DATA GATHERING

There were 3 main datasets needed for this project. Two of them were provided: the twitter archive dataset for the WeRateDogs account and the image prediction dataset. These were provided as flat files (CSV) and could be easily accessed.

The final dataset was to be extracted from Twitter using their API endpoints. Using a tool called Tweepy, we were able to connect to the Twitter API after signing up and download the Tweet data for the needed IDs. There were some errors here as 16 IDs did not return any values. Tweets saved in a file called “Tweet_json.txt”.

DATA WRANGLING AND CLEANING

Twitter Archive Dataset.
This was one of the given datasets. It contained 2356 rows (tweets) with 16 columns. The tweet ids were unique which was a good sign. The following issues were discovered and treated as discussed below:

- Some columns used the string value “None” instead of NaN for missing data.
  This can cause issues e.g. a count of the values in such column would include those string values. This was handled by converting to values to NaNs.

- Timstamp values were coded as strings.
  It is always better to use the Datetime object of Pandas for all date related variables as they allow us handle them with more accuracy and provide more functionality. This issue was handled by converting those string timestamps to Datetime objects.

- Removing Retweets as stated in Project directive.
  One of the instructions was not to deal with Retweets but only original tweets. We handled this issue by removing all the retweets in the dataset. There were a total of 181 retweets. Bringing this dataset down to 2175 rows.

- Standardizing the Ratings column.
  Although it was stated the ratings could have numerators higher than the denominators, it was noticed that most of the tweets had denominators of 10 apart form a rogue few. Exploration of the Tweet text also shows that this is what the account owners expected and rating denominators not equal to 10 were erroneous. We handled this issue by getting rid of such observations. There were 22 such rows, bringing the dataset down to 2153.

- Some dog names were wrong.
  Assessment of the dog names column revealed that some Dog names were just words such as “incredible” or “very”. This error was made more pronounced as the real names began with uppercase while the error began with lowercase. We handled this issue by replacing those erroneous names with NaNs so they do not produce false insights downstream.

- Dog stage value made into columns.
  We noticed that there were columns like “Pupper”, “Doggo” etc in the dataset. These, according to the rules of tidy data, should have been values under one column: Dog Stage. We handled this by coalescing the values and creating the one column Dog Stage and getting rid of the now redundant value-named columns reducing our number of columns to 14.

The image prediction dataset was clean and tidy and ready to be used so not wrangling efforts were needed for it.

Twitter Data From API

This dataset initially contained 2059 rows and 27 columns. The following issues were discovered and handled:

- Useless Columns.
  There were a few columns (geo', 'coordinates', 'place', 'contributors') which were over 99.9% empty and this added clutter to the dataset without much value. We got rid of them.

- User variable contained many nested variables with same values.
  AS it was data from the same user, we did not need to keep all the user columns in the dataset. We left in user id and then took all the other columns to another dataset (series in this case as it was basically the same info over and over.

After all this, we merged the datasets.

- Unmatched IDs after merge.
  After merging the dataset in a manner to satisfy the project spec, some IDs were not matched between the image prediction dataset and tweet archive dataset. As these were datasets provisioned, there was little to do about it. It was however noted for future users.

- Drop Empty columns.

After the joins, there were more empty columns ('retweeted_status', 'retweeted_status_id', 'retweeted_status_user_id', 'retweeted_status_timestamp') which dropped to make reduce clutter.

- Remove Duplicate columns.
  After the join, some columns contained in two of the datasets joined were both returned as part of the final dataset. One set of these were dropped.

We then saved this final dataset as the master csv for analysis.