Project 2 Wrangling Data Report

The objective of this report is to present my wrangling efforts and thought process for this project. This report is divided into three main sections; Gathering, Assessing, and Cleaning.

1 Gathering Data

WeRateDogs Twitter archive data (twitter_archive_enhanced.csv) is a CSV file provided by Udacity and is downloaded directly and loaded into a pandas data frame using the pd.read_csv() method.

The tweet image predictions data (image-predictions.tsv) is a tsv file that is downloaded programmatically using the requests package in the following procedure:

- Import the requests library
- Create the request using requests.get(url)
- Assess the request and write to a file (image-predictions.tsv) in the data folder
- Load the downloaded data into a pandas data frame using pd.read_csv() and specify the delimiter as “\t”

The additional data for this project is gathered using the Twitter API. This additional data (tweet_json.txt) is a JSON file that contains tweet ids, retweet counts, and favorite(likes) counts for the tweets in the twitter_archive_ehanced.csv. The data is queried using the Tweepy library via the Twitter API. The JSON data for each tweet ID in the Twitter archive is written to a text file (tweet_json.txt) and stored in the data folder. To load the data in a data frame:

- The retweet_count and favorite_counts data in the tweet_json file is read and appended as dictionaries into an empty list (df_list).
- The list (df_list) is then converted into a pandas data frame using pd.DataFrame().
- Save the data to a CSV file (additional.csv)

Finally, I use the pandas' method (.head()) to verify whether the gathering process worked successfully.

2 Assessing Data

In this section, I employed both Visual and Programmatic assessing techniques to identify data quality and tidiness issues with the three datasets. Using a combination of for loops, custom functions and pandas methods such as .sample(), .head(), describe(), .info(), .shape(), .isnull(), and .duplicated(), I identified 10 quality issues and 2 tidiness issues. These issues are listed in the wrangle jupyter notebook document.

3 Cleaning Data

The quality and tidiness issues identified in the assessing phrase were corrected using the define, code, and test framework. First, I create copies of the three datasets before cleaning them. This helps to retain the original data. Below, I concisely present the cleaning process.

Tidiness #1:

- Using pd.melt(), put the dog stages columns under one column named dog_stage
- Drop the variable column that the above code creates and use .value_counts() to check if the code worked.

Quality #1:

- Using pandas' .replace() method, rename the "None" values in the name and dog_stage columns to NumPy's np.nan and test if the code worked.

Tidiness #2:

- Merge the twitter_archive_enhanced data with the additional data on tweet_id using pandas' .join() (left join)

Quality #3:

- Extract the date from the timestamp column using regular expression in pandas
- Convert the date (in string format) into DateTime format using pd.DateTime() and drop the timestamp column.

Quality #4:

- Find the index of tweets with rating_denominators greater than 10 and has images containing more than 1 dog and drop them using the pandas .drop() method.

Quality #5:

- Create a list containing the correct ratings of the identified indexes
- Create a function (rating_corrector) that corrects the ratings, loops through the list to make the corrections, and checks whether the code worked.

Quality #6 , #7, #8:

- Using the rating_corrector function, correct the rating_numerator of the identified indexes.
- Check if the code worked using the .loc() function.

Quality #2:

- Create a function, missing_values, that takes a data frame as input and displays the percentage of missing values in each column.

- Using the missing_values function, identify columns with high missing values and drop them using pandas' .drop()

Quality #9:

- Rename columns in the image-prediction data using pandas' .rename()

Quality #10:

- Using the pandas' .columns method, I identified the names of columns in the three datasets which may be irrelevant for my analysis, and drop them.

The project requires analyzing only original ratings that have images i.e., no retweets. Hence, I dropped all the observations which are retweets and replies from the twitter_archive_enhanced dataset. Finally, I combined the clean versions of the three datasets into one master dataset (twitter_archive_master.csv) using pd.merge().