wrangle_report

January 13, 2023

0.1 Reporting: wragle_report

I started my data wrangling process by importing all necessary packages to perform the wrangling and analysis of our data. I then read in the twitter archive data from from csv using the pandas read_csv function, and then I downloaded the tsv file for our image predictions data from the url programmatically and read the data into a dataframe. When trying to to setup a twitter account to utilize tweepy I ran into authentication issues and was unable to continue with a Twitter account so I had to follow the directions for access the Twitter data without actually creating a Twitter account. I think read the tweet_json.txt data into a dataframe. With our data now in panda dataframes I began the assessing data process. I assessed the data by displaying each dataframe and performing a visual inspection of the data and also using pandas functions like describe and info to see if there was issues that needed to be cleaned up. I created clean copies of the dataframes to start using for my cleaned data. From my inspection of the data I found that the columns doggo, floofer, pupper, and puppo had "None" instead of null values and these columns should also be combined into one column that is used to describe the dog category. I changed the "Nones" to null and then combined the columns into a dog_category column. I noticed that there were many retweets in the the twitter_archives dataframe that needed to be droped so I identified those using the retweet id column and droped the rows with that data. The source column was messy as it was in html so I converted the html into a more friendly format for analysis by replacing the html with what source was used. The datatype for the timestamp column was a string instead of a datetime so I used the to datetime function to swap it over to the correct datatype. tweet_id in the dataframes were also showing as an integer instead of a string which would be the proper datatype for an identifier so I changed the datatype for those columns to a string. Many of the denominators where not 10 and having the numerator and denominator seperate would make it more challenging to perform analysis on the data. To solve this I created a new column "dog_rating" and normalized the values by dividing the numerators by the denominators so they are all in the same base. I then droped the numerator and denominator columns as they would now be redundant. The dog types in p1,p2, and p3 were not all uniform as some were capitlized and some were not. I decided to capitilize all of the dog breeds which will allow us to perform analysis on the breeds without them being seen as different values. Once all of this was done I determined that in order for the data to be tidy it should all be in the same dataframe so I merged all three to be combined into one master dataframe. Once that was over I saved my master dataframe programatically.