E-Commerce Churn Case Study

Author: **Tom Mathews**

Overview

1. Import necessary modules
2. Start Spark Session with required configuration

Data Ingestion

1. Read the inputdata and created Spark dataframes for each of the above datasets.
2. Checked the data structure.

Data Exploration

1. 5 most popular products sold?

```
+----------+-----+
|product_id|count|
+----------+-----+
|   1004856|28944|
|   1004767|21806|
|   1004833|12697|
|   1005115|12543|
|   4804056|12381|
+----------+-----+
```

2. 5 most popular brands

brand	count
null	6113008
samsung	5282775
apple	4122554
xiaomi	3083763
huawei	1111205
lucente	655861

The most popular single brand is "Samsung" but the by count the most popular would be the other brands.

3. Number of unique users

count(DISTINCT user_id)
3022290

4. The most active user on the platform

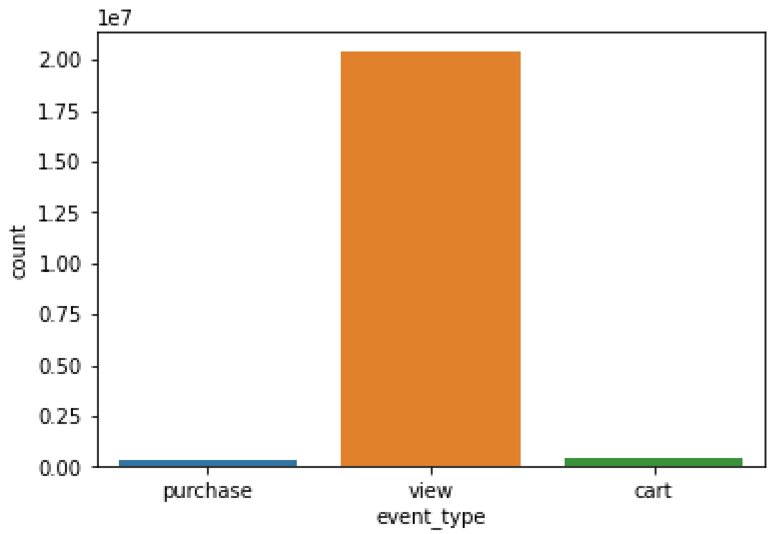
user_id	count
512475445	7436
512365995	4013
526731152	2912
512505687	2894
513021392	2862

5. Average and Maximum price for smartphones purchased by the customers

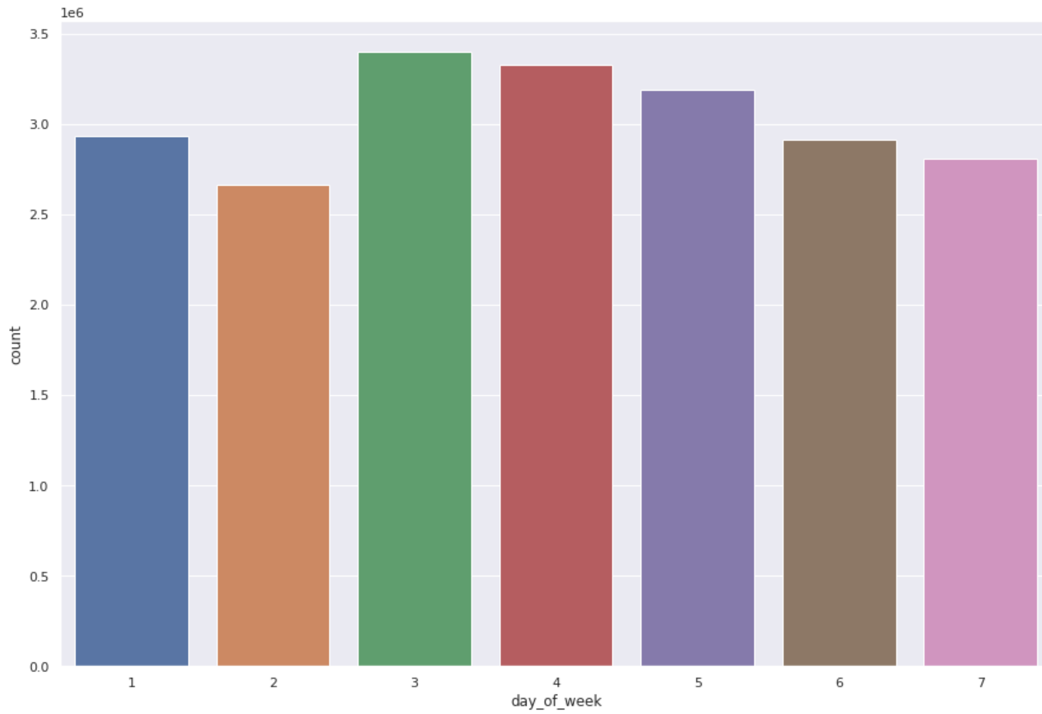
price_avg	price_max
471.9470819575878	2110.45

6. Event-type funnel distribution

```
+-----------+-------+
|day_of_week|  count|
+-----------+-------+
|          1|5855995|
|          6|5829660|
|          3|6801885|
|          5|6380367|
|          4|6652532|
|          7|5606796|
|          2|5321529|
+-----------+-------+
```



7. Traffic on different days of the week



Feature Engineering

Missing Values

```
+----------+----------+----------+-----------+-------------+---------+-----+-------+------------+-----------+
|event_time|event_type|product_id|category_id|category_code|    brand|price|user_id|user_session|day_of_week|
+----------+----------+----------+-----------+-------------+---------+-----+-------+------------+-----------+
|       0.0|       0.0|       0.0|        0.0| 1.3515609E7|6113008.0|  0.0|    0.0|         2.0|        0.0|
|       0.0|       0.0|       0.0|        0.0|     31.8398|  14.4009|  0.0|    0.0|         0.0|        0.0|
+----------+----------+----------+-----------+-------------+---------+-----+-------+------------+-----------+
```

The first row is the count and the second row is the percentage as compared to the total number of values.

- Removed all duplicate rows where all columns are the same.
- category_code: We can mark all missing category codes as "unavailable".
- brands: Later in the code, we'll be first checking if we can find the missing brand from other records or mark it as "others".

Additional Columns

1. Generating 2 columns from category code
2. Activities in a session by the user
3. View count for a product by the user
4. View count for the secondary category by the user
5. Average shopping expense for a product category
6. Session count for a user
7. Generating the hour variable

Modifications

- Binning the column hour to 4 bins.
- Reduction in brands for analysis: Top 20 + ‘others'
- Generating 'is_purchased' variable
- Remove duplicate rows
- Remove redundant columns

Model Engineering

Transform the above cleaned data and then run the following steps.

- **StringIndexer** for categorical columns that have string values.
- **OneHotEncoder** for all numerical categorical columns and the string indexed columns.
- **VectorAssembler** for numerical columns and all the one hot encoded columns.

Save the transformed dataframe to a parquet file so that we can use the same for each of the different models.

Logistic Regression

Best Parameters after HyperParameter Tuning

- regParam = 0.001
- maxIter = 15
- threshold = 0.5

Decision Tree

Best Parameters after HyperParameter Tuning

- maxDepth = 20

- maxBins = 128
- impurity = gini

Random Forest

Best Parameters after HyperParameter Tuning

- numTrees = 30
- maxDepth = 9
- maxBins = 64
- impurity = entropy

Model Performance and Inferences

Logistic Regression

```
*************************
Model: LogisticRegression
Confusion Matrix
[[ 29471.  76855.]
 [ 21230. 142849.]]

False Positive Rate : 0.12938889193620146
Precision           : 0.6501884353493792
Recall              : 0.8706111080637985
F1-Score            : 0.3753621988575213
Accuracy            : 0.637266322738115
*************************
```

Decision Tree

```
*************************
Model: DecisionTree
Confusion Matrix
[[ 80641.  25685.]
 [122783.  41296.]]

False Positive Rate : 0.7483163598022904
Precision           : 0.6165330466848807
Recall              : 0.2516836401977096
F1-Score            : 0.5206844229217111
Accuracy            : 0.4509421053604778
```

****

Random Forest

****

Model: RandomForest
Confusion Matrix
[[41047. 65279.]
 [50927. 113152.]]

False Positive Rate : 0.3103809750181315
Precision : 0.6341498954778038
Recall : 0.6896190249818684
F1-Score : 0.4139889056984367
Accuracy : 0.5702520293633624
****

	Name	ConfusionMatrix	FalsePositiveRate	Precision	Recall	F1-Score	Accuracy
0	LogisticRegression	[[29471.0, 76855.0], [21230.0, 142849.0]]	0.129389	0.650188	0.870611	0.375362	0.637266
1	DecisionTree	[[80641.0, 25685.0], [122783.0, 41296.0]]	0.748316	0.616533	0.251684	0.520684	0.450942
2	RandomForest	[[41047.0, 65279.0], [50927.0, 113152.0]]	0.310381	0.63415	0.689619	0.413989	0.570252

Feature Importance

Logistic Regression

	feature_index	feature_name	coef	mean	std	std_coef	feature_importance
0	76	user_product_count	0.069376	8.327858	11.987979	0.831679	0.831679
1	78	category_secondary_count	-0.006947	33.429711	59.136615	-0.410850	0.410850
2	79	session_count	0.002625	70.474443	108.726912	0.285459	0.285459
3	55	brand_stridx_ohe_xiaomi	-0.400531	0.113412	0.317096	-0.127007	0.127007
4	54	brand_stridx_ohe_others	-0.311619	0.181774	0.385658	-0.120178	0.120178
...	...	...	...	...	...	...	...
75	47	category_code_secondary_stridx_ohe_tshirt	-0.430444	0.000021	0.004550	-0.001958	0.001958
76	60	brand_stridx_ohe_acer	-0.018840	0.010155	0.100261	-0.001889	0.001889
77	49	category_code_secondary_stridx_ohe_cultivator	-0.073110	0.000010	0.003091	-0.000226	0.000226
78	40	category_code_secondary_stridx_ohe_trainer	-0.009815	0.000384	0.019585	-0.000192	0.000192
79	41	category_code_secondary_stridx_ohe_ironing_board	0.008253	0.000416	0.020382	0.000168	0.000168

Decision Tree

	FeatureNum	FeatureName	FeatureImportance
0	6	session_count	0.168214
1	7	category_code_primary_stridx_ohe_electronics	0.168214
2	3	user_product_count	0.152178
3	5	category_secondary_count	0.147429
4	2	total_activities_count	0.127708
5	1	day_of_week	0.122082
6	4	average_price	0.099803
7	21	category_code_secondary_stridx_ohe_unavailable	0.025748
8	79	hours_ohe_2	0.015651
9	78	hours_ohe_1	0.014460
10	0	price	0.013018
11	60	brand_stridx_ohe_apple	0.011905
12	63	brand_stridx_ohe_huawei	0.011212
13	62	brand_stridx_ohe_xiaomi	0.007774
14	8	category_code_primary_stridx_ohe_unavailable	0.006899
15	9	category_code_primary_stridx_ohe_appliances	0.005326
16	64	brand_stridx_ohe_oppo	0.004740
17	10	category_code_primary_stridx_ohe_computers	0.004708
18	61	brand_stridx_ohe_others	0.004395
19	65	brand_stridx_ohe_lg	0.003474

Random Forest

	FeatureNum	FeatureName	FeatureImportance
0	3	user_product_count	0.328183
1	2	total_activities_count	0.220038
2	6	session_count	0.117040
3	7	category_code_primary_stridx_ohe_electronics	0.117040
4	5	category_secondary_count	0.052541
5	8	category_code_primary_stridx_ohe_unavailable	0.036847
6	4	average_price	0.036380
7	1	day_of_week	0.035777
8	62	brand_stridx_ohe_xiaomi	0.034705
9	21	category_code_secondary_stridx_ohe_unavailable	0.031552
10	9	category_code_primary_stridx_ohe_appliances	0.022174
11	60	brand_stridx_ohe_apple	0.020196
12	61	brand_stridx_ohe_others	0.013736
13	63	brand_stridx_ohe_huawei	0.011203
14	23	category_code_secondary_stridx_ohe_audio	0.005478
15	79	hours_ohe_2	0.004456
16	22	category_code_secondary_stridx_ohe_kitchen	0.004449
17	10	category_code_primary_stridx_ohe_computers	0.003646
18	25	category_code_secondary_stridx_ohe_environment	0.002362
19	78	hours_ohe_1	0.001879

Inferences

- From the statistics we can see that for our three models, **Logistic Regression** seems to have performed the best with about 63.73% accuracy whereas Random Forest and Decision Tree performed comparatively bad with 57.02% & 45.09% accuracy respectively.
- From feature importance of all three models, we can see that total_activites_count, session_count, user_product_count are few very important features.