Project Description

Datasets:

In the `wrangle_act.ipynb` notebook we cleaned and analyzed three datasets:

- `twitter-archive-enhanced.csv`
- `tweet_json.txt`
- `image-predictions.tsv`

The first file contains some data on each tweet from WeRateDogs Twitter acount, like the tweet-ID, timestamp, description, device that it was sent, etc.

Using Python's `tweepy` API we created a second file which has an array that contains dictionaries with the ID, like-count, and retweet count for each tweet-ID in the first file.

And the third file was obtained through making a web request to Udacitity's server and it contains the dog-breed prediction for each tweet-ID

Data Wrangling:

First of all we joind the `twitter-archive-enhanced.csv` and `tweet_json.txt` together so we have a dataset that contains all the relevant tweet information.

Then we joined the predicitons dataset, but with a twist: We only join the first breed prediction for each tweet-ID that was classified as a dog.

Then we addressed the following data issues:

Quality Issues:

- Remove retweets
- Remove replies
- keep only original tweets with image
- Keep only records with a rating
- favorite_count column should be integer, not float
- retweet_count column should be integer, not float
- time_stamp column should be datetime, not object
- convert `'None'` in string columns to `None/np.nan`

Tidiness Issues:

- ['doggo', 'floofer', 'pupper', 'puppo'] columns should all be in the same (Categorical) column
- source column contains two variables (url and device name)

Approach used for some of the issues:

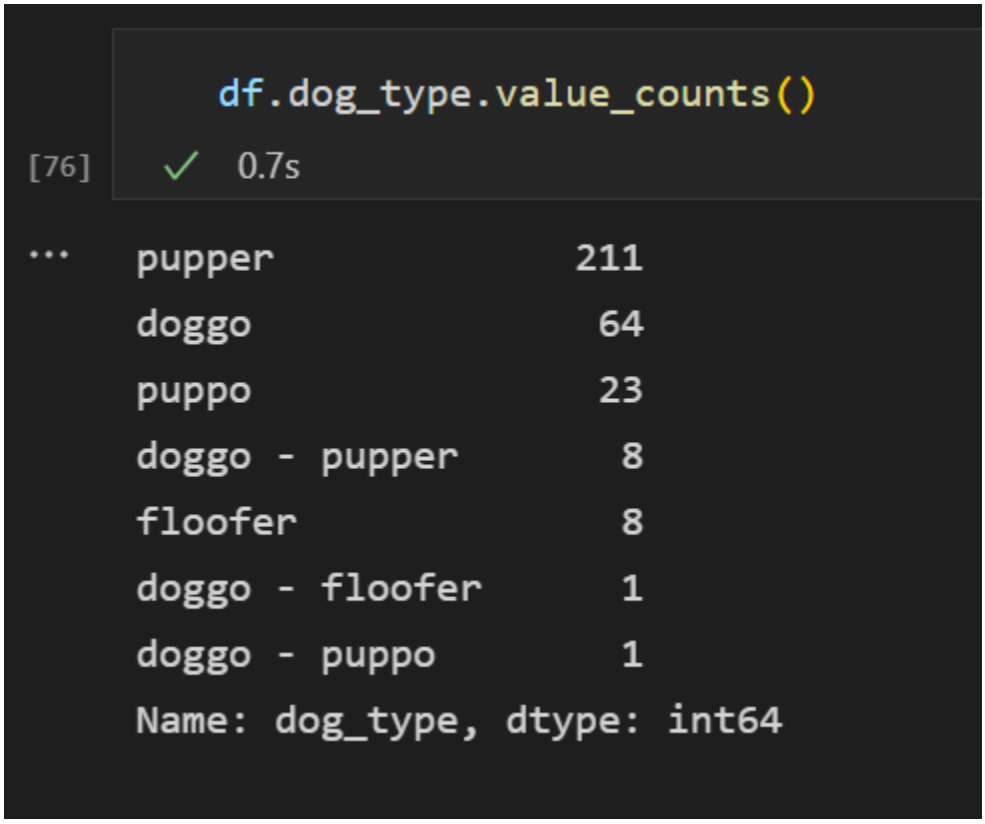
For the dog-type column (1st tidiness issue) we looped through all the rows in these four columns: doggo, floofer, pupper, puppo,

and we checked how many non-null values there are,

if there was one then we returned the single non-null value,

if there were multiple: we joined them together with a dash,

and if there were none, then we just returned `None`. The result is the following:



For the 2nd tidiness issue we simply used a regex pattern to extract the device used from the HTML text.

Output

And finally, after cleaning and joining all three datasets we save the main DataFrame as `twitter_archive_master.csv`.