DATA WRANGLING REPORT: 'WERATEDOGS' TWITTER DATA PROJECT

BY DANIEL OGIKU

A. INTRODUCTION

Data wrangling is a vital skill that every data analyst should be familiar with since so much of the world's data is not clean and messy.

Project Details

My tasks in this project involved.

1. Data wrangling, which consists of:

● Gathering data

● Assessing data

● Cleaning data

2. Storing, analyzing, and visualizing our wrangled data

3. Reporting on:

a) Data wrangling techniques and (b) data analysis and visualization.

The above analysis was carried on data related to WeRateDogs' twitter account.

WeRateDogs is a Twitter page that regularly shares pictures of dogs along with a catchy description and often a rating out of 10 for the dog in the picture, sometimes exceeds 10. Created in November 2015, it became popular so fast and at this moment has more than 8 million followers. In this analysis, there was an exploration for changes in the tweets' favorites, retweets, and ratings over time.

Tools, libraries and programming language used in this project:

- Pandas library
- Numpy library

- Python
- Requests library
- Tweepy
- Json library
- Matplotlib library
- Twitter's API
- Jupyter Notebook

B. GATHERING DATA FOR THIS PROJECT

Data was gathered from 3 different sources:

1. WeRateDogs Twitter archive given by Udacity in csv format:

The WeRateDogs Twitter archive was provided for this analysis. I downloaded this file manually by clicking the link with this url text provided in the workbook: twitter_archive_enhanced.csv.

Read the data stored in the 'twitter-archive-enhanced.csv' file using panda's method 'read_csv' file. I stored it in a DataFrame 'twitter_archive'. The data had so many issues that needed to be cleaned and resolved for the purpose of this project.

2. Image prediction file downloaded programmatically using Requests library and the URL provided by Udacity in tsv format:

This file (image_predictions.tsv) is hosted on Udacity's servers and was downloaded programmatically as instructed using the Requests library and the following URL:
https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predictions/imagepredictions.tsv

3. Data retrieved by querying Twitter's APIs and using Tweepy library.

My application for a Twitter developer account was rejected, to save time on my project I used the shortcut data provided from my Udacity dashboard and stored each tweet's entire set of JSON data in a file called tweet_json.txt file and later stored this data in a dataframe name 'dfjson'

C. ASSESSING DATA FOR THIS PROJECT

After gathering the data and storing them in DataFrames, the next step was assessing the data for quality and tidiness issues. Data were assessed both programmatically and visually.

These were the following Quality and Tidiness Issues I found and addressed for this project.

Quality Issue

- Incorrect datatype

img_num Column should be string not Integer datatype

Change tweet_id from an integer to a string

Source Column should be in Category datatype

Timestamp: this column should be date-time format instead of string

- Name Column contain some invalid names like a, an .....these are articles, adjectives not real dog names
- Remove columns with too many missing values, to make the final dataset more neat and tidy.
- Delete retweets status - not necessary for this analysis
- Check and drop duplicate values to make the final dataset more neat and tidy
- p1, p2, p3: dog breed names are not all in lowercase
- Source column is not readable
- The numerator and denominator columns have invalid values

Tidiness Issue

- Move doggo, floofer, pupper and puppo columns into one column 'dog_type'.
- Drop doggo, floofer, pupper, puppo columns
- Merge the dataframe twitter_archive, dataframe image_predictions, and tweet_json dataframes cleaning Data for this Project

D. CLEANING DATA FOR THIS PROJECT

The (define, code, and test) steps were used in the cleaning process.

First, copies of the DataFrames were created before cleaning. After making copies of the dataframes, I commenced the cleaning process on each quality and tidiness issues listed above.

Codes used in the cleaning process included:

Copy(), Concat(), drop(), groupby(), count(), astype(), replace(), duplicated(), columns.

E. STORING DATA FOR THIS PROJECT

At this stage I had a cleaned and structured data

I stored the Cleaned data into a CSV File :
dfnew.to_csv('twitter_archive_master.csv', encoding='utf-8')

F. VISUALIZATION, INSIGHTS AND OBSERVATIONS

Visualizations and insights are provided in ‘act_report.pdf

G. CONCLUSION

This werateDogs-twitter data wrangling project was a good test of what I learnt in the 4th course of this Udacity program.

From finding data problems, and cleaning those hard, time-consuming data quality problems. In the end, I was able to build a well-structured wrangling processing notebook with details in every part, detailing the steps of how I gathered, assessed, and cleaned the data with every problem and its solving process highlighted.

Visualizations and insights are provided in ‘act_report.pdf