Gathering the required data involved collecting three different data sets from three different sources. The twitter archive data was provided and just had to be read in as a csv. The image prediction data was downloaded using requests and read in as a tab-separated csv since it was a tsv file. The favorite and retweet counts data had to be obtained from the twitter API. I read in the json data from the twitter API line by line to extract the favorite and retweet counts, as well as the corresponding tweet id, which would be needed to match this information to the other information for that tweet.

Now that the data was gathered, it had to be assessed for quality and tidiness issues so that it would be in a usable state for analysis. I started with a visual assessment of all the columns to get a sense of the data. I noticed a few things right away, such as a timestamp column that I would need to check if it was a datetime type programmatically. I also noticed many null values in the reply and retweet columns and realized this meant the tweets were original tweets and not replies or retweets. Another thing I noticed was there were 4 columns for the dog stage, a clear tidiness issue since this is one piece of information that should be sorted in 1 dog stage column with those 4 columns as the values for the new column.

With the visual assessment done, I programmatically assessed the data, starting with the info function to check data types and null values. The timestamp was a string and would need to be changed to datetime. It was now clear that 181 tweets were retweets and 78 were replies, which would need to be removed since they are not wanted. Also, in the image predictions some predictions were not dog breeds and would need to be removed for the purpose of analysis. There were also tweets that contained dog stages but were not marked as such in the dog stage columns.

With the issues identified, I began cleaning the data. Several issues were easy to solve, such as converting type to datetime and dropping unwanted columns to simplify the data sets. Before dropping the retweet and reply columns, I queried only the rows with null values in those columns and saved the dataframe to remove the retweets and replies. Then, the columns were dropped. Since the archive and counts datasets both contained tweet information, they needed to be merged into the same table. This was done with an inner merge on tweet id. A similar merge was done with the top predicted breed and the archive data since the information is very similar to the dog stage information and was the only column needed for the analysis. To handle the dog stage issue, I created a function to reevaluate the tweet text for the dog stage terms since some were not correct in the provided data. This function was applied to the archive dataframe to create the new dog stage column. The original 4 separate columns were dropped.