Data Intake Report

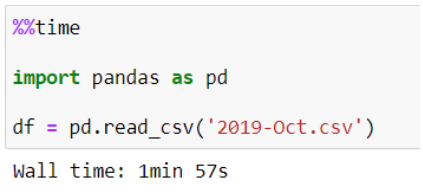
Name: File ingestion and schema validation
Report date: 19.04.2021
Internship Batch:LISP01
Version:3.0
Data intake by:Ines Perko
Data intake reviewer:<intern who reviewed the report>
Data storage location: https://github.com/inesp93/File-ingestion-and-schema-validation

Tabular data details:

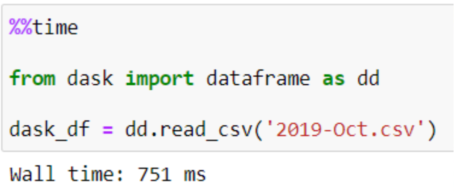
Total number of observations	42448764
Total number of files	1
Total number of features	9
Base format of the file	.csv
Size of the data	5,27 GB

After problems with Modin installation and memory capacity in Version 1.0 and 2.0, I finally overcame those problems and moved forward with the task.

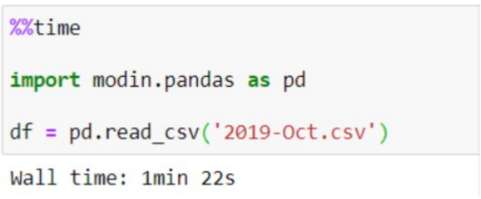
Proposed Approach:
The initial dataset is https://www.kaggle.com/mkechinov/ecommerce-behavior-data-from-multicategory- store?select=2019-Oct.csv, where the file 2019-Oct.csv has 5.28 GB. First I tried to read the .csv file with pandas pd.read_csv( ). Even though it was 5GB, it was successful and it was done in 1 min 57 sec. I tried to read it with Dask and it was done in 751 milliseconds. After importing Ray with the code (import ray ray.init()), I got results after 1min 22 sec.



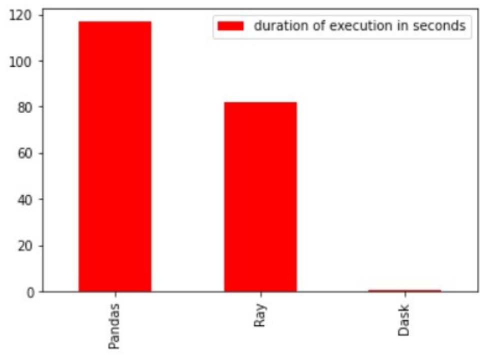
```
%%time

import pandas as pd

df = pd.read_csv('2019-Oct.csv')
Wall time: 1min 57s
```



```
%%time

from dask import dataframe as dd

dask_df = dd.read_csv('2019-Oct.csv')
Wall time: 751 ms
```

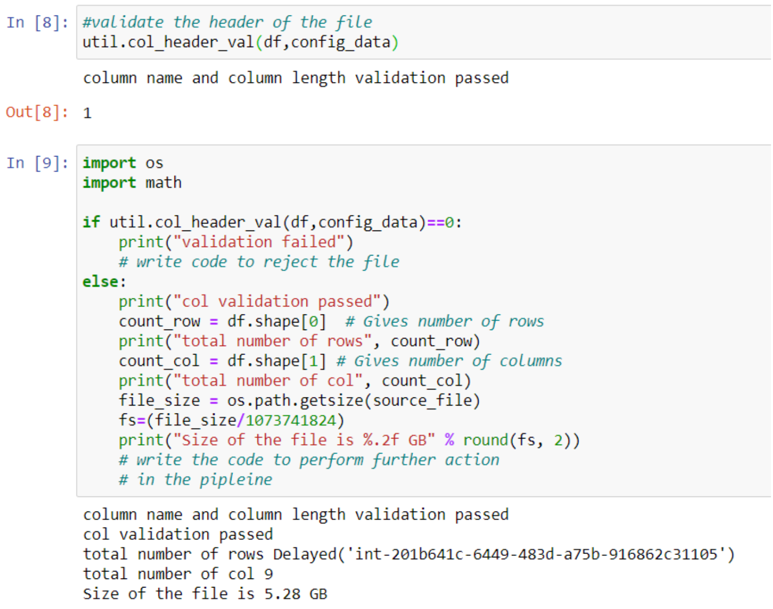


```
%%time

import modin.pandas as pd

df = pd.read_csv('2019-Oct.csv')
Wall time: 1min 22s
```



One can see a graphical comparison of the execution times in the left graph.

After successfully applying the code, the following results are:

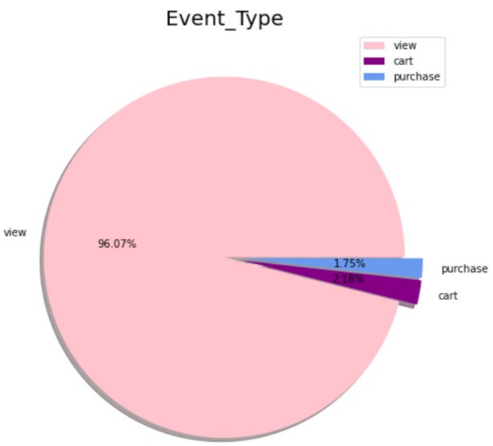


```
In [8]: #validate the header of the file
        util.col_header_val(df,config_data)

        column name and column length validation passed

Out[8]: 1
```

```
In [9]: import os
        import math

        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)
            file_size = os.path.getsize(source_file)
            fs=(file_size/1073741824)
            print("Size of the file is %.2f GB" % round(fs, 2))
            # write the code to perform further action
            # in the pipleine

        column name and column length validation passed
        col validation passed
        total number of rows Delayed('int-201b641c-6449-483d-a75b-916862c31105')
        total number of col 9
        Size of the file is 5.28 GB
```

I wanted to perform a further investigation with the dataset but I cannot use normal pandas and dask functions like “groupby(), count(), nunique()”. All the resources about dask said that it should work, but it doesn’t work. I cannot correct mistakes anymore so I decided to read the dataset with normal pandas way and inspect it that way.

The dataset consists of the columns: event_time, event_type, product_id, category_id, category_code, brand, price, user_id, user_session.

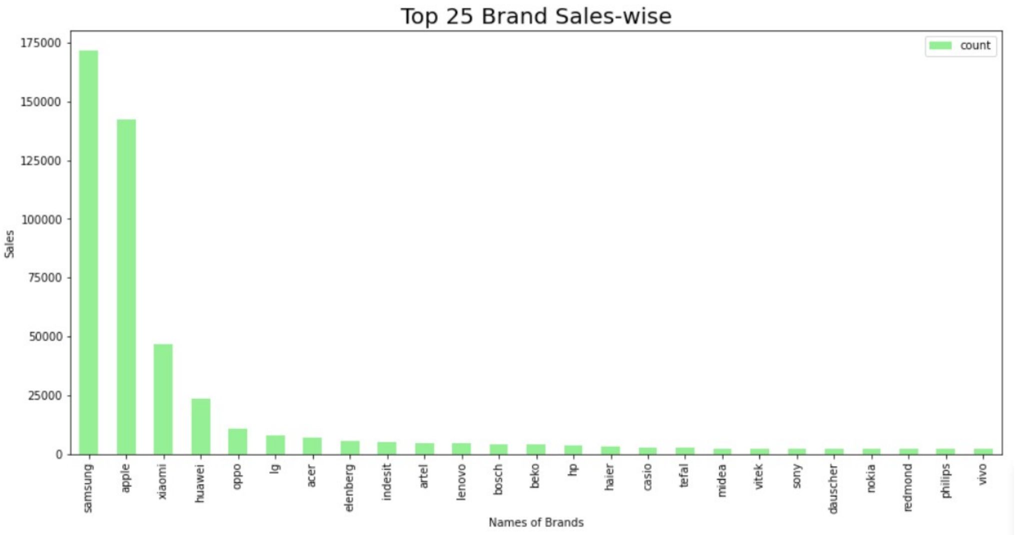


The column ‘event_type’ has 4 different inputs: view, cart, remove_from_cart, and purchase. The following pie chart shows the percentage of events view, cart and purchase. In numbers, that would be:

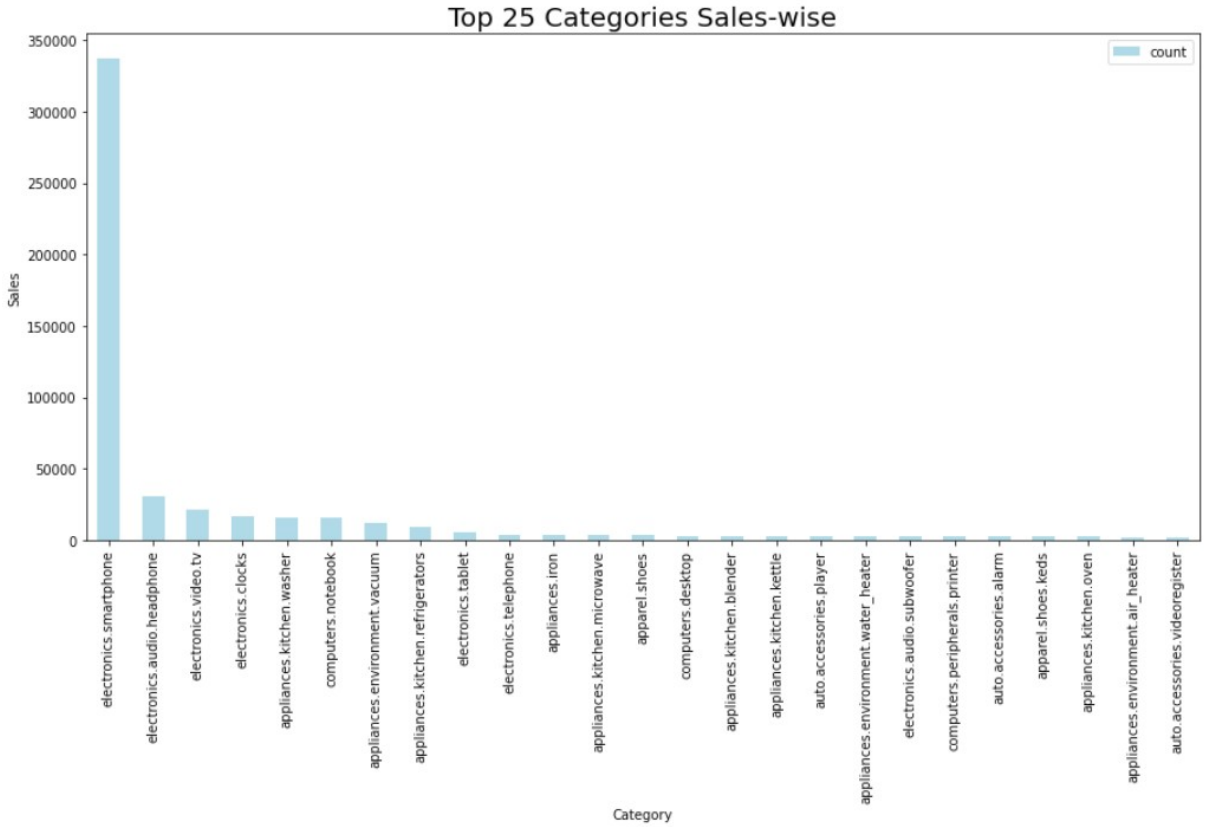
view	40779399
cart	926516
purchase	742849

The number of users based on the unique value of user_id is 3022290.

Furthermore, the following graph shows the top 25 brands based on the number of sales.

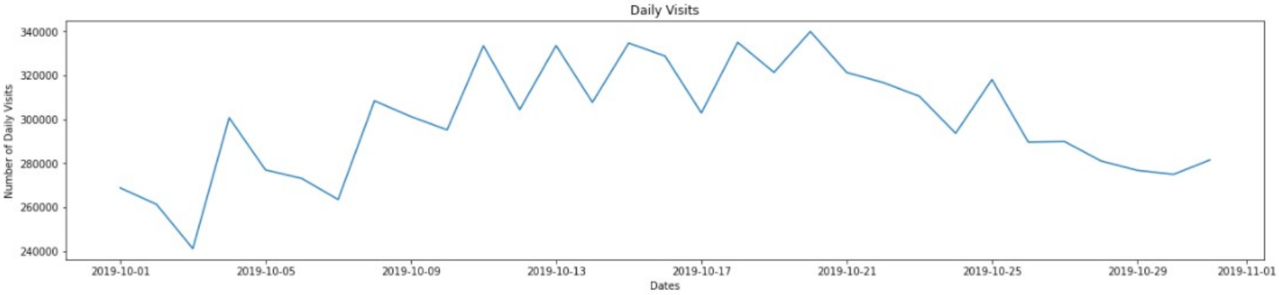


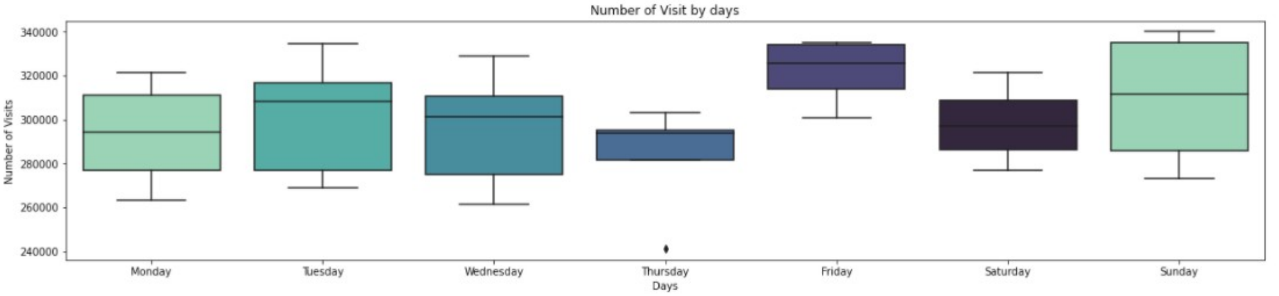
One can see that the most popular brand is Samsung, with 171706 sold products. The following graph shows the top 25 categories based on the number of sales. One can see that electronics, in particular, smartphones, are the most sold (337575 products).



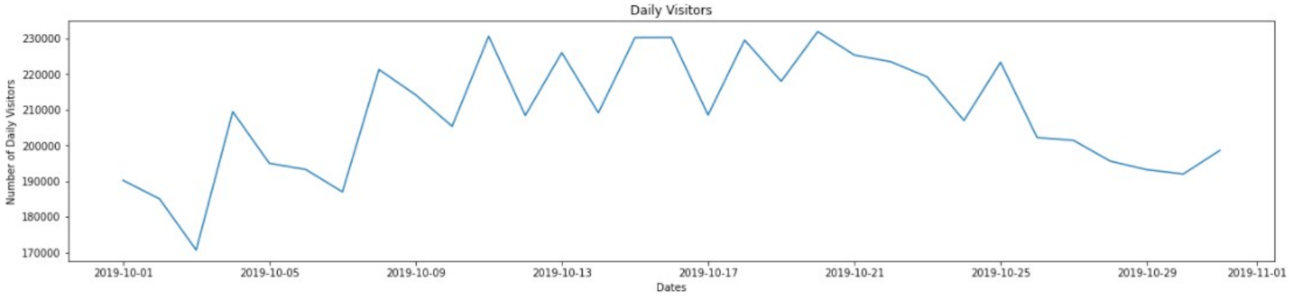
Also, one can see that number of daily visits increases in the middle of the month and then decreases towards the end of the month. The peak is 339943 daily visits and minimum is 241086 daily visits. Sunday seems to be the best day for shopping 339943 visits in total (during 4

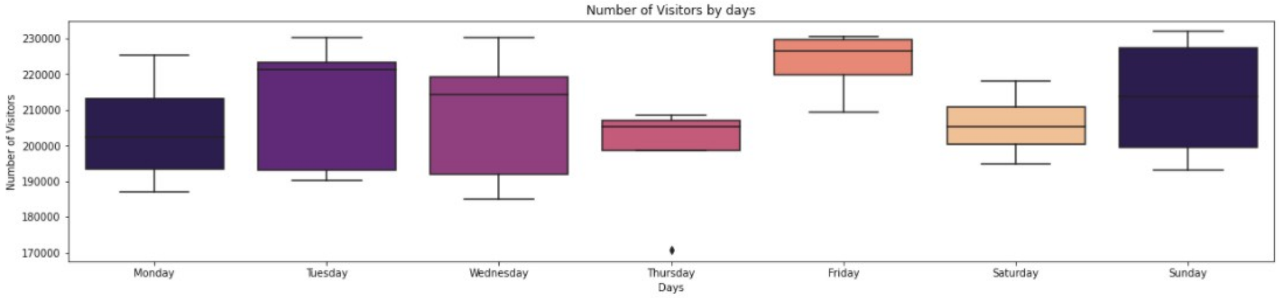
Sundays in October 2019), while Thursday is the worst days for shopping with 241086 visits even though there were 5 Thursdays in October 2019.



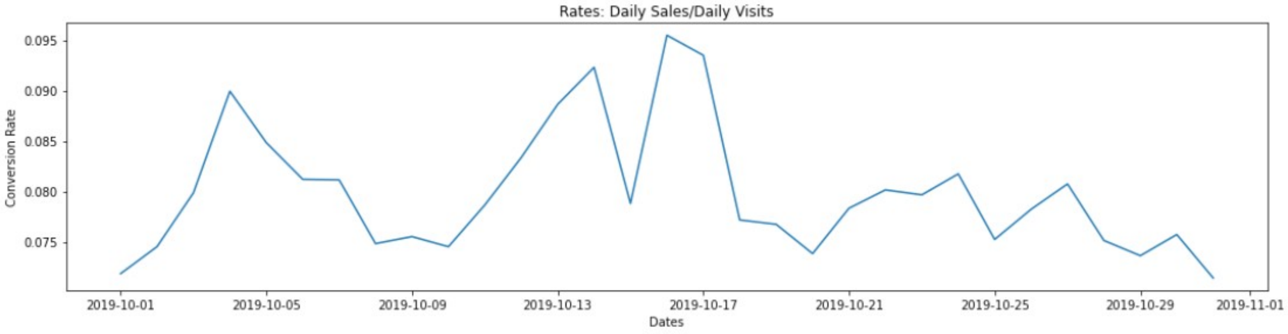


Naturally, the number of daily visitors increases during the mentioned time, so the peak is on Sundays with 231849 visitors in total and a minimum on Thursday with 170668 visitors in total.





On Wednesdays, the conversion rate is the highest, i.e., 9.54%. The users shopped the most.


Rates: Daily Sales/Daily Visits
Conversion Rate
Dates