Data Wrangling Report

We Rate Dogs

Prepared by: Maria Latysheva, September 2020

This report briefly presents my data wrangling efforts in the Data Wrangling Project as part of the Udacity Data Analyst Nanodegree.

Data Wrangling consisted on the following three parts:

- Gathering data
- Assessing data
- Cleaning data

Gathering data

In this project I gathered data about dog ratings (from the Twitter account We Rate Dogs) and dog breeds from three different sources and in three different formats:

1) I collected 2,356 entries from the twitter_archive_enhanced.csv file, which was provided by Udacity;
2) I downloaded 2,075 tweet image predictions programmatically from the file `image_predictions.tsv`, which is hosted on Udacity's servers by using the *requests* library.
3) I gathered 2,340 entries from the `tweet_json.txt` file and then read this .txt file line by line into a pandas DataFrame. Unfortunately, Twitter did not approve my developer's application and I was not able to us the Twitter API in real life to download this data myself.

Assessing data

I assessed the three datasets both visually and programmatically and detected 8 quality-related and 4 tidiness-related issues:

Quality issues included:

Some entries were retweets or replies while we were only interested in original tweets
Some entries did not have images and we were only interested in entries with images
The name column contained invalid dog names like the words "a" or "one"
The timestamp column has the type *str* instead of *datetime*
Some rating numerators and rating denominators seemed to be wrong or two high
Some breeds started with lowcase letters and some with capital letters

The issues detected were documented to be cleaned to simplify subsequently the analysis.

Cleaning data

I created copies of all three datasets and performed cleaning operations on the copies.
I cleaned all the issued detected during the data assessment stage. For the cleaning purposes, I used such *pandas* or *python* methods and functions as:
drop(), notnull(), notna(), conditions, pd.melt, pd.merge, rename(columns), str.contains(regular expression), str.split, str.capitalize, etc.
After cleaning I merged all three datasets into one master dataset.

Storing data

I stored the cleaned dataset into a master csv file by using pd.to_csv() function.