Report:wragle_report¶

Introduction

The purpose of this project is to put in practice what I learned in data wrangling data section from Udacity Data Analysis Nanodegree program. The dataset that it will be wrangling (and analyzing and visualizing) is the tweet archive of Twitter user @dog_rates, also known as WeRateDogs. WeRateDogs is a Twitter account that rates people's dogs with a humorous comment about the dog. These ratings almost always have a denominator of 10.

This report briefly describes my wrangling efforts.

Project details

The tasks of this project are as follows:

•Gathering data

•Assessing data

•Cleaning data

- Storing data
- Analizing and visulization
- Reporting

Gathering data

The data for this project consist on three different datasets that were obtained as following:

•Twitter archive file: the twitter_archive_enhanced.csv was provided by Udacity and downloaded manually.

•The tweet image predictions, i.e., what breed of is present in each tweet according to a neural network.

This file (image_predictions.tsv) is hosted on Udacity's servers and was downloaded programmatically using the Requests library and URL information

•Twitter API & JSON: by using the tweet IDs in the WeRateDogs Twitter archive, I queried the Twitter API for each tweet's JSON data using Python's Tweepy library and stored each tweet's entire set of JSON data in a file called tweet_json.txt file. I read this .txt file line by line into a pandas dataframe with tweet ID, favorite count, retweet count, followers count, friends count, source, retweeted status and url.

Assessing data

After gathering all three pieces of data, assess them visually and programmatically for quality and tidiness issues

•**Visually**, I used two tools. One was by printing the three entire dataframes separate in Jupyter Notebook and two by checking the csv files in Excel.

•**Programmatically**, by using different methods (e.g. info, value_counts, sample, duplicated, describe,head, etc).

Then I separated the issues encountered in quality issues and tidiness issues.

Cleaning data

This second part is divided in three parts:

•Define

•code

•test the code.

Clean all of the issues you documented while assessing. Perform this cleaning in the "Cleaning Data" section in the wrangle_act.ipynb.

•First step is to create a copy of the three original dataframes. I wrote the codes to manipulate the copies. If there is any error, I can create another copy from the original.

•Second During cleaning, use the define-code-test framework and clearly document it

•Third Cleaning includes merging individual pieces of data according to the rules of tidy data. The result should be a high-quality and tidy master pandas DataFrame (or DataFrames, if appropriate).

Storing, Analyzing, and Visualizing Data for this Project

Store the clean DataFrame(s) in a CSV file with the main one named twitter_archive_master.csv. If additional files exist because multiple tables are required for tidiness, name these files appropriately. Additionally, you may store the cleaned data in a SQLite database.

Analyze and visualize your wrangled data in your wrangle_act.ipynb Jupyter Notebook. At least three (3) insights and one (1) visualization must be produced.

Reporting for this Project

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

•Create a >250-word written report called act_report.pdf that communicates the insights and displays the visualization(s) produced from your wrangled data. This is to be framed as an external document, like a blog post or magazine article, for example.

Conclusion

Data wrangling is a crucial skill that whoever handles data should be familiar with.

I have used Python programming language and some of its packages.

- For gathering data there are several packages that help scraping data off the web, that help using APIs to collect data (Tweepy for Twitter) or to communicate with SQL databases.
- It is strong in dealing with big data super-fast than Excel.
- It can deal with a multiple type of data (unstructured / structured).
- It is simple to document each single step and if needed re-run each single step.
- One can re-run analysis automatically every period.
- Handling, assessing, cleaning and visualizing of data can be programmatically using co