WeRateDogs Twitter Data Analysis

Wrangling Report

Kalman Heyn

Problem

Real-world data is messy. The process of assessing & cleaning data is called data wrangling. This project runs an analysis on the humorous twitter account @WeRateDogs. The data comes from three different sources – twitter_archive_enhanced.csv, image_predictions.tsv, and tweet_json.txt. Since these are real-world data sources, they each come with issues that will need to be assessed and cleaned before any analysis can be performed.

Data Gathering

I gathered the data from a given CSV, a website, and Twitter's Tweepy API. I used Tweepy to access the API and gather the JSON data for the tweets. I stored the JSON data in a text file, then loaded what I needed into a pandas dataframe.

Assessing & Cleaning

I audited the data by checking data types, value counts, lower/uppercase statuses, number of non-null entries, and numeric summaries. I combined (inner-joined) all three tables because each column is a feature of the individual tweet. I reshaped the dog stages (floofer, puppo, pupper, doggo) into a single column rather than multiple columns, and fixed names that were labelled incorrectly (lowercase, in 'text' column, etc). Numerators were fixed to be more within a reasonable range. The sources of each tweet was cleaned up to be easier to understand as well. Lastly, I converted several columns to new data types:

- Dog_stage to 1 column as a category
- Tweet_id, in_reply_to_status_id, in_reply_to_user_id to strings
- Timestamps to datatime objects

For each issue, I defined, coded, and tested each changes to make sure they were made correctly.

Storage

All the data was combined in to one master dataframe.

Conclusion

Data wrangling is a core skill that data analysts should be familiar with.

Proper data wrangling can benefit from the Python programming language and some of its packages. There are several advantages of these tools, compared to Microsoft Excel for example, that by many data scientists, including those at large tech companies like Facebook and Intuit, take advantage of:

- For gathering data, there are several packages that help scraping data off the web. These can be used as API’s to collect data (Tweepy for Twitter) and to communicate with SQL databases.

- Faster processing is needed when dealing with big data. Excel is great when working with under 1 million rows, but there is big data out there that Excel cannot handle

- Python can deal with a large variety of data, such as unstructured data like JSON (Tweets), and can also handle structured data, such as data from ERP/SQL databases.

- Due to Python’s Jupyter Notebook capabilities, is easy to document each single step and if needed re-run each single step. Thus, one can leave a trail for experimenters to follow when reevaluating the trials.

- Handling, assessing, cleaning and visualizing of data is possible programmatically and automatically using code.