Wrangle Report

1. Gather

-The data had to be gathered in 3 different ways. One file was directly given, the other had to be programmatically downloaded from a web page and the other had to be created by querying Twitter's API.

-Creating the first 2 DataFrames was relatively simple, but it was important to remember that the image_predictions file had a '.tsv' extension. This meant that the 'sep' argument in the .read_csv method had to be explicitly given.

-Querying twitter's API to get data on the tweets was more complicated however. My initial idea was to create a dictionary object with the keys being the columns of the future DataFrame. This dictionary was then filled by querying the Twitter API. However, this became problematic once I realised I had to run this loop every time I opened the notebook which took about 30min. To solve this problem, I decided to create a JSON file and save it so I could ignore the loop once the data had been gathered.

2. Assess/Clean

-In this stage, the data had to be assessed and then organised to make it easier to understand.

-I first did a visual assessment and a programmatic assessment. The main quality and tidiness issue for each table were then noted in a markdown cell below.

-The main issues with the data lied in the twitter_archive table. Certain issues were complicated resolve, like the fact that certain names were invalid. There are several reasons for this: no name was given to the pet, or the pet wasn't a dog. Since the dataset is large, I chose to adopt a simplistic approach and drop these rows.

-Some of the other key cleaning steps were: creating a master DataFrame which compiled the 3, melting columns for the dog stage, dropping rows relating to retweets/tweets without images.

-Another complicated cleaning task was to find and remove other rows in which the pets weren't dogs. I decided to do some string parsing and use common key words like 'we only post dogs' to search for such tweets and drop the rows.