wrangle_report

June 27, 2022

0.1 Reporting: wragle_report

- Create a **300-600 word written report** called "wrangle_report.pdf" or "wrangle_report.html" that briefly describes your wrangling efforts. This is to be framed as an internal document.

0.2 The objectives of the project

The aim and objective of this project is to wrangle the data of dog ratings, the data source is from the twitter user @WeRateDogs. Gathering, assessing and cleaning would be done to prepare the data for analysis. The process would be discussed as follows

0.3 Data gathering

```
Three data were provided in this project which are Twitter archive file, Tweet image prediction
```

The first data which is the Twitter archive file was already provided bu Udacity, i downloaded it into my project workspace by clicking on the yellow jupyter icon at the top right corner and then uploaded the file. I my work space i imported the pandas library that i used to read the file into a dataframe using a pandas method called read_csv() and stored the result using a variable called df_A. Secondly, downloaded the second data image prediction file file programmatically using the request and os libraries that i mported at the begining and used the with open() method of the request library to generate a response of 200 which i obtained success. Then i wrote the response content into a tsv file and named it image-prediction.tsv which i later read it into a dataframe then called it image_prediction Lastly, creating a twitter developer account and creating an application for the project. I used the app credentials (consumer_key, consumer_secret, access_toke, and access_secret), for the twitter API i imported tweepy and json, authenticated tweepy.OAuthHandler and set wait_on_limit to True. I used the code already provided by udacity to extract the tweet id and created the file tweet_json.txt . with the python with open() function i read the tweet_json.txt line by line and loaded each line as json file, i then read it into a pandas dataframe saving the result using a variable called tweetcount

0.4 Data assessment

I used two method to assess the data;

Visual assessment: i printed out all the data by calling their variable names used to store them to see hoh they look like

Programmmatic assessment: i used the pandas function such as .info(), .describe(), .isnull(), .head() and .value_counts() method to assess the data

0.5 Data cleaning

```
At this point made a copy of all the data and created an new variable name by adding _clean to t
i then define the problem each data has, code the problem in order to solve it and test to see t
```

1. replaced the 'None', 'a', 'an' and 'the' etc. values with NaN using the replace method
2. replaced the 'None' values with NAN using the replace method and dropping the NaN values
3. masked the df_A_clean to remove the retweeted data using the isnull() method
4. used the pandas drop method to drop the columns 'in_reply_to_status_id', 'in_reply_to_user_id', 'retweeted_status_id', 'retweeted_status_user_id', 'retweeted_status_timestamp all have more than 50% of its data missing
5. changed the datatype of the 'timestamp' column using the pandas to_datatime funtion
6. used the regular expression method to extract the name of the source
7. used the pandas rename method to rename the colums 'img_num' and 'jpg_url'
8. used the pandas rename method to rename the 'id' column
9. used the pandas DatetimeIndex function to split the 'timestamp' into 'year', 'month' and 'day'
10. used the pandas string(split) and indexing method to remove dulipcate links
11. i merged the three dataframes together

0.6 storing the data

After the gathering assessing and cleaning processes were concluded i saved the merged data in a csv file named twitter_archive_master.csv

0.7 conclusion

The data wrangling process was quite hectic and tidious but i was able to apply the methods that i learnt from the course which was exciting.