WRANGLE AND ANALYZE DATA

WRANGLE REPORT

#WeRateDogs

As part of Udacity's Data Analyst Nanodegree, we get to work on a Data Wrangling Project. The goal is gathering data from a variety of sources and formats (csv, tsv, json files for examples), assessing its quality and tidiness, then cleaning it. We eventually get to showcase our wrangling efforts through analyses and visualizations.

The data come from 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.

1st step: GATHERING DATA

We gathered data from 3 sources:

- **Enhanced Twitter Archive**, which contains basic tweet data for all 5000+ of their tweets → twitter_archive_enhanced.csv
- **Additional Data via the Twitter API**, which contains retweet count and favorite count are two of the notable column omissions. Fortunately, this additional data can be gathered by anyone from Twitter's API → tweet_json.txt
- **Image Predictions File,** which is a table full of image predictions (the top three only) alongside each tweet ID, image URL, and the image number that corresponded to the most confident prediction (numbered 1 to 4 since tweets can have up to four images). → image_predictions.tsv

2nd step: ASSESSING DATA

This step allows us to identify quality and tidiness issues.
Low-quality data are considered as **dirty data** and have content issues. We have to assess at least height quality issues. As for **untidy data**, they are considered as **messy data** and are structural issues. We'll identify at least two of them.

There are two types of assessment:

- **Visual assessment**, which consists in scrolling through the data in Google Sheets or Excel to name a few - I first took a look at the cvs file in Google Spreadsheet.
- **Programmatic assessment,** for which we use code such as pandas' head, tail, describe, shape, sample, value_counts and info methods.

Here's a list of the issues that have been highlighted:

Quality Issues:

- Tweet_id, timestamp, sources, img_num and dog_stages need to be converted into the right datatype
- Sources to be clearly defined such as Twitter for iPhone, Twitter Web Client, Vine - Make a Scene & TweetDeck
- Dog name is not always accurate: 'a', 'actually', 'all', 'by', 'getting' etc
- Missing values in the dog stages column showing up as 'None'
- 183 retweets to be deleted
- In the text, we can notice some decimal numbers for the ratings (x5) ==> numerator part wrongly extracted
- Some numerators are higher than 10
- Some tweets don't include images
- Some breeds in p1, p2, and p3 (Image Prediction File) have upper cases (first letter).

Tidiness Issues:

- Merging the three dataframes into one using tweet_id with a (inner) join condition
- Joining the dog stages into a single column instead of four
- Numerous columns to be deleted.

3rd step: CLEANING DATA

The data have been cleaned thanks to the programmatic method.
With this method, we need to define (definition or instruction list) the cleaning task. Then, we code the issue to get it cleaned (drop, extract, islower, loc, etc., methods).
At the end, we test the dataset, visually or with code, to assure that the cleaning operations work correctly.

CONCLUSION

Through the data wrangling and analysis, we used many libraries such as pandas, NumPy, requests, tweepy, and json, which allow us to gather, assess, and clean the data.
Finally, we put the following documents together:

- wrangle_act.ipynb: code for gathering, assessing, cleaning, analyzing, and visualizing data
- wrangle_report.pdf: documentation for data wrangling steps: gather, assess, and clean
- act_report.pdf: documentation of analysis and insights into final data

- twitter_archive_enhanced.csv: file as given
- image_predictions.tsv: file downloaded programmatically
- tweet_json.txt: file constructed via API
- twitter_archive_master.csv: combined and cleaned data

Ressources

https://stackoverflow.com/questions/28384588/twitter-api-get-tweets-with-specific-id
http://docs.tweepy.org/en/v3.2.0/api.html#API
https://stackoverflow.com/questions/21308762/avoid-twitter-api-limitation-with-tweepy
https://wiki.python.org/moin/ForLoop
https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/tweet-object.html
https://stackoverflow.com/questions/28056171/how-to-build-and-fill-pandas-dataframe-from-for-loop/28058264
http://stackabuse.com/reading-and-writing-json-to-a-file-in-python/
https://stackoverflow.com/questions/7370801/measure-time-elapsed-in-python
http://docs.tweepy.org/en/v3.2.0/api.html#API
https://stackoverflow.com/questions/28384588/twitter-api-get-tweets-with-specific-id
https://wiki.python.org/moin/HandlingExceptions
https://stackoverflow.com/questions/8784396/python-delete-the-words-between-two-delimiters
https://docs.python.org/3.4/howto/regex.html
http://www.pythonlearn.com/html-007/cfbook012.html