MANAGEMENT INFORMATION SYSTEM MASTER OF SCIENCE: PADERBORN UNIVERSITY

Individual Study Research

Work : Applying RF and ANN to financial data and analyze which model performs best for your data

---

Lecturer: Prof. Yuanhua Feng

Start of Project: Tuesday, 9. November 2020

End of Project: Tuesday, 2. February 2021

Submitted by :

Name: Fotso Tenku Joel Cedric

Number: 6810782

E-Mails: joelf@campus.uni-paderborn.de

Paderborn, 02.02.2021

1. Introduction

To predict the stock market of a financial time series, several statistical learning methods have been developed. Either Among those methods, we can mention: Box and Jenkins' Autoregressive Integrated Moving Averages (ARIMA), the Random Forest (RF), Artificial Neural Networks (ANN), Support Vector Machine (SVM), Backpropagation, Radial Basis Function (RBF) networks (Mahmoud K. Okasha, 2014.in its study on Using Support Vector Machines in Financial Time Series) In this project, we will analyse only two techniques for financial forecasting namely, the Random Forest (RF) and the Artificial Neural Network (ANN).

According to L. Breiman (2001), the Random Forest consists of many deep but uncorrelated decision trees built up on different samples of the data. It is an ensemble technique able to perform both regression and classification tasks with the use of multiple decision trees to classify a new sample (Wenji Mao, Fei, 2012).

Unlike the Random Forest (RF), the ANN is a method that can be used for forecasting in non linear time series and can overcomes the problems of nonlinearity and nonstationary (Chang Sim Vui et al, 2013).

The purpose of this work is to compare the RF and ANN and to determine which of the two techniques performs best the financial data. For this, we will rely on a financial time series data, namely a S&P 500 index of the New York Stock Exchange (NASDAQ), starting from June 2015 up to June 2020. Then, we will fit for five years the ANN and RF models to S&P 500 of the New York Stock Exchange Market time series data, and one-year future points will be forecast. And finally, we will evaluate and compare the results of the application of the two methods and the accuracy of forecasting through the minimum root-mean-square error (RMSE) of the natural logarithms of the data to determine the model that better performs.

The paper is organized as follow. The section 2 will start with statistical learning for financial data. And then, it will present detailed summary of the RF and ANN. Section 3 will describe the selected financial data namely S&P 500 and will present the results of the RF and ANN application on the financial data. Section 4 will contain the conclusions of the work.

2. Statistical learning for financial data.

Based on some studies **(**such as the Time Series Forecasting for Nonlinear and Nonstationary Processes from Changqing Cheng et al ,2015), The financial data has some characteristic namely, non-stationary, nonlinearity and noisiness (there are variations in financial time series)

The Studies of James Gareth, 2013 on the Introduction to Statistical Learning, Statistical learning is a vast set of supervised or unsupervised tools which is based on the coefficient estimator and requires a good understanding of the data. Thus, in the case of the supervised statistical learning the inputs are associated with one or more outputs. However, those inputs in unsupervised statistical learning are not associated with output. It study presents that supervised statistical learning is the method that is most used in statistical learning to analyse financial data. Since the financial data are non-linear, we can have as statistical learning methods (based on the Studies from James Gareth,2013) in the forecast of the financial data:

- Moving beyond linearity (polynomials, step functions, splines, local regression, and Generalized Additive Models (GAMs) for regression and classification problems with a single input variable)

- Tree -based methods: it applies to both regression and classification. As Tree-based methods we have for example
  - Bagging: it is used for reducing the variance of statistical learning method. In the case of the regression, it trains on the different bootstrapped training data set in order to get the prediction and in the case of the classification, it is based on the class predicted that has the majority vote
  - The Random Forest: it is used to improve the bagging method
  - Boosting: unlike the other approaches, in the Boosting, the trees are grown sequentially

These Tree-based methods are used to improve the prediction resulting from a decision tree

- Super Vector Machine (SVM): it is used to perform both linear and non-linear classification or regression with the help of the kernel functions

- Analysis of the prediction: it consists to determine the accuracy of the estimate value. Thus, there are many indicators such as R^2 $(rsquared), RMSE$ (root mean square error) for the regression analysis , and the *Error rate* for the classification analysis that measures the estimation. The cross-validation, the bootstrap can be used to estimate the accuracy of the statistical learning methods. And therefore, the smaller the indicators, the better the estimate

2.1 The Random Forest (RF)

From Zeng Tang et all, 2019 in his study on the Stock Selection With Random Forest: An exploitation of excess return in the Chinese stock market, the RF is a classification and regression process consisting of several uncorrelated decision trees, in which all decision trees have grown under a certain type of randomization during the learning process. Its study shows that, the process of constructing a random forest is as follows:

- We first randomly generate **ntree** bootstrap sample from the original dataset: the bootstrap refers to the process of drawing the samples with replacement which means that some of the samples will be selected for several times. Furthermore, it allows to overcome the overfitting of Decision Tree

- For each bootstrap sample, we grow unpruned tree by choosing best split based on a random sample of $mtry$ (the number of variables randomly selected at each split)
  - $mtry = \sqrt{p}$ for the classification
  - $mtry = p/3$ for the regression, with p the number of predictors

- We predict new data using majority votes for classification and average for regression based on **ntree** trees: in RF, each decision tree will produce a class

prediction and the class with the highest votes will becomes the prediction of the model.

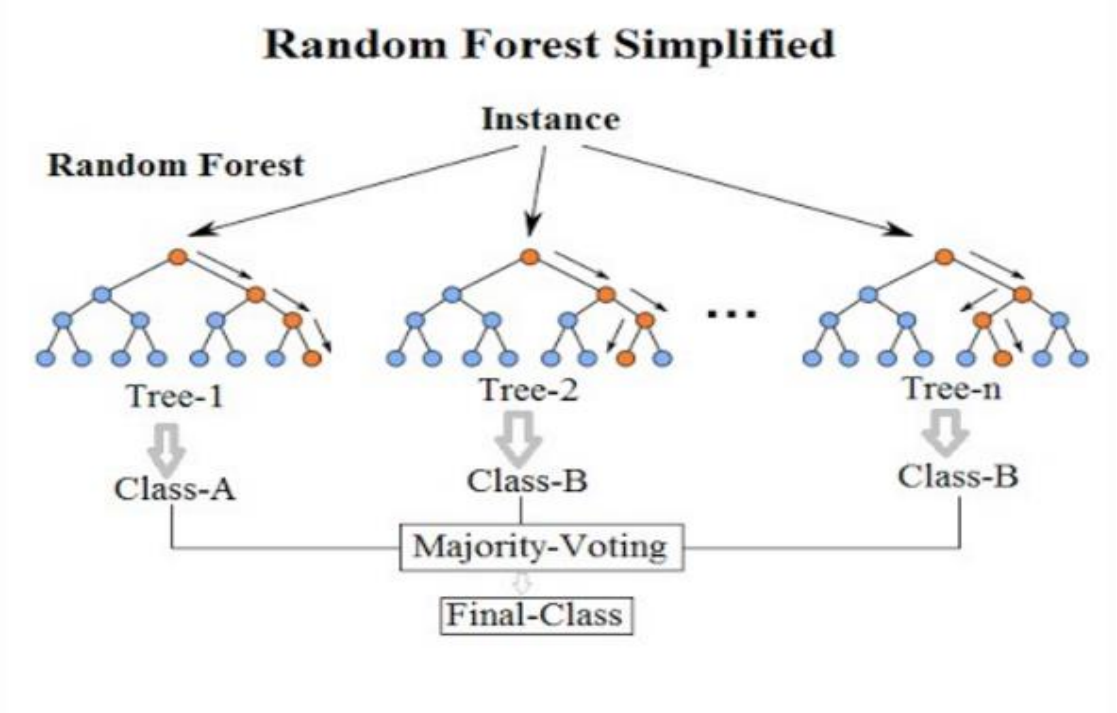

Figure 1: The Random Forest[1]

We can conclude that, the RF is a supervised classification algorithm which forms multiple decision trees at the training stage, and outputs the class with most of votes among all classes at the testing stage

The procedure that will be used to apply RF in forecasting the close values to S&P 500 index of the NASDAQ time series data is described as follows:

- Set train and test data: to form the RF model for forecasting time series, we separate the financial database into data train and data set

- Fit the data: to fit the training data, the number of trees to grow and the number of variables available for splitting at each tree node must be set in advance. A preliminary round of fine -tuning is carried out based on the plot of Out-of-bag (OOB) error whereby it is a method of measuring the prediction error of random forests to determine the best combination of the hyperparameter of the RF algorithm. The best combinations which are obtained are used in the forecasting procedure
- Prediction of the Close values on the basic of the test data and the fit data

- Evaluation of the model: to verify the estimated model fittest the historical data well, we compute the root means square error (RMSE) of the predicted data to measure the accuracy in predicting the price of the stock

[1] Wikipedia: https://i.ytimg.com/vi/ajTc5y3OqSQ/hqdefault.jpg

➢ $$RMSE = \sqrt{\frac{1}{n}\sum_{n=1}^{N}(y_t - \hat{y}_t)^2}$$

$$y_t = the\ actual\ value$$

$$\hat{y}_t = the\ forecast\ value$$

$$n = number\ of\ observations$$

Thus, the lower RMSE, the higher the accuracy of the forecasting model

2.2 Artificial Neural Network (ANN)

The ANN consist of predicting output target feature by dynamically processing output target and input predictors data through multi-layer network of optimally weighted connection of nodes (C.M.Bishop,1995). Then, the Network outputs depend on the input units, hidden units, weights of the Network and the Activation function.

- The activity of the input units represents the raw information that is fed into the network.
- The activity of each hidden unit is determined by the activities of the input units and the weights on the connections between the input and the hidden units.
- The behaviour of the output units depends on the activity of the hidden units and the weights between the hidden and output units.
- The Activation function which describes linear or non-linear connection between nodes. The various types of activation functions present in a neural network are follows: linear(it is used for the simple regression problems), ReLU (Rectified Linear Unit), sigmoid (logistic function) that is a non-linear activation function, Tanh (Hyperbolic tangent activation function)

As algorithm in the ANN, we have:

- Backward Propagation of Errors consists of finding optimal nodes connection weights by minimizing information loss measured through sum of squared errors.
- $$SSE = \frac{1}{2}\sum_{k}(Y_k - \hat{Y}_k)^2$$

$$SSE = Sum\ of\ squared\ errors$$
$$Y_k = the\ actual\ value$$
$$\hat{Y}_k = the\ predicted\ value$$

- The Resilient Backpropagation (RPROP) with or without weight backtracking: it is faster than the Backward Propagation and does not require to specify any free parameter values

- The modified globally convergent version (GRPROP)

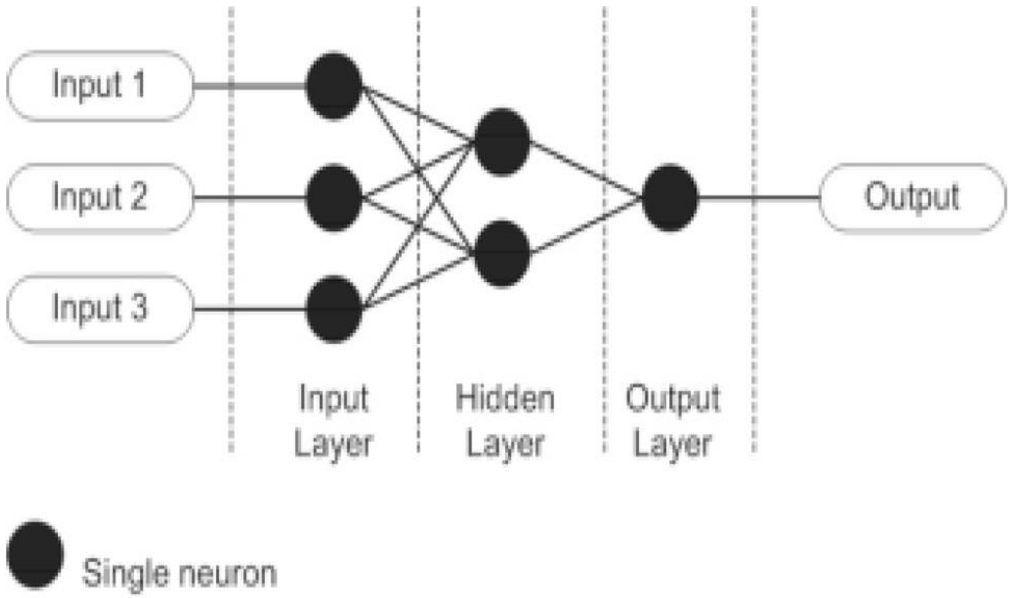

Figure 2: The Artificial Neural Network (ANN)[2]

The above graphic is a network which comprises of three layers, namely the input layers, hidden layers, and output layers. The input layer aims to receive the input values and pass on the information to the hidden layer for data processing. The hidden layer that can be in the form of single or multiple layers, is used to perform the computations on the information received from input layer and transfer it to the output layer once it completed the computations. Finally, the output layer transfers the results to the outside world.
The procedure that will be used to apply ANN model in forecasting the close values to S&P 500 index of the NASDAQ time series data can be described as follows:

- stabilize the time series with the help of the logarithms

- Set train and test data: we separate the financial database into data train and data set

- Fit the data: here, the training data is set as the input of the ANN model and the suitable number of hidden layers and neurons is determined on basic of the errors. Furthermore, these errors allow to determine the best combination of the hidden layers for the ANN algorithm. The smaller the number of hidden layers, the greater the errors. Therefore, the errors decrease as the number of hidden layers increases

- Prediction the values of the variable price: to predict the price values, we use the ANN model developed in the previous step

[2] Krenker A, Bešter J, Kos A, 2011

- Evaluation of the model: finally, the predicted price values will be evaluated to verify that the estimated model fitted well the historical data . Thus, we compute the root means square error (RMSE) of the predicted data to measure the accuracy in predicting the price of the stock

3. Application

In this case, we want to predict price values ,namely close price from the S&P 500 database, using the RF algorithm and the ANN algorithm and thus, compare them to the current values in order to determine which of the models is most appropriate.
First, we will transform the database using natural logarithms to stabilize the time series. The time series comprises 1280 observations representing S&P 500 stock price index in the period from June 1,2015 to June 30,2020, and then we will divide the database into 2 parts: the dataset of 01.06.2015 to 28.06.2019 represents training data and is used for the estimate, and the dataset of 01.07.2019 to 29.06.2020 represents the test data and is used for the prediction

3.1 Description of the data

The data used in this investigation is a time series that represents the daily scores of S&P 500 (Standard & Poor's) index of the New York Stock Exchange (NASDAQ). The S&P 500 is a stock market index that measures the stock performance of 500 large companies listed on stock exchanges in the United States. Therefore, It describes changes in stock prices in the market.
The data table breaks down into 7 columns:

- Date: it corresponds to the date of the Stock Data
- Open: it corresponds to the opening price of company's shares
- High: it corresponds to highest price at which a stock traded during the trading day
- Low: Lowest price at which a stock traded during the trading day
- Close: Last close price of the company's shares on the relevant stock exchange
- Adjusted Close: it corresponds to close price
- Volume: it corresponds to the number of shares sold, traded over a certain period of time (daily)

The data come from Yahoo-finance and goes from 01.06.2015 to 30.06.2020 working on the financial market

Source : https://finance.yahoo.com/

3.2 Application of the Random Forest (RF)

we will here apply the RF model described in section 2 on the S&P 500 stock price index for the prediction of the price values of the last year.

1. Overview over the S&P 500 index of the NASDAQ time series data

Date	Open	High	Low	Close	Adj.Close	Volume
Min	1833	1847	1810	1829	1829	1.3e+09
1st Qu.	2161	2166	2150	2161	2161	3.25e+09
Median	2573	2584	2561	2577	2577	3.58e+09
Mean	2535	2547	2521	2535	2535	3.81e+09
3rd Qu.	2836	2852	2820	2839	2839	4.03e+09
Max.	3380	3394	3379	3386	3386	9.04e+09

1. Display the stock price time-series

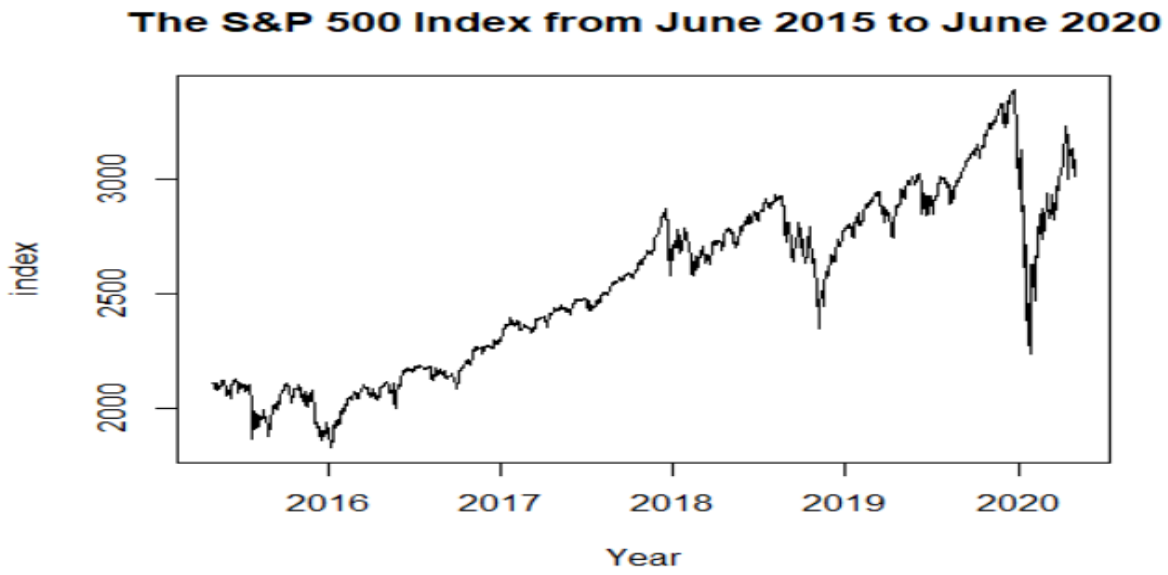

Figure 1. S&P 500 Index Daily Data

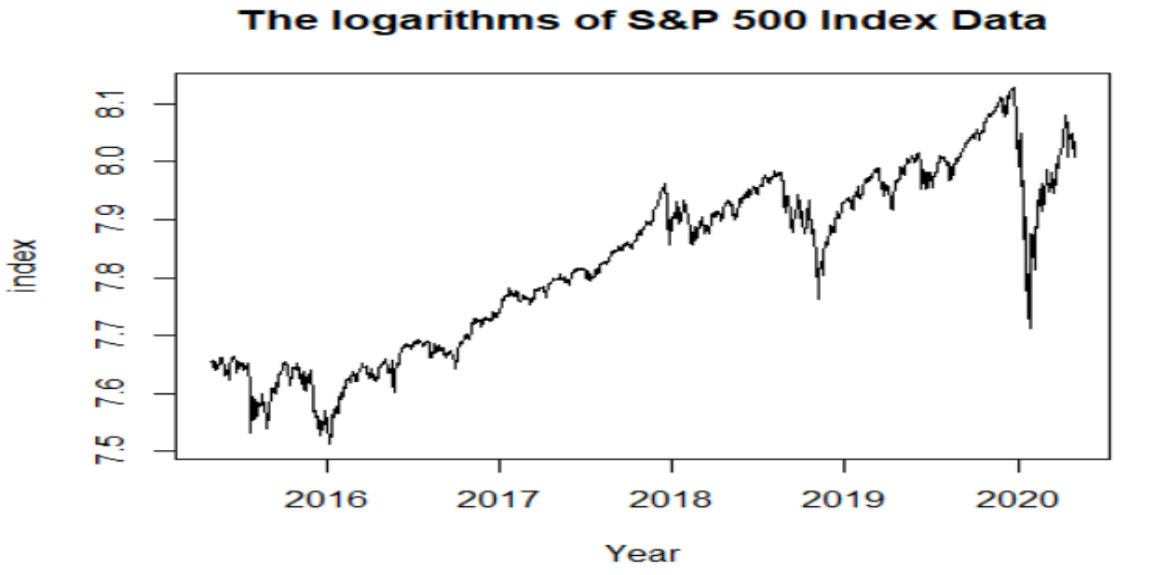

Figure 2. the logarithms of S&P 500 Index Daily Data

Figure 1 represents the original time series and indicates that the time series is non-stationary. It shows a continuous increase in stock market indices until the end of 2019 and then, a sharp decline in stock market indices at mid-2020 caused by the appearance of the corona pandemic virus (Covid19), which causes a slowdown in economic activity. And after that, the stock market indices increase again. The natural logarithms of the series are displayed in figure 2. Its curve is the same as well the original data

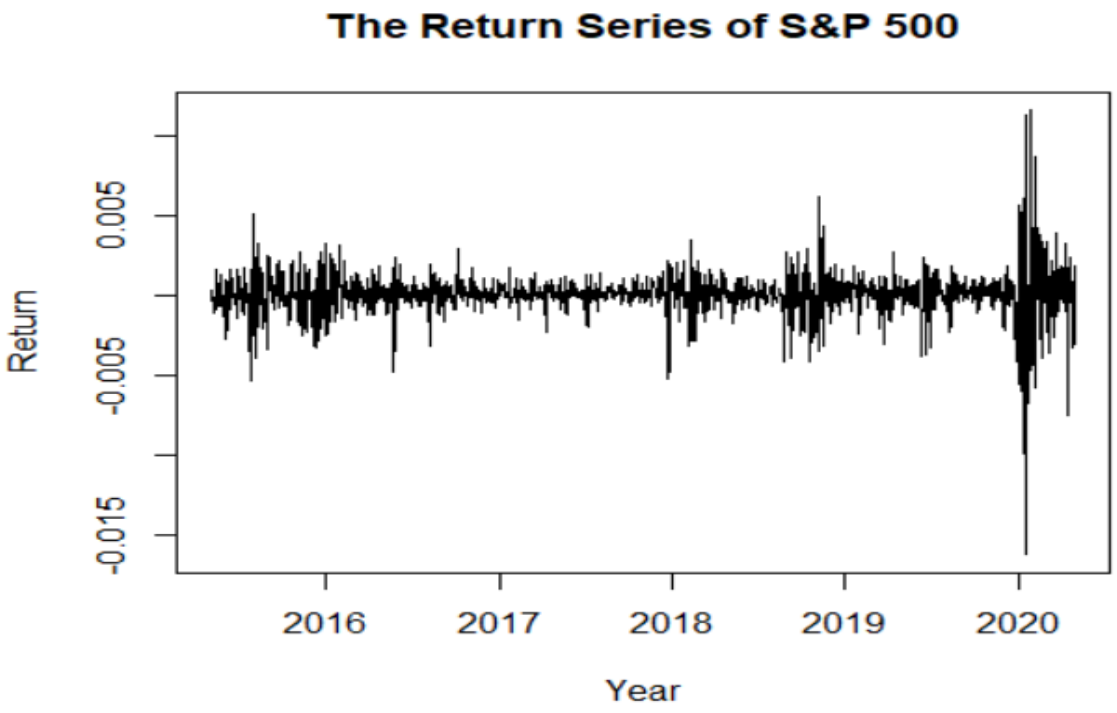

Figure.3 The Return series of S&P 500 Index Series Data

The figure.3 represents the return series of S&P 500 Index Series Data and shows that it is stable and consequently stationary. The deviation is minimal and stable about zero until the begin 2020, and then, it increases and becomes unstable. This is caused by the epidemic of the covid19 which leads to a fall in the close price of the index

2. the find out the best combination of the hyperparameter of the RF algorithm

The fitting of the data in RF depends on the number of variables available for splitting at each tree node ($mtry$). Thus, the number of variable available is equal to the one that has the smallest error

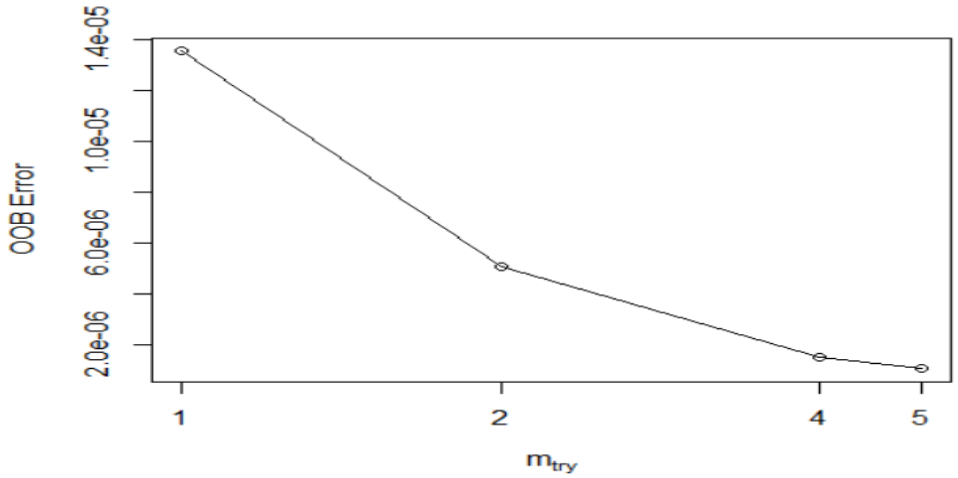

Figure 4 . the best combination of the hyperparameter

The figure 4 shows that the best combination of the hyperparameter for the forecasting procedure is 5, which corresponds to the smallest error (the Out-of-bag (OOB). This error is equal to 1.030718e-06.

3. Display the errors and the importance of variables in the Random forest estimate

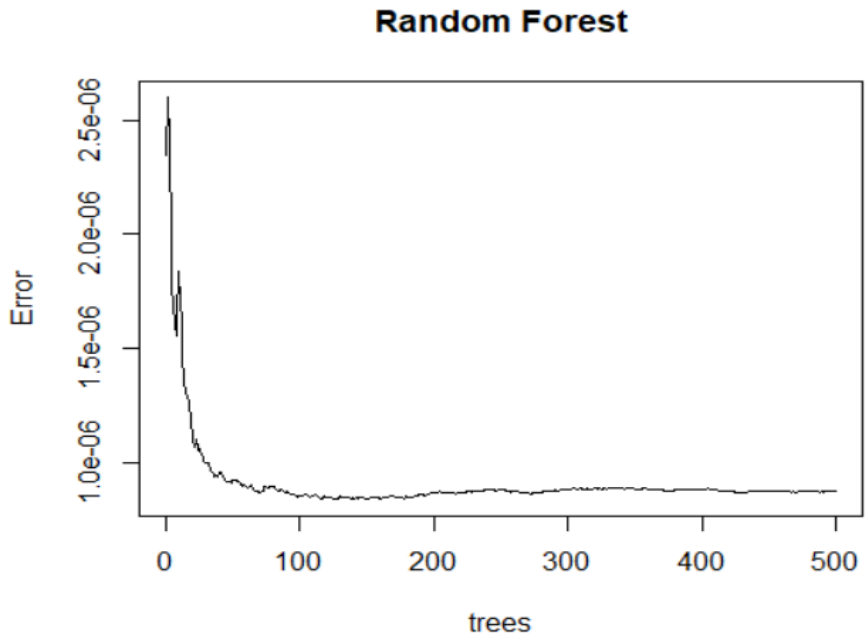

figure 5. the errors of Random forest

the figure 5 shows the relation between Mean square error (MSE) and the number of trees. In this case, the MSE decreases when the number of trees increases and therefore the optimal nu mber of trees is equal to 200. This number corresponds to the smallest error (MSE=8.724697 e-07)

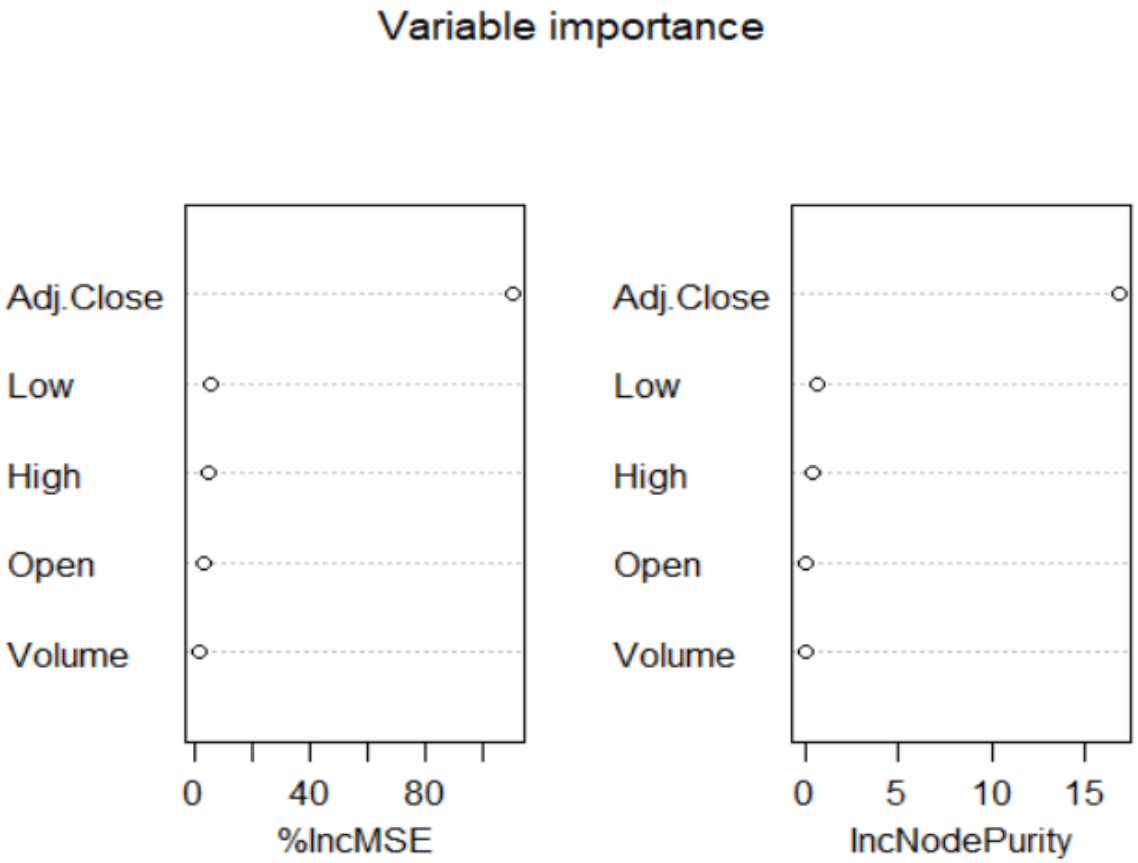

figure 6. The variable important in the estimate

The higher the value of Mean Decrease Accuracy (%IncMSE) and Mean Decrease Gini (Inc NodePurity), the higher the importance of the variable in the model. In the plot show above, A dj.Close is most important variable .

4. Display the observed, fitted and forecasted values

We compare the curves of the fitted and forecast close values with that of the observed values in a graph.

through the RF model, we obtain the forecasting results or predicted values for S&P 500 index of the NASDAQ time series using the test set which is represented by the red line and the fitted values (blue line) using the training set as show below in figure 7. The curves of the observed and fitted values are the same. However, through this figure we can observe that the values of forecasting are not total identical to the actual values of the times series.

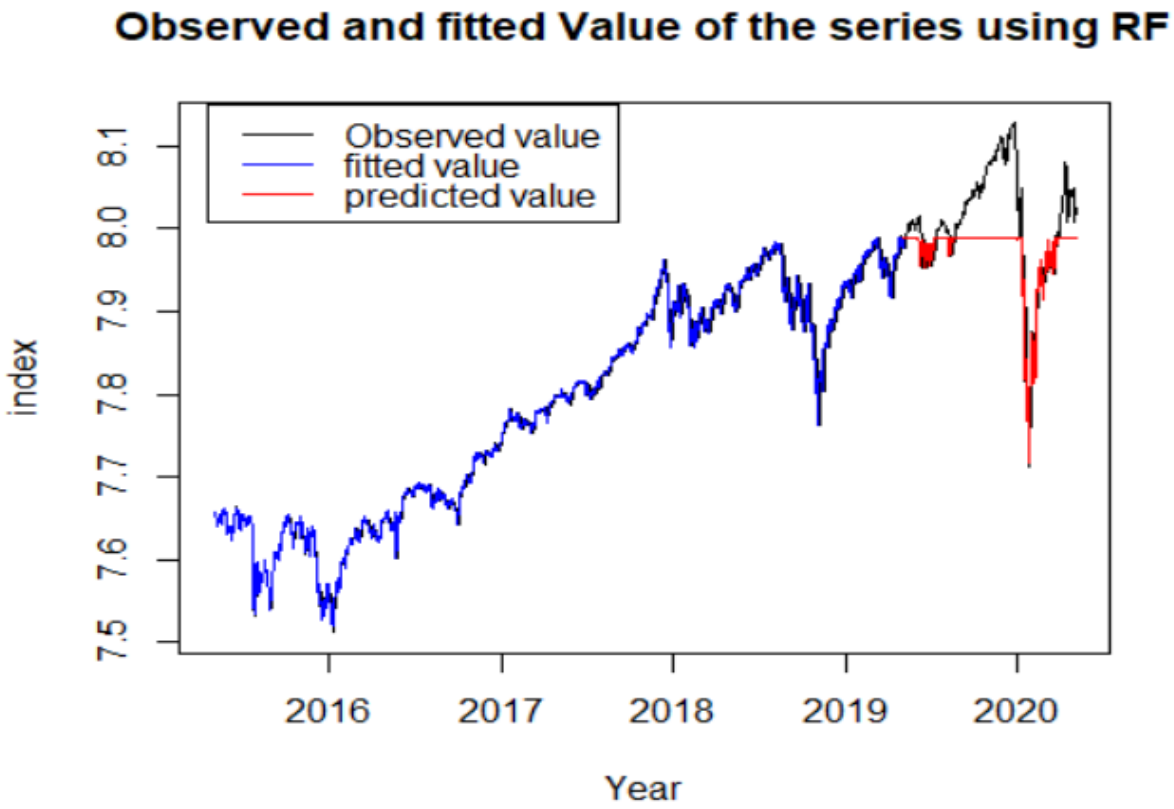

Figure 7. Observed and Fitted Values of the Logarithms of the Series using RF

5. Display the residuals of the RF model

Figure 8 shows the residuals of the RF model on S&P 500 dataset and indicates that they are very low and for the majority close to zero between 2016 and 2019 (training set) and then, go beyond to zero between 2019 and 2020 (test data)

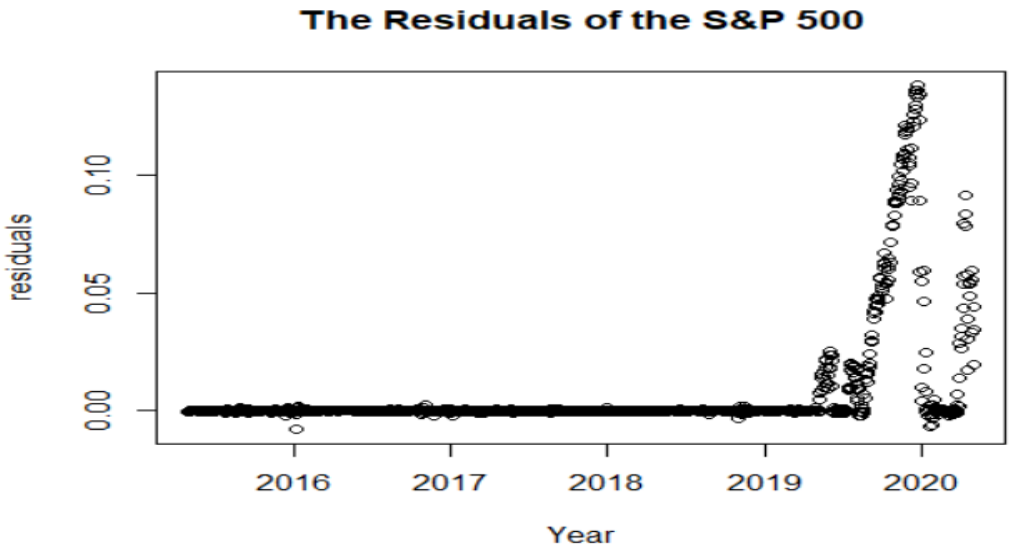

Figure 8. the residuals of the RF model of the logarithms series

6. Display the forecast and observed values for the 10 first and last days using the RF

For the first 10 days

Date	Close	Forecast
01.07.2019	2964.33	2949.916
02.07.2019	2973.01	2949.894
03.07.2019	2995.82	2949.869
05.07.2019	2990.41	2949.869
08.07.2019	2975.95	2949.869
09.07.2019	2979.63	2949.869
10.07.2019	2993.07	2949.894
11.07.2019	2999.91	2949.894
12.07.2019	3013.77	2949.869
15.07.2019	3014.30	2949.869

For the last 10 days

Date	Close	Forecast
16.06.2020	3124.74	2949.976
17.06.2020	3113.49	2950.075
18.06.2020	3115.34	2950.075
19.06.2020	3097.74	2949.976
22.06.2020	3117.86	2950.075
23.06.2020	3131.29	2950.075
24.06.2020	3050.33	2949.976
25.06.2020	3083.76	2950.075
26.06.2020	3009.05	2949.976
29.06.2020	3053.24	2950.075

The table presents actual observed and forecast Close values of the daily scores of S&P 500 index time series (test data) for the 10 first and last days. It indicates that the tendencies of the forecast Close values do not converge to observed Close values. The RMSE of the natural logarithms of S&P 500 index time series using RF method is equal to 0.05405772

3.3 Application of the Artificial Neural Network (ANN)

In this section, we will apply the ANN model described in section 2 on the time series for S&P 500 daily stock price index. The data has 1280 observations and is divided into training set and test set. The number of observations in the training set and test set is the same as the number of observations used in the Random Forest Model. That is 1028 observations of the series used as training set for fitting the model and 252 observation of the series used as test set for the prediction the models. The fitting of data in ANN depends on the number of hidden layers. The less the errors in the estimation the better the model

1. The find out the suitable number of hidden layers

It is determined on basic of the errors. The errors are determined from RMSE (root means square error). In this case we will work with one hidden layer. To find the suitable number of units of the neuron for the ANN algorithm of the S&P 500 model, we will start with one unit in the hidden layer and progressively increase the number of neurons up to 15. Finally, we will choose the number of neurons with the smallest error.

Number of neurons in the hidden	RMSE
1	0.23484155
2	0.23485591
3	0.23485874
4	0.01843224
5	0.02098598
6	0.06577817
7	0.02070733
8	0.01410909
9	0.01118283
10	0.03718731
11	0.02773552
12	0.02123317
13	0.01819200
14	0.01909540
15	0.01922632

In this case, the suitable number of hidden is 9 and the smallest error (RMSE) associated with this error is equal to 0.01118283.

We can conclude that the network with 9 units in the hidden layer und resilient propagation algorithm as method is the best for the prediction of the price values of the S&P 500 index time series.

2. Display the stock price, the forecast and predicted value of the logarithms of the series

The figure 9 shows the logarithms of the S&P 500 index time series (observed values) together with its fitted values on the training set and the forecast values on the test set. Through this figure we find that the values of the forecasting are almost identical to the observed values of the time series. So , we can conclude that the ANN model predicts exactly the actual values

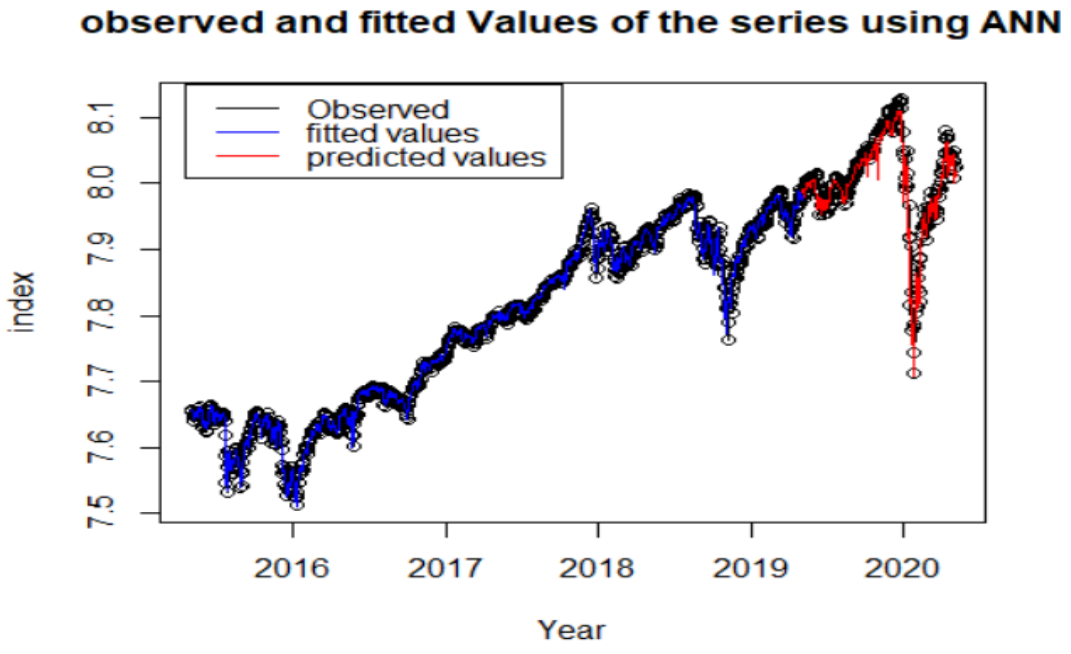

Figure.9 Observed and Fitted Values of the logarithms of the Series using ANN

1. Display the Residuals of the ANN model of the logarithms of the series

The figure 10 below presents the residuals of the ANN model of the logarithms of the S&P 500 index time series and indicates that they are very small, with the majority close to zero

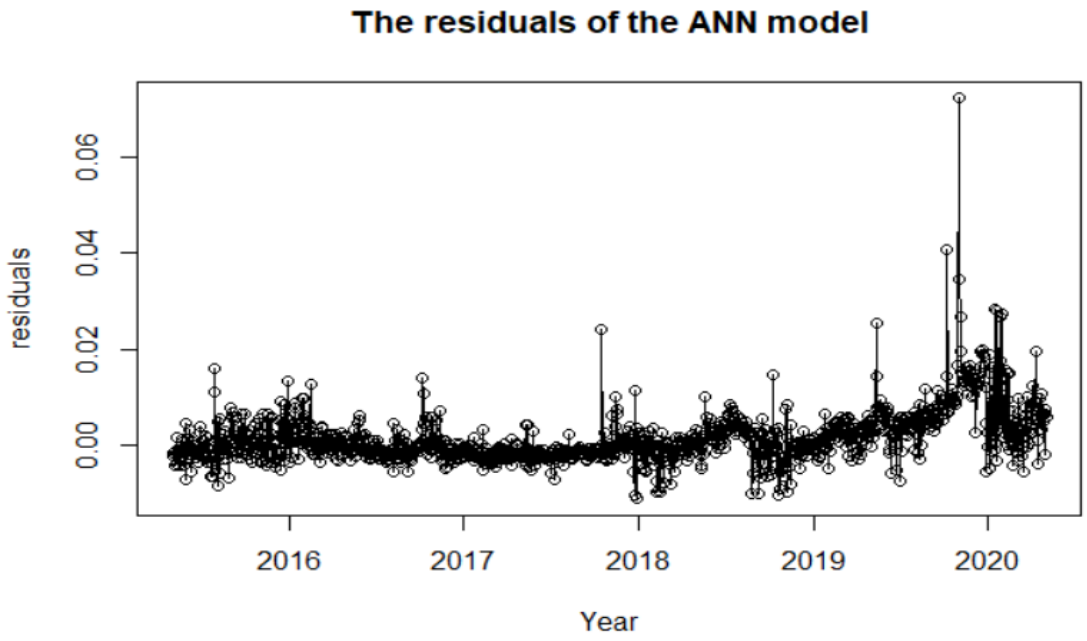

Figure.10 The Residuals of the ANN model of the logarithms of the series

2. Display the observed and forecasted values of the 10 first and the last 10 days of S&P 500 time series using the ANN

Date	Close	Forecast
01.07.2019	2964.33	2955.316
02.07.2019	2973.01	2954.498
03.07.2019	2995.82	2920.609
05.07.2019	2990.41	2947.914
08.07.2019	2975.95	2952.608
09.07.2019	2979.63	2955.579
10.07.2019	2993.07	2972.628
11.07.2019	2999.91	2979.280
12.07.2019	3013.77	2984.972
15.07.2019	3014.30	2985.478

For the last 10 days

Date	Close	Forecast
16.06.2020	3124.74	3109.863
17.06.2020	3113.49	3103.588
18.06.2020	3115.34	3095.610
19.06.2020	3097.74	3064.136

22.06.2020	3117.86	3097.331
23.06.2020	3131.29	3115.864
24.06.2020	3050.33	3055.924
25.06.2020	3083.76	3063.695
26.06.2020	3009.05	2985.981
29.06.2020	3053.24	3035.302

In conclusion, through the analysis and several rounds of testing with the ANN model, we have identified the number of units in the hidden layer equal to 9 as the best selection for the analysis of the S&P 500 index time series and used in addition this parameter to train the model again. Following this, we have predicted the test set. Finally, we obtain a RMSE = 0.01118283.

figure 9 gives a comparison of forecast Close values with actual Close values, while above table presents actual observed and forecast Close values of the daily scores of S&P 500 index time series (for the 10 first and last days). The results shown in Figure 9 and Table indicate that the tendencies of the forecast Close values curve are almost identical to those of the observed Close values curve, and the fitted values correspond to the observed values very well

Table: Comparison of the prediction errors of RF and ANN models

Model	ANN	RF
RMSE	0.01118283	0.05405772
R squared(R^2)	0.9966571	0.9218859

From table we can conclude that ANN performed slightly better with better performance metrics value(lower RMSE, and higher R-squared)

3. Conclusion

In this paper, we have applied the RF and ANN to financial forecasting. Furthermore , we ha ve fitted the RF and ANN model to S&P 500 index of the NASDAQ time series data (train da ta) and used these models to forecast future Close price (test data). The results of applying the RF and RF models are compared through the RMSE results. The RMSE of the natural logarithms of S&P 500 index time series are equal to 0.01118283 and 0.05405772 respectively for ANN model and RF model. Finally, we find that the ANN model has the minimum RMSE. So, we can conclude through these results that it is more suitable in the forecasting of the S&P 500 financial data than the RF model

References

1. Mahmoud K. Okasha, (2014), Using Support Vector Machines in Financial Time Series. International Journal of Statistics and Applications, 4(1),28-39, DOI: 10.5923/j.statistics.20140401.03
2. L. Breiman, (2001); Random Forests; Mach. Learn., 45 (1); pp. 5-32
3. Wenji Mao, Fei-Yue Wang, (2012), New Advances in Intelligence and Security Informatics, 91-102
4. Chang Sim Vui, Gan Kim Soon, Chin Kim On, and Rayner Alfred, Patricia Anthony, (2013), A Review of Stock Market Prediction with Artificial Neural Network (ANN), Conference: Control System, Computing and Engineering (ICCSCE), DOI: 10.1109/ICCSCE.2013.6720012
5. Changqing Cheng et al, (2015), Time Series Forecasting for Nonlinear and Nonstationary Processes: A Review and Comparative Study, DOI: 10.1080/0740817X.2014.999180
6. James, Gareth, Witten D., Hastie T., Tibshirani R., (2013). An introduction to statistical learning. Vol. 112.
7. Zheng Tan, Zigin Yan, Guangwei Zhu, (2019), Stock Selection With Random Forest: An exploitation of excess return in the Chinese stock market, Journal homepage: www.heliyon.com, Volume 5
8. Wei Hong Hong, Jia Hui Yap, Ganeshsree Selvachandran, Pham Huy Thong and Le Hoang Son. Complex & Intelligent Systems (2020): Forecasting mortality rates using hybrid Lee–Carter model, artificial neural network, and random forest, Complex and Intelligent Systems, https://doi.org/10.1007/s40747-020-00185-w
9. Wikipedia: https://i.ytimg.com/vi/ajTc5y3OqSQ/hqdefault.jpg
10. C.M.Bishop (1995), Neural-Networks for Pattern Recognition, Oxford University Press
11. Krenker A, Bešter J, Kos A (2011), Introduction to the Artificial Neural Networks. In: Suzuki K (ed) Artificial Neural Networks - methodological advances and biomedical applications, 1–18., DOI: 10.5772/644