DATA WRANGLING REPORT

Introduction

Data wrangling, according to Wikipedia, is the process of transforming and mapping data from one "raw" data form into another format with the intent of making it more appropriate and valuable for a variety of downstream purposes such as analytics. Below are the three major steps I followed in my wrangling efforts

- Gathering
- Assessing
- Cleaning
- Storing

Gathering

I gathered three different data from three different data sources:

1.WeRateDogs Twitter archive downloaded Manually from Udacity.

To get the file "(*twitter_archive_enhanced.csv*)" into my notebook after downloading from Udacity, I read it using pandas "*read_csv*" function and stored it as "*t_archive*". Choose "*t_archive*" so it would be easier to work with using a short name.

2. tweet image predictions file downloaded programmatically using requests library from Udacity's provided URL.

I was provided a URL which stored the file "(*image_predictions.tsv*)". And to get that, I used the Requests library "get" function to retrieve the file from the URL provided. The content of the file was read into a python data from using pandas and stored as pred. Using a short name again.

3. Additional data retrieved by querying Twitter's API using Tweepy libary.

Here comes the tricky part. After getting the required credentials from Twitter to use the Twitter API. I queried each tweet Id stored in t_archive for "favorite count" and "retweets" using a for loop and storing the result if the tweet Id was found in

"*tweets_id_data*" while those that were not found were stored in "*notweets_id_data*". Saved the final result after in a file called "*tweets_json.txt*" and stored it as "*tweet_data"* in my notebook.

- The process took 3047 seconds to complete
- 2296 tweet id found
- 60 tweet id not found

Assessing

Data was assessed for quality and tidiness issues both visually and programmatically.

The following issues were found

Quality (*Issues with file content*)

- t_archive data - Timestamp column is in object format
- t_archive data - Lots of dog rating numerator greater than 15
- t_archive data - Lots of dog rating denuminator greater than 10
- t_archive data - The source column needs cleaning
- t_archive data - Colunms with empty values have None as a value which is not showing as Null or NaN(null object not null)
- pred data - The names in the dog breed predictions are not uniform some in lower case while others in mixed case
- t_archive data - Some dogs have invalid names
- pred data - Three dog breed prediction available
- all_merged- Retweets and reply present
- all_merged- Unnecessary columns
- all_merged - Tweets without image
- all_merged - Wrong assigned datatype

Tidiness (*issues with file structure*)

- t_archive - Dog stage is scattered in 4 columns
- The three dataset have tweet id in common

Cleaning

The steps involved in cleaning of the data Is well documented in each issue (Define, Code and Test) cells.

I define the issue, document a possible way to fix it, attempt to fix it in the code cell. Test if my solution works as expected in the test cells.

Storing

The final data "*all_merge_clean*" was stored in a CSV file called "*twitter_archive_master.csv*" using "to_csv" function. It contains 1963 rows and 14 columns.