Data Wrangling Report

Gathering:

In order to obtain the data for this project, I need to gather data from three sources- from a file on hand csv from the Udacity servers and from the twitter API. For file on hand, I needed to only use pd.read_csv(). To download the file from Udacity servers, I had to use the Python requests library, and write the data to a file. Then I used pd. read_csv() to open the tsv document. The third step required me to use the Twitter API. I used tweepy and was able to store all the data in a json format in a text document. Also, I stored the retweet and favorite counts features in a csv and loaded it in a pandas dataframe called df. During the gathering process, I also decided to drop the old tweets from the twitter archive csv. This is because my analysis was largely going to deal with the retweet and analysis features, and I thought that ensuring that there were no missing values for these variables would lead to more accurate conclusions.

Asses:

I then defined the issues in the dataset that I wanted to clean. I did this through visualization and code.

The problems to clean were: 1. Columns in reply status id and in reply to user_id, along with retweeted status have to many nans. 2. There are Retweets in the twitter archive and retweet count/favorite count Dataset. 3. There are urls in tweet text. 4. The timestamp is not in datetime. 5. There are numerators whose ratings are puns, and not actual ratings. Ratings with decimal values may be wrong 6. Some of the Expanded Urls are duplicated and some are not from twitter. 7. Some names are incorrect. The values that are lowercase are not names. 8. None is not the proper way to describe missing values.

The problems to tidy were to join the data I obtained using the Twitter API, and to join the dog t=stages columns into one column. I decided not to combine the predictions data frame with the tweets data, as this is a different observational unit and thus should be represented by a new table.

Clean:

I then went through the process of using a combination of python code, pandas, and regular expressions, to solve the above problems. I would first implement the code and then implement a check to see whether it worked.

Store:

Because, I had two final data frames, I decided to store my final data in a SQL light database.