Data Wrangling Internal Report
WeRateDog Project

This project aims to analyze tweets from WeRateDogs account with data collected from different sources. The data include a CSV file with selected Twitter post from November 2015 to August 2017. Additionally, I programmatically downloaded a TSV file from Udacity's network with information about the dog breeds using a face recognition algorithm. And I collected information from the Twitter API (retweet and favorite counts) to be able to have the needed information for the analysis. After a quick glance at the data, I noticed that it was not tidy and it had many quality issues that need to be addressed.

The following tidiness and quality issues were identified:

Tidiness:

1. Merge retweet and favorite counts with the tweet data in a common dataframe
2. Dog stages should be only one column
3. Merge image prediction data into the same dataframe

Quality:

1. Remove tweets that were not available in the Twitter API
2. Remove retweets based on the `retweeted_status_id`
3. Remove tweet's reply based on the `in_reply_to_status_id`
4. Remove unneeded columns
5. Dog stages data type should be categorical
6. Timestamp should be Date/Time data type
7. Change `rating_numerator` to float
8. Remove no dogs related tweets and tweets without score
9. Fix score for several tweets
10. Adjust names for several tweets
11. Calculate an overall rating score by dividing `rating_numerator/rating_denominator`

I started with the tidiness issues. Initially, I cleaned the Twitter data by melting the 4 different columns for the dog stages into one. Then I combined the image recognition data by choosing the algorithm with the highest probability and I also added the retweet and favorite count data into the master dataframe.

Then I addressed the quality issues. I removed those tweets that did not have retweet and favorite counts since we are not able to get those values with the information provided. Then I removed all the retweets and tweets replies; those were identified based on the retweet's status ID or in reply status ID since "original" tweets do not have fields. Following that, I adjusted some data types and then I fixed ratings since they were either a date (like 7/20) or the decimal portion of a float number (75/10 instead of 9.75/10). I also adjusted some of the dog stages since some entries had 2 stages and only one was accurate. To be able to compared the do`g's scores, I added a new column to the dataframe "rating_score" which includes the score fractional values and it was calculated by dividing the rating numerator over the rating denominator.

Finally, the clean data was stored in a master CSV file (twitter_archive_master.csv) and in a SQL database (twitter_archive_master.db) to be used for visualization and insights.