Data Wrangling Report

Project objectives

The project main objectives were:

- Perform data wrangling (gathering, assessing and cleaning) on provided three sources of data.
- Store, analyse, and visualise the wrangled data.
- Reporting on 1) data wrangling efforts and 2) data analyses and visualisations.

Step 1: Gathering Data

In this step we gather data from three sources.

1. The WeRateDogs Twitter archive is a file given to me. I manually downloaded "twitter_archive_enhanced.csv".
2. The tweet image predictions "image_predictions.tsv"is a file hosted in Udacity servers. I downloaded it using the Requests library.
3. I downloaded the raw JSON tweets from the twitter archive using the Tweepy library, and the Twitter API. I then write those JSON formatted tweets to a "tweet_json.txt" file.

Step 2 and 3: Assessing and Cleaning Data

In this step we assess, observe our assessment and clean our assessment for all the data from our datasets

- Quality

DataFrame	Observation	Solution
df_archive_clean	df_archive contains rows for tweets, retweets and replies.	Delete or filter only rows where retweeted_status_id or in_reply_to_user_id equal null.
	in_reply_to_status_id', 'in_reply_to_user_id', 'retweeted_status_id', 'retweeted_status_user_id', and 'retweeted_status_timestamp' are redundant columns	Drop/ Remove redundant columns ('in_reply_to_status_id', 'in_reply_to_user_id', 'retweeted_status_id', 'retweeted_status_user_id', 'retweeted_status_timestamp') using DataFrame.drop
	The dataset contains tweets with no media.	Use the expanded_urls column to filter out all rows where expanded_urls equals Nan.
	The source column has html formatted text as entries.	Use str.extract to extract the source device for each tweet.
	The ratings with decimal values are incorrectly extracted	Re-extract both the numerator and denominator from the text column using pd.Series.str.extract using the '(\d+\.?\d*/\d+\.?\d*)' regex pattern and convert the columns from object type to a float.
	Dog names not fully extracted. tweet_id 778408200802557953 and 740373189193256964 have different structures for the tweet.	Manually set the names for the two dogs in their respective rows using pd.DataFrame.loc

	The name column contains 'None', 'a', 'the', 'an', 'very', 'quite', 'my', etc as dog names.	Select all the rows with lower case and assign using '.loc[row_indexer, col_indexer] = np.nan' and Replace None with np.nan using Nan using pd.Series.replace
	The timestamp column is stored as a string object.	Convert the column from string to datetime object using pd.to_datetime
	Need a year_month column from the timestamp column (for ease analysis and visualisation).	Create the column using pd.Series.apply and use strftime('%Y-%m')
df_predictions_clean	'p1', 'p2', and 'p3' contain entries with '-' and '_' between the predictions.	Remove the symbols using str.replace

- Tidines

DataFrame	Observation	Solution
df_archive_clean	doggo', 'floofer', 'pupper', 'puppo' columns are contain dog stages	Create a single 'dog_stage' column using string concatenating on the columns ('doggo', 'floofer', 'pupper', 'puppo'). Replace 'None' with Nan using pd.Series.replace and finally Drop redundant columns using pd.DataFrame.drop
df_clean	Multiple DataFrames were created for each datasets	Merge all dataset into one single dataset called df_clean using DataFrame.merge

The final data after cleaning. I focused on the column I need for analysis.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1963 entries, 0 to 1962
Data columns (total 24 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   tweet_id            1963 non-null   int64
 1   timestamp           1963 non-null   datetime64[ns, UTC]
 2   source              1963 non-null   object
 3   text                1963 non-null   object
 4   expanded_urls       1963 non-null   object
 5   rating_numerator    1963 non-null   float64
 6   rating_denominator  1963 non-null   float64
 7   name                1344 non-null   object
 8   year_month          1963 non-null   object
 9   dog_stage           292 non-null    object
 10  retweet_count       1963 non-null   int64
 11  favorite_count      1963 non-null   int64
 12  followers_count     1963 non-null   int64
 13  jpg_url             1963 non-null   object
 14  img_num             1963 non-null   int64
 15  p1                  1963 non-null   object
 16  p1_conf             1963 non-null   float64
 17  p1_dog              1963 non-null   bool
 18  p2                  1963 non-null   object
 19  p2_conf             1963 non-null   float64
 20  p2_dog              1963 non-null   bool
 21  p3                  1963 non-null   object
 22  p3_conf             1963 non-null   float64
 23  p3_dog              1963 non-null   bool
dtypes: bool(3), datetime64[ns, UTC](1), float64(5), int64(5), object(10)
memory usage: 343.1+ KB
```