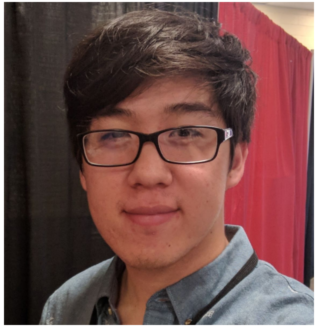
Twitter and it's impact on NASDAQ

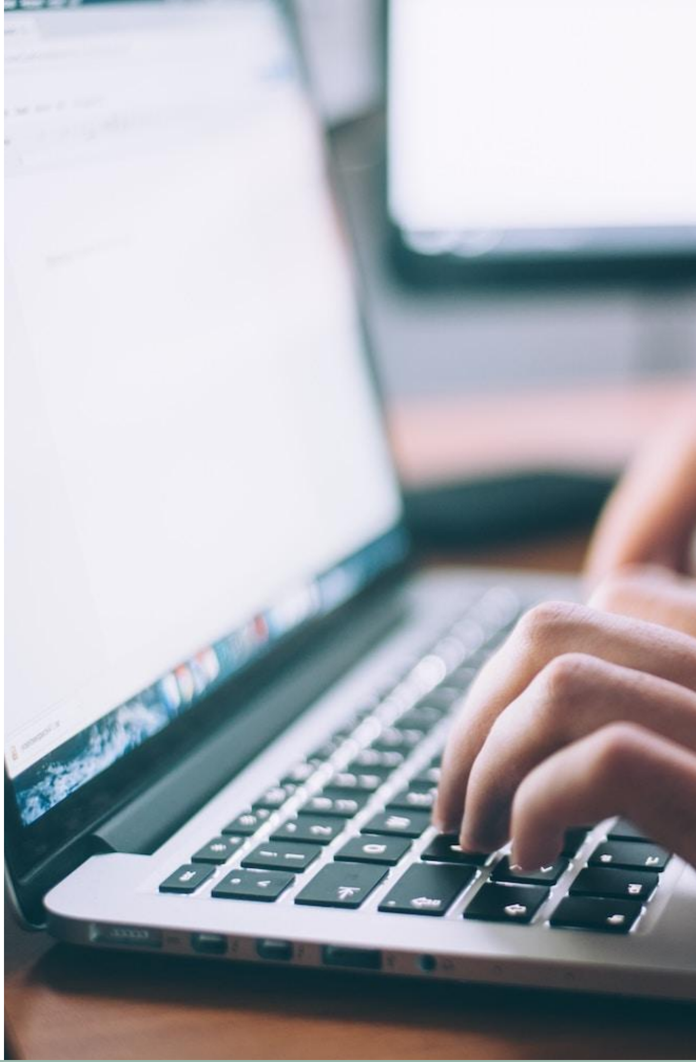
Tara Flynn, Sam Choi, Peter Chen, Sumed

Project Summary and Selection Journey

Project Summary

Objective:

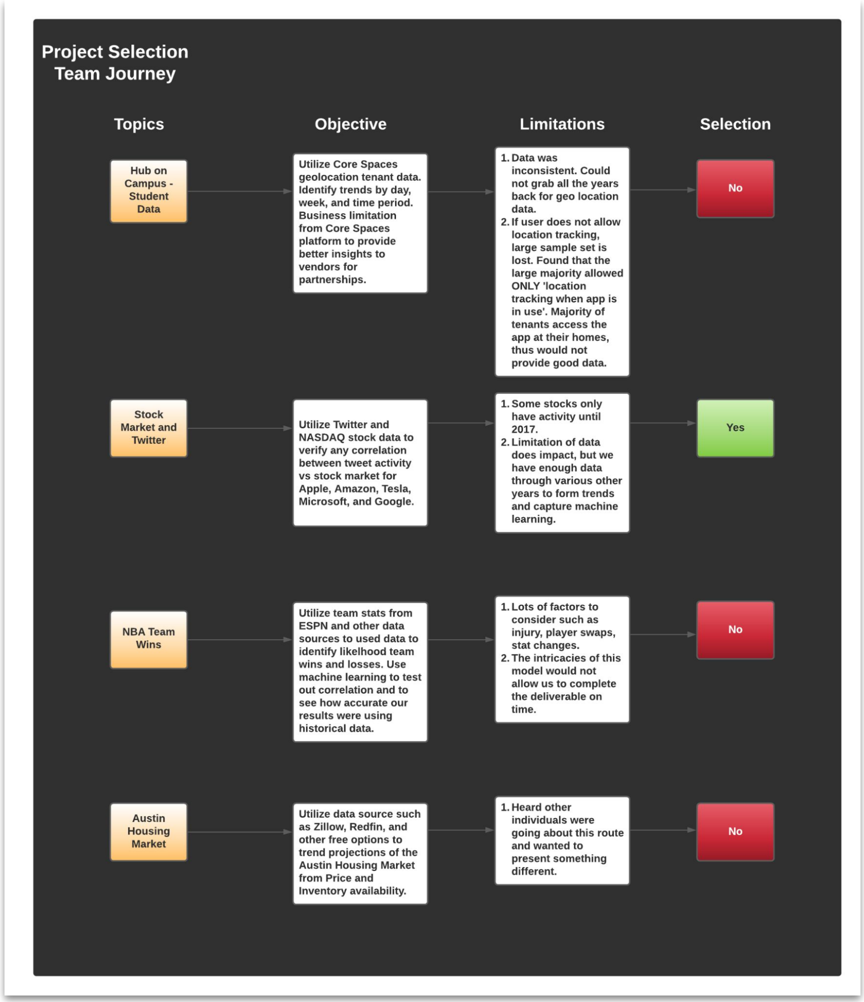
- Identifying correlation between Stocks and Tweets.
- We are using the top 5 NASDAQ stocks (Apple, Amazon, Tesla, Microsoft, and Google)
- Does Twitter activity affect opening and closing prices in the market?
- Does Twitter activity predict stock liquidity?
- Identify correlation and utilize machine learning to test our our hypothesis and replicate accuracy and precision through machine learning models.



Project Selection Journey

Objective: Choosing the Topic

- We researched 4 different topics from various different sectors of industry.
- The final selection, '**Stock Market and Tweets'** was selected after reviewing objective, needed technology, and limitations from the available data sets.
- For easier viewing of the journey model, [CLICK HERE](#).

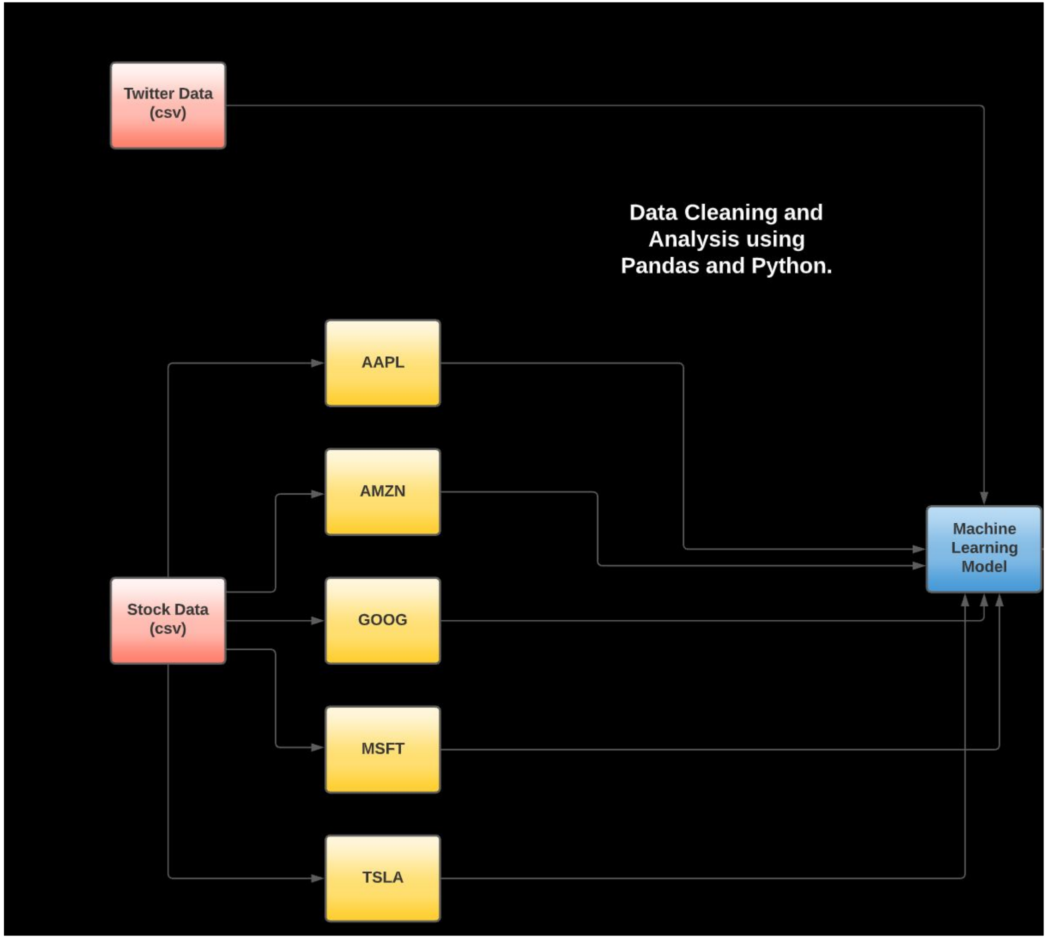


Data Source

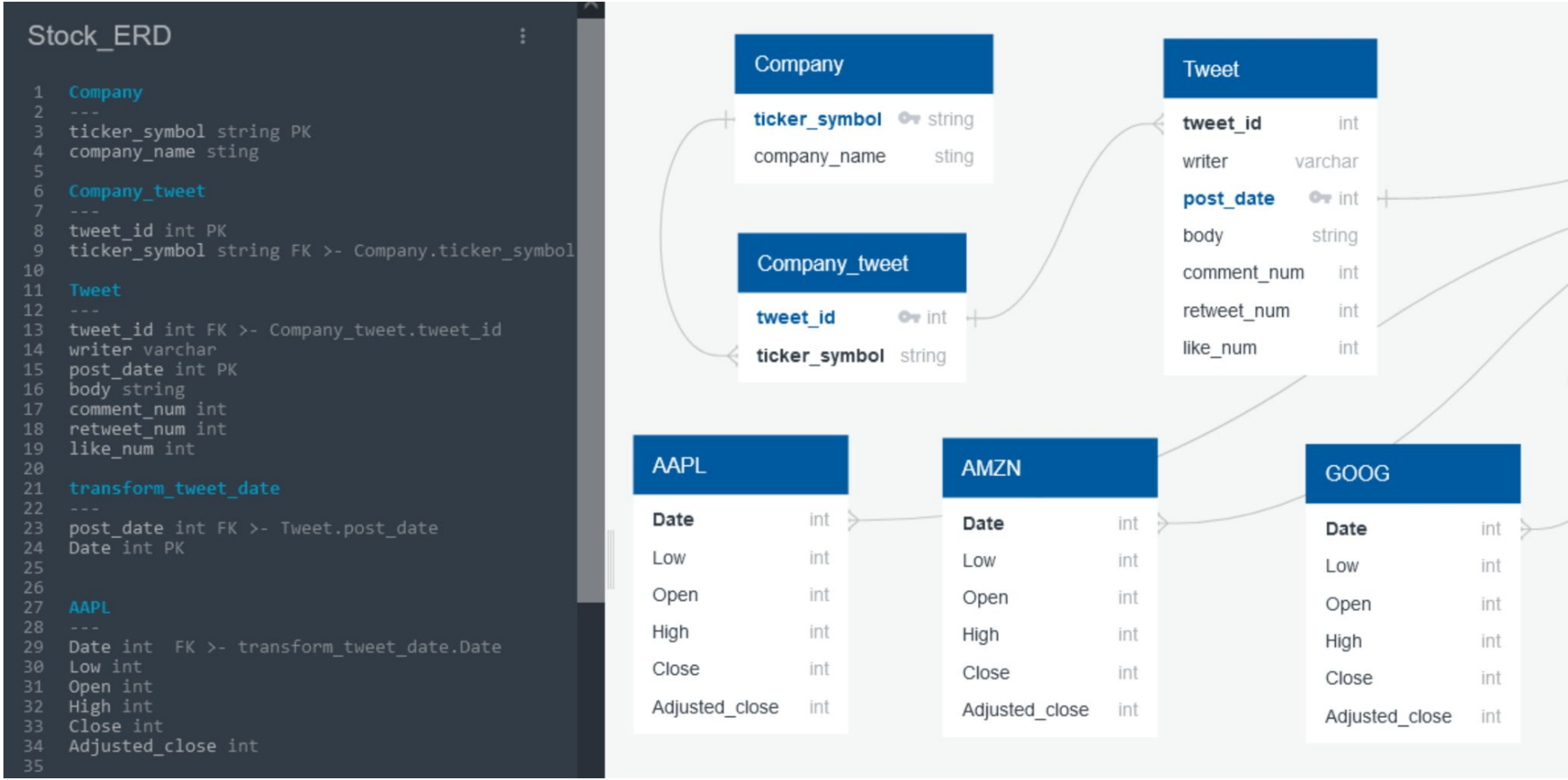
Data Source

Source:

- Data for Stocks (Amazon, Apple, Microsoft, Tesla, and Google) were obtained in csv format through kaggle.
- Data for Tweets through 2015 - 2019 were obtained in csv format through Kaggle.



Data Source: ERD



Discovery, Data Exploration, and Data Analysis

Discovery: Questions to Answer

- What level of correlation and predictability can be gathered to test out our hypothesis that tweets impact stocks in terms of prices or stock volume transactions?

- Which primary and foreign keys can be utilized to JOIN these data sets together for use?

- Are there key points within the year where stock prices and volume transactions go up naturally with or without influence from twitter end users and influencers?

- Can this be tested through supervised learning to replicate results from SKLearn train-test-split?

- If so, what is the best model to use for machine learning that will not over fit or under fit?

What can I solve/answer with this data?

Discovery: Data Exploration and Analysis

- The team uploaded csvs (stocks and tweets) into Amazon S3 and connected to PgAdmin Postgres. S3 was also connected to a Jupyter Notebook database for use with Python and Pandas.

- Stocks and Tweets were joined based on stock company ticker through python and postgres.

- We identified null data sets from Tweets and Stocks and dropped any ‘N/A’ and ‘Null Values’.

- Dates were formatted for acceptable use for postgres and python work.

Discovery: Data Exploration and Analysis pt 2

- Calculated fields were created for price action from the delta of opening and closing stock prices. These were compared with tweet volume counts tied to each day.

- All stock dataframes were consolidated into 1 master table to capture all information for use of data analysis.

- Data went through pre-viz using matplotlib pyplot to understand distribution of daily percent price changes. Identified the mean as well as all quartile ranges for each stock.

- Final-viz using Tableau illustrations.

- Supervised machine learning model using Pandas and SKLearn libraries
  - Variables were price action and tweet counts.
  - Correlation was found on stock liquidity/volume quantity exchange.
  - No strong correlation between price action vs tweet activity.

Regression Model and Machine Learning

Data Analysis: Regression Correlation (Apple)

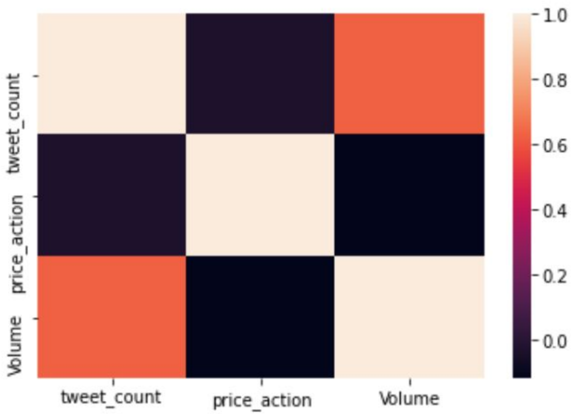


Figure 1: Heat map tweet, price action, volume

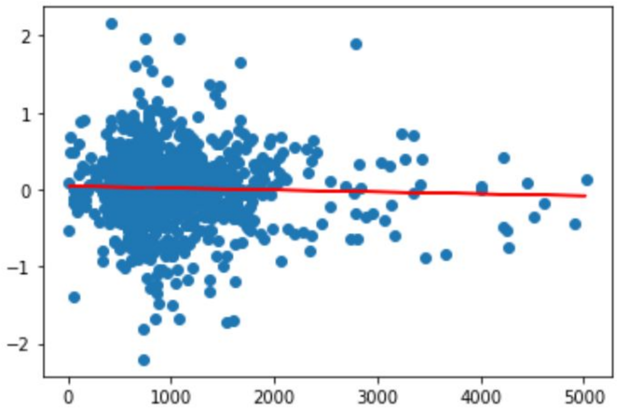


Figure 2: Tweet count and price action relationship

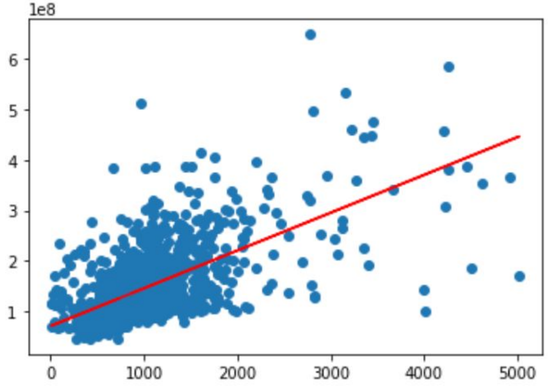


Figure 3: Tweet count vs Volume Transaction

Data Analysis: Regression Correlation (Amazon)

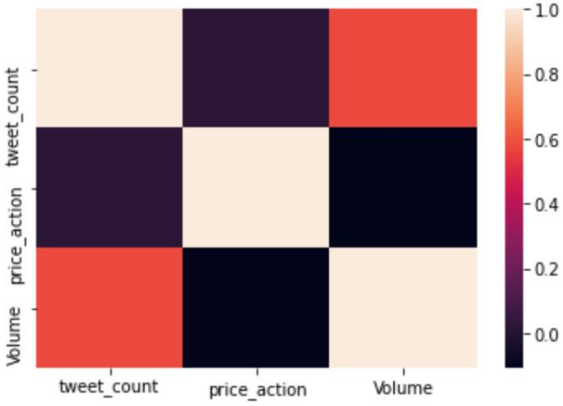


Figure 1: Heat map tweet, price action, volume

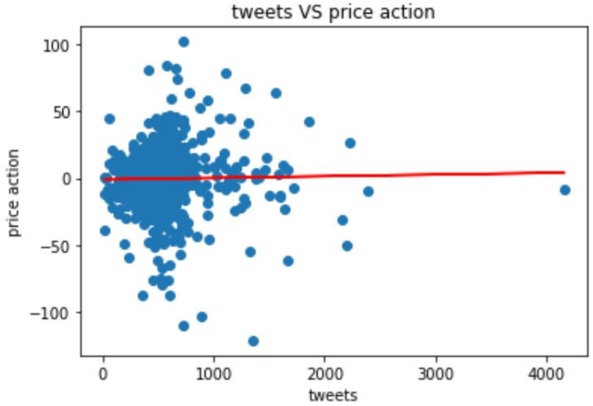


Figure 2: Tweet count and price action relationship

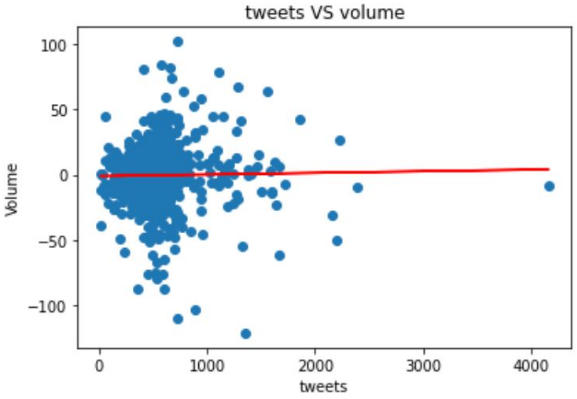


Figure 3: Tweet count vs Volume Transaction

Data Analysis: Regression Correlation (Google)

Figure 1: Heat map tweet, price action, volume

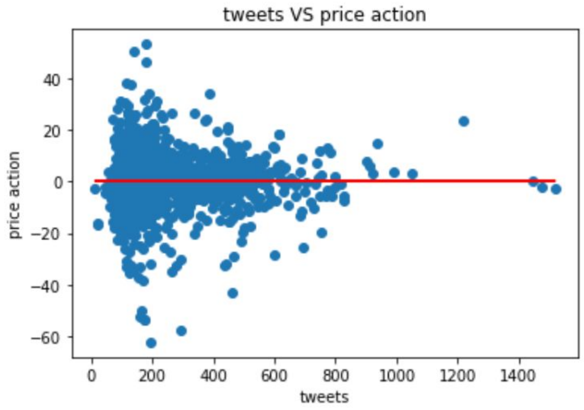


Figure 2: Tweet count and price action relationship

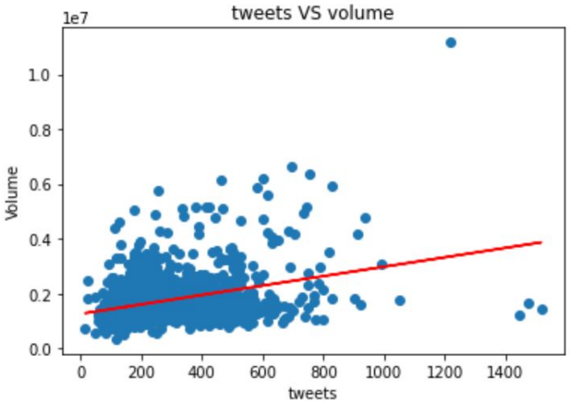


Figure 3: Tweet count vs Volume Transaction

Data Analysis: Regression Correlation (Microsoft)

Figure 1: Heat map tweet, price action, volume

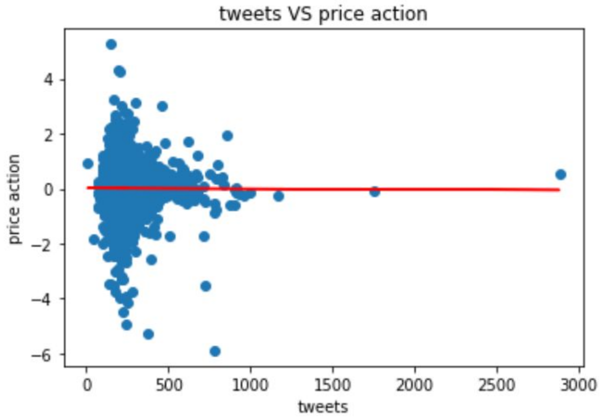


Figure 2: Tweet count and price action relationship

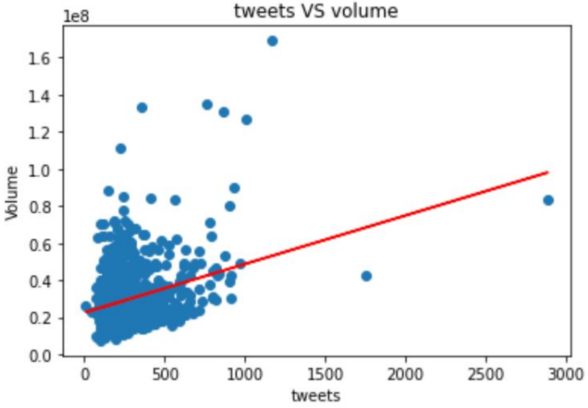
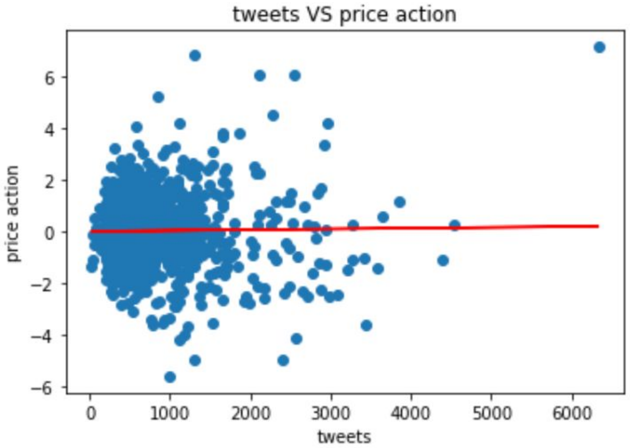


Figure 3: Tweet count vs Volume Transaction

Data Analysis: Regression Correlation (Tesla)



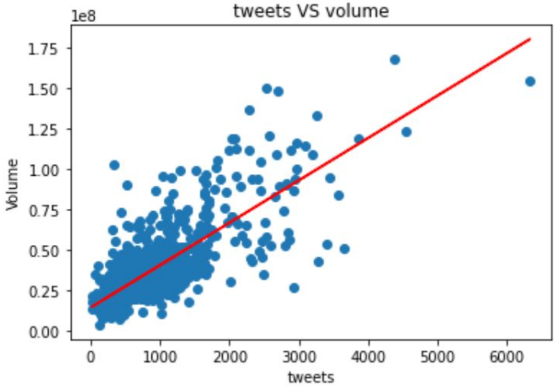


Figure 1: Heat map tweet, price action, volume

Figure 2: Tweet count and price action relationship

Figure 3: Tweet count vs Volume Transaction

Machine Learning

Utilized SKLearn libraries:

- Train_test_split
- StandardScaler
- Logistic Regression
- Max Iterations - 200
- Accuracy: 57.46%

Tensor Flow:

- Train_test_split
- Standard Scaler
- 1st Hidden Layer Activation: Relu
- Output Layer Activation: Sigmoid
- Max Iteration - 100

```
from sklearn.metrics import accuracy_score
print(accuracy_score(y_test, y_pred))
```

```
0.5746031746031746
```

```
Model: "sequential_3"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_6 (Dense)              (None, 6)                 24
_________________________________________________________________
dense_7 (Dense)              (None, 1)                 7
=================================================================
Total params: 31
Trainable params: 31
Non-trainable params: 0
```

Machine Learning: pt 2

Tensor Flow (Continued):

- 97.46% Accuracy
- 9.51% Loss

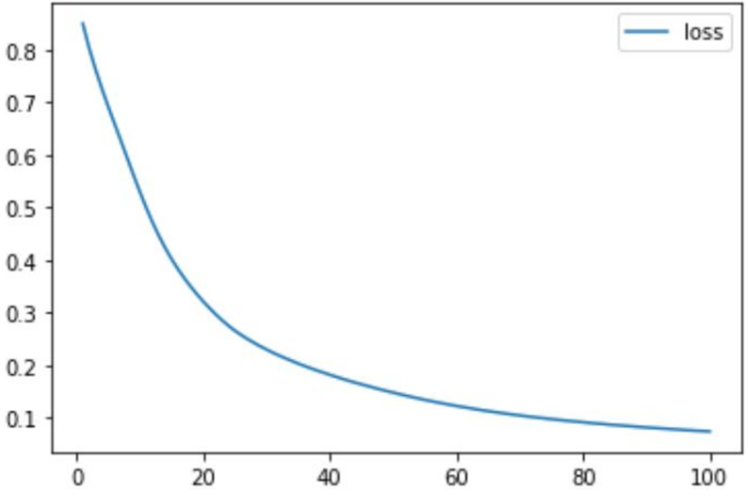


Figure 1: Loss over epoch

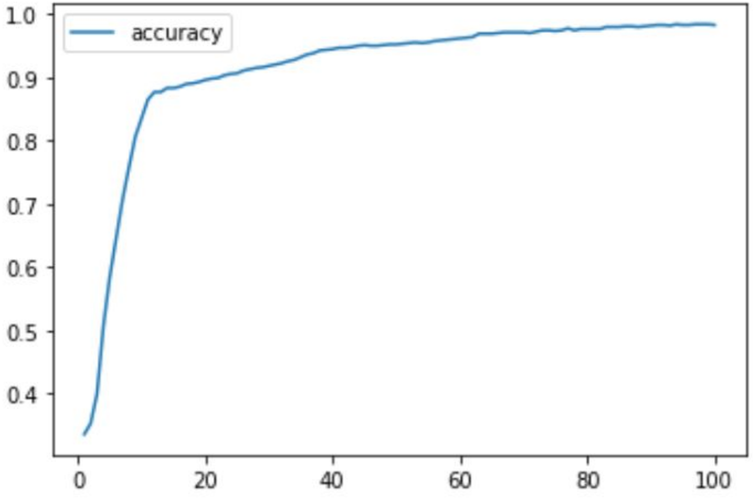
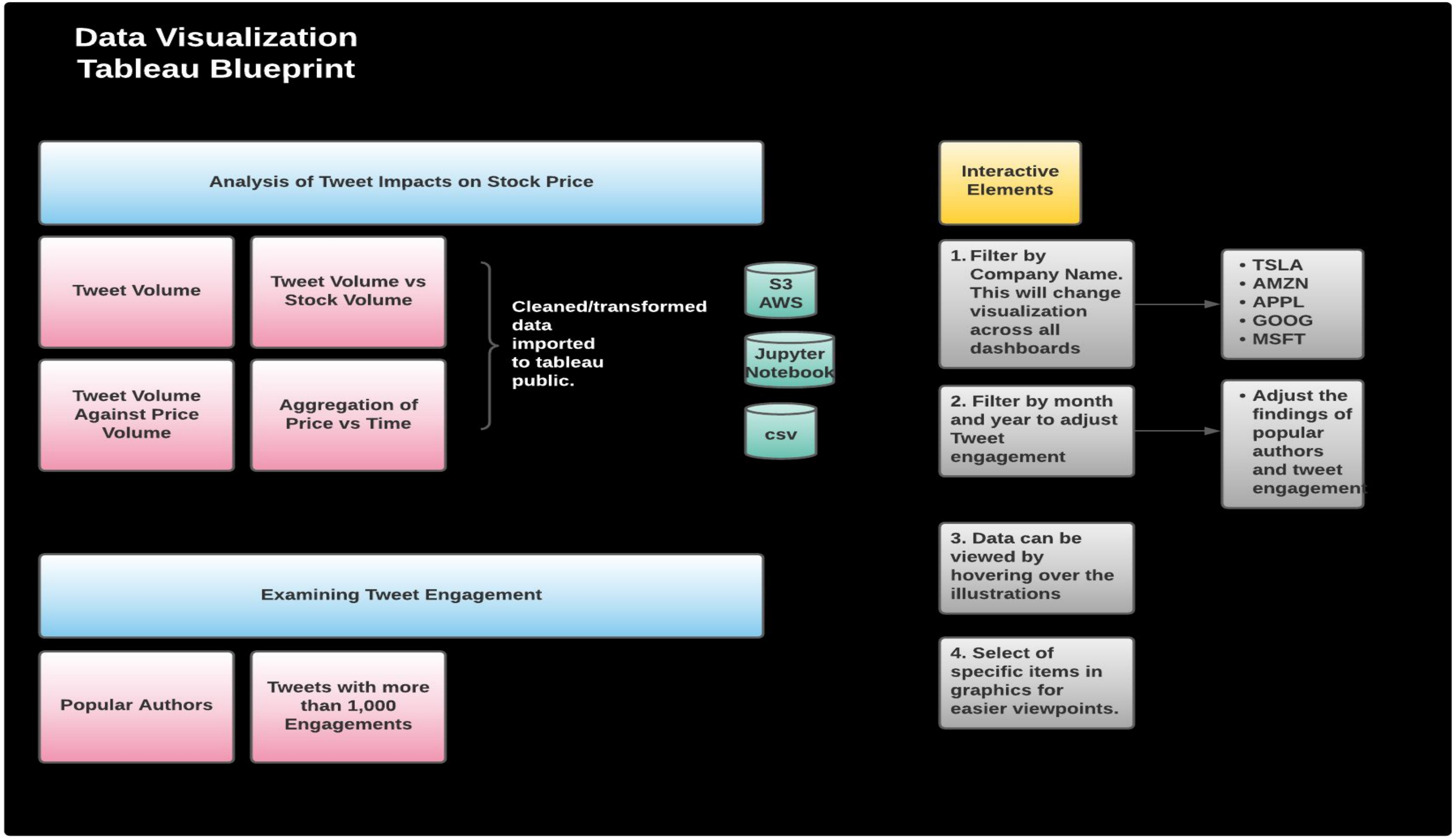


Figure 2: Accuracy over epoch

Tableau Storyboard: Blue Prints → Construction

Data Visualization: Tableau Planning



Data Analysis: Tableau Visualization

- **Link to Tableau Public Dashboard** - CLICK HERE

- **Analysis of Tweet Impacts on Stock Prices:**
    - Tweet Volume
    - Aggregate Price Action Over Time
    - Tweet Volume vs Stock Volume
    - Tweet Volume vs Price Volume

- **Tweet Engagement:**
    - Popular Authors
    - Tweets with more than 1,000+ engagement

Technology Used

Technology Used

- **Data Cleaning and Analysis:**
  - Excel and Panda were used to clean the data and perform an exploratory analysis. Python was used for further drill down analysis and data manipulation.

- **Database Storage:**
  - Amazon RDS, PgAdmin, and S3 buckets were used to store our raw and cleaned csv data.

- **Machine Learning:**
  - SKLearn was used as our Machine Learning Library.
    - Logistic Regression
    - Standard Scaler Train-Test-Split (75:25)
    - Tensor Flow: Train-Test-Split (75:25)
      - 6 nodes; Relu model (first hidden layer); Sigmoid (output layer)

- **Dashboards:**
  - Tableau was used for visualization presentation on all of our finds and ML predictions.

Thank you!