Data Intake Report

Name: File ingestion and schema validation
Report date: 12.4.2021
Internship Batch:LISP01
Version:1.0
Data intake by:Ines Perko
Data intake reviewer:<intern who reviewed the report>
Data storage location: https://github.com/inesp93/File-ingestion-and-schema-validation

Tabular data details:

Total number of observations	84897528
Total number of files	3
Total number of features	37
Base format of the file	.csv
Size of the data	8,4 GB

Proposed Approach:

The initial dataset is https://www.kaggle.com/mkechinov/ecommerce-behavior-data-from-multi-category-store?select=2019-Oct.csv, where the file 2019-Oct.csv has 5.28 GB. First I tried to read the .csv file with pandas pd.read_csv( ). Even though it was 5GB, it was successful and it was done in 2minutes.

```
In [2]: import pandas as pd #normal pandas

In [5]: %%time
        pd.read_csv('Desktop/week 6/2019-October/2019-Oct.csv')
        Wall time: 2min
```

I wanted to try different methods of file reading with Modin, Ray and Dask, since it speeds up pandas workflows. Specifically, the Modin library has benefits such as the ability to scale up pandas workflows with one line of code. Therefore, I used the following code to install Modin with all (i.e., Modin dependencies and Ray to run on Ray and Modin dependencies and Dask to run on Dask) https://modin.readthedocs.io/en/latest/installation.html. Then I wanted to use read_csv( ) and measure time to compare with other ways but I got an error "MemoryError: Unable to allocate 512. KiB for an array with shape (65536,) and data type int64".

```
!pip install modin[all]
```

```
import modin.pandas as pd
```

```
import ray
ray.init()
```

```
%%time
pd.read_csv('Desktop/week 6/2019-October/2019-Oct.csv')
```

Kernel Restarting

The kernel appears to have died. It will restart automatically.

I really wanted to try Modin, so I decided to try with another, smaller dataset so I took https://www.kaggle.com/hhs/health-insurance-marketplace?select=Rate.csv where Rate.csv has 1.83 GB. Again, the kernel died. Since whenever I try to install any additional library in pandas, or the environment at Anaconda I have an issue with either memory or Python version, or missing packages. I believe that the real problem is in an operating system (I have Windows instead of Linux). So I decided to move on with the assignment and upload the first file.

The next problem occurred in writing the yaml file. I was getting an "ERROR:root:while scanning a simple key in "file.yaml", line 12, column 5 could not find expected ':' in "file.yaml", line 13, column 5. I was not sure what was the problem, so I decided to reduce the size of my dataset to track the progress easier. The new dataset is called New and it has 1,30 GB. It has columns: 'product_id', 'price', 'brand', 'user_id'. I replaced the Nan values in the column brand with the string 'unknown'. I modified the yaml file with columns: {'product_id', 'price', 'brand', 'user_id'}.

```
In [33]: config_data

Out[33]: {'file_type': 'csv',
          'dataset_name': 'newdata',
          'file_name': 'New',
          'table_name': 'edsurv',
          'inbound_delimiter': ',',
          'outbound_delimiter': '|',
          'skip_leading_rows': 1,
          'columns': {'product_id': None,
           'price': None,
           'brand': None,
           'user_id': None}}
```

```
In [34]: #read the file using config file
         file_type = config_data['file_type']
         source_file = "Desktop/week 6/" + config_data['file_name'] + f'.{file_type}'
         #print("",source_file)
         df = pd.read_csv(source_file,config_data['inbound_delimiter'])
         df.head()
```

Out[34]:

	product_id	price	brand	user_id
0	44600062	35.79	shiseido	541312140
1	3900821	33.20	aqua	554748717
2	17200506	543.10	unknown	519107250
3	1307067	251.74	lenovo	550050854
4	1004237	1081.98	apple	535871217

I checked if the header of the file is validated.

```
In [35]: #validate the header of the file
         util.col_header_val(df,config_data)

         column name and column length validation passed

Out[35]: 1
```

```
In [36]: print("columns of files are:" ,df.columns)
         print("columns of YAML are:" ,config_data['columns'])

         columns of files are: Index(['product_id', 'price', 'brand', 'user_id'], dtype='object')
         columns of YAML are: {'product_id': None, 'price': None, 'brand': None, 'user_id': None}
```

Furthermore, I continued with the code, but I am still not finished with the inspection.

```
In [20]: if util.col_header_val(df,config_data)==0:
             print("validation failed")
             # write code to reject the file
         else:
             print("col validation passed")
             count_row = df.shape[0]  # Gives number of rows
             print("total number of rows", count_row)
             count_col = df.shape[1] # Gives number of columns
             print("total number of col", count_col)
             #source_file
             # write the code to perform further action
             # in the pipleine

         column name and column length validation passed
         col validation passed
         total number of rows 42448764
         total number of col 4
```