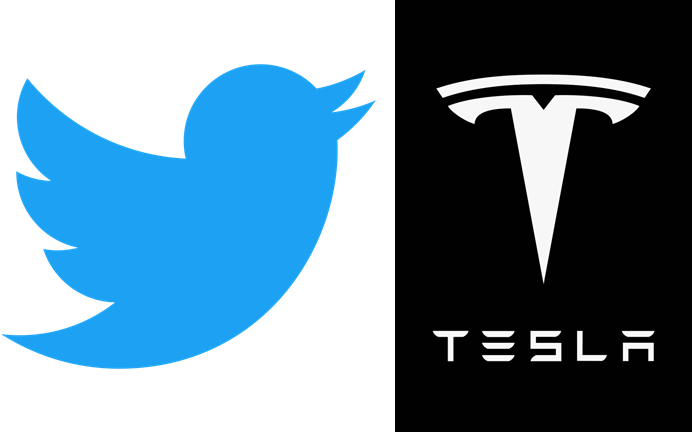
**Using Social Media to Predict**

**Stock Prices**



**Final Report**

**Aaron Zheng**

**MGT 4803 Business Analytics II**

**Fall 2022**

**Introduction**

Investors have been trying to find ways to predict stock prices and estimating the direction of the financial environment for hundreds of years since the beginning of stock markets. With the prevalence of technology and the internet unfathomably increasing the amount of information available, stock markets have become extremely fluid. This meant that investing became an attractive option for many more people and technology has facilitated our semi-strong market efficiency. Aside from the standard newspapers, articles, and publicly released financial reports, another source of information has boomed in the last couple decades – social media.

The main question of this research project is: can social media be used to predict stock prices? To narrow down the scope of this project, I focused on a very prominent figure in the Twitter community, Elon Musk, and his potential impact on his company Tesla’s stock through his social media presence. Can Elon Musk’s Tweets predict TSLA stock prices?

The business case for answering this question regarding the impact of social media on stocks lies in the anatomy of a tweet. To reiterate, a semi-strong market efficiency means that stock prices reflect all historical data and all public data. In the tweet itself, we potentially have hints of insider information regarding the performance of the company or future plans. On the other side of the tweet, in the number of likes, replies, and retweets, lies the reaction of the market. If social media can be used to predict where the stock market is going, it becomes an invaluable tool for investors.

Before looking at a TSLA or Elon Musk related example, let’s consider a very recent incident that happened in mid-November 2022. A tweet from a falsely verified account acting as the pharmaceutical company Eli Lilly said the following: “We are excited to announce that insulin is now free.” This caused Eli Lilly’s stock prices to drop dramatically from $368 per share to $346 per share. This was an approximate $15 billion loss in market capitalization. Understandably so, Eli Lilly has since left the Twitter platform.

Back in 2018, Elon posted about potentially making Tesla private. The result of this was an 11% increase in TSLA’s stock. In 2020, Elon posted one word, “Doge,” and the result was a 20% increase in Dogecoin value. In 2021, Gamestop surged 50% with Elon posting “Gamestonk!” This occurrence may not be directly tied to Elon, but it can be argued that his post drew much attention to the stock. In 2021 again, Elon added “#bitcoin” to his Twitter bio, and Bitcoin saw a 14% increase in price. A few months later, Bitcoin fell 7%, with Elon posting “#bitcoin” with a broken heart emoji. Even if it has been inconsistent, there have been clear examples of how social media almost directly impacted stock prices.

**Impact of Research**

The impact of this research, in short, is extremely difficult to estimate and put a dollar value on. A few important values to look at is the number of shares TSLA has outstanding and its current stock price. TSLA has 3.16 billion shares outstanding with a $580 billion market capitalization as of December 5, 2022. This has since dropped down to roughly $490 billion in the past couple weeks as the stock price has decreased nearly 18% in the last month. Even with all this information, it’s extremely hard to put a number on how much this could impact gains for investors. People have a wide variety of habits in regard to investing habits, such of the type of company they like to invest in, how much they invest, levels of risk tolerance, and so on. Investing institutions have differing habits as well in comparison to individual people. However, it can be said with certainty that there will be an advantage for those that can glean info from social media to predict prices and they will be able to realize sizable gains.

**Assumptions and Data Cleaning**

I was able to find a dataset from Kaggle that contained all of Elon Musk’s tweets from January 30, 2015 to the end of 2020. This can be found in the following link: <https://www.kaggle.com/datasets/vidyapb/elon-musk-tweets-2015-to-2020?resource=download>

This dataset contained many unnecessary variables or repetitive information so data cleaning extremely important. I chose to narrow down the range that I would analyze and decided on a full four years from 2017 to 2020, which included a total of 9712 tweets. I also needed to obtain historical financial data and used Yahoo Finance as a resource. This included history of TSLA, NASDAQ, Dow Jones, S&P 500, and a Tech ETF called XLK. A comprehensive list of the final variables I selected include the following:

|  |  |  |
| --- | --- | --- |
| nlikes | nreplies | nretweets |
| Normalized nlikes | Normalized nreplies | Normalized nretweets |
| Day | Hour | 5-Day Momentum |
| 10-Day Momentum | NASDAQ Change | DJIA Change |
| S&P 500 Change | Monday | Tuesday |
| Wednesday | Thursday | Friday |
| AVG Sentiment | SUM Sentiment | **TSLA Change** |
| **TSLA Up** |  |  |

A full snapshot of the raw dataset can be found using the following link:

<https://docs.google.com/spreadsheets/d/1OztsGHYTrDMc3EZL54MZ_VxU_77t_HEsm5GOfi4YsmM/edit?usp=sharing>

A problem that needed to be considered when analyzing the data was: what if Elon posts at times when the stock market is not open? This may include weekends, holidays, or any other special occasions. My solution was to match the tweets that do not fall on a day during which the stock market is operating to the next business day. Matching all tweets for each day may lead to issues, however, as the stock market is not open for the entire day (only until 4 PM). It could have been more accurate to link tweets after 4 PM to the next business day as well. Pre and post market trading exists but addressing this would mean having to analyze the differences between the main market and the pre and post market. To avoid unnecessarily complicating the project, I settled on simply linking all tweets to the next business day if the stock market is closed.

Another problem to address is: what happens if there are multiple tweets per day? We know Elon posts very frequently, so how would we analyze this when the date needs to be the unique index of the dataset? The potential models that can be formed include the Max model, the Sum model, and the Average model. First, the Max model assumes that the most popular tweet will have the main impact for that day, thus that tweet will be linked to the other financial data while the other tweets are ignored. The Sum model assumes that all tweets for each day will have impact and a combination of them would be that day’s metric. The Average model balances the other two models out using the average metrics of all tweets for that day. The most realistic model, based on the concept of anchoring bias, is likely the most realistic, as people will usually remember the most popular tweets and respond accordingly.

The raw dataset that I attached above shows many of the very complex spreadsheet functions I had to utilize to make sure the financial data aligned with the tweet data. That being a challenge in itself, creating the Max, Sum, or Average models was another issue. For example, creating the Max dataset meant locating the most popular tweet by number of likes for that day and then returning the rest of the columns for that row. The query function in Google Sheets along with some lack of experience made this impossible to fully complete in Google Sheets. Therefore, I had to import the dataset into an SQL database and use SQL code to query the data. It wasn’t as simple as grouping by date and returning the max of each column, as the max of each column for each given day may not fall within the same row of a tweet. Because there were potential duplicate values in the number of likes, replies, and retweets, I had to concatenate the date with those variables into a string. This can be seen in the “Search Keys” sheet in the Google Sheets link. After reuploading the dataset into the spreadsheet, I created the same keys for the spreadsheet and was finally able to match all of the data for the Max model.

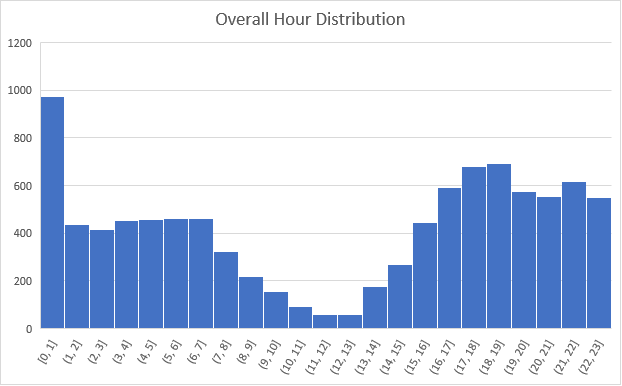
**Data Visualization**

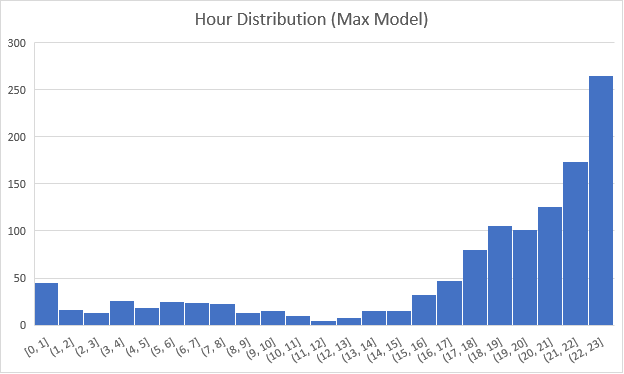
The next pages illustrate the distributions for most of the important variables. Note that if the title includes “Overall,” it is referring to the larger dataset with 9712 observations, while the   
“Max Model” refers to the most popular tweets for each given day.

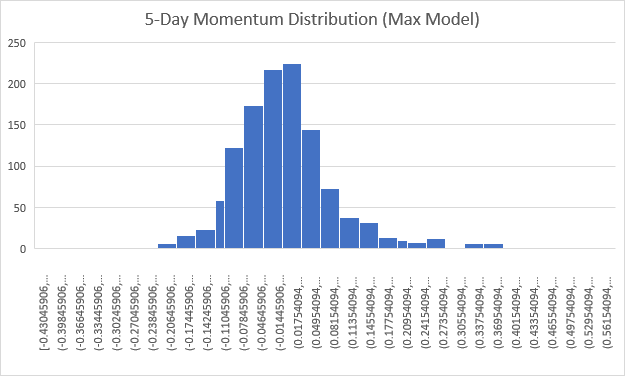
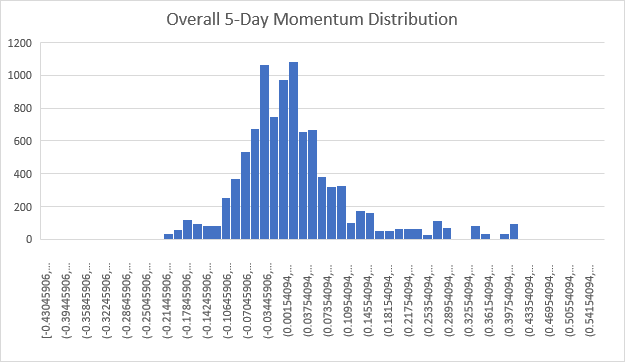
For the first chart showing the nlikes variable, we can see it is extremely skewed. We have a majority of the tweets having less than 10,000 likes in the first group or column, and much less in the more popular buckets. The graph is seemingly empty on the right side but there were a few extremely popular tweets which skewed the visualization. Doing a log transformation is pretty typical for skewed data like this. Instead, another method was used which will be explained later. The replies and retweets distributions were very similar to the likes chart and will not be included.

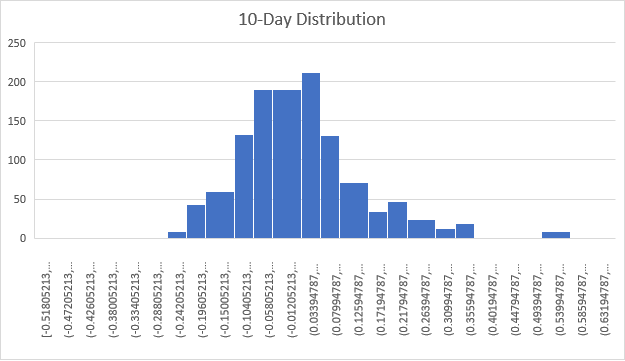
**Text

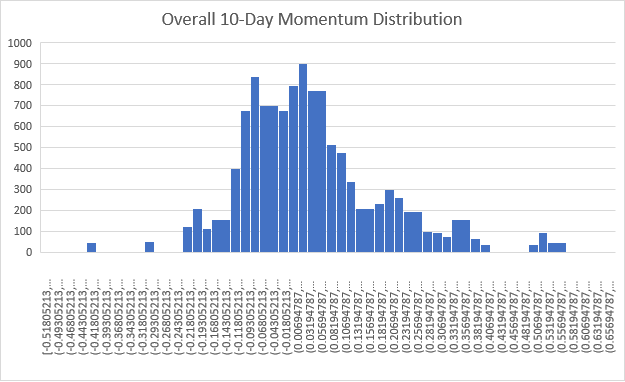
Description automatically generated**

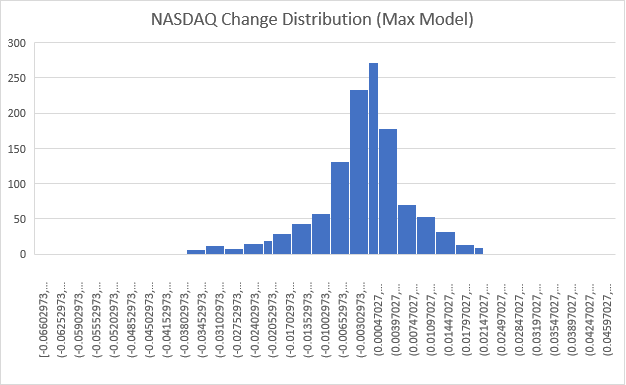


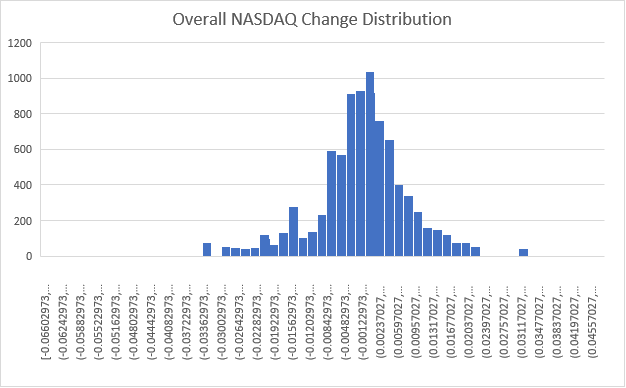


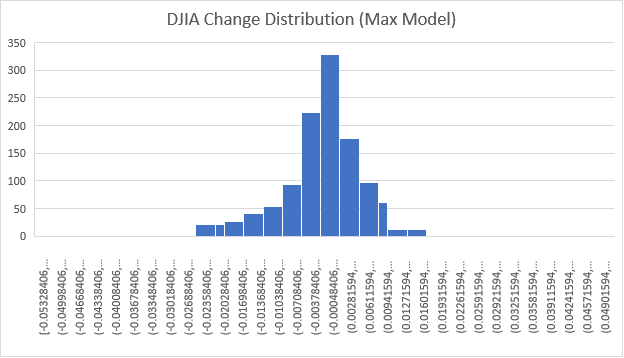


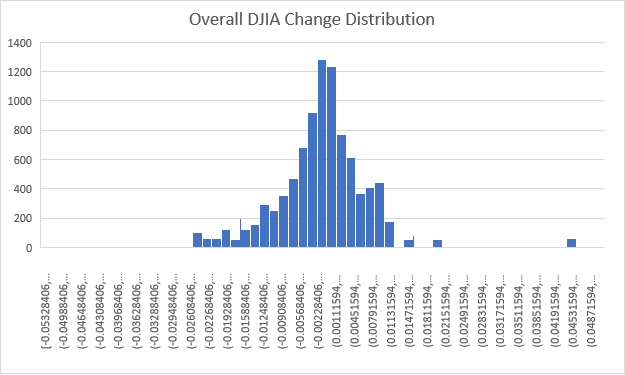


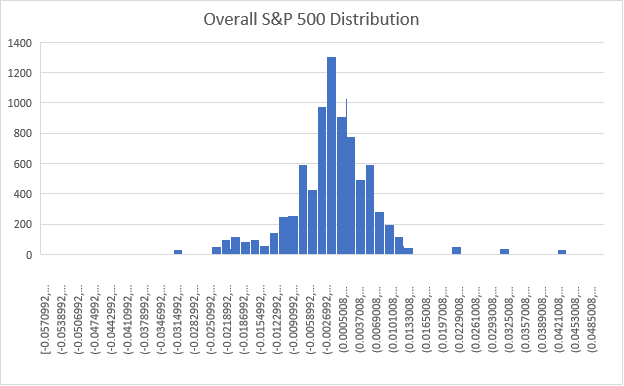


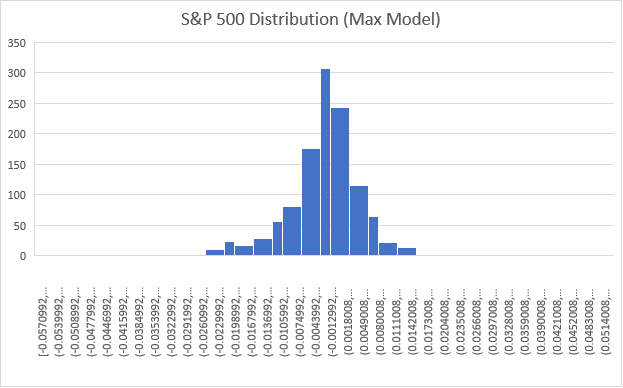


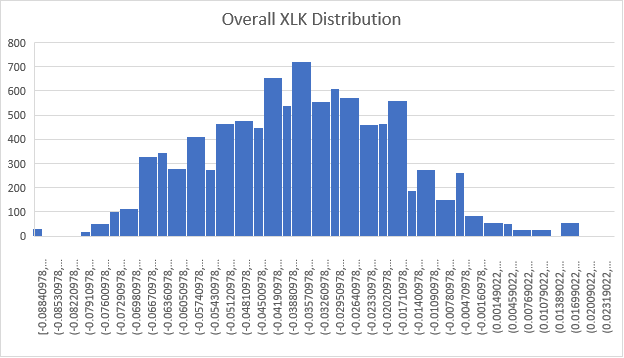


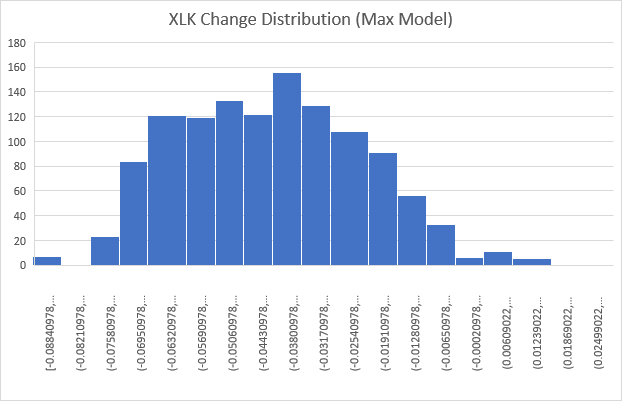






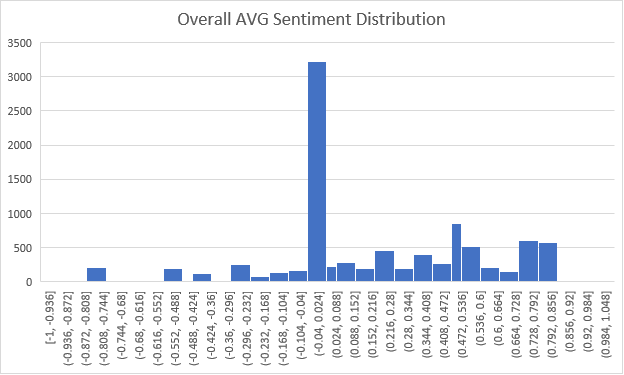


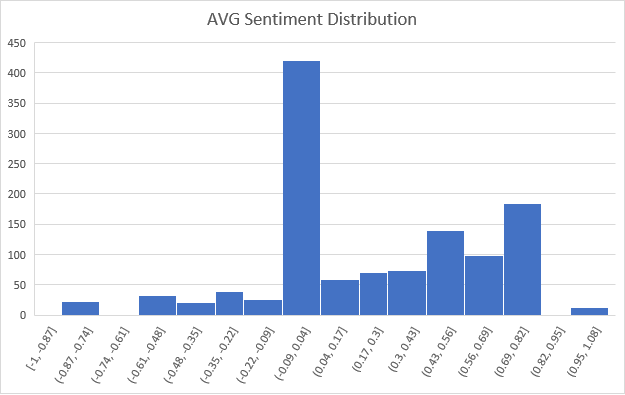


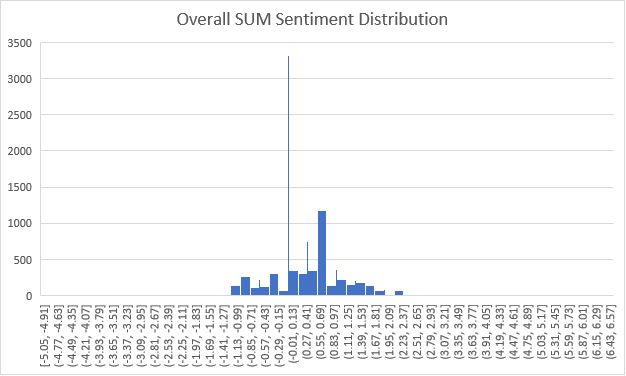


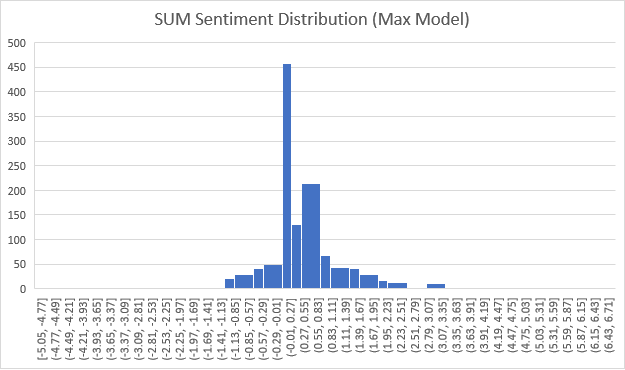
**Chart, bar chart

Description automatically generated**









The majority of the distribution charts do not need further commentary, as most of the overall distributions for certain variables were very similar to its Max model counterpart. An potentially interesting chart to look at is the Hour variable. In the overall dataset, Elon seems to be tweeting less between 7 AM and 3 PM. Perhaps that is when he is resting or in meetings! In the Max model, a majority of the popular tweets seem to have been posted later in the evening past 8 PM.

**Sentiment Analysis**

To perform sentiment analysis, Syuzhet lexicon was used, which scores the positive or negative emotion of the word using a range from -1 to +1. This library was chosen because it contained a much larger list of words than the other general-use libraries such as AFINN, bing, and NRC. Syuzhet has around 14,000 words while AFINN has only around 3300 words. However, there is still a high probability that words cannot be matched, as Twitter language may differ greatly from regular speech or other publications. Something important to note is the inherent weakness of this type of sentiment analysis. For example, a Tweet saying “The least cool characters…it’s embarrassing.“ As English speakers, we can easily tell that this post has a negative connotation. However, the library only picks up on two key words, “cool” with a +0.75 score and “embarrassing” with -0.75 score, meaning that this tweet receives a zero or neutral sentiment when it is actually negative. Without diving deeper into natural language processing, which is a direction this project could be taken later on, this is the limitation of this basic sentiment analysis.

**Linear and Stepwise Regression**

Before starting with any regressions, it is important to understand the relationship between the variables and to check for multicollinearity. Any highly correlated variables result in a redundant and therefore inaccurate regression. Oftentimes, RStudio would refuse to run the formula if this was detected. The correlation table attached below shows some of the aspects about the variables that was already known. First, the likes, replies, and retweets and their normalized versions, using the (x - mean) / standard deviation formula, are directly correlated. When running any regression, it is important to use either the integer version or the normalized version and not both at the same time. If the categorical Weekday variables (0/1) were to be included, the Day variable (1-7) was exluded. The industry indexes had relatively high correlation, which was expected. However, it was important to include all of them in the analysis in order to check which index would be a better predictor variable.

Chart, scatter chart

Description automatically generated

The first linear regression model included all variables with the following results:

Application

Description automatically generated with medium confidence

Text

Description automatically generated with medium confidence

Immediately, it was observed that the features and variables previously thought to be important such as the popularity of a tweet and the sentiment of a tweet were not significant at all. The most significant variables were noted to be 5-Day Momentum, NASDAQ Change, and S&P 500 Change. 10-Day Momentum proved to be too long of a time window to use for analysis. It makes sense that two weeks of trading cannot be used to predict the next day. After all, many things can happen in two weeks, and more recent data typically serves as the better predictor. As the Dow Jones does not include TSLA, it was interesting that in this linear regression it was not significant. All of the weekdays having negative coefficients meant that weekends (the base case where all weekday variables are equal to zero), had the largest effect on stock price. These weekday variables had low significance but had some impact. This also proves that there is a fundamental difference in performance based on what day of the week it is. SUM Sentiment having the lowest p-value without actually being significant shows potential. As mentioned earlier, the basic sentiment analysis had some weaknesses which could be improved on. It can be hypothesized that sentiment score would be significant if a more advanced analysis was performed. Overall, the model including all variables had an adjusted R-squared value of 0.3112.

Utilizing the Max model assumes that the most popular tweets will have the most impact and this idea needs to be incorporated into the regression analysis. Furthermore, with the significant skewing of the nlikes variable, it was hypothesized that ignoring the unpopular tweets would allow for a more accurate model. However, it is difficult to figure out the best cutoff range for the number of likes with trial-and-error. The following code was used determine the best cutoff value:

Graphical user interface, text, application, email

Description automatically generated

The for loop iterates through different cutoff ranges in increments of 100 up to 150,000 likes. It was important to keep at least one significant variable, so whenever the number of variables with a p-value less than 0.001 fell below one, the loop stops immediately and the maximum likes cutoff is returned. It was determined that if all tweets below 50,600 likes were ignored, the R-squared value increases to 0.4778. The result is shown below:

Text

Description automatically generated

With this many variables to consider in the regression, it was important to perform a stepwise regression to identify the most important variables to include. Before I realized that stepwise regression was a built-in function, I ran code to perform it manually.

Graphical user interface, text, application, email

Description automatically generated

First, I created a data frame with the numbers of predictor variables as the number of columns. The data frame was composed of 0’s and 1’s, which indicated whether or not to include a variable. To improve on computation time, the rows with only one variable were ignored. However, that still left 65,399 possible combinations of variables that needed to be tested in regressions to find the maximum adjusted R-squared value. After a long computation time, the following variables were included as the optimal combination:

|  |  |
| --- | --- |
| 5-Day Momentum | 10-Day Momentum |
| NASDAQ Change | DJIA Change |
| S&P 500 Change | Tuesday |
| Wednesday | Thursday |
| Friday | SUM Sentiment |

While DJIA and SUM Sentiment were insignificant variables in this combination, they were still needed to optimize adjusted R-squared. The most significant variables remained the same with 5-Day Momentum, NASDAQ Change, and S&P 500 Change.

Because my rudimentary stepwise regression was done manually, I validated it by using the built-in functions and received nearly exact results with the exception of 10-Day Momentum not being included. The results are shown below:

Graphical user interface, text

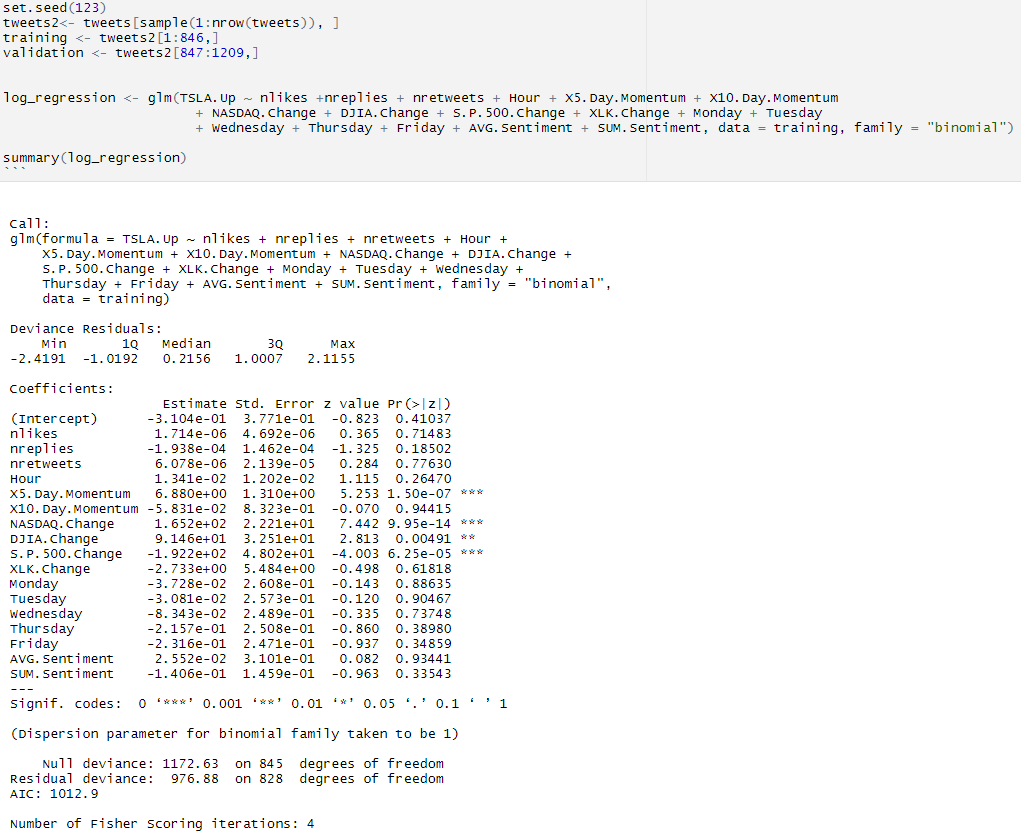
Description automatically generated

Despite the 9 or 10 variable linear regression model having the highest adjusted R-squared value, the simplest model is typically the best model. With a 3 variable model, using the variables that have repeatedly been seen as significant throughout the different iterations of the regression, 5-Day Momentum, NASDAQ Change, and S&P 500 Change, this simpler model only lost 0.007 in its R-squared value. Thus, inclusion of seven additional variables had minimal benefit.

The like cutoff assumption was again tested with the simple model using the for loop. In this iteration, the loop found that 54,600 likes was the best cutoff value to maximize adjusted R-squared. This value increased to 0.358 from 0.3139.

**Logistic Regression**

For the first logistic regression model, all variables were to be included. Although the simple model with only a few variables were tested as well, the accuracy was slightly higher in the model including all variables. The dataset was split in a 70/30 ratio. 70% of a shuffled dataset would be used to create the model, and the remaining 30% would be used to validate and test the model. The results are shown below:



Again, we see the same few variables turning out to be significant. The next step was to choose a cutoff for the prediction value. In other words, tweets with a value below the cutoff would be predicted as 0, or “FALSE” in the TSLA Up variable. If it was above the cutoff, the predicted would be 1 and TSLA Up was true. To avoid arbitrarily picking a random cutoff value, another for loop was utilized to loop through all possible cutoff values from 0 to 1 in increments of 0.01. It was found that 0.52 was the highest cutoff value in order to minimize the misclassification rate of the model.

Text

Description automatically generated

Using this cutoff range produced the following confusion matrix:

Graphical user interface, table

Description automatically generated

Because a 67.7% accuracy isn’t the best, another filtering method was considered by posing the question: under what conditions should an investor *absolutely* invest? By filtering the dataset to only look at tweets that had a prediction value of above 0.75, the accuracy rate was increased to around 82%. Comparing the characteristics of these high predicted value tweets provided valuable insights.

Text, letter

Description automatically generated

To interpret the results, these high predicted value tweets had a much higher 5-Day Momentum, NASDAQ Change, DJIA Change, and S&P 500 Change than the rest of the dataset. In other words, when these conditions are true with TSLA having great momentum and the rest of the market doing better than usual, there is an 82% chance that TSLA stock will increase, according to this model. This information may not seem groundbreaking to the experienced investor, but inexperienced ones may benefit from a lot by taking notice of these predictors.

**Conclusion**

To review some of the features and metrics that were used to potentially predict stock price, which ones were useful? Tweet popularity was not useful as a predictor. The number of likes, replies, and retweets were never significant in any of the models. Sentiment analysis was not useful as a predictor but was inconclusive. As mentioned earlier, the sentiment scores were the closest out of all other insignificant variable to being significant and had the lowest p-value. By exploring better sentiment analysis or natural language processing, this variable could prove to be a lot more impactful. Linear regression in general was not very useful either. The R-squared values were extremely low, only accounting for around 31% of the data. Furthermore, investors do not usually care to guess how much a stock may go up or down, only whether it went up or not. Thus, logistic regression was the best feature to explore in predicting stock prices with social media. As discussed in the previous section, knowing the conditions under which one has the highest chance in obtaining gains on their investment is extremely helpful.

**Future Improvements**

This project has been very interesting but there are a few areas that I could improve on if I were to revisit this project.

* Natural language processing: As I mentioned earlier, the basic sentiment analysis that I performed had many loopholes, such as words not matching or sentiment not being accurate.
* Topic modeling: Because Elon often tweets about many random topics, filtering the data based on TSLA-related things could result in a much more accurate model. This was not explored given the time frame of the project.
* Stock market time: It was mentioned earlier that all tweets on a day no matter the time were linked to that day’s performance. This could have been improved so that tweets after 4 PM are linked to the next day.
* Trend lag: Not all investors immediately invest upon hearing news. It is uncertain the percentage of investors that do react immediately, but the trend of a popular tweet lagging over a period of time after the posting date could be an interesting area of research.
* Follow count normalization: Elon Musk’s follower count has grown *tremendously* over the past few years. My research used data during which he had fewer followers and was less controversial. His follower count also increased a lot over the period of study. While it could have been beneficial to normalize this way, an accurate follower count by day was extremely difficult to find.