Project Report (Data Sc. Tools - 2)

Customer Behavioral Analysis & Product Recommendation System (E-commerce)

-- Submitted: By Anshul Dabas

1. Dataset Motivation:

I wanted to work with an e-commerce customer dataset to analyze customer's behavior in terms of platform activities and build a product recommendation system with ground as customer behavioral analysis. I required user activities, user details and product details data as a potential dataset. I managed to finalize a multi-category retail store(ecommerce) dataset from the following resource link – https://www.kaggle.com/mkechinov/ecommerce-behavior-data-from-multi-category-store?select=2019-Oct.csv

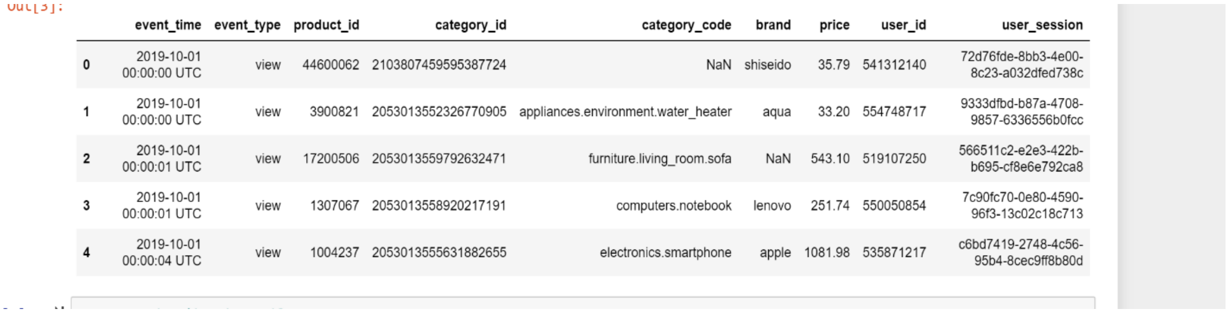
This dataset consisted of one-month (October-2019) details for users, products, and user-activities. It consisted of almost all the required features to analyze user-platform relationship and dive deep into customer behavior to build product recommendations further.

Metadata:

User Data – user_id, user_session

Product Data – product_id, category_id, category_code, brand, price

User Activity Data – event_time, event_type (event – if a customer only viewed product/ added product to the cart/ purchased the product)



Out[3]:

	event_time	event_type	product_id	category_id	category_code	brand	price	user_id	user_session
0	2019-10-01 00:00:00 UTC	view	44600062	2103807459595387724	NaN	shiseido	35.79	541312140	72d76fde-8bb3-4e00-8c23-a032dfed738c
1	2019-10-01 00:00:00 UTC	view	3900821	2053013552326770905	appliances.environment.water_heater	aqua	33.20	554748717	9333dfbd-b87a-4708-9857-6336556b0fcc
2	2019-10-01 00:00:01 UTC	view	17200506	2053013559792632471	furniture.living_room.sofa	NaN	543.10	519107250	566511c2-e2e3-422b-b695-cf8e6e792ca8
3	2019-10-01 00:00:01 UTC	view	1307067	2053013558920217191	computers.notebook	lenovo	251.74	550050854	7c90fc70-0e80-4590-96f3-13c02c18c713
4	2019-10-01 00:00:04 UTC	view	1004237	2053013555631882655	electronics.smartphone	apple	1081.98	535871217	c6bd7419-2748-4c56-95b4-8cec9ff8b80d

2. Actual task definition:

Analyze customers' behavior on a multi-category retail e-commerce platform with respect to products' involved activities and use it to further to build a product recommendation system by using required data modelling techniques.

It will help the e-commerce business to increase their sales and user-platform engagement activities by targeting potential products to required users. Also, overall, it can help a business to target its user as per their needs and preferences to maximize profit via increasing sales of recommended products.

Input - all the dataset features including user, user-activity, and product details

Output - groups of similar users and recommended products modelling techniques in them

3. Literature Review:

There has been a great work done in building recommendation systems and studying customer behavior by using various models and techniques. Most of the systems were either based on ratings provided by the customers or product descriptions and reviews. There are various kinds of product recommendation systems – content-based, collaboration-based, hybrid (includes both previous ones) etc.

I have built my product recommendations for clustered(similar) users by using associative rules and correlated products via different data modelling techniques. Also, I have incorporated product recommendations as per the user journey and behavior on the platform. It will help the firm to work on product recommendations as per required by a specific user group. This will further enhance the diversity in product recommendations on the platform by keeping in mind the required target customers.

4. Workflow:

a) Data Collection:

I read the dataset csv file into a DASK data frame. My dataset consisted of ~42 million records, due to which pandas failed to read the data. So, I had to use DASK data frames to process the data, they work via distributed and parallel processing of data to resolve scalability issues in python. Although, later I have used pandas data frames as well with required data (cleaned and less records as per case requirements) for data analysis and data modelling.

b) Data Exploration:

In this section, I wanted to explore the data distribution, potential data cleaning requirements and outliers in the data.

Missing Values: In the dataset, only 3 features had missing values - category_code, brand and user_session.

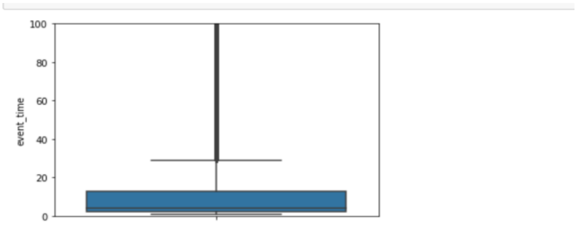
First, for category_code, I checked if I could extract missing values using their category ids. However, I found that all the values were null for a category id if it had a single null category_code value. So, to impute category_code with required category names was not possible by using category ids. So, I decided to impute them rather than dropping missing category_code rows to save from huge data loss (13515609 records). I tried imputing them with 'No category_code', however DASK data frame was unstable with the impute function (sometime bug and sometime run fine.). I decided to break my data into groups as per the distribution and apply required impute method in each group later. Same would be followed for other data cleaning methods as well. All the data cleaning methods were applied groupwise due to the data scalability issue.

Secondly, for brand, all the missing values were imputed as 'No brand' group wise as the reason stated above. I did not drop them as well due the same reason of huge data loss as there were around 6117080 rows with missing brand values.

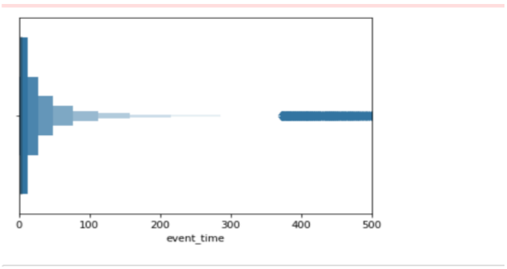
Third, for user_session, I ended up dropping the 2 records with missing user_session values, later in group-wise analysis. This is because there was no concern of data loss due to dropping 2 records.

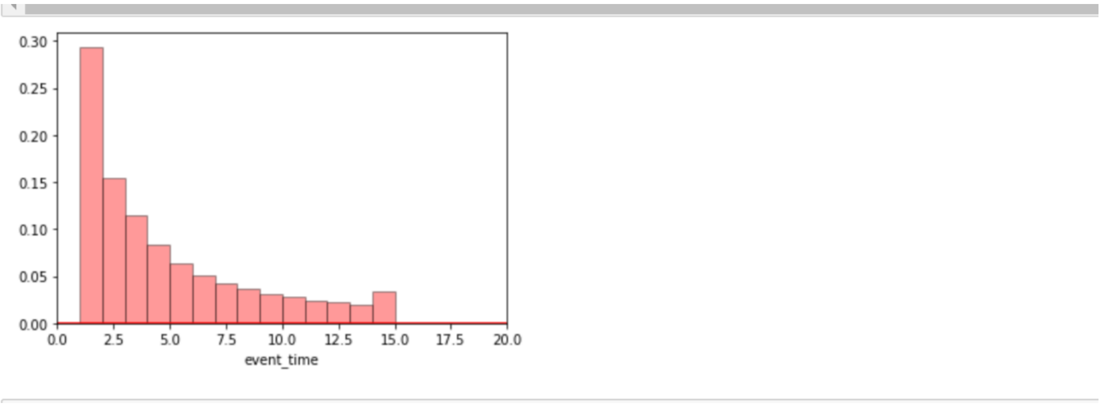
<u>Datatype Conversion</u>: Only event_time column had the wrong datatype. Initially, I converted it to datetime object. But this column was not required in my analysis or modelling, so I ended up dropping this column.

<u>Explore Data Distribution and Potential Outliers:</u> I began with the primary most important trait of a user required to study user-platform journey data by a business – user visits/events (visit/event can be considered any record/row associated to a user irrespective of visit/event type(view/add_cart/purchase)). To understand the distribution of the customers, I grouped the data using user_id and analyzed their visit counts to search the diversity among them according to the data. Following plots were used to analyze and explore the customers' data distribution.



Exponential distribution of number of visits of the users



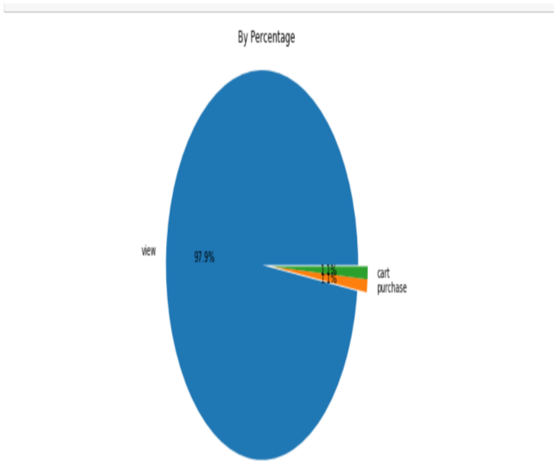


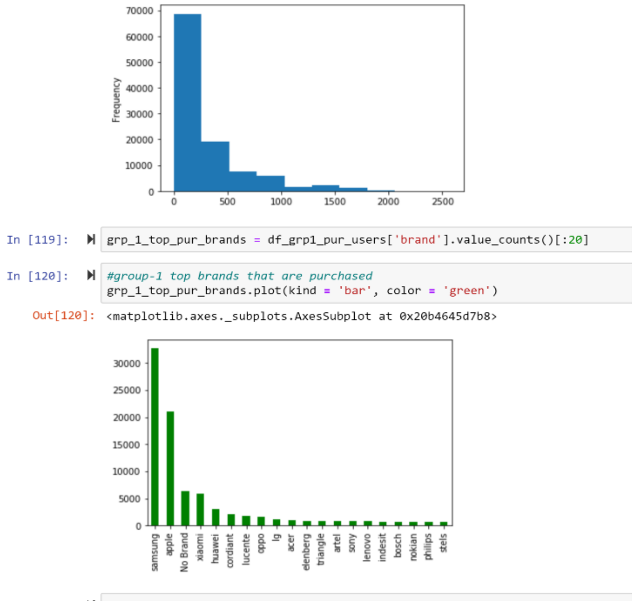
All the above plots helped me to understand that the data was distributed exponentially i.e. customers with high variance(diversity). So, it is required to get rid of outliers and group data accordingly to proceed with analysis group-wise. Almost 25% of the data was outlier data, I analyzed it separately as all customer groups should be studied separately in a business real-world scenario to understand them. However, I did not use the outlier data in modelling section. I used it only for analysis purpose.

Finally, after studying the data distribution, I divided my dataset into 3 groups. All the groups were processed via data cleaning techniques as explained above. Also, a group-wise analysis was implemented to understand customer behavior separately.

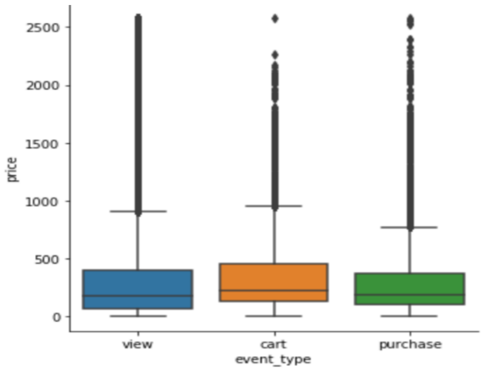
c) Group-wise Data Cleaning & Analysis:

<u>Group-1:</u> Data was sliced from the DASK data frame for all the users with number of visits or events to be <= 4. Further, data cleaning was implements for this group as explained in previous section. Also, I discovered a data discrepancy in the dataset that can be handled as a future scope of this project. The discrepancy was that the event- cart should be greater or equal to counts of purchase events as per the user journey is concerned to buy a product on the platform. Following is some of the required plots with interpretations to understand the user activities and behavior.





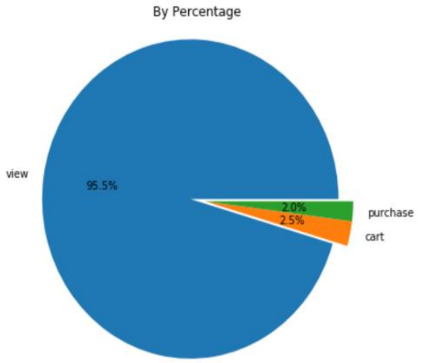
```
In [119]:  grp_1_top_pur_brands = df_grp1_pur_users['brand'].value_counts()[:20]

In [120]:  #group-1 top brands that are purchased
           grp_1_top_pur_brands.plot(kind = 'bar', color = 'green')

Out[120]:  <matplotlib.axes._subplots.AxesSubplot at 0x20b4645d7b8>
```

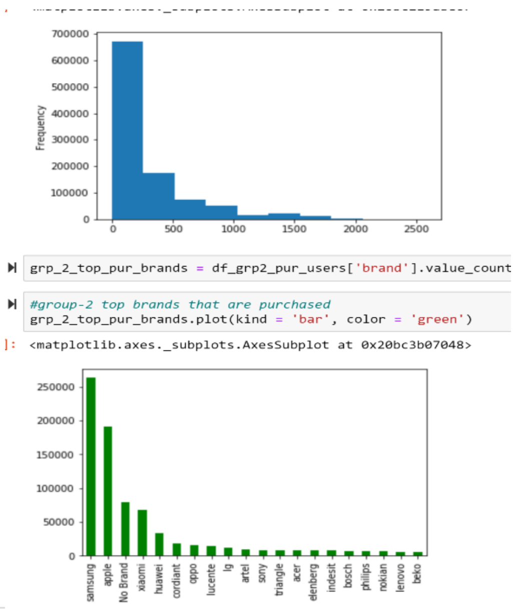


Inferences - As expected most of the users just viewed the products. They can be recommended the products having maximum ratings. Also, product/user similarity(correlations) matrix as build in data modelling section later, can be used to recommend products based on similar products viewed.

There are many outliers as per price range is concerned, with 100-500 dollars range for all 3 event types (view, cart, purchase) i.e., to be in the same price range. That makes sense, purchased products will be in view and cart category as well. Also, views and purchase for a product are positively correlated mostly (assumption as per real-life scenarios). Tops purchased products are from smartphone or electronics brands/categories.

Group-2: Similar data cleaning operations and plots were implemented as group-1 for this group separately. It consisted of users with number of visits between 5 and 15(included). Following is some of the required plots with interpretations to understand the user activities and behavior.

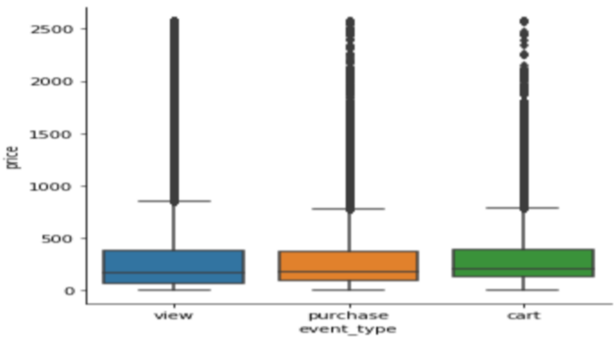




```
grp_2_top_pur_brands = df_grp2_pur_users['brand'].value_count
```

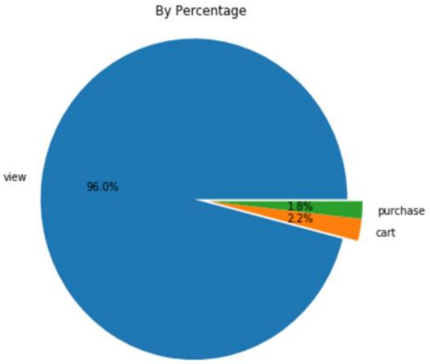
```
#group-2 top brands that are purchased
grp_2_top_pur_brands.plot(kind = 'bar', color = 'green')
```

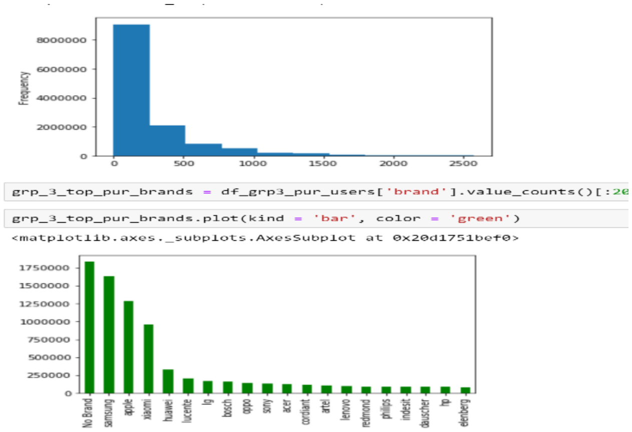
```
]: <matplotlib.axes._subplots.AxesSubplot at 0x20bc3b07048>
```



Inferences- Unexpectedly, this group have the similar trends for all features as group-1**.**

Group-3(Outliers group): Similarly, all the data cleaning operations and plots were implemented as group-1 for this group separately. It is the outlier users' group. It consisted of users with number of visits more than 15. Following is some of the required plots with interpretations to understand the user activities and behavior.

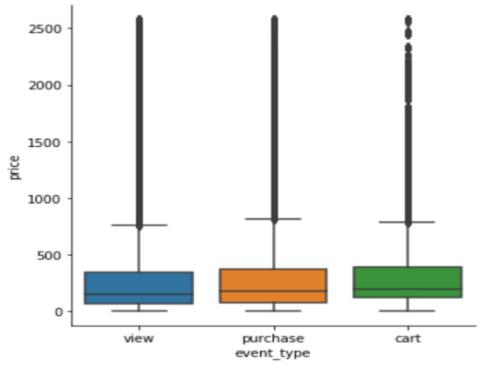




```
grp_3_top_pur_brands = df_grp3_pur_users['brand'].value_counts()[:20
```

```
grp_3_top_pur_brands.plot(kind = 'bar', color = 'green')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x20d1751bef0>
```



Inferences- Surprisingly, outlier group had the same trend as well as the other previous groups.

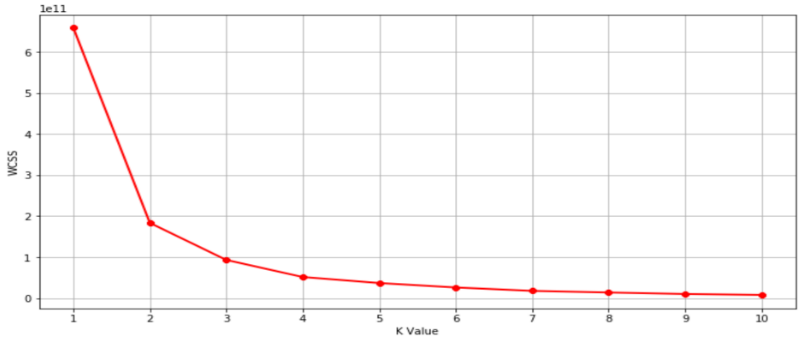
Overall group analysis Interpretation- All the three groups divided based on their visits/events on the platform had shown similar trends. This implies that number of visits failed to provide a significant criterion to group similar users in same groups but groups being different from each other. So, let's use clustering technique to model our users into different groups with similar users by using other features to further recommend required products.

d) Data Modelling:

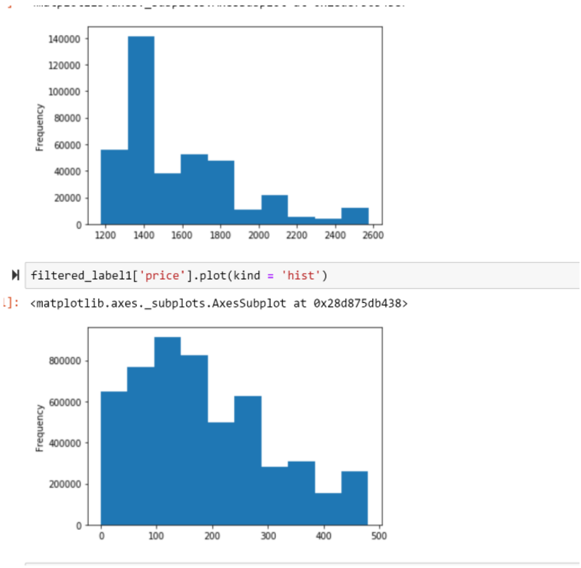
For this section, I not just applied required data cleaning steps but also removed the outlier data. In data analysis section, I could not find any significant difference in users between different groups based on their visits. So, I decided to apply k-means clustering on the clean data to find similar user groups based on their event_type, ratings, and price. I also wanted to use category_code for clustering. However, due to memory issues, I could not implement it. It can be added as a future scope to cluster users based on other required or domain knowledge features like category_code.

Also, I added a new column as ratings by using purchase history of users. Finally, before applying k-means, I checked the data distribution again. As a result, very few outliers and manageable variance in the data was reflected.

I used elbow method to determine optimal number of clusters as 3. Also, it was the same number of clusters as I assumed for the users by interpreting the data distribution plots in previous section for group-wise analysis.

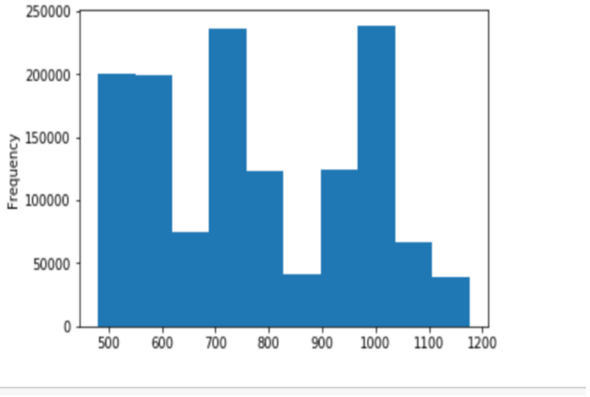


<u>K-Means Clustering – </u>After applying k-means clustering, the users were clustered in 3 groups without outlier users based on their event_type, ratings, and price. Following plots provide an idea about different traits of different user groups. However, users in a group are similar. So, product recommendations can be made to same user clusters based on their price range and ratings. As, different groups have users interested in different price ranges. Similarly, we can increase the scope of recommendation by introducing more features of user activity to the clusters and target accordingly.



```
filtered_label1['price'].plot(kind = 'hist')
```

```
]: <matplotlib.axes._subplots.AxesSubplot at 0x28d875db438>
```

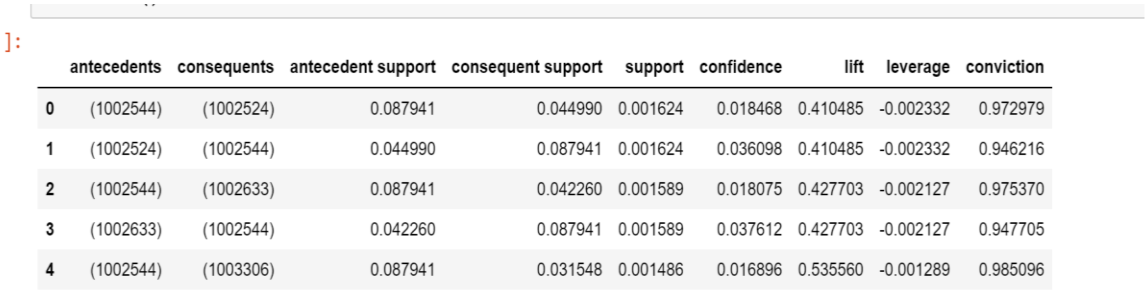


As per above graphs, the price range for the users in different clusters is not the same as we got in analysis in previous section. Now, for example- It can be used as a criterion further to target users for product recommendations that fall in specific purchasing power price ranges. Likewise, we can use different features in these groups for the product recommendations criterions. Further, product recommendations can be made to the users in same group. Also, category of products would have given more specific clusters. But due to scalability issue, it can be considered in future scope for this model. I received memory errors when applied category feature in k-means algorithm. So, I had to take it off the input features list.

Also, all the clustered users have majority of view/cart users. We can use purchase history users of each group with the ratings to further target them. Further, below are some of the modelling techniques - Association rules (Apriori algo) and product correlation matrix that can be applied in each cluster to find product utility similarity and user similarity to recommend products in future based on purchase history as discussed previously.

<u>Below are two algorithms that can be applied for each cluster to have user-group specific product recommendations- It will take care of both user similarity (like users) and high product utility correlations:</u>

<u>Association Rules (Apriori Algorithm)-</u> To find highly associated products for recommendations, I sliced the top purchased products from the dataset. Then, Apriori algorithm was applied to mine product patterns using associations rules. Finally, by using support, confidence, and lift values we can find strong product correlated sets. The data that was used in this section had less purchased frequent products, due to which the metrics values (support, confidence, and lift) were very low. Basically, I did not get strong correlated products due to the data. As the purchase history will increase on platform. We will get more frequent purchased products, and this will increase the accuracy to find strong correlated products from this model. So, it will get better with more purchased products by customers.

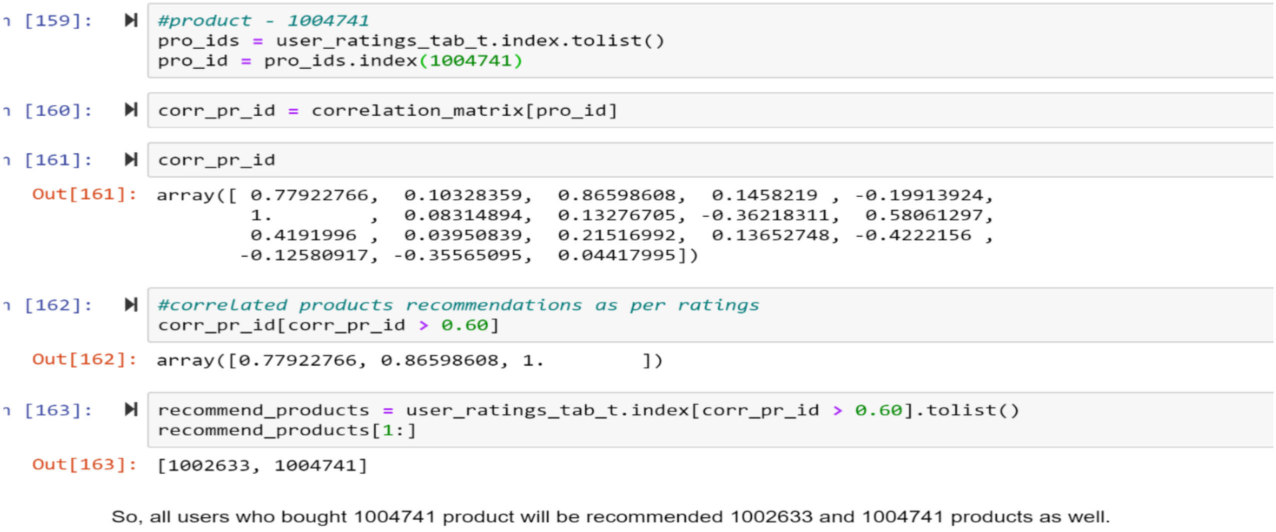


]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
0	(1002544)	(1002524)	0.087941	0.044990	0.001624	0.018468	0.410485	-0.002332	0.972979
1	(1002524)	(1002544)	0.044990	0.087941	0.001624	0.036098	0.410485	-0.002332	0.946216
2	(1002544)	(1002633)	0.087941	0.042260	0.001589	0.018075	0.427703	-0.002127	0.975370
3	(1002633)	(1002544)	0.042260	0.087941	0.001589	0.037612	0.427703	-0.002127	0.947705
4	(1002544)	(1003306)	0.087941	0.031548	0.001486	0.016896	0.535560	-0.001289	0.985096

Also, we can use it to mine associated users in a similar group to increase recommendation system efficiency by recommending highly correlated products to associated users in same group/cluster.

***Products Correlation Matrix (Based on ratings)* –** This matrix will provide correlated products for recommendation by using the rating provided by users for a product. So, according to the utility metrics for a product, recommendations will be facilitated to the users. We need good amount of purchase history to improve its performance. Again, I applied it to top products and finally used 10 users' rating to find correlations between the products. For each product, we can find n number of highly correlated products for the recommendations based on users' ratings. I was able to find 2 product recommendations based on utility-correlation for a randomly chosen product.



```
n [159]:    #product - 1004741
            pro_ids = user_ratings_tab_t.index.tolist()
            pro_id = pro_ids.index(1004741)

n [160]:    corr_pr_id = correlation_matrix[pro_id]

n [161]:    corr_pr_id

 Out[161]: array([ 0.77922766,  0.10328359,  0.86598608,  0.1458219 , -0.19913924,
                   1.        ,  0.08314894,  0.13276705, -0.36218311,  0.58061297,
                   0.4191996 ,  0.03950839,  0.21516992,  0.13652748, -0.4222156 ,
                  -0.12580917, -0.35565095,  0.04417995])

n [162]:    #correlated products recommendations as per ratings
            corr_pr_id[corr_pr_id > 0.60]

 Out[162]: array([0.77922766, 0.86598608, 1.        ])

n [163]:    recommend_products = user_ratings_tab_t.index[corr_pr_id > 0.60].tolist()
            recommend_products[1:]

 Out[163]: [1002633, 1004741]
```

So, all users who bought 1004741 product will be recommended 1002633 and 1004741 products as well.

e) Conclusion:

In conclusion, I would like to first comment on the scalability issues that impacted my analysis and models. Due to memory errors, I had to apply models on a smaller version of data or dataset with reduced features. So, as a future scope, this big data should be handled by other distributed and parallel technologies like spark, so that we can apply operations and models on any number of records or features.

Then, my analysis helped me understand that recommendations can applied to the users according to different scenarios as below –

New user – Recommend top rated products, track their views to recommend those products and other correlated/associated products. (Content- based recommendations)

User with a purchase history – Target the user into an appropriate group/cluster that they belong to, based on relevant features used in clustering. Then, use purchase history of that group to recommend products to the customers by using association rules and products-utility correlated matrix as explained above. (Collaborative recommendations)

There are many other ways as well to incorporate in a product recommendation system. However, as per my analysis of the customers' platform behavior I used the required models as mentioned in the above sections. As a future scope, we can dive more into the customer data and other models to improve the product recommendations via other techniques as well. Also, accuracy metrics should be developed for the models to improve the performance. Although, the best case would be to track if the recommendations are converting in sales in real-time scenario to measure the effectiveness of the system. Finally, all the above incorporated models and analysis can be used by an e-commerce store to increase their customer engagement and profits via sales.