Data Wrangling Report

Introduction

The purpose of this project is to gather data from a variety of sources and in a variety of formats, assess its quality and tidiness, then clean it.

The dataset that I wrangled (and analyzing and visualizing) is the tweet archive of Twitter user @dog_rates, also known as WeRateDogs.

Data Gathering

twitter_archive: The WeRateDogs Twitter archive, provided by the Udacity Course as a download

image_predictions: The tweet image predictions, hosted Udacity's servers and downloaded programmatically.

tweet_data: JSON document downloaded using the Twitter API (Tweepy library).

Data Assessing

After gathering each of the above pieces of data, I assessed them visually and programmatically for quality and tidiness issues. I then cleaned each of the issues I documented while assessing.

Quality Issues

Validity

1 - There are entries with odd dog names (a, an, the, just, one, very, quite, not, actually, mad, space, infuriating, all, officially, 0, old, life, unacceptable, my, incredibly, by, his, such), remove them (df_1) **Made a list of the odd and used an 'if' within a 'for' loop to replace them**

2 - tweet_id are int64, convert to str (df_1, df_2, df_3) **Used 'astype'**

3 - timestamp is an object, convert to timestamp (df_1) **Used 'to_datetime'**

4 - retweeted_status_timestamp is an object, convert to timestamp (df_1) **Used 'to_datetime'**

5 - in_reply_to_status_id is float64, convert to str (df_1) **Used 'astype'**

6 - in_reply_to_user_id is float, convert to str (df_1) **Used 'astype'**

7 - retweeted_status_id is float, convert to str (df_1) **Used 'astype'**

8 - retweeted_status_user_id is float, convert to str (df_1) **Used ‘astype’**

Consistency

9 - id needs to be changed to tweet_id (df_3) **Used ‘rename’**

10 - There are inconsistent numerators and denominators, some numerators are below 10 and some denominators are not equal to 10 (df_1) **Left as is**

11 - Change the source contents for better readability; change the long url links into words (df_1).

- **used a for loop to replace all the 'None'**
- **used cat**
- **dropped the old columns**
- **used np.nan**

Tidiness Issues

1 - There are 3 separate tables (let's merge them) **Used ‘merge’ inner join on tweet_id**

2 - dog stages are in multiple columns (we can combine them into one column) **Used ‘replace’**

Store

I then stored the clean DataFrame(s) in a CSV file with the main one named twitter_archive_master.csv.