wrangle report

For my data wrangle project which required three data set for the analysis, gotten from different sources this is my analysis report.

Firstly, I imported all the needed libraries into my jupyter notebook, read the first data from a csv file" twitter-archive-enhanced.csv" into a data frame 'enhanced', using the requests library I save the prediction data set into a .tsv file,read then to a second data frame 'image'. The third data set I used was gotten from WE RATE DOG twitter page which needed to be assessed through twitter API , I was not able to get the twitter developer approval for my keys, I used the json.txt file. Read it to a data frame as 'tweets' then extracted three columns need for my analysis.

Assessing Data:

Here I assessed all my three data frames both Programmatic assessment and visual assessment.

The twitter enhanced data set: the data set contains 2354 rows and 17 columns

The image predictions data set: the data set contained 2075 row and 12 columns

The tweet json data set: the data contained 2354 columns and 3 rows

Assessment issues:

- the "retweeted_status_id,in_reply_to_status_id" column have lot of null values
- not all columns are needed for this analysis which needs to be dropped
- the data type for some of the columns are incorrect
- the doggo,floofer,pupper,and puppo should be a single column and a category type of data
- the rating_ columns should be a single colum
- the name column has lot of unfidined values
- some images and false and are not dogs
- the url has duplicated links
- the id columns is not the same with other values

Cleaning data: Before cleaning I made copies of each of my data frame

Quality issues

enhanced data set:

- Here for the enhanced data set I dropped the columns in the retweet_ and reply_ columns that are not original tweet this are tweet that have values in them which made the column empty.
- Dropped columns that are not needed for my analysis which are in_reply_to_status_id, in_reply_to_user_id, retweeted_status_id, retweeted_status_user_id, retweeted_status_timestamp, expanded_urls
- Changed data types then created a single column for all the dog stages
- Cleaned the ratings column into one and used '/' as the delimiter
- Since the name column contain may words that are undefined, cleaned the undefined words into null

Image data set:

Here I removed all the p images that are false which mean the row do not contain any dog image. For my **tidiness** I extracted the highest confidence level for the p1,p2,p3 which are true along with their corresponding breed name into a new column using the np.select() statement and dropped the old columns.

Tweet data set:

Changed the id name to tweet_id to make my merging possible.

For my final cleaning I merge all the data set into a single dog table and dropped empty. It contained 1666 row and 13 columns.