CECS 550- Pattern Recognition

Group 6
Progress Report

Team Members :

Dharaneeswar Reddy Rami Reddy

Madhana Mohit Konduru

Shruthi Venkatachalam

Ashwanth Kumar Alagesan

Sai Abhijyan Tanikella

Under the guidance of Prof. Mahshid Fardadi

PROBEM STATEMENT:

Merchants often gain many new customers through promotions, but a significant portion of these customers are only interested in one-time deals. Therefore, the impact of promotions on long-term sales may be limited. To maximize return on investment (ROI) and reduce promotion costs, it is crucial for merchants to distinguish between one-time buyers and potential loyal customers and focus their efforts on converting the latter group.

In this project, you are provided a dataset with information on promotional shopping event from e-commerce platform. Your task is to design a system which will increase the ROI (in other words, you need to predict the probability that these new buyers would purchase items from the same merchants again within 6 months), reduce promotional cost, and identify one-time buyers.

Methodology/Steps taken:

1. Data:

- Initially, Data format_1 was chosen as we felt the feature engineering would be easy.
- However, not long after we started working with Data_format_1, we understood data cleaning might not be easy.
- We also encountered problems with merging and a tremendous number of Null values.
- To tackle these issues, we started working with Data_format_2 as it provided a single unified CSV file and was easier to manipulate and figure out its patterns.

```
df.head()
```

	user_id	age_range	gender	merchant_id	label	activity_log
0	34176	6.0	0.0	944	-1	408895:1505:7370:1107:0
1	34176	6.0	0.0	412	-1	17235:1604:4396:0818:0#954723:1604:4396:0818:0...
2	34176	6.0	0.0	1945	-1	231901:662:2758:0818:0#231901:662:2758:0818:0#...
3	34176	6.0	0.0	4752	-1	174142:821:6938:1027:0
4	34176	6.0	0.0	643	-1	716371:1505:968:1024:3

2. Pre-processing:

- The initial pre-processing step was replacing the null and unknown values with their respective numeric encoding.
- The second pre-processing step consisted of Splitting the activity_log column at every occurrence of #.
- The third step involved separating the features encoded in the activity_log by the ":" symbol.
- Now the new Data sort of looks like this.

```
new_df
```

	user_id	age_range	gender	merchant_id	label	item_id	category_id	brand_id	time_stamp	action_type

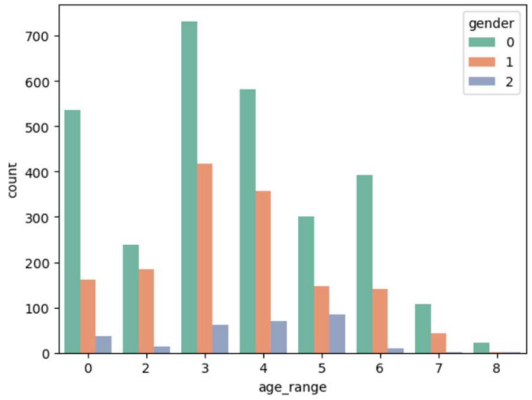
- Then finally, we filtered out the data based on our group's assigned item_id values. (group 6 : item_id : 801- 960).
- We also created a copy of our new pre-processed data and transformed it into Data_format_1, which we thought would be helpful during feature Engineering.

	user_id	age_range	gender	merchant_id	label	item_id	category_id	brand_id	time_stamp	action_type
598	34944	5	0	2116	-1	867	656	7334	1017	0
10455	252288	3	0	3990	-1	825	662	5644	0819	0
21417	210048	3	1	4255	-1	866	1213	1573	0711	2
21421	210048	3	1	4255	-1	866	1213	1573	0711	0
21422	210048	3	1	4255	-1	866	1213	1573	0711	0
...	...	...	...	...	...	...	...	...	...	...
27346150	230015	4	0	4255	-1	866	1213	1573	0710	2
27346152	230015	4	0	4255	-1	866	1213	1573	0710	0
27367647	64895	3	0	3491	-1	860	1238	3969	0529	0
27370519	5759	6	0	4255	-1	866	1213	1573	0815	2
27382574	96383	4	1	3462	-1	916	563	2997	0904	0

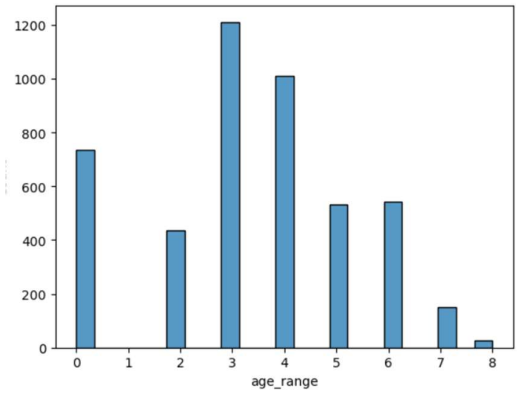
4644 rows × 10 columns

3. Exploratory Data Analysis:

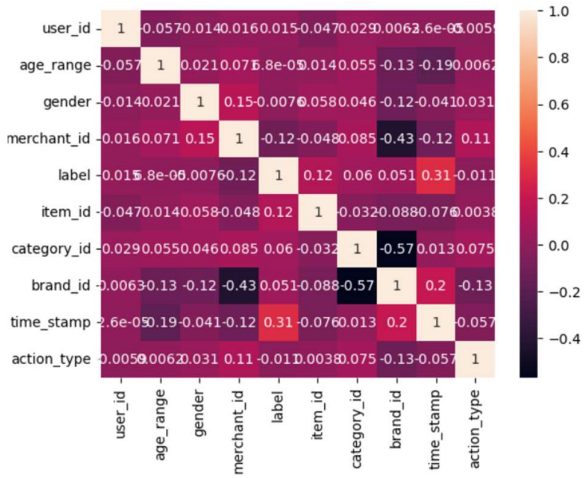
- Now with the pre-processed Dataset, we started our EDA.
- Our first experiment found that the highest number of females belonged to age_range three, and age_range 3,4,5 consisted of almost equal and most unknown genders.



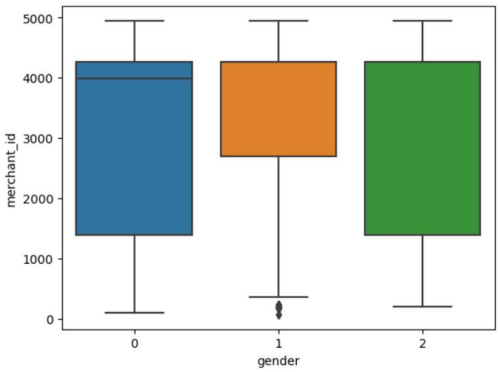
- Also found that age_range 1 was empty.



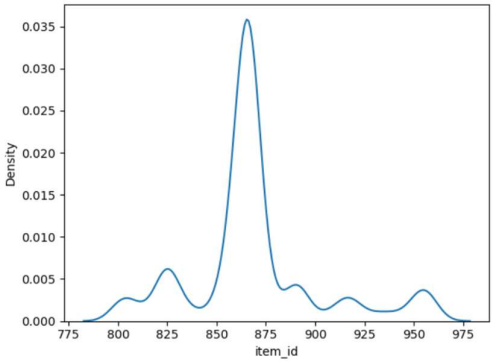
- Found the mean age range and gender plot using a box plot.
- We created a correlation heatmap to determine the relations between our initial features.

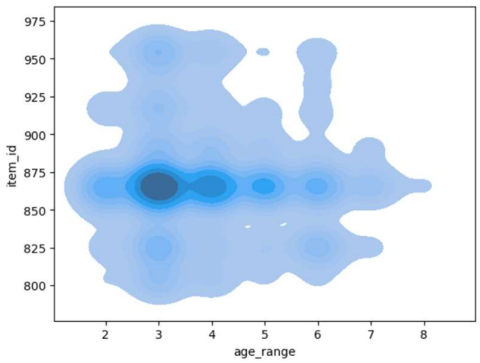


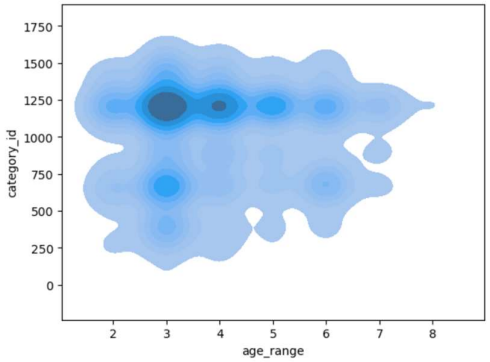
- Similarly figured out the gender split based on the merchants.

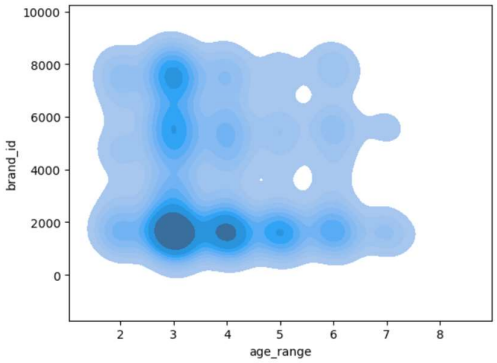


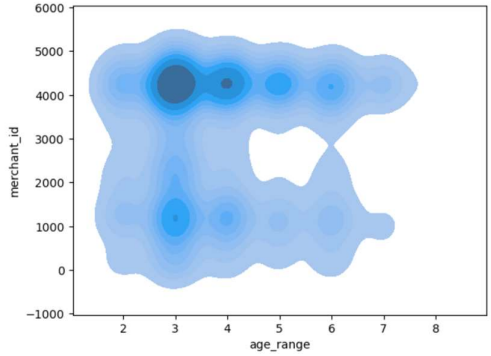
- The items around 850 to 875 seemed to be popular. So, the users, merchants and categories around these item_ids were analyzed.



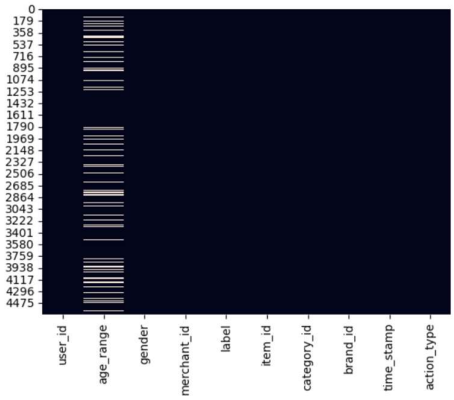


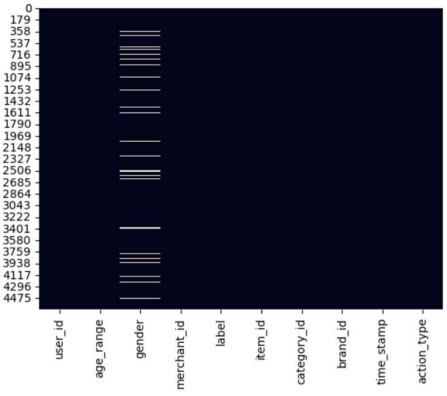






- Replacing and plotting the null values in age_range and gender.





4. Feature Engineering :

- As mentioned in the last pre-processing step, we created a new dataset in the form of Data_format_1 created from the processed_data in the pre-processing stage.

```
df=pd.read_csv('df_t.csv')
df_user_info=pd.read_csv('df_info.csv')
df_user_log=pd.read_csv('df_log.csv')
```

- We started with the df_user_log:

```
df_user_log.head()
```

	user_id	item_id	category_id	merchant_id	brand_id	time_stamp	action_type
0	34944	867	656	2116	7334	1017	0
1	252288	825	662	3990	5644	819	0
2	210048	866	1213	4255	1573	711	2
3	210048	866	1213	4255	1573	711	0
4	210048	866	1213	4255	1573	711	0

- Converted all the data into integer type for easy handling.
- Now we transform the age category into a binary response variable.

	user_id	merchant_id	label	gender	age_0	age_2	age_3	age_4	age_5	age_6	age_7	age_8
0	34944	2116	-1	0	0	0	0	0	1	0	0	0
1	252288	3990	-1	0	0	0	1	0	0	0	0	0
2	210048	4255	-1	1	0	0	1	0	0	0	0	0
3	210048	4255	-1	1	0	0	1	0	0	0	0	0
4	210048	4255	-1	1	0	0	1	0	0	0	0	0

- We considered three different perspectives user perspective, merchant perspective, and user_merchant combined perspective.

```
users = df_user_log.groupby('user_id')
```

```
merchants = df_user_log.groupby('merchant_id')
```

```
users_merchants = df_user_log.groupby(['user_id', 'merchant_id'])
```

- We could add features from each perspective, like **items_user or merchant_user**, where **user_items** describes the number of items a user has interacted with, and **merchant_user** describes the number of merchants the user has interacted with.
- Similarly, we were able to add a feature for merchant perspective and user merchant perspective
- **User_perspective features added : 19**

- **Merchant Perspective features added : 10**
- **User_merchant Perspective features added : 9**

ısers_merchant	items_merchant	categories_merchant	brands_merchant	dates_merchant	action_types_merchant	periods_merchant	items_user_merchant	categories_use
115	1	1	1	24	3	2	1	
155	1	1	1	80	3	5	1	
681	1	1	1	103	3	5	1	

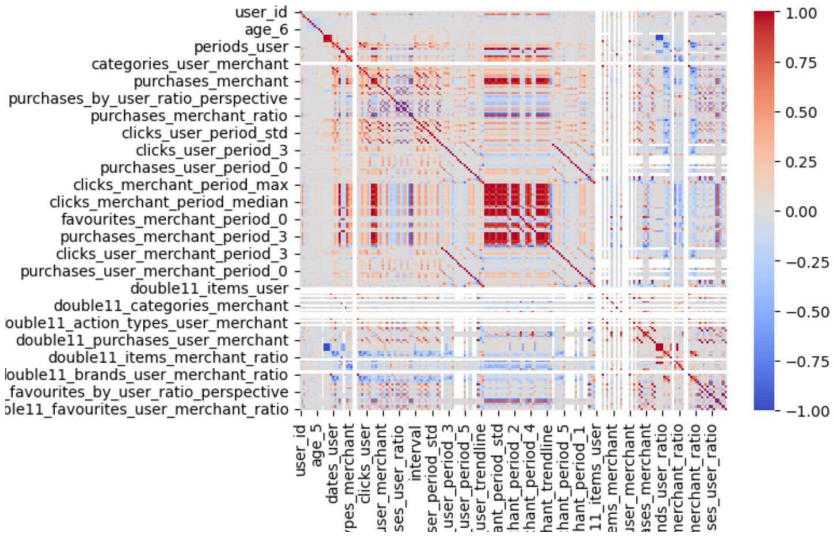
- Next were ratio features. They were chosen as they can represent a cumulative characteristic of all the features involved in the ratio.
- For example in the ratio features we can see how many **clicks were done for one merchant by a user**. These are really useful for determining the relationships between features.
- User_perspective features added : 6
- Merchant Perspective features added : 6
- User_merchant Perspective features added : 3
- To figure out the **time periods** where the user has been active or where the merchant has been interacted with we need to split the time_stamp into different no of periods.
- And as previously mentioned we need to consider all the perspectives while measuring the time periods and also Keeping in track the max and mean periods for each perspective
- By the end of this step we have around **120 features** with us.
- Next we check out the sale day November 11th features. Named as Double11 features
- We count the interaction done by user on D11 as new feautures.

ses_user	double11_favourites_user	double11_clicks_merchant	double11_purchases_merchant	double11_favourites_merchant	double11_clicks_user_merchant	do
NaN	NaN	81.0	7.0	2.0	NaN	
NaN	NaN	15.0	2.0	0.0	NaN	
NaN	NaN	25.0	12.0	0.0	NaN	
NaN	NaN	25.0	12.0	0.0	NaN	
NaN	NaN	25.0	12.0	0.0	NaN	
...	...	...	...	...	...	

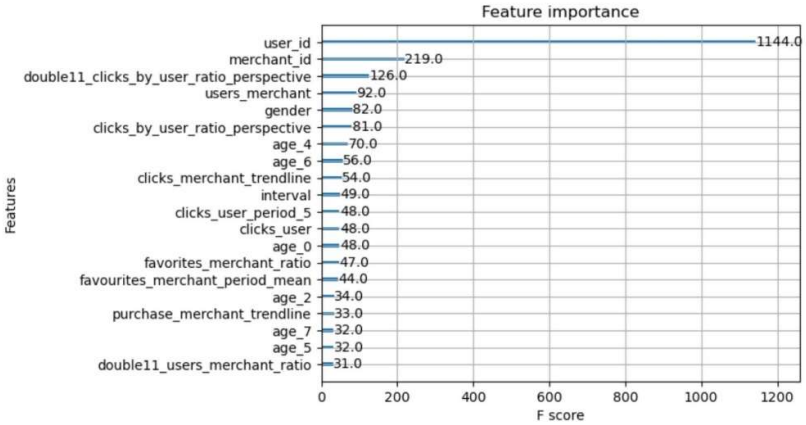
- Now we have around 160 features overall we had to figure out the best possible features out of these and ignore the others to improve **training efficiency** and avoid the **curse of dimensionality**.

5. Feature Ranking :

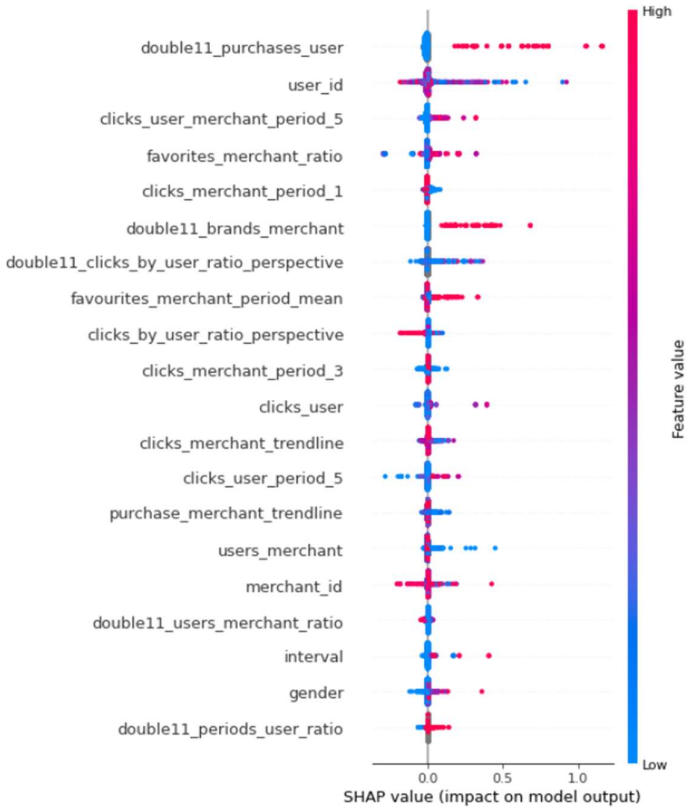
- Our initial idea was to use a correlation matrix and a threshold to weed out highly correlated factors and thereby reducing the dimensions. However, we had around 200 features and a heat map for corr matrix, something like below, which was illegible and not very useful for us.



- Then our second attempt was successful as we used a regression model to rank feature importance.



- Finally we decided to also check out SHAP values and cros check our features. The below gives th result for the same.



Conclusion:

In conclusion, we started with data format 2 as our Dataset and performed pre-processing techniques to clean out the data. Secondly, performed EDA on the data to understand how the data is distributed. Finally started feature engineering and generated a lot of valuable features. We also achieved some basic feature rankings on the final set of feature.

This was a brief on our progress, the next steps such as final feature selection, model selection, train and testing and finally insights from the data are yet to be performed and will be completed by the assigned Date.