Wrangle Report

The **Udacity Project 5 - Wrangle and Analyze Data – WeRateDogs**, entailed wrangling the WeRateDogs Twitter data to carry out trustworthy analyses. In addition to the basic info we could get from the Twitter archive, we also gathered, assessed and cleaned the Twitter data.

The first step was "**Gathering Data**". It consisted of getting three datasets:

- Data from the Twitter archive provided as a .csv file, then imported in a Pandas DataFrame named "**archive**"
- Data for predictions of the breed of the dog for the image of each record, downloaded programmatically from an URL provided, using the *request* library ad imported in a Pandas DataFrame named "**image_predictions**"
- Data about the actual tweets, retweets and favs in the WeRateDogs Twitter account, were gathered through the Twitter API with the support of *tweepy* library and ultimately imported in a Pandas DataFrame named "**tweet**".

Once we got all the three datasets we went through the "**Assessing Data**" phase, which entailed discovering and documenting at least 8 Quality issues and 2 Tidiness issues. Last step of the wrangling process was the "**Cleaning Data**" phase, in which we fixed the issues found during the assessment phase.

We discovered ten **Quality issues**:

In the "***archive***" DataFrame:

1) Looking at the dataset we noticed that some of the dog names, in the column *name*, weren't real names. They were found using *str.islower* function and then replaced with *"None"*.
2) With the *.info()* function we found that the column *retweeted_status_id* had 181 non-null instances, meaning that they weren't original tweets. These rows were found with the *.notnull()* function and then dropped using the .drop() function.
3) *tweet_id* was converted from a number to string value using *.astype(str)* function.
4) In the *rating_denominator* column the value wasn't constant, making the rating inconsistent. Issue fixed creating a new column *dog_rating* as a ratio between *rating_numerator* and *rating_denominator*.

In the "***image_predictions***" DataFrame:

5) Not all the values of the columns *p1*, *p2* and *p3* started with a capital letter. Issue fixed using s*tr.capitalize()* function to achieve consistency.
6) In the columns *p1, p2* and *p3* we remove the underscore "_" between the words with the function *.str.replace('_', ' ').*
7) *tweet_id* was converted from a number to string value using *.astype(str)* function.

In the "***tweet***" DataFrame:

8) *id* was converted from a number to string value using *.astype(str)* function.
9) *id* was renamed as *tweet_id* with the *.rename()* function to achieve consistency with the other two dataframes.
10) 167 records having a non-null *retweeted_status* were found with the *.isnull()* function and then removed using the *.drop()* function.

We discovered four **Tidyness issues**:

In the "***archive***" DataFrame:

1) The *timestamp* column was first converted from string to datetime object using *.to_datetime* function, then split in two different columns (*date* and *time*) to keep those two pieces of information separated using .strftime() function.
2) The variable “dog stage” was stored in four columns (*doggo*, *floofer*, *pupper*, and *puppo*). This issue was fixed by creating a new column named *stage*, filled with the appropriate dog stage through the use of the *.str.extract()* function.
3) We dropped the columns that were not required for the analysis.

All the three DataFrame:

4) Merged into a unique dataframe named *twitter_archive_master* using *.merge()* function with a inner join on *tweet_id* columns.

This project was very helpful to understand the importance of assessing and cleaning the data in a structured way before the analysis. Not cleaning the data would potentially lead to wrong / misleading conclusions during the *Analysis* phase.

It’s important that the assessment is carried out before the cleaning, and the cleaning before the analysis, as this is the correct sequence to follow to fix the issues with the lowest possible amount of efforts.