# Data Wrangling Report:

By\ Amr Hesham Maali

The following report illustrates the steps taken to successfully wrangle the dataset provided in the Data Wrangling project.

## Data Gathering:

- Data were gathered from 3 different sources:
- 1st source was the .csv file containing the data for the twitter archive of tweets.
- 2nd source was programmatically from the link provided for the image predictions dataset into a .tsv and then reading it into a data frame.
- 3rd source was the .json file of Twitter API Tweepy data extract. (access to API directly wasn't available as an option - code was added to the wrangle_act.ipynb).

## Data Assessment:

- Various visual and programmatic assessment techniques have been used and the found issues were documented in the same cells they were found inside the notebook to facilitate readability of the code.
- The found problems were also reported inside the notebook at the end of the assess phase then again at the head of each cleaning section.

## Data Cleaning:

- The data was first cleaned for unwanted data and completeness issues to process the final required pool of data together without need to redo any steps.
- Tidiness issues were then fixed to remove structural hurdles that hinder proper analysis of the data.
- Quality issues were adjusted last and the scheme of define, code, test was used to document every step of the cleaning process.
- The table below has a summary of all the cleaning efforts.

## Project Output files:

- The project produced 2 master data files:
  1- twitter_archive_master.csv
  2- image_predictions_master.csv
- Wrangle_report.pdf (we are here now!)
- Act_report.pdf (Containing analysis summary in a presentation form).

➢ All the above-mentioned file were saved in the directory of Udacity project workspace.

## Data Assessment & Cleaning Summary Table:

| Issue Type | Table | Issue | Solution |
|---|---|---|---|
| **Tidiness Issues** | archive_df table | 1- column headers are values and not variables. ('doggo', 'floofer', 'pupper', 'puppo') | The data were cleaned for the proper dog stages, removed none values and compiled into one column for a proper structure to support analysis |
| **Tidiness Issues** | image_predictions_df table | 2- column headers are values and not variables. (p1,p2,p3/conf/_dog) | Utilized pd.wide_to_long() to reshape the table in a more proper format for analysis. |
| **Tidiness Issues** | api_df table | 3- Better as part of the archive_df table | Solved by merging to the archive_clean_df dataframe. |
| **Quality Issues – Unwanted Data** | archive_df table | 1- 181 retweets. *(unwanted data)* | Utilized the retweet info columns to drop these records then dropped the columns. |
| **Quality Issues – Unwanted Data** | archive_df table | 2- 78 replies. *(unwanted data)* | Utilized the replies info columns to drop these records then dropped the columns. |
| **Quality Issues – Unwanted Data** | archive_df table | 3- Some columns not necessary after removing retweets & replies. | Utilized pd.drop to remove columns that weren't necessary for analysis to facilitate visual inspection of table during analysis. |
| **Quality Issues – Unwanted Data** | archive_df table | 4- 2075 tweets only have images in `image_predictions_df`. *( we only want those with pictures) (unwanted data)* | Utilized merging of predictions and tweets data frames to drop the records for tweets that have no pictures. |
| **Quality Issues – Completeness issues** | archive_df table | 5- 745 names as none, 55 as a. | Replaced 'none' values with Nan and 'a's with Nan. |
| **Quality Issues** | archive_df table | 6- some tweets have dog classification as none instead of Nan. | Records were cleaned while fixing the tidiness issue. |

| | | | |
|---|---|---|---|
| **Quality Issues** | archive_df table | 7- some rating numerators are missing (ones with decimals) | Re extraction of the decimal ratings was done |
| **Quality Issues** | archive_df table | 8- erroneous datatypes, 'tweet_id', 'timestamp'. Some columns not necessary for analysis. | Utilized astype() to correct all the faulty data types. |
| **Quality Issues** | image_predictions_df table | 9- erroneous data types. | Utilized astype() to correct all the faulty data types |
| **Quality Issues** | image_predictions_df table | 10- inconsistent capitalization of first letter in the names of dog breeds predicted | Utilized capitalize() method to fix this problem. |
| **Quality Issues** | api_df table | 11- errenous data types ('retweet_count','favorite_count') . | Utilized astype() to correct all the faulty data types. |