WE RATE DOGS DATA

WRANGLING REPORT

This report was created to document the wrangling process taken in the creation of this project.

According to the Udacity AL-X Data analyst Nano-degree lesson on wrangling, data wrangling is the process of gathering your data, assessing its quality and structure, and cleaning it before you do things like analysis, visualization, or build predictive models using machine learning.

Wrangling is an iterative process that consists of three steps:

Gathering

Assessing

Cleaning

My report will be divided into these 3steps.

GATHERING

I downloaded the first dataset a CSV file from the Udacity portal name the twitter-archive-enhanced.csv which was the first dataset I downloaded.

The second dataset I downloaded was downloaded programmatically using the Request library to download the dataset from a url and stored in my Udacity project folder.

The third dataset was downloaded from a Json text using the pd.read_json function to read the json file into a dataframe line by line.

At the end of this process I had 3 files gathered stored in 3 dataframes.

- Twitter-archive-enhanced.csv
- Image_predictions.tsv
- Tweet-json-txt

ASSESSING

In this step, I viewed the datasets both visually and programmatically observing the completeness, validity, accuracy and consistency of the datasets. During this process I noted some quality issues from the datasets namely;

1. Timestamp column wasn't in time format

2. Rating denominator should be 10

3. The tweet_id column should be a string not an integer

4. Some dog names in the df_twitter column are in lower case and they aren't actual names

5. We needed only original tweets and not retweets or replies

6.The rating denominator column are unnecessary as well as the rating_denominator','in_reply_to_status_id','retweeted_status_id','retweeted_status_user_id','retweeted_status_timestamp','in_reply_to_user_id' columns.

Also some Tidiness issues;

- The puppo, doggo,floofer and pupper column should be under one column called dog_stages instead of 4.
- The twitter archived data (df_twitter) should merge with the jsondata (df) and the image predictions dataset (df_image).
- Merge the archive data with the image predictions dataset to get the image predictions url and number of images.

CLEANING

During this process I solved all quality and tidiness issues that were highlighted in the accessing stage. Some of these steps include;

- Use the pandas to time function to convert the timestamp column from string to datetime.
- Bring out all values of the denominator not equal to 10 and convert them to 10.
- Use split function to extract from the source the particular application used for the tweets excluding the anchor tags and store them in a new column called application.
- Use the astype function to convert the tweet_ids and ids column to a string.
- Create a mask to select those dog names in lowercases and replace them with 'None'
- Subset only colums where the retweet_id and reply_id is null to remove retweet and replies from the dataset.
- Drop all rating denominator column as it doesn't matter as they are all 10 and also drop the reply status, retweet status columns as the rows have been dropped.
- The ratings_numerator column should be renamed using the .rename function
- Subset the 'id',"favorite_count",'retweet_count' columns and save them to the df_clean variable.
- Use the rename function to rename the df_clean dataframe column to tweet_id so it can merge with the archive data frame(df_twitter_clean).
- Add the doggo', 'floofer', 'pupper', 'puppo' columns together and attach it to a column named dog stage,use the replace function nested in the apply function to replace all rows in the dog_stage column that have the string None and replace with an empty string. This way we can get the various dog stages and then drop the columns

- Use the pd.merge using the left join to merge both the json dataset and the image prediction dataset to the archive dataset.