# act_report

March 12, 2023

### 0.1 Report: act_report

- Create a **250-word-minimum written report** called "act_report.pdf" or "act_report.html" that communicates the insights and displays the visualization(s) produced from your wrangled data. This is to be framed as an external document, like a blog post or magazine article, for example.

## 1 We Rate Dogs Tweet Analysis

***A look at dog rating tweets by `@weratedogs` on twitter.***

### 1.1 Introduction

The data wrangling project was to gain insights into the possibility of patterns emergent in the we rate dogs twitter account's dog rating tweets.

First we import the libraries that will be used during the data analysis

```
In [1]: import pandas as pd
```

```
In [2]: df = pd.read_csv('twitter_archive_master.csv')
```

### 1.2 Analyzing and Visualizing Data

During the analysis process, we will be taking a look at the data, its properties, and creating visualisations to infer information about the emergent properties and potential relationships.

#### 1.2.1 Insights:

During the analysis the folowing insights were deduced:

1. Most tweets only have one image attached to them.
2. Most images are predicted by the model to be golden retrievers.
3. Tweets with only one image have the highest like and retweet counts.
4. Golden Retrievers have the highest number of retweets and favourites.
5. Samoyed has more likes and retweets than chow and pug besides being in fewer tweet images.

6. There is a strong positive correlation between retweet count and favourite count.

To conclude upon the aforementioned insights, the following functions were used to make descriptive analytics and visualisations:

```
In [3]: df[['favorite_count', 'retweet_count']].corr()
```

```
Out[3]:                favorite_count  retweet_count
        favorite_count        1.000000       0.861616
        retweet_count         0.861616       1.000000
```

We can see from above that there is a strong positive correlation between the number of retweets and the number of favourites that a tweet got.

```
In [4]: df.describe()
```

```
Out[4]:            tweet_id      img_num      p1_conf       p2_conf       p3_conf  \
        count  2.056000e+03  2056.000000  2056.000000  2.056000e+03  2.056000e+03
        mean   7.377004e+17     1.204280     0.594386  1.347672e-01  6.040199e-02
        std    6.751437e+16     0.563359     0.271242  1.007437e-01  5.100277e-02
        min    6.660209e+17     1.000000     0.044333  1.011300e-08  1.740170e-10
        25%    6.762329e+17     1.000000     0.363272  5.389383e-02  1.623400e-02
        50%    7.110026e+17     1.000000     0.587797  1.186345e-01  4.947920e-02
        75%    7.928013e+17     1.000000     0.844247  1.956673e-01  9.215672e-02
        max    8.924206e+17     4.000000     1.000000  4.880140e-01  2.734190e-01

               rating_numerator  rating_denominator  favorite_count  retweet_count
        count       2056.000000         2056.000000     2056.000000    2056.000000
        mean          12.265078           10.516051     7228.888132    2310.704280
        std           40.867720            7.210005    10936.676602    4038.860995
        min            0.000000            2.000000        0.000000      11.000000
        25%           10.000000           10.000000     1376.500000     490.000000
        50%           11.000000           10.000000     3168.500000    1088.500000
        75%           12.000000           10.000000     8975.000000    2663.500000
        max         1776.000000          170.000000   141298.000000   69157.000000
```

from the dataset description we can see the following: * The number of embeded images in a tweet ranges from 1 to to 4, and that most tweets only have one image, we also note that * The minimum favourite count is 0, while the minimum retweet count is 2 * The maximum favourite count is 141298 while the maximum retweet count is 69157

```
In [5]: df.img_num.value_counts()
```

```
Out[5]: 1    1764
        2     195
        3      66
        4      31
        Name: img_num, dtype: int64
```

We can see from the counts that most tweets have one image with only 31 having 4, 66 having 3, and 195 having 2.

```
In [6]: df[['favorite_count', 'retweet_count', 'img_num']].groupby('img_num').sum()
```

```
Out[6]:          favorite_count  retweet_count
         img_num
         1              11721090        3778723
         2               2070526         612254
         3                664378         225923
         4                406600         133908
```

Here we see that tweets with one image have the most likes and favourites, and the likes and favourites reduce as the number of images increases.

```
In [7]: df.shape
```

```
Out[7]: (2056, 22)
```

We observe that there are 2056 different datapoints (tweets), with 22 feature columns for each datapoint (tweet)

```
In [8]: print('-' * 55)
        print(df[['favorite_count', 'retweet_count', 'p1']]\
            .groupby('p1')\
            .sum()\
            .sort_values(by=['favorite_count', 'retweet_count'], ascending = False)\
            .head())
        print('-' * 55)
        print(df.p1.value_counts().head(8))
        print('-' * 55)
```

```
-------------------------------------------------------
                    favorite_count  retweet_count
p1
golden_retriever           1432239         462811
Labrador_retriever          884993         302873
Pembroke                    841341         222232
Chihuahua                   580931         195705
Samoyed                     434980         160085
-------------------------------------------------------
golden_retriever      150
Labrador_retriever     96
Pembroke               88
Chihuahua              81
pug                    57
chow                   44
Samoyed                42
Pomeranian             38
```

```
Name: p1, dtype: int64
--------------------------------------------------------
```

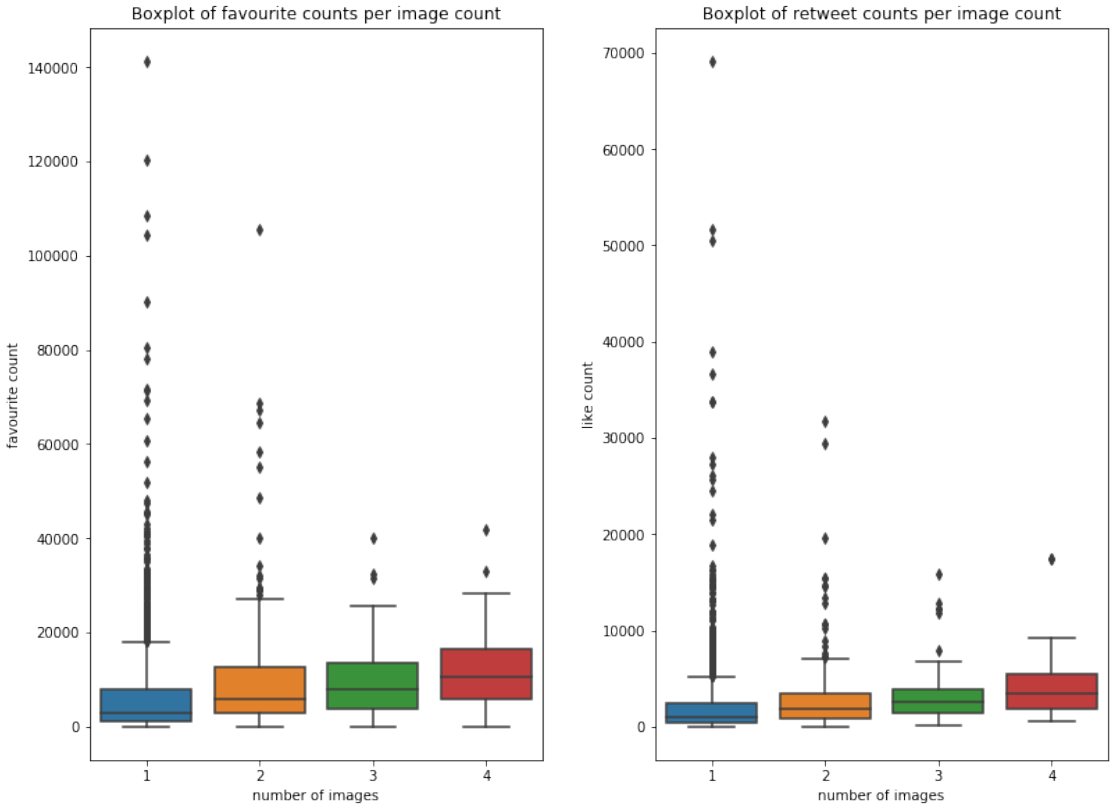
Here we see that **Samoyed** breed dogs are the 4th most liked and retweeted despite being only the seventh most represented by tweet count.

### 1.2.2 Visualization

```
In [9]: import matplotlib.pyplot as plt
        import seaborn as sns

        % matplotlib inline
```

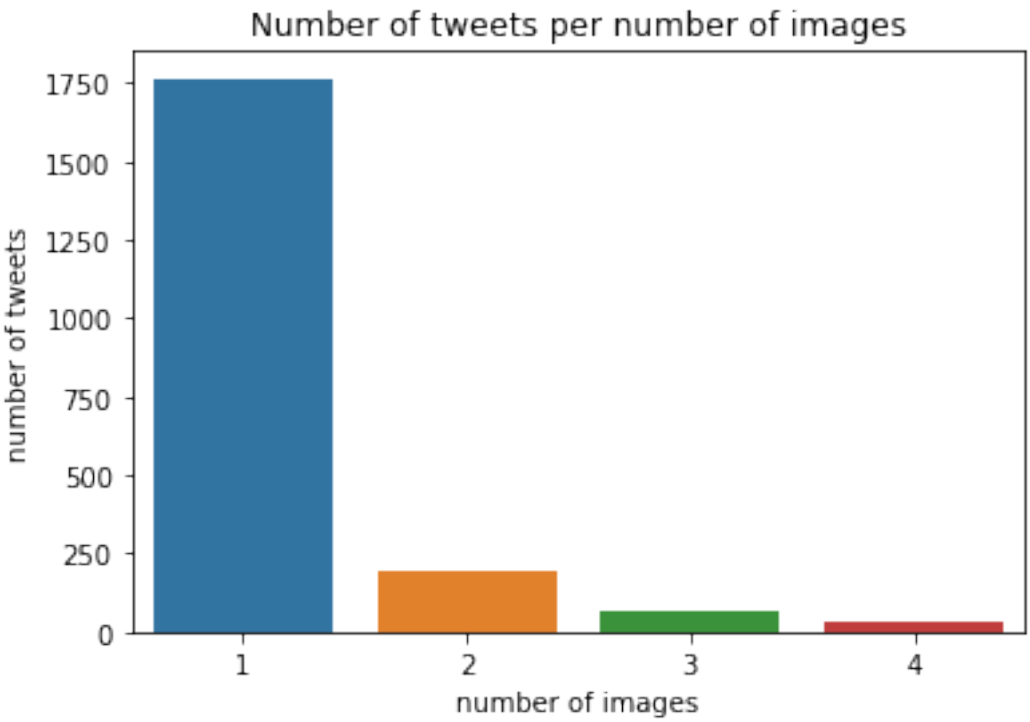
```
In [10]: fig, axs = plt.subplots(ncols=2, figsize=(12, 9))
         fig.tight_layout(pad=5.0)

         plt.title('Boxplot of favourite and retweet counts per image count')


         sns.boxplot(df.img_num, df.favorite_count, ax=axs[0])
         ax=axs[0].set(xlabel='number of images', ylabel='favourite count')
         axs[0].set_title('Boxplot of favourite counts per image count')

         sns.boxplot(df.img_num, df.retweet_count, ax=axs[1])
         ax=axs[1].set(xlabel='number of images', ylabel='like count')
         axs[1].set_title('Boxplot of retweet counts per image count')
         fig.show()
```

```
/opt/conda/lib/python3.6/site-packages/matplotlib/figure.py:418: UserWarning: matplotlib is curr
  "matplotlib is currently using a non-GUI backend, "
```



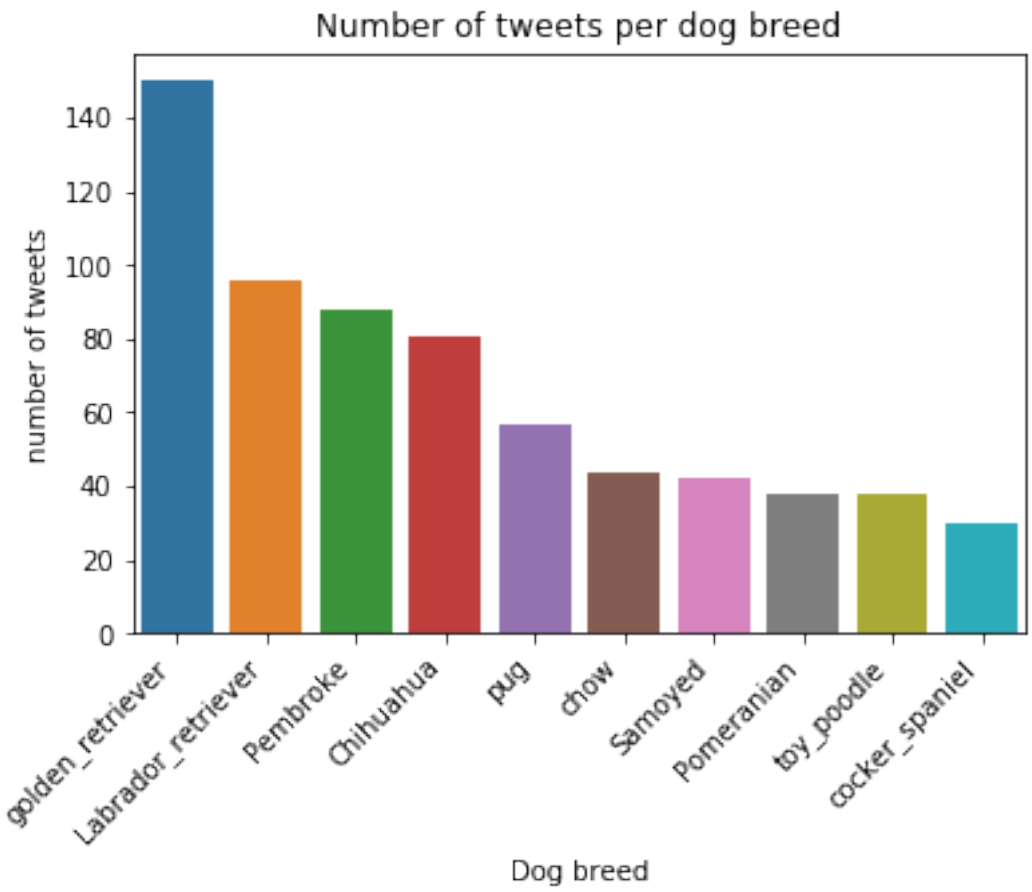

From the box plots above we can see that: * tweets with one image have on average, lower number of likes and retweets close to mean, but also skew more towards high vcalues than all other image counts * tweets with 4 images have the hights number of likes and retweets close to mean, and also the lowest variance in the counts, despite having the lowest overall likes and retweet counts.

```
In [11]: ax = sns.countplot(df.img_num)
         plt.title('Number of tweets per number of images')
         ax.set(xlabel='number of images', ylabel='number of tweets')
         plt.show()
```




Here we see that most tweets have one image, while tweets with 2, 3, and 4 images each have less than 300 tweets

```
In [12]: ax = sns.barplot(x = df.p1.value_counts().index[:10], y=df.p1.value_counts().head(10))
         plt.title('Number of tweets per dog breed')
         ax.set(xlabel='Dog breed', ylabel='number of tweets')

         plt.xticks(rotation=45, ha='right')
         plt.show()
```




This image shows us that according to the neural network AI, golden retrievers are the most represented dog breed, followed by labrador retrievers, then pembrokes, and so on.