Data Gathering:

I gather data using the given CSV file as the ‘twitter_archive’ data frame. Then created another data frame named ‘image-prediction’ using the ‘requests’ package. Then one last using Twitter API (tweepy package) as ‘tweet_data’.

Assessing Data:

I assess data using both visual and programmatic approaches. Use functions like .info(), .value_counts(), .head(), .sample(5), .tail() and find following issues:

Quality issues

1. Datatype

2. duplicate data in the archive table

3. Replace lowercase name

4. Denominator has 15 at a few places

5. Inconsistency in Numerator

6. Extract data from the Text column

7.

8. Drop columns that are not required

Tidiness issues

1. Multiple columns with the same info in the archive table.\ clean_image: clean columns p1, p2, p3\ clean_image: clean columns p1_dog, p2_dog, p3_dog\ clean_image: clean column p1_conf, p2_conf, p3_conf.

2. Each type of observational unit forms a table.

Cleaning Data:

Before cleaning the Data I created copies of the original data as clean_archive, clean_image, and clean_tweet.

Quality issues

1. Datatype:

   Change the data type of all three data frames using the astype() function. ‘tweet_id’ to string that is given as integers and the ‘timestamp’ column in DateTime which is given as a string.

2. Duplicate data in the archive table:

In the clean_archive data frame, I removed retweeted tweets. Only keep the original one by checking if retweet_status_id is null.

3. Replace lowercase name :

   The ‘name’ column has lowercase names some of those are just alphabets. I replace them with a null value.

4. Denominator has a different value in some places:

   Some rows of rating_denominator have different values. Ratings are given from the 10 so I assign 10 to the whole column.

5. Inconsistency in Numerator:

   The numerator is different from the actual data given in the text column. so I use the extract function and regex pattern to separate the float value and access the numerator. After that convert it into the float.

6. Extract data from the column:

   The ‘text’ column has the comment and URL both in the same column. I separate text data from the URL using the string replace function and regex pattern.

7. Replace the ‘None’ string with nan:

   The name column has some null values which are shown as the string ‘None’. I replace them with np. Nan.

8. Drop the columns:

   I drop the columns which are not required for insight and visualization. Dropped columns are ‘in_reply_to_status_id', 'in_reply_to_user_id', 'source', 'expanded_urls', 'retweeted_status_id', 'retweeted_status_user_id', ‘retweeted_status_timestamp', 'name'

Tidiness issues

1. Multiple columns with the same info in the twitter_archive table: Four columns ‘doggo’, ‘floofer’, ‘pupper’, and ‘puppo’ are given as four different columns I combine them as one column ‘stage’ using NumPy select () function.

   clean_image: Clean p1, p2, p3: I use NumPy select function to access the predicted breed column as pre_breed from p1, p2, and p3. And only keep the highest confidence predicted breed in a single column.

   clean_image: clean column p1_dog, p2_dog, p3_dog with this process, I keep the image prediction true or false based on high prediction confidence in column ‘isdog’. After that, I drop all three columns.

   clean_image: clean columns p1_conf, p2_conf, p3_conf Here in this cleaning process, I keep only high-confidence data in column ‘high_conf’ that is higher than 2 other columns and then drop all three columns.

2. Each type of observational unit forms a table. The tweet id column is common in all three Data frames. All three columns have the twitter data they all make sense together so I merge them into one master column ‘twitter_archive_master’. I take all the columns of the clean_tweet and clean_image data frame but some selected ones('tweet_id', 'timestamp', 'text', 'rating_numerator', 'rating_denominator', 'stage') from the clean_archive Data frame.