

Wrangle and Analyze Data Project

By: Ali Alsuhaibani

Introduction

This project is to put in practice what I learned in data wrangling data section from Udacity Data Analysis Nanodegree program and from other sources.

Project details

The tasks of this project are as follows:

- Gathering data
- Assessing data
- Cleaning data

Gathering data

This project data consists of three different dataset that were obtained as following:

- **Twitter archive file:** the `twitter_archive_enhanced.csv` was provided by Udacity and downloaded manually.
- **The tweet image predictions:** 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

Once the three tables were obtained, I assessed the data as following:

- 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`, `groupby`, etc).

Then I separated the issues encountered in quality issues and tidiness issues. Key points to keep in mind for this process was that original ratings with images were wanted.

Cleaning data

This part of the data wrangling was divided in three parts: Define, code and test the code. These three steps were on each of the issues described in the assess section.

First and very helpful step was to create a copy of the three original dataframes. I wrote the codes to manipulate the copies. If there was an error, I could create a new copy from the original.

Whenever I made a mistake, I could create another copy of the dataframes and continue working on the cleaning part.

There were a couple of cleaning steps that were very challenging. One of them was in the image prediction table. I had to create a 'nested if' inside a function in order to capture the first true prediction of the type of dog. The original table had three predictions and confidence levels. I filtered this into one column for dog type and one column for confidence level.

Conclusion

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 (much better than Excel).
- It can deal with a large variety of data (unstructured data like JSON (Tweets) or also structured data from ERP/SQL databases).
- It is easy to document each single step and if needed re-run each single step. Thus, one can leave a perfect audit trail (perfect for the accountant).
- One can re-run analysis automatically every period. Thus, we could actually re-run the dog analysis every month with much less effort now because I have set it up once.
- Handling, assessing, cleaning and visualizing of data is possible programmatically using code.