PYTHON II - FINAL PROJECT REPORT

President Trump tweets and Stock Market Performance

Gizem Baranoglu Dammati - 11924000; Brigitta Földi - 11849064; Tara Shirehpazazari - 61803573

For the main edge and topic of the final project we have chosen to try and relate President Trump's tweets and stock market performance data. Our main question was *"Is there any relationship between Trump's tweets and the stock market performance?"*. We researched whether there could be an underlying trend set between the two datasets, and whether we could apply machine learning which could successfully predict whether there would be a raise or decrease in the prices of stocks.

In order to implement this, first we needed to obtain President Trump's tweets and properly clean the data. As for the collection of tweets we used the readily available "Twitter Tweets for Donald J. Trump (@realdonaldtrump)" dataset from Harvard Dataverse as an ndjson file and loaded into a dataframe. As can be seen from the outputs of the code, there are many columns included with additional information we might not need. In the dataset there were a total of 40241 tweets collected from 2009 until July 2019.

We had to first clean the loaded data and filter the tweets for the ones which happened after Trump was elected and started his presidency, which happened on the 20th Jan, 2017. This criteria of filtering kept only a total of 8537 tweets eventually, which have been also saved as a text.

As a next step of cleaning the data we have used word tokenization, removed punctuation and separated stopwords, also adding a column which included complete lowercase text. From the original tweet text and the cleaned text we have extracted the features: text length, word count, stopwords, count of nouns, verbs, adverbs, and adjectives. This has been done by further adding more columns into the original dataframe.

At this point in the project we needed to obtain and clean the raw stock market data. We have used the database with S&P500 historical index from 2016 up until the start of 2020. First problem we came across was the different date formats and time intervals, these had to be restructured and made so that they match perfectly in the original and the stock market frame, and so the original dataset can be further expanded with stock market data, then to be used for the machine learning initiative.

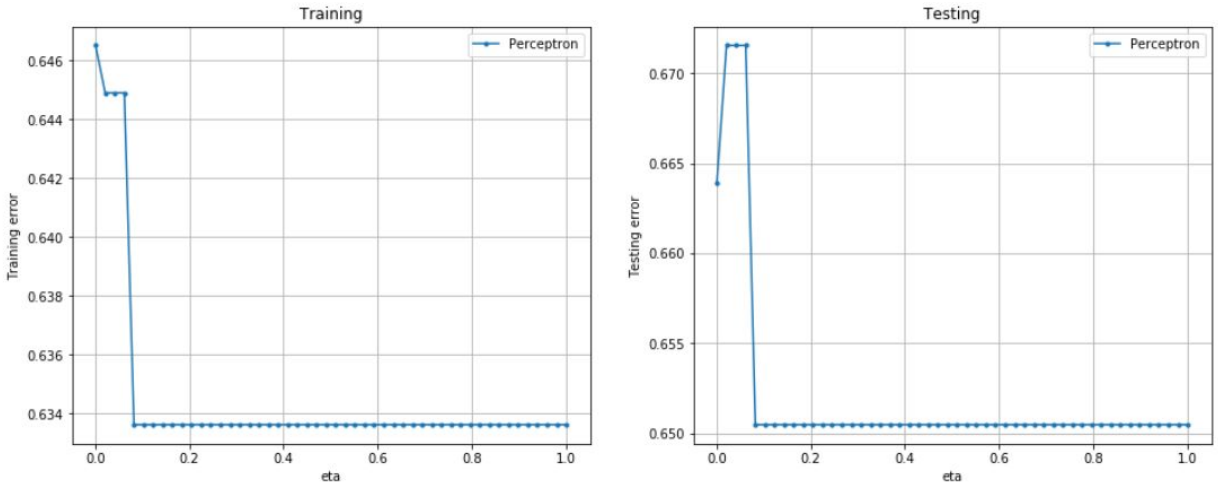
We have included both the stock market data and the tweet data as attachments to the original code, both being necessary for the correct execution of the code.

The S&P500 data has been read into a dataframe first while parsing the date and sorting by time. After that, two more columns have been added to this frame, one called " Daily Difference", which is the difference between opening and closing prices of the respective day, and " Closing Prices Diff", which is the difference between today's and yesterday's closing prices.

We merge the tweets data and the daily stock market performance data based on the "Date" column. Since the stock market data is daily frequency and is also not available on certain days (non-working etc), the merged data would have missing values for the days where tweets were made and no stock market data is available. We addressed this issue in the Preprocessing step later. There were also some null-values included in the dataframe, these were removed in order to make the set of data more stable, processable and later avoid any machine learning mistakes made due to these values.

As a next step Preprocessing has been implemented on the now merged dataframe.
This means, the features vector X and output vector Y have been defined, and the data itself has been split into training and testing sections, then scaled using StandardScaler as a finishing step to standardize the values.

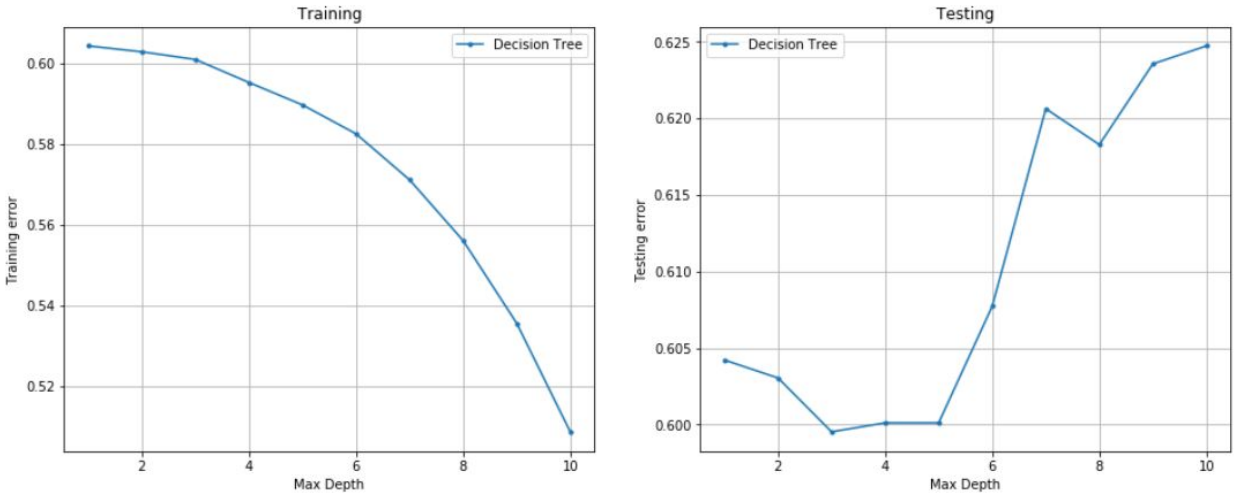
We have set and implemented two different models for checking the relationship of Trump tweets' and stock market performance data. The first model is a Perceptron based one with a range of eta values defined, training and testing data predicted and error rates computed. The result of this model can be seen on the following graphs:



Graph 1: Training and Testing errors with different eta values

As can be seen, both the training and testing error is constant after the eta value 0.08, the testing error fixes around 0.65. From the results presented we cannot find a good, "learnable" relationship between stock performance and tweet content.

Our second model was a Decision Tree, which included a variety of maximum depth values as well.

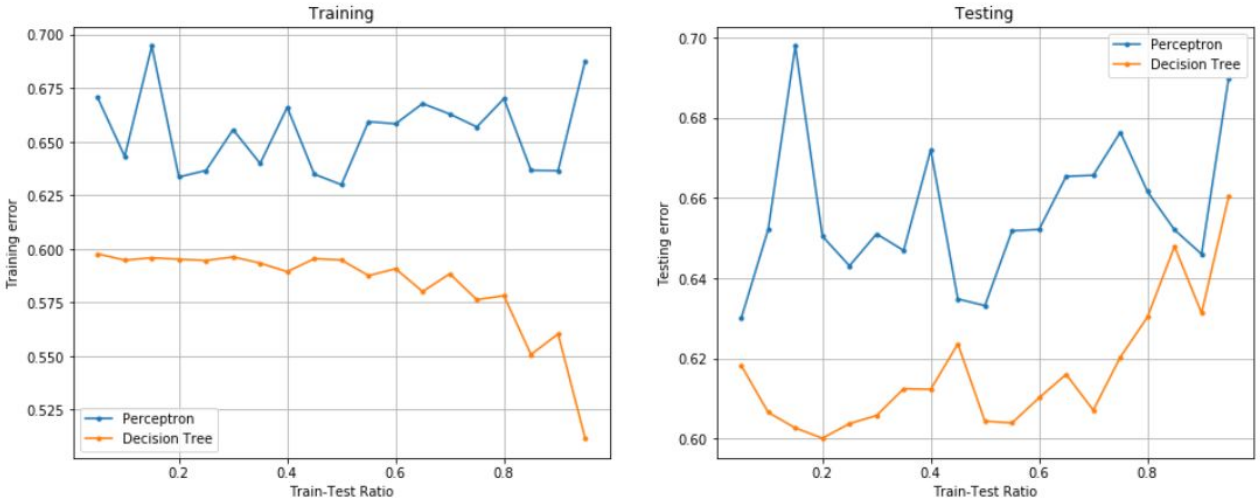


Graph 2: Training and Testing errors with different max depth values

In graph 2, in case of the training data we see the errors slowly improving with an increase in max. depth, which is normal and can be expected from a Decision Tree.

In case of the real testing data, we see that the higher the max depth value is, the higher the produced error values are as well. This is a rather bad sign which also lets us know, we don't have enough proof to conclude that there is a learnable relationship between tweet contents and the used stock market performance data.

We have also measured the effect of test sizes and a comparison between the models, which produced the following results:



Graph 3: Perceptron and Decision tree, Train-Test ratio

The graphs represent a summary of the training and testing data error, taking into consideration the different test and train size of the data. In case of the Perceptron it is evident we cannot improve the error results by adjusting the training and testing splitting of the data. In case of the decision tree, we can see the training data is getting better results once we increase the testing data size, however the testing error is getting worse in case of the testing decision tree.

In conclusion, unfortunately we couldn't find a strong relationship in the given datasets with both machine learning methods we have used.

As a suggested improvement of the research, we would advise to develop the basis for a sentiment analysis. With a large pool of sentiment-tagged tweets, there could be machine learning newly implemented on the same datasets, which could lead to a different outcome. As can be seen from our result, text-specific features alone are not good proxies for determining a relationship between stock market performance and tweets. Our results would most probably improve with a correct sentiment analysis implemented on the datasets, given we have a pool of sentiment specified tweets data, which can be used as training material.