Wrangle Report

Introduction

Udacity Data Analyst Nanodegree required me to do a wrangle and analysis project called Wrangle and Analyze Data. My purpose was to collect tweets from Tweet user @WeRateDogs. @WeRateDogs is a user which rates dogs in quite unusual rating system, more on that later.

Gathering

Udacity actually provided all of needed files which are:

- twitter-archive-enchanced.csv
- tweet-json.txt

There was possibility to download tweet-json.txt via Twitter API but I didn't manage to do this approach.

Another important file for our analysis was:

- image_predictions.tsv

This file analyses images of the dogs within twitter archive and presents us with
-mostly- accurate breed predictions, there was few troubles, but we will get to that.

Assessing

I have opened each file and took brief look at quality and tidiness of those three files. Main issues which I have found were:

Quality Issues:

- 'tweet_id' & 'id' should be converted to string as we will not perform operations on that

- Timestamp columns are not datetime format

- Some of the colums have 'None' values

- Diffrent amounts of images and tweets (2075 vs 2356)

- Image_predictions have strange answers -> torch for example

- Some of the tweets are retweets

- 'None' or 'a' in name column

- 'rating_numerator' and 'rating_denominator' have some strange values sometimes

Tidiness Issues:

- Too many columns and not every single one is needed for data analysis, we should drop them

- All of dataframes include ID (either 'tweet_id' or 'id' can be merged into one big DF

- Breed of the dog - doggo/ floofer/ pupper/ puppo could be merged into one new column 'dog_breed'

- There is normalized score for dogs, sometimes its 88/80 for example when normalized should be 1,1 etc

Cleaning

Cleaning of the data was quite easy, except:

- 'rating_numerator' and 'rating_denominator' have some strange values sometimes

For most of our problems basic Python functions was enough. I have used:

.drop()

.merge()

.value_counts()

.describe()

.info()

.loc[]

Cleaning 'rating_numerator' and 'rating_denominator' was bit more difficult and except few rows, data was scaled – which means for example instead of 12/10 WeRateDogs gave rate of 144/120 – which is 10times our normal scale. To walk around this issue I had to create new column and calculate normalized score:

df_all['normalized_score'] = df_all['rating_numerator']/df_all['rating_denominator']

Next I have created new DataFrame called ‘df_score’ which had raiting (ex 1.2) and value_count (ex 454), it allowed me to make visualization for this specific data.

For Visualisations I kept it brief making three plots, by using:

.plot()

For top N-values there was need to use

.nlargest(N)

Next, adding labels is used by simple matplotlib functions:

.xlabel()

.ylabel()

.title()

Additional Files

Three additional files are created:

- df_names.csv
- df_race_count.csv
- df_score.csv

Those files are needed for easier access to data which was used for visializations.

Lastly, as Udacity requested, I have created master file:

- twitter_archive_master.csv

In this file you will find all merged, corrected and cleaned data.