Using Autoregressive Modelling and Machine Learning for Stock Market Prediction and Trading: ICICT 2018, London

Article *in* Advances in Intelligent Systems and Computing · February 2018

DOI: 10.1007/978-981-13-1165-9_70

1 author:

Phillip Hushani
University of Bedfordshire

Using Autoregressive Modelling and Machine Learning for Stock Market Prediction and Trading

Phillip Hushani

School of Computer Science and Technology, University of Bedfordshire, UK
phillip.hushani@study.beds.ac.uk

Abstract. Investors raise profit from stock market by maximising gains and minimising loses. The profit is difficult to raise because of the volatile nature of stock market prices. Predictive modelling allows investors to make informed decisions. In this paper we compare four forecasting models: Autoregressive Integrated Moving Average (ARIMA), Vector Auto-Regression (VAR), Long Short-Term Memory (LSTM), and Nonlinear Autoregressive Exogenous (NARX). The results of predictive modelling are analysed and compared in terms of prediction accuracy. The research aims to develop a new profitable trading strategy. Our findings are: (i) the NARX model has provided accurate short-term predictions but failed long forecasts, and (ii) the VAR model can form a good trend line required for trading. Thus the profitable trading strategy can combine the Machine Learning predictive modelling and Technical Analysis.

Keywords: Machine Learning, VAR, ARIMA, LSTM, NARX, Trading

1 Introduction

Many factors can influence the stocks, making the prices difficult to predict because of fluctuations. To raise profit from the stock market, investors make informed decisions. An efficient trading strategy is based on the information about the market and also dependent on the ability of investors to predict the stock prices. A simple strategy of making the profit is to sell the shares when the prices go up, and to buy when the prices go down [9, 6].

Stock price prediction models are based on past and present data. The predictions are made under the assumption that a future price is a result of historical patterns which appear in the future. The investors believe that the patterns of interest can be learnt from the historical data, and then can be used for predicting the stock prises and making informed decisions. The factors which influence the markets make this task difficult. Machine Learning and Artificial Neural Networks (ANN) have provided efficient solutions to the prediction problems [2]. Machine Learning methods have been used for solving related problems, as described in [16, 19, 15]. The efficient solutions have been developed by using evolving neural networks, as described in [17, 10].

This paper describes a new approach to stock price prediction by combining Machine Learning and conventional models of time series prediction. This approach will assist investors and traders with making informed decisions by using more accurate prediction models. The proposed approach will also help identify a better trading strategy.

The following Section 2 describes the conventional time series prediction models and models which are based on Machine Learning. Sections 3 and 4 describe the data and the results which were achieved on these data by using the predictive models within the proposed approach. Section 5 draws the main conclusion.

2 Method

Multiple Linear Regression (MLR) models assume that a dependent variable (or model output) is a function of independent variables (or input variables) [3, 14], that can be written as: $\hat{y}(t) = x_1(t)\beta_1 + \cdots + x_m(t)\beta_m + e(t)$, where $y(t)$ is the output variable, $x_1(t), \ldots, x_m(t)$ are m independent variables, $e(t)$ is the noise component, and β are the regression coefficients. Here the coefficients β are adjusted on the given observations so as to minimise the prediction error.

Autoregressive Integrated Moving Average (ARIMA) models combine Autoregressive Models (AR) and integrated Moving Averages (MA). Following [7], an ARIMA model $f(c, d, e)$ can have the following parameters: c is the number of autoregressive terms, d is the non-seasonal differences needed, and e is the number of lagged forecast errors in the prediction. The model is thus written as: $\hat{y}(t) = \mu + \beta_1 y(t-1) + \cdots + \beta_p y(t-p) - \theta_1 e(t-1) - \cdots - \theta_q e(t-q)$, where $\hat{y}(t)$ is the estimated outcome, μ is the constant, $\beta y(t)$ are p autoregressive terms (or lagged values of y), and $\theta e(t)$ are the q moving average terms.

Multivariate Vector Autoregressive (VAR) models [9] make the predictions as follows: $\hat{y}(t) = \beta_0 + \sum \beta_i y(t-i) + e(t)$, where $\hat{y}(t)$ is the forecast at time t, β are the model coefficients, and $e(t)$ are the noise component. A case study of autoregressive predictive modelling, e.g., is described in [12, 13].

Long Short-Term Memory (LSTM) models are based on Recurrent Neural Networks (RNN), as described in [4]. The LSTM model includes three gates of the state and a unit that remembers previous values. A gate is described by the *sigmoid* function, and the *tanh* function is used at the output layer for prediction [5]. The LSTM networks can be efficiently trained by using evolutionary strategies such as described in [18, 16, 15].

Nonlinear Autoregressive exogenous (NARX) models are based on the RNNs, as described in [1]. The model parameters can be adjusted by gradient descent optimisation which provides efficient solutions [11]. The output $\hat{y}(t)$ is dependent on the delayed values of $y(t)$ and the input variables u, also known as exogenous inputs. The output of the NARX model f can be written as: $\hat{y}(t) = f(y(t-1), \ldots, y(t-n_y), u(t-1), \ldots, u(t-n_u))$.

3 Data

In the experiments we use a data set represented by the following variables:

1. NASDAQ Closing price (Price)
2. NASDAQ Volume (Volume)
3. Moving Average Convergence/Divergence (MACD)
4. Relative Strength Index (RSI)
5. NASDAQ Volatility Index (VXN)
6. ISE Sentiment Index (ISE).

Variables VXN and ISE are the investor sentiments indices which reflect the investor behaviour in response to the price movements. For example, in 2009 when the prices fell, the indexes VXN and ISE went up, reflecting the investor fear. Variables MACD and the RSI are part of the Technical Analysis.

In our study the data are represented by the closing prices between January 2007 and 31st December 2016. In total, the data include 2,519 trading days. Following [8], seasonal fluctuations were removed from the data in order to improve the accuracy of prediction models. The data were also normalised.

The prediction models will be trained on 70% of the data and tested on the remaining part, as described in the next section.

4 Experiments

This section describes the results obtained with the prediction models described in Section 2. The ARIMA and the VAR models make predictions using the NASDAQ closing price. The ANN models were trained on the data described in Section 3. The parameters of the models are described in Tables 1, 2 and 3.

Table 1. Parameters of the VAR model

Parameters	Value	MSE	t-statistic	p-value
Constant	0.0003	0.001	0.251	0.801
AR1	0.974	0.021	46.294	0.001
AR2	-0.004	0.029	-0.139	0.889
AR3	0.013	0.029	0.4528	0.651
AR4	0.017	0.021	0.8170	0.414

Fig. 1 shows the predicted prices, the VARM yellow and NARX red line. We can see that the predictions are close to the actual prices, and that the ARIMA model has the largest MSE.

Fig. 2 shows the MSE over three years. The models make predictions with a small MSE in the first months. It is interesting to observe that during the remaining period the ARIMA model provides more accurate predictions.

Table 2. Parameters of the ARIMA model

Parameter	Value	MSE	t-statistic
Intercept	-0.002	0.047	-0.03
AR1	0.177	0.020	8.66
β	1	0.001	669.25
Variance	0.287	0.009	32.07

Table 3. Parameters of the LSTM and NARX models

Parameter	LSTM	NARX
Training (Days)	2154	2154
Estimation (Days)	365	365
No. of Input Variables	6	6
Number of Delays	2	2
Hidden Layers	5	5
Performance	MSE	MSE

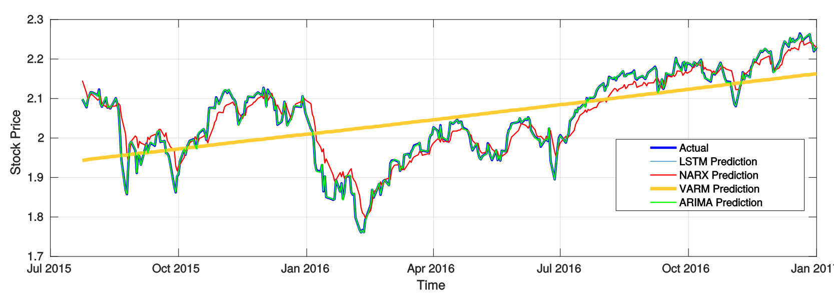

Fig. 1. NASDAQ Index: Actual Price vs Model Forecasts.

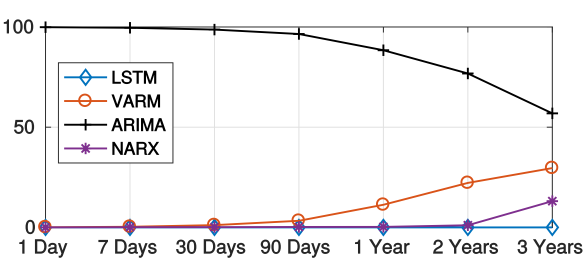

Fig. 2. MSE comparison over 3 years.

The ARIMA predictions shown in Fig. 3 seem to hold over the three-year period. This is consistent with the results shown in Fig. 2. The accuracy of the NARX predictions are different on the training and test data sets.

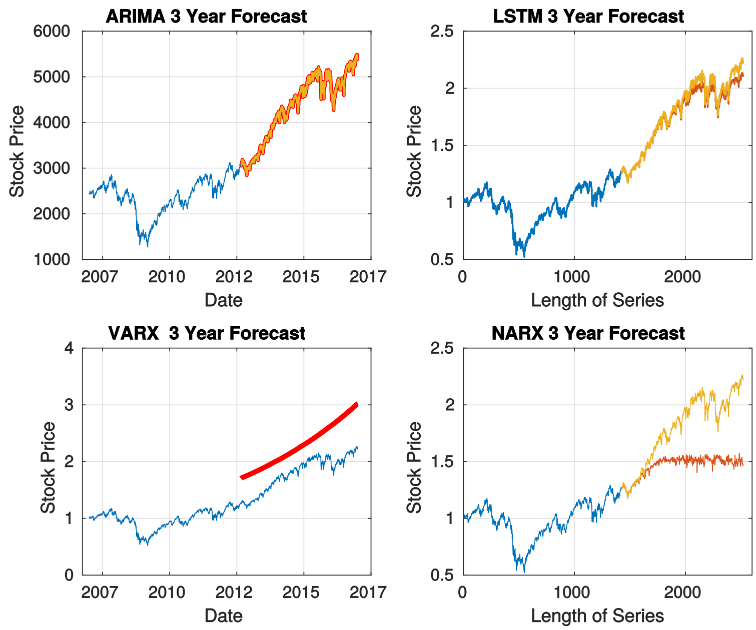

Fig. 3. Forecasts over 3 years.

The EGARCH model accurately predicts the returns and estimates the conditional variances. Fig. 4 shows the actual conditional variances compared against the predicted outputs. The actual prices are close to the ARIMA and LSTM predictions and observed as one line. However, the ARIMA blue line can be seen above the red line in some periods. The NARX dotted line is significantly lower than the actual values. The VARM is a straight line at the bottom of the figure. This figure shows that the LSTM model provides the most accurate predictions.

Based on these experiments, a medium-term trading strategy has been developed. The analysis shows that the LSTM is the most accurate model. The VARM model is able to predict the trend more accurately than the other models. These two models were used for designing a trading strategy which employs the exponential moving average over 30 and 90 days, which is described below.

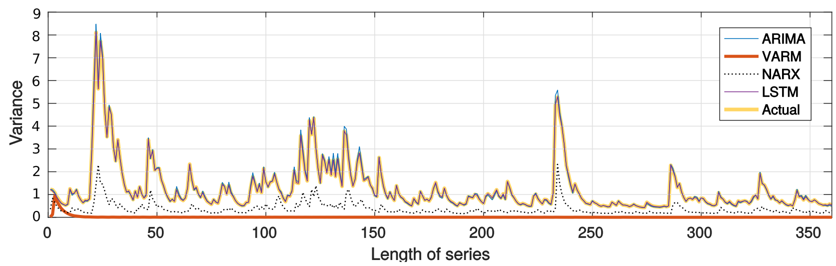

Fig. 4. Conditional Variance.

1. BUY, when the Price Line crosses the exp30 from below
2. SELL, when the Price Line crosses the exp30 from above
3. BUY, when the exp30 crosses the exp90 from below.
4. SELL, when the expo30 crosses the exp90 from above
5. Trade with trend
6. Trade only if the LSTM forecast agrees with the actual price.

Table 4. Trading signals

LSTM	VARM	Exp30	Exp90	Signal
2.0455	2.2061	2.0095	1.985	BUY
2.0194	2.2067	2.0102	1.9855	BUY
2.0012	2.2074	2.0096	1.9855	SELL
2.0008	2.208	2.009	1.9859	SELL
1.9959	2.2087	2.0082	1.9864	SELL
1.9972	2.2093	2.0075	1.9877	SELL
1.9779	2.2099	2.0056	1.9892	SELL
1.9945	2.2106	2.0048	1.991	SELL
1.9969	2.2112	2.0043	1.9926	SELL
1.9922	2.2119	2.0035	1.9945	SELL
2.0242	2.2125	2.0049	1.9962	BUY
1.9425	2.2131	2.0009	1.9982	SELL

Fig. 5 shows the tradings signals – traders buy with the trend using the above rules.

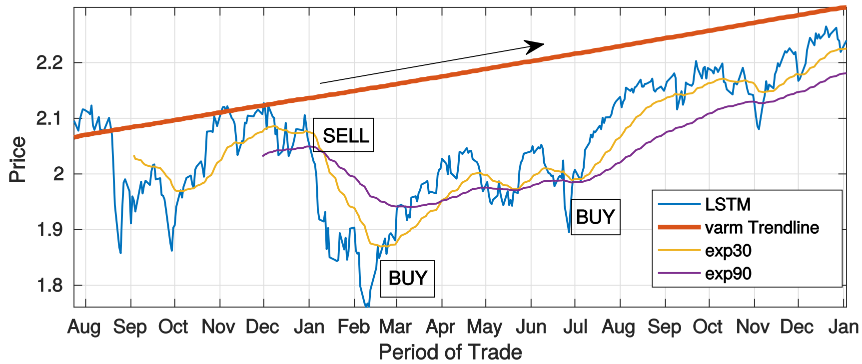

Fig. 5. NASDAQ index medium term trading strategy.

5 Conclusions

In this study we aimed to compare conventional and neural-network models in terms of accuracy of predicting time series. We have attempted to find a model which performs better on the NASDAQ Composite Index data set. These data represent the daily closing prices, and thus our study is limited to the case of daily predictions.

In the experiments, the high accuracy was observed for all models during the first three months. The predictions of the LSTM and the ARIMA models were most accurate during the first year.

The main findings are as follows: (i) the NARX model has provided accurate short-term predictions but failed long forecasts, and (ii) VARM model forms a good trend line. Thus we conclude that traders can benefit from a strategy which combines the Machine Learning predictive modelling and Technical Analysis.

References

1. Billings, S.A.: Nonlinear System Identification: NARMAX Methods in the Time, Frequency, and Spatio-Temporal Domains. John Wiley & Sons (2013)
2. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer (2007)
3. Fabozzi, F.J., Focardi, S.M., Rachev, S.T., Arshanapalli, B.G.: Multiple Linear Regression, pp. 41–80. John Wiley & Sons (2014). DOI 10.1002/9781118856406.ch3
4. Gers, F.A., Schmidhuber, J.A., Cummins, F.A.: Learning to forget: Continual prediction with lstm. Neural Comput. **12**(10), 2451–2471 (2000). DOI 10.1162/089976600300015015
5. Gers, F.A., Schraudolph, N.N., Schmidhuber, J.: Learning precise timing with lstm recurrent networks. J. Mach. Learn. Res. **3**, 115–143 (2003). DOI 10.1162/153244303768966139
6. Kordos, M., Cwiok, A.: A new approach to neural network based stock trading strategy. In: H. Yin, W. Wang, V. Rayward-Smith (eds.) Intelligent Data Engineering and Automated Learning - IDEAL 2011, pp. 429–436 (2011)

7. Makridakis, S., Hibon, M.: Arma models and the boxjenkins methodology. Journal of Forecasting **16**(3), 147–163 (1997)
8. Nelson, M., Hill, T., Remus, W., O'Connor, M.: Time series forecasting using neural networks: should the data be deseasonalized first? Journal of Forecasting **18**(5), 359–367 (1999)
9. Neusser, K.: Time Series Econometrics. Springer, Cham (2016). DOI 10.1007/978-3-319-32862-1
10. Nyah, N., Jakaite, L., Schetinin, V., Sant, P., Aggoun, A.: Evolving polynomial neural networks for detecting abnormal patterns. In: 2016 IEEE 8th International Conference on Intelligent Systems (IS), pp. 74–80 (2016). DOI 10.1109/IS.2016.7737403
11. Sahoo, H., Dash, P., Rath, N.: Narx model based nonlinear dynamic system identification using low complexity neural networks and robust h filter. Applied Soft Computing **13**(7), 3324 – 3334 (2013). DOI 10.1016/j.asoc.2013.02.007
12. Schetinin, V., Jakaite, L.: Classification of newborn EEG maturity with Bayesian averaging over decision trees. Expert Systems with Applications **39**(10), 9340–9347 (2012). DOI https://doi.org/10.1016/j.eswa.2012.02.184. URL http://www.sciencedirect.com/science/article/pii/S0957417412004538
13. Schetinin, V., Jakaite, L.: Extraction of features from sleep EEG for Bayesian assessment of brain development. PLOS ONE **12**(3), 1–13 (2017). DOI 10.1371/journal.pone.0174027. URL https://doi.org/10.1371/journal.pone.0174027
14. Schetinin, V., Jakaite, L., Nyah, N., Novakovic, D., Krzanowski, W.: Feature extraction with GMDH-type neural networks for EEG-based person identification. International Journal of Neural Systems (2018). DOI 10.1142/S0129065717500642
15. Schetinin, V., Schult, J.: A neural-network technique to learn concepts from electroencephalograms. Theory in Biosciences **124**(1), 41–53 (2005). DOI 10.1016/j.thbio.2005.05.004
16. Schetinin, V., Schult, J.: Learning polynomial networks for classification of clinical electroencephalograms. Soft Computing **10**(4), 397–403 (2006). DOI 10.1007/s00500-005-0499-3
17. Schetinin, V., Schult, J., Scheidt, B., Kuriakin, V.: Learning multi-class neural-network models from electroencephalograms. In: V. Palade, R.J. Howlett, L. Jain (eds.) Knowledge-Based Intelligent Information and Engineering Systems, pp. 155–162. Springer Berlin Heidelberg, Berlin, Heidelberg (2003)
18. Schmidhuber, J., Wierstra, D., Gagliolo, M., Gomez, F.: Training recurrent networks by evolino. Neural Computation **19**(3), 757–779 (2007). DOI 10.1162/neco.2007.19.3.757
19. Zharkova, V.V., Schetinin, V.: A neural-network technique for recognition of filaments in solar images. In: 7^{th} International Conference Knowledge-Based Intelligent Information and Engineering Systems KES 2003, Oxford, pp. 148–154. Springer Berlin Heidelberg (2003). DOI 10.1007/978-3-540-45224-9_22