Wrangle Report

Introduction

In this report I will describe the data wrangling process performed in the project named *Data Wrangling Project - Twitter Account WeRateDogs.*
Data wrangling process consists of 3 steps:

- Gathering data
- Assessing data
- Cleaning data

Gathering Data

Gathering data is the first step of data wrangling. For this project the data needed to be gathered was the following:

1. The WeRateDogs Twitter archive stored in a csv file: twitter_archive_enhanced.csv. The WeRateDogs Twitter archive contains basic tweet data for all 5000+ of their tweets, but not everything. One column the archive does contain though: each tweet's text, which I was used to extract rating, dog name, and dog "stage" (i.e. doggo, floofer, pupper, and puppo) to make this Twitter archive "enhanced."
2. The tweet image predictions file, 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 was downloaded programmatically using the Requests library and the following URL: https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predictions/image-predictions.tsv
3. Additional Data via the Twitter API: each tweet's retweet count and favorite ("like") count. Using the tweet IDs in the WeRateDogs Twitter archive, Twitter API was queried for each tweet's JSON data using Python's Tweepy library and stored in a file called tweet_json.txt file.

Gathering data for this project was done in Python Jupyter Notebook using pandas, requests and tweepy libraries. What I found challenging is using tweepy library to query Tweepy API, especially because it’s the first time using such library, it’s the first time accessing Tweeter via secret keys and access. It took a while to download the data and many attempts and many failures, but in the end I managed to extract it successfully.

Assessing Data

Assessing data is the second step in data wrangling. Assessing means inspecting the dataset for two things: data quality issues (i.e. content issues) and lack of tidiness (i.e. structural issues).
In this project I found the following:
I assessed the datasets visually and programmatically and I found the following issues:

Quality:

- twitter data frame
  - Text lines contain links
  - timestamp column is a string
  - Missing dog names (replaced with 'None')
  - Incorrect dog names
  - Rating denominator higher than 10
  - Retweets present in the file: texts start with "RT @"
  - "&amp" characters present in text

- images data frame
  - Some breed names have the first letter lowercase in p1, p2, p3 columns

Tidiness:

- Irrelevant columns in twitter data frame: "in_reply_to_status_id", "in_reply_to_user_id", "source", "retweeted_status_id", "retweeted_status_user_id", "retweeted_status_timestamp"
- Column name timestamp to be renamed in twitter data frame
- Column name "expanded_url" not explicit enough in twitter data frame
- Dog stages split into 4 different columns in twitter dataframe
- "id" column name from json_tweets data frame not aligned with the rest of data frames
- 3 separate data frames

Cleaning Data

Cleaning data is the third step in data wrangling. It is where the quality and tidiness issues that were identified in the assessing data step are fixed.

The cleaning was done programatically. At first I created copies of the 3 files and then followed the process: define, code, test. What I found challenging here is that I didn't know all the needed codes at first, but researching on how to fix similar issues helped me a lot. I can say that I am very proud of the codes that I implemented.