Project Goal:

We will wrangle WeRateDogs Twitter data to create interesting and trustworthy analyses and visualizations. The Twitter archive is great, but it only contains very basic tweet information. Additional gathering, then assessing and cleaning is required for "*Wow!*"-worthy analyses and visualizations.

Gathering Data

The first dataset is `twitter-archive-enhanced.csv`. The WeRateDogs Twitter archive contains basic tweet data for all 5000+ of their tweets, but not everything. One column the archive does contain though: each tweet's text, which I used to extract rating, dog name, and dog "stage" (i.e. doggo, floofer, pupper, and puppo) to make this Twitter archive "enhanced." Of the 5000+ tweets, It has been filtered for tweets with ratings only (there are 2356). This is downloaded directly and imported and read into a pandas DataFrame called `dog_ratings_df`.

The second dataset is `image_predictions.tsv` is present in each tweet according to a neural network. It is hosted on Udacity's servers and should be downloaded programmatically using the Requests library and the following URL: https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predictions/image-predictions.tsv. The DataFrame is called `img_pred_df`.

The third dataset is `tweet_json.txt` is Gathering **each tweet's retweet count** and **favorite ("like") count** at the minimum and any additional data you find interesting. Using the tweet IDs in the WeRateDogs Twitter archive, query the Twitter API for each tweet's JSON data using Python's Tweepy library and store each tweet's entire set of JSON data. This dataset contains **tweet_id**, **retweet_count** (number of shares), **favorite_count** (number of likes). The DataFrame is called `tw_df`.

Assess Data

Quality issue

`dog_ratings_df`

- tweet_id: is a int64 not a string
- timestamp: is string does not timestamp data type
- name: missing name represented as 'None' or not reasonable naming (such as: a) will be converted into None.
- Rating_numerator columns: converted into float64 rather int64 and captured the decimals from text and replaced with correct ones.
- Rating_denominator columns: converted into float64 rather int64.

- There are some ratings are not extracted correctly from the tweet status. After investigating in the tweet status. Here are ones:
  - index 1165: wrongly extracted from day format (4/20 – April 20) and it becomes numerator/denominator. It should be **13/10**
  - Index 2335: extracted 1/2 but correct is **9/10** after investigating in tweet's text
  - Index 1662: extracted 7/11 but correct is **10/10** after investigating in tweet's text
  - Index 1068: extracted 9/11 but correct is **14/10** after investigating in tweet's text
  - Index 784: extracted 9/11 but correct is **14/10** after investigating in tweet's text
  - Index 516: not relevant data in text 24/7 is not a rating. Will be **removed**
  - Index 342: not relevant data in text 11/15/15 is date format and the tweet_id is not matching into other datasets. Will be **removed.**
  - Index 313: 960/00 is not valid. It's a retweet. It should be **dropped** later.

`img_pred_df`

- tweet_id: is a int64 not a string
- The p1, p2, p3 columns contain mixed upper/lower case words. It can be standardized into title and strip out "_" by " ".

Tidiness issue

- The dataframe includes all the tweets which are re-tweeted from the original ones. This could consider the duplicated rows for analysis; therefore, we will exclude these rows out the the dataframe. The columns `retweeted_xxx` non-null data is the one will be dropped out.
- The columns which are the stages of dog life: doggo, floofer, pupper, puppo are in the wide format. It will make many None rows which is not a good format. Therefore, we will combine these ones into one column called `dog_stage`
- All three tables could be joined into one table by tweet_id.
- Dropout non-related columns for later to be analyzed.