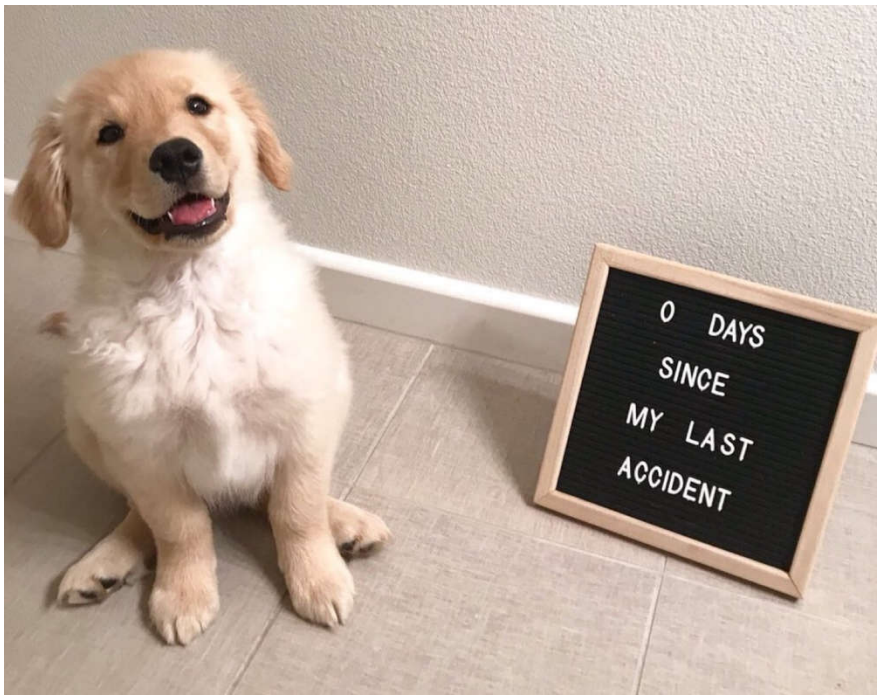
Udacity Data Analysis Second Project

[We Rate Dogs]

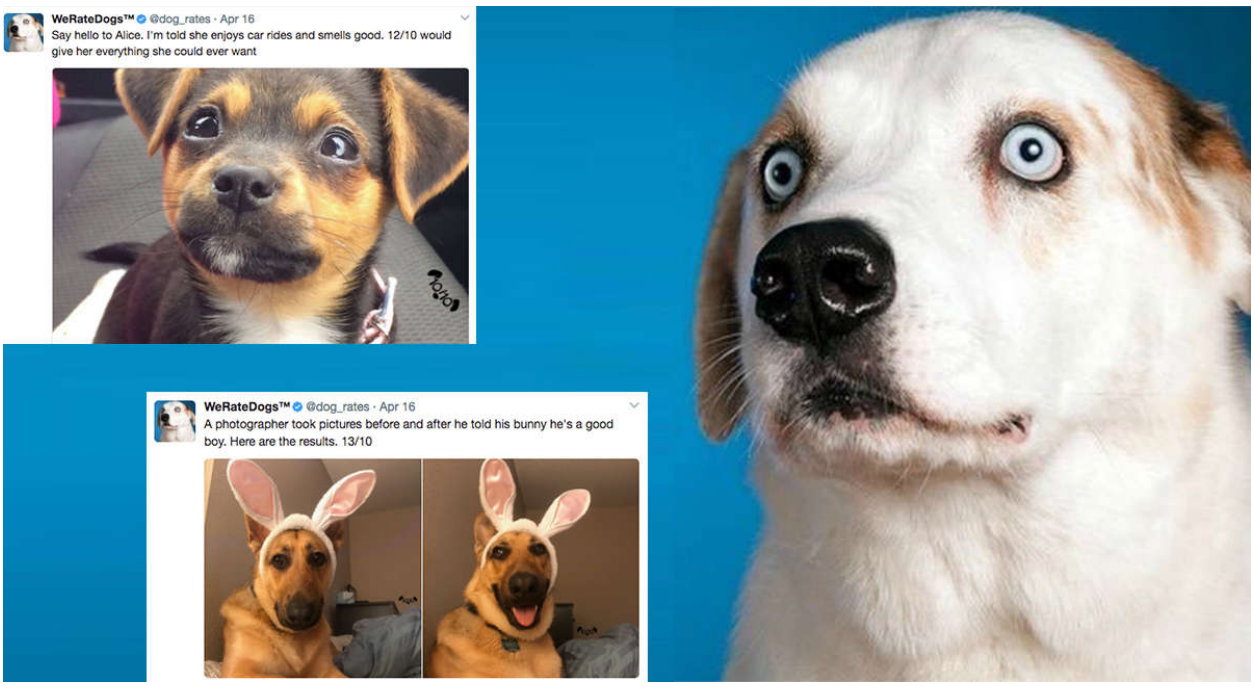
Data Wrangling Project



By: Yousef Ezzeldeen

Introduction

Real-world data rarely comes clean. Using Python and its libraries, I will gather data from a variety of sources and in a variety of formats, assess its quality and tidiness, then clean it. This is called data wrangling. I document my wrangling efforts in a Jupyter Notebook attached in the project folder, plus showcase them through analyses and visualizations using Python (and its libraries) and/or SQL.



The dataset that I will be wrangling (and analyzing and visualizing) is the tweet archive of Twitter user @dog_rates, also known as WeRateDogs. is a Twitter account that rates people's dogs with a humorous comment about the dog. These ratings almost always have a denominator of 10. The numerators, though? Almost always greater than 10. 11/10, 12/10, 13/10, etc. Why? Because "they're good dogs Brent." WeRateDogs has over 4 million followers and has received international media coverage. Let’s take a look.

Project index :

Data Gathering

1- File in hand 'Twitter-archive-enhanced.csv'

The WeRateDogs Twitter archive. I am giving this file to you, so imagine it as a file on hand. Download this file manually by clicking the following link: twitter_archive_enhanced.csv

2- File download programmatically

The tweet image predictions, i.e., what breed of dog (or other object, animal, etc.) is present in each tweet according to a neural network. This file (image_predictions.tsv) is hosted on Udacity's servers and should be downloaded programmatically using the Requests library and the following URL: https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predictions/image-predictions.tsv

3- File from Twitter API

Each tweet's retweet count and favorite ("like") count at minimum, and any additional data found interesting. Using the tweet IDs in the WeRateDogs Twitter archive, we query the Twitter API for each tweet's JSON data using Python's Tweepy library and store each tweet's entire set of JSON data in a file called tweet_json.txt file. Each tweet's JSON data should be written to its own line. Then we read this .txt file line by line into a pandas DataFrame with (at minimum) tweet ID, retweet count, and favorite count.

Data Assessing

After gathering each of the above pieces of data, we need to assess them visually and programmatically for quality and tidiness issues.

1- Quality Issues

Quality issues are concerning:

1- ***Completeness:***
Data is preferred to be complete.

2- ***Validity:***
Data must to be valid.

3- ***accuracy:***
Data must to be accurate.

4- ***Consistency:***
Data must to be constant.

2- Tidiness issues

These issues comes from the concept of Tidy Data which means :

1- ***Each variable forms a column and contains values.***

2- ***Each observation forms a row.***

3- ***Each type of observational unit forms a table.***

Data Storing

After gathering, Assessing and cleaning data, We need to store the data into a .CSV file to make it easy for access. Which you can find in the project folder.

Data Visualization

After that, We now have a clean, tidy and stored data set we can now use our visuals to extract some insights from the data, I have made two types of insights. Let's have a lookon them.

1- Dogs Rating Insights

Some insights about dogs rating based on type and period.

2- Account Insights

Some insights about the account based on Tweet count and total interaction.