Table of Contents:

Problem Statement

Gauge how change in company executive leadership is received by the general public and investors, and in turn affects company's stock price..
Customer/Client: Corporate Public Relations department

Preparing the data

1. Data gathering

1.1. Data sources: Data sources are Twitter full archive search that allows access to tweets from up to 2006 and News API with access to 1 month previous news. Both allow only 100 results at a time.

Due to restrictions on how many results can be brought back each time and how many times a search can be performed for free access, I was only able to get 100 results per data source per day. Because of this, in some days when there is a large number of tweets or news generated, the time range where the data gathered can be less than a day - in some cases, only 20 minutes. Conversely, if there is less activity in the day, there will be less than 100 tweets or news for that day.

Data is accessed via Web APIs provided by Twitter and Newsapi.org. Once the data is pulled from APIs, they are stored as txt files.

1.1.1. Twitter - Full archive search
https://developer.twitter.com/en/docs/tweets/search/overview/enterprise

1.1.2. News API: https://newsapi.org/

1.2. Data storage and data cleanup:

1.2.1. Data downloaded and saved as txt files in local machine is stored into MongoDB.
The reason for this is for me to review the data and add flag indicating whether the tweet/news is relevant.
Many of the tweets are not related to the company being searched or even if they are, they only have the URL to the web article which is not very useful. It is not in the scope of this project to pull in and parse the web articles.
Also for News articles from NewsAPI.org, they only display up to 1024 characters. Content related to the company being searched may be in the article but not included in the first 1024 characters, in which case, it will be

considered not relevant.
I also manually added topics for each of the relevant tweets/articles to understand what are the common topics.

1.2.2. Notebooks:
Get_Twitter_Feed_Full_Archive_WC
Get_NewsFeed_WC

2. Data wrangling

After the data cleanup by removing non-relevant tweets and articles, the concern is that not enough tweets/articles are useful. This may have to be handled via resampling.
Notebook: Wrangle_data_in_Pymongo

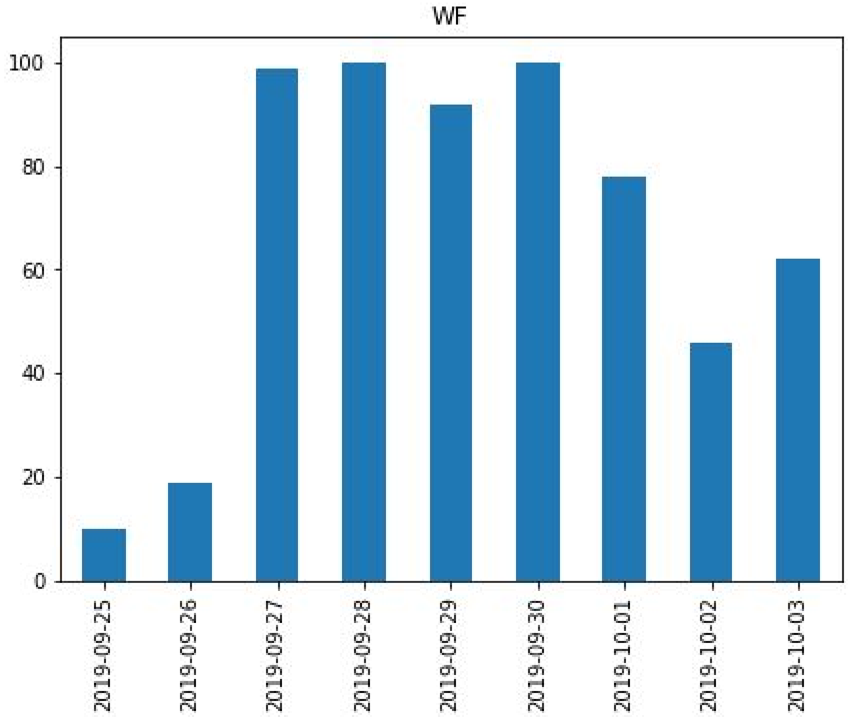
2.1. Read data from MongoDB and add to dataframe
Data stored in MongoDB is read in and added to dataframe. Only data necessary for analysis is added to dataframe.
Although topic information is within each tweet or news article row, topic can be in “1 to many” relationship with a tweet or news so topic is stored in separate dataframe and will be joined with tweet or news dataframe as needed.

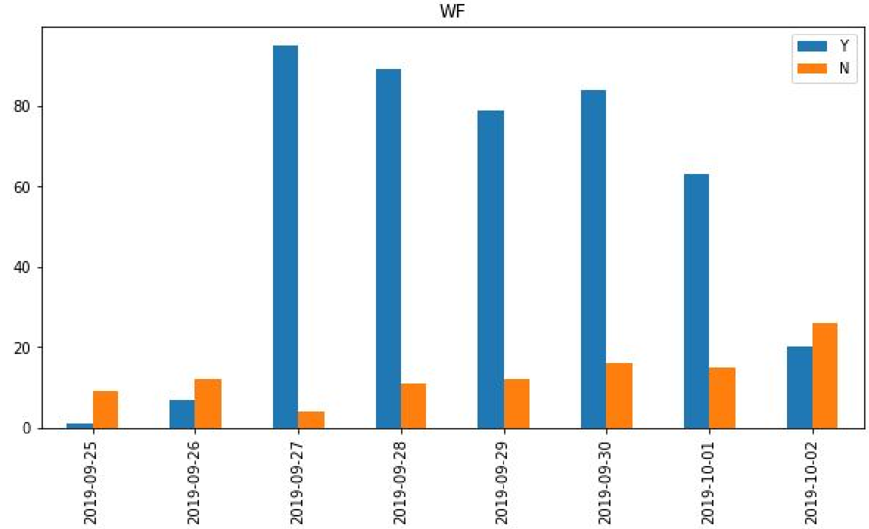
2.2. Remove duplicates from News.
News data has duplicates and have to be removed to avoid double-counting.
Duplicates can be identified by same url and timestamp.

Exploratory Data Analysis:

Wells Fargo

Wells Fargo announced the CEO on 9/27.
Below shows the number of tweets for each day. You can see that tweet activity increased starting 10/27. Similarly, there were some differences in number of not-relevant, there is sharp increase in relevant tweets starting from 9/27 and the number of relevant tweets don’t go below the number of not-relevant tweets until 10/2.

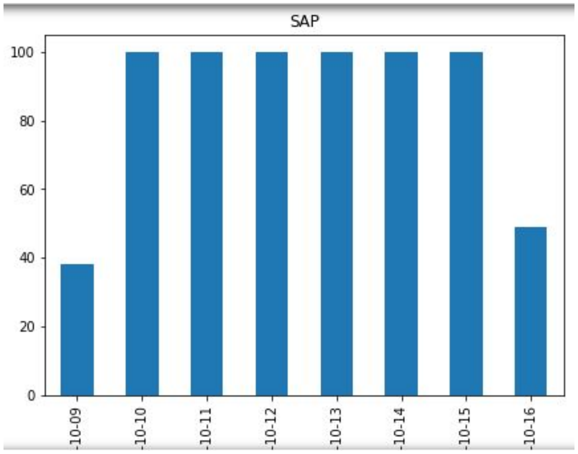
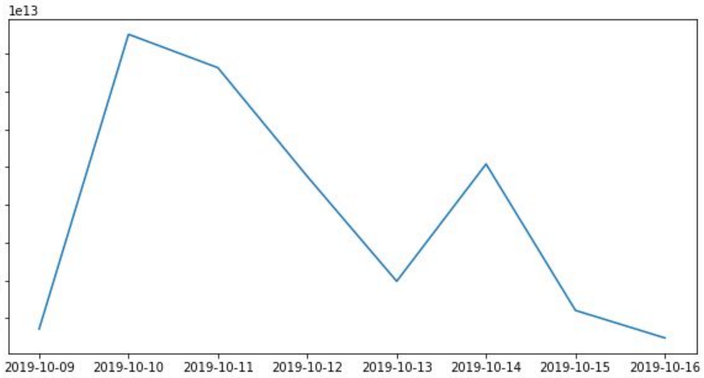


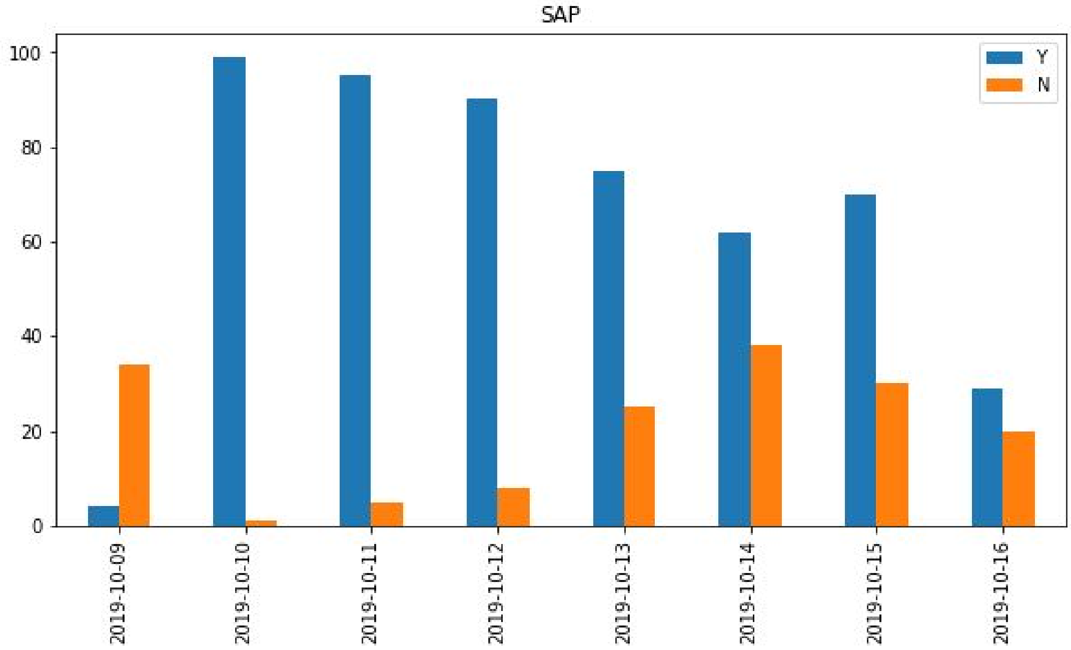


SAP

SAP announced the change in leadership on 10/11 in Europe which was 10/10 in U.S. Tweets started increasing on 10/10 and due to the large amount of activity, there were more than 100 tweets for the day. For the days where there were more than 100 tweets, 2nd chart shows the range of time collected for the day (inverted. The amount of time 100 tweets were generated was less on 10/10, the day of the announcement, and started falling in the next few days and down close to the amount of activity before the announcement on the 6th day after the announcement.

Similar to Wells Fargo, the number of relevant tweets compared to not-relevant tweets sharply grow the day of the announcement and slowly dies off towards the 6th day of the announcement.

	
# of tweets per day	1 day - range of time 100 tweets were collected for the day. (inverted)



# of tweets per day - divided by relevant (Y) and not relevant(N)

Overall approach for Sentiment analysis:

Using the sentiment score generated by Vader and TextBlob, gauge the direction of the stock price.

Steps:

1. Generate Sentiment score via Vader and TextBlob.
2. Apply weight to the sentiment scores by multiplying by how many times the tweet was favorited, quoted, replied to and retweeted.
3. Resample by bootstrap method to increase the size of the sample to 500.
4. Simple OLS linear regression with sentiment scores as predictor variables to gauge stock price trend. Stock price at the close is used.
5. Perform Bayesian Linear regression to predict sentiment scores for future days.
6. Compare the trend for sentiment scores against actual stock prices after the leadership change announcement.

Limitations:

1. Limited number of examples of leadership changes. Only two companies, Wells Fargo and SAP were used.
2. I was only able to get 100 tweets per search at a time using historical search.
   If there were more than 100 tweets a day matching the search criteria, search brings back the most recent tweets. I am also limited to how many free searches I can make with the free account I have with Tweet API.
   Due to this, I retrieved the last 100 tweets of the day for each company ending at mid-night of the day. Depending on how many tweets met the criteria, 100 tweets for the day can be for the whole day range or within the last 20 minutes of the day.
3. Even though search criteria was refined to get the most relevant tweets as possible, before the announcement of the leadership change, most tweets were not relevant, not even with the company itself, thus the number of tweets available for analysis was extremely small for number of days.
   In many cases, tweet content only comprises of link to an article without any other content making it impossible to know what the tweet was about thus contributing to less number of relevant articles.
   Following the link and analyzing the content of the article is not in scope of this project.
4. Tweet contents tend to be sarcastic or use jargons and in-jokes that sentiment analysis package were often not able to score correctly.
   For example, “NBA-style free agency comes to banking", “Wells Fargo Selects Charles Scharf To Make History As The First Female CEO Of Major U.S. Bank”.

5. There can be more than one issue that can affect the price of the stock (and the sentiment for/against a company) at a time. In this project, only one issue - change in CEO leadership is acknowledged. However, this one issue might not be enough to accurately and entirely describe the stock trend.
   For example, quarterly earning report was announced at the same time the CEO leadership change. Quarterly earnings report was exceptional which might explain why the stock price trend was positive while sentiment scores for SAP CEO trended slightly negatively.

Wells Fargo

Stock prices jumped by close of 09/27 after the announcement was made but the rally was short-lived and stock went down as much as 5-6% by 10/8. The stock price went back up again after 10/8 but that could be due to a different issue. Looking at 09/25 - 10/15/19, there is a slight downward trend (linear regression slope -0.02).

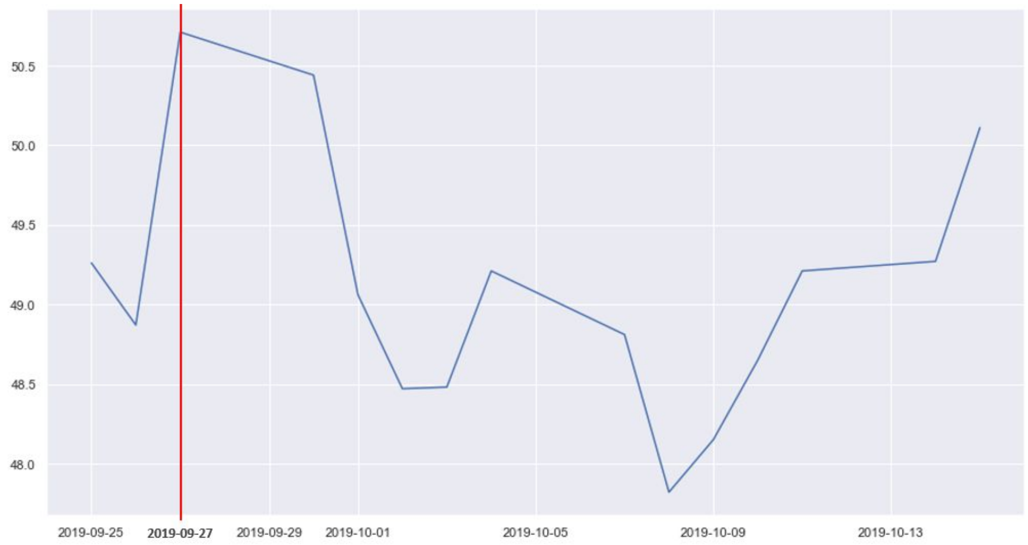
Sentiment scores are widely distributed within each day (09/25/19 - 10/2/19), both for Vader and Textblob sentiment scores but Vader score has slightly upward trend (linear regression slope: 0.06) while Textblob Polarity score has slightly downward trend (linear regression slope -0.01).

Although sentiment score trend for Vader package may look different from the stock price trend, if you look at the mean scores, the mean score jumps from -0.1 (9/25) and 0.29 (9/26) to 0.35 (9/27) and 0.47 (9/28) before going down starting 9/29. Trend is similar for Textblob Polarity score. Sentiment score trends are generally in line with stock prices the first 3-4 days of the announcement.

OLS Linear regression predicts downward trend for stock price, however Bayesian linear regression predicts upward trend for the stock price after 10/2. Actual was 48.48. (10/2 was 48.47). For 10/3 stock price at close, with Vader compound score, 49.18 was predicted while Textblob Polarity predicted around 50. 10/2 close price was 48.47 while 10/3 close was 48.48 so the trend was predicted correctly.

Stock:

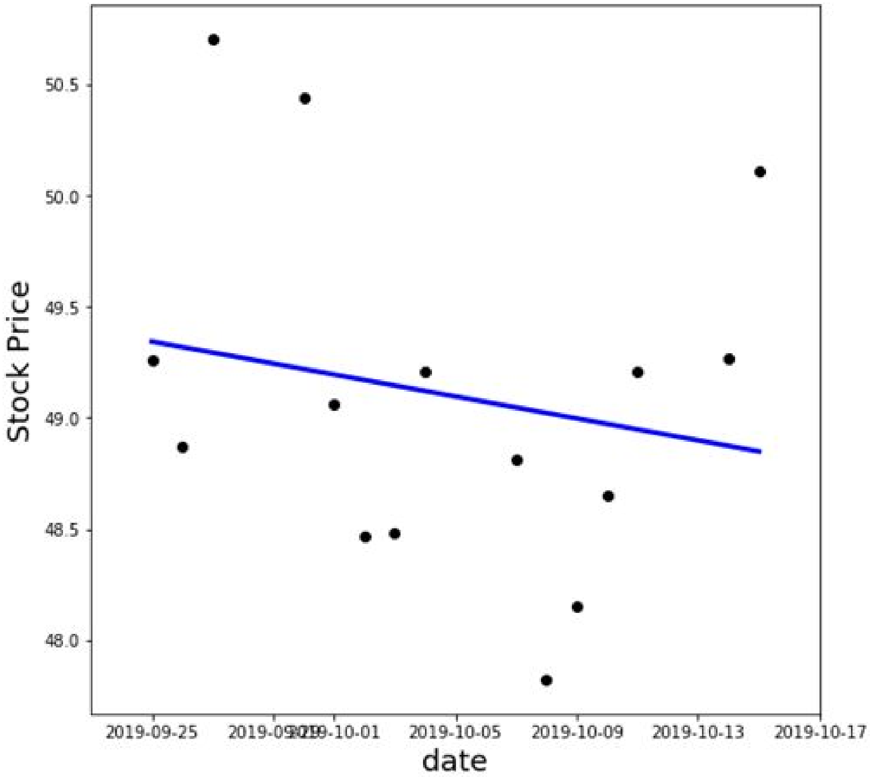
Announcement on 09/27/2019 (red line).



Simple regression line for stock price:

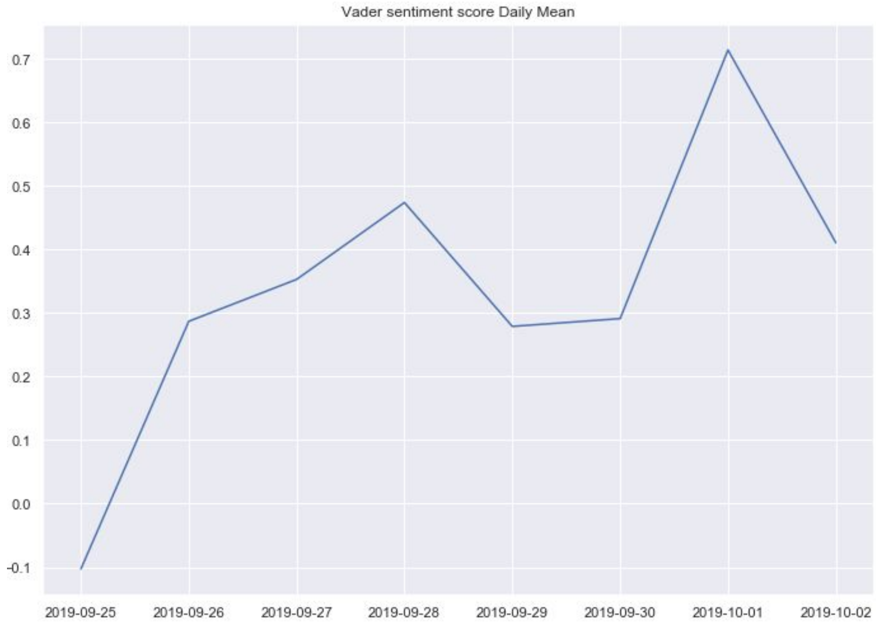
Intercept 49.29

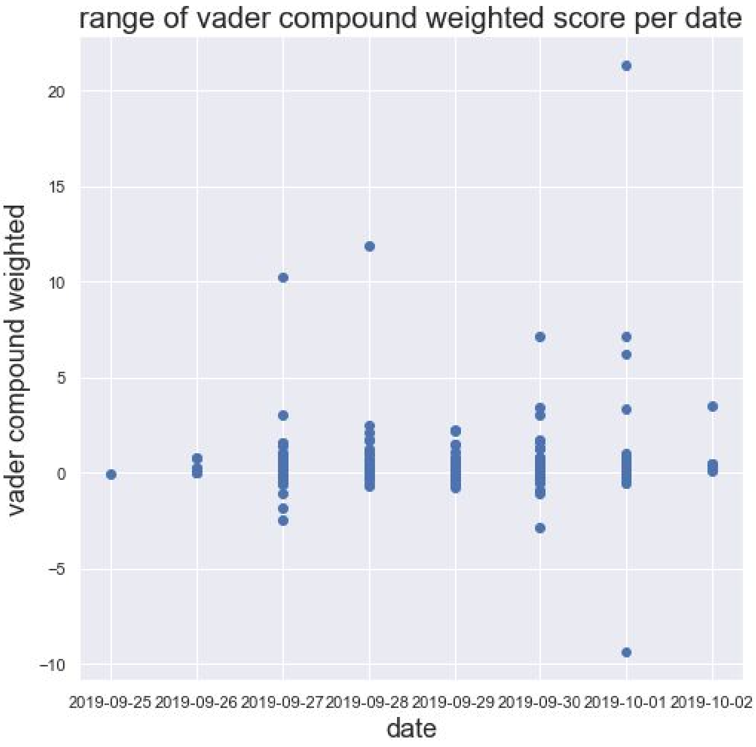
Slope: -0.0247



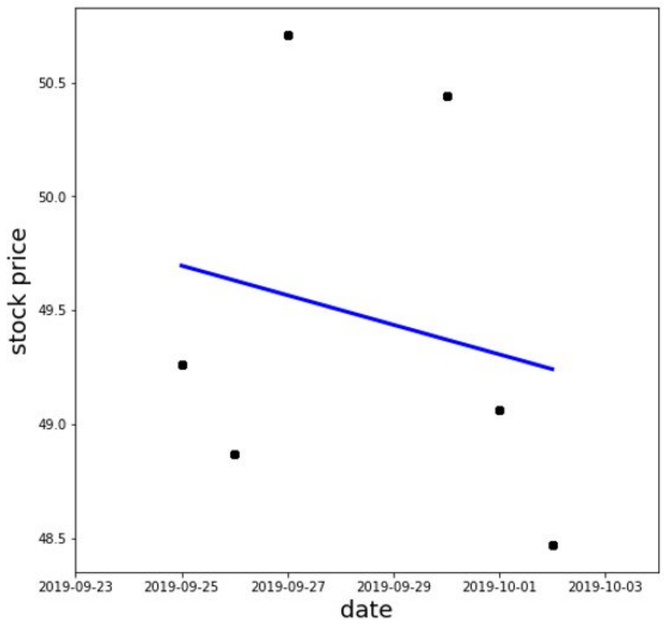
Tweet Sentiment analysis:

Vader Sentiment

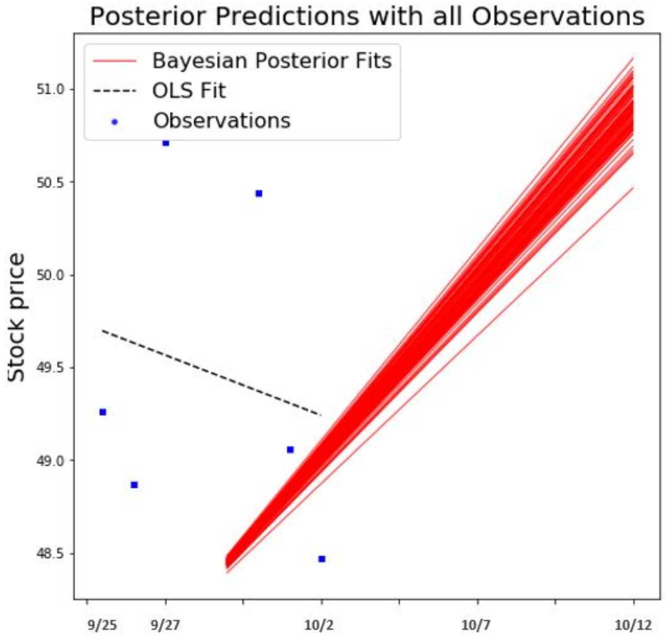




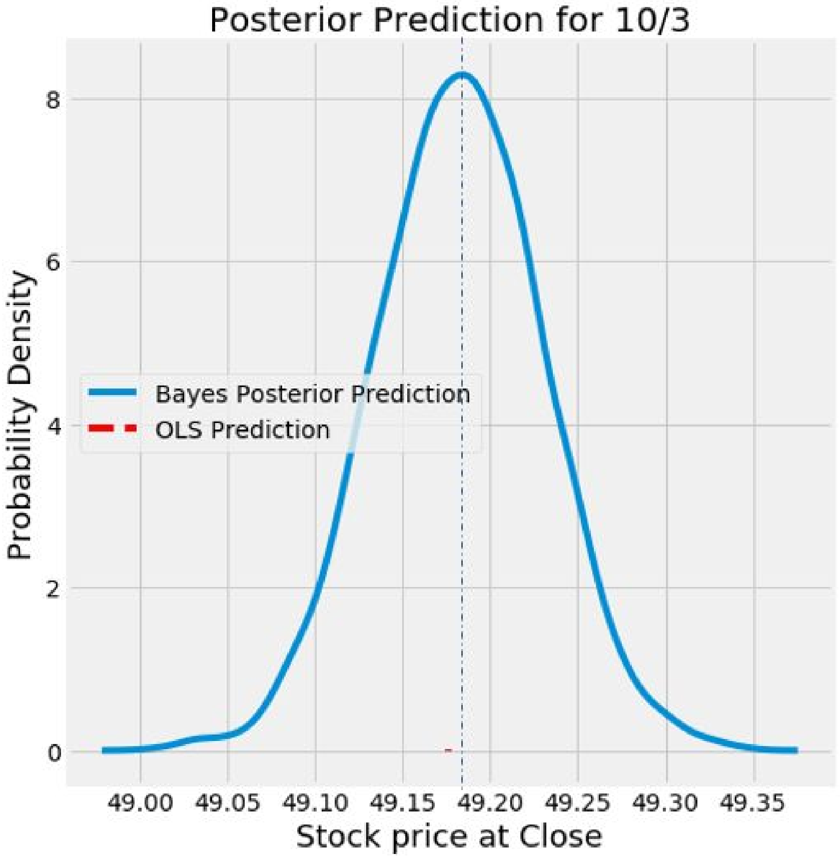
1) Ordinary Linear regression with resampling for stock price with weighted Vader compound score as predictor variable:
   Intercept : 49.57
   Slope: -0.0649



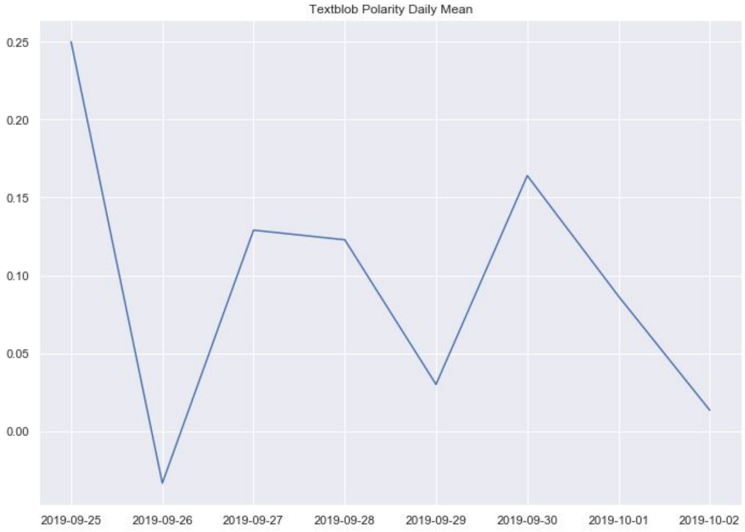
2) Baysian Linear regression with resampling

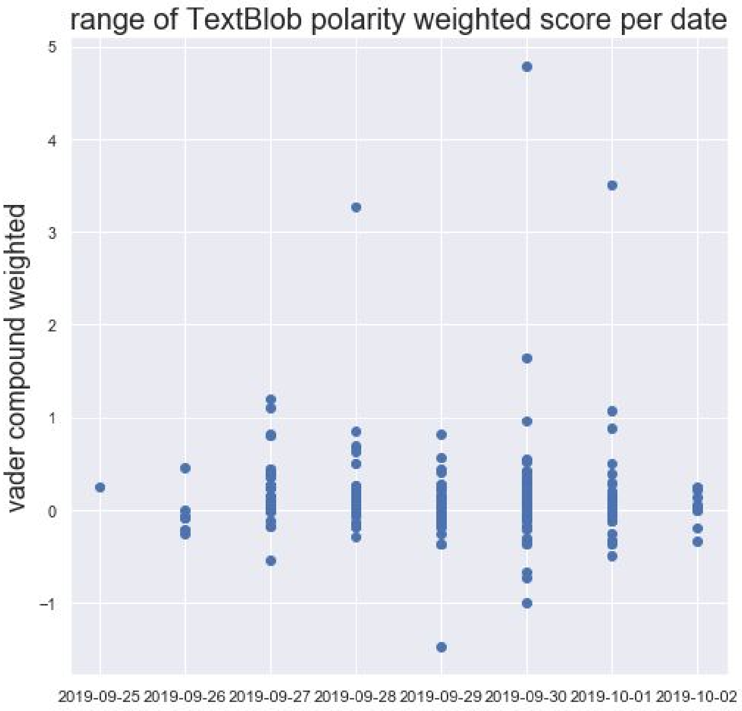


3) Posterior Predictiction for 10/3 stock price at close
   Predicted around 49.18. Actual was 48.48. (10/2 was 48.47).



TextBlob Polarity score

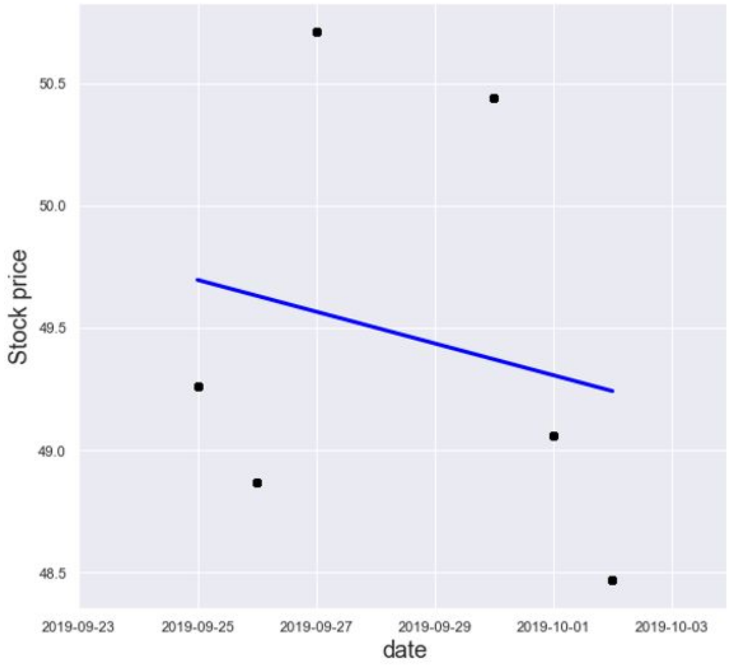




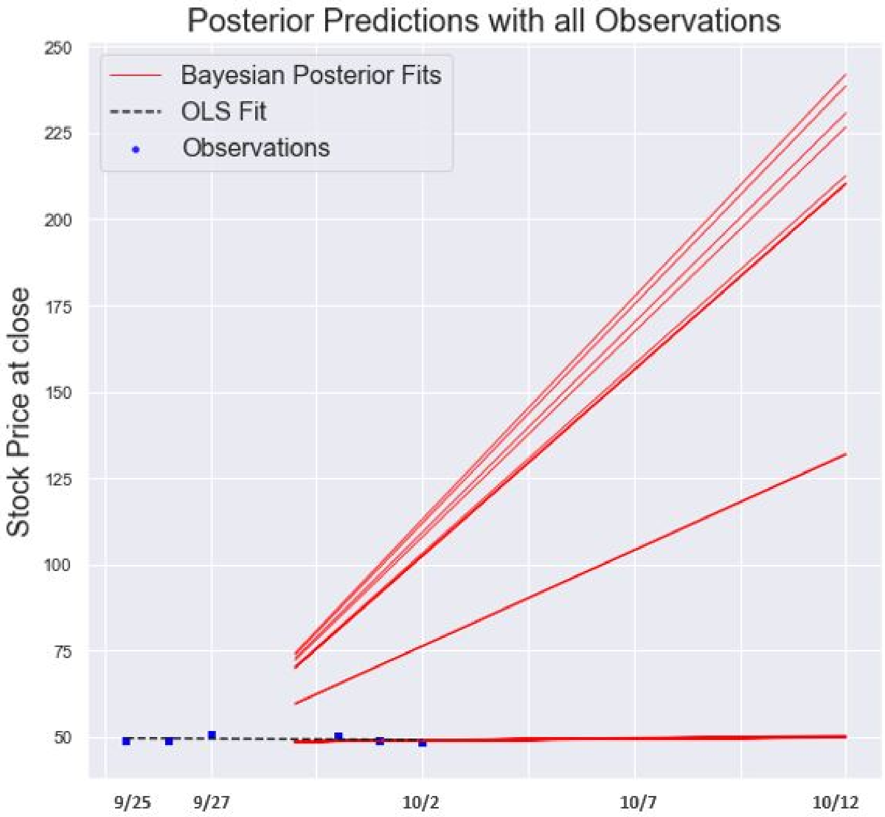
1) Ordinary Linear regression with resampling for stock price with weighted Textblob Polarity score as predictor variable::

   Intercept: 49.57

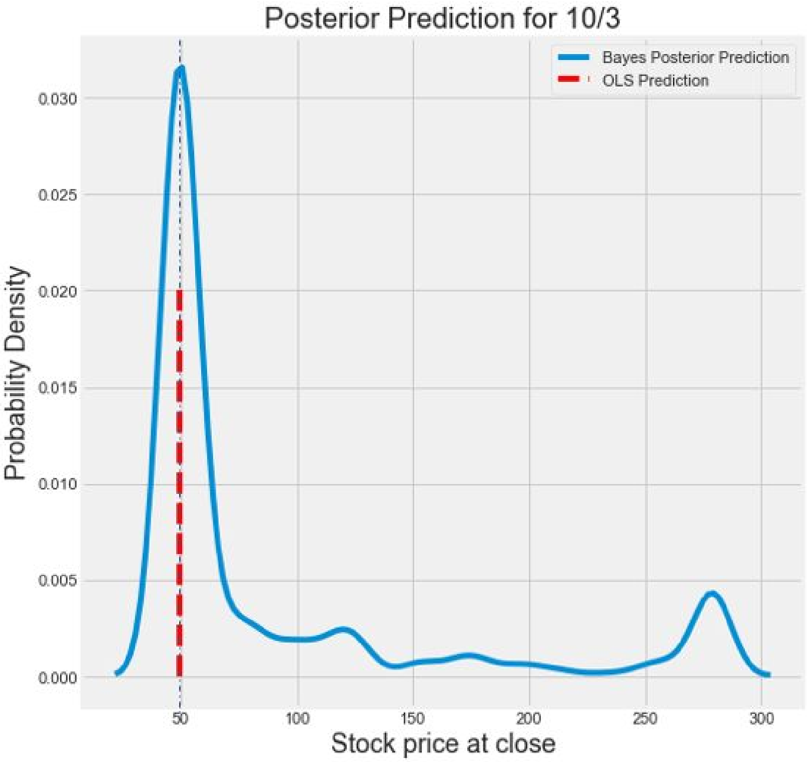
   Slope: -0.0649



2) Bayesian Linear regression



3) Posterior Predictiction for 10/3 stock price at close
   Predicted around 50. Actual was 48.48. (10/2 was 48.47).



SAP

Stock prices jumped by the close of 10/11 after the announcement was made and the stock price stays to trend positive (linear regression slope: 0.91). Even after the 10/11, the slope was 0.51.

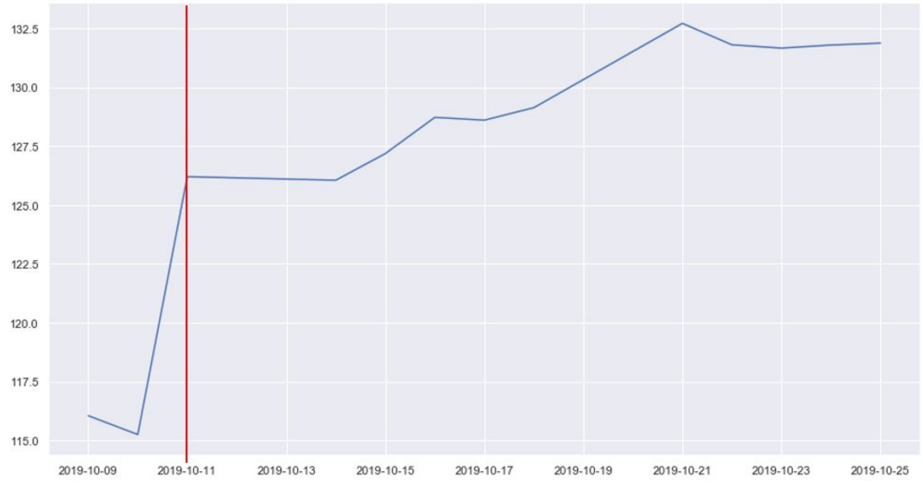
Sentiment scores are widely distributed within each day (10/09/19 - 10/16/19), both for Vader and Textblob sentiment scores and they both were slightly downward trend (linear regression slope: -0.10 for Vader and -0.025 for TextBlob.

None of the sentiment scores seem to be in line with stock price trend at least for the few days after the announcement. My take is that sentiment scores may trend downward because the CEO stepped down unexpectedly and the current CEO was considered quite effective and well respected. Still, the downward trend was very slight, considered neutral.

OLS Linear regression predicts upward trend for the stock price, and Bayesian linear regression also predicted upward trend for the stock price after 10/16. For 10/17 stock price at close, with Vader compound score as predictor variable, 131.26 was predicted while Textblob Polarity predicted around 130. 10/17 close price was 128.6. while 10/16 close was 128.72 so predicted price was higher but was not too different. Overall trend for the stock was up and the predicted value fits the overall trend.

Stock:

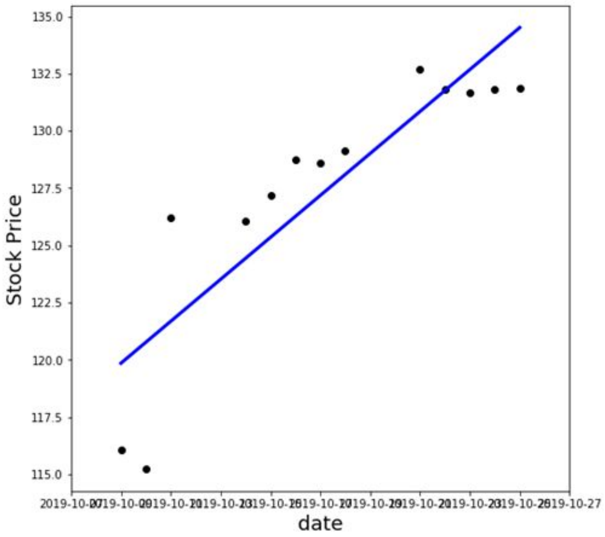
Announced 10/10/19 US time (10/11/19 in Europe) - red line.



Simple regression line:

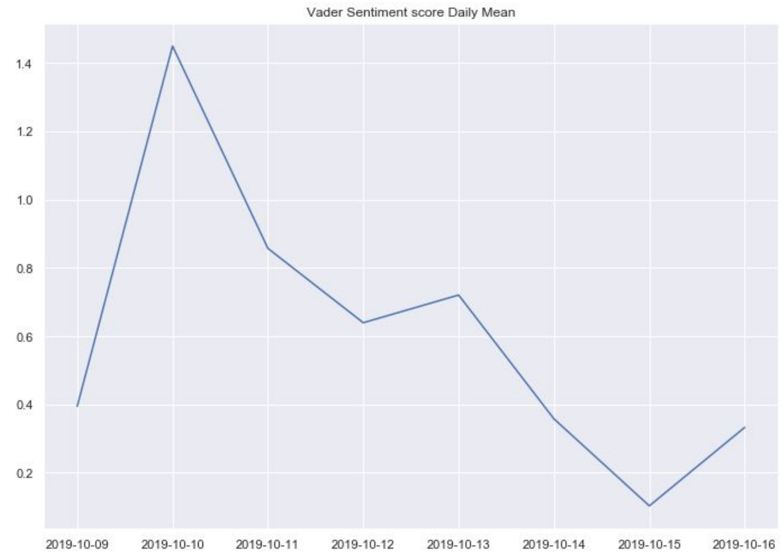
Intercept : 120.77

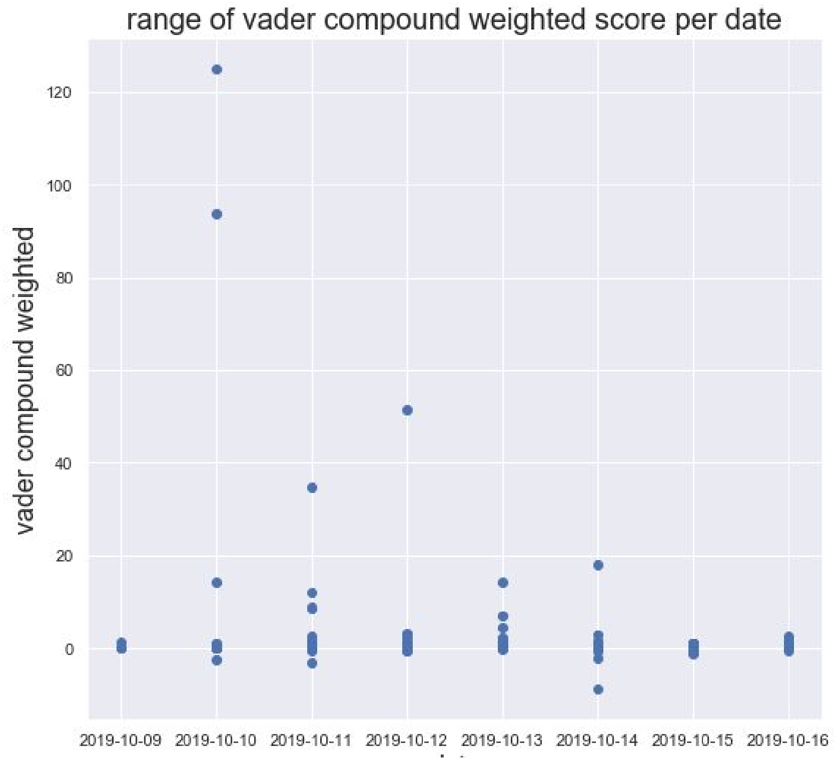
Slope: 0.9156


135.0
132.5
130.0
127.5
125.0
122.5
120.0
117.5
115.0
Stock Price
2019-10-07
2019-10-09
2019-10-11
2019-10-13
2019-10-15
2019-10-17
2019-10-19
2019-10-21
2019-10-23
2019-10-25
2019-10-27
date

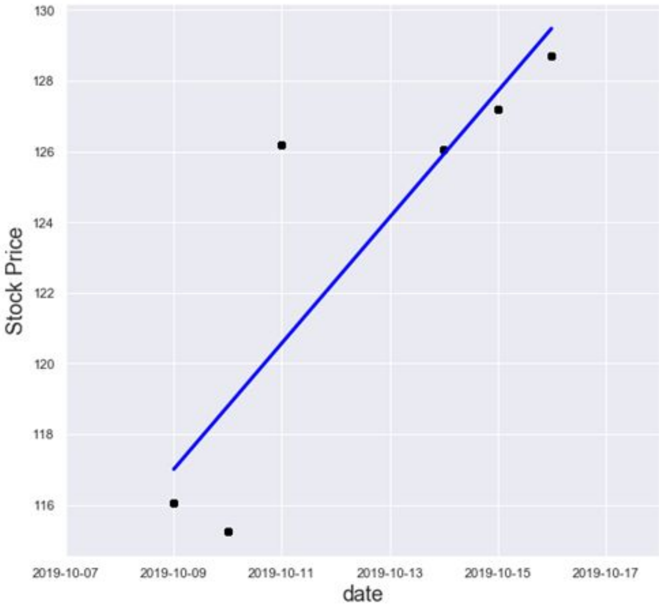
Tweet Sentiment analysis:

Vader Sentiment

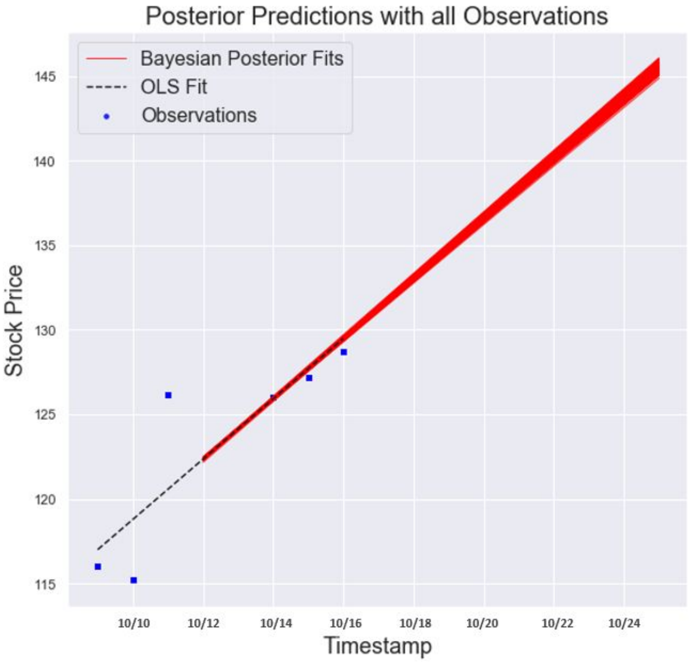

Vader Sentiment score Daily Mean


range of vader compound weighted score per date

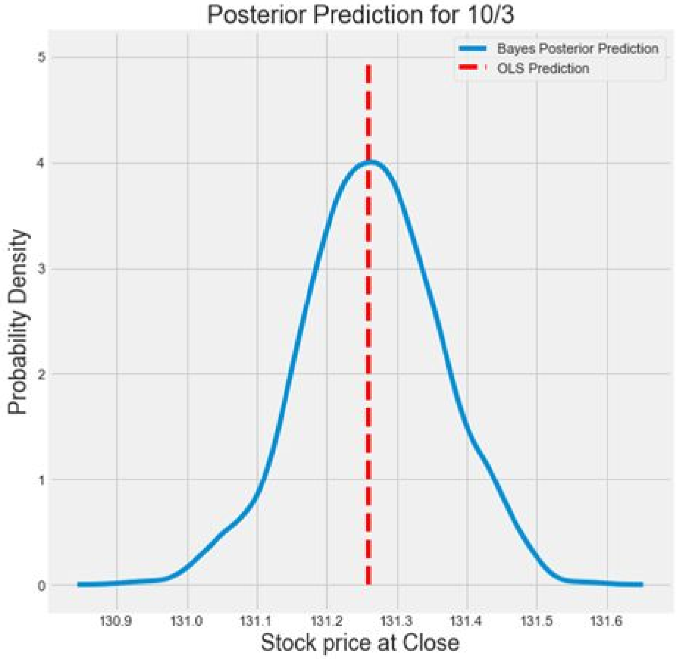
1) Ordinary Linear regression for stock price with weighted Vader compound score as predictor variable:
   Intercept: 118.79
   Slope: 1.7818



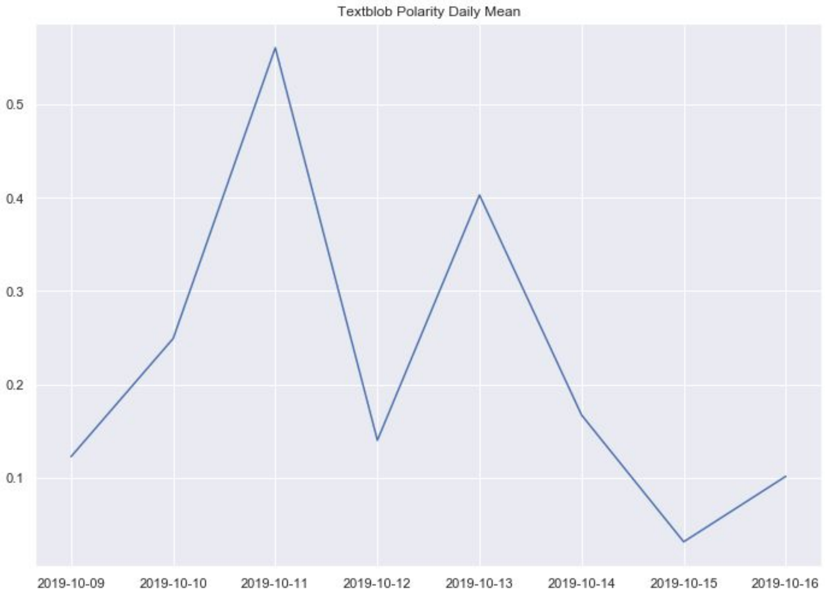
2) Baysian Linear regression

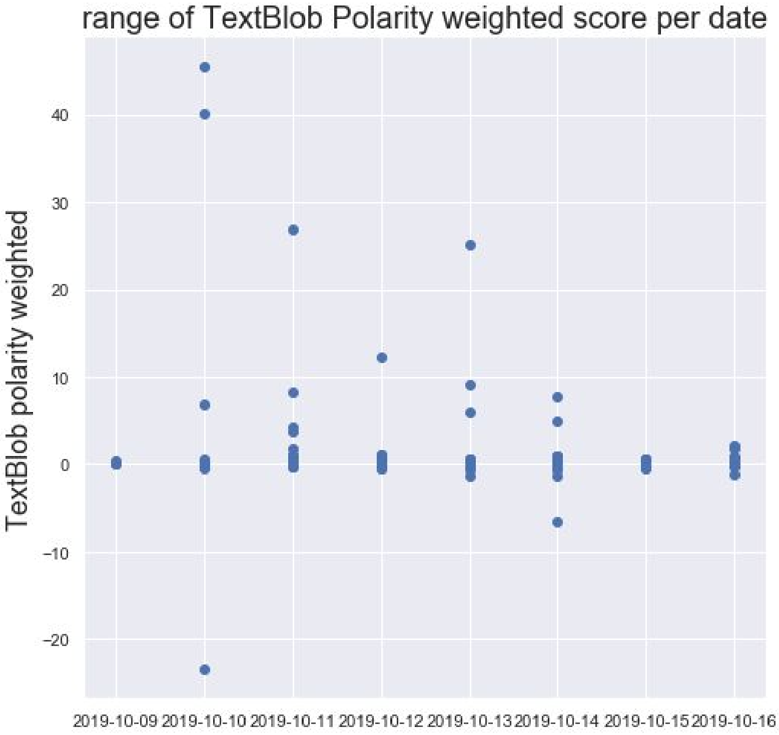


3) Posterior Predictiction for 10/17 stock price at close
   Predicted around 131.26. Actual was 128.6. (10/16 was 128.72)

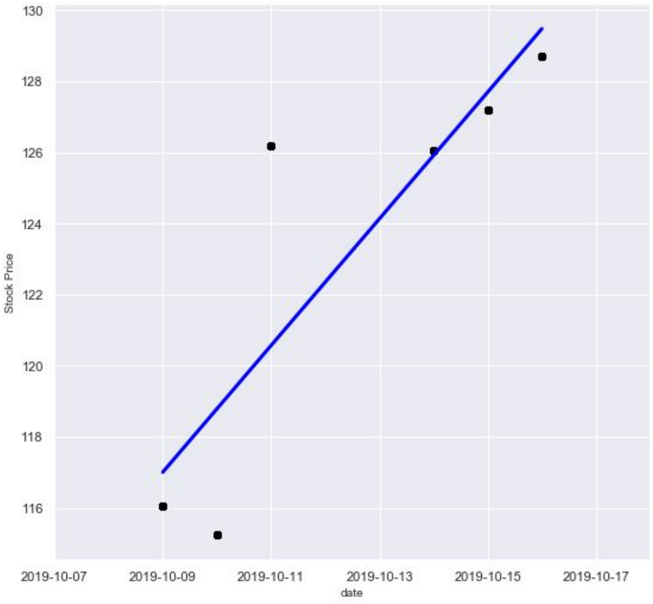


TextBlob Polarity

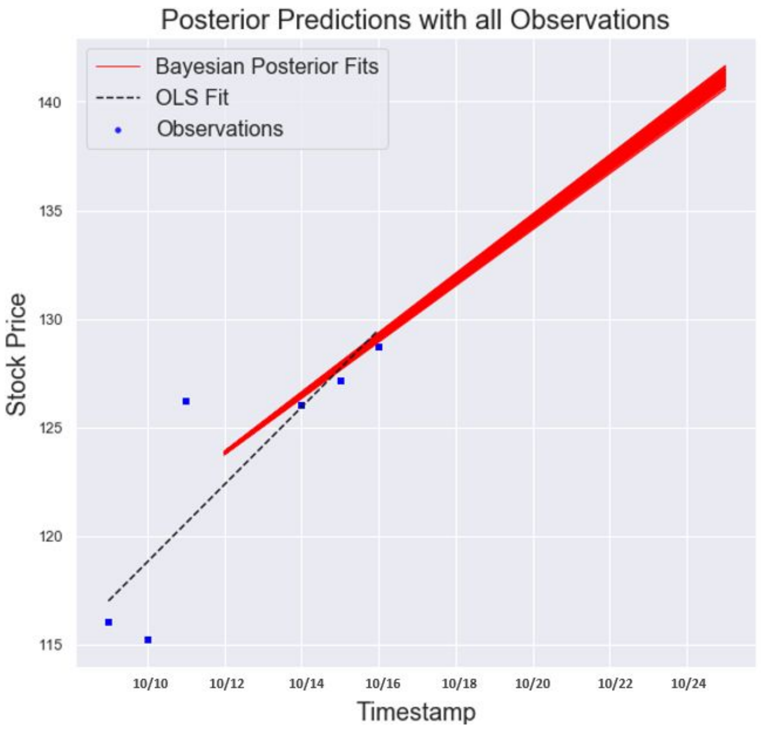




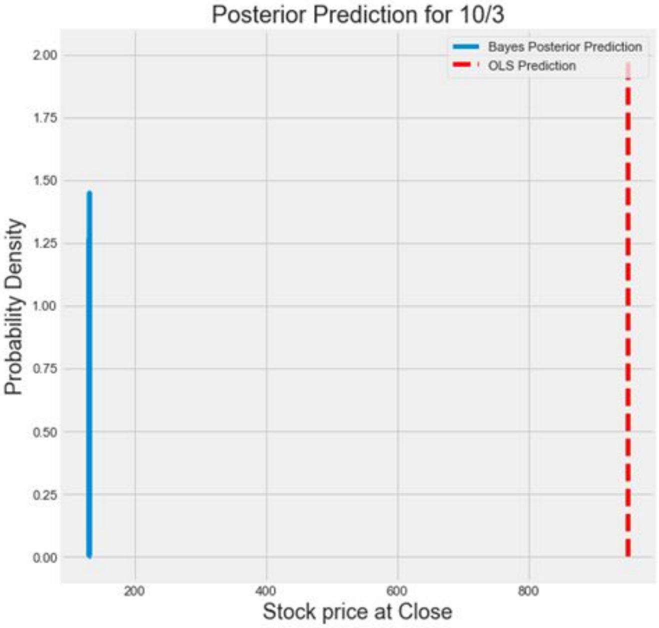
1) Ordinary Linear regression for stock price with weighted Textblob Polarity score as predictor variable:
Intercept: 118.78
Slope: 1.7818



2) Baysian Linear regression



3) Posterior Predictiction for 10/17 stock price at close
Predicted around 130. Actual was 128.6. (10/16 was 128.72)



Jupyter notebooks:

Data Gathering	● https://github.com/yulmee/springboard/blob/master/CapStone2/Get_Twitter_Feed_Full_Archive_WC.ipynb (Read data from api and store in text file) ● https://github.com/yulmee/springboard/blob/master/CapStone2/Store_twitter_data_in_Pymongo.ipynb (read data from text file and store in MongoDB)
Wrangling and clean up	● https://github.com/yulmee/springboard/blob/master/CapStone2/Wrangle_data_in_Pymongo.ipynb (read data from MongoDB and clean up data)
Sentiment Analysis	● https://github.com/yulmee/springboard/blob/master/CapStone2/Sentiment_Analysis_Twitter_Vader.ipynb ● https://github.com/yulmee/springboard/blob/master/CapStone2/Sentiment_Analysis_Twitter_TextBlob.ipynb
EDA	● https://github.com/yulmee/springboard/blob/master/CapStone2/Stock_Prices.ipynb ● https://github.com/yulmee/springboard/blob/master/CapStone2/Explore_data_in_Pymongo_Twitter.ipynb
Modeling	● https://github.com/yulmee/springboard/blob/master/CapStone2/Sentiment_Analysis_Twitter_Explore_Stock_and_Twitter.ipynb ● https://github.com/yulmee/springboard/blob/master/CapStone2/Sentiment_Analysis_Twitter_Explore_Stock_and_Twitter.ipynb

Data Files

Twitter files	https://github.com/yulmee/springboard/tree/master/CapStone2/Data_Twitter

Stock Price	https://github.com/yulmee/springboard/tree/master/CapStone2/Stock_Price
CSV files for intermediate data	https://github.com/yulmee/springboard/tree/master/CapStone2/Data_CSV