Data Wrangling Report

Sean Kan

A lot of effort was put into the wrangling process as we had to work with 3 raw files for this project. It was good practice as this mimics real-world situations where data rarely comes in clean. The data-wrangling process was broken down into three steps: gathering, assessing and cleaning. In the gathering stage, I learned how to download tweets into JSON format via API and convert them to a Panda data frame. The ability to capture all these tweets was an eye-opener as public sentiment can now be tracked with relative ease.

Next, visual and programmatic assessments were used to identify issues that required further clean-up. Out of the 3 files, most attention was devoted to twitter_archive_enhanced.csv (raw tweet data provided by Udacity) as I came across numerous data quality issues such as *tweet duplicates, bad denominators, outliners,* and *entries not associated with a dog breed,* as well as tidiness issues such as *misuse of column headers (variable names instead of values)* and *absence of a normalized rating*.

The clean-up was fairly straightforward as we had already identified our issues in prior. The complete code for this process could be found in wrangle_act.ipynb. However, I have included additional details for two of the more challenging tasks below:

1. Determine the dog breed – We had to assume the algorithm used in prediction_df can accurately classify each dog breed. It would be ideal if we can view the codes which were used to generate the results, but machine learning is beyond the scope of this course. So based on the premise stated earlier, I wrote a for loop that returned predictors with the highest confidence level and omitted entries that were not associated with a dog breed.

2. Create a normalized rating - During the assessment, I noticed there was an inconsistency in the dominators of our ratings. In order to resolve this issue, we had to create a normalized column by dividing the numerators by the dominators. Given the comical nature of this Twitter account, most of the ratings were above 1.0 (above 100%). However, I did notice there were a few outliners that skewed our results and have decided to use the 99^{th} percentile as the cutoff; all tweets with a normalized rating of greater than 1.4 were omitted from our analysis.

After completing the clean-up, I had to merge relevant information from all three data frames by using a left join. As we will be performing linear regression for our analysis, I took an extra step to convert the “nones” to null values as well.

The final output of this wrangling project was saved as twitter_archive_master.csv. Please refer to act_report.pdf for insights produced from this data.