Data Wrangling Report

Introduction

Data wrangling is one of the most important processes nowadays. It gives an important information and a better decision to take of a certain problem. We get from the raw data after cleaning, structuring and enriching a useful information in a very fast time. Using wrangling tools allow the analysts to produce much faster data tackle and accurate results, allowing a better decision making.

Wrangling is divided into three steps:

- Gathering Data
- Assessing Data
- Cleaning Data

Gathering Data

Having three different sources of data:

1. The WeRateDogs Twitter archive, includes tweet IDs and other data for the tweets and it was loaded manually.
2. The tweet image predictions, 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, which loaded programmatically.
3. Data from Twitter using Twitter API, using the tweet IDs in the WeRateDogs Twitter archive, 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.

Assessing Data

In this step, we assess the gathered data from the three sources above. Looked for a quality issues and tidiness in the data, where data could be missing, inconsistent, duplicated or incorrect data types. Tidy data could explained by three definition:

1. Each variable forms a column.
2. Each observation forms a row.
3. Each type of observational unit forms a table.

Also, data can be assess with two different ways: visually and programmatically. Some tools that was used in this assessing are .head(), .sample(), .value_counts() and .duplicated(). After doing those assessing, I chose eight quality issues and two tidiness in data.

Cleaning Data

In this step, I cleaned and resolved all the issues that was stated in the previous step. At first, I made a copy of the three datasets and then do the cleaning step on it. This copy is made in case of any error happened to the copy of the dataset, I could get back the copy the do the cleaning without changing in the original data. For every issue, it is solved in a cell with a comment at the top of the cell and then I wrote the code to solve it after that I test what is changed in the output. After finishing this step, I stored the all three-merged DataFrame in one file and saves it.

Conclusion

Good quality and well-organized data powers further analysis, visualization, and modeling. Doing these steps, now the data is cleaned and if we done any analysis, it should give a better accuracy than the original data could give.