Wrangle Report

By Linn Olsson

Introduction

This project uses tweet data from the Twitter account WeRateDogs @dog_rates to put into practice what I have learned about data wrangling in Udacity's Data Analyst Nanodegree.

Project Details

The parts of this tweet data wrangling project are:

- Data wrangling, which consists of:
  - Gathering data
  - Assessing data
  - Cleaning data
- Storing, analyzing, and visualizing the wrangled data
- Reporting:
  - Data wrangling efforts
  - Data analyses and visualizations

Gathering Data

The tool used to conduct this project was Jupyter Notebooks and the data used in this project consists of 3 different parts which where loaded into 3 different data frames in a notebook:

Enhanced Twitter Archive: The CSV file *twitter-archive-enhanced.*csv is a local file that was provided by Udcacity and manually downloaded. This file contains basic tweet data about all tweets by WeRateDogs that includes a rating.

Tweet Image Predictions: The TSV file *image-predictions.tsv* is hosted on Udacity's servers and was downloaded programmatically from a URL provided by Udacity. This file contains image predictions of the images in the tweets provided.

Additional Tweet Data: The text file *tweet_json.txt* contains additional tweet data for the tweets included in the Twitter Archive, which has been obtained by querying the Twitter API.

Assessing Data

After the 3 files had been loaded into separate data frames I assessed these in two ways:

Visually: Visual assessment was made by changing the display settings in the Jupyter Notebook to allow all rows and columns in the data frame to be visible at the same time. I loaded one line of JSON data in its correct structure during the gathering stage to visually assess it.

Programmatically: Programmatic assessment was also conducted on the data to see what kind of data types and number of entries where present and if there was data missing in some columns. I also checked value counts, sums and more.

After assessing the data I made a list of observations by adding things related to *quality* under one header and things relater to *tidiness* under another.

Cleaning Data

When the list of observation had been made it was time to put these into action items and to clean the data according to these. Cleaning the data included only keeping original tweets with a rating and an image of a dog, removing duplicates and merging all tables into one. I changed the data type of multiple columns and removed columns that were empty or unnecessary to keep.

Storing, Analyzing and Visualizing Data

After all the cleaning had been made I stored this data in a CSV file called *twitter_archive_master.csv*. Analyzing and visualizing some of the data was also a part of this project and I decided to look at the engagement rate of each tweet and how it relates to dog breeds and dog stages.

Reporting

Reporting the results from the data wrangling included creating this internal document, named *wrangle_report.pdf* and a report for external communication called *act_report.pdf*.

Reflections

This project was fun and challenging and it feels meaningful because I got to work with real data. Working with Twitter data and their API might be something I will do at work as a data analyst. Getting experience in how to fetch data directly myself is very useful as I have more control over the resources available.

The cleaning part of the data wrangling process took the most time but the most challenging part was to set up the API connection and import the JSON data into the Jupyter Notebook. I had to use a lot of the information from the provided *twitter-api.py* script to get this to work. It also took me a long time to figure out how to parse the JSON data. It was also hard to figure out how to set up the function to fetch the data for the new dog_breed_prediction variables.