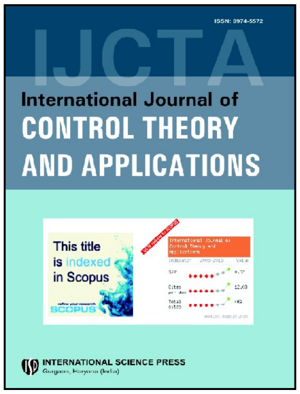


**International Journal of Control Theory and Applications**

ISSN : 0974-5572



**Volume 10 • Number 11 • 2017**

# Performance Analysis of Different Data Division Ratio for Time Series Data Using Multi Layer Feed Forward Neural Network with Tracking Signal Approach

**Ashok Kumar D.[a] and Murugan S[b]**

[a] *Department of Computer Science, Government Arts College, Trichirappalli - 620 022, Tamilnadu, India*
*E-mail : drashoktrichy@gmail.com*
[b] *Department of Computer Science, Alagappa Government Arts College,Karaikudi - 630 003, Tamilnadu, India*
*E-mail : muruganjit@rediffmail.com*

***Abstract:*** This study, proposes a novel neural network approach for predicting the closing index of the stock market. It strives to adapt the number of hidden neurons of a Multi Layer Feed Forward Neural Network (MLFFNN) model. It uses the Tracking Signal (TS) and rejects all models which results in values outside the interval of[-4, 4]. In addition, it determines the percentages of training, validation, and test set. The effectiveness of the proposed model is verified with one step ahead of Bombay Stock Exchange (BSE100) closing stock index of Indian stock market. This novel approach reduces the over-fitting problem, reduces the neural network structure and improves prediction accuracy. In addition, the proposed approach has been tested on standard NN3 forecasting competition time series dataset and this approach outperforms the statistical model and the computational intelligence model in NN3 competitor.

***Keywords:*** Neural Network, Time Series Data, Tracking Signal, Stock Index, Data Division Ratio, Prediction

## 1. INTRODUCTION

A time series is a collection of observations made chronologically. The characteristics of time series data is naturally of large size, high dimension and need to update frequently. Accuracy of time series forecasting is a difficult task for facing decision makers in many areas. Despite the numerous time series models are available, the researchers are still find the alternative model for effective forecasting.

In the past decade numerous researchers widely used MLFFNN models in financial time series forecasting. The reason is that the MLFFNN is a universal function approximation which is capable of mapping any linear or nonlinear functions, but there exists no general guideline to choose the appropriate network architecture for solving a given problem [1]. While designing a neural network model, the choice of number of hidden nodes in the hidden layer is data dependent and there is no systematic rule in deciding this parameter [2]. In [3] Min Qi and Guoqiang Peter Zhang investigated and reported the in-sample model selection criteria is not able to provide a reliable guide to out-of-sample performance and there is no apparent connection between in-sample model fit and out-of-sample forecasting performance. To solve the above mentioned problem, this paper introduces a novel MLFFNN with TS approach for forecasting the closing index of the stock market and also reduces the neural network structure and improves prediction accuracy.

In [4], Cecil Bozarth reported that, the TS is a statistical measure which is used to evaluate the presence of bias in the forecast model; and also it warns when there are unexpected outcomes from the forecast. Lean Yu *et al.,* [5] proposed that adaptive smoothing techniques are used to adjust the neural network learning parameters automatically by tracking signals under dynamic varying environments. In their study TS is used during the neural network training. In this study, the TS is used to analyze and select the best neural network model after the neural network training.

Traditionally, the neural network model selection is based on minimum forecasting error in validation set of some performance measure (SMAPE, NMSE, RMSE, etc) and reports its corresponding results in test set to avoid over-fit problem. After selecting the optimum model, still, there exists over-forecast or under-forecast in training set, validation set and test set. For example, the level of over-forecasting and under-forecasting is identified by the performance measure TS in test set of BSE100 stock market with different neural network model is represented in the Figure 1. The performance of neural network model degrades if over-forecast or under-forecast occurs. To solve this problem, the TS measure is used to rejects all neural network model which results in values outside the interval of [-4, 4]. This study suggests that the optimum neural network model selection is based on the interval value [-4, 4] in the training set and validation set which contains minimum forecasting performance error in SMAPE (instead of SMAPE, some other performance may used) of validation set. The reason for selecting the interval value is explained in section 2.3.

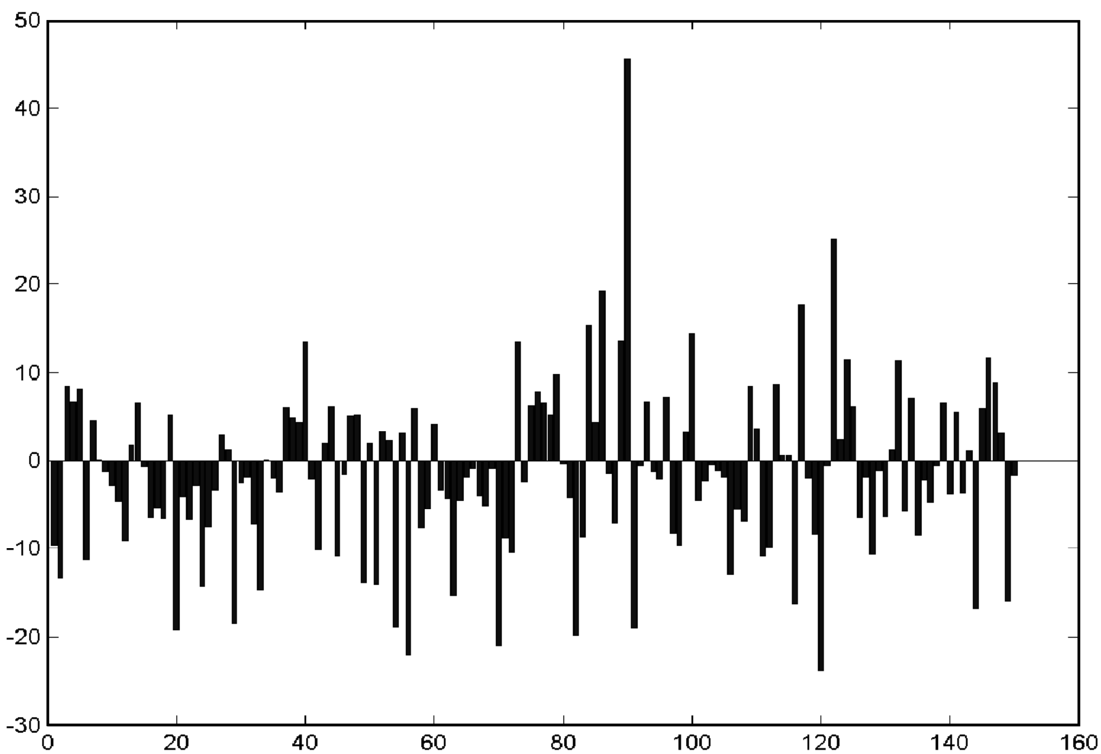


**Figure 1: Over-forecast and under-forecast in different model identified by TS in BSE100 index for the year 2001-2010**

This paper comprises (i) The systematic study to find all possible parameters for neural network model for time series forecasting problem. (ii) The performance measure tracking signal (TS) is introduced to select the optimum model. (iii) The in-sample and the out-of-sample forecasting performance analyzed using the different performance measure such as SMAPE, POCID and TS. (iv) The number of neurons in the hidden layer and the best data division ratio is identified for BSE100 stock market.

The remainder of this study is organized as follows: section 2 reviews the issues related to designing a neural network models for forecasting time series data, fundamentals of MLFFNN model, TS and performance metrics which are used to assess the performance of the proposed approach; section 3 describes the details of proposed MLFFNN with TS approach and traditional approach, i.e., MLFFNN without TS approach; section 4 reports the experimental results attained by the MLFFNN with TS approach and MLFFNN without TS approach using real world financial time series, as well as a comparison between the results achieved here and those given by standard computational intelligence method and statistical method in NN3 forecasting competition time series dataset. Finally this study is concluded in section 5.

## 2. LITERATURE REVIEW

### 2.1. Related Work

The author has extended the study based on the previous work [6], [7]. The author have studied and reported related to the existing statistical model, neural network model and hybrid model for time series forecasting; analyzed the performance of various types of training algorithms. The Levenberg-Marquardt training algorithm has better performance than all other training algorithms and also its error rate is very low when compared to all other training algorithms.

A detailed Artificial Neural Network (ANN) designing methodology and training process is described in the literature [6], [8], [9] and [10]. Greg Heath [11] suggests that design of ten neural networks with different types of random initial weights to mitigate the occasional bad random start. Jeff Heaton [12] reported that, a network with one hidden layer and 2N + 1 hidden neuron is sufficient for N inputs, and states that the optimum number of hidden neurons and hidden layers are highly problem dependent. Three rules of thumb methods for determining number of neurons in the hidden layer. AdebiyiAyodele [13] noted that training a great number of ANN with different configurations and selects the optimum model.

The choice of train/validation/test data can be partitioned into 50/25/25 [14] or 60/0/40 [15] or 70/0/30 [16] or 80/0/20 [17] in the literature. The published research articles reported that the optimum neural network model selection is based on the minimum error in the MAE, MSE, MAPE , RMSE, MPE, Theil'U, NMSE [18] or highest value in R [19] and POCID [14] in the validation set. This study analyses the performance of neural network model by using Tracking Signal measure.

### 2.2. Multi Layer Feed Forward Neural Network Model

MLFFNN consists of an input layer, one or more hidden layers and an output layer. The hidden layer receives weight from input layer. Each subsequent layer receives weight from the previous layer. The neurons present in the hidden and output layers have biases, which are the connection from the units and its activation is always one as shown in Figure 2. The bias term also acts as weights and it shows the architecture of Back Propagation Neural Network, depicting only the direction of information flow for the feed forward phase. During the back propagation phase of learning, signals are sent in reverse direction. The inputs are sent to the back propagation network and the output obtained from the net could be either binary 0, 1 or bipolar -1, +1 activation function. The error back propagation training algorithm is purely based on the gradient descent method [20].

## 2.3. Tracking Signal

In [4] Cecil reported that the Tracking Signal is calculated as the ratio of cumulative forecast error divided by the mean absolute deviation (MAD). It can be represented in the equation (3). If the forecast value is lower than the actual value then the model is in under forecasting and TS will be positive. If the forecast value is higher than the actual value then the model is in over forecasting and TS will be negative. If the TS limit is between the interval [-4, +4] then the forecast model is working correctly. The threshold of 4 is really a threshold of 3.75 (3SD). This 3.75 number comes from the statistical control limit theory which establishes the relationship between Mean Absolute Error or Deviation and Standard Deviation. The relationship between MAD and the Standard deviation in a normally distributed population is established as 1.25 MAD = 1 SD (standard deviation of the distribution).

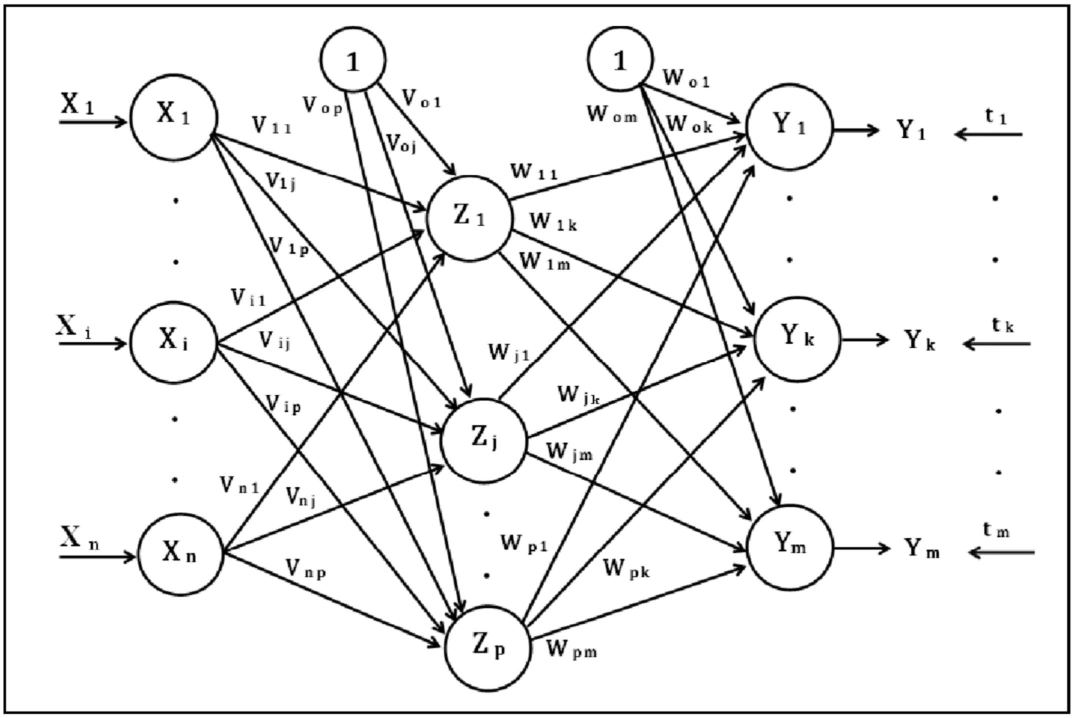


**Figure 2: Multi Layer Feed Forward Neural Network**

## 2.4. Forecasting performance measure

The forecasting performance is evaluated using the statistical measures, namely, symmetric mean absolute percentage error (SMAPE), percentage of change in direction (POCID) and Tracking Signal (TS).

In each of the following measure $y_t$ is the actual value, $f_t$ is the forecasted value. $e_t = y_t - f_t$ is the forecast error and n is the size of the test set.

The global performance of a forecasting model is evaluated by the SMAPE [1] which is used in NN3, NN5 and NNGC1 forecasting competition. A smaller SMAPE value suggests the better forecasting accuracy. It can be expressed as

$$SMAPE = \frac{1}{n}\sum_{t=1}^{n}\frac{|e_t|}{(y_t + f_t)/2} \times 100 \tag{1}$$

POCID (Percentage of Change in Direction) [14] maps the accuracy in the forecasting of the future direction of the time series. A larger POCID value suggests the better forecasting accuracy. It tends to 100% is a perfect modeling. It can be represented as

$$POCID = 100\frac{\Sigma_{t=1}^{n} D_t}{n} \tag{2}$$

$$where\ D_t = \begin{cases} 1\ if\ (y_t - y_{t-1})(f_t - f_{t-1}) \geq 0 \\ 0\ otherwise \end{cases}$$

In [4] Cecil reported that the Tracking Signal (TS) is used to pinpoint forecasting models that need adjustment. As long as the TS is between -4 and +4, assume the model is working correctly. The TS is a simple indicator that forecast bias is present in the forecast model. TS is used to verify the validity of the forecasting model. It can be represented as,

$$TS = \frac{\Sigma_{t=1}^{n} e_t}{MAD} \tag{3}$$

The Mean Absolute Deviation (MAD) measures the average absolute deviation of forecasted values from original ones.

$$MAD = \frac{\Sigma_{t=1}^{n} |e_t|}{n} \tag{4}$$

## 3. PROPOSED METHODOLOGY

Over fitting is one of the main issues in neural network modeling. In order to reduce over fitting problem, this study proposed a novel approach MLFFNN with TS is used to forecast the closing index of the stock market. MLFFNN trains different network by using different random initial weight with different neurons. TS measure is used to rejects all neural network model which results in values outside the interval of [-4, 4] in train set and validation set of different neural networks.

In neural network modeling, training parameter and the weight play an important role to increase the forecasting accuracy. The proposed MLFFNN with TS approach is tried to find optimal parameter, particularly, number of neurons in the hidden layer, optimum data division ratio and optimal weight for the time series forecasting problem.

In this study, forecasting strategies are taken a step ahead of prediction. Let $y_1$, $y_2$, $y_3$…..$y_t$ be a time series. As time t for t>=1, the next value $y_{t+1}$ is predicted based on the observed realizations of $y_t$, $y_{t-1}$, $y_{t-2}$….. $y_1$. The resulting network can be used for multi-step prediction by feeding the prediction back to the input of network recursively. The MLFFNN with TS approach is constructed from MLFFNN and the performance measure TS is represented in Figure 3.

In Figure 3, $X_i$ is the closing stock index vector, $Y_i$ is the predicted closing stock index from neural network model and $N_j$ is neurons size in hidden layer. For every neural network model, verify the presence of tracking signal interval[-4, +4] in training set and validation set. If it is present, the model is considered as feasible model otherwise the model is rejected. This process is repeated until the specified trial number (random initial weight) and maximum neuron size is reached.

The implementation procedure of MLFFNN with TS approach is represented in Algorithm 1, and explained further as follows. Neural network training process is an iterative process. Before training the neural network, the input and target data should be normalized. During this process the input data converted into -1 to +1.

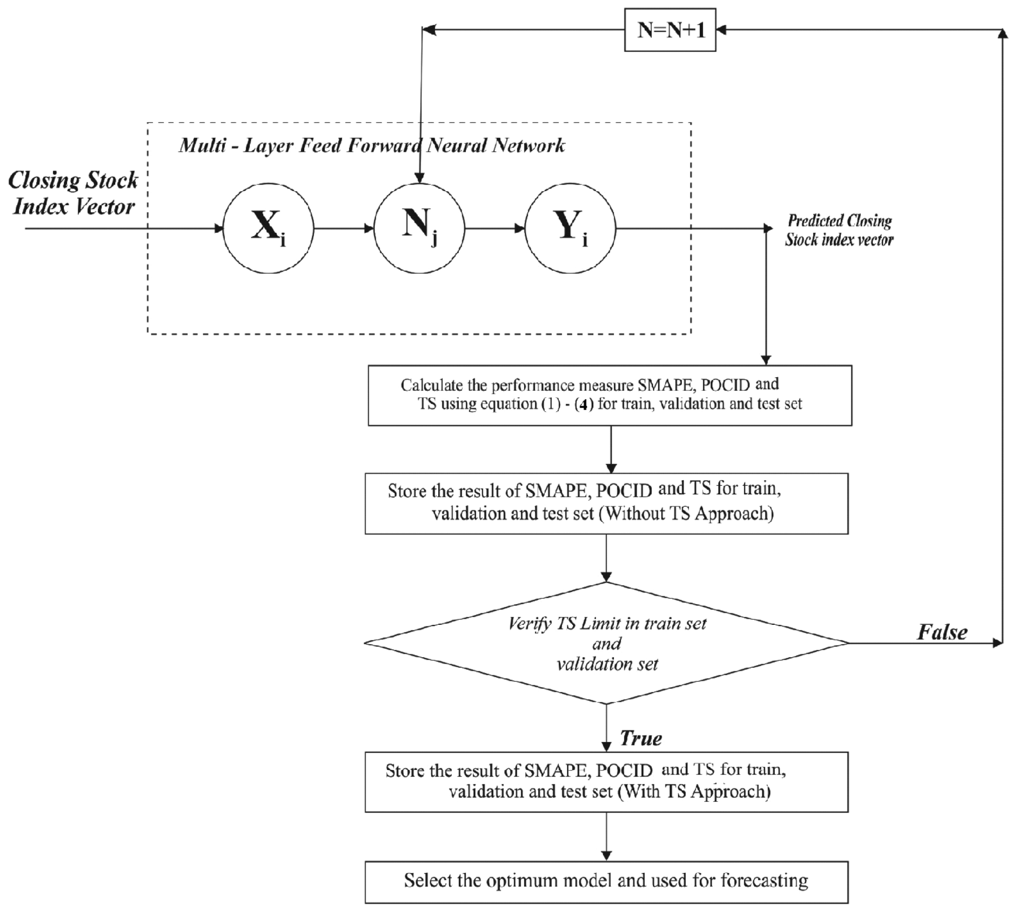


**Figure 3: Multi Layer Feed Forward Neural Network with Tracking Signal Approach**

The preprocessed data can be divided into three categories: a training set, a validation set and a test set. Training set is repeatedly used to fit the models, validation set is used to estimate the forecasting error for model selection; test set is used to assessment of the generalization error of the final chosen model. Divide block method is used to distribute the dataset into train, validation and test data set. The MLFFNN with TS approach is applied to different data division ratio such as 50/25/25, 60/20/20, 70/15/15 and 80/10/10 used as train, validation and test set.

After the division of data chosen, MLFFNN model with tansig neuron in the hidden layer and linear neuron in the output layer is used. The tan sigmoidal function and linear function is defined in equation (5) and (6).

$$\tan sig(x) = \frac{2}{1+e^{-2x}} - 1 \tag{5}$$

$$purelin(x) = x \tag{6}$$

Levenberg Marquardt is used as a training algorithm. After training the neural network, simulate the neural network and post process the simulated output. Finally analyze the performance of neural network using performance measure [18] equation (1) - (4).

The traditional approach of neural network training process is represented in step 1 to step 12 of Algorithm1 is known as MLFFNN without TS approach and the remaining steps are known as MLFFNN with TS approach proposed by the author of this study. For every neural network model, verify the presence of tracking signal interval [-4, +4] in training set and validation set.

In MLFFNN with TS approach, for every neural network model, check the presence of TS interval [–θ, + θ] in the training set and validation set, where θ = 4 and SD=3. It rejects all neural network model which results in values outside the interval of [-4, +4]; it accepts the neural network model which results in values inside the interval of [-4, +4]. Finally, the optimum neural network model selection is based on the interval value [-4, +4] in the training set and validation set which contains minimum forecasting performance error in SMAPE of validation set. This proposed approach, leads to better forecast; and there exist a connection between in-sample model fit and out-of-sample forecasting performance.

In MLFFNN without TS approach, after post processing the data, stores the results of performance measure SMAPE, POCID and TS of training set, validation set and test set for different neural network model. The optimum neural network model selection is based on minimum forecasting error in validation set of SMAPE.

In MLFFNN with TS approach, after post processing the data, stores the results of performance measure SMAPE, POCID and TS of training set, validation set and test set for different neural network model which contains the values inside the interval of [-4,+4] in the training and validation set. From the different neural network model, the optimum neural network model selection is based on minimum forecasting error in validation set of SMAPE.

---

Algorithm 1.Multi Layer Feed Forward Neural Network Model using Tracking Signal (MLFFNN with TS) Approach

---

1. Read the input and target pair from the data file and normalize or pre-process the data using mapminmaxfuction.
2. Initialize the neuron size in hidden layer, number of trial (random initial weight) for random weight generation and SD (Standard Deviation) value for assigning TS limit.
3. Repeat the step 4 to 13 until the number of neurons specified in the step 2.
4. Repeat the step 5 to 13 until the number of trial specified in the step 2.
5. Create neural network architecture here; specify the input and target vector, number of hidden layer, training function, transfer function used in the hidden and output layer.
6. Select the data division ratio using divide function and divide the dataset into train, validation and test set using divideparam function.
7. Train the neural network using train function.
8. Simulate the neural network using sim function.
9. Denormalize or post-process the simulated neural network output data
10. Calculate the performance measure SMAPE, POCID and TS for train, validation and test set using equation (1) - (4).
11. Record the result of neuron size, trial number, epoch, convergencetime and performance measure specified in step 10. It contains the performance of different neural network model using MLFFNN without TS approach.
12. Verify the interval [–θ, +θ] of Tracking Signal in training set (TStrain) and validation set (TSvalidation) from step 11, where θ = round (SD * 1.25).

    If (TStraine" –θ &&TStrain ≤ + θ) and (TS validation ≥ –θ && TSvalidation ≤ +θ) then goto step 13. Otherwise, goto step 4.
13. Record the result of neuron size, trial number, epoch, time and performance measure specified in step 10. It contains the performance of different neural network model using MLFFNN with TS approach.

14. From the step 11, select the optimum neural network model, which provides less error in SMAPE for MLFFNN without TS approach.
15. From the step 13, select the optimum neural network model, which provides less error in SMAPE for MLFFNN with TS approach.

## 4. EXPERIMENTAL RESULT

In this section, there are two main issues: first, to verify the effectiveness of the proposed MLFFNN with TS approach for closing stock index forecasting; second, to demonstrate the superiority of the proposed MLFFNN with TS approach by comparing it with existing time series forecasting methods. The results were carried out in MATLAB 7.10.0.499(r2010A) - 32 Bit with INTEL i3 processor @ 2.20 GHz and 4 GB RAM.

### 4.1. BSE100 Index

The effectiveness of the proposed MLFFNN with TS approach is tested on BSE100 index. The dataset consists of BSE100 closing stock index for the period from January 1, 2010 to December 31, 2012 from the BSE Website [21]. For each neural network created with different random initial weight for neuron 1 to neuron 10. The choice of random initial weight (trial) size and maximum neuron size is selected by user. In this study, random initial weight size is 15 and maximum neuron size is 10. This configuration is applied to different data division ratio such as 50/25/25, 60/20/20, 70/15/15 and 80/10/10.

The results of performance measure of 10 models from 1-1-1 to 1-10-1 were generated. Every neural network model contains fifteen different random initial weight generations. From the ten architectures of different trial, some models are extracted by the MLFFNN with TS approach which contains the interval [-4, +4] in the tracking signal of training set and validation set for different data division ratio. Rejection of model for every data division ratio is represented in Table 1 which does not contains the interval [-4, +4] in the training set and validation set of tracking signal.

**Table 1**
**Model rejection in different data division ratio**

| *Ratio* | *Model Rejection* |
|---|---|
| 50/25/25 | 1-2-1, 1-5-1, 1 -7-1, 1-10-1 |
| 60/20/20 | 1-4-1, 1-5-1, 1-6-, 1-8-1, 1-9-1 |
| 70/15/15 | No models are rejected |
| 80/10/10 | 1-1-1, 1-2-1, 1-3-1, 1-4-1,1-4-1, 1-6-1, 1-9-1, 1-10-1 |

The performance measure of SMAPE, POCID and TS of training set, validation set and test set using MLFFNN with TS approach and MLFFNN without TS approach for the BSE100 index in the year 2010 to 2012 with different training ratio are shown in the Table (2) to (5). In every data division ratio, only the results of optimum models are reported in the table.

**Table 2**
**Performance measure of train, validation and test set with data division ratio 50/25/25**

| *50/25/25 Ratio* | | *MLFFNN without TS* | *M* | | *LFFNN with TS* | |
|---|---|---|---|---|---|---|
| *Measure* | *Train* | *Val* | *Test* | *Train* | *Val* | *Test* |
| SMAPE | 0.85 | 0.73 | 0.92 | 0.87 | 0.76 | 0.84 |
| POCID | 77.10 | 75.90 | 74.90 | 77.60 | 73.30 | 76.50 |
| TS | 2.22 | 5.89 | 34.20 | 0.00 | 0.17 | 0.04 |

**Table 3**
**Performance measure of train, validation and test set with data division ratio 60/20/20**

| *60/20/20 Ratio* | *MLFFNN without TS* | | | *MLFFNN with TS* | | |
|---|---|---|---|---|---|---|
| *Measure* | *Train* | *Val* | *Test* | *Train* | *Val* | *Test* |
| SMAPE | 0.85 | 0.72 | 0.91 | 0.89 | 0.76 | 0.75 |
| POCID | 77.50 | 74.70 | 74.00 | 77.30 | 76.00 | 74.00 |
| TS | 0.80 | -18.10 | 22.60 | -0.08 | -1.41 | 18.30 |

**Table 4**
**Performance measure of train, validation and test set with data division ratio 70/15/15**

| *70/15/15 Ratio* | *MLFFNN without TS* | | | *MLFFNN with TS* | | |
|---|---|---|---|---|---|---|
| *Measure* | *Train* | *Val* | *Test* | *Train* | *Val* | *Test* |
| SMAPE | 0.86 | 0.70 | 0.84 | 0.85 | 0.72 | 0.83 |
| POCID | 75.60 | 75.00 | 80.40 | 75.60 | 74.10 | 81.30 |
| TS | -0.52 | 9.00 | 26.30 | 1.37 | -0.71 | 18.80 |

**Table 5**
**Performance measure of train, validation and test set with data division ratio 80/10/10**

| *80/10/10 Ratio* | *MLFFNN without TS* | | | *MLFFNN with TS* | | |
|---|---|---|---|---|---|---|
| *Measure* | *Train* | *Val* | *Test* | *Train* | *Val* | *Test* |
| SMAPE | 0.88 | 0.64 | 0.85 | 0.85 | 0.73 | 0.80 |
| POCID | 76.00 | 80.00 | 76.00 | 77.00 | 68.00 | 77.70 |
| TS | -70.20 | -4.16 | -13.60 | -0.20 | 1.22 | -3.60 |

The optimum model for every data division ratio using MLFFNN without TS approach and MLFFNN with TS approach is represented in the Table (2) to Table (5). The best forecasting model is identified by a smaller value in SMAPE and a larger value in POCID.

From the Table (2) to (5), the results of performance measure in train, validation and test set is reported in four aspects. (i), whether the forecasting error is high or low; (ii) over fitting problem, i.e., whether the in-sample model selection criteria is able to provide a reliable guide to out-of-sample performance or not?; (iii) correctness of the predicted direction in the test set; (iv) effectiveness of the tracking signal.

First, the performance measure SMAPE of test set in MLFFNN with TS approach is low when compared to MLFFNN without TS approach in all types of data division ratio. It indicates the forecasting error is minimum in the proposed approach. In addition, it is observed that the forecasting error in validation set is high in all data division ratio in MLFFNN with TS approach when compared to MLFFNN without TS approach in all data division ratio; the proposed approach produce lowest forecasting error in SMAPE of the test set. .

Second, the difference between training set and test set in MLFFNN with TS approach is very close to each other in all data division ratio when compared to MLFFNN without TS approach in all data division ratio. This is the main purpose of tracking signal used in this study. This closeness of training and testing performance measure of SMAPE indicates that the in-sample model selection are able to provide a reliable guide to out-of-sample performance and there is a connection between in-sample model fit and out-of-sample model forecasting performance. It happens due to the model selection based on tracking signal.

Third, the performance measure POCID of test set in MLFFNN with TS approach is high when compared to MLFFNN without TS approach. It indicates the correctness of the forecasting direction is high in the proposed approach. Higher in POCID value indicates better forecast.

Fourth, the tracking signal value is within the interval [-4, +4] in all data division ratio. It indicates the level of over-forecasting and the level of under-forecasting is controlled by tracking signal measure. The value of tracking signal in the test set is very low when compared to MLFFNN without TS approach in all types of data division ratio. Particularly, the data division ratio 50/25/25 and 80/10/10 contains the interval [-4, +4] in the test set of the performance measure TS. In addition, the data division ratio 50/25/25 contains the value closer to zero in the training and test set of the performance measure of tracking signal.

From the table (2) to (5), the best data division ratio is 80/10/10 for MLFFNN without TS approachand the best data division ratio is 60/20/20 for MLFFNN with TS approachwith respect to minimum error in SMAPE of validation data set.

After the analysis of train, validation and test set of various data division ratio, the performance measure of optimum model reported in table (2) to (5) and their corresponding number of neurons in the hidden layer, training time and convergence speed using MLFFNN without TS approach and MLFFNN with TS approach is reported in the Table 6.

**Table 6**
**Optimum model selection using MLFFNN without TS approach and MLFFNN with TS approach for different data division ratio**

| | *MLFFNN without TS* | | | *MLFFNN with TS* | | |
|---|---|---|---|---|---|---|
| *Ratio* | *Neuron* | *Time (Sec)* | *epoch* | *Neuron* | *Time (Sec)* | *epoch* |
| 50/25/25 | 4 | 0.69 | 10 | 1 | 0.46 | 7 |
| 60/20/20 | 2 | 1.48 | 47 | 1 | 0.77 | 16 |
| 70/15/15 | 3 | 0.84 | 15 | 4 | 0.70 | 9 |
| 80/10/10 | 3 | 0.62 | 3 | 9 | 0.82 | 8 |

From the Table 6, it is observed that the neural network complexity (number of neurons in the hidden layer) is reduced; training time is reduced and fast convergence in MLFFNN with TS approach when compared to MLFFNN without TS approach except the data division ratio 80/10/10.

## 4.2. Prediction using large dataset BSE100 (2001 to 2012)

This section deals with the large amount of data, 12 years dataset of BSE100 closing stock index. The dataset consists of BSE100 closing stock index for the period from January 1, 2001 to December 31, 2012 from the BSE Website [21]. The experimental parameter is same as mentioned in BSE100 stock market data with the data division ratio 50/25/25. The results of SMAPE, POCID and TS in train set, validation set and test set using the MLFFNN with TS approach and MLFFNN without TS approach for the BSE100 of year 2001 to 2012 is shown in Table 7.

**Table 7**
**Performance measure of train, validation and test set using MLFFNN without TS and MLFFNN with TS approach for BSE100 of year 2001 to 2012**

| *50/25/25 Ratio* | *MLFFNN without TS* | | | *MLFFNN with TS* | | |
|---|---|---|---|---|---|---|
| *Measure* | *Train* | *Val* | *Test* | *Train* | *Val* | *Test* |
| SMAPE | 1.28 | 1.11 | 0.83 | 1.25 | 1.17 | 0.82 |
| POCID | 76.20 | 75.70 | 78.40 | 76.00 | 76.20 | 79.60 |
| TS | 7.44 | -75.40 | -6.67 | -0.07 | 2.87 | -1.39 |

It is noted that, the value of test set of all performance measure of Table 7 is relatively closer to the previous experiment in Table (2) to Table (5). It indicates the MLFFNN without TS and MLFFNN with TS approach perfectly fit on small size dataset to large size dataset.

Figure 4. shows the prediction graph of test dataset for 12 years data set of BSE 2001 to 2012, solid line represents forecasted data, dotted line represents actual data, X –axis represents time period t and Y –axis represents closing stock index. It shows the performance accuracy of the actual data versus forecasted data. From the figure, this MLFFNN with TS approach perfectly forecast the future value.

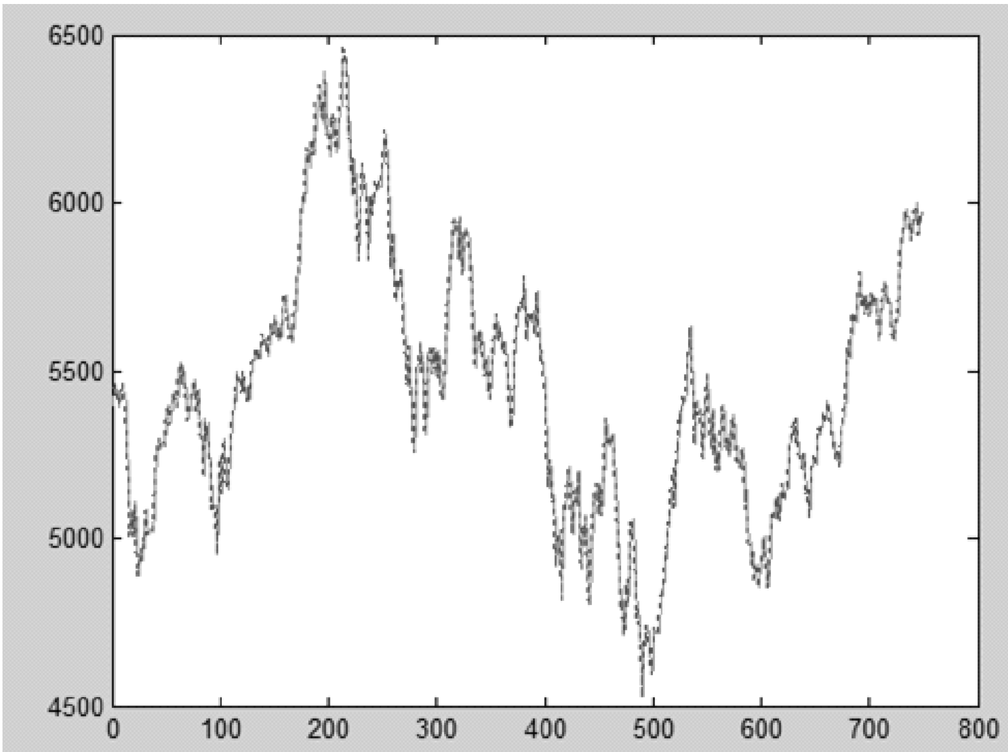


**Figure 4: Actual versus forecasted for BSE 2001 to 2012 testdata set**

### 4.3. NN3 Forecasting competition time series data

The superiority of the proposed MLFFNN with TS approach is compared with NN3 forecasting competition time series dataset. The dataset A contains 111 monthly time series drawn from a homogeneous population of empirical business time series. In every time series of NN3 dataset, last 18 points are reserved for test dataset. Remaining data points are divided into two parts, 50% of the total data points used for training and 24% of the total data points are used for validation. The experimental parameter is same as mentioned in BSE100 stock market data.

The performance measure SMAPE of this study is compared with the NN3 forecasting result [22]. From the NN3 forecasting result, this study extracts the best five benchmark statistical methods and the best five CI-based methods [23] for the purpose of comparison. The average SMAPE of this study and different method in forecasting competition using 111 time series (NN3 dataset A) as shown in Table 8is arranged by least error in SMAPE. The best results are highlighted in boldface. This study observed that this MLFFNN with tracking signal approach beats the benchmark statistical methods and CI based methods using 111 time series. MLFFNN without TS approach is secured in third place as shown in the Table 8.

**Table 8**
**Rank on SMAPE of MLFFNN with TS and without TS approach compared with NN3 111 time series data**

| *Participant* | *SMAPE* |
|---|---|
| MLFFNN with TS approach | 14.70 |
| Stat. Contender – Wildi | 14.84 |
| Stat. Benchmark – Theta Method (Nikolopoulos) | 14.89 |
| MLFFNN without TS | 15.00 |
| Illies, Jager, Kosuchinas, Rincon, sakenas, Vaskevcius | 15.18 |
| Stat. Benchmark – ForecastPro (Stellwagen) | 15.44 |
| CI Benchmark – Theta AI (Nikolopoulos) | 15.66 |
| Stat. Benchmark – Autobox (Reilly) | 15.95 |
| Adeodato, Vasconcelos, Aranaud, Chunha, Monteiro | 16.17 |
| Flores, Anaya, Ramirez, Morales | 16.31 |
| Chen, Yao | 16.55 |
| D'yakonov | 16.57 |

## 5. CONCLUSION

This study proposed a Multi Layer Feed Forward Neural Network with Tracking Signal (MLFFNN with TS) approach. It is proposed to forecast one-step-ahead closing index of stock market. It has analyzed the performance measure of SMAPE, POCID and TS in the training set, validation set and test set. After the analysis of various neural network models, finally MLFFNN without TS approach and MLFFNN with TS approach identified the number of neurons in the hidden layer and best data division ratio for improving prediction accuracy and reduce over fitting problem. This study recommends to increase the prediction accuracy, the best forecasting model is selected by the presence of tracking signal interval [-4, +4] in training set and validation set; and minimum error value in SMAPE of validation set;

The in-sample and the out-of-sample forecasting performance analyzed; and the results indicate that the in-sample model selection is able to provide a reliable guide to out-of-sample performance and there is a connection between in-sample model and out-of-sample model forecasting performance by using MLFFNN with TS approach. The experimental result with BSE market real datasets indicate that the proposed MLFFNN with TS approach can be an effective way in-order-to yield accurate prediction result. MLFFNN with TS approach is perfectly fitted on stock market data range from small dataset to large dataset. In addition, the proposed approach has been tested on standard NN3 forecasting competition time series dataset and this model outperforms the NN3 competitor. This study is also found that the tracking signal is the best performance measure for time series data and it controls the level of over forecasting and under forecasting.

The proposed MLFFNN with TS approach can be used as an alternative forecasting tool for time series forecasting. In this study, only single variable is taken for prediction; In future, multi variables will be taken for prediction to improve the accuracy of stock market; It will be applied to identify hidden neurons in the multiple hidden layer; and also it will be applied to different types of neural network model for predicting closing stock index/price of stock market data.

## REFERENCES

[1] Md. MustafizurRahman, Md. Monirul Islam and Xin Yao, "Layered Ensemble Architecture for Time Series Forecasting", *IEEE Transaction cybernetics,* 2015.

[2] Mehdi Khashei and Mehdi Bijari, "A novel hybridization of artificial neural networks and ARIMA models for time series forecasting", *Applied Soft Computing,* Vol. 11, pp. 2264-2275, 2011.

[3] Min Qi and Guoqiang Peter Zhang, "An investigation of model selection criteria for neural network time series forecasting", *European Journal of Operational Research,* Vol. 3, pp. 666-680, 2001.

[4] CecilBozarth, "Measuring Forecast Accuracy Approaches to Forecasting A Tutorial", http://scm.ncsu.edu/scm-articles/article/measuring-forecast-accuracy approaches-to-forecasting-a-tutorial.

[5] Lean Yu, Shouyang Wang and Kin Keung Lai, "Adaptive Smoothing Neural Networks in Foreign Exchange Rate Forecasting", Computational Science ICCS 2005 5th International, Vol. 3516, pp. 523-530, 2005.

[6] D. Ashok Kumar, and S. Murugan, "Performance Analysis of Indian Stock Market Index using Neural Network Time series Models", *IEEE Explore Digital Library*, pp. 72-78, 2013.

[7] D. Ashok Kumar, and S. Murugan, "'Performance Analysis of MLPFF Neural Network Back propagation Training Algorithms for Time Series Data", *IEEE Explore Digital Library*, pp. 114-119, 2014.

[8] P.P. Balestrassi,E. Popova, A. P. Paiva, and J. W. MarangonLima, "'Design of experiments on neural network training for nonlinear time series forecasting", *Neurocomputing,* Vol. 72, pp. 1160-1178, 2009.

[9] Abbas Vahedi, "The Predicting Stock Price using Artificial Neural Network", *Journal of Basic and Applied Scientific Research,* Vol. 2, pp. 2325-2328, 2012.

[10] A. IebelingKaastra, and B. Milton Boyd, "Designing a neural network for forecasting financial and economic time series", *Neurocomputing*, pp. 215-236, 1996.

[11] Greg Heath, "Problem about getting optimum output in Neural Network MATLAB 2012a", *http://in.mathworks.com/matlabcentral/newsreader/view thread/331714.*

[12] Jeff Heaton , "Introduction to Neural Networks for Java", Heaton Research, Inc, Second Edition, pp. 159-160, 2008.

[13] A. AdebiyiAyodele, K. Ayo Charles, O. Adebiyi Marion and O. Otokiti Sunday, "Stock Price Prediction using Neural Network with Hybridized Market Indicators", *Journal of Emerging Trends in Computing and Information Sciences,* Vol. 3, pp. 1-9, 2012.

[14] Ricardo de A.Araujo, and Tiago A.E. Ferreira, "An intelligent hybrid morphological-rank-linear method for financial time series prediction", *Neurocomputing*, Vol. 72, pp. 2507-2524, 2009.

[15] T. Victor Devadoss, and T. Antony AlphonnseLigori, "Stock Prediction Using Artificial Neural Networks", *International Journal of Data Mining Techniques and Applications*, Vol. 2, pp. 283-291, 2013.

[16] T. Victor Devadoss, and T. Antony AlphonnseLigori, "Forecasting of Stock Prices Using Multi Layer Perceptron", *International Journal of Computing Algorithm*, Vol. 2, pp. 440-449, 2013.

[17] Zhang Yudong, and Wu Lenan, "'Stock market prediction of S&P 500 via combination of improved BCO approach and BP neural network", *Expert Systems with Applications*, Vol. 36, pp. 8849-8854, 2009.

[18] R. Adhikari, and R. K. Agrawal, "An introductory study ontime series modeling and forecasting", LAPLambert Academic Publishing, Germany, pp. 1-67, 2007.

[19] K. K. Suresh Kumar,"Performance analysis of stock priceprediction using artificial neural network", *Global Journal of Computer Science and Technology,* pp. 154-170, 2012.

[20] S. N. Sivanandam, andS. N. Deepa,"Principles of softcomputing", Wiley India, First Edition, pp. 79-80, 2008.

[21] Bombay Stock Exchange India, *http://www.bseindia.com/indices/ IndexArchiveData.aspx? expandable=3*

[22] Artificial Neural Network and computational intelligence forecasting competition NN3, *http://www.nueral-forecasting-competition.com/NN3/results.htm*

[23] Sven F. Cronel., KonstantinosNikolopoulos and Michele Hibon, "Automatic Modelling and Forecasting with Artificial Neural Networks a forecasting competition evaluation", IIF/SAS Grant 2005/6 Research Report,Vol. 81, number 3, pp.1-52, April 2008.