Module 4: Assessing dog breeds for maximum social media outreach

The rubric needs 8 quality issues and 2 tidiness issues to be documented. In this section, I will proceed to identify as many quality/tidiness issues that would be crucial to solve before analysis.
Before proceeding with the assessment of data, I want to list the dataframes created so far and the information they capture

1. dogs_df which contains the basic information about the dogs in the dataframe - including but not limited to tweet id, rating, category, name, description,urls and so on
2. image_prediction which contains the top 3 predictions for the breed of the dog with the confidence of prediction
3. twitter_data which contains the retweets and likes garnered by the tweet.

Next I will create the common segment where the quality and tidiness issues will be stacked. This will be the single place we'll be returning to in case we want to refer issues later on.

Quality

I) dogs_df

- Some of the observations are retweets so need to be discarded. Filter out only those tweets that have an image url
- timestamp column should be of datatype datetime
- We need to replace all None occurences with NaN to make the dataframe easy to work with
- ~~in_reply_to_status_id,in_reply_to_user_id and the retweeted categories have a lot of missing values - can be retreived from the tweet_json.txt file if present~~
- The ratings (rating_numerator) are indicated to be incorrect a lot of times. This can be corrected for.
- Dog stages too are sometimes missing/sometimes incorrect and need to be corrected
- ~~Dog names too can be looked at for inconsistencies (although we might need to examine the feasibility of this since this can be a pretty complex endeavour)~~ Can't be done with current knowledge (might need NLP knowledge)
- Dog names are incorrectly identified (bizarre name "such" at index 23, or the name "quite", "an", "not", "one", "very", "O","just","old","life","officially", "space","light")

II) image_prediction

- ~~Need to check for duplicate tweet id rows and consider the prediction confidence to choose which one to keep~~

III) twitter_data

- ~~Need to check if any other columns can be added to the dataframe (Not a data cleaning task but rather a gathering task. Gather-Assess-Clean can be iterative!!!).~~

Tidiness

I) dogs_df

- doggo,floofer,pupper and puppo should be under a single column category. One thing to note here that a single dog may be identified to belong to several categories
- text column has information pertinent to other features too and needs to be cleaned as per rule - every variable should be in a new column

II) image_prediction

- ~~Need this be a separate table? This upon examination does satisfy the condition for tidiness - each type of observational unit must be a table~~ This does not need to be a separate table

III) twitter_data

- ~~Need this be a separate table? This needs to examined since maybe this may not satisfy the condition for tidiness - each type of observational unit must be a table~~ This does not need to be a separate table.

Quality

Quality issues in data are resolved in this section

1. Retweets are to be removed. For this we can check retweeted_status_id columns and remove if not null.
   As already stated earlier, the replies are all from WeRateDogs and hence will be retained for analysis.
   I will also drop the 3 columns that captures retweet information since that is no longer required.
2. The timestamp column should be of datatype datetime. We can use the pandas to_datetime() method to get the column in the correct format
3. Replace the null occurences with NaN. This will make it easier for us to work with the data in the later stages. We can use the replace() method to achieve this.
4. The ratings numerator and denominator needs to be corrected. This can be achieved by extracting the correct values from the cleaned dataframe using the extract() method with appropriate regular expression
5. Missing values for dog stages. This can be resolved by converting text in column to lowercase, extracting the category using extract method with appropriate regular expression
6. The names of some dogs have been incorrectly identified. We can set these names to NaN. We can resolve this by extracting rows with names starting with a lowercase character (observation reveals this pattern). Now we can set the values of name column in these rows to np.nan

Tidiness

1. doggo,floofer,pupper and puppo should be under a single column category. One thing to note here that a single dog may be identified to belong to several categories. We will create another column category to capture this information and drop the other columns. This might not be the most elegant solution (Duh! It runs in $O(n^2)$) but it works (the melt function was driving me nuts). Note that here we're putting back 'Unidentified' in place of NaN. This is so as to maintain some consistency in datatype of the column
2. text column in dogs_df has information pertinent to other features too and needs to be cleaned as per rule - every variable should be in a new column. Since there's two parts - one the rating and two the link, both of which will not appear as discrete words (ie not have spaces) and both contain numbers, we can simply filter for the segments that do not contain numeric characters. This may however lead to some loss of context and incomplete sentences since a lot of places ratings are included as parts of a sentence. This might however be unavoidable.
3. twitter_data does not need to be a separate table since it contains information and metadata of the tweets which is precisely what dogs_df seeks to capture. We'll be using the merge function to join the columns we are interested in - favorite_count and retweet_count. We will also merge image_predictions table with the resultant quantity.