Research on the Stock Price Forecasting of Netflix Based on Linear Regression, Decision Tree, and Gradient Boosting Models

Xinwen Xu[1, *]

[1] New College, University of Toronto, Toronto, M5S 3J6, Canada

*Corresponding author. Email: fdvsv1067299@mail.wcccd.edu

Abstract. Stock return forecasting has always been a popular research topic in the stock market. This paper adopts three models, including linear regression, decision tree, and gradient boosting approaches, to predict the eighth day's stock return of Netflix stock based on its last seven days' stock return, based on the price data of Netflix stock from 2002 to 2021. Prediction results and model performances are compared with the five-fold cross-validation and Python score method. The results indicates that the linear regression model is the best model for predicting Netflix-type stocks' return on a long-term scale and has no sharp nor abnormal fluctuations. This research result enriches the existed stock return forecasting literature and provides a certain revelation for investors towards predicting stock return growth trends and stock investment values accurately.

Keywords: Stock return forecast; Linear regression; Decision tree; Gradient boost; k-fold cross-validation.

1 Introduction

The stock market is prevalently defined as unstable and hard to predict, which seems reasonable as factors that could affect a stock's return are somehow uncertain. Some doctrines also argue that market returns and developments are not predictable based on past data [1]. A portion of the public believes those factors include the economic conditions of the company issuing the stocks, policy trends, and market participants' attitudes, which mutually influence each other, are hard to trace or monitor, which makes predictions hard to get start with [2]. However, previous research indicates that the stock market's possible movement and the near-future trend are predictable with modelling methods as factors like those rarely affect the monthly and daily stock market return [3].

Numerous researches have indicated that the stock market has certain regularity and volatility. Under different circumstances, the market will tend to show various patterns. Based on the market transaction data of NYSE stock from 1980 to 1984, a study shows that the daily variability of its return in a minute followed a U shape pattern [4]. A recent discovery illustrates that stocks that have been negatively affected by the

V. Escudero et al. (Eds.): BADS 2022, AHCS 11, pp. 1232–1242, 2023.
https://doi.org/10.2991/978-94-6463-102-9_127

COVID-19 epidemic recover in a V or L shape, depending on their financial stress [5]. More general examples also include that the stock market return has turned out to be typically higher at the beginning of a day's stock market and at the end of it based on the data from 1964 to 1989 [6].

All those results are obtained based on different investigations the public have made with the historical data. And that is when machine learning comes into the public. Machine learning methods and techniques enable people to gather and reorganize the data throughout time, investigate its possible internal structure, and even make predictions. Nowadays' research has already proved that machine learning methods could help predict stock return with relatively high accuracy [7]. The properties of different data may vary, but a corresponded strategy for making predictions based on it could always be found or developed.

This paper uses three approaches, Linear Regression, Decision Tree, and Gradient Boosting, to predict the Netflix stock return's growth trend based on the data from 2004 to 2021 collected by Yahoo Finance. Using the time series split approach, the whole data is segmented into a training set and a test one for machine learning models' training and evaluating purposes. This study aims to predict the growth proportion of Netflix stock return on the eighth day using its daily growth proportion in the last seven days as an input source, where the daily growth proportion of the stock return is calculated by

$$\frac{adjusted\ closed\ price - open\ price}{open\ price} \tag{1}$$

Meanwhile, comparing the three different models' effectiveness in predicting the daily growth proportion of the stock return.

The remaining part of this paper is organized follows. Section 2 is a general presentation of three machine learning methods, including linear regression, decision tree, and gradient boost methods. Section 3 is an illustration of the design of the experiment and the used dataset, along with some graphical presentation of the results. Section 4 compares the performances of the three models and discusses the potential factors causing the result to be like that. Section 5 is the overall conclusion of this research, along with some future exploration expectations.

2 Machine learning methods

In order to predict the Netflix sock return growth, this study uses linear regression, decision tree, and gradient boost methods.

2.1 Linear Regression Method

The linear regression method always comes along with a linear assumption. When applying the linear regression method to the data, the assumption that the relationship among the target variables is linear will be automatically imposed on the data. But

sometimes, the relationship may not hold for the used data. Meanwhile, the problem of overfitting occurs a lot with the linear regression model [8].

The concepts of linear regression can be further divided into simple linear regression and multiple linear regression. A simple linear regression is defined as a linear regression with a single independent variable, while the multiple linear regression is the term standing for a linear regression which includes multiple independent variables. Each independent variable within the regression works as a predictor. Usually, Y denotes the dependent variable, and x stands for the independent one. The basic formula to show how linear regression works is,

$$Y_{\{i\}} = \beta_{\{0\}x_{\{i,0\}}} + \beta_{\{1\}x_{\{i,1\}}} + \cdots + \beta_{\{k\}x_{\{i,k\}}} + \varepsilon_{\{i\}} \tag{2}$$

where i is for indexing purposes, and k stands for a random index that is smaller than i but bigger than the previous ones. The β here is the unknown linear regression coefficient. If β equals zero, the dependent variable has no linear relationship to the independent one. ε stands for the error term with an expectation value that equals zero. When discussing multivariate or multidimensional problems along with time series, the model needs to be considered in terms of a vector concept [9].

2.2 Decision Tree Method

The decision tree approach here is for prediction purposes. Based on data collected in the past, the decision tree model can make predictions about future data. A decision tree model always includes multiple nodes and branches. Each node contains one requirement. If the requirement is met, the data proceed to another node at the next level; if not, the data will go to another one. The branches within the model indicate the flow between the nodes. The beginning of the model is a root node, a start point of the model, while the other nodes under it are leaf nodes or could be called leaves. There may be other leaf nodes under a leaf node. And a leaf node with other leaf nodes under it is a parent node of those underneath leaf nodes. The final result of a decision tree model is the node with no other leaf node under it. There could be multiple results within a model. Each represents a possible outcome that could be gotten with a certain scenario.

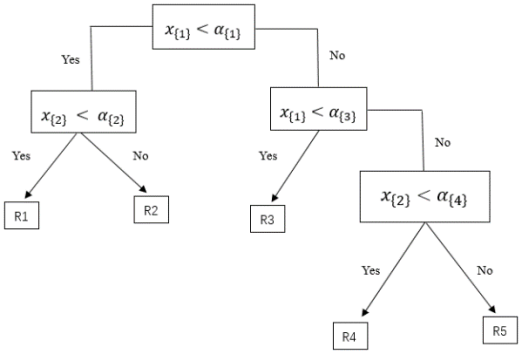

Fig. 1. A simple sample of how a decision tree model works [drawn by the author]

The decision tree model's workflow is logical and clear. And it works well with a large dataset. If the sample size is small, overfitting or overclassifying may occur with the model [10].

A good tree needs to be deep and broad, or the tree may be either low efficient or not very accurate [11]. For improving a decision tree model to let it make better predictions or classification with certain efficient, forward pruning or backward pruning are two possible methods that could be taken into consideration. Forward pruning could remove unnecessary branches before a tree forms completely through chi-square tests or multiple comparisons, while backward pruning can be used after the generation of the tree to enhance the model's accuracy level [12].

2.3 Gradient Boosting Method

The gradient-descent-based gradient boosting process is iterative. Each new training iteration is for improving the results of the previous one. It involves a new model and will generate a new residual. This residual will then be used in the next iteration to train a new model and perform a new round of fitting. The whole process will be repeated until a final model is created, which will be a combination of all the previous models. Based on the research of [13], the pseudo-code could be summarized in Figure 2.

Input: x

1. $F_{\{x\}}^{*} = \arg\min \sum_{i=1}^{n} \varphi(y_{\{i\}}, \beta)$
2. $For\ m = 1\ to\ m = M$:
3. $\quad \widetilde{y_{\{im\}}} = -\left(\frac{\partial \varphi(y_{\{i\}}, F(x_{\{i\}}))}{\partial F(x_{\{i\}})}\right)\ where\ i \in [1, n]\ and\ i \in Z$
4. $\quad \alpha_{\{m\}} = \arg\min \alpha, \rho \sum_{i=1}^{n} (\widetilde{y_{\{im\}}}) - \rho h(x_{\{i\}}; \alpha)^2$
5. $\quad \beta_{\{m\}} = \arg\min \beta \sum_{i=1}^{n} \varphi(y_{\{i\}}, F_{\{m-1\}(x_{\{i\}})} + \beta h(x_{\{i\}}; \alpha_{\{m\}}))$
6. $\quad F_{\{m\}(x)} = F_{\{m-1\}(x)} + (\beta_{\{m\}} h(x_{\{1\}}; \alpha_{\{m\}}))$
7. End
8. End

Fig. 2. Pseudo-code of gradient boosting method [drawn by the author]

The gradient boosting method is a method that organizes simple and weak models to form one complex one with better functioning performance. It fully uses the loss functions, which makes it more stable than the AdaBoost method when extreme or abnormal data occurs [14]. However, it is still likely to show overfitting circumstances. To improve the accuracy of a gradient boosting model, adjusting the parameters used to create the model could be a strategy. Besides that, introducing randomization to each iteration's used data is also an efficient method to enhance the gradient boosting model's accuracy as it could decrease the correlation between each iteration's result [13]. In addition to the two methods illustrated above, another strategy is to use a combination of the gradient boosting method and decision tree method, the gradient boosting regression tree method, which is also a well-known method for making predictions. Although adding too many trees will make the overfitting problem more significant, an adequate amount of short trees could effectively create better results [15].

3 STOCK PRICE return FORECAST: case of NETFLIX

3.1 Data and Variables

The used dataset for this research is from Kaggle. The dataset contains the Netflix stock data from 2002-05-23 to 2021-09-30, with all data collected from Yahoo Finance. For the illustrated period, the data has recorded each day's Netflix stock's open price, closed price, the highest price in a day, lowest price in a day, and the corresponded transaction volume. And the data contains 4874 observations in total. For research purposes, a new variable called "return" is created by,

$$\frac{adjusted\ closed\ price - open\ price}{open\ price} \tag{3}$$

which indicates the relative return growth value of the Netflix stock. Based on the data pre-processing result, neither missing nor extraordinary value is in the dataset.

Table 1. Display of the Variables in the Used Data [drawn by the author]

Variable Name	Definition
Date	Date
Open	The stock's initial unit sale price on a market day
High	The stock's highest unit sale price on a market day
Low	The stock's lowest unit sale price on a market day
Close	The stock's final unit sale price of its last sale on a market day before the day ends
Adj Close	The adjusted stock's final unit sale price of a its last sale on a market day before the day ends, reflects the real value of the stock's closed price without being interfered with by any other influencer
Volume	The transaction volume of the stock within a certain period, including selling out and buying in
return	The return of a stock, including both gains and losses

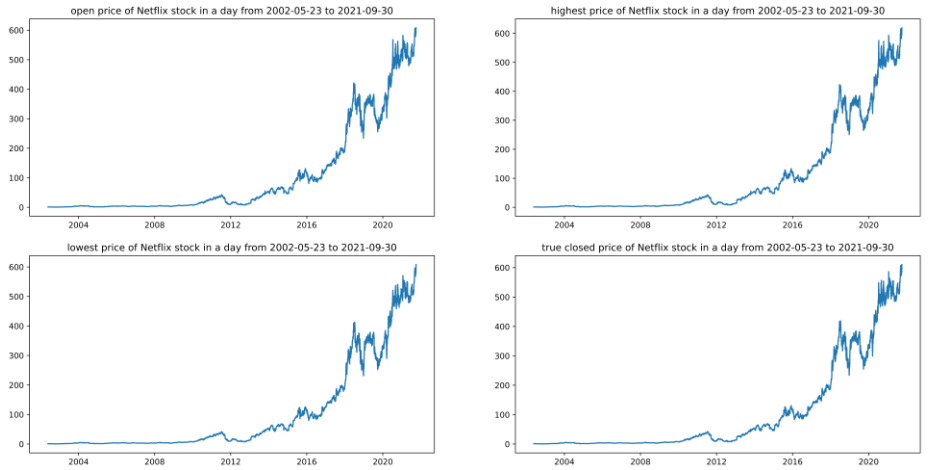

Fig. 3. Graphical display of Netflix stock's daily open price, the highest and lowest price, and the adjusted closed price from 2002-05-23 to 2021-09-30 [drawn by the author]

Stock price adjustment needs to be applied when checking errors or lagged values are detected. The adjustment records and corrects the problems, and it could be either a long-term work or a short-term one [16].

In terms of this data, there is no difference between the closed price and the adjusted one.

3.2 Experiment

This experiment aims to predict the stock return on the eighth day based on the last seven days' stock return. A correct time sequence is necessary. Therefore, the chosen methodology for train-test splitting is time serie**S.** split. With the input variable "Date", the time series split method generates the desired training and testing sets based on the set time interval.

The same training set has been used to train the three models: linear regression, decision tree, and gradient boost models. Each of them has then been tested on the same test set. All parameters used in the models are pre-set parameters corresponding to the different modelling commands in Python 3.0. Each model's output can be plotted into a scatter graph as Figure 4 to Figure 6.

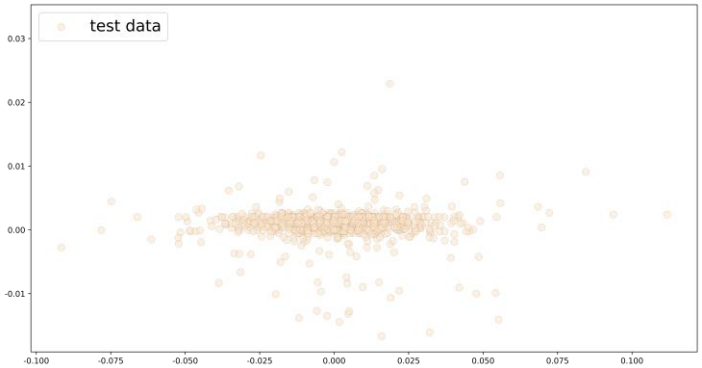

Fig. 4. A scatter plot displaying linear regression model's prediction outcomes using the test set as an input source [drawn by the author]

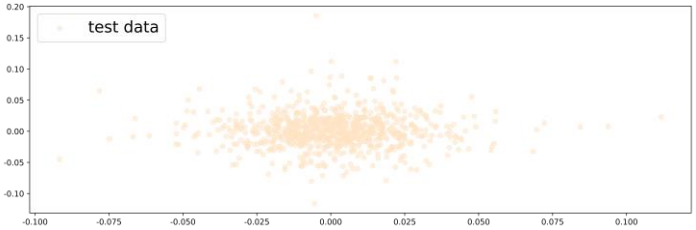

Fig. 5. A scatter plot displaying decision tree model's prediction outcomes using the test set as an input source [drawn by the author]

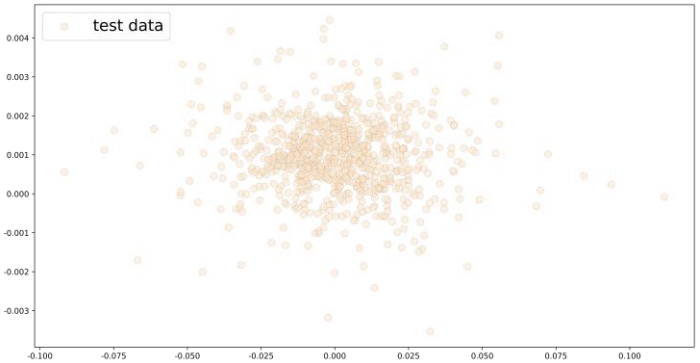

Fig. 6. A scatter plot displaying gradient boost model's prediction outcomes using the test set as an input source [drawn by the author]

In this research, each model has to run through the five-fold cross-validation to evaluate the model's performance as well. Each run provides a mean squared error. And after the five-fold cross validation process ends, the average value of the five mean squared errors will be taken as a cross-validation error. The cross-validation error and the

Python score method's results will then help compare the three different model's performances on prediction based on the dataset.

4 Result and discussion

4.1 Model Performance Analysis

Table 2 displays the result of the scoring method, as well as the five-fold cross-validation errors.

Table 2. Linear regression, decision tree, and gradient boost models' performance evaluation results, based on five-fold cross-validation and score method respectively [drawn by the author]

Assessment Method	Linear Regression	Decision Tree	Gradient Boost
Five-fold Cross-validation Error	0.000648	0.001715	0.000675
Score Method	-0.007227	-1.765061	-0.033744

The cross-validation errors and the scoring method show that the linear regression model performs the best while the decision tree model has the poorest performance. The gradient boost model's cross-validation error is similar to the linear regression one's with only about 0.000018 differences, while the two models' scores differ a lot from each other.

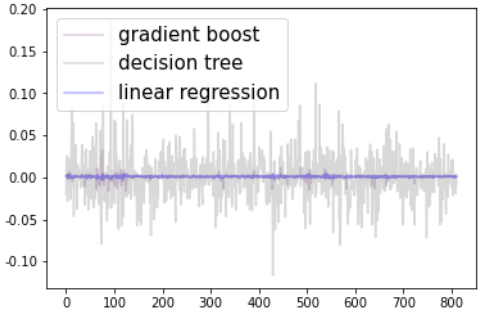

Fig. 7. Graphical presentation of the predictions that have been made by linear regression, decision tree, and gradient boost models based on the same test set [drawn by the author]

According to the graphical exploration, the difference between the predictions of the linear regression model and the gradient boost model is not very large. There is overlap between the predictions of the two models towards the same test set. This phenomenon, to some extent, suggests that the growth of Netflix stock returns and time could probably have an linear relationship, which may be abnormal as some researchers have revealed that the relationship between the stock return and time is nonstationary and nonlinear [17]. But other research shows that the linear model is the best predictive model

in cases where the test set does not have many extreme values [18]. As the data used in this experiment is on a large time scale and does not have many extreme values, in line with the normal long-term development of an ordinary stock, the discovery of a linear relationship is not an unexpected exception [19].

For the gradient boost model, it is a complex model composed of multiple weak models that improve on previous residuals in a continuous iterative process. Its composition structure means that the cumulative variance of the model will keep increasing with each iteration, which could be one of the reasons it did not perform better than the linear regression model in this experiment. Besides that, the gradient boost method works more efficiently with a small dataset rather than a large one. Although a small sample size will cause an increase in the variance produced in each iteration, the correlation between each estimate will decrease, causing another decrease in the overall cumulated variance of the combined model [13]. Furthermore, as a gradient boost model is a complex composition of numerous weak learners, the setting of parameters is necessary for making an accurate prediction. However, the parameters that have been used for this model within this research are just the pre-set ones.

The decision tree model does not show good performance as an individual prediction model within this experiment. As the dataset is relatively large and contains many numeric data, the absence of pruning may cause the decision tree model to create multiple nodes with the wrong splitting strategy [10].

4.2 Time Series Split Analysis

As the research direction for this experiment is to predict the eighth day's Netflix stock return based on its last seven days' data, the time series split method is required for this goal. However, the impact the splitting method leaves on the models' performances should be considered and discussed.

According to research, splitting techniques for generating training sets and test sets affect the prediction results of different models based on the same data [20]. The most significant problem with the time series split is that variability will be added to the data with each split [21]. But since this problem also occurs with other splitting techniques and the time series split method for splitting the data is necessary for the designed experiment, this paper will not further discuss its downsides.

5 Conclusion

The linear regression model and the gradient boost model, as two models with good performances based on this dataset, present adequate predictions towards Netflix stock return before the COVID-19 epidemic outbreak, which is not surprising as the data for training and testing the model is on a long-term scale and does not have really significant growth or drop. Nevertheless, the experiment in this paper reflects a linear relationship between long-term stock market data and the time it crosses, under the circumstance that no severe drop or growth occurs within the data. The gradient boost model, as a combined model based on weak learners, also demonstrates relatively good

performance in this experiment. Although its overall performance is not as good as that of the linear regression model, its accuracy without parameter adjustment is noteworthy.

What is clear is that stock market forecasts are important for a wide range of the public, including the company issuing the stocks, the stockholder, and the potential buyers within the stock market. But when a stock suffers a severe breakdown or increase in return, whether the models should be trained and evaluated using long-term historical data should be taken into consideration. For future predicting with high accuracy and maintaining the forecast model's adequate coordination and freedom, the gradient boosting decision tree model may be a good choice. It has high efficient performance when dealing with small data and low dimension, and also could make improvements with its efficiency and scalability with the help of specific sampling and bundling methods when facing a dataset with a large size in high dimension.

After all, this research testifies that the relationship between time and stock returns could be linear in the long run with no extreme outbreak or fall. Meanwhile, the conclusion contributes to the existing research on stock return prediction and provides reference comments for the investors in the stock market to make better choices towards the purchase and investment choices.

References

1. E. F. Fama, The Journal of Finance, Papers and Proceedings of the Twenty-Eighth Annual Meeting of the American Finance Association New York, N.Y. December, 28-30, 1970, Vol. 25, No. 2, pp. 383-417.
2. W. Huang, Y. Nakamori, S. Wang, Forecasting stock market movement direction with support vector machine, Computers & Operations Research, 2005, Vol. 32, No. 10, pp.2513-2522, ISSN 0305-0548.
3. A. Joseph, C. Turner, R. Jeremiah, Comparative Analyses of Stock Returns Properties and Predictability, Procedia Computer Science, Vol. 95, 2016, pp. 272-280.
4. T. H. McInish, R. A. Wood, A transactions data analysis of the variability of common stock returns during 1980–1984, Journal of Banking & Finance, Vol 14, No. 1, 1990, pp. 99-112, ISSN 0378-4266.
5. A. Mahata, A. Rai, Md. Nurujjaman, Om. Prakash, Modeling and analysis of the effect of COVID-19 on the stock price: V and L-shape recovery, Physica A: Statistical Mechanics and its Applications, Vol. 574, 2021, 126008, ISSN 0378-4371.
6. L. J. Lockwood, S. C. Linn, An Examination of Stock Market Return Volatility During Overnight and Intraday Periods, 1964–1989, The journal of FINANCE, Vol. 45, No. 2, 1990, pp. 591-601.
7. P. Ou, H. Wang, Prediction of Stock Market Index Movement by Ten Data Mining Techniques, The Canadian Center of Science and Education, Modern Applied Science, Vol. 3, No. 12, 2009.
8. D. M. Hawkins, The Problem of Overfitting, Journal of Chemical Information and Computer Sciences, 2004, 44 (1), pp. 1-12. DOI: 10.1021/ci0342472.
9. W. Ploberger, W. Krämer, K. Kontrus, A new test for structural stability in the linear regression model, Journal of Econometrics, Vol. 40, No. 2, 1989, pp. 307-318, ISSN 0304-4076.

10. S. Singh, P. Gupta, Comparative Study ID3, CART and C4.5 Decision Tree Algorithm: A Survey, International Journal of Advanced Information Science and Technology (IJAIST) ISSN: 2319:2682 Vol.27, No.27, 2014.
11. H. Hauska, P. H. Swain, The Decision Tree Classifier: Design and Potential, 1975.
12. Y. Song, Y. Lu, Decision tree methods: applications for classification and prediction. Shanghai Arch Psychiatry, 2015.
13. J. H. Friedman, Stochastic gradient boosting, Computational Statistics & Data Analysis, Vol. 38, No. 4, 2002, pp. 367-378, ISSN 0167-9473.
14. A. Natekin, A. Knoll, Gradient boosting machines, a tutorial, 2013.
15. Y. Zhang, A. Haghani, A gradient boosting method to improve travel time prediction, Transportation Research Part C: Emerging Technologies, Vol. 58, Part B, 2015, pp. 308-324, ISSN 0968-090X.
16. M. Wahab, Price dynamics and error correction in stock index and stock index futures markets: A cointegration approach, Journal of Futures Markets, 1993.
17. P. C. Biswal, An analysis of stock prices in India: wavelets and spectral applications, 2002.
18. P. H. Frances, D. V. Dijk, Forecasting Stock Market Volatility Using (Non-Linear) Garch Models, Journal of Forecasting, Vol. 15, 1996, pp. 229-235.
19. J. Bouchaud, R. Cont, A Langevin approach to stock market fluctuations and crashes. Eur. Phys. J. B 6, 1998, pp. 543–550.
20. B. LeBaron, A. S. Weigend, A bootstrap evaluation of the effect of data splitting on financial time series, IEEE Transactions on Neural Networks, Vol. 9, No. 1, 1998, pp. 213-220.
21. J. J. Faraway, On the Cost of Data Analysis, Journal of Computational and Graphical Statistics, 1:3, 1992, pp. 213-229.