WRANGLE REPORT FOR TWITTER DATASET:

The project explore and wrangle twitter dataset was quite challenging and very interesting.

The dataset used is the tweet archive of Twitter user @dog_rates, also known as WeRateDogs. WeRateDogs is a Twitter account that rates people's dogs with a humorous comment about the dog. WeRateDogs downloaded their Twitter archive and sent it to Udacity via email exclusively for this project.

Basically, we need to gather the data from 3 different sources:

1. From the twitter archive we got from Udacity. (tweet data)
2. From image_predictions.tsv file (image predictions of dogs)
3. From the Twitter API (each tweet's JSON data)

Let's look at them one-by-one in detail.

TWITTER ARCHIVE:

This archive contains basic tweet data (tweet ID, timestamp, text, etc.) for all 5000+ of their tweets as they stood on August 1, 2017. We also got the data filtered for tweets with ratings only (2356 tweets). It was easy to read the given csv file using pandas read_csv function and storing it in a dataframe.

IMAGE PREDICTIONS:

Every image in the WeRateDogs Twitter archive was run through a neural network that can classify breeds of dogs. The given URL was downloaded programmatically using the Requests library. The results of image predictions are stored in a tsv file. We can read the same using read_csv function.

TWITTER API:

This was the toughest part of the project. I had spent weeks together trying to learn from every website available. Finally I found two ways to get the result:

One approach was to get the status of each tweet using the tweet id and filter only the required fields and store it in a list, which can eventually be converted into a dataframe. This dataframe will have the tweets json data. It can be dumped into a text file.

Second one is after getting the status, dumping all the information into a file and then reading the file line by line and storing them in a dataframe. Filtering the necessary columns comes next.

The latter was the method specified in the project instructions. So I used the second method and was successful.

ASSESS AND CLEAN:

After gathering the data into 3 data frames, I proceeded to assess and clean them. Prior to that I made copies of the data frames.

First I looked for null values and missing values which were predominant in twitter archive dataframe. The data type mismatch is another quality issue I was looking for. In archive dataframe the tweet id has to be string and the timestamp must be a datetime object. On doing the visual assessment, I was able to find that many dog names are either misspelt or null.

To clean that, I extracted the dog names from the text attribute and created a new column for dog names. Instead of having many columns for dog stages, I condensed them into one, to make it tidy. The rating numerators and denominators did not match with the ratings displayed in the text column. So, I extracted the correct ones from the text and used it for the analysis.

Likewise in images dataframe, there were many images not recognized as dogs. As they are not required for analysis, I removed them. There are 3 different algorithms which predict the breed of the dog in the image. Thus there are 3 different prediction, 3 different confidence values. Whichever algorithm predicts that it is a dog and has more confidence value was quite right in predicting the breed. Depending on these values, I condensed them into one column.

It was mentioned in the project details that, we only need original tweets and not retweets. Hence I cleaned the retweets from all the data frames. Twitter archive table has the retweeted_status_id attribute. Similarly, there is retweeted_status attribute for twitter API dataframe. Retweets can be distinguished from typical Tweets by the existence of a retweeted_status attribute.

Finally after removing the unwanted columns from each table, we get the clean data frames. They can now be merged to get a single dataframe with all necessary columns which will be used for analysis.