Wrangle Report

I am presented with a problem to wrangle and analyse the data obtained through Twitter api and Udacity, but dirty and messy data comes with a lot of headaches. It is the job of a data analyst to clean the data and prepare it for further analysis.
The three datasets (i.e twitter_dogs, Twitter_df and image_df) have some quality and tidiness issues i had to deal with . I realised during my data wrangling process on all three datasets that there were more than 15 quality and 3 tidiness issues I had to deal with, but before I continued with my analysis I made copies of all 3 datasets.

Having noticed that the dog_stage wasn’t extracted properly from the text and they were represented as columns, I then created a function to extract the dog_stage name from the text whether pupper ,doggo, floofer and puppo are in capital letters or small letters , or ‘s ‘ was added with the dog_stage or any other stylish word was added (like PUPPERGEDON, puppers) , they were all extracted and categorised under one column called ‘dog_stage’. Source column looking more like a html tag than an observation, I decided to strip the source column of the ‘a-tag’ and the ‘href’ tag so as to give way for a pure text.

After taking a careful look at the twitter_dogs dataframe, I found several inconsistencies in the rating_numerator and denominator, as some ratings do not correspond with the ratings in the text (e.g 'After so many requests, this is Bretagne. She was the last surviving 9/11 search dog, and our second ever 14/10.’ The ratings_numerator and denominator were represented as 9 and 11 respectively instead of 14 and 10). I also found a rating of 240/170 which is not consistent with the 10 values given to the denominator. Therefore I replaced the ratings that do not correspond with the ratings in the text manually, and changed all the rating_numerator values that are not in the range of 10+ to a range of 10+ with functions, and also changed all the rating_denominator to 10 using a lambda function.

With names like ‘a’, ‘an’ and ‘this’ present in the name column, i decided to take a deep look at the text and its corresponding name values, then i observed , that names like ‘a’ , ‘an’ and ‘this’ in the name column were as a result of a slightly faulty programmatic extraction. Therefore , I replaced these values with None in the name column . I then converted datetime related columns from an object type to datetime type. I dropped duplicate records present in the Twitter_df table and capitalised the values of p1, p2 and p3 present in the image_df table.

Finally I merged all three data frames together , then dropped the p2, p2_conf, p2_dog , p3, p3_conf and p3_dog as they were not as accurate as p1, p1_conf and p1_dog in terms of prediction. I then saved the master_df to a twitter-archive-master.csv before any further analysis