# Predicting Stock Market Trends with Multiple Machine Learning Models

Brendan Bebb, Brett Rogers, Jared Zwycewicz

December 6, 2022

## 1 Abstract

In this project we work to recreate and improve a prediction model designed in a dissertation by Bin Weng at Auburn University to analyze stock market trends with the use of two machine learning models: support vector machines and adaptive boosting trees. Using previous stock market data for Apple Inc., we recreate the five support vector machine models outlined in the dissertation. Instead of attempting to add in more variables to the models as done in the dissertation, we use AdaBoost with the same variables as the SVMs to see if the misclassification rate can be decreased. Our models are run on actual AAPL data and results on prediction accuracy are provided.

## 2 Introduction

Following a pandemic that has led to a spike in inflation, the United States begins to brace for a potential economic landslide. According to Business Insider, [1] "the US economy may avoid a recession, but even if it does, the stock market is royally screwed."
In a market that is flooded with competitors trying to market themselves as the best source for predicting stock market trends, this project attempts to find an alternative method to predict changes in the Apple Inc. stock using historical data obtained from Yahoo Finance to train an adaptation of an SVM model developed by Bin Weng of Auburn University. This project then aims to give users an even more optimal predictor by using the same predicting parameters Weng implemented, but transcends the model into adaptive boosting trees to improve model accuracy and decrease the misclassification rate obtained in the support vector machine models.

## 3 Related Work

In Weng's study, he uses five different targets to predict AAPL stock, created using the daily open, close, high, and low values of each stock, accompanied with the volume of stock trading present for the stock each day. Below is the list of targets created.

| Target | Formula |
|---|---|
| Target 1 | $Open(i+1) - Close(i)$ |
| Target 2 | $Open(i+1) - Open(i)$ |
| Target 3 | $Close(i+1) - Close(i)$ |
| Target 4 | $Close(i+1) - Open(i)$ |
| Target 5 | $Trade\ Volume(i+1) - Trade\ Volume(i)$ |

Weng formed these SVM targets using macroeconomic predictors and by modifying artificial neural networks that targeted excess stock returns designed by Enke and Thawornwong in 2005, as well as an adaptation of ANN, PCA, and CART that Tsai and Hsiao created in 2010 that produced a 79% accuracy. [2] Weng combined both of these previous works to implement one of the only SVM models used to predict stock market outcomes in recent history. The five targets were used to predict each individual target in each of the five predictive models: one target serves as the response, and the other four targets serve as predictors for that specific target.
It is important to note that all targets are binary variables, with 0 representing a decrease, and 1 representing an increase from the previous day. For example, Target 1 from the above table would produce a 1 for a day in which the following day's opening stock value is higher than the current day's closing value, or a 0 for a day in which the following day's opening stock value is lower than the current day's closing value.

## 4 Dataset/Features

The Apple stock (AAPL) market data used in Weng's work was obtained [3] publicly from the Yahoo Finance website. The stock movement was measured for a 37-month period from May 1, 2012 to June 1, 2015. The daily opening and closing prices, daily high

and low, and volume of trades of the AAPL stock were in the data. With the data, Weng used the variables to generate his five one-day-ahead targets. We have added in columns of the binary classification aforementioned above to directly create five predictive models with each target as a response, and the other four targets as predictors.

# 5 Methods

## 5.1 Additional Features

To further improve his model, Weng added daily counts of Google News articles posted referencing the AAPL stock, the number of unique visitors on pertinent Wikipedia pages per day, and commonly used technical indicators that reflect the price variation over time. Some of his technical indicators included a moving average, exponential moving average, disparity, rate of change, relative strength index, and multiple momentum variables for the Google News and Wikipedia data.
After generating each of the five SVM targets, recursive feature elimination was used to select significant features for predicting each of the five targets. Weng applied this method to find a subset of predictors that could result in accurate predictions without overfitting the models. As expected, each model obtained a different set of predictors. Each model was reduced to having 20 unique features.

## 5.2 Kernelization

With a dataset of this size, Weng determined that it was unlikely that the data would be linearly separable, thus he opted to handle this issue by using a kernel function. His analysis concluded that the Radial Basis Kernel function in his SVM classification algorithms resulted in the best performance.

## 5.3 Performance Measures

For analyzing the results of both the initial model and the complex final model, Weng utilized these eight evaluation metrics: accuracy, sensitivity, specificity, precision, F-measure, Matthews Correlation Coefficient, Geometric Mean and the Area Under the Curve. In particular, Weng focused the accuracy, sensitivity and the Matthews Correlation Coefficient(MCC).

- Accuracy: $\frac{TP+TN}{TP+FP+TN+FN}$

  The percentage of the sum of true positives and true negatives divided by the total number of predictions.

- Sensitivity: $\frac{TP}{TP+FN}$

  The percentage of the correctly predicted positives divided by the total number of actual positives.

- MCC: $\frac{(TP \cdot TN)-(FP \cdot FN)}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$

  The correlation coefficient between the predicted values and the true values. The MCC produces a high value only if the prediction obtained good results in all four confusion matrix categories. For example, an MCC of 1 implies perfect predictions, and an MCC of 0 implies predictions no better than random.

## 5.4 Improvement

Below is the change to the evaluation metrics from Model 1 to Model 2. The metrics are the average of the five SVM models before and after including the Google News, Wikipedia, and technical indicators parameters and running feature elimination on each SVM.

| Model | Accuracy | Sensitivity | MCC |
|---|---|---|---|
| 1 | 0.616 | 0.618 | 0.232 |
| 2 | 0.858 | 0.838 | 0.719 |

The importance of increasing all terms, especially the MCC, is that not only is the model becoming more accurate in its predictions, but it is also becoming more consistent in its predictions. This results in less variability amongst correct predictions.
Based on the fact that Weng's model produced 85.8% accuracy, we chose to focus specifically on that performance measure when recreating his targets as well as our attempts to improve it in a different manner.

# 6 Experiments/Results

## 6.1 SVM Recreation

In our project recreation of Weng's work, we began by obtaining the exact data from Yahoo finance over the described time period. After importing the data, we used the five target structures outlined above to create additional columns in the dataset with each of their values. We then split the dataset into two sets: 70% training and 30% test. While Weng used an 80/20 split of the data, we chose to use a 70/30 split because we wanted to see more of the model's

capabilities to predict unseen results while not having as much data to learn from initially. These sets were then used to train and test our five SVM models, predicting each of the five targets using the other four targets. Using the trained models to predict the testing data group, we obtained the following results:

| Target | Test Error | Accuracy |
|---|---|---|
| 1 | 0.324 | 0.676 |
| 2 | 0.204 | 0.796 |
| 3 | 0.169 | 0.831 |
| 4 | 0.416 | 0.584 |
| 5 | 0.658 | 0.342 |

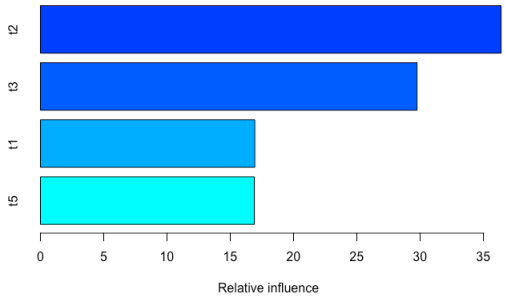
As a baseline, we use Target 3's model to compare our other targets against, as the third target is the most naive: if the current day's stock value went down, then we expect the next day's stock value to be higher than the current day's. This naive inference produced a test error of 0.169, which turned out to be the smallest test error of all of the models used in our recreation.
The most simple target SVM model produces the most sought after prediction in the stock market: when to buy low value stocks to profit financially. Given this, we thought that it would be better to use a different predictive modeling system to further improve the accuracy of each target's model compared to its SVM model: Adaptive Boosting trees.

### 6.2 Adaptive Boosting

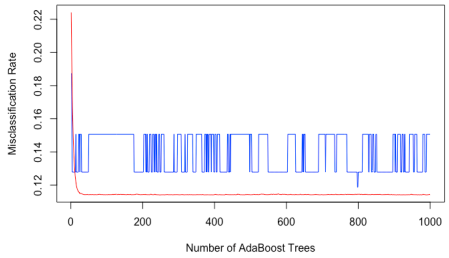
Staying within the scope of this class, our thought for advancing the accuracy of the ability to predict AAPL stock's movements was to implement an adaptive boosting algorithm using gradient boosting machines and binary search trees. We created gradient boosting machines for each of the targets, using the other four targets as predictors for tree amounts between 1 and 1,000. Comparing the training and testing error for each of the models, we observed the following:

| Target | Test Error | Ideal Number of Trees |
|---|---|---|
| 1 | 0.3014 | 45 trees |
| 2 | 0.1644 | 20 trees |
| 3 | 0.1781 | 15 trees |
| 4 | 0.1187 | 800 trees |
| 5 | 0.3470 | 1 tree |

When predicting Target 4, the adaptive boosting model using the other four targets has the lowest test error, as shown in the test error table above.
Below is the graph of the relative influence for the adaptive boosting prediction of Target 4. We see that Target 2 has the greatest influence on the prediction of Target 4, while Targets 1 and 5 have the smallest influence on the prediction of Target 4.




Pictured below is the graph of the training and test error for the adaptive boosting model of Target 4. We see that the minimum training error is attained with very few decision trees, roughly 15, compared to the test error that is minimized with 800 decision trees. These are the ideal number of trees because minimizing the test error also maximizes the accuracy of the model. The reason for the variance and non linear shape of the test error is due the model struggling to recognize a general output pattern due to the nature of adaptive boosting decision trees: being jointly related to its input variables.




## 7 Conclusion

### 7.1 Summary

We investigated the accuracy of two machine learning models for stock market trend predictions on a historical real-world dataset. Using just the dataset's values and Weng's target parameters, we were able to lower the misclassification rate of SVM models in the form of AdaBoost models. Although this work was unable to lower the misclassification rate for all five models previously created, it was able to not only identify the most and least ideal SVM target models from the dissertation, but also decrease the misclassification rate for four of the five SVM target models.

While the accuracy of trees did not improve the models as much as Weng's additional parameters investigating Google search volume did, AdaBoost has proven to be an effective and optimal machine learning model to predict trends in the stock market, as the misclassification rate decreased for a majority of the basic target models that it modified. This is because of the effectiveness of AdaBoost and its ability to avoid overfitting common in basic decision tree models as well as support vector machines. Adaptive boosting trees do not jointly optimize the input parameters, which means that the models are less prone to overfitting.

We were able to identify Target 4 as the best model from Weng's dissertation, as it had the most relative influence as a predictor for the other four models it worked to predict the response to. Logically, this makes sense to use, since the opening of the current day and the closing value of the next day are two predictors that are very effective at showing predictive results: you would want to know what the next day's closing value will be in order to determine if you should buy or sell.

In addition, based on the testing and training error received for both SVM and AdaBoost versions of each target, we were able to conclude that Target 5 was the least effective target to use to predict stock market trends. This is logical, as the trade volume for a given stock on one day compared to another is not the best way to predict whether or not a stock's value will increase or decrease.

### 7.2 Future Work

Adaptive Boosting has proven itself to be an optimal machine learning model to predict stock market trends. While we settled on accuracy as our performance measure of focus, future steps would include training the AdaBoost model on additional evaluation metrics, such as MCC and sensitivity. The first change that we could make to all models would be to use an 80/20 split of the data instead of 70/30 to see if the AdaBoost models perform better with more training data. Just as Weng added in parameters measuring the Google News traffic and Wikipedia search frequency involving Apple stock over this time period, we could also include these in our AdaBoost model to improve the overall success and performance of the predictions. In addition to including the parameters from Weng's study, we could also add more historical data from Apple Inc., as well as other large blue-chip companies so that the model has more historical data to train itself on. This provides a strong baseline for future works involving AdaBoost models that could be enhanced with the aforementioned tuning parameters and assessed by evaluation metrics other than accuracy.

## 8 Contributions

**Brett Rogers:**
Created SVM models and all tests carried out with them, wrote Methods, Experiments/Results
**Brendan Bebb:**
Created AdaBoost models and all tests carried out with them, wrote Abstract, Introduction, Conclusion
**Jared Zwycewicz:**
Project proposal, wrote Related Work, Dataset/Features

## References


[1] https://www.businessinsider.com/stock-market-crash-wall-street-2023-us-economy-avoids-recession-2022-11

[2] Bin Weng. *Application of machine learning techniques for stock market prediction.* Dissertation, Auburn University, Auburn, AL, USA, 2017.

[3] https://finance.yahoo.com/quote/AAPL/history?period1=1335830400&period2=1433116800&interval=1d&filter=history&frequency=1d&includeAdjustedClose=true