Gather

First, I gathered data from three sources: the WeRateDogs Twitter Archive CSV, the Twitter API, and the image prediction neural network hosted by Udacity. I used Pandas read_csv() to create a dataframe (df) from the WeRateDogs Twitter Archive. From the Twitter API, I collected each tweet's JSON and saved them as lines in a .txt file, then created a dataframe (tweet_selected_attr) with tweet id, favorites, retweets, and timestamps from the .txt file. I then used requests to get the image predictions tsv file from Udacity's servers and used read_csv() to create a third dataframe (image_prediction).

Assess

For each dataframe, I first completed visual assessment by looking at the dataframes, then programmatic assessment using particular methods.

For the WeRateDogs Twitter Archive (df), my visual assessment showed that there are Reply and Retweet rows in the dataframe. This project is focused on original tweets, so I determined the Reply and Retweet entries needed to be removed to make sure they do not skew the analysis. I also observed that the word "None" was written in the columns "doggo," "floofer," "pupper," and "puppo" - which meant there were many rows for which a dog stage was not assigned, but programmatic assessment would miss this because the entries have "None" typed in (as opposed to a blank entry). Furthermore, there were 2 tidiness issues visible: dog_stage should be in one column (as opposed to separate columns for "doggo," "floofer," "pupper," and "puppo") and numerator_rating and denominator_rating should be in one "score" column. Programmatic assessment also revealed long urls as source entries (found using value_counts()) and some rating_numerator outliers (found using .describe().

For the Tweet Image Predictions (image_prediction), my visual assessment showed some ambiguous column headers ('img_num,' 'p1', 'p1_conf', 'p1_dog', 'p2', 'p2_conf', 'p2_dog', 'p3', 'p3_conf', 'p3_dog'), and some entries that were predicted to NOT be dogs. Programmatic assessment using info() showed that there were not the same number of rows in the "image_prediction" dataframe as "df," and the predicted breeds were object datatypes rather than category datatypes.

For the dataframe tweet_selected_attr, the visual assessment revealed that the dataframe had rows for each tweet, and from a tidiness perspective, it would make the most sense to combine it with the other dataframes to make one combined dataframe where tweets are the unit of analysis.

Clean

I first addressed missing data by removing retweet and reply rows using isnull() and drop(). Then I combined the WeRateDogs Twitter Archive dataframe (df) with the Image Prediction dataframe (image_prediction) using Pandas merge() to produce one combined dataframe that only included rows with image predictions.

I then addressed tidiness issues, starting with changing the “None” entries for   “doggo,” “floofer,” “pupper,” and “puppo” to blanks using replace(), creating a new dog_stage column combining entries from those columns and replacing blanks with NaN using Numpy’s NaN method, and dropping the old separated columns. Afterwards, I removed rows with outlier numerator_rating entries using loc[] and drop(), and combined numerator_rating and denominator_rating into a new column, dropping the separate numerator and denominator columns. Finally, I added the retweet and favorite data for each tweet to the dataframe using Merge(), producing “dataframe_combined_2” - a dataframe with the required data about each tweet from all 3 sources.

Then I addressed quality issues, including dropping rows where the most likely image prediction was not a dog using drop(), making the most likely prediction a category datatype using astype(), and making the tweet source a category and easier to read using replace() and astype().

Save

Finally, I saved the combined dataframe as a csv using to_csv().