IT1244 Stock Market Predictions

By: Ng Simin (A0240646H), Woo Kean Jin Brandon (A0233835A), Muzi Chen (A0240456J), Jonas Lim (A0223850L), Tan Le Jun (A0199755E)

Abstract

The application of artificial intelligence and machine learning can help investors make better investing decisions. We propose a practical model that investors can train at the end of a week using the opening, minimum, maximum, and closing prices of each of the weekdays of that week. The model will be able to predict whether the price of a particular stock will increase by next Friday. This can help investors decide whether to continue investing in a particular stock. Testing results show that our optimized Logistic Regression model was able to achieve 20.92% of True Capital predicted, which is quite significant.

1. Introduction to Artificial Intelligence in stock market price prediction

In this project, our team examined the effectiveness of machine learning and artificial intelligence to make informed stock picks for an active investment approach.. The goal of our report is to paint an unbiased picture of the possibility for retail investors, without powerful computers and sophisticated algorithms, to make financial gain. This project utilizes the data preparation and the types of machine learning models taught in this module to investigate the possible avenues one can use to make more accurate stock predictions.

Previous research done has been extensive on machine learning models in making stock predictions. A research paper done by the Journey of Physics reviewed studies done on primarily 4 different groups of models; Traditional Machine Learning Algorithm, Deep Learning, Time-Series and Graph-Based. The relevant findings from the study shows that traditional machine learning models are generally more accurate even with high dimension dataset and as such appears to be the most suitable for individuals with relatively limited computing power to train. Our study serves to tackle the problem from the perspective that the "investor" uses the week's data to train at the end of the week when the market is close which prevents any price fluctuations to affect the model ***[1].***

2. Methodology

Our objective is to build a model that is able to best predict the decision to whether one should buy a stock at the end of the week, comparing the Friday's closing price to the following Friday's closing price. The model will be trained on current week's opening, max, min and closing prices.

2.2 Variable Selection

In our preliminary testings, we explored all 22 attributes of the original dataset, including running a natural language processing model and lasso regression ***[2]*** on the business summaries of the different companies. We conclude that the most significant attributes are the opening, minimum, maximum, and closing prices.

2.3 Metric

In this project, we will be testing our model based on each test data set. Each model will start off with $1000, and the model will decide to buy one share if the predicted closing price next Friday (response variable, Y) is higher than the current week's Friday closing price.

We then evaluate the effectiveness of the model by "Percentage of True Capital Predicted", the proportion of "Final Predicted Capital" to "Final True Capital". We define the "Final Predicted Capital" to be the capital our model will have based on the decision made by the predicted response. We define "Final True Capital" to be the highest attainable capital assuming the model is omniscient about the trend of the next week's closing price. The higher the percentage, the better our model is at predicting the test data.

2.4 Data Cleaning

As there were some missing dates in the given dataset, we added rows for those dates by referencing data from the immediate previous day.

After some cleaning and rearrangements, the output data consists of 203934 rows and 26 columns. Each row represents a week's worth of data for a particular week of a particular stock. The 26 columns include the opening, minimum, maximum, and closing prices for each of the weekdays of the week. The last column shows Y, which is 1 if the following Friday's closing price is higher than the current Friday's closing price and 0 if it is not.

The data was then split into training and test data, 80% and 20% respectively. For our neural network models, the test data was 40% and further split into test and validation data, 50% and 50% each. All the selected attribute data (X data) are then standardized to negate the effects of varying scales.

2.5 Prediction Models

We trained four different models, Neural Network ***[3],*** Logistic Regression ***[4]***, Stochastic Gradient Descent Classier ***[5]*** and K Nearest Neighbor Classifier ***[6]***. The results of the execution of these models and the percentage of True Capital predicted with each model are summarized in the following table:

Prediction Models	**Prediction Accuracy**	**Final Predicted Capital**	**Final True Capital**	**% of True Capital Predicted**
Neural Network (Figure 1)	3.33(MSE)	9479	58247	16.27%
Logistic Regression (Figure 2)	54.13%	10546	58201	18.12%
SGD Classifier (Figure 3)	53.98%	9083	58201	15.61%
KNN Classifier (Figure 4)	53.88%	9217	58201	15.84%

Table 1: Summary of Models Execution Results

2.6 Optimisation of Logistics Regression Model

As our results showed that the Logistics Regression Model would yield the highest percentage of True Capital predicted, we decided to expand on this model.

To achieve a higher accuracy, we tried varying the type of linear regression solver and also the proportion of training and testing data.

Logistic Regression Solver used	**Prediction Accuracy**	**Final Predicted Capital**	**Final True Capital**	**% of True Capital Predicted**
libklinear	54.13%	10546	58201	18.12%
lbfgs	54.14%	10591	58201	18.20%
newton-cg	54.13%	10546	58201	18.12%

Table 2: Results Using Different Logistic Regression Solver

The three solvers tested are liblinear, lbfgs, and newton-cg. lbfgs performed slightly better than the other two solvers.

Ratio of training to testing data	**Prediction Accuracy**	**Final Predicted Capital**	**Final True Capital**	**% of True Capital Predicted**
7:3	54.14%	15802	88280	17.90%
8:2	54.14%	10591	58201	18.20%
9:1	54.38%	6248	29862	20.92%

Table 3: Results Using Different Train-Test Ratio

The three ratios tested are 7:3, 8:2, and 9:1. We observe that the best ratio is 9:1.

3. Conclusion and Discussion

3.1 Conclusion of Finding

Out of the models that we trained, the Logistic Regression model performed the best. Upon further optimization, we conclude that using lbfgs as the logistic regression solver and a training to testing data ratio of 9:1.

Using this model has allowed us to obtain 20.92% of the True Capital predicted, which is quite significant.

3.2 Limitations and Future Outlook

In the real world, unexpected events that could lead to market disruptions might occur, an example would be the recent COVID-19 pandemic. At this stage, a limitation to our prediction models is that it could neither account for the influence these events had on the training data nor incorporate the impact of possible future disruptive events on the stock market into the prediction results.

Moreover, due to the nature of binary classifier models and the relatively low prediction accuracy indicated (not much higher than random, 50%), there could be a significant number of false positives and false negatives in these model's predictions. This adversely affected the accuracy of our model's decisions and consequently the reliability of our results.

In future iterations, we envision training our model with bigger data sets, for example a month, a year or even live predictions, to gain a better insight of the trends and patterns that each stock market has. This will provide better and more holistic guidelines for prospective retail investors seeking to venture into the stock market.

Appendix

1. Figures and Diagrams

Figure 1: Neural Network

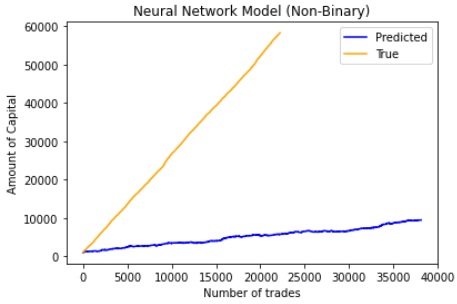


Figure 2: Logistic Regression

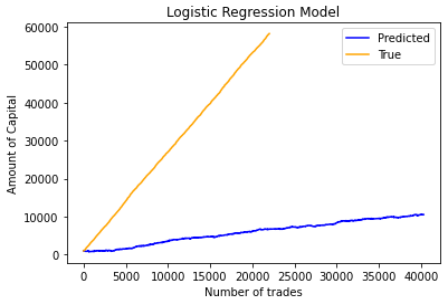


Figure 3: SGD Classifier

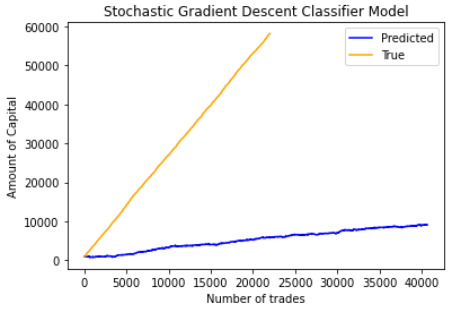
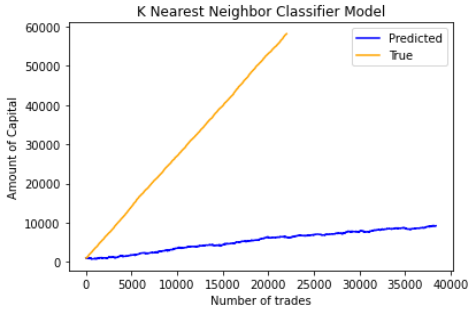


Figure 4: KNN Classifier



2. References

[1] Soni, P., Tewari, Y., & Krishnan, D. (n.d.). Machine learning approaches in stock price prediction: A ... - iopscience. Retrieved October 31, 2022, from https://iopscience.iop.org/article/10.1088/1742-6596/2161/1/012065

[2] Carlmcbrideellis. (2022, September 21). Feature importance using the lasso. Kaggle. Retrieved October 31, 2022, from https://www.kaggle.com/code/carlmcbrideellis/feature-importance-using-the-lasso

[3] Zhailat. (2020, August 19). Introduction-to-machine-learning-python/EX02-svm.ipynb at master · zhailat/introduction-to-machine-learning-python. GitHub. Retrieved October 31, 2022, from https://github.com/zhailat/Introduction-to-machine-learning-Python/blob/master/Part%2008%20-%20Constructing%20a%20Binary%20Classifier%20Using%20SVM%20with%20Python/Ex02-SVM.ipynb

[4] Bonthu, H. (2021, July 11). An introduction to logistic regression. Analytics Vidhya. Retrieved October 31, 2022, from https://www.analyticsvidhya.com/blog/2021/07/an-introduction-to-logistic-regression/#:~:text=Logistic%20Regression%20is%20a%20%E2%80%9CSupervised,used%20for%20Binary%20classification%20problems.

[5] Introduction to SGD classifier - Michael Fuchs Python. MFuchs. (2019, November 11). Retrieved October 31, 2022, from https://michael-fuchs-python.netlify.app/2019/11/11/introduction-to-sgd-classifier/

[6] Raeisi Shahraki, H., Pourahmad, S., & Zare, N. (2017). k important neighbors: A novel approach to binary classification in high dimensional data. BioMed research international. Retrieved October 31, 2022, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5742505/#:~:text=K%20nearest%20neighbors%20(KNN)%20are,classification%20in%20high%20dimensional%20problems.