Gathering:

I gathered the data from several sources. First, I download the twitter-archive-enhanced.csv file manually from the Udacity website. That file contained data about tweets from the WeRateDogs twitter archive but it missed a lot of important information. To augment it, I set up a twitter developer account in order to get access to the Twitter API. With the tweet IDs provided in the file, I queried the Twitter API using the Tweepy library to gather more information about the tweets. I also downloaded image_predictions.tsv programmatically from the internet using the Request library. After getting all those files, I was ready to go to the next step.

Assessing:

Before proceeding to any analysis, I had to assess the data to look for inconsistencies and errors that could give misleading conclusions later on. Also, getting a feel for the dataset helped me identify the most suitable structure to ease my analysis further.

The data was pretty messy. It contained several quality issues:

- I discovered that some tweets had more that one ratio in their text. I used regular expression to extract a list of ratios for each tweet. I noticed that some ratios did not represent dog ratings but because the original manipulator of the data used probably the 'extract' function instead of 'findall', only the first ratio in the text was extracted while the real rating appeared further in the text.
- Some tweets were replies to other tweets or retweets. Those were not interesting for the analysis.
- The dogs were rated out of 10 but many denominators were different from 10. Most of those tweets were not dog ratings. It was also the case with the ratings that had a numerator below 5 or superior to 20.
- Some rating numerators were decimals but the regex that had been used to extract them did not take into account the decimal points. For example: 13.5/10 was stored as 5/10.
- After joining the breed predictions with the tweets, I noticed that the breed recognizer marked some pictures as not dogs.
- The timestamp column was represented as a string.

For the tidiness issues:

- The doggo, puppo, pupper and floofer columns represented one variable which is the dog stage.
- The breed predictions, retweet count, dog stage and favorite count were spread across several tables.

Cleaning:

In this phase, I solved many of the issues that I had discovered in the assessing phase but I also discovered new ones.

- I filled the missing values for the doggo, puppo, pupper and floofer table using regular expression to catch the occurrences that had not been extracted in the original file.
- For the tweets that contained more than one ratio, I inspected the list of ratios of each tweet and selected the most plausible one.
- I deleted the tweets that were replies or retweets.
- I deleted the tweets that had a denominator different from 10, a numerator below 5 or superior to 20.
- For the decimal numerators that were incorrectly stored, I used a regex that took into account the decimal point. I also converted the column's data type to float.
- I converted the timestamp's column data type to datetime.
- I used the pandas melt function to transform the doggo, puppo, pupper and floofer columns into one dog stage column.
- I joined retweet count, favorite count, breed prediction and dog stage in a single table.

After all those operations, I was finally ready to analyze my data.