Predicting Stock Prices

Elias Haddad
ehaddad2@u.rochester.edu
University of Rochester
Rochester, NY

Mason Wischhover
mwischho@u.rochester.edu
University of Rochester
Rochester, NY

ABSTRACT

This paper provides a comprehensive overview of the data mining process used to predict Apple's (AAPL) closing stock prices given historical stock data from 1984-2017. This project was conducted as the final project for CSC 240 (Data Mining) at the University of Rochester in the Spring of 2023.

KEYWORDS

Stock Price Prediction, Regression Models, SVM, Data Analysis, Data Preprocessing

ACM Reference format:
Elias Haddad, Mason Wischhover. 2023. Stock Price Prediction. In Proceedings of CSC 240 (Elias Haddad, Mason Wischhover). ACM, New York, NY, USA

1 INTRODUCTION

The stock market has always been an important aspect of the global economy, influencing and being influenced by various factors such as economic, political, and social changes. The ability to accurately predict future stock prices has always been a topic of interest for investors, traders, and researchers.

In this paper, we analyze past stock market history and utilize various technical indicators including the Stochastic Oscillator (STOCH), Relative Strength Index (RSI), and Simple Moving Average (SMA). We use these indicators along with linear regression and Support Vector Machine (SVM) models, to predict future stock prices. Our goal is to assess the effectiveness of these models in predicting stock prices and to compare their performance with each other.

We begin with an overview of the data input and preprocessing used to test the models, followed by a discussion of the technical indicators and machine learning models used in our analysis. Finally, we present our methodology and results, and conclude with implications for future research and practical applications. We detail the data preparation and modeling in the following three discrete sections**:**

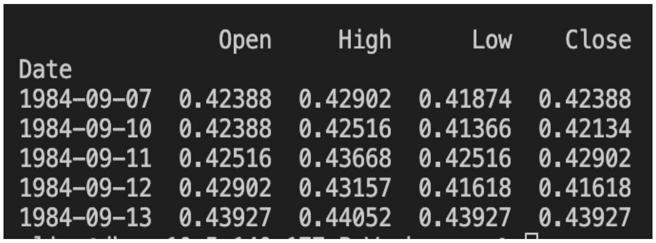
- *Data Analysis*: We analyze here the content of the data and initial cleaning steps we take to ensure successful feature extraction. We also discuss model selection and intuition in this section.
- *Data Preprocessing:* We detail here the distribution analysis of the data attributes and outlier detection process. We also provide an overview of the feature selection process taken to extract the three key features used to train the models.
- *Model Application and Results:* We discuss our results in this section for both models and summarize the short term and long-term predictive abilities of each.

2 DATA ANALYSIS

2.1 The Dataset

We chose the "Huge Stock Market Dataset" available publicly on Kaggle which contains historical stock data for over 7,000 companies. For the scope of this project, we focused on Apple's data, who's file contained 7 different attributes (Date, Open, High, Low, Close, Volume, OpenInt) and over 8,000 records ranging from 1984 to 2017.

2.2.1 Initial Analysis and Cleaning. All the data contained in Apple's dataset was continuous and contained no missing values. This dataset was loaded into a Pandas dataframe with the Date column set as the index. Furthermore, OpenInt was determined to be a 0 column and was thus removed from the dataset. Also, Volume was not used in feature extraction or calculation (which will be discussed later) and was also removed. After this initial cleaning, the remaining attributes of importance were the "Open", "High", "Low", and "Close" which detail the opening stock price on the specific day, the peak, the low, and the daily closing price respectively.



```
               Open     High      Low    Close
Date
1984-09-07  0.42388  0.42902  0.41874  0.42388
1984-09-10  0.42388  0.42516  0.41366  0.42134
1984-09-11  0.42516  0.43668  0.42516  0.42902
1984-09-12  0.42902  0.43157  0.41618  0.41618
1984-09-13  0.43927  0.44052  0.43927  0.43927
```

Figure 2.1: The resulting dataframe after initial cleaning

2.2 Model Selection

We selected a base model and advanced model to perform analysis and predictions with. Since this is a regression problem, we naturally chose Linear Regression as our base model. We then decided to research some common advanced models used for regression analysis and came across the Support Vector Machine (SVM) model which is a well-known model for multidimensional regression. It excels at nonlinear regression problems, and though it's less commonly used for time series analysis or prediction, we decided to test it on our data.

2.2.1 Model Description and Selection Intuition. Linear Regression and SVMs are two popular machine learning algorithms that can be applied to stock price prediction tasks due to their effectiveness in modeling relationships between features and outputs.

Linear Regression is a supervised learning algorithm that models the relationship between the target variable, which is the closing price in this study, and independent variables (or features) by fitting a linear equation to the observed data. We predicted its effectiveness for our regression task since it can accurately capture linear trends and relationships between financial indicators, such as moving averages or the other technical indicators that we employ and closing stock prices.

SVMs are another supervised learning algorithm used for both classification and regression tasks. In the context of stock price prediction, it's typically used as a regression algorithm, called Support Vector Regression (SVR). SVM finds the optimal hyperplane or decision boundary that separates different classes or predicts continuous values with minimal error. It uses kernel functions to handle non-linear relationships in the data, making it more versatile for stock price prediction, as financial markets often exhibit non-linear behaviors[1]. Especially due to this versatility, we predicted it will excel at this regression task of predicting stock prices, more than the basic linear regression model.

$$K(X_1, X_2) = \exp\,(-\frac{\|X_1 - X_2\|^2}{2\sigma^2})$$

Figure 2.2: The SVM's Radial Basis Function (RBF) kernel function that divides the Euclidian distance of two points by the variance, or the hyperparameter.

3 DATA PREPROCESSING

3.1 Distribution Analysis and Outlier Detection

After the initial data analysis and cleaning, we decided to analyze the distribution of our four attributes before creating the features for the models. Using Python's matplotlib library, we produced a series of distribution plots. We also utilized this library to allow us to produce box plots for these attributes for outlier analysis and detection.

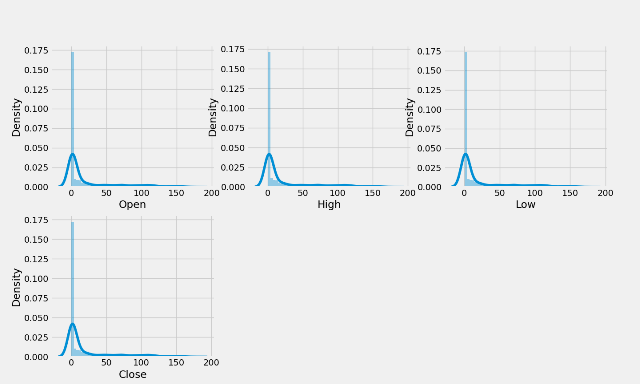


Figure 3.1.1: Distribution Plots of Four Key Attributes

We observed the data is heavily skewed to the left as shown in Figure 3.1.1. However, we decided to preserve the original data when producing our features for three primary reasons. First, skewing the data becomes an issue, especially when using a min-max scalar for target variable normalization. This poses an issue for an attribute like Close, which is later used to evaluate the models' predictive ability given training data that has been normalized using the same scalar. This method would result in meaningless results since it's indirectly contaminating training data with testing reserved data. The second reason is that since our goal in this study is stock price prediction, normalized results would be meaningless since they aren't representing real prices. With this in mind, we need to preserve at the very least the original closing prices. Finally, regression models don't require normalized data to provide a suitable function.

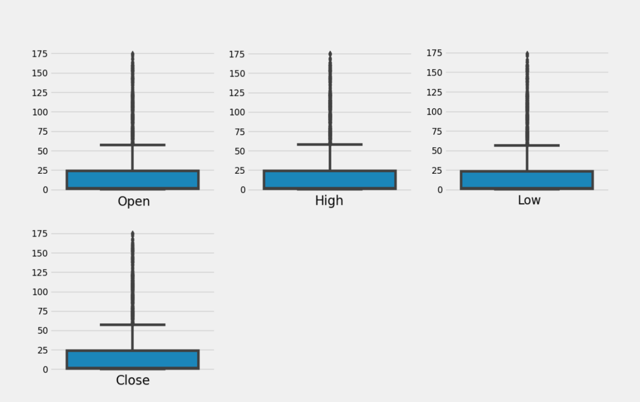


Figure 3.1.2: Box Plots of Four Key Attributes

Based on the box plots of the distributions shown in figure 3.1.2, we determined there were no outliers in the four attributes

3.2 Feature Selection

[1] https://towardsdatascience.com/radial-basis-function-rbf-kernel-the-go-to-kernel-acf0d22c798a

As mentioned prior, our goal in this study was to calculate three features that could help describe trend patterns and indicators in given stock price's historical records. Our brief search of literature and other credible finance-based sources prompted us to include STOCH, RSI, and SMA as the features which we calculate for Apple's dataset using the four raw attributes analyzed in the previous section. These features act as important indicators for pivotal points in analyzing a stock's short- and long-term trend, identifying its general behavior over time while also signaling if there are any sudden reversals that will directly affect the closing price.[2]

Below are the respective features that we calculated for Apple's data. Since the typical number of periods (*n*) analysts favor when looking at historical stock data is 14, we used this value in our feature calculation.

3.2.1 Stochastic Oscillator. This works as a momentum indicator by signaling if the stock is overbought or oversold by looking at past stock data. Usually if a stock is overbought, holders tend to sell their shares resulting in the price of the stock decreasing. The opposite happens when it's oversold. Because of this property, it's especially suitable for predicting whether the closing price in the near future will be higher or lower.

$$\%K = \frac{C - L_n}{H_n - L_n} * 100$$

Figure 3.2.1: Stochastic Oscillator *(%K)* formula where *C* represents the current period's closing price, *L* is the low *n* number of periods in past stock history, and *H* is the high at the historical *n*'th period.

Specifically for the oscillator, a calculated value above 70 indicates the stock is overbought and below 30 indicates it's oversold. The expectation is that exceeding the higher threshold indicates that the overbought stock will cause a trend reversal, thereby resulting in a decreased closing price within the near future.

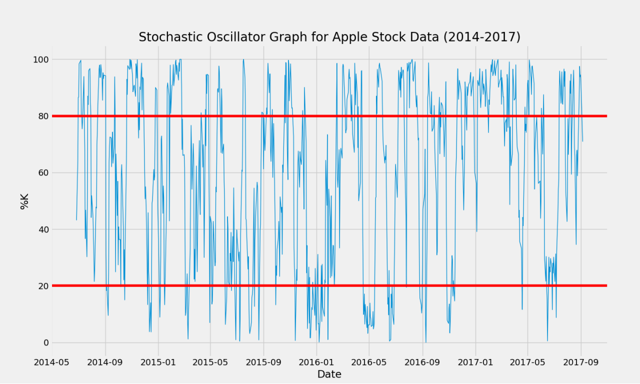


Figure 3.2.2: Apple's STOCH plot from 2014-2017 with the horizontal threshold markers in red

3.2.2 Relative Strength Index. The RSI works also as a momentum indicator, similar to STOCH, identifying critical trend reversal and pullback signals.[3] One distinct difference, however, is that RSI measures the velocity of price movements while the STOCH works to confirm that the closing price should conform closest to the trend. With this in mind, analysts tend to choose the former for relatively trending markets while the latter for more choppy markets. Both indicators are commonly known to be used together for this reason to help better understand market trends and possibly make more sound predictions.[4]

$$RSI = \frac{100}{1 + \frac{n_{up}}{n_{down}}} * 100$$

Figure 3.2.3: RSI formula where n_{up} represents the average of *n* positive return days and n_{down} represents the average number of negative return days.

The overbought and oversold thresholds for this indicator are 20 and 80 respectively.

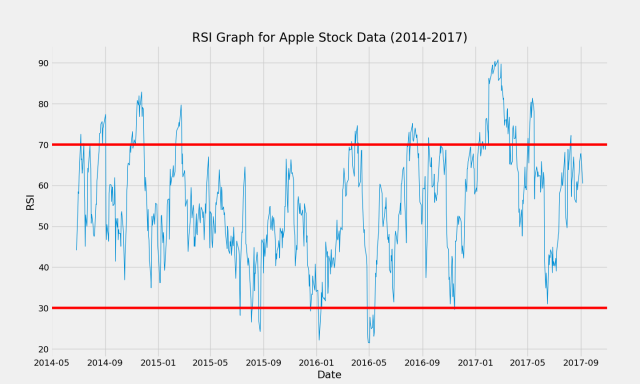


Figure 3.2.4: Apple's RSI plot from 2014-2017 with the horizontal threshold markers in red

3.2.3 Simple Moving Average. This relatively straightforward indicator measures the average trend of the stock based on the closing prices *n* days into the past records. The SMA also smooths out volatility and allows the overall trend of a stock to be analyzed, a suitable compliment to the previous momentum indicators.

$$SMA = \frac{C_1 + C_2 + \cdots + C_n}{n} * 100$$

Figure 3.2.5: Simple Moving Average (SMA) formula where C_n represents the closing price at day *n* in the past.

[2] https://www.investopedia.com/terms/s/stochasticoscillator.asp

[3] https://www.investopedia.com/terms/r/rsi.asp

[4] https://www.investopedia.com/ask/answers/012815/what-are-best-technical-indicators-complement-stochastic-oscillator.asp

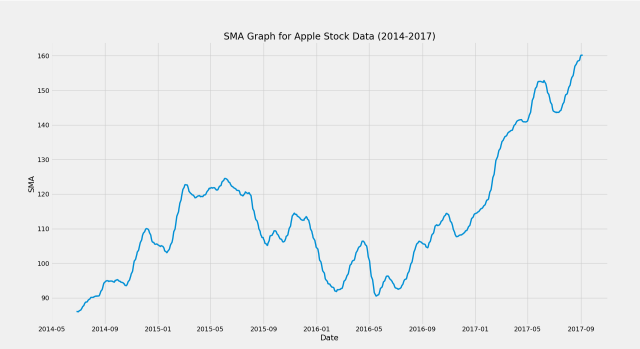


Figure 3.2.6: Apple's SMA plot from 2014-2017

3.3 Train-Test Split

As part of the data preparation portion of the study, we split Apple's data into training data and testing data for fitting the models to. We used the traditional 30-70 split method, where 30% of the data was reserved as testing data, and 70% was allocated for training the models with an arbitrary random state using Sklearn's train-test split function.

4 MODEL APPLICATION AND RESULTS

As mentioned earlier, our goal in the study was to predict the closing prices of Apple's stock using our now prepared features to train suitable Linear Regression and SVM models. To best understand this predictive ability comprehensively, we decided to test both models' performance on predicting Apple's closing stock price m days into the future, where $m \in \{14,48\}$. Using such values of m allows us to better understand short term and long term predictive abilities for both models. In this section, we will also detail the hyperparameterization process for the SVM model as well as the predictive ability results of both models. The Sklearn's implementation of Linear Regression and SVM models were used for this study.

4.1 Linear Regression Model

4.1.1 Parameter Tuning Fitting the model to this data, we kept the original parameters. These included calculating the intercept for the model, especially since the data isn't centered, and other less relevant parameters.

4.2.2 Results Our results at n=14 for the Linear Regression model indicated an R^2 coefficient value of 0.99436 and with a Root Mean Squared Error (RMSE) of 2.8293. At n=48, R^2=0.98482 and RMSE=4.59852.

4.2 Support Vector Machine

4.2.1 Parameter Tuning For tuning the SVM, we found it best to use Sklearn's GridSearchCV which applies specified parameter values to train the model and identify the most suitable combination of parameters. Since we wanted a more versatile model for Apple's nonlinear data, we decided to specify an RBF kernel but allow all other inputs for the remaining C and Gamma parameters. Using GridSearchCV, we found that a C=1e3 and Gamma=0.0001 gave the best results.

4.2.2 Results Our results at n=14 for the SVM model indicated an R^2=0.99512 and RMSE=2.63187. At n=48, R^2=0.98586 and RMSE=4.43898.

5 CONCLUSIONS

As seen in the prior section, the two models performed similarly, producing nearly identical R^2 and RMSE results at m=14 and similar ones at m=48, with the SVM model slightly outperforming the Linear Regression model. In addition, both of these models also have a consistent decrease in predictive ability as they attempt to guess the price further in the future, as the R^2 value decreases by approximately .01 and the RMSE value increases by approximately 1.8 for both. In relation to Kaggle users who performed analysis on the same Apple dataset with more common time series data models such as Long Short Term Memory (LSTM) and Autoregressive Moving Average (ARIMA), our RMSE's in particular are slightly higher however were relatively successful at the regression task, nonetheless. Users who used these more well-known models for this task for instance achieved RMSE's of below 0.3[5].

6 FUTURE GOALS

Though our models were successful at making relatively accurate predictions given the abundant dataset we were able to use, there's opportunity for improvement. In future studies, we would consider first trying make the feature calculation more suitable for longer term prediction, which could involve further manipulation of the current equations or simply updating the number of stock history analysis periods (n) relative to the number of days we want to predict. Furthermore, this study only tested the models on Apple's stock history data. Future goals will include expanding the training and testing data to include a more diverse dataset from the available 7000 companies it offers. The addition of key and useful features for better modeling trends could also very well help in improving the model's stock price prediction capabilities. This could include an Exponential Moving Average (EMA) indicator that's more suitable for recent price actions or other external indicators such as inflation rates or GDP factors. Finally, including other models such as ARIMA or LSTM, which as mentioned before are more commonly used for time series analysis and prediction, would be an option we'd consider in a future study.

ACKNOWLEDGMENTS

We want to thank Professor Pawlicki and the TAs of CSC 240 for making the course an enjoyable and greatly informative one. The knowledge we gained from it was priceless and pivotal to shaping our understanding of data analysis, extraction, manipulation, and numerous other skills.

[5] https://www.kaggle.com/code/neomatrix369/everything-you-can-do-with-a-time-series-stocks

REFERENCES

[1] Marjanovic, B. (2017) *Huge stock market dataset*, *Kaggle*. Available at: https://www.kaggle.com/datasets/borismarjanovic/price-volume-data-for-all-us-stocks-etfs (Accessed: April 25, 2023).

[2] Hayes, A. (2023) *Stochastic oscillator: What it is, how it works, how to calculate*, *Investopedia*. Investopedia. Available at: https://www.investopedia.com/terms/s/stochasticoscillator.asp (Accessed: April 25, 2023).

[3] Fernando, J. (2023) *Relative strength index (RSI) indicator explained with formula*, *Investopedia*. Investopedia. Available at: https://www.investopedia.com/terms/r/rsi.asp (Accessed: April 25, 2023).

[4] Pupale, R. (2019) *Support vector machines(svm) - an overview*, *Medium*. Towards Data Science. Available at: https://towardsdatascience.com/https-medium-com-pupalerushikesh-svm-f4b42800e989 (Accessed: April 25, 2023).

[5] Wang, J. (2020) *How to model time series data with linear regression*, *Medium*. Towards Data Science. Available at: https://towardsdatascience.com/how-to-model-time-series-data-with-linear-regression-cd94d1d901c0 (Accessed: April 25, 2023).

[6] Sreenivasa, S. (2020) *Radial basis function (RBF) kernel: The go-to kernel*, *Medium*. Towards Data Science. Available at: https://towardsdatascience.com/radial-basis-function-rbf-kernel-the-go-to-kernel-acf0d22c798a (Accessed: April 25, 2023).