Wrangle Act

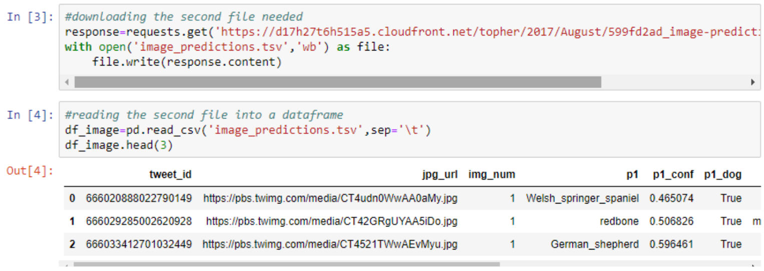
This project set out to extract data from the Weratedogs twitter database, clean the data and derive useful insights from the data. In the project, three sets of data were used for the analysis, two out of the three were provided by the Udacity instructors, while I had to obtain the third from the twitter database using their API.

Obtaining the first data:

The first data, labelled twitter_enhanced_data was pretty straightforward to obtain as it was provided directly. All I did was download the file and read it into a pandas dataframe.

Obtaining the second data:

For the second data, I was provided the url, and I had to make use of the request library in python. After obtaining the file which was stored temporarily in my system memory, I wrote it into a file, after which I read it into a pandas dataframe. The syntax and the codes for this are represented in a snapshot below:



```
In [3]: #downloading the second file needed
        response=requests.get('https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predict:
        with open('image_predictions.tsv','wb') as file:
            file.write(response.content)

In [4]: #reading the second file into a dataframe
        df_image=pd.read_csv('image_predictions.tsv',sep='\t')
        df_image.head(3)
```

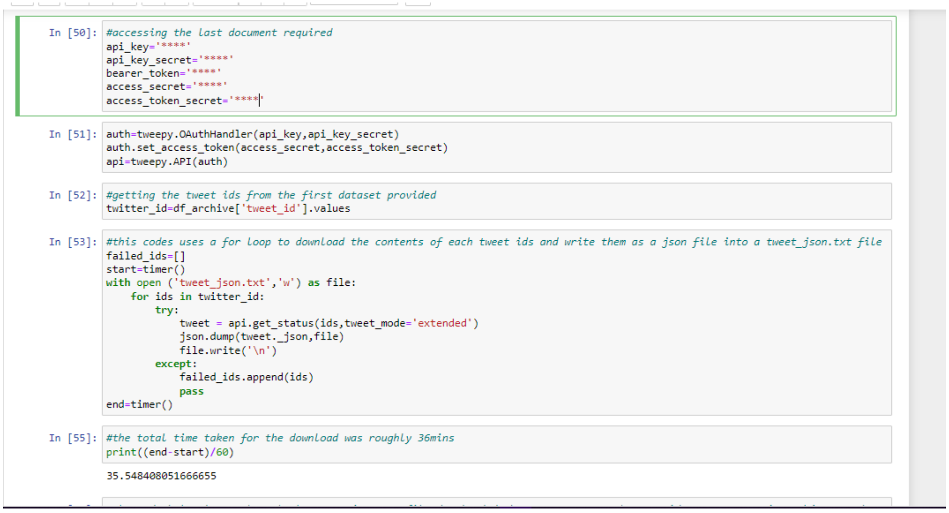
Out[4]:

	tweet_id	jpg_url	img_num	p1	p1_conf	p1_dog	
0	666020888022790149	https://pbs.twimg.com/media/CT4udn0WwAA0aMy.jpg	1	Welsh_springer_spaniel	0.465074	True	
1	666029285002620928	https://pbs.twimg.com/media/CT42GRgUYAA5iDo.jpg	1	redbone	0.506826	True	m
2	666033412701032449	https://pbs.twimg.com/media/CT4521TWwAEvMyu.jpg	1	German_shepherd	0.596461	True	

Obtaining the third data:

For the third dataset, I used the API for twitter to obtain the data. In order to get access to the API, I applied for a twitter developer account, after which I was granted access in about two weeks.

I pip installed tweepy (a library for querying twitter API). The syntax for the codes I used for this are displayed below.



```
In [50]: #accessing the last document required
         api_key='****'
         api_key_secret='****'
         bearer_token='****'
         access_secret='****'
         access_token_secret='****'

In [51]: auth=tweepy.OAuthHandler(api_key,api_key_secret)
         auth.set_access_token(access_secret,access_token_secret)
         api=tweepy.API(auth)

In [52]: #getting the tweet ids from the first dataset provided
         twitter_id=df_archive['tweet_id'].values

In [53]: #this codes uses a for loop to download the contents of each tweet ids and write them as a json file into a tweet_json.txt file
         failed_ids=[]
         start=timer()
         with open ('tweet_json.txt','w') as file:
             for ids in twitter_id:
                 try:
                     tweet = api.get_status(ids,tweet_mode='extended')
                     json.dump(tweet._json,file)
                     file.write('\n')
                 except:
                     failed_ids.append(ids)
                     pass
         end=timer()

In [55]: #the total time taken for the download was roughly 36mins
         print((end-start)/60)

         35.548408051666655
```

The codes took roughly 35mins to download the contents and write to a twitter_json.txt file.

Creating a Dataframe from twitter_json.txt file:

After writing the contents of each tweet_id to a txt file, I converted the file to a pandas dataframe. I made use of the json library method (json.dumps) in order to convert the json.txt file to a json object. I did this so that I could access the contents I needed in the form of a python dictionary.

The additional data I gathered from the json file include:

- 'tweet_id':[],
- 'date':[],
- 'retweet_count':[],
- 'favorite_count':[],
- 'full_text':[],
- 'followers_count':[]

Data Cleaning

After obtaining the data, I performed some data wrangling processes. 11 Quality issues and 2 Tidiness issues were identified and fixed using the **Define, Code and Test** format.

Notably, for the tweet_ids that had erroneous numerator and denominator ratings, I went to the werate_dogs twitter database to obtain the correct ratings for those posts. I did this so I could have a dataset that was near perfect for the exploratory data analysis.

After cleaning the datasets, I set to explore the data to generate insights about the data.