An Application of Machine Learning for Economic Trend Prediction

Akhil Ganesan[1] and Prabhas Adivi[1]

[1] *Centreville High School Class of 2023*

Abstract

In this paper we present the results of testing a variety of machine learning architectures (statistical analysis, support vector regression, & a neural network) in their ability to predict economic trends. The economic trends explored were stock prices in the market; in particular, we tested the artificial intelligence's ability to predict Microsoft (MSFT), Apple (AAPL), & Netflix (NFLX) share prices over a 5 year period. The results support the hypothesis claiming the neural network would be the most accurate followed by the statistical analysis & support vector regression respectively. It also concluded that the MSFT dataset was the most predictable, followed by AAPL & NFLX respectively (which coorelated to their broader range of prices). Overall, this application provided a better insight of how machine learning can be applied into economics; this application has much potential when scaled to predict economic trends other than stocks, future recession/inflation periods, etc.

Keywords: artificial intelligence, machine learning, economics, stocks

1 Introduction

Advanced computer systems - including artificial intelligence (AI) - are predicted to lead the next generation of technological breakthroughs. The neural network, the hallmark of modern machine learning architectures, is of particular interest in how its operation varies from prior statistical methods of analytics. And although machine learning has existed for decades, their application is still being explored in nearly every possible aspect of the complex world. To test the capability of this technology, the challenge provided was stock price prediction as it was a small-scale model proxy for economic trend prediction in general. Furthermore, the stock market testing environment is rich in data availability and expert predictions aren't consistently accurate. This paper presents a comparative analysis of the stocks price prediction accuracy of 3 different machine learning architectures: a neural network, a statistical (regression) analysis (the control baseline for accuracy), & a support vector regression (repurposed from classification problems, it serves as a control with expected poor results). In comparing the statistical analysis & neural network, the neural is typically more attentive to small-scale variations while the statistical regression analysis is strong in recognizing the general trend with poorer small-scale accuracy. While the statistical analysis can't be altered to fit more small-scale trends without changing the model, by selectively fitting the neural network such that it isn't underfitted (only accounting for larger-scale trends) or overfitted (only accounting for smaller-scale trends), it could become a template model for accurate economic prediction.

2 Materials & Methods

In this study, the stocks chosen for prediction were Apple (AAPL), Microsoft (MSFT), & Netflix (NFLX). The daily data for these stocks' prices from 2013 to 2018 was obtained from Yahoo Finance. The open price data was chosen for prediction. The table below shows the division of the data into training (used to tune the AI) & testing (used to assess the AI's accuracy) ranges (which are typically split by 80% & 20% respectively). The figures that follow show the graphed stock data.

Table 1: Stock Dataset Structure

Dataset	**Full Set**	**Training**	**Testing**
AAPL	(0,1259)	(0,1000)	(1000,1259)
MSFT	(0,1259)	(0,1000)	(1000,1259)
NFLX	(0,1259)	(0,1000)	(1000,1259)

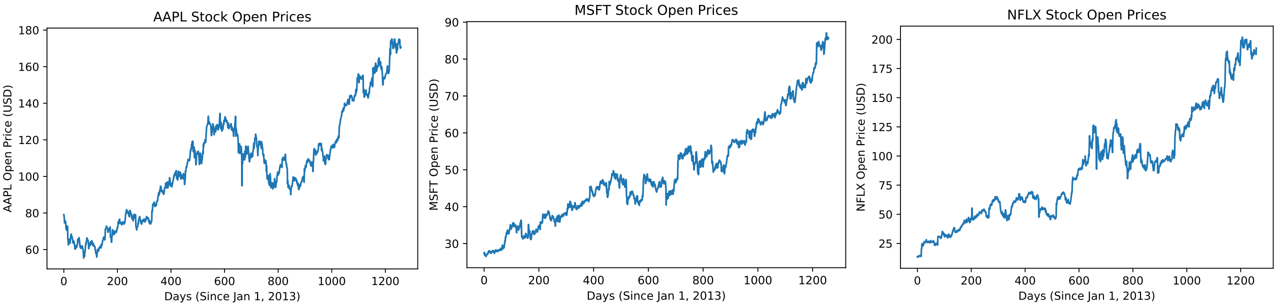

Figure 1: Stock Dataset Structure

After obtaining the stock data, the AIs were programmed using python (using libraries such as sklearn & TensorFlow). All of the code can be found on the project github repository. For training & testing, predictions & stock data were input into a mean absolute error cost function represented by the below equation (where n is the total number of days, i represents a specific day in the iteration, $C(x)$ is the function returning the open stock price for day x, & $\mathrm{AI}(x)$ is the function returning the predicted stock price for day x).

$$\mathrm{MAE}(C, \mathrm{AI}) = \frac{1}{n} \sum_{i=0}^{n} |C(i) - \mathrm{AI}(i)| \tag{1}$$

From here, 3 trials of training & testing for each AI type & stock were conducted; the best performance was recorded. The variations are due to the randomized starting tuning of the AIs, which is especially prevalent in the neural networks.

3 Results

The results of the experiment supported the hypothesis in its assertion of the neural network (NN) having the highest accuracy, then the statistical analysis (SA), and finally, the support vector regression (SVR). The table & graph below includes the final mean absolute errors (in USD) of the machines on the stocks of Apple (AAPL), Microsoft (MSFT), & Netflix (NFLX). The graphed performance of each AI on each stock is also included.

Table 2: Machine Testing Results (Error Table)

Datasets	**SA**	**SVR**	**NN**	**Average**
AAPL	19.60	53.3	20.49	31.13
MSFT	9.05	27.49	2.29	12.94
NFLX	33.14	92.65	8.81	44.87
Average	20.60	57.81	10.53	29.65

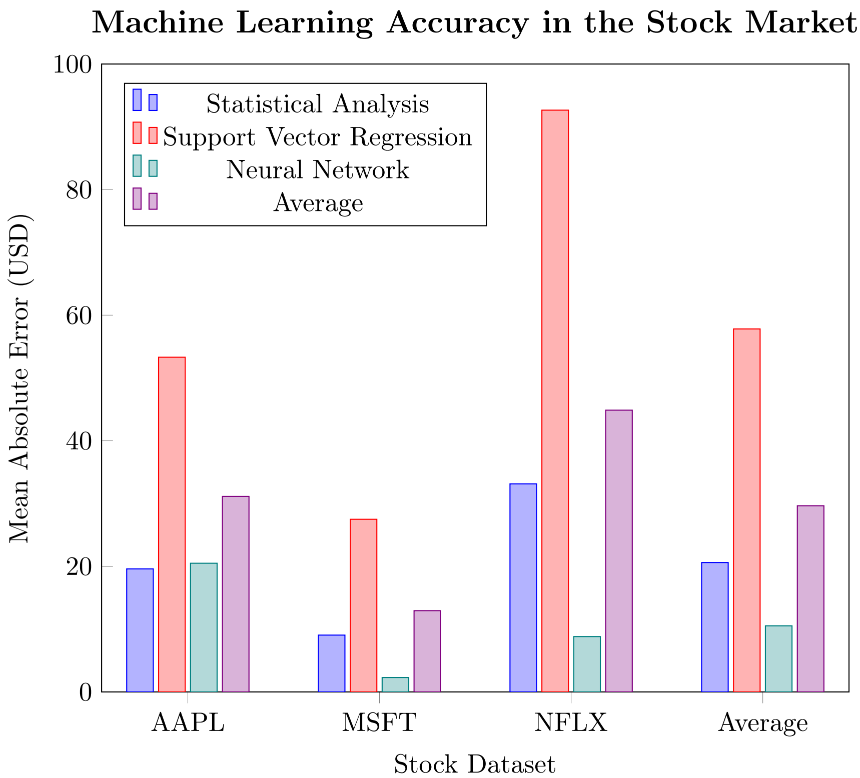

Figure 2: Machine Testing Results

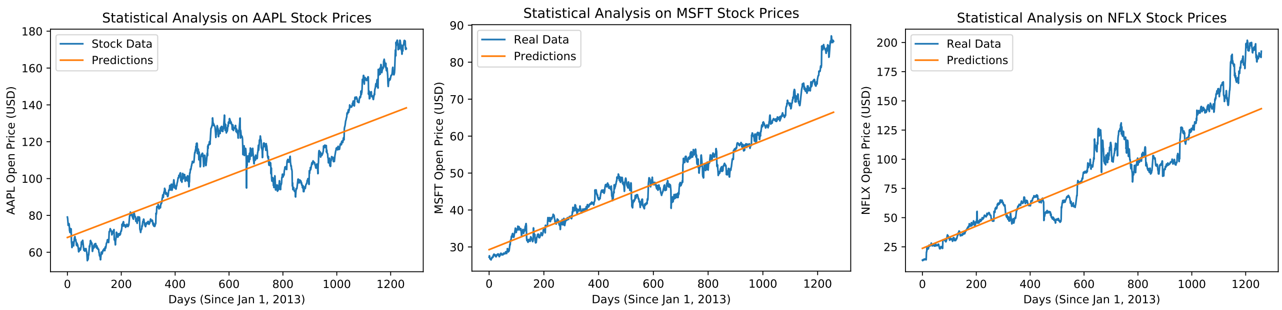

Figure 3: Statistical Analysis Stock Predictions

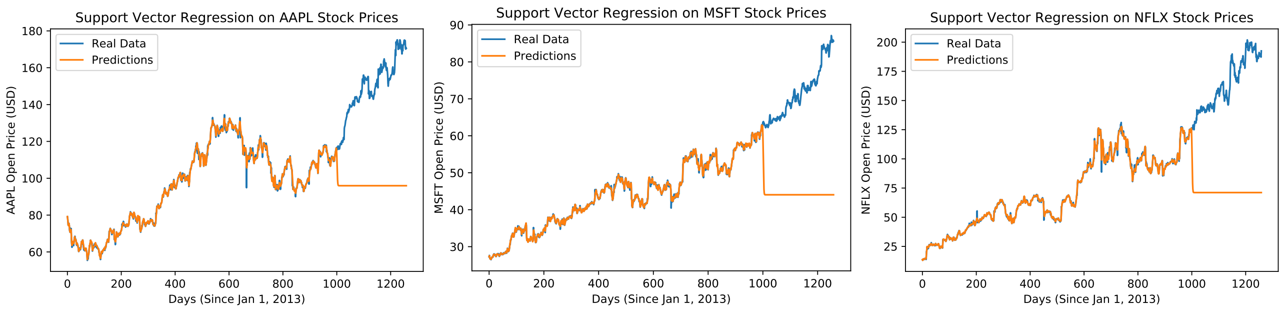

Figure 4: Support Vector Regression Stock Predictions

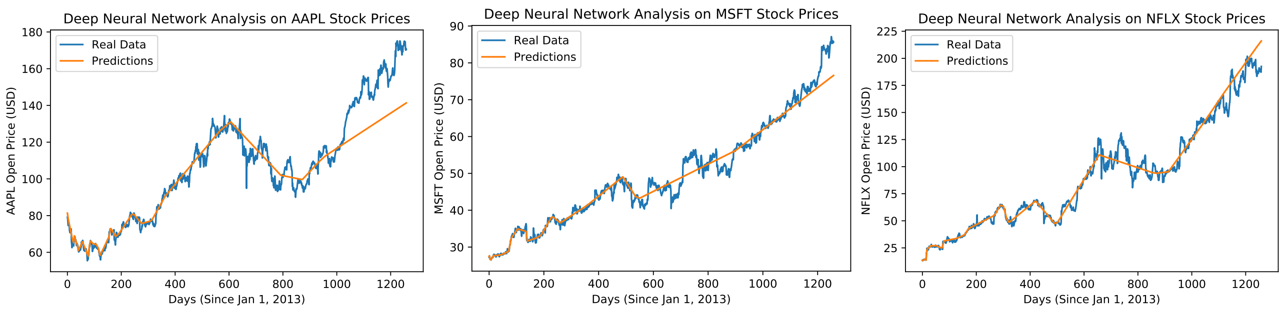

Figure 5: Neural Network Stock Predictions

4 Discussion

Upon resolving the data from the experiment, a couple conclusions arose. First and foremost, the neural network was the most accurate in predictions, then the statistical analysis, and finally the support vector regression (as previously predicted). The neural network was most likely effective due to its far higher complexity in comparison to the other machines (in which it could recognize smaller trends while still being able to interpret the big picture through its weighted sum infrastructure). For this experiment, the neural network was structured with 2 hidden layers of 64 nodes each. Furthermore, to accommodate for the random preset tuning, 10000 epochs of training were performed. Regardless, the statistical analysis was probably the next most effective predictor because it summarized the data to the simplest form possible (a linear regression) while taking into account the trends of a continuous variable (the output could lie outside the range of outputs of the training data). Finally, the support vector regression was most likely last due to the fact that it summarized the data into a line too generally (in the sense that it wasn't expecting a continuous variable, rather, one that came previously, resulting in its plateauing at a constant value within the range of training data price values). It also was a classical example of overfitting data by its close accuracy to the training set but poor prediction of the testing set. Another important trend to notice was the difference in data set toughness. As shown by the average column, Microsoft data tended to be the most predictable, then Apple (due to its increased curvature compared to the others, it requires a higher degree model to predict accurately), then Netflix (due to its higher range of price values compared to the other stocks resembling a scaling factor of the error function when compared to the other stocks). Almost all of the machines followed these trends except for 2 instances: the statistical analysis outperformed the neural network on the Apple dataset (the model slightly overfit resulting in a lower sloped prediction curve) & the neural network performed better on the Netflix dataset than the Apple dataset (again for the aforementioned reason in addition to the fact that the NFLX dataset in more linear in shape compared to the Apple dataset, allowing better prediction with a lower-degree model). While the experiment yielded useful results and AIs that could predict stocks quite accurately, the machines used in this experiment are far from ready for practical usage due to the multitude of sources of errors possible. For example, one major confirmed source or error was the fact that all of the datasets were positively trending & generally not too volatile (in the sense that overall prices were rising & the graphs were nearly continuously). However, this isn't realistic and may not predict as well on a smaller scale of negatively trending, volatile markets. Another error lies in the use of support vector regression. The main problem with this is that it wasn't expecting a continuous value (one of which is out of the train range); instead, it expected a value provided and averaged the training outputs to achieve that. Furthermore, the specifications of the neural network (including its structure, scaling functions, etc.) could be further optimized to yield better results in followup experiments. And on top of all of this, the only input information being prior stock data also serves as a barrier for maximizing performance; if more data like global economic conditions were also included for analysis by each machine, it would result in significantly improved results.

5 Conclusions

In conclusion, this project yielded not only AIs that could predict certain stocks to a high degree of accuracy, but it also served as an example for implementing machine learning in the field of economics. Further optimization of neural networks structurally to better process more data than just prior prices (such as market conditions, corporate actions, local news, etc.) and to predict broader economic trends (such as diversified market prices/funds) would be the next step developing an accurate economic predictor. The use of such a predictor would help predict deleterious economic trends such as recessions, inflations, unemployment, stagnation, etc. It could also be used by the private sector for setting reasonable targets for production, retail, annual/quarterly earning, etc. Overall, the potential an economic predictor (based on a further optimized deep neural network framework presented in this paper) is near limitless in its uses to impact the world for the better.

References

Linear regression. URL http://www.stat.yale.edu/Courses/1997-98/101/linreg.htm.

Artificial intelligence. 2019. URL https://link.gale.com/apps/doc/VOJSVI051385296/SUIC?u=fairfaxcps&sid=SUIC&xid=7cc5c749.

Thierry Denoeux. Logistic regression, neural networks and dempster–shafer theory: A new perspective, Mar 2019. URL https://www.sciencedirect.com/science/article/abs/pii/S0950705119301558.

P Ghosh. How is machine learning used in the stock market? pc quest, 2019. URL http://search.ebscohost.com/login.aspx?direct=true&db=f6h&AN=134620092&site=ehost-live.

Arielle O'Shea and Kevin Voigt. Stock market basics: What beginner investors should know, Jun 2017. URL https://www.nerdwallet.com/blog/investing/stock-market-basics-everything-beginner-investors-know/.