Data Wrangling Process

- Data wrangling, which consists of:
  - Gathering data
  - Assessing data
  - Cleaning data
- Storing, analyzing, and visualizing your wrangled data.
- Reporting on 1) your data wrangling efforts and 2) your data analyses and visualizations.

Gathering Data for this Project

Gather each of the three pieces of data as described below in a Jupyter Notebook titled `wrangle_act.ipynb`:

1. The Waterdogs Twitter archive.
   I downloaded the file manually from the website, the file name is **`twitter_archive_enhanced.csv`**.

2. The tweet image predictions
   is the second file associated with prediction regarding what the breed of the dog according to a neural network. This file (`image_predictions.tsv`) 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**

   1. The third one is tweet_josn, we Using the tweet IDs in the Waterdogs Twitter archive, query the Twitter API for each tweet's JSON data using Python's **Tweedy** library and store each tweet's entire set of JSON data in a file called `tweet_json.txt` file. Each tweet's JSON data should be written to its own line. Then read this .txt file line by line into a pandas Data frame with (at minimum) tweet ID, retweet count, and favorite count.

   Note: do not include your Twitter API keys, secrets, and tokens in your project submission.

Assessing Data

After gathering Data, I assessed the data both visually and programmatically for quality and tidiness.

By using the pandas functions and methods we get:
.info(), .describe(), .head(), .sample(), .unique() and .value_counts()

Eight (8) quality issues

in `df_twitter_archive file`

1- in_reply_to_status_id column and in_reply_to_user_id column have many celles with NaN, so in_reply_to_status_id column has 78 non-null only instead of 2356.

2- in_reply_to_user_id column has 78 non-null only instead of 2356.

3- retweeted_status_id, retweeted_status_user_id, retweeted_status_timestamp columns have 181 non-null olny intstead of 2356.

4- missing values in expanded_urls column which has 2297 non-null instead of 2356.

5- name column
some values saved as a lowercase ['such', 'a', 'quite', 'not', 'one', 'incredibly', 'mad', 'an',none...etc]
some valuse are non titled like ['BeBe','CeCe','DonDon','DayZ','JD'].

6- We are interested in dogs, however text column shows some tweets are not related to dogs.

7- we have missing data in doggo,floofer,pupper, puppo columns.

8- timestamp , retweeted_status_timestamp are saved as object datatype (str) instead of date/timestamp.

9- tweet_id is saved as int and it will better to be (str).

10-source column wrote in html containg" <" "a" ">" tags.

11- incorrect data in rating_denominator and rating_numerator 1776, 960 and 666

12- rating_numerator should be float not int.

`image_prediction file`

1- 'p1', 'p2', 'p3' inconsistent capitalization (sometimes first letter is capital)

Overall

We have challenge in the shape of each file since we have a (2356, 2345, 2075) of 'twitter_archive','tweet_json' and 'image_pred' respectivily.

Tidiness

1- Three data frames `twitter_archive`, `image_pred`, and `tweet_json` should be one (combined table)

`df_twitter_archive` table:
1- we have two variables text and short urls,create short_urls column, drop expanded_urls.

2- one variable in four columns ('doggo', 'floofer', 'pupper', and 'puppo')

3- we need from tweet_json these columns only 'id', 'favorite_count' and 'retweet_count'

4- rating_numerator and rating_denominator columns in twitter_archive dataset should form one column dog_rating normalized out of 10.

5- tweet id presents in 'twitter_archive' and 'image_predictions' and as id in 'tweet_info' so we will rename it.

Clean

1. get a copy from each file to protect the source clean
   twitter_archive_clean = twitter_archive.copy()
   image_pred_clean = image_pred.copy()
   tweet_json_clean = tweet_json.copy()
2. Missing Data
3. Tidiness
4. Quality issues.

   ## resources
   1. Google.
   2. 2. Stackoverflow.
   3. Pandas websites.