Wrangle Report

1. Gathering Data:

The wrangling dataset consists 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. This archive/dataset consists of 2356 tweet data from November 2015 to August 2017. Based on the tweet ID from above dataset, we are also provided with another dataset, consists of image predictions (the top three probability of objects/dogs breeds), also including user's tweet ID, image URL, the image number that corresponded to the most confident prediction.

- **Gather Twitter archive CSV file**
  I directly downloaded the WeRateDogs Twitter archive csv file manually as twitter_archive_enhanced.csv from udacity website. Then use pandas to import to dataframe df1.

- **Gather tweet image predictions**
  I downloaded the tweet image predictions file hosted on url provided by udacity (https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predictions/image-predictions.tsv) with Python's Requests library and saved it locally to image_predictions.tsv. Then use Pandas to import to dataframe df2.

- **Gather data from Twitter API**
  Using the tweet IDs in the Twitter archive, I accessed the entire data for every tweet from Twitter API and stored every tweet ID JSON response data in tweet_json.txt file. Then imported to dataframe df3 including only tweet_id, retweet_count, favorite_count, created_at, text, source and display_text_range data.

2. Assessing data:

- **Visual Assessing**

  I opened the archive and prediction two datasets with excel, scroll down to check each column, notice 2 quality and 2 tidiness issues.

  Quality issues 1:
  Archive data source field include html tags, which should be cleared to only has "tweet for iphone" etc.
  Quality issues 2:
  Archive data text field has some special character like üêáüêð, is this all emoji?

  Tidiness issues 1:
  In archive data, there are 4 columns for 4 different types of dog breeds, which can be merge into 1 single column.
  Tidiness issues 2:
  There are several columns like "retweeted_status_id", "retweeted_status_user_id" and "retweeted_status_timestamp", missing too many values, which can be dropped.

- **Programmatic assessing**
  I use pandas dataframe included functions to programmatic assessing the two datasets. Basically use df.info() to check the data column value if data type is correct, df.describe() and df[column].value_counts() to check the numerical value. Thus, I found the following 11 quality issues and 2 tidiness issues.

  Quality Issues:
  - There is much less data in prediction than archive tweet datasets. (2075 prediction /2356 tweet archive data)
  - *'in_reply_to_status_id', 'in_reply_to_user_id', columns seem to miss a lot of values. Only 78 is not null.*
  - *'retweeted_status_id', 'retweeted_status_user_id', 'retweeted_status_timestamp' columns seem to miss a lot of values. Only 181 is not null.*
  - Some rating numerators and denominators are 0. Some rating scale is very different from others, e.g. rating denominator is 110, 170, 130…, which could be not parsing data correctly.
  - Some dog names are missing with value "None"
  - Some expanded URLs have multiple or 0 urls
  - *Several columns data should be in timestamp format instead of string.*
  - The source field in archive data include html tags, which could be much more simplified.
  - Two tweet id in tweet data is not in tweet_txt.json, these ids are 888202515573088257 and 771004394259247104
  - Some prediction says that image prediction is not dog, but actually is a dog image, by checking url directly.

- Prediction datasets say imag_num is more than 1, but in the link of image url, there is only 1 url exist with only 1 image.
- Total dog types are only 4 stages in tweet data. doggo(97), pupper (257), floofer(10), puppo(30). which is much less than the whole datasets of size 2356.

Tidiness issue:

- In tweet_data, too many columns for the dog stage, as in doggo, pupper, floofer and puppo, which can be combined in a single category column, ‘dog_stage’.
- All these files should be joined together to provide the final dataset.

3. Cleaning data

As in the above discuss issues with archive data, prediction data and tweet_json.txt, I create a copy from them, then use pandas dataframe to address the above discussed quality and tidiness issues. But some issues that related to the prediction data is hard to clean from me.

- Remove tweet archive data of retweet data, so only use original tweet data.
- Drop several columns which has too many null values.
- Change timestamp to correct.
- Merger tweet archive data with prediction data, remove tweet id which do not have image prediction info.
- Remove html tag from “source” column and set them as category variable.
- The text box in cleandf has text of various length, use tweet_txt.json with text_range to set correct range, remove url in the end.
- Since 4 dog stages are in 4 different columns, we could map them into a single column called 'stage'. Then map them into category type.
- Due there are prediction probability included for top3 breeds, we only need 1 column keep highest probability. If highest probability is dog, put breed in column, if not, then just use None.

4. Storing data

After completion the cleaning process, I stored the dataframe into tweet_archive_cleaned.csv