Data Wrangling Documentation

Reem Alansary

September 10, 2020

Wrangling efforts for this project went through the three typical phases of *gathering, assessing,* and *cleaning*. In the gathering phase, the required data was acquired through three sources by three different methods. Assessment of the gathered data was performed in two ways to identify tidiness as well as quality issues and was performed iteratively. Methods for gathering and assessing data will be explained in detail in their own sections in this document. The third and final phase of wrangling, the cleaning phase, was broken down to small coherent steps as dictated by the *define, code, test* template; the steps were organized for each table such that closely linked actions were done in a single step, however some actions that may seem related were separated into more than one step as they were detected in separate iterations of the wrangling process.

1 Gathering

As written in the project requirements, data had to be gathered from three sources; these were an on-hand file, a file whose content had to be downloaded programmatically, and over a couple thousand json objects accessible through the Twitter API. The json objects were stored in a text file.

File Names in Order:

1. twitter-archive-enhanced.csv
2. image-predictions.tsv
3. tweet_json.txt

Libraries Used Exclusively for Gathering:

- `requests`
- `tweepy`
- `json`
- `time`
- `os`

Data from the on-hand file were simply read into a DataFrame named *ratings* using `pandas.to_csv()`. The second file was programmatically downloaded and read into another DataFrame using the same method with a tab separator and named *image_predictions*. After storing the json objects in a text file, they were reread and only the fields `[id, retweet_count, favorite_count]` were extracted to be put in a third DataFrame named *tweet_counts*; this DataFrame was built using similar dictionaries.

2 Assessing

Assessment of data for all tables was carried out both visually in the workspace Jupyter Notebook (wrangle_act.ipynb) as well as externally in Excel and programmatically via `pandas` functions and methods. All issues were documented in the workspace organized by *tidiness* and *quality* then by table names (*ratings, image_predictions, tweet_counts*). Later, during the cleaning phase after tidiness issues were cleaned, new and edits to existing quality issues were detected and documented appropriately signifying when they were discovered. After the wrangling process was finished and analysis of the cleaned data was being performed a new quality issue was observed, documented appropriately and cleaned. All issues found within the data summed up to 15.

3 Cleaning

The first step in this phase was to make copies of the existing DataFrame objects to begin cleaning, such that all cleaning operation were performed on *ratings_clean*, *image_predictions_clean*, and *tweet_counts_clean*. In order to clean for tidiness, some quality issues had to be resolved first and these are documented in order in the Jupyter Notebook (wrangle_act.ipynb). After all tidiness issues were resolved the remaining quality issues were cleaned as well. Each cleaning step is well documented through use of the *define, code, test* template. This phase used a wide variety of `pandas` functions and methods and some `numpy` utilities to render two sufficiently clean datasets, which are *ratings_clean* and *image_predictions_clean*; *tweet_counts_clean* was no longer necessary. Both tables were saved as **twitter_archive_master.csv** and **twitter_archive_image_predictions.csv** respectively.

In conclusion, 3 tidiness issues and 12 quality issues in total were detected and cleaned. The resulting datasets were used to generate visualizations and derive insights.