The following issues with the data were identified. Each cleaning process was defined. Most codes are provided following the issues, except for the very basic ones because of the word count limit:

Quality

'archive' table

- Tweet ID, in_reply_to_status_id, in_reply_to_user_id are integers not strings
  - Define: turned each columns into string values
  - Code:
    - *archive_clean.tweet_id = archive_clean.tweet_id.astype(str)*
    - *img_pred_clean.tweet_id = img_pred_clean.tweet_id.astype(str)*
- In the case where the rating has decimals, only the digits behind the decimal points were picked up
  - Define: there are only a few of these situations and they are fairly easy to identify so I manually corrected the values
- Some tweets rate dogs in group and used total rating
  - Define: identify these situations and
  - Code:
    - *group_dog_list = [433, 902, 1120, 1202, 1228, 1254, 1274, 1351, 1433, 1634, 1635, 1779, 1843]*
    -
    - *for id in group_dog_list:*
    - *num_dog = archive_clean.at[id,'rating_denominator']/10*
    - *archive_clean.at[id,'rating_numerator'] = archive_clean.at[id,'rating_numerator']/num_dog*
    - *archive_clean.at[id,'rating_denominator'] = archive_clean.at[id,'rating_denominator']/num_dog*
- The wrong numbers are picked up as ratings in some tweets, e.g. 9/11, 7/11
  - Define: identify and drop those rows
- Tweet 832088576586297345 has irrelevent info and the wrong ratings were picked up
  - Define: drop row
- 'None' as a string is in place of null-object in columns name and all three dog types
  - Define: Replace all the ‘None’ value with null value
  - code :
    - *archive_clean.replace('None', np.NaN, inplace = True)*
- Words such as 'a', 'this' were recognized as names
  - Define: after scanning the values, those none name values don’t start with capital letters. I want to identify the words that don’t start with capital letters and replace them with null value. This took quite some effort, as I found out that NaN was recognized as a float object and the other names were string objects. I used a

try/except in a for loop to collect all the tweet IDs for the wrong names and then used this list of IDs to change those values.

    - Code:

```
wrong_name = []
new_df.reset_index()

for i in range(0,len(new_df)):
    try:
        if new_df['name'][i].istitle():
            pass
        else:
            wrong_name.append(new_df['tweet_id'][i])
    except:
        Pass
```

- The plural forms of dog types were not recognized
    - Define: solved along with a tidiness issue that is discussed in the following
- Timestamp is a str object not a datetime object
    - Define: Turn the column into a datetime object
    - code :

```
archive_clean['timestamp'] = pd.to_datetime(archive_clean['timestamp'])
```

'img_pred' table

- Tweet ID is an integer not a string
    - Define: turned into a string like above

Tidiness

- Dog types: doggo, pupper, puppo, floofer should be combined into one column
    - Define: Created a new column dog_type that contains all the dog type values. Attempted a iterrows() function to directly change the column values where dog types are identified but cell values could not be altered in a np.iterrows() loop (only the copy of the data frame is altered in the function). Eventually I used an algorithm to pick up all tweet IDs where the dog types are identified, put them into a list and used another line of code to change those values using filter. Example for one type of dog is provided
    - Code:

```
archive_clean['dog_type'] = np.NaN

puppo = []

for row in archive_clean.iterrows():
    if row[1]['text'].lower().find('puppo') > 0 :
```

            - *puppo.append(row[1]['tweet_id'])*
            -
            - *puppo_mask = np.in1d(archive_clean['tweet_id'], puppo)*
            - *archive_clean.loc[puppo_mask, 'dog_type'] = 'puppo'*
- The three tables should be combined into one
    - Define: combine all three tables using pd.merge function
    - Code:
        - *new_df = pd.merge(archive_clean, tweet_counts, on='tweet_id')*
- Some of the columns are irrelevant and can be dropped
    - Define: dropped those columns