Wrangle Report

Introduction:

The purpose of this project to is to apply practically what we learned in wrangling course and the main task is to Gather, Assess and clean data. The wrangled data are tweets archived from Twitter account called WeRateDogs.

WeRateDogs is a Twitter account started in 2015 by student called **Matt Nelson,**
This account specialized in rating the dog's by sharing the owners their dog pic and rated by the people's comments.
The dog's pic has been rated on the scale from one to ten but usually they got more than ten.
In this project, we will make data wrangling for most favorite tweets and we will assessing many of information about it.

Project Details:

Your tasks in this project are as follows:

- 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. Twitter archive (twitter_archive_enhanced.csv) file is download manually by clicking the following link:
2. Image predictions file (image_predictions.tsv) file is downloaded programmatically using the request library and download from given URL.

3. Twitter JSON and API file using the tweet IDs in the WeRateDogs Twitter archive, query the Twitter API for each tweet's JSON data using Python's library Tweepy 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 DataFrame with (at minimum) tweet ID, retweet count, and favorite count.

Assessing Data:

Once my data gathered correctly I started to assess it by two ways:

1- Visual assessing: visual review of the data.
2- Programmatically assessing: using Python and different libraries.

Based on last step the following are my assessing points:

Quality:

(Twitter_Archive)

1- Change denominator ratings & numerator ratings str to float.

2- Adjust mistyping of numerator ratings values >10.

3- replace all denominator ratings to be 10

4- Convert all dogs 'name' to be 1st letter capitalized.

5- Rename columns 'name' to 'dog_name' and 'text' to 'tweets'.

6- Drop 'Doggo', 'Floofer', 'Pupper' and 'Puppo' Columns.

7- Separate between words DoggoPupper, DoggoFloofer and DoggoPuppo to (Doggo, Pupper), (Doggo, Floofer) and (Doggo, Puppo)

8- Remove Retweets but Keep original ratings that have images.

9- Drop all unuseful data columns for data analysis.

(image_Prediction)

10- Drop all duplicated jpg_url.

11-Drop all non-dogs predictions.

(Tweet_json)

12 - Drop all unuseful Coulmns for data analysis.

13- Rename id to tweet_id to be consistent with others files during merging.

14- Keep only English language tweets

Tidiness:

1- Convert 'timestamp' column dtype to Year, Month and Day datetime Columns.
2- Drop 'timestamp' Column.
3- Creat new Dog Stage column for Dogs classification.
4- Combine all three different files to master data set.

Cleaning:

For this part of the data wrangling was divided in three parts: Define code and test the code. These three steps were on each of the issues described in the assess section. First very helpful step was to create a copy of the three original dataframes. I wrote the codes to manipulate the copies. If there was an error, I could create a new copy from the original. Whenever I made a mistake, I could create another copy of the dataframes and continue working on the cleaning part.
There were some of cleaning steps that were very challenging. One of them was in the image prediction table. I had to create a ‘nested if’ inside a function in order to capture the first true prediction of the type of dog. The original table had three predictions and confidence levels. I filtered this into one column for dog type and one column for confidence level. Other interesting cleaning code was to melt the dog stages in one column instead of four columns as original presented in twitter archive. Another challenge when start to clean tidiness issues to prepare the structure data frame at 'timestamp' column change from string to date format day, month , year columns and delete timestamp column, after clean tidiness issues, I jumped to solve quality issues starting In rating denominator column has a standard rate 10 but it have different rate start to set all to 10 rate, same mistyping happen in rating numerator columns, some columns have None value which mean nothing but not NaN Changed empty cell in doggo floofer ,pupper ,puppo to add in dogs stage during merge in one columns to be easy to replace and no need for doggo,floofer,pupper,puppo columns so deleted.