This project is to implement the lessons I learned from the Data Wangling section of Udacity's Nanaodegree program. The repository I'm using is the Twitter account @WeRateDogs, which is known for its humorous comments about dogs. The ratings it gives are usually based on a 10-point scale.

Gathering Data

The data used for this project consisted of three different datasets that were obtained as following:

The first dataset that Udacity provided was a .csv file that contains the basic details about WeRateDogs users' tweets. It was downloaded manually and uploaded manually to the Jupyter Notebook server and then loaded into the notebook by using the pandas read_csv() function to create a data frame called df_twitter_enhanced.

The second dataset was an image prediction file, which was hosted on Udacity's server. It contained over 2,000 predictions made by an unsupervised neural network. It was acquired by using the get() function of the requests library via it's URL and saved to a data frame called df_predict using the pd.read_csv command.

The third dataset contains data that was scraped using the Tweepy Twitter API. An attempt was made to obtain this information programmatically but due to authorization issues with the Twitter API I was provided a text file by an instructor which I loaded into a pandas data frame by converting the txt file to a list by iterating through the txt file and appending the list. I then extracted the id, retweet_count and favorite_count and saved as a pandas data frame.

Assessing Data

Once the data was loaded into it's respective data frame I inspected the files visually and then assessed using a variety of programmatic techniques. I discovered the following issues that needed to be cleaned.

Quality issues

1. Incorrect datatypes for the timestamp and retweeted_status_timestamp columns in the df_twitter_enhanced data frame. Datatype is int instead of datetime.

   Solution: Datatype was changed to datetime with the .astype(str) command

2. Incorrect datatype for the tweet_id in all 3 dataframes. Datatype is int when it should be string because it is categorical data.

   Solution: Datatype was changed to string with the .astype(str) command

3. Missing values in the name column are "None" instead of nan in the df_twitter_enhanced data frame.

   Solution: Replaced all the null values that are written as "None" with Nan using the to_replace command.

4. The df_predict data frame has inconsistent use of capitalization in the p1,p2 and p3 columns that affects readability.

   Solution: Changed all the p1, p2, and p3 values to lower case using the .str.lower() command

5. The df_twitter_enhanced data frame contains retweets that are not needed.

   Solution: Removed retweets and then removed unused columns by using the .drop() command

6. 'name' values in the df_twitter_enhanced dataset that are all lowercase are invalid entries (i.e. not dog names). Confirmed that the dogs names that started with a lower case are not dog names.

   Solution: Removed all the values in the name columns in the df_twitter_enhanced data frame that started with lower case letters and replaced with the string none. This was achieved by extracting all the names with lowercase letters and adding them to a list using a loop method. Then those values were replaced with None and then the None was replaced with NaN.

   Had I been more experienced with data wrangling I would have completed this task before doing number 3 to avoid having to redo some of the cleaning I already done.

7. Column names in the predict_df data frame not descriptive enough.

   Solution: Renamed columns with a more descriptive name using the .rename() command.

8. Source column in the twitter_enhanced data frame is in HTML-formatted string not a normal string.

   Solution: Corrected to a normal string using the .str.extract command to remove extra characters.

Tidiness issues

1. 1.The column label for tweet ids should be consistent across the 3 datasets so they can be joined correctly.

   Solution: Change the column label from ‘id’ to ‘tweet_id’ using the .rename command. I had to recast the tweet_id as a string for the tweets_df_clean dataset in order to be able to join it in the next task. Had I been more experience I would have done the recasting as a string as my last task to avoid having to repeat it.

2. retweet_count and favorite_count should be part of df_twitter_enhanced data frame.

   Solution: Joined all 3 datasets into one master dataframe using the pd.merge command to perform an inner join on the “tweet_id” variable.

Storing Data

The dataset was then saved as twitter_archive_master.csv" using the .to_csv command