediction.

* Examining the proposed networks in order to investigate the best choice of the network architecture by using other higher-order for the input layer such as the third-order or utilise the functional expansion model to produce extra input units.

* The proposed networks can be used for further investigation analysing more data-sets in order to validate the performance on different prediction domains.

* Investigating the direct optimisation of AR instead of MSE, in order to directly optimise and balance financial performance.

* Investigating the use of a Self-organising layer after a standard hidden layer, which would require the definition of an error signal for back-propagation from the Self-organising layer.

* Combining the M-FL-SMIA and recurrent networks.

The future research work that proposed in this chapter may find a positive side using extra learning methods or alternative methods lead to improving the performance of neural network models in the financial field or on any other domain.

References

[1] Acuña, G., Ramirez, C., and Curilem, M. (2012). Comparing narx and narmax models using ann and svm for cash demand forecasting for atm. In *The 2012 International Joint Conference on Neural Networks (IJCNN)*, pages 1–6. IEEE.

[2] Agarap, A. F. (2018). Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*.

[3] Alligier, R., Gianazza, D., and Durand, N. (2015). Machine learning and mass estimation methods for ground-based aircraft climb prediction. *IEEE Transactions on Intelligent Transportation Systems*, 16(6):3138–3149.

[4] Altavilla, C. and De Grauwe, P. (2010). Forecasting and combining competing models of exchange rate determination. *Applied Economics*, 42(27):3455–3480.

[5] Ang, A. Q.-X., Yeong, Y. Q., and Wee, W. (2017). Emotion classification from eeg signals using time-frequency-dwt features and ann. *J Comput Commun*, 5(03):75.

[6] Baba, N. and Kozaki, M. (1992). An intelligent forecasting system of stock price using neural networks. In *[Proceedings 1992] IJCNN International Joint Conference on Neural Networks*, volume 1, pages 371–377. IEEE.

[7] Bajari, P., Nekipelov, D., Ryan, S. P., and Yang, M. (2015). Machine learning methods for demand estimation. *American Economic Review*, 105(5):481–85.

[8] Barunik, J., Krehlik, T., and Vacha, L. (2016). Modeling and forecasting exchange rate volatility in time-frequency domain. *European Journal of Operational Research*, 251(1):329–340.

[9] Basak, S., Kar, S., Saha, S., Khaidem, L., and Dey, S. R. (2019). Predicting the direction of stock market prices using tree-based classifiers. *The North American Journal of Economics and Finance*, 47:552–567.

[10] Bebarta, D. K., Rout, A. K., Biswal, B., and Dash, P. (2012). Forecasting and classification of indian stocks using different polynomial functional link artificial neural networks. In *2012 Annual IEEE India Conference (INDICON)*, pages 178–182. IEEE.

[11] Bilbao, I. and Bilbao, J. (2017). Overfitting problem and the over-training in the era of data: Particularly for artificial neural networks. In *2017 Eighth International Conference on Intelligent Computing and Information Systems (ICICIS)*, pages 173–177. IEEE.

[12] Birgul Egeli, A. (2003). Stock market prediction using artificial neural networks. *Decision Support Systems*, 22:171–185.

[13] Bishop, C. M. (1995). *Neural networks for pattern recognition*. Oxford university press.

[14] Brockwell, P. J., Davis, R. A., and Calder, M. V. (2002). *Introduction to time series and forecasting*, volume 2. Springer.

[15] Burnet, F. M. et al. (1957). A modification of jerne's theory of antibody production using the concept of clonal selection. *Australian J. Sci.*, 20(3):67–9.

[16] Carelli, A., Silani, S., and Stella, F. (2000). Profiling neural networks for option pricing. *International Journal of Theoretical and Applied Finance*, 3(02):183–204.

[17] Carpenter, G. A. and Grossberg, S. (1987a). Art 2: Self-organization of stable category recognition codes for analog input patterns. *Applied optics*, 26(23):4919–4930.

[18] Carpenter, G. A. and Grossberg, S. (1987b). A massively parallel architecture for a self-organizing neural pattern recognition machine. *Computer vision, graphics, and image processing*, 37(1):54–115.

[19] Carpenter, G. A. and Grossberg, S. (1991). *Pattern recognition by self-organizing neural networks*. MIT Press.

[20] Carpenter, G. A. and Grossberg, S. (2010). *Adaptive Resonance Theory*, pages 87–90. MIT Press, Cambridge, MA, 2 edition.

[21] Chan, Z. S., Ngan, H., Rad, A. B., David, A., and Kasabov, N. (2006). Short-term ann load forecasting from limited data using generalization learning strategies. *Neurocomputing*, 70(1-3):409–419.

[22] Chatfield, C. (1975). The analysis of time series: theory and practice. *Chapman Hall*.

[23] Claesen, M. and De Moor, B. (2015). Hyperparameter search in machine learning. *arXiv preprint arXiv:1502.02127*.

[24] Coates, A., Ng, A., and Lee, H. (2011). An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 215–223.

[25] Cortés, N. C. and Coello, C. A. C. (2003). Multiobjective optimization using ideas from the clonal selection principle. In *Genetic and Evolutionary Computation Conference*, pages 158–170. Springer.

[26] Cutello, V. and Nicosia, G. (2002). Multiple learning using immune algorithms. In *Proceedings of 4th International Conference on Recent Advances in Soft Computing, RASC*, pages 102–107.

[27] Cutello, V. and Nicosia, G. (2005). The clonal selection principle for in silico and in vitro computing. In *Recent developments in biologically inspired computing*, pages 140–147. Igi Global.

[28] Dase, R. and Pawar, D. (2010). Application of artificial neural network for stock market predictions: A review of literature. *International Journal of Machine Intelligence*, 2(2):14–17.

[29] David, G., Oosterlinck, K., and Szafarz, A. (2013). Art market inefficiency. *Economics Letters*, 121(1):23–25.

[30] de Brito, R. F. and Oliveira, A. L. (2012). A foreign exchange market trading system by combining ghsom and svr. In *The 2012 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7. IEEE.

[31] De Castro, L. N. and Timmis, J. (2002). *Artificial immune systems: a new computational intelligence approach*. Springer Science & Business Media.

[32] De Gooijer, J. G. and Hyndman, R. J. (2005). 25 years of iif time series forecasting: A selective review. *Tinbergen Institute Discussion Papers No. TI*, pages 05–068.

[33] D'haeseleer, P., Forrest, S., and Helman, P. (1996). An immunological approach to change detection: algorithms, analysis and implications. In *Proceedings 1996 IEEE Symposium on Security and Privacy*, pages 110–119. IEEE.

[34] Dorofki, M., Elshafie, A. H., Jaafar, O., Karim, O. A., and Mastura, S. (2012). Comparison of artificial neural network transfer functions abilities to simulate extreme runoff data. *International Proceedings of Chemical, Biological and Environmental Engineering*, 33:39–44.

[35] Dunis, C. and Williams, M. (2002). Modelling and trading the eur/usd exchange rate: Do neural network models perform better? *Derivatives use, trading and regulation*, 8(3):211–239.

[36] Dunis, C. L., Laws, J., and Sermpinis, G. (2009). The robustness of neural networks for modelling and trading the eur/usd exchange rate at the ecb fixing. *Journal of Derivatives & Hedge Funds*, 15(3):186–205.

[37] Dunis, C. L. and Williams, M. (2003). Applications of advanced regression analysis for trading and investment. *Applied quantitative methods for trading and investment*, pages 1–40.

[38] Dupernex, S. (2007). Why might share prices follow a random walk. *Student Economic Review*, 21(1):167–179.

[39] El-Jerjawi, N. S. (2018). Diabetes prediction using artificial neural network. *International Journal of Advanced Science and Technology*, 121(1):55–64.

[40] Engelbrecht, A. P. (2007). *Computational intelligence: an introduction*. John Wiley & Sons.

[41] Eswaran, C., Logeswaran, R., et al. (2010). A comparison of arima, neural network and linear regression models for the prediction of infant mortality rate. In *2010 Fourth Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation*, pages 34–39. IEEE.

[42] Fama, E. F. (1970). Efficient capital markets: A review of theory and empirical work. *The journal of Finance*, 25(2):383–417.

[43] Fama, E. F. (1991). Efficient capital markets: II. *The journal of finance*, 46(5):1575–1617.

[44] Fama, E. F. (1995). Random walks in stock market prices. *Financial analysts journal*, 51(1):75–80.

[45] Fathabadi, H. (2016). Power distribution network reconfiguration for power loss minimization using novel dynamic fuzzy c-means (dfcm) clustering based ann approach. *International Journal of Electrical Power & Energy Systems*, 78:96–107.

[46] Fausett, L. (1994). *Fundamentals of neural networks: architectures, algorithms, and applications*. Prentice-Hall, Inc.

[47] Finnerty, J. E. (1976). Insiders and market efficiency. *The Journal of Finance*, 31(4):1141–1148.

[48] Forbes, D. P. and Milliken, F. J. (1999). Cognition and corporate governance: Understanding boards of directors as strategic decision-making groups. *Academy of management review*, 24(3):489–505.

[49] Forrest, S., Perelson, A. S., Allen, L., and Cherukuri, R. (1994). Self-nonself discrimination in a computer. In *Proceedings of 1994 IEEE computer society symposium on research in security and privacy*, pages 202–212. Ieee.

[50] From, A. (2007). Iigss academic publisher are foreign exchange rates predictable? a survey from artificial neural networks perspective*. *Exchange Organizational Behavior Teaching Journal*, 8:207–227.

[51] Ghazali, R. (2007). *Higher order neural networks for financial time series prediction*. PhD thesis, Liverpool John Moores University.

[52] Ghazali, R., Hussain, A., and El-Deredy, W. (2006). Application of ridge polynomial neural networks to financial time series prediction. In *The 2006 IEEE International Joint Conference on Neural Network Proceedings*, pages 913–920. IEEE.

[53] Giles, C. L. and Maxwell, T. (1987). Learning, invariance, and generalization in high-order neural networks. *Applied optics*, 26(23):4972–4978.

[54] Göçken, M., Özçalıcı, M., Boru, A., and Dosdoğru, A. T. (2016). Integrating metaheuristics and artificial neural networks for improved stock price prediction. *Expert Systems with Applications*, 44:320–331.

[55] Golovko, V., Savitsky, Y., Laopoulos, T., Sachenko, A., and Grandinetti, L. (2000). Technique of learning rate estimation for efficient training of mlp. In *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium*, volume 1, pages 323–328. IEEE.

[56] Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. (2016). *Deep learning*, volume 1. MIT press Cambridge.

[57] Grossberg, S. (1987). Competitive learning: From interactive activation to adaptive resonance. *Cognitive science*, 11(1):23–63.

[58] Gupta, A., Chaudhary, D. K., and Choudhury, T. (2017). Stock prediction using functional link artificial neural network (flann). In *2017 3rd International Conference on Computational Intelligence and Networks (CINE)*, pages 10–16. IEEE.

[59] Guresen, E., Kayakutlu, G., and Daim, T. U. (2011). Using artificial neural network models in stock market index prediction. *Expert Systems with Applications*, 38(8):10389–10397.

[60] Hamid, K., Suleman, M. T., Ali Shah, S. Z., Akash, I., and Shahid, R. (2017). Testing the weak form of efficient market hypothesis: Empirical evidence from asia-pacific markets. *Available at SSRN 2912908*.

[61] Haykin, S. (1994). *Neural networks: a comprehensive foundation*. Prentice Hall PTR.

[62] Haykin, S. (1999). Multilayer perceptrons. *Neural networks: a comprehensive foundation*, 2:156–255.

[63] Haykin, S. (2010). Neural networks: a comprehensive foundation, 1999. *Mc Millan, New Jersey*.

[64] Hill, T., O'Connor, M., and Remus, W. (1996). Neural network models for time series forecasts. *Management science*, 42(7):1082–1092.

[65] Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800.

[66] Hinton, G. E. (2012). A practical guide to training restricted boltzmann machines. In *Neural networks: Tricks of the trade*, pages 599–619. Springer.

[67] Hinton, G. E., Osindero, S., and Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554.

[68] Hinton, G. E. and Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507.

[69] Hofmeyr, S. A. and Forrest, S. (1999). Architecture for an artificial immune system. *Evolutionary Computation*, 7(1):45–68.

[70] Hofmeyr, S. A. and Forrest, S. (2000). Architecture for an artificial immune system. *Evolutionary computation*, 8(4):443–473.

[71] Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558.

[72] Hornik, K., Stinchcombe, M., and White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366.

[73] Jerne, N. (1974). Clonal selection in a lymphocyte network. cellular selection and regulation in the immune response. *Journal of Society of General Physiologists*, pages 39–48.

[74] Kaita, T., Tomita, S., and Yamanaka, J. (2002). On a higher-order neural network for distortion invariant pattern recognition. *Pattern Recognition Letters*, 23(8):977–984.

[75] Kamruzzaman, J. and Sarker, R. (2003). Comparing ann based models with arima for prediction of forex rates. *Asor Bulletin*, 22(2):2–11.

[76] Karystinos, G. N. and Pados, D. A. (2000). On overfitting, generalization, and randomly expanded training sets. *IEEE Transactions on Neural Networks*, 11(5):1050–1057.

[77] Kayri, M. (2016). Predictive abilities of bayesian regularization and levenberg–marquardt algorithms in artificial neural networks: a comparative empirical study on social data. *Mathematical and Computational Applications*, 21(2):20.

[78] Kelsey, J. and Timmis, J. (2003). Immune inspired somatic contiguous hypermutation for function optimisation. In *Genetic and Evolutionary Computation Conference*, pages 207–218. Springer.

[79] Khan, A. and Ikram, S. (2010). Testing semi-strong form of efficient market hypothesis in relation to the impact of foreign institutional investors'(fii's) investments on indian capital market. *International Journal of Trade, Economics and Finance*, 1(4):373.

[80] Khoa, N. L. D., Sakakibara, K., and Nishikawa, I. (2006). Stock price forecasting using back propagation neural networks with time and profit based adjusted weight factors. In *2006 SICE-ICASE International Joint Conference*, pages 5484–5488. IEEE.

[81] Kia, A. N., Fathian, M., and Gholamian, M. (2012). Using mlp and rbf neural networks to improve theprediction of exchange rate time series with arima. *International Journal of Information and Electronics Engineering*, 2(4):543.

[82] Kiang, M. Y. (2001). Extending the kohonen self-organizing map networks for clustering analysis. *Computational Statistics & Data Analysis*, 38(2):161–180.

[83] Kohonen, T. (1982). Analysis of a simple self-organizing process. *Biological cybernetics*, 44(2):135–140.

[84] Kohonen, T. (1992). Learning vector quantisation and the self organising map. In *Theory and applications of neural networks*, pages 235–242. Springer.

[85] Koskivaara, E. (2000). Artificial neural network models for predicting patterns in auditing monthly balances. *Journal of the operational research society*, 51(9):1060–1069.

[86] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105.

[87] Kumar, K. and Haynes, J. D. (2003). Forecasting credit ratings using an ann and statistical techniques. *International journal of business studies*, 11(1).

[88] Kumar, L. and Rath, S. K. (2016). Hybrid functional link artificial neural network approach for predicting maintainability of object-oriented software. *Journal of Systems and Software*, 121:170–190.

[89] Larochelle, H. and Bengio, Y. (2008). Classification using discriminative restricted boltzmann machines. In *Proceedings of the 25th international conference on Machine learning*, pages 536–543. ACM.

[90] Lawrence, S. and Giles, C. L. (2000). Overfitting and neural networks: conjugate gradient and backpropagation. In *Neural Networks, 2000. IJCNN 2000, Proceedings of the IEEE-INNS-ENNS International Joint Conference on*, volume 1, pages 114–119. IEEE.

[91] Le Roux, N. and Bengio, Y. (2008). Representational power of restricted boltzmann machines and deep belief networks. *Neural computation*, 20(6):1631–1649.

[92] Leung, M. T., Quintana, R., and Chen, A.-S. (2008). A paradigm for group technology cellular layout planning in jit facility. In *2008 IEEE International Conference on Industrial Engineering and Engineering Management*, pages 1174–1178. IEEE.

[93] Li, H. and Sun, J. (2011). Principal component case-based reasoning ensemble for business failure prediction. *Information & Management*, 48(6):220–227.

[94] Li, K. (2008). Forecasting of government's financial educational fund by using neural networks model. In *Genetic and Evolutionary Computing, 2008. WGEC'08. Second International Conference on*, pages 120–123. IEEE.

[95] Li, Y. and Ying, F. (2011). Multivariate time series analysis in corporate decision-making application. In *Information Technology, Computer Engineering and Management Sciences (ICM), 2011 International Conference on*, volume 2, pages 374–376. IEEE.

[96] Liao, Y.-L., Hu, Y.-C., and Peng, Y.-F. (2016). Application of recurrent functional-link-based cmac networks for identification and prediction. In *2016 5th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI)*, pages 620–621. IEEE.

[97] Lo, A. W. and MacKinlay, A. C. (1988). Stock market prices do not follow random walks: Evidence from a simple specification test. *The review of financial studies*, 1(1):41–66.

[98] Lo, A. W. and MacKinlay, A. C. (2002). *A non-random walk down Wall Street*. Princeton University Press.

[99] Longo, G. A., Zilio, C., Ceseracciu, E., and Reggiani, M. (2012). Application of artificial neural network (ann) for the prediction of thermal conductivity of oxide–water nanofluids. *Nano Energy*, 1(2):290–296.

[100] Lu, T. and Li, Z. (2017). Forecasting csi 300 index using a hybrid functional link artificial neural network and particle swarm optimization with improved wavelet mutation. In *2017 International Conference on Computer Network, Electronic and Automation (ICCNEA)*, pages 241–246. IEEE.

[101] Mahdi, A., Hussain, A. J., Lisbo, P., and Al-Jumeily, D. (2009). The application of the neural network model inspired by the immune system in financial time series prediction. In *Developments in eSystems Engineering (DESE), 2009 Second International Conference on*, pages 370–376. IEEE.

[102] Mahdi, A., Weyde, T., and Al-Jumeily, D. (2017). The fl-smia network: A novel architecture for time series prediction. In *Developments in eSystems Engineering (DeSE), 2017 10th International Conference on*, pages 31–36. IEEE.

[103] Mahdi, A. A., Hussain, A. J., and Al-Jumeily, D. (2010). The prediction of non-stationary physical time series using the application of regularization technique in self-organised multilayer perceptrons inspired by the immune algorithm. In *Developments in E-systems Engineering (DESE), 2010*, pages 213–218. IEEE.

[104] Majhi, R., Panda, G., and Sahoo, G. (2009a). Development and performance evaluation of flann based model for forecasting of stock markets. *Expert systems with Applications*, 36(3):6800–6808.

[105] Majhi, R., Panda, G., and Sahoo, G. (2009b). Efficient prediction of exchange rates with low complexity artificial neural network models. *Expert systems with applications*, 36(1):181–189.

[106] McDonald, S., Coleman, S., McGinnity, T. M., Li, Y., and Belatreche, A. (2014). A comparison of forecasting approaches for capital markets. In *2014 IEEE Conference on Computational Intelligence for Financial Engineering & Economics (CIFEr)*, pages 32–39. IEEE.

[107] Mirea, L. and Marcu, T. (2002). System identification using functional-link neural networks with dynamic structure. *IFAC Proceedings Volumes*, 35(1):205–210.

[108] Mishra, N., Soni, H. K., Sharma, S., and Upadhyay, A. (2018). Development and analysis of artificial neural network models for rainfall prediction by using time-series data. *International Journal of Intelligent Systems and Applications*, 11(1):16.

[109] Moghaddam, A. H., Moghaddam, M. H., and Esfandyari, M. (2016). Stock market index prediction using artificial neural network. *Journal of Economics, Finance and Administrative Science*, 21(41):89–93.

[110] Morelli, M. J., Montagna, G., Nicrosini, O., Treccani, M., Farina, M., and Amato, P. (2004). Pricing financial derivatives with neural networks. *Physica A: Statistical Mechanics and its Applications*, 338(1-2):160–165.

[111] Moreno-Torres, J. G., Sáez, J. A., and Herrera, F. (2012). Study on the impact of partition-induced dataset shift on k-fold cross-validation. *IEEE Transactions on Neural Networks and Learning Systems*, 23(8):1304–1312.

[112] Mostafa, M. M. and El-Masry, A. A. (2016). Oil price forecasting using gene expression programming and artificial neural networks. *Economic Modelling*, 54:40–53.

[113] Nagesh, D. and Datta, G. (2010). Genetic algorithm for optimization of welding variables for height to width ratio and application of ann for prediction of bead geometry for tig welding process. *Applied soft computing*, 10(3):897–907.

[114] Nahil, A. and Lyhyaoui, A. (2018). Short-term stock price forecasting using kernel principal component analysis and support vector machines: the case of casablanca stock exchange. *Procedia Computer Science*, 127:161–169.

[115] Padhi, S., Behera, S., Sahu, B., and Mohanty, M. (2015). A trigonometric functional link single layer neural network approach for non linear dynamic plant identification using second order levenberg-marquardt algorithm. In *2015 International Conference on Condition Assessment Techniques in Electrical Systems (CATCON)*, pages 22–27. IEEE.

[116] Panda, C. and Narasimhan, V. (2007). Forecasting exchange rate better with artificial neural network. *Journal of Policy Modeling*, 29(2):227–236.

[117] Pao, Y. (1989). *Adaptive pattern recognition and neural networks*. Reading, MA (US); Addison-Wesley Publishing Co., Inc.

[118] Patel, J., Shah, S., Thakkar, P., and Kotecha, K. (2015). Predicting stock market index using fusion of machine learning techniques. *Expert Systems with Applications*, 42(4):2162–2172.

[119] Pourtaheri, Z. K. and Zahiri, S. H. (2016). Ensemble classifiers with improved overfitting. In *2016 1st Conference on Swarm Intelligence and Evolutionary Computation (CSIEC)*, pages 93–97. IEEE.

[120] Probst, P., Bischl, B., and Boulesteix, A.-L. (2018). Tunability: Importance of hyperparameters of machine learning algorithms. *arXiv preprint arXiv:1802.09596*.

[121] Qiu, M., Song, Y., and Akagi, F. (2016). Application of artificial neural network for the prediction of stock market returns: The case of the japanese stock market. *Chaos, Solitons & Fractals*, 85:1–7.

[122] Ren, C., An, N., Wang, J., Li, L., Hu, B., and Shang, D. (2014). Optimal parameters selection for bp neural network based on particle swarm optimization: A case study of wind speed forecasting. *Knowledge-based systems*, 56:226–239.

[123] Ricalde, L. J., Cruz, B., Catzin, G., Alanis, A. Y., and Sanchez, E. N. (2012). Forecasting for smart grid applications with higher order neural networks. In *World Automation Congress (WAC), 2012*, pages 1–6. IEEE.

[124] Ripley, B. D. and Hjort, N. (1996). *Pattern recognition and neural networks*. Cambridge university press.

[125] Rossi, M. (2015). The efficient market hypothesis and calendar anomalies: a literature review. *International Journal of Managerial and Financial Accounting*, 7(3-4):285–296.

[126] Ruder, S. (2016). An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*.

[127] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning internal representations by error propagation. In *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, volume 1.

[128] Sahoo, D. M. and Chakraverty, S. (2017). Functional link neural network learning for response prediction of tall shear buildings with respect to earthquake data. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 48(1):1–10.

[129] Salehinejad, H. and Valaee, S. (2019). Ising-dropout: A regularization method for training and compression of deep neural networks. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3602–3606. IEEE.

[130] Schaeffler, N. W. and Jenkins, L. N. (2006). Isolated synthetic jet in crossflow: experimental protocols for a validation dataset. *AIAA journal*, 44(12):2846–2856.

[131] Shen, X., Gao, X. Z., and Bie, R. (2008). Artificial immune networks: Models and applications. *International Journal of Computational Intelligence Systems*, 1(2):168–176.

[132] Singh, S., Bhambri, P., and Gill, J. (2011). Time series based temperature prediction using back propagation with genetic algorithm technique. *International Journal of Computer Science Issues (IJCSI)*, 8(5):28.

[133] Smith, G. and Ryoo, H.-J. (2003). Variance ratio tests of the random walk hypothesis for european emerging stock markets. *The European Journal of Finance*, 9(3):290–300.

[134] Smolensky, P. (1986). Information processing in dynamical systems: Foundations of harmony theory. Technical report, Colorado Univ at Boulder Dept of Computer Science.

[135] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

[136] Stogryn, A., Butler, C., and Bartolac, T. (1994). Ocean surface wind retrievals from special sensor microwave imager data with neural networks. *Journal of Geophysical Research: Oceans*, 99(C1):981–984.

[137] Tay, F. E. and Cao, L. (2001a). Application of support vector machines in financial time series forecasting. *Omega*, 29(4):309–317.

[138] Tay, F. E. and Cao, L. (2002). Modified support vector machines in financial time series forecasting. *Neurocomputing*, 48(1-4):847–861.

[139] Tay, F. E. H. and Cao, L. J. (2001b). Improved financial time series forecasting by combining support vector machines with self-organizing feature map. *Intelligent Data Analysis*, 5(4):339–354.

[140] Tenti, P. (1996). Forecasting foreign exchange rates using recurrent neural networks. *Applied Artificial Intelligence*, 10(6):567–582.

[141] Thawornwong, S. and Enke, D. (2004). The adaptive selection of financial and economic variables for use with artificial neural networks. *Neurocomputing*, 56:205–232.

[142] Thimm, G. (1997). *Optimization of high order perceptrons.* Technical Report. École Polytechnique Fédérale de Lausanne.

[143] Timmis, J. (2000). *Artificial immune systems: a novel data analysis technique inspired by the immune network theory.* PhD thesis, Department of Computer Science.

[144] Timmis, J. I. (2001). Artificial immune systems: A novel data analysis technique inspired by the immune network theory 2000. *Department of Computer Science, University of Wales.*

[145] Van Horne, J. C. and Parker, G. G. (1967). The random-walk theory: an empirical test. *Financial Analysts Journal*, 23(6):87–92.

[146] van Rijn, J. N. and Hutter, F. (2018). Hyperparameter importance across datasets. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2367–2376. ACM.

[147] Vogl, T. P., Mangis, J., Rigler, A., Zink, W., and Alkon, D. (1988). Accelerating the convergence of the back-propagation method. *Biological cybernetics*, 59(4-5):257–263.

[148] Wang, W. and Ding, J. (2003). Wavelet network model and its application to the prediction of hydrology. *Nature and Science*, 1(1):67–71.

[149] Widyanto, M. R., Nobuhara, H., Kawamoto, K., Hirota, K., and Kusumoputro, B. (2005). Improving recognition and generalization capability of back-propagation nn using a self-organized network inspired by immune algorithm (sonia). *Applied Soft Computing*, 6(1):72–84.

[150] Widyanto, R., Megawati, Y. T., and Hirota, K. (2002). Clustering analysis using a self-organized network inspired by immune algorithm. In *Proceeding of the IASTED International Conference on Artificial and Computational Intelligence, ACTA Press, Tokyo*, pages 197–202.

[151] Wright, J. H. (2008). Bayesian model averaging and exchange rate forecasts. *Journal of Econometrics*, 146(2):329–341.

[152] Yao, J., Poh, H.-L., and Jasic, T. (1996). Foreign exchange rates forecasting with neural networks. In *International Conference on Neural Information Processing (Hong Kong*. Citeseer.

[153] Zaknich, A. (2003). *Neural networks for intelligent signal processing*, volume 4. World Scientific.

[154] Zhang, G. and Hu, M. Y. (1998). Neural network forecasting of the british pound/us dollar exchange rate. *Omega*, 26(4):495–506.

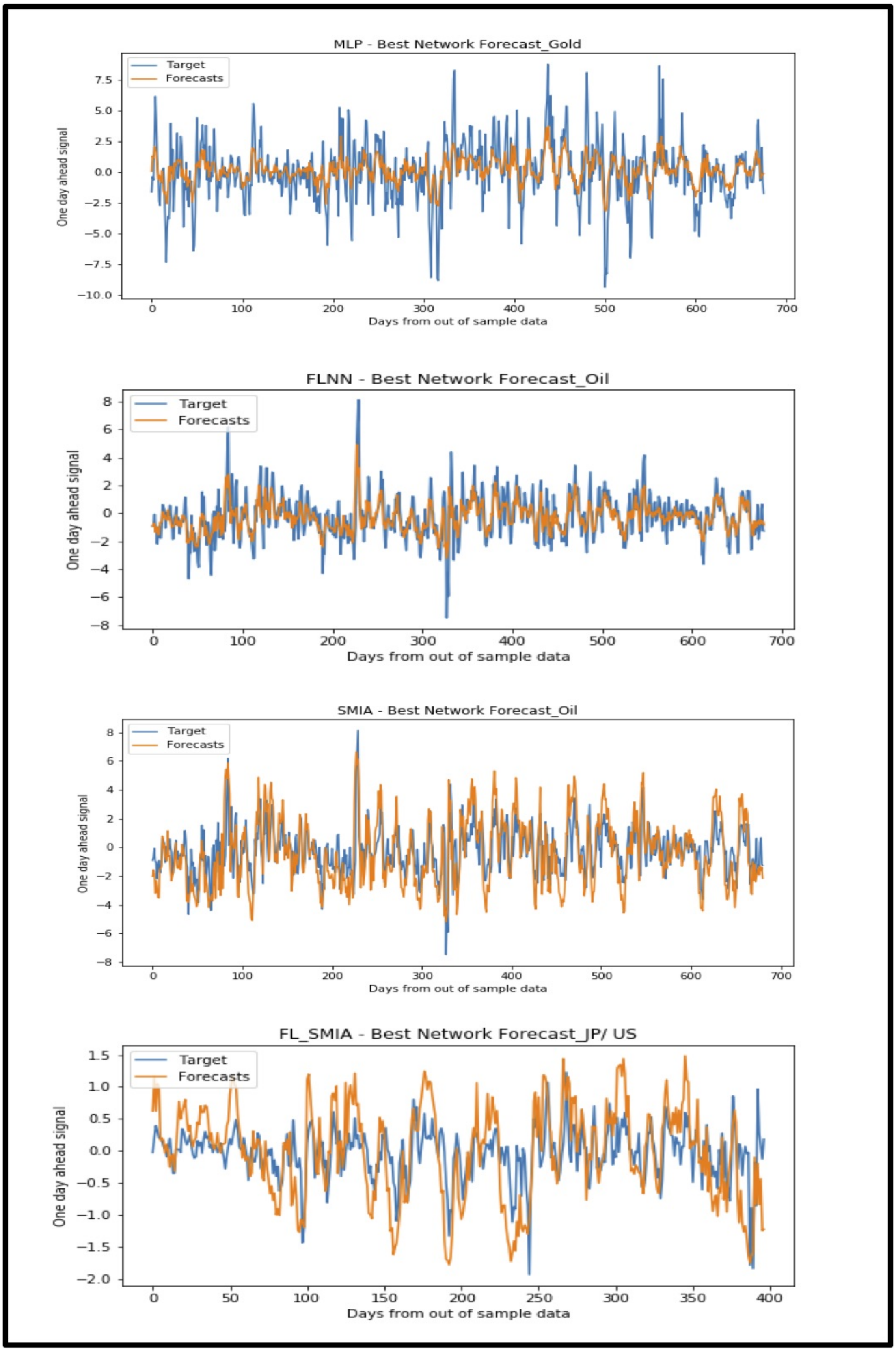
[155] Zhang, Y.-D., Pan, C., Sun, J., and Tang, C. (2018). Multiple sclerosis identification by convolutional neural network with dropout and parametric relu. *Journal of computational science*, 28:1–10.

[156] Zou, K. and Dong, R. (2010). A novel approach for time series analysis based rbf neural network. In *Information Technology and Applications (IFITA), 2010 International Forum on*, volume 3, pages 139–142. IEEE.

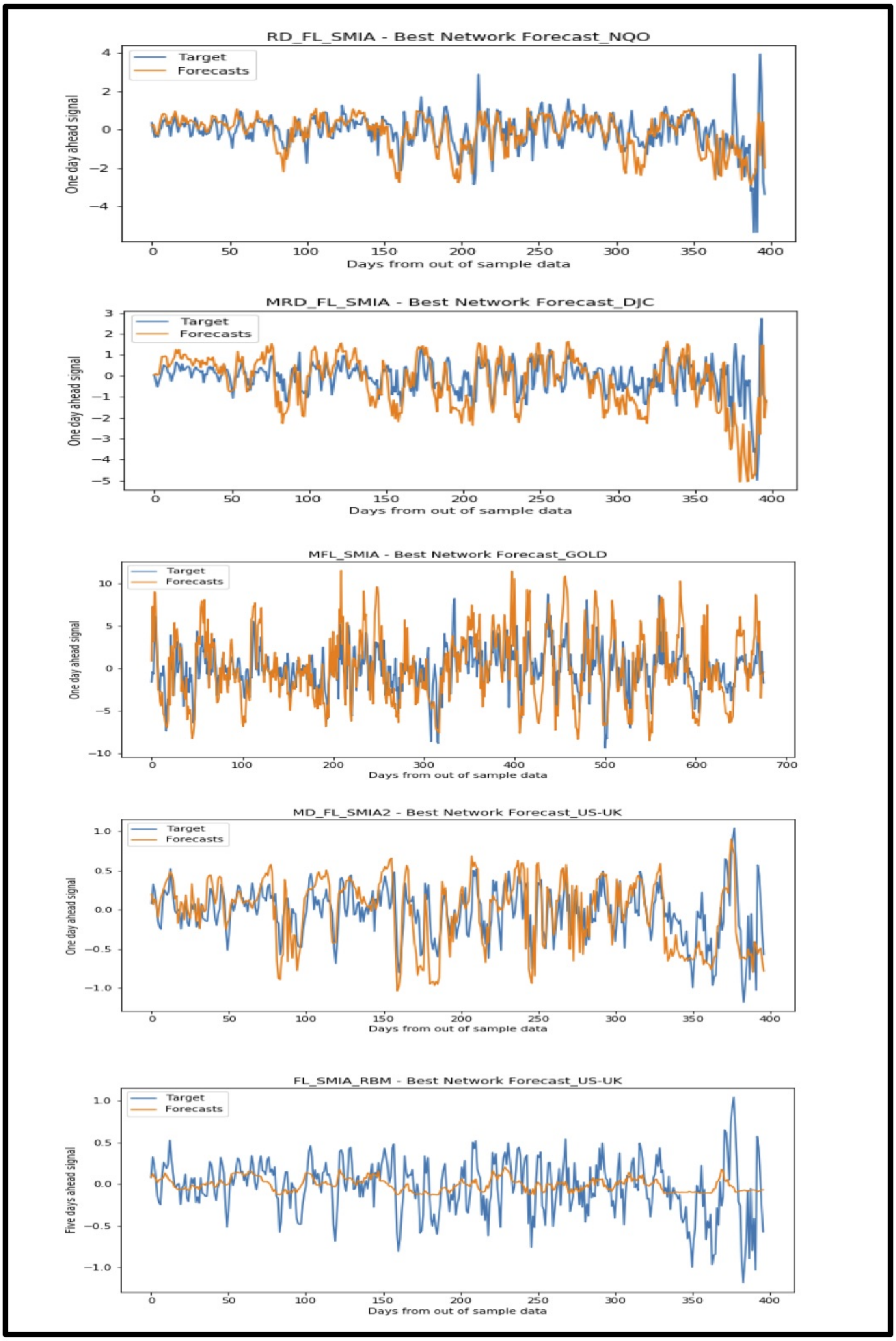
One Day Ahead Prediction Figures

This appendix includes plots illustrates as examples for the forecasting, as well for training and validation errors when using the data listed in table 3.3 (chapter 3) for one day ahead prediction.

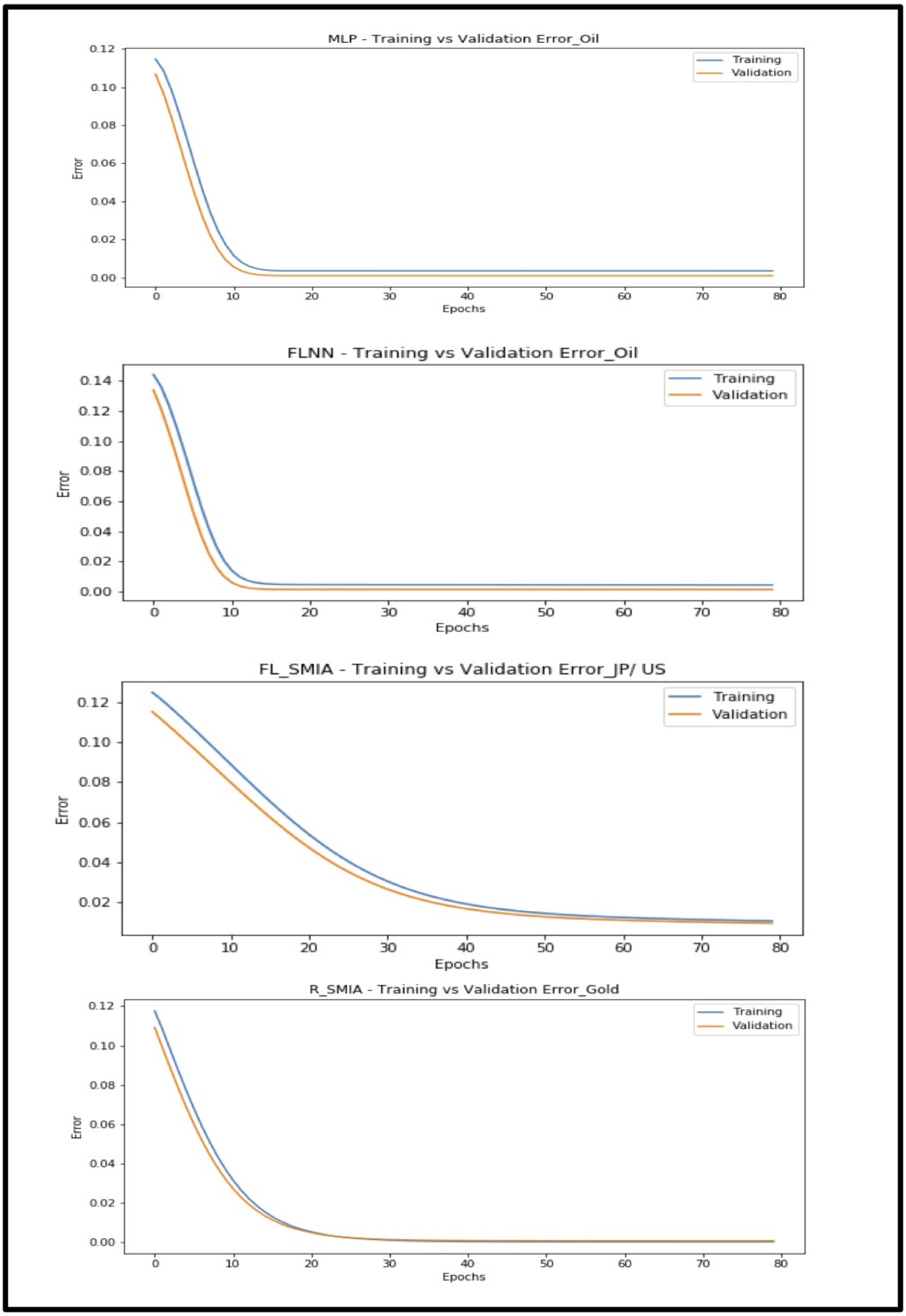
Appendix 1-1: The best forecasting on testing data for one day ahead prediction



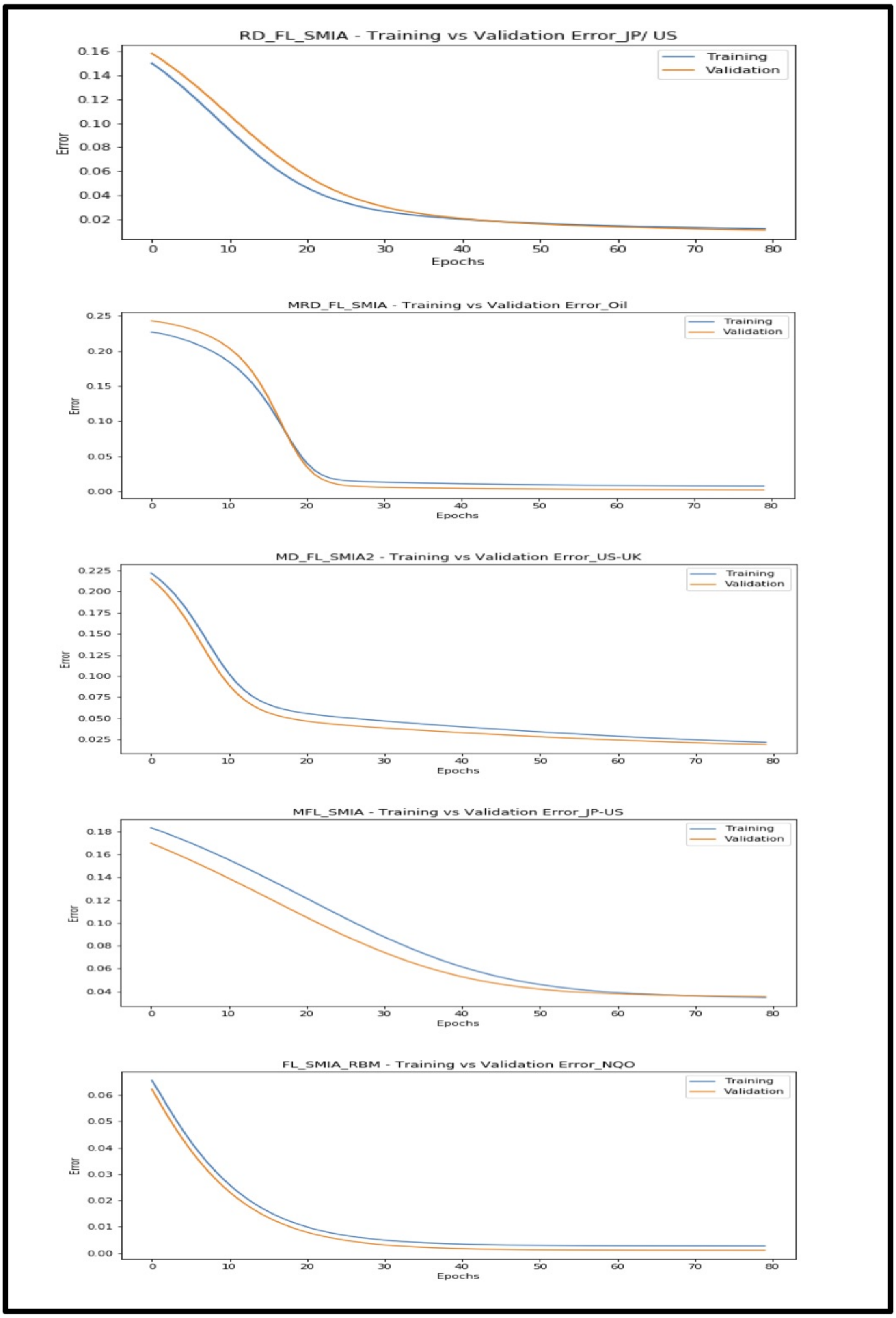
Appendix 1-2: The best forecasting on testing data for one day ahead prediction



Appendix 2-1: The best MSE results for training and validation errors for one day ahead prediction



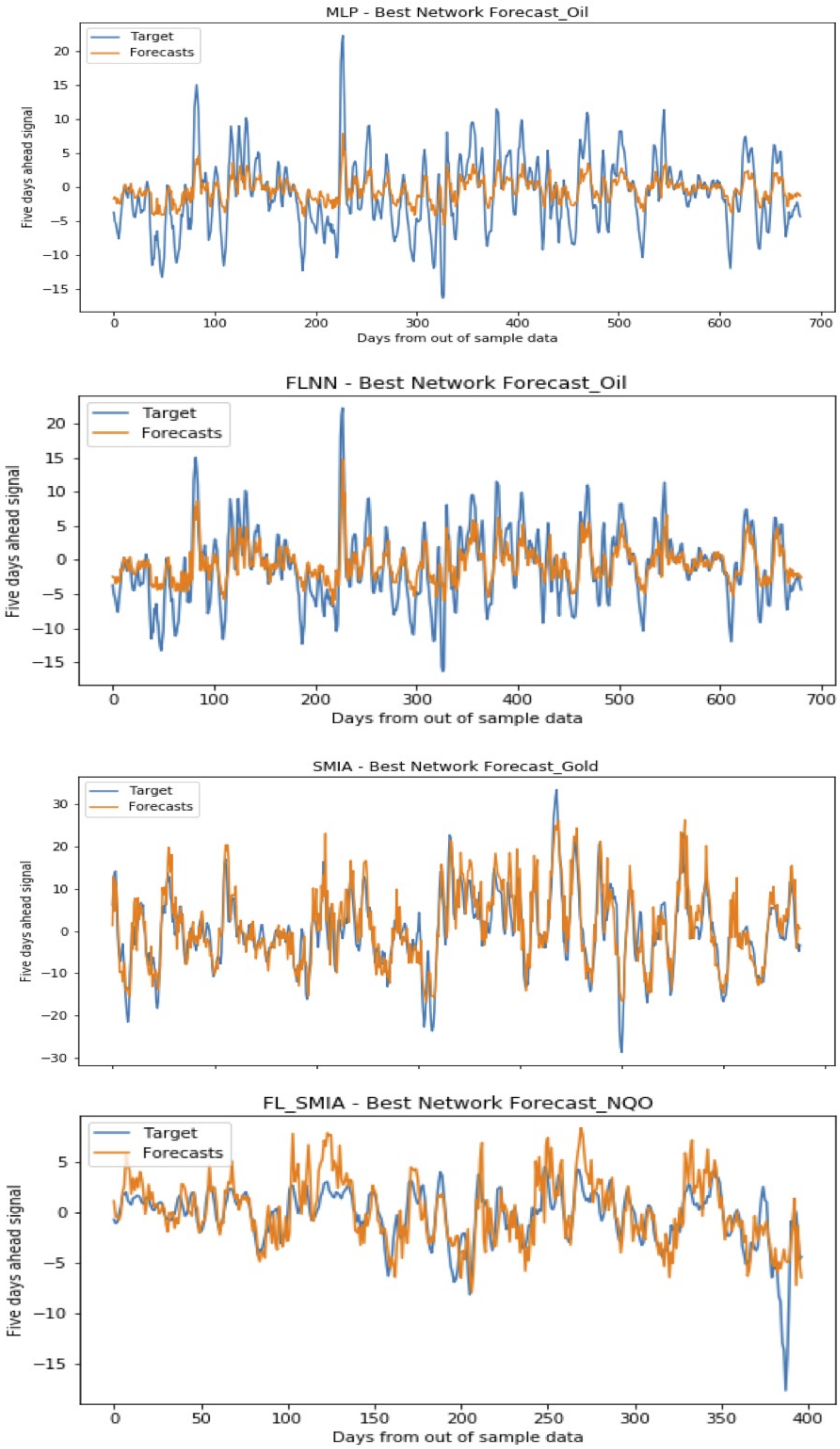
Appendix 2-2: The best MSE results for training and validation errors for one day ahead prediction



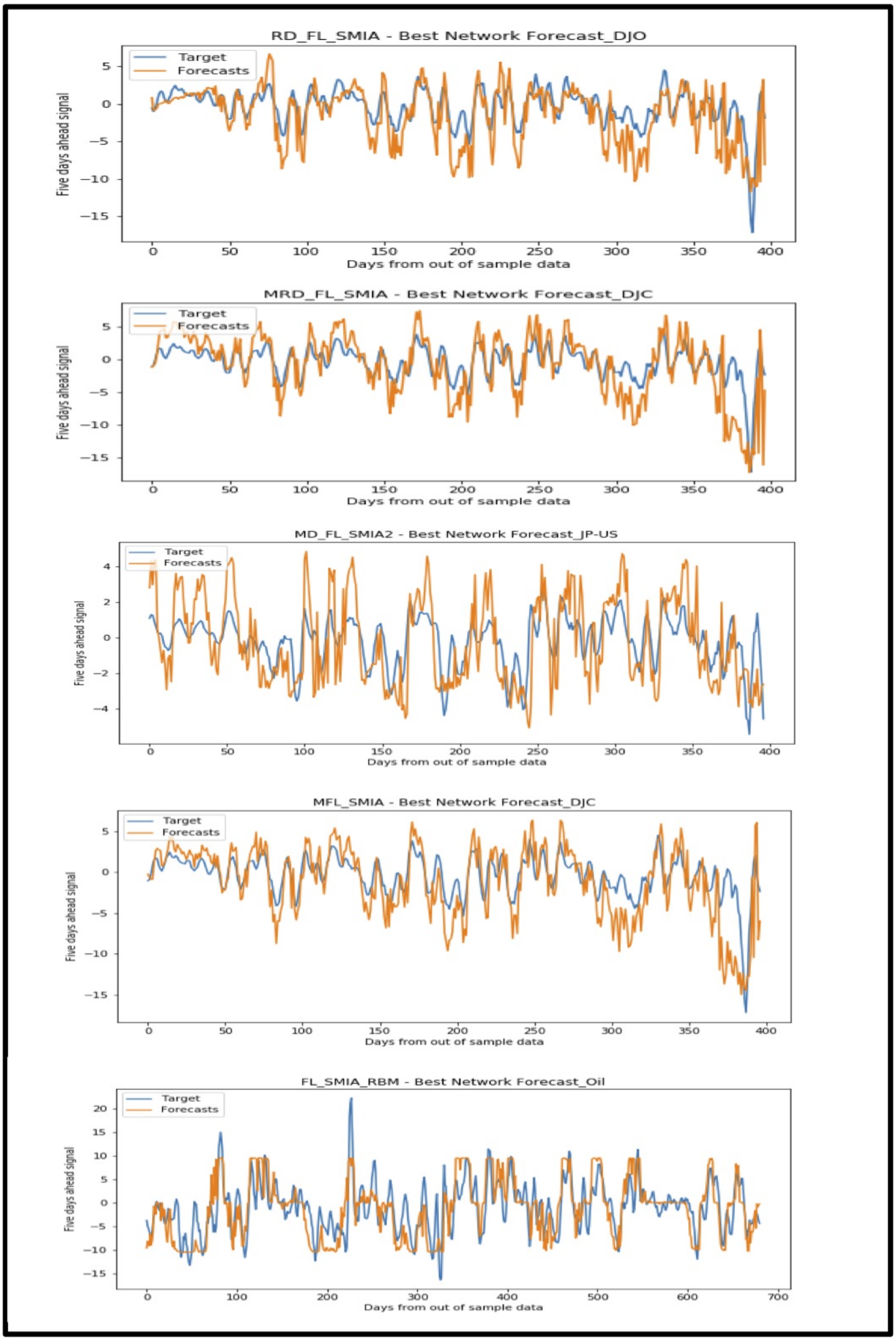
Five Days Ahead Prediction Figures

This appendix includes plots illustrates as examples for the forecasting, as well for training and validation errors when using the data listed in table 3.3 (chapter 3) for five days ahead prediction.

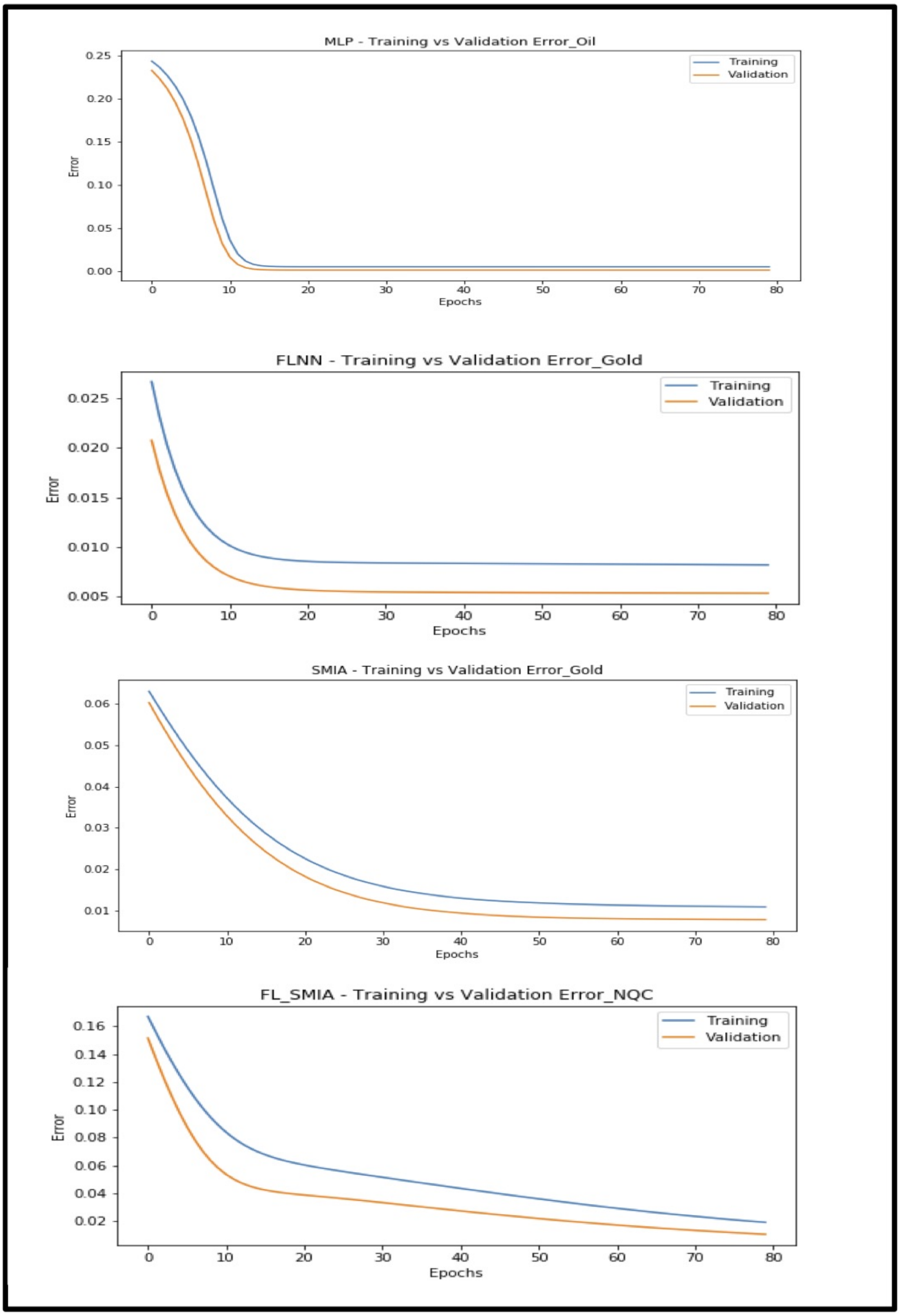
Appendix 1-1: The best forecasting on testing data for five days ahead prediction



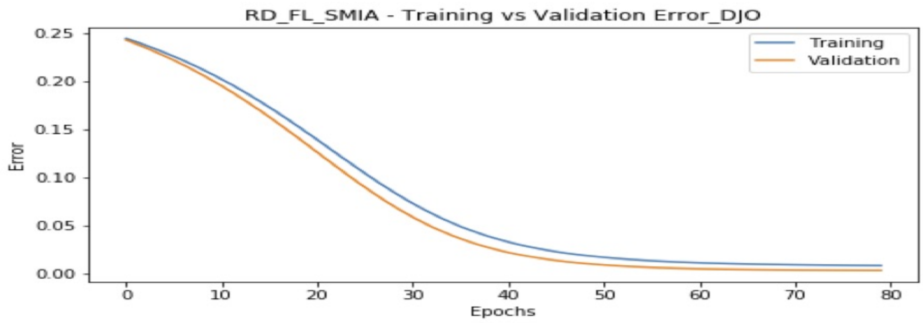
Appendix 1-2: The best forecasting on testing data for five days ahead prediction

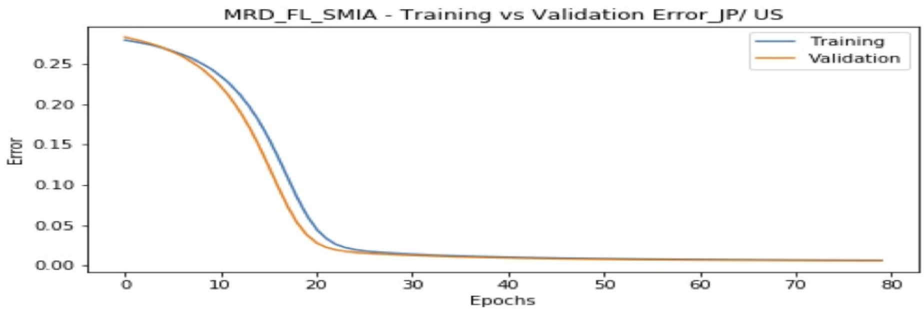


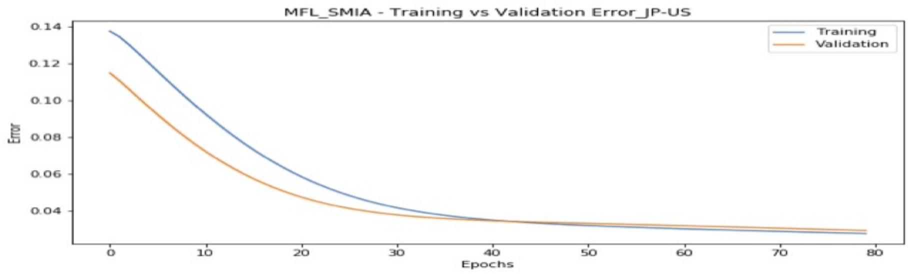
Appendix 2-1: The best MSE results for training and validation errors for five days ahead prediction

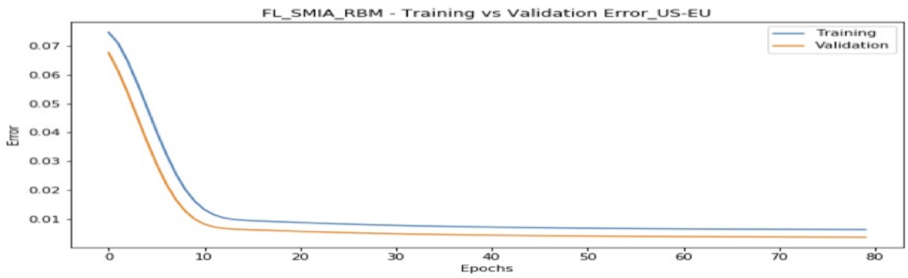


Appendix 2-2: The best MSE results for training and validation errors for five days ahead prediction









Publications

A) Publications before starting PhD study:

1) **A. Mahdi**, A. J. Hussain, and D. Al-Jumeily. “Adaptive Neural Network Model Using The Immune System for Financial Time Series Forecasting”, published at the CSSim workshop, International Conference on Computer Modelling and Simulation, Brno, Czech Republic, 2009, IEEE computer society, pp.104-109.

2.) **A. Mahdi**, A. Hussain, P. Lisboa, and D. Al-Jumeily. “The Application of the Neural Network Model Inspired by The Immune System In Financial Time Series Prediction”, Accepted for publication at the 2nd International Conference on Development in E-Systems Engineering (DESE09), Abu Dhabi, 2009 IEEE Computer Society, pp. 370-376.

3.) **A. Mahdi**, A. Jaafar Hussain, and D. Al-Jumeily. “The prediction of non-stationary physical time series using the application of regularization technique in self-organised multilayer perceptrons inspired by the immune algorithm.” In 2010 Developments in E-systems Engineering, pp. 213-218. IEEE, 2010.

4.) A. Al-Azzawi, M. Al-Saedi, **A. Mahdi**. “Speeding and Efficiency Increasing of Real Time Identification of Biometric Finger Print Using Particle Swarm Optimization Based On Selected Feature by Multiwavelet Transform” published at the International Conference on The 12th Annual PostGraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting PGNet (2011), Liverpool. UK, IEEE Computer Society, pp. 258-263.

B) Publications during the PhD study:

1.) **A. Mahdi**, T. Weyde and D. Al-Jumeily, 2017. The FL-SMIA Network: A Novel Architecture for Time Series Prediction. In: 10th International

Conference on Developments in eSystems Engineering (DeSE), IEEE, pp. 31-36.

2.) D. Al-Jumeily, A. Al-Azzawi, **A. Mahdi** and J. Hind, 2017. “A Robust Spatially Invariant Model for Latent Fingerprint Authentication Approach”. In 10th International Conference on Developments in eSystems Engineering (DeSE), IEEE, pp. 94-99.

3.) **A. Mahdi**, T. Weyde, D. Al-Jumeily. “Comparing Unsupervised Layers in Neural Networks for Financial Time Series Prediction” has been accepted for presentation and publication in the conference proceedings of the 2019 Developments in eSystems Engineering (DeSE).