Wrangle Report

For this project we wrangled the tweet archive of Twitter user 'WeRateDogs'. WeRateDogs is a Twitter account that rates people's dogs with a humorous comment about the dog. It is extremely popular with millions of followers on twitter. We gathered the data from three different sources, assessed for quality and tidiness issues and then cleaned it and merged all the data from different sources into one master dataframe.

Data Wrangling involves three important steps.

1. Gathering Data from different sources
2. Assessing the data for data quality and tidiness issues
3. Cleaning the dirty data

Gathering Data

We gathered data from three different sources:

1. Twitter Archive

This file had all the tweet archives of the 'WeRateDogs' account. The csv file was downloaded and moved to the Project Jupyter Notebook. It was then imported into a dataframe called 'twitter_archive.

2. Image Predictions

This is a tsv file containing image predictions by a neural network about the breed of dogs based on each dog's picture from the tweet. This file was hosted on Udacity's servers and was downloaded programmatically using the Requests library from the following URL: https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predictions/image-predictions.tsv The Udacity server was sent the request and then the response was written onto a tsv file which was then imported into dataframe using a separator ('/t')

3. Twitter API

In order to access the Twitter API I needed tokens and secret keys. For this I had to create a Developer's Account with Twitter to access these tokens. Using these access tokens and secret keys data was extracted from Twitter API using Python's Tweepy library.

Each tweet's retweet count,favorite ("like") count and follower count was extracted. Using the tweet IDs in the WeRateDogs Twitter archive I queried the Twitter API for each tweet's JSON data using Python's Tweepy library and store each tweet's entire set of JSON data in a file called tweet_json.txt file. Each tweet's JSON data was written to its own line then I read this .txt file line by line into a pandas Dataframe

Assessing Data

First I increased the column width to 200 so that it was easy to read all the texts of the tweets. I then visually assessed each file to see how the data looks. I also programmatically assessed the data using the .info() method. The following issues were identified:

- checked for duplicates; none found
- dog classification had a lot of missing data which was not null type
- a lot of dogs had a numerator greater than 10. I checked the images of these dogs and found that they were mostly dogs in group.
- unusual or weird numerators were also studied. They were mostly for ratings which were in float or for several dogs in a group.
- weird names which are not names but English words like 'a' 'an','the','this,'not','very
- tweet id was an integer instead of a string
- image prediction file had non intuitive names, which were corrected.
- some dog breed data was not capitalized
- The dog category had four columns which were melted to form one column.
- All the three datasets were merged on 'tweet id' to form one master dataset.

Cleaning Data

A copy of all data was made before starting the cleaning operations. the following issues were cleaned

- The first cleaning operation was to keep only the original tweets and do away with all retweets and remove the columns relating to retweets like 'in_reply_to_user_id', 'in_reply_to_status_id', 'retweeted_status_id', 'retweeted_status_user_id', 'retweeted_status_timestamp'
- all the datatype conversions were done e.g. timestamp was changed to datetime, tweet id to string, ratings to float
- non null values in dog category was made null values
- The ratings, both numerator and denominator were fixed manually and programmatically. The ratings were converted to float datatype, and manually corrected ratings where it was incorrectly recorded. The records where there were no ratings were deleted.
- A new column called 'ratings' was created(rating=rating_numerator/rating_denominator)
- The tidiness issues were also addressed e.g. the four columns of doggo,floofer,pupper and puppo were melted into one columns called 'dog_stage'
- All the three datasets were merged into one big dataset called 'master_df'

After cleaning the data was exported to a csv file