Data Analyst Nanodegree Program

Udacity Nanodegree (In Collaboration with Kaggle)
Course-3: Data Wrangling
Project-3: Wrangle and Analyse Data (WeRateDogs)

1. Gathering the Data :

About the Datasets :

Twitter Archive-Extended - twitter_archive_extended.csv

The data for this project was gathered from multiple sources including flat-files(CSV/TSV) and APIs. The Twitter Archive Enhanced dataset (Extracted from twitter-user: @dog-rates) (https:// d17h27t6h515a5.cloudfront.net/topher/2017/August/59a4e958_twitter-archive- enhanced/ twitter-archive-enhanced.csv) contains over 5000+ Tweets and includes information like Comment, Ratings, Timestamp of individual tweet. This dataset is filtered to include only tweets with ratings. After filtering, it consists of 2,356 tweets. We can import this dataset into our project workspace aka Jupyter Notebooks (Anaconda Environment) directly by using the Pandas Library.

Twitter Additional Data API - tweet_json.txt

Next, we import additional data from the twitter-user (@dog-rates) by making use of 'Tweepy' Library (Python Library for accessing Twitter API) (http://www.tweepy.org). This additional information includes retweet_count and favorities_count. We also have an option to include additional fields in our analysis. In order to use this library we must first create, Twitter Developer account and Create Dummy Application that will allow us to make use of confidential information or credentials (like Access Key, Access Token, Consumer Key, Consumer Token). Twitter also restricts amount of data that can be extracted from its servers in a given time-span. We can set appropriate parameters to control this. Finally, live data is converted into format suitable for analysis.

Twitter Image Predictions - image_predictions.tsv

In addition to above, we are provided with image predictions data that consists information like JPG URL, Image Number, Prediction. (https:// d17h27t6h515a5.cloudfront.net/topher/ 2017/August/599fd2ad_image-predictions/ image-predictions.tsv). We can extract this data programmatically using the 'Requests' Library (http://www.python-requests.org). We can read this tab-separated file using appropriate parameters (like sep='\t').

2. Assessing the Data :

Visual Assessment :

Initial observation of the data in spreadsheet programs (like Excel, Google Sheets) indicates quality and tidiness issues. For instance, we can find that:

- Unnecessary HTML Tags in the Source Column like Twitter for iPhone
- Missing Values throughout the columns such as NaN or None
- Dog Stages like doggo, floofer, pupper, puppo in multiple columns instead of one.
- Duplicate data in the form of columns like retweeted_status_user_id, retweeted_status_id.

Programmatic Assessment :

Using the various methods and functions in the pandas library, we can spot quality and tidiness issues. Using df.info() method we can determine if there are any missing values. Using, df.col.value_counts() we can check range of values. Using these methods and few more, we could determine several quality issues such as:

- rating_numerator: has values higher than 10
- rating_denominator: has values other than 10
- text: has retweets beginning with 'RT'
- in_reply_to_status_id, in_reply_to_user_id: has values other than NaN
- in_reply_to_status_id, in_reply_to_user_id: has different data-types
- name: has erroneous dog names, shorter in length and size
- tweet_id: has several missing tweets in image predictions
- source: has extra HTML Tag information

Using df.head() and df.sample() methods, we could determine which columns existed in the data frames and using these methods and several others, we could determine several tidiness issues such as:

- Joining archives, predictions and images dataset into single data frame using appropriate matching column and df.merge() function.
- Dropping unnecessary columns from the dataset using df.drop() function.

3. Cleaning the Data :

Before, beginning with the cleaning phase, different copies of original datasets were created like archives_clean, predictions_clean and api_clean. Issues were separated based on Tidiness and Quality rules. Data Cleaning was performed in three-step process namely, Define/Code/Test.

Storing the Data :

After data-cleaning step, it was time to store data into 'Master-Dataset' where it can be analysed and visualised clearly. Using, df.to_csv(index=False) we were able to export the final data frame into text file. We later read this file using pd.read_csv() function to begin our final analysis.