Wrangle and Analyze Project (weRateDogs) - Wrangle Report

- ## Project Objectives

  - Gather data using 3 different approaches.
  - Assess data and identify quality and tidiness issues.
  - Clean the identified issues and iterate.

- ## Step1: Gathering Data

```
archive = pd.read_csv('twitter-archive-enhanced.csv')
```

- The WeRateDogs Twitter archive

read the given file twitter_archive_enhanced.csv .

```
url = 'https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predictions/image-predictions.tsv'

with open('image_predictions.tsv' ,'wb') as file:
    file.write(requests.get(url).content)


image_prediction = pd.read_csv('image_predictions.tsv' , sep='\t')
```

- The tweet image predictions

Used request library requests.get.(url) to download this data from the url: https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predictions/image-predictions.tsv Into a file then reading the data from the file into image_predictions data frame.

- Additional data from the Twitter API

```
tweet_data=[]
with open('tweet-json.txt', 'r') as file:
    # read each line of the file
    for line in file:
        tweet_data.append(json.loads(line))

tweet_df=pd.DataFrame(tweet_data)
```

This skill is not applied yet, for now I used the given tweet-json.txt file
Which contains data in the form of json dictionaries, I looped through each line in the text file and used json.loads for each line to convert it to object format and append to tweet_data list and finally convert the tweet_data list into tweet_df data frame.

- ## Step2: Assessing Data

At this point there are 3 data frames `archive`, `image_prediction`, and `tweet_df`. Applied programmatic assessment on all three data frames and here are the results:

1. archive :

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2356 entries, 0 to 2355
Data columns (total 17 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   tweet_id                    2356 non-null   int64
 1   in_reply_to_status_id       78 non-null     float64
 2   in_reply_to_user_id         78 non-null     float64
 3   timestamp                   2356 non-null   object
 4   source                      2356 non-null   object
 5   text                        2356 non-null   object
 6   retweeted_status_id         181 non-null    float64
 7   retweeted_status_user_id    181 non-null    float64
 8   retweeted_status_timestamp  181 non-null    object
 9   expanded_urls               2297 non-null   object
 10  rating_numerator            2356 non-null   int64
 11  rating_denominator          2356 non-null   int64
 12  name                        2356 non-null   object
 13  doggo                       2356 non-null   object
 14  floofer                     2356 non-null   object
 15  pupper                      2356 non-null   object
 16  puppo                       2356 non-null   object
dtypes: float64(4), int64(3), object(10)
memory usage: 313.0+ KB
```

This data frame has 2356 rows and 17 columns, for `name` column there are 745 rows with "None“ name 55 rows with “a” name and some rows has other non valid names like “the” , “just” and more. `rating_denominator` column isn’t always 10 and there are some ratings with 0 This data frame has no duplicated rows.

2. image_prediction:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2075 entries, 0 to 2074
Data columns (total 12 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   tweet_id  2075 non-null   int64
 1   jpg_url   2075 non-null   object
 2   img_num   2075 non-null   int64
 3   p1        2075 non-null   object
 4   p1_conf   2075 non-null   float64
 5   p1_dog    2075 non-null   bool
 6   p2        2075 non-null   object
 7   p2_conf   2075 non-null   float64
 8   p2_dog    2075 non-null   bool
 9   p3        2075 non-null   object
 10  p3_conf   2075 non-null   float64
 11  p3_dog    2075 non-null   bool
dtypes: bool(3), float64(3), int64(2), object(4)
memory usage: 152.1+ KB
```

image_prediction dataframe has 2075 rows and 12 columns which is less than the archive dataframe
no missing data in this data frame

3. tweet_df:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2354 entries, 0 to 2353
Data columns (total 31 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   created_at                      2354 non-null   object
 1   id                              2354 non-null   int64
 2   id_str                          2354 non-null   object
 3   full_text                       2354 non-null   object
 4   truncated                       2354 non-null   bool
 5   display_text_range              2354 non-null   object
 6   entities                        2354 non-null   object
 7   extended_entities               2073 non-null   object
 8   source                          2354 non-null   object
 9   in_reply_to_status_id           78 non-null     float64
 10  in_reply_to_status_id_str       78 non-null     object
 11  in_reply_to_user_id             78 non-null     float64
 12  in_reply_to_user_id_str         78 non-null     object
 13  in_reply_to_screen_name         78 non-null     object
 14  user                            2354 non-null   object
 15  geo                             0 non-null      object
 16  coordinates                     0 non-null      object
 17  place                           1 non-null      object
 18  contributors                    0 non-null      object
 19  is_quote_status                 2354 non-null   bool
 20  retweet_count                   2354 non-null   int64
 21  favorite_count                  2354 non-null   int64
 22  favorited                       2354 non-null   bool
 23  retweeted                       2354 non-null   bool
 24  possibly_sensitive              2211 non-null   object
 25  possibly_sensitive_appealable   2211 non-null   object
 26  lang                            2354 non-null   object
 27  retweeted_status                179 non-null    object
 28  quoted_status_id                29 non-null     float64
 29  quoted_status_id_str            29 non-null     object
 30  quoted_status                   28 non-null     object
dtypes: bool(4), float64(3), int64(3), object(21)
memory usage: 505.9+ KB
```

`tweet_df` data frame has 2354 rows and 31 columns , there are missing data in some columns.

When we get all 3 data frames together we can see that `tweet_id` , `source` , `in_reply_to_status_id` and `in_reply_to_user_id` are duplicated columns.

Quality & Tidiness issues:

`archive` table

- `in_reply_to_status_id` `in_reply_to_user_id` `retweeted_status_id` `retweeted_status_user_id` `retweeted_status_timestamp``source` columns are not needed.
- Removing the above columns does not remove the related rows , related rows need to be dropped first.
- validity issue in `rating_denominator` in some rows its not equal to 10.
- `names` with "a" , "not" , "my" , "an", "the" and "just" values.
- misssing `names` (can't clean).
- some `names` are lowercased.
- `timestamp` is string and ends with `+0000` which makes it not convertable (and should be `datetime`).
- alot missing in the `doggo`,`floofer`,`pupper`,`puppo` columns.
- `Text` data includes short url at the end.

`image_prediction` table

- some predictions are not even animals .
- some predications are lowercased.
- some predictions are separated using underscores.
- `img_num` column is not needed.

`tweet_df` table

- there are too many columns that are not needed.

Tidiness

- In `archive` table `doggo`,`floofer`,`pupper`,`puppo` should be one column `dog_stage`.
- `image_prediction` and `tweet_df` should be part of `archive` table.

- ## Step3: Cleaning Data

Quality issues cleaning:

`archive` table

- Removed rows that has non null values in each of these columns: `retweeted_status_id` , `retweeted_status_user_id` and `retweeted_status_timestamp`.
- Dropped `in_reply_to_status_id`, `in_reply_to_user_id`, `retweeted_status_id`, `retweeted_status_user_idretweeted_status_timestamp`,`source` columns.
- Set all `rating_denominator` to 10.
- Replaced every `a`,`an`,`not`,`my`,`just` name value with `None`.
- Removed `+0000` from the end of the `timestamp` string then convert it to datetime type.
- Merged `doggo`,`floofer`,`pupper`,`puppo` columns in one column and removed every `None` string.
- fix `dog_stage` some rows have two dog stages, other rows are empty.
  For rows that has 2 dog stages: find out which one is right from `text` and replace.
  For empty rows means that there is no stage mentioned in the `text` fill the empty cells with `None` value.
  and finally, drop `doggo`,`floofer`,`pupper`,`puppo` columns.
- Replaced every url in `text` column with blank.

`image_prediction` table

- capitalized predictions in `p1`,`p2` and `p3`.
- Replaced each underscore in prediction columns with space.
- Dropped `img_num` column.

`tweet_df` table

- Removed all columns from `tweet_info_clean` only kept `id`,`retweet_count`,`favorite_count` columns.
- Changed label of `id` column to `tweet_id` to match other dataframes.

Tidiness issues cleaning:

- Merge `tweet_info_clean` datafram with `archive_clean` by the tweet_id.
- Merge `image_prediction_celan` datafram with `archive_clean` by the tweet_id.
- Removed rows with missing `jpg_url` by making empty cells nulls then drop the nulls, these empty cells occurred after merging to due to data frames size mismatch.

- ## Step4: Storing Data

```
archive_clean.to_csv('archive_clean.csv' , index=False)
```

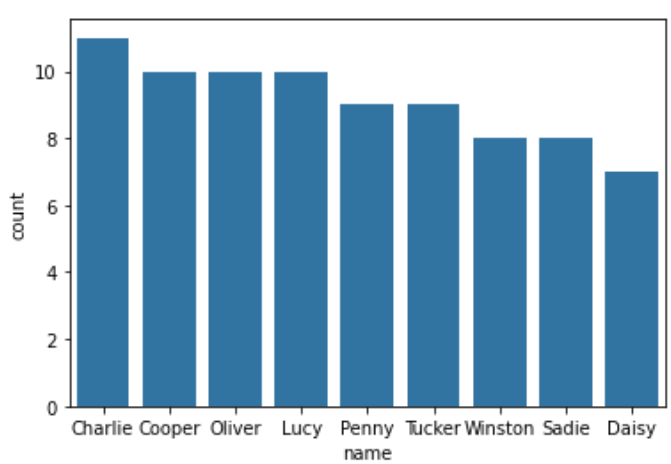
Saved the master data frame as a csv file.

- ## Step5: Analyzing and Visualizing Data

  - Top 5 rated dogs

	tweet_id	text	timestamp	rating_numerator	rating_denominator	name	dog_stage	retweet_count	favorite_count	
804	749981277374128128	This is Atticus. He's quite simply America af....	2016-07-04 15:00:45	1776	10	Atticus	None	2772.0	5569.0	https://pt
1895	670842764863651840	After so many requests... here you go.\n\nGood...	2015-11-29 05:52:33	420	10	None	None	4324.0	7989.0	https://pt
942	731156023742988288	Say hello to this unbelievably well behaved sq...	2016-05-13 16:15:54	204	10	None	None	1434.0	4196.0	https://pt
733	758467244762497024	Why does this never happen at my front door.....	2016-07-28 01:00:57	165	10	None	None	2539.0	5316.0	https://(
1600	677716515794329600	IT'S PUPPERGEDDON. Total of 144/120 ...I think	2015-12-18 05:06:23	144	10	None	None	1104.0	3323.0	https://pb:

  - Most popular dog names



Charlie is the most used dog name.

- Most Liked tweet

	tweet_id	text	timestamp	rating_numerator	rating_denominator	name	dog_stage	retweet_count	favorite_count	
348	822872901745569793	Here's a super supportive puppo participating ...	2017-01-21 18:26:02	13	10	None	puppo	48265.0	132810.0	https://pbs.twimg.co

This is the most liked dog it has no name but its a puppo and he's a `Lakeland terrier`