

## Wrangle and Analyze Data : WeRateDogs

### Introduction

WeRateDogs is a Twitter account that scores people's pets and includes a funny comments about the dog. The denominator of these scores is generally invariably ten. But what about the denominators? It is almost usually larger than ten. 11/10, 12/10, 13/10, etc.

In this project, we used Tweepy to query Twitter's API for data included in the WeRateDogs Twitter archive. These statistics will include the number of retweets and favorites. We create a Twitter application before executing our API querying code, after which we construct an API object that collects Twitter data. We then write each tweet's JSON data to a tweet json.txt file after querying each tweet ID, with each tweet's JSON data on its own line. We then read this file line by line to generate a pandas DataFrame, then we inspect and clean. Simply put, we will gather, assess, and clean data Twitter Data then act on it through analysis, visualization, and/or modelling. We executed this project with the following steps: Gathering data; Assessing data; Cleaning data; Storing data; Analyzing, and visualizing data and finally Reporting

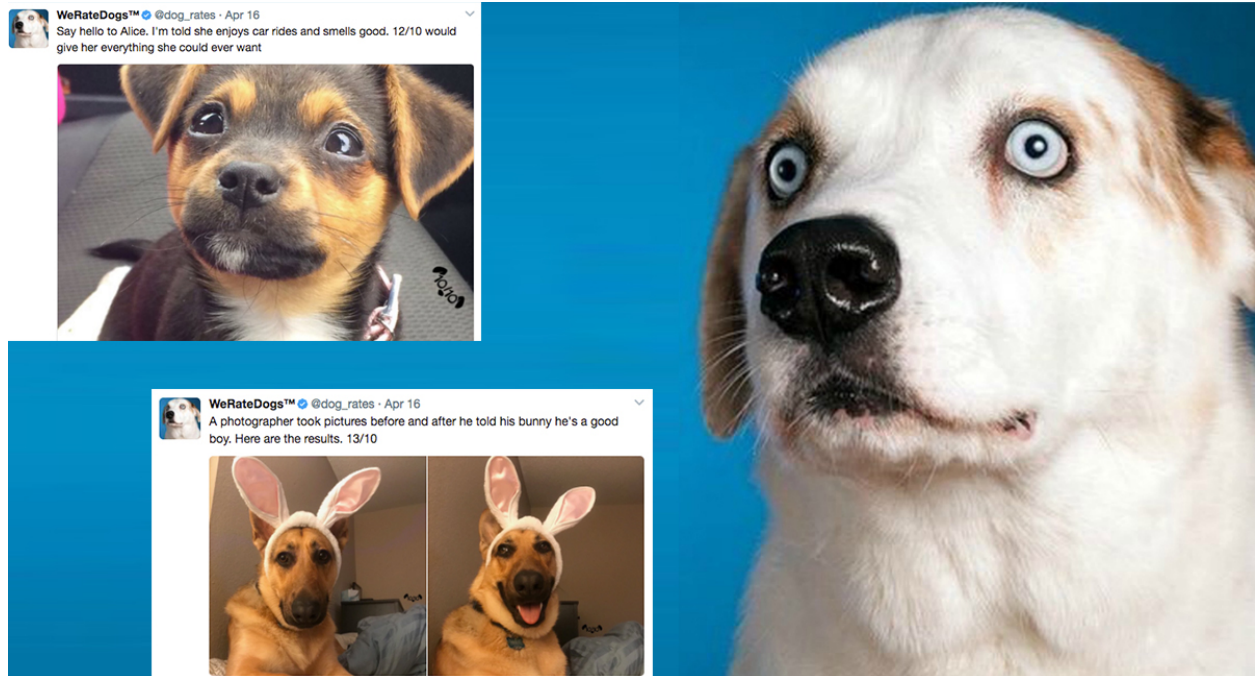
### **Gathering data;**

Data was obtained from three different sources.

- (1) The WeRateDogs Twitter archive
- (2) The tweet image predictions
- (3) Additional data from the Twitter API

#### (1) The WeRateDogs Twitter archive

WeRateDogs downloaded their Twitter archive and emailed it directly to Udacity so that we could use it for this project. This collection contains all 5000+ of their tweets' basic tweet information (tweet ID, timestamp, content, etc.) as of August 1, 2017. This file was manually downloaded as 'twitter archive enhanced.csv'. We read the data and loaded it into a pandas DataFrame.



## (2) The tweet image predictions

According to a neural network, this file (image predictions.tsv) is contained in each tweet. It is hosted on Udacity's servers and may be downloaded programmatically by using the Requests library and this URL

[https://d17h27f6h515a5.cloudfront.net/topher/2017/August/599fd2ad\\_image-predictions/image-predictions.tsv](https://d17h27f6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predictions/image-predictions.tsv)

## (3) Additional data from the Twitter API

We collected the number of retweets and favorites ("likes") for each tweet. Using the tweet IDs from the WeRateDogs Twitter archive, we next use Python's Tweepy package to query the Twitter API for each tweet's JSON data, and we save each tweet's whole set of JSON data in a file named tweet json.txt.

The JSON data for each tweet is provided on its own line. Then, line by line, we read this.txt file into a pandas DataFrame with the tweet ID, retweet count, and favorite count.

## Assessing data

After gathering data we visually and programmatically examine them for quality and tidiness concerns. We discovered and documented certain quality and tidiness concerns in the

### Quality

- Erroneous datatype assigned timestamp
- Missing values (retweeted\_status\_id, retweeted\_status\_user\_id, retweeted\_status\_timestamp, expanded\_urls, in\_reply\_to\_status\_id, in\_reply\_to\_user\_id)
- Incorrect dog names (e.g., not, a, one, the)
- "None" in column names instead of null
- Found 181 retweets. We only want original ratings (no retweets) that have images
- Irrelevant columns to be removed (in\_reply\_to\_status\_id, in\_reply\_to\_user\_id, source, retweeted\_status\_id, retweeted\_status\_user\_id, retweeted\_status\_timestamp, and expanded\_urls.)

### Tidiness

- No separate column for the most accurate dog breed prediction from the three existing (p1, p2, and p3) predictions in the df\_img\_predictions (The tweet image predictions) dataset
- retweet\_count and favorite\_count in the df\_tweet (Additional data from the Twitter API) dataset should be part of the df\_twitter\_archive (The WeRateDogs Twitter archive) dataset.
- id column name not consistent with tweet\_id as found in df\_img\_predictions and df\_twitter\_archive
- Normalization is required to merge the tables to a master table for analysis

