Stock Price Prediction Using Empirical Mode Decomposition

Enes Sinan Parildi

December 20, 2017

Abstract

Predicting future values of various financial data can be important for various business applications. However these financial data may exhibit high level of non-linearity and non-stationary which may severely impair accuracy of a predictive model. Pre-processing raw data with empirical mode decomposition algorithm has been found to improve the accuracy of predictive models. This paper proposes hybrid approach which involves emd before predictive models to forecast one step future stock values of Apple Inc. Artificial neural network(ANN) , recurrent neural network ,auto-regressive integrated moving average(ARIMA), Support Vector Regression(SVR) and Gaussian Process Regression(GP) has been tried as predictive models.

1 Introduction

Stock markets are complex dynamical systems affected by various factors such as socio-political events, news to investors behavior and preferences.[1] Dependence of too many such factors gives stock markets behaviour like Brownian motion or wiener process but not exactly. So the price movements are very random and unpredictable in a sense that they are highly non-stationary and non-linear which make it close to impossible to predict accurately with statistical model such as arima which assumes stationary and linearity.[2] On the other hand machine learning based models such as in our case ANN and RNN can handle non-stationary and non-linearity to some degree , because of that they have recently gained widespread usage in forecasting time series.[3] However they often suffer from over-fitting and getting trapped in local minima.[4] This may lead also impair accuracy. To improve accuracy the raw time series must be pre-processed beforehand. Empirical Mode Decomposition proposed by Huang can be excellent tool for preprocessing non-stationary time series. Since it has a capacity to decompose the signal to its adaptive basis functions. These functions are called intrinsic mode functions(imf). This basis functions are completely data dependant and adaptive make emd suitable for non-stationary data. This paper proposes four hybrid models namely EMD-ANN , EMD- RNN, EMD-ARIMA

,EMD-SVR and EMD-GP to forecast one day ahead stock prices Apple Inc. The basic procedure is firstly decomposing via EMD , decomposed signals are predicted independently by aforementioned predictive models, finally predictions obtained from decomposed signals are combined to get final predictions.To our knowledge this method and work although similar to this are long present ,widespread and mature for wavelet decomposition[4] is quite new and novel for empirical mode decomposition. Performance was measured with respect to mean square error and classification accuracy improvement compared to prediction without pre-processing with emd.

2 Methodology

2.1 Empirical Mode Decomposition

Empirical Mode decomposition is a algorithm that reduces signal to the collection of intrinsic mode functions(imf) which as a defining characteristic has this two conditions [6]:

a) In the whole data set , the number of extrema and the number of zero crossings must either equal or differ at most by one.

b) At any point, the mean value of the envelope defined by the local maxima and the envelope defined by the local minima is zero.

Some number of imf are computed iteratively until we are left with final residue function which is constant ,monotonic function or a function with only one maximum and one minimum from which no more imf can be extracted.[2]. So at the end we get n imf components and a residue such that

$$X(t) = \sum_{j=1}^{n} C_j + r_n$$

Where C is imf components r is residue and number of components n depends on original signal. This simple fact is very crucial for our method. Since it implies that we can get appropriate combination of predictions by simply summing predictions belong to each imf component and residue. Although we will also look at combination method based on linear regression, linear regression brings minuscule mse improvement compared to simple summing. Moreover another important aspect of emd that make it so important in our application is that later imf components have smooth low frequency characteristic[2] which makes them easier to be extrapolated by prediction model. This is the main reason behind the performance improvement comes with emd. Regardless of underlying prediction method we will see substantial improvement with emd highlighting importance of signal processing based pre-processing techniques in machine learning.

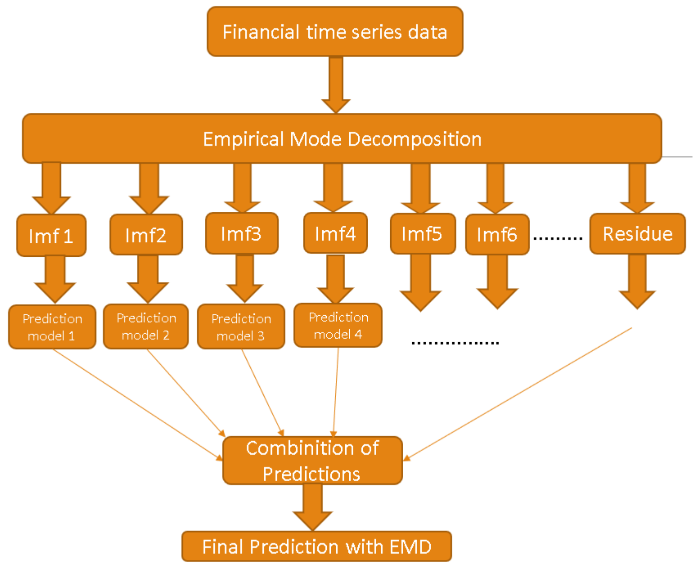


Figure 1: Block Diagram of the method

2.2 Artificial Neural Networks

Deep learning based methods such as artificial neural network has gained attention in financial time series prediction for its ability approximate complex mapping relations which can be highly non-linear. So neural network can catch non-linear characteristics of financial data.[6] Among these deep learning methods classical feed-forward neural network is widely preferred because of its simplicity.

There are some limitations regarding of these neural network in terms of processing time series type inputs, first of all these networks accept fixed size inputs, secondly neural network often need to have normalized inputs for faster and better convergence.[7] To solve first problem we choose fixed window of size 5. So one data point is consist of the value of five consecutive days and value of next day.

$$X = \left\{x_t, x_{t+1}, x_{t+2}, x_{t+3}, x_{t+4}\right\}$$

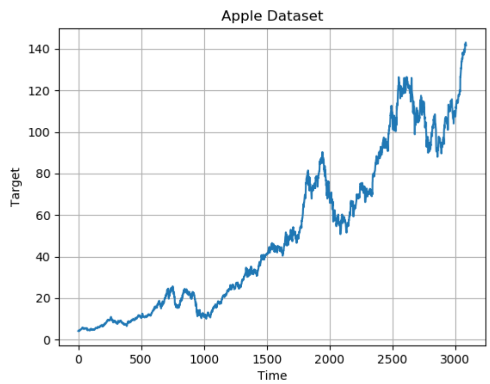
$$f(X) = x_{t+5}$$

Secondly to normalize data, first difference was taken. So predictions are based on getting change in price right. Second type of normalization can compress data points to [0,1] which is described as:

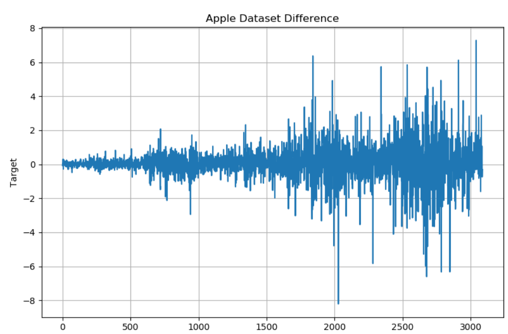
$$x_{new} = \frac{x_{max} - x_{old}}{x_{max} - x_{min}}$$

Results with this type normalization almost bring no improvement at all. So we just use first difference normalization and omit the results obtained from this prediction type.

The one downside of neural networks is that there are many hyperparameters to optimize such as input node dimension, number of hidden layer , number of node in hidden layers and activation function. It is in general very computationally intensive to search optimal parameters. In this case we just choose architecture after some number of trial of different parameters rather than exhaustively search all parameters. At the end 3 hidden layer each with 20 nodes using tanh activation function is chosen.



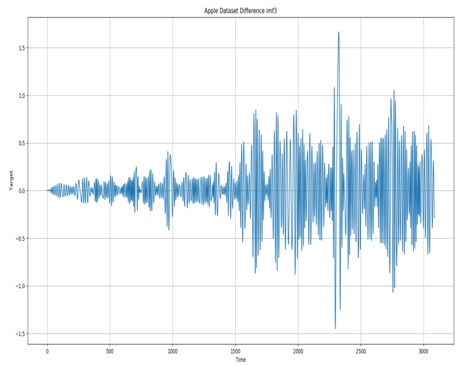
(a) Original Apple Inc end of the day stock prices for 3089 day



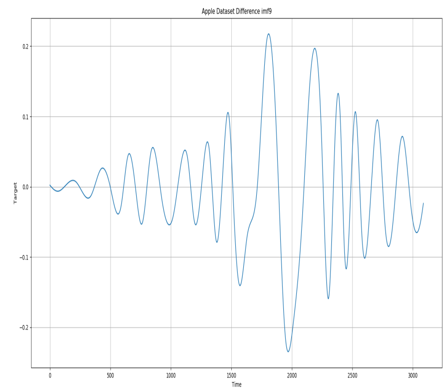
(b) First difference transformation of apple dataset

2.3 Autoregressive Integrated Moving Average

Autoregressive integrated moving average (ARIMA) model is a generalization of an autoregressive moving average (ARMA) model. ARMA model are only



(a) Third imf component of the apple difference data



(b) Ninth imf component of the apple difference data

suitable for stationary time series. On the other hand ARIMA tries to model non-stationary time series by trying to make it stationary by applying finite differencing of the data points.[8] In general after differencing operation time series is to fit combination of moving average and autoregressive components. Their optimal order was determined by grid search over the optimal accuracy for validation set. Arima is in general much weaker than deep learning based methods.[9] Since arima is linear model , so it has limited expression power , unlike neural networks in general. This is also reflected in results that we get. Arima performed poorly than other methods we studied. However incorporating emd will lead to some improvement.

$$X_t - \alpha_1 X_{t-1} - \cdots - \alpha_{p'} X_{t-p'} = \varepsilon_t + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q},$$

Figure 4: General ARMA expression

2.4 Recurrent Neural Networks

Recurrent Neural Networks is a special type of neural network for modelling time series. General structure resembles feed - forward neural networks with the exception of feedback connection. With these connections RNN can have a internal memory that remembers the past information.[10] This enables RNN to discover temporal correlations in the data which is crucial for learning financial time series.

Recurrent neural network achieved state of the art performance in image captioning, speech recognition, and many language tasks.[11] So it is worthwhile to use them in stock price prediction. In principle they have much more capacity than feed-forward neural network for time series prediction but they are also harder to train and tune. The architecture that we used consist of 256 lstm(Long short term memory) cells with 2 layer. Lstm is special type cell observed to archive excellent performance.[13] Below you can see lstm diagram RNN that we use is basically consist of bunch of stacked lstm cells.

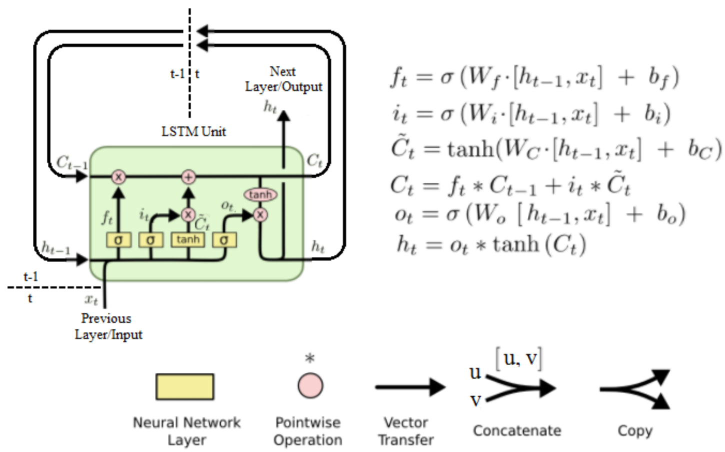


Figure 5: LSTM diagram

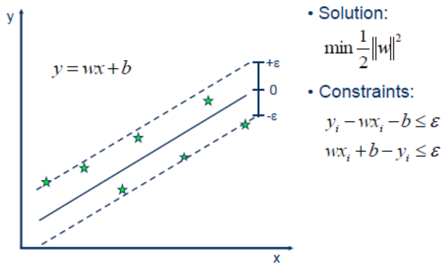
2.5 Support Vector Regression

SVR(Support Vector Regression) is version of well-known support vector machines applied to regression problem. It is developed by Vapnik[12]. This method involves finding optimal weights and bias after the data are projected to higher dimension according to its associated kernel. In our application polynomial with second degree, linear and Radial basis function kernel are used. Optimal weights are determined by minimizing norm of weights such that absolute value of residuals belongs to each training points should be smaller than some chosen epsilon. We use validation set and grid search to find best optimal kernel parameters and epsilon. For support vector regression data point creation is same as we did in artificial neural networks. We use again previous 5 days.

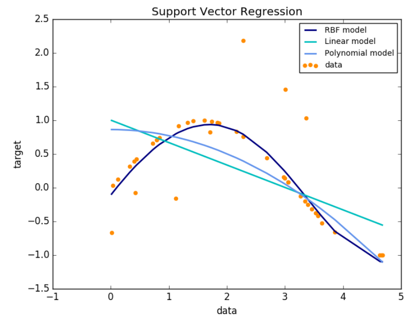
2.6 Gaussian Process Regression

A Gaussian process(GP) as its simplest explanation defines probability distribution over function. So a gaussian process contain all regression curves with its associated probability.

$$f(x) \sim GP(m(x), k(x, x^{'}))$$



(a) Review of SVR method



(b) Comparison of kernels

This give them extreme flexibility to them compared weight space based regression method like linear or polynomial regression. In gaussian process inference takes place directly in function space.[14] Inference can be done with maximum aposteriori estimation using training data as a evidence. Gaussian process can be defined simply by choosing apriory mean and covariance function at first and then with known data we update our beliefs getting posterior mean and covariance. To get prediction, we can simply choose posterior mean and evaluate it at test points. At these test points we also receive posterior variance which can give information about uncertainty in prediction. This uncertainty information may lead important future extensions regarding financial time series forecasting. Uncertainty information can help investors in terms of risk management. Simply a investor may forfeit its money or invest less money when standard deviation is large as we have less certainty in that case. Moreover we may get lowerbound or upperbound estimation with the standard deviation information rather than mean estimation, these can be useful depending on application.

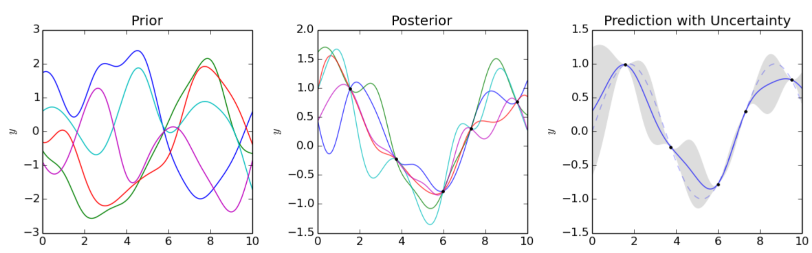


Figure 7: Inference in gaussian processes

We parse the data same way as ANN and SVR for gaussian process regression with dimensionality of five step. For gaussian process regression choice of apriori covariance function is important. It influences general characteristics of sampled functions. In our application we use sum of RBF kernel and Matern kernel.

Exact definitions of these kernels can be easily found , since they are commonly used.

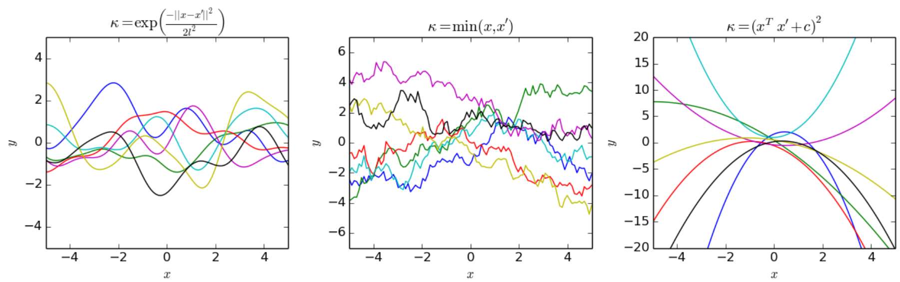


Figure 8: Example of sampled functions according to prior covariance function

3 Results and Conclusion

We will assess the performance in terms of two things. First we will look at mean square error(MSE). It is defined as:

$$E = \frac{1}{n}\sum_{j=1}^{n}(y_{true} - y_{prediction})^2$$

Note that mse is not best error measure for stock price forecasting since it ignores scaling of input but in our case we want to see performance improvement comes with emd preprocessing. In this case mse can assess performance accurately. In general mean absolute percentage error can be used:

$$MAPE = \frac{1}{n}\sum_{j=1}^{n}\left|\frac{y_{true} - y_{prediction}}{y_{true}}\right|$$

One problem with this method when any of the true value is zero it cannot be used , in our case since we work on first difference, we may encounter zero value. So mape is not suitable.

Secondly, in some cases rather than predicting the exact value of the next day stock price , we may only want to know whether the price will go up or down. So we may want to look at binary classification performance. Accuracy measure in classification is defined as number of correctly classified labels divided by number of all labels. This measure can be misleading when we have unbalanced class distribution. This is something we can expect from stock movement , a stock may have general trend to go up or down, so the class distribution can be unbalanced. Precision , recall and f1 score measure which are derived from confusion matrix can be used in this case. For all algoritms first 2800 days is used as training phase , we tried to predict remaining days.

The performance of all algorithms without emd is almost same with the exception of ANN slightly better. This illustrates the importance of EMD pre-processing , we have almost no meaningful prediction without EMD , even prediction of whether the stock price will go up or down next day is completely random close to 0.5 accuracy. On the other hand with EMD we see true power ranking of our predictive algorithms. Deep learning based methods preformed best , and the performance of SVR and GP comes after them, the worst is arima as we expected but even in arima we see substantial improvement with emd. We also omit results belongs polynomial kernel svr since they were very poor. At the end the best one is actually is EMD-ANN with linear regression combining but the problem is that we assume that predicted values are known so it cannot be used in practise for now , we just try to find out whether we have better fit than summing all the data. In this case most optimal linear combination of emd prediction increase mse performance to some level , but not in the classification performance. Gaussian process gave surprisingly good performance even it is a in principle much simpler model than deep learning based methods and even SVR.

In general all classifiers work much better in latter imf components which have significantly low frequency profile , but even for first imf component fitting quality is much better than original data , but for all imf components there are fitting errors, so we see substantial error at the end when the combination is done. Interestingly even mse error of first imf component is much better than combination. It is around 0.53 for ANN and for subsequent imf components it is even lower than 0.53. The fact that final combination have the fitting error much larger than all of the individual imf components shows that there is error propagation. I think this issue can be interesting research topic for future extensions. There is definitely a room for improvement.

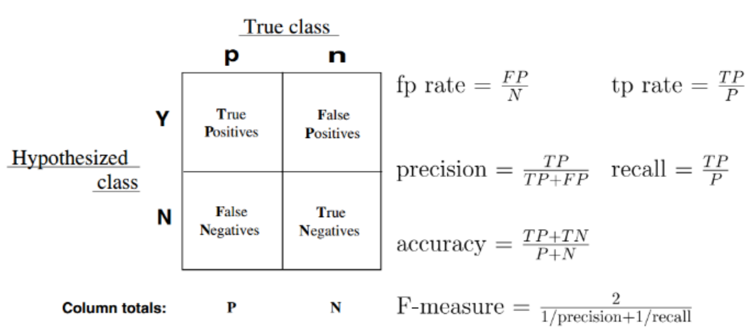
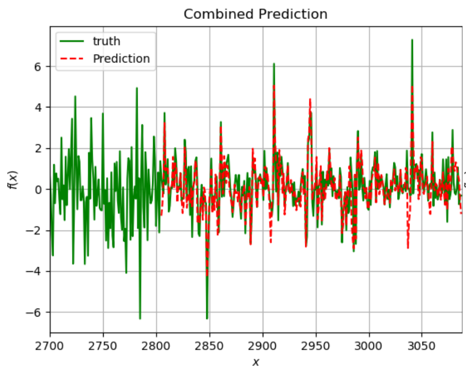
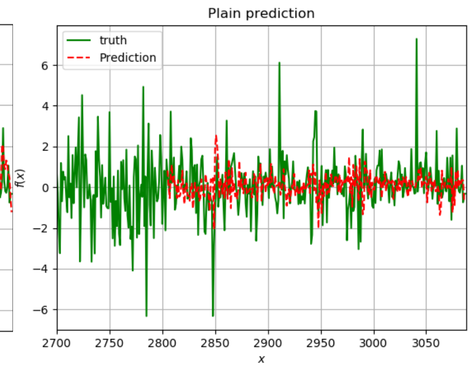


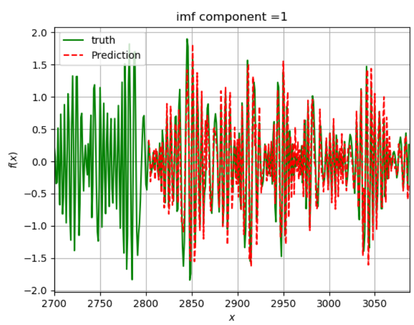
Figure 9: Review of different measures



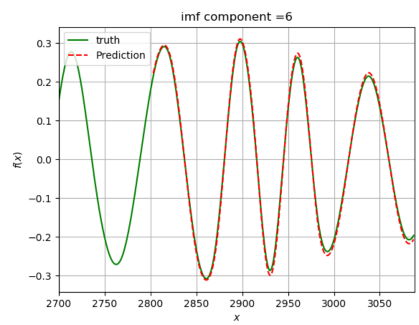
(a) EMD-ANN result



(b) Plain-ANN result



(a) Fitting of Second Imf component by ANN



(b) Fitting of Seventh Imf component by ANN

4 Possible Extensions

There are many open possibilities to extend this work. The accuracy and performance for some algorithms can be improved with more rigorous hyperparameter optimization. We have limited time to do optimization properly.

Secondly , ensemble methods can be used in conjunction with discrete wavelet transform. Simply we get final prediction from both emd and discrete wavelet transform and combine them with averaging. Ensemble based methods which involves training bunch of different models with disjoint subset of training data can be also be used directly with just emd pre-processing but it would probably be too cumbersome to deal with large number of models.

As one final extension modified version of EMD which is called ensemble emd [15] can be used for stock price prediction. EEMD utilize noise assisted data analysis method and it provides better decomposition[15]. EEMD is calculated by simply taking emd of noise added original data many times and taking mean of each imf component ensemble. As I mentioned before probabilistic models

	MSE
ANN	1.64
EMD- ANN	0.56
EMD-ANN-Linear Regression	0.52
ARIMA	1.69
ARIMA-ANN	1.32
RNN	1.65
RNN-ANN	0.59
RBFSVR	1.68
EMD-RBFSVR	0.67
LINSVR	1.69
EMD- LINSVR	0.66
GP	1.67
EMD-GP	0.65

(a) Result table with respect to MSE ARIMA-ANN should be EMD-ARIMA

	Accuracy	Precision	Recall	F1Score
ANN	0.491	0.351	0.424	0.384
EMD- ANN	0.761	0.695	0.754	0.723
EMD-ANN-Linear Regression	0.747	0.703	0.725	0.714
ARIMA	0.505	0.362	0.402	0.383
EMD-ARIMA	0.647	0.407	0.679	0.509
RNN	0.517	0.393	0.458	0.423
EMD-RNN	0.709	0.669	0.680	0.674
RBFSVR	0.529	0.234	0.410	0.298
EMD-RBFSVR	0.708	0.695	0.669	0.681
LINSVR	0.505	0.234	0.410	0.298
EMD- LINSVR	0.715	0.718	0.671	0.694
GP	0.512	0.385	0.449	0.415
EMD-GP	0.770	0.732	0.750	0.741

(b) Result table with respect to classification performance

such as gaussian processes can provide many additional interesting information that other models cannot give. Bayesian neural networks is probabilistic version of regular neural networks in which weigths are random variables. It can be a alternative to Gaussian Processes worth implementing when speed is crucial.[16] They have also edge over regular neural networks in as sense that they also give uncertainty information.

5 References

1. Novak, M. G., Velušček, D.,Prediction of stock price movement based on daily high prices. Quantitative Finance,doi:10.1080/14697688.2015.1070960(2015).

2.N. E. Huang et al. Applications of Hilbert–Huang transform to non-stationary financial time series analysis. Applied Stochastic Models in Business and Industry 2003; 19(3): 245–268.

3. Gamboa, John Cristian Borges. ”Deep Learning for Time-Series Analysis.” arXiv preprint arXiv:1701.01887 (2017).

4. Jothimani, Dhanya, Ravi Shankar, and Surendra S. Yadav. ”Discrete wavelet transform-based prediction of stock index: a study on National Stock Exchange Fifty index.” arXiv preprint arXiv:1605.07278 (2016).

5. The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis Norden E. Huang, Zheng Shen, Steven R. Long, Manli C. Wu, Hsing H. Shih, Quanan Zheng, Nai-Chyuan Yen, Chi Chao Tung, Henry H. Liu Proc. R. Soc. Lond. A 1998 454 903-995; DOI: 10.1098/rspa.1998.0193. Published 8 March 1998

6.Vui, C. S., Soon, G. K., On, C. K., Alfred, R., and Anthony, P. (2013).

A review of stock market prediction with artificial neural network (ann). In Control System, Computing and Engineering (ICCSCE), 2013 IEEE International Conference on , pages 477–482. IEEE

7.Zhang, Guoqiang, B. Eddy Patuwo, and Michael Y. Hu. "Forecasting with artificial neural networks:: The state of the art." International journal of forecasting 14.1 (1998): 35-62.

8. Adhikari, Ratnadip, and R. K. Agrawal. "An introductory study on time series modeling and forecasting." arXiv preprint arXiv:1302.6613 (2013).

9.Ayodele Ariyo Adebiyi, Aderemi Oluyinka Adewumi, and Charles Korede Ayo, "Comparison of ARIMA and Artificial Neural Networks Models for Stock Price Prediction," Journal of Applied Mathematics, vol. 2014, Article ID 614342, 7 pages, 2014. doi:10.1155/2014/614342

10.Pascanu, Razvan, Tomas Mikolov, and Yoshua Bengio. "On the difficulty of training recurrent neural networks." International Conference on Machine Learning. 2013.

11. Lipton, Zachary C., John Berkowitz, and Charles Elkan. "A critical review of recurrent neural networks for sequence learning." arXiv preprint arXiv:1506.00019 (2015).

12.Vapnik, Vladimir. The nature of statistical learning theory. Springer science business media, 2013.

13.Greff, Klaus, et al. "LSTM: A search space odyssey." IEEE transactions on neural networks and learning systems (2017).

14.C. E. Rasmussen C. K. I. Williams, Gaussian Processes for Machine Learning, the MIT Press, 2006, ISBN 026218253X.c© 2006 Massachusetts Institute of Technology.www.GaussianProcess.org/gpml

15.Wu, Zhaohua, and Norden E. Huang. "Ensemble empirical mode decomposition: a noise-assisted data analysis method." Advances in adaptive data analysis 1.01 (2009): 1-41.

16.Kononenko, Igor. "Bayesian neural networks." Biological Cybernetics 61.5 (1989): 361-370.