Project Report

CART ABANDONMENT PREDICTION IN AN E-COMMERCE WEBSITE

Table of Contents

INDUSTRY REVIEW

Abstract:

The ease and Convenience provided by Online shopping has made Online Stores grow significantly than the brick-and-mortar stores. Since the experience has moved on to the digital world, the touch points of the products have been completely changed. We must identify and study the touch points to optimize the customers purchase journey.

Customers will be directed to the ecommerce website either through advertisements or from search engines. They view the products and interact with the information provided on the site. Once they have done that, they either choose to add it to the cart for future purchase or as a list like Wishlist. This is different from our traditional purchases where the customer will see the product in person and choose whether to buy it or not.

Now after adding the products to the cart, most of the time the cart gets abandoned without purchasing. Numerous factors contribute to this phenomenon, leading to substantial revenue losses for companies. This aspect becomes a crucial area for study and improvement in the realm of sales optimization.

Current Practices:

There are various measures in place to decrease the cart abandonment that have been a major hurdle for many ecommerce platforms. But all the measures are only taken post cart abandonment. Few of the Practices are mentioned below,

a) Remarketing: The Platforms often employ remarketing strategies targeting the users who have abandoned the cart using targeted ads using cookies.
b) Email: Automated Emails to remind the customers about their cart being left unpurchased to remind the users sometimes includes discounts or reduction in price.
c) Data Analysis: Performing Data Analysis to understand the user behavior and thereby find the reason behind the cart abandonment.
d) A/B testing: Optimizing the websites and webpages in the site and monitoring the changes in cart abandonment.

Business Problem Statement:

a. Understanding the Problem:

Like footfalls for a retail store or a supermarket, Views of the products plays an important role in Ecommerce Website. Even though customers view the products several times, only a few views it with a purchasing intention and add it to their cart. But not all those who have added the product to the cart will go through with the purchase. Understanding and addressing cart abandonment is a critical challenge for online retailers. Cart abandonment leads to revenue loss and affects the overall conversion rate.

b. Business Objective:

The objective is to help the company understand the factors and behaviors affecting the purchase decision of a customer and then with those factors predict the possibility of purchase of a product after being added to the Cart. Hence potentially identifying the transactions that may lead to cart abandonment in the future.

c. Approach:

By understanding the data at hand along with domain knowledge, we extract the metrics and features that are relevant to the field of study. By using the features and appropriate machine learning model, which is most likely to be a classification model, this project aims to develop a predictive model to identify and classify users who are likely to convert after adding products to their carts. Thus, helping the company to make informed decisions such as Personalized Retargeting Ads, Incentives and Discounts to recover revenue loss from cart abandonment.

DATA UNDERSTANDING:

The dataset contains the record of all the types of events that have happened in an Ecommerce platform for the month of October 2019. Every time a user is logging in, a new session id will generate in which the user may view any number of products, add the product to the cart and purchase the product. The dataset is very useful in studying customer behavior on the Ecommerce site.

We have 4,24,48,764 records in our dataset each recording unique event of an user for a particular product.

Data Dictionary:

Dataset title	eCommerce behavior data from multi category store
Source	Kaggle
Dataset Owner	Michael Kechinov
Link to Dataset	[Kaggle Website]

Variables	**Definition**
event_time	Time when event happened at (in UTC)
event_type	Different kind of event: view,cart,purchase
product_id	ID of a product
category_id	Product's category ID- Unique for each product
category_code	Product's category code name - Combination of Category with Subcategory
brand	Name of the brand of the product
price	Float price of a product.
user_id	Permanent user ID that has been assigned to each user
user_session	Temporary user's session ID. Generated for each session of the users

	event_time	event_type	product_id	category_id	category_code	brand	price	user_id	user_session
0	2019-10-01 00:00:00+00:00	view	44600062	2103807459595387724	NaN	shiseido	35.79	541312140	72d76fde-8bb3-4e00-8c23-a032dfed738c
1	2019-10-01 00:00:00+00:00	view	3900821	2053013552326770905	appliances.environment.water_heater	aqua	33.2	554748717	9333dfbd-b87a-4708-9857-6336556b0fcc
2	2019-10-01 00:00:01+00:00	view	17200506	2053013559792632471	furniture.living_room.sofa	NaN	543.1	519107250	566511c2-e2e3-422b-b695-cf8e6e792ca8
3	2019-10-01 00:00:01+00:00	view	1307067	2053013558920217191	computers.notebook	lenovo	251.74	550050854	7c90fc70-0e80-4590-96f3-13c02c18c713
4	2019-10-01 00:00:04+00:00	view	1004237	2053013555631882655	electronics.smartphone	apple	1,081.98	535871217	c6bd7419-2748-4c56-95b4-8cec9ff8b80d

Variable Categorization

a. Variables:
   i. Numerical : 4
   ii. Categorical : 5
b. Total columns : 9

We can see that there is a discrepancy in the data types of the variables which we must handle before further analysis.

```
RangeIndex: 42448764 entries, 0 to 42448763
Data columns (total 9 columns):
 #   Column         Dtype
---  ------         -----
 0   event_time     object
 1   event_type     object
 2   product_id     int64
 3   category_id    int64
 4   category_code  object
 5   brand          object
 6   price          float64
 7   user_id        int64
 8   user_session   object
dtypes: float64(1), int64(3), object(5)
memory usage: 2.8+ GB
```

```
RangeIndex: 42448764 entries, 0 to 42448763
Data columns (total 9 columns):
 #   Column         Dtype
---  ------         -----
 0   event_time     datetime64[ns, UTC]
 1   event_type     category
 2   product_id     object
 3   category_id    object
 4   category_code  object
 5   brand          object
 6   price          float64
 7   user_id        object
 8   user_session   object
dtypes: category(1), datetime64[ns, UTC](1), float64(1), object(6)
memory usage: 2.6+ GB
```

We must change event time into datetime datatype and all the id columns into object datatypes.

a. Variables:
   i. Numerical : 1
   ii. Categorical : 7
   iii. Datetime : 1
b. Total columns : 9

Data Preprocessing:

a. Redundant Columns:
   The column category_id has 624 categories which cannot be explained in a proper manner, we drop the column from further analysis.
b. Null Values treatment:
   We can see that there are lot of Null values in columns category code and brand variable and few in user session column which will be handled after feature engineering.

```
event_time        0.000000
event_type        0.000000
product_id        0.000000
category_id       0.000000
category_code    31.839818
brand            14.410502
price             0.000000
user_id           0.000000
user_session      0.000005
dtype: float64
```

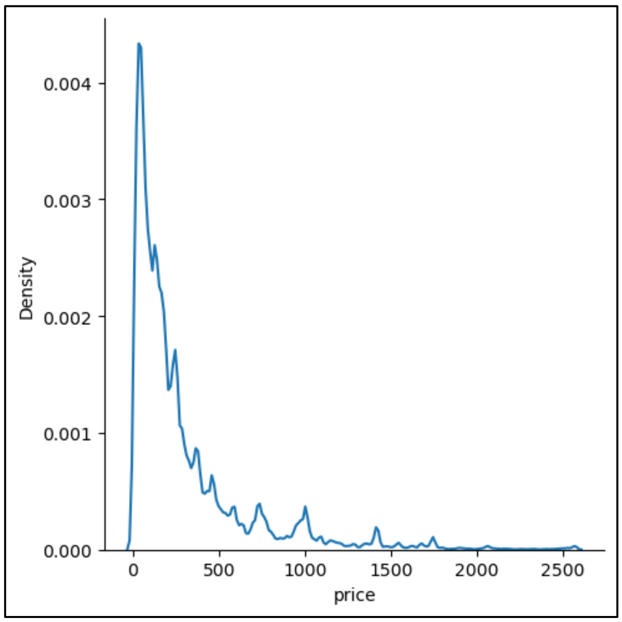
Problem complexity:

The dataset is not the correct format for our prediction of the purchase event; Hence, we need to transform it to usable format with Feature Engineering.

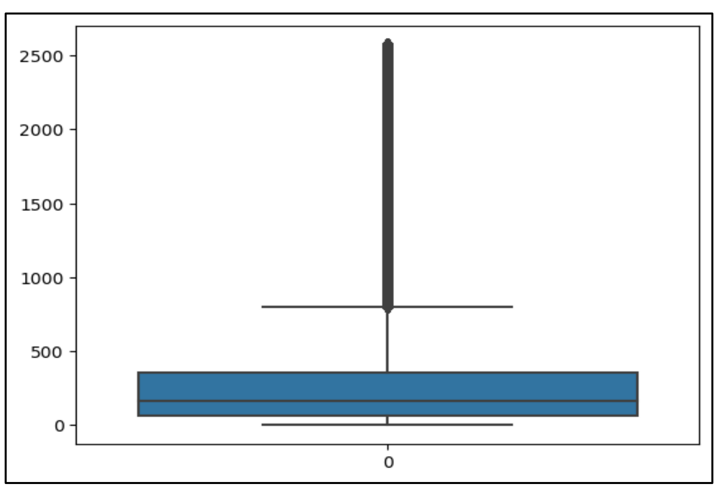
PRIMARY DATA EXPLORATION:

Distribution of Variables and Outlier Detection:

We have a single numeric column, Price. We will check for its distribution and presence of outliers.



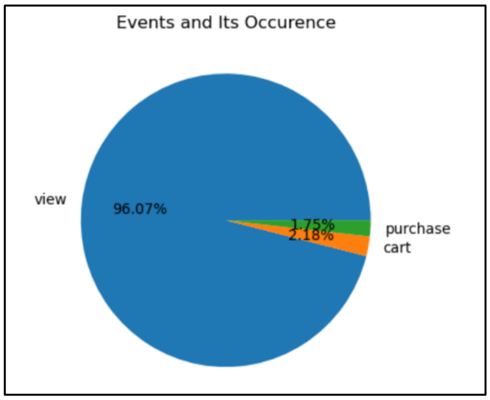
The Price columns is right skewed meaning that are large number of products with low price (0 to 500 dollars) and a smaller number of products with high price (above 500 dollars).



There are large number of outliers present in the upper region of the data in the price column. We are not removing the outliers since they are crucial for our analysis and model building.

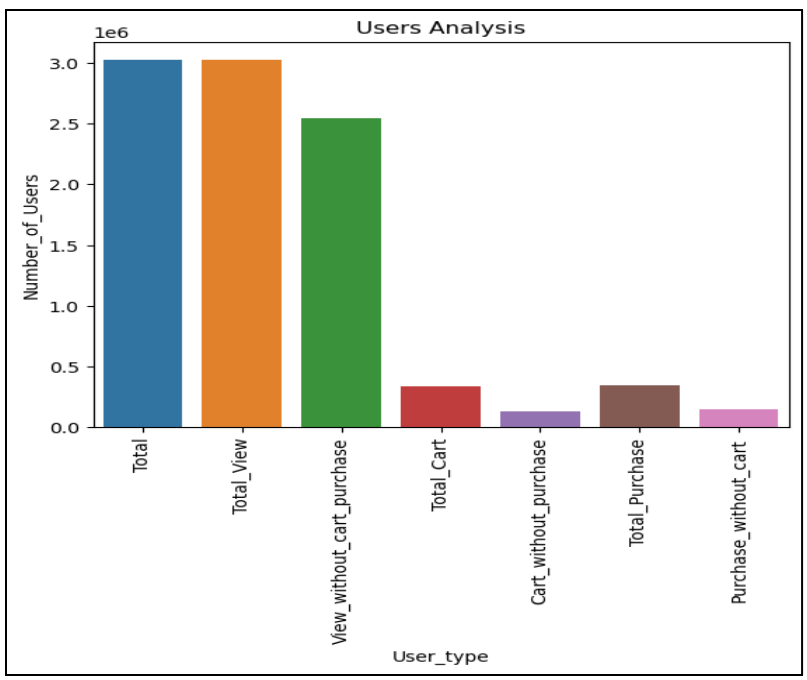
Project Justification:

We can see that in the dataset nearly 96 % of records are for view event types, only 2.18 % are for the event of adding to the cart and 1.75 % of the data are for purchasing event. Our focus will be the events - cart and purchase for our problem.

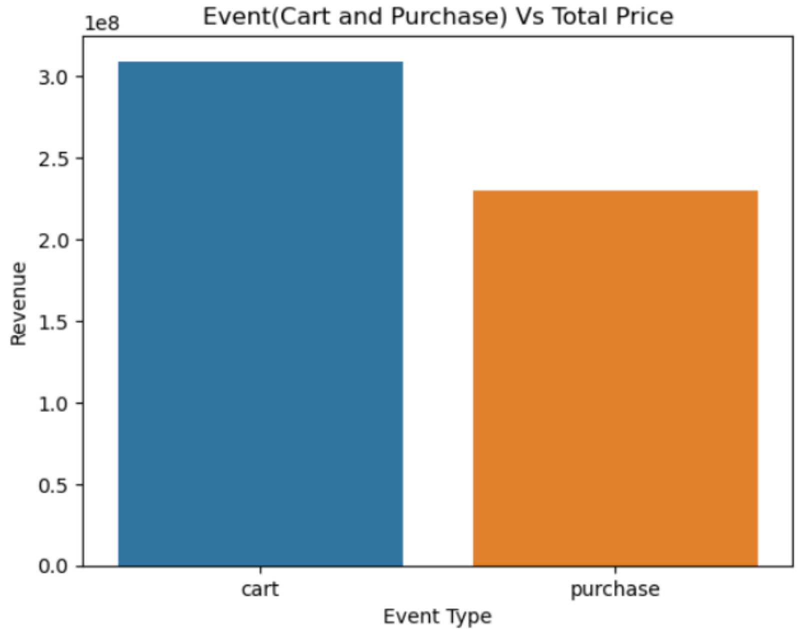


By understanding the number of users in each event type, we can identify the opportunity for growth for the business and we can implement a strategy based on the findings.

	User_type	Number_of_Users
0	Total	3022290
1	Total_View	3022130
2	View_without_cart_purchase	2540832
3	Total_Cart	337117
4	Cart_without_purchase	134340
5	Total_Purchase	347118
6	Purchase_without_cart	144341



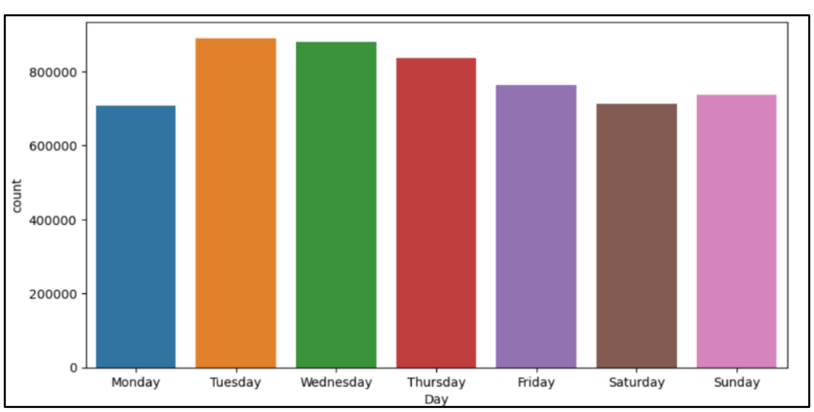
In the chart, the number of users who add products to the cart and those who purchase may seem to be the same or even higher. But after proper analysis, we can find that a significant number of users who added the product to the cart but didn't purchase. It is approximately 40% of the users who had added the products to their cart.



We add up all the products in the cart and find their revenue if they have been sold and compare that amount with the actual revenue from the products sold. There is a difference in the Expected revenue and Actual. This is our Revenue loss due to the cart abandonment. In this analysis we have not removed the products which have been purchased without adding them to the cart. If we have done that, the difference will be much higher. Hence by solving the problem, we can boost our sales to a significant level.

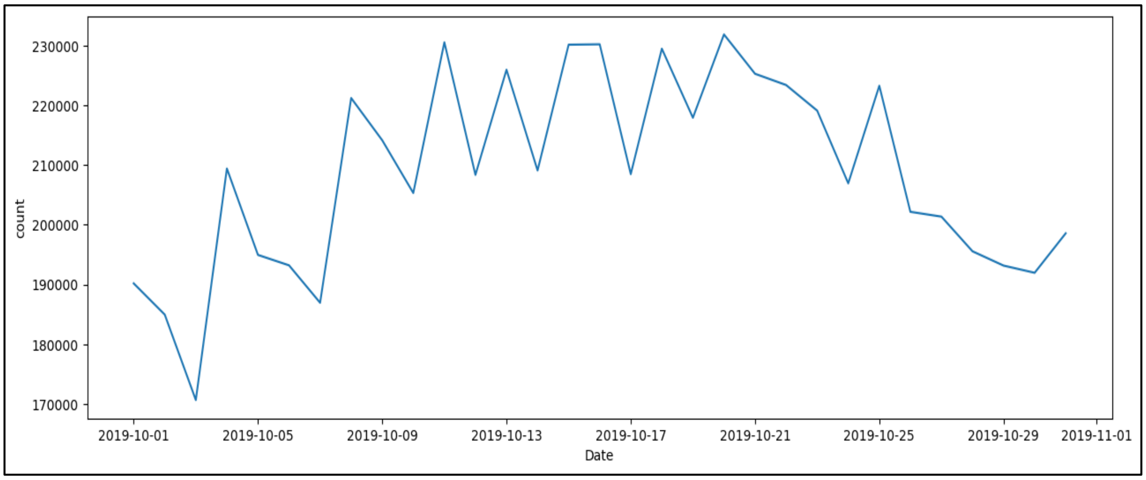
Website Traffic Analysis:

a. Weekday Traffic:



The Ecommerce website had high traffic on Tuesday, Wednesday, Thursday, and the same amount of traffic on other days.

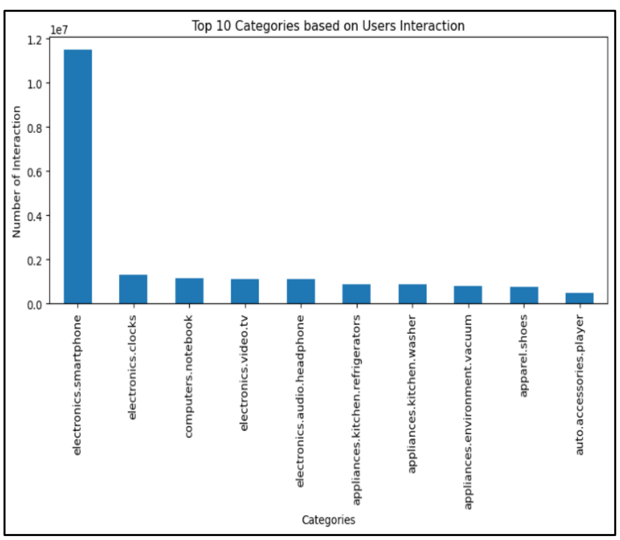
b. Daily Trend:

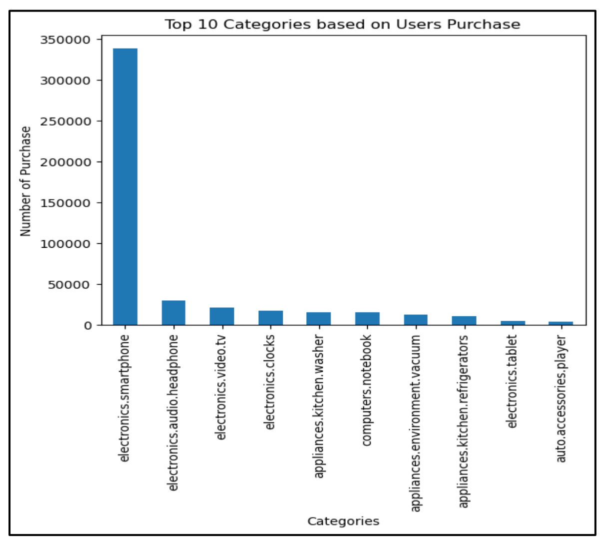


We can see that the number of users to the site increased up to the middle of the month and started to decrease at the month end.

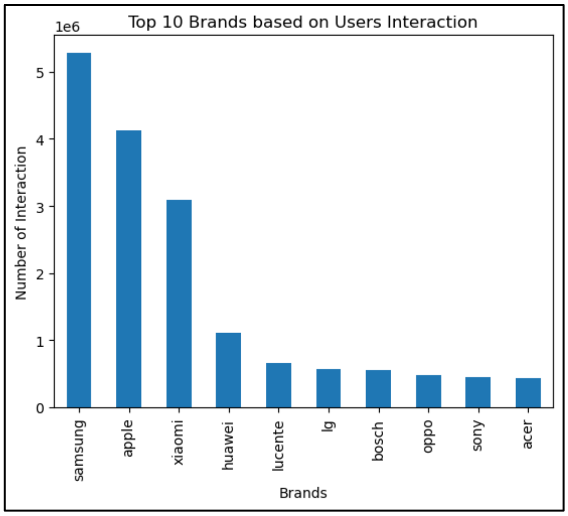
Product Analysis:

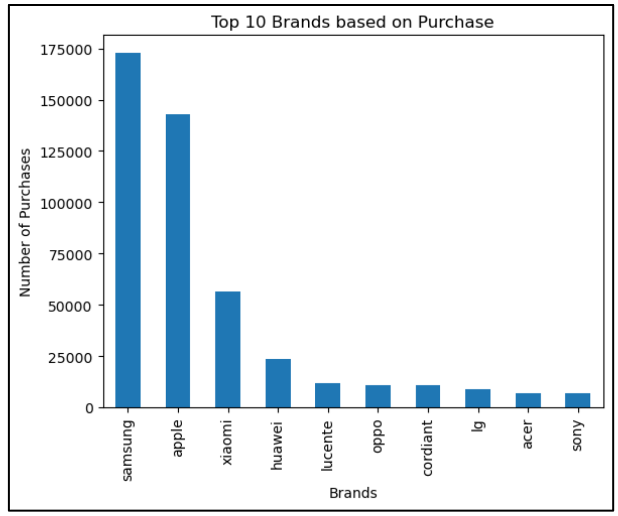
We will now try to understand more about the products that are sold on the site. We can gather insights from the analysis that can help us understand how we can solve the problem at hand.





The smartphone sub-category is the most interacted and purchased category. The electronics and appliances category occupies the top position in the traffic as well as purchase.





Samsung, Apple, Xiaomi, Huawei, Lucent are the top 5 selling brands on the platform, While the top 3 being exponentially higher than the rest of the brands.

FEATURE ENGINEERING:

To predict whether the product added to the cart will be purchased or not, we must study how customers behave inside the ecommerce platform. The metrics which define the customer's behavior are found using domain knowledge and extracted from the already existing variables.

This can be done by using Feature Engineering where we create new variables or extract information from the existing variables.

Variables to be Extracted:

a) Day of the week
b) Main Category
c) Subcategory

Variables to be Created:

a) Duration of the session(seconds)
b) Number of Activities in a Session
c) Time between session(minutes)
d) Is_purchased

Is_purchased column is the target variable that we are trying to predict.

Here we are transforming the dataset majorly by splitting it into two datasets. It is based on whether the customer is coming to the website for the first time or a returning customer. This is because the Time between session will not be available for the First-time customer.

DATA TRANSFORMATION:

We need to transform the dataset into a format which will be usable for our Event – prediction. For this purpose, we select only users who have added products to their carts or users who have purchased the products.

Now for each user on a particular session, the user may have added a particular product to the cart. We need information on the status of the transaction i.e. either purchased or abandoned in a single record.

	event_time	event_type	product_id	category_id	category_code	brand	price	user_id	user_session	is_purchased
331	2019-10-01 00:05:14 UTC	cart	5100816	2053013553375346967	NaN	xiaomi	29.51	550121407	6f623695-9581-4633-813f-825b8760c7ae	0.0
583	2019-10-01 00:09:33 UTC	cart	1002524	2053013555631882655	electronics.smartphone	apple	515.67	524325294	0b74a829-f9d7-4654-b5b0-35bc9822c238	1.0
680	2019-10-01 00:11:00 UTC	cart	4804056	2053013554658804075	electronics.audio.headphone	apple	161.98	533624186	e5ac3caa-e6d5-4d6b-ae06-2c18cd9ca683	0.0
1325	2019-10-01 02:17:59 UTC	cart	1004833	2053013555631882655	electronics.smartphone	samsung	174.76	536415846	685b5b42-f597-4a69-ab4c-ef96a30bc454	0.0
1654	2019-10-01 02:19:36 UTC	cart	1005003	2053013555631882655	electronics.smartphone	huawei	258.21	513632293	f2cc68f7-39d1-4a50-9dcf-f2a0921bdfda	1.0

Final Data frame for the First-time users after data transformation and feature engineering.

	brand	price	Day	category_1	category_2	activity_count	duration	is_purchased
0	apple	515.67	Tuesday	electronics	smartphone	4	112.0	1.0
1	apple	161.98	Tuesday	electronics	audio	4	147.0	0.0
2	samsung	174.76	Tuesday	electronics	smartphone	7	742.0	0.0
3	huawei	258.21	Tuesday	electronics	smartphone	16	843.0	1.0
4	xiaomi	360.08	Tuesday	electronics	smartphone	16	843.0	1.0

Final Data frame for the Returning users after data transformation and feature engineering.

	brand	price	Day	category_1	category_2	activity_count	time_between_session	duration	is_purchased
0	samsung	241.19	Tuesday	electronics	smartphone	8	0.57	334.0	0.0
1	apple	809.72	Tuesday	electronics	smartphone	3	7.80	24.0	0.0
2	xiaomi	197.55	Tuesday	electronics	smartphone	3	1.02	117.0	0.0
3	meizu	101.65	Tuesday	electronics	smartphone	5	5.23	285.0	1.0
4	samsung	388.68	Tuesday	electronics	smartphone	3	8.47	301.0	0.0

EDA – Transformed Dataset

Since we have transformed the dataset, we need to perform EDA on the transformed datasets.

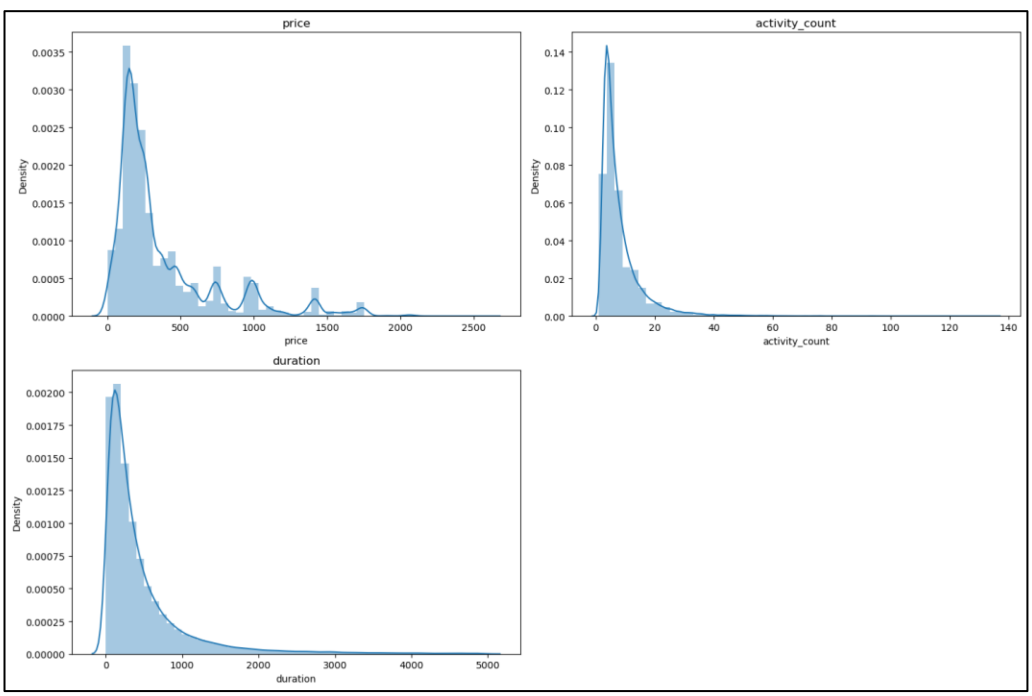
Variable Categorization-First Time Users

a. Independent Variables:
   i. Numerical : 3
   ii. Categorical : 4
b. Target Variable
   i. Categorical : 1
c. Total columns : 8

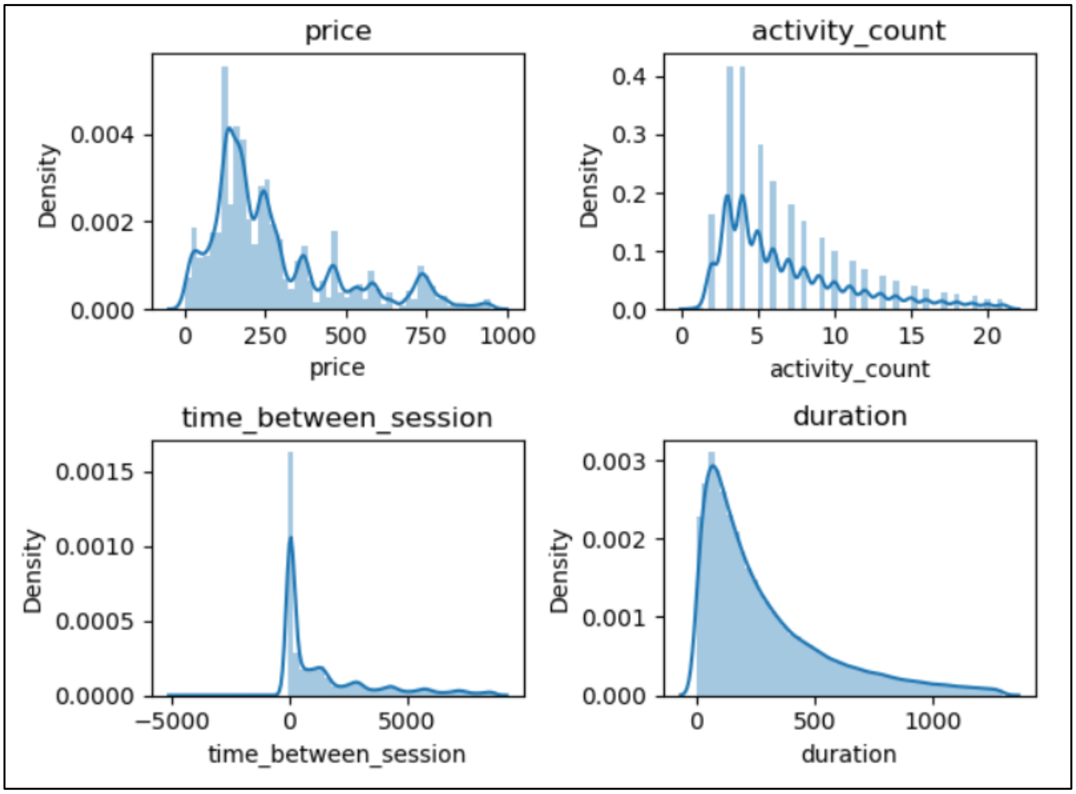
Variable Categorization- Returning Users

a. Independent Variables:
   i. Numerical : 4
   ii. Categorical : 4
b. Target Variable
   i. Categorical : 1
c. Total columns : 9

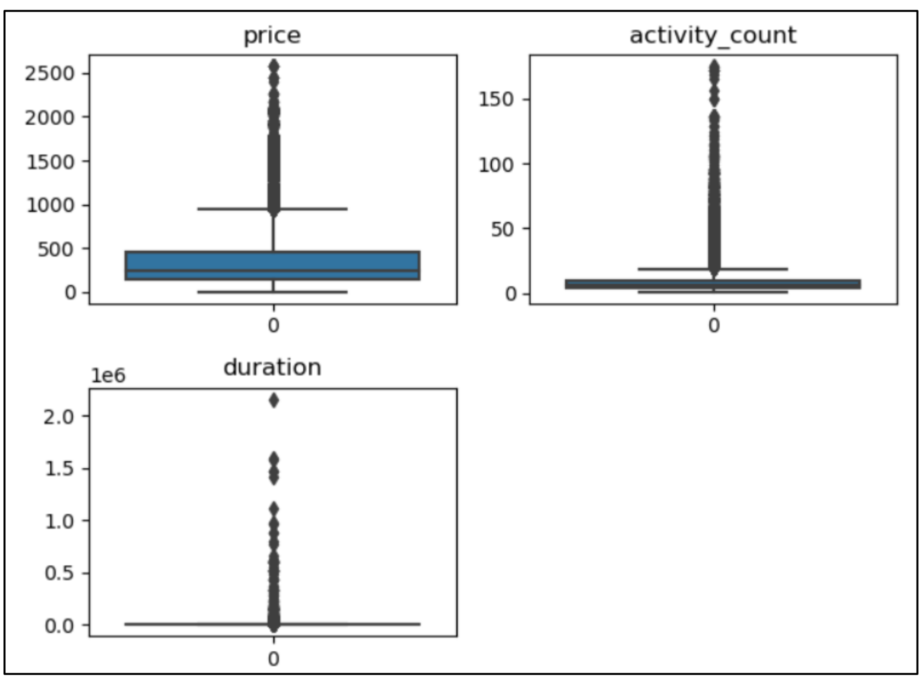
Distribution of Variables – First Time Users:



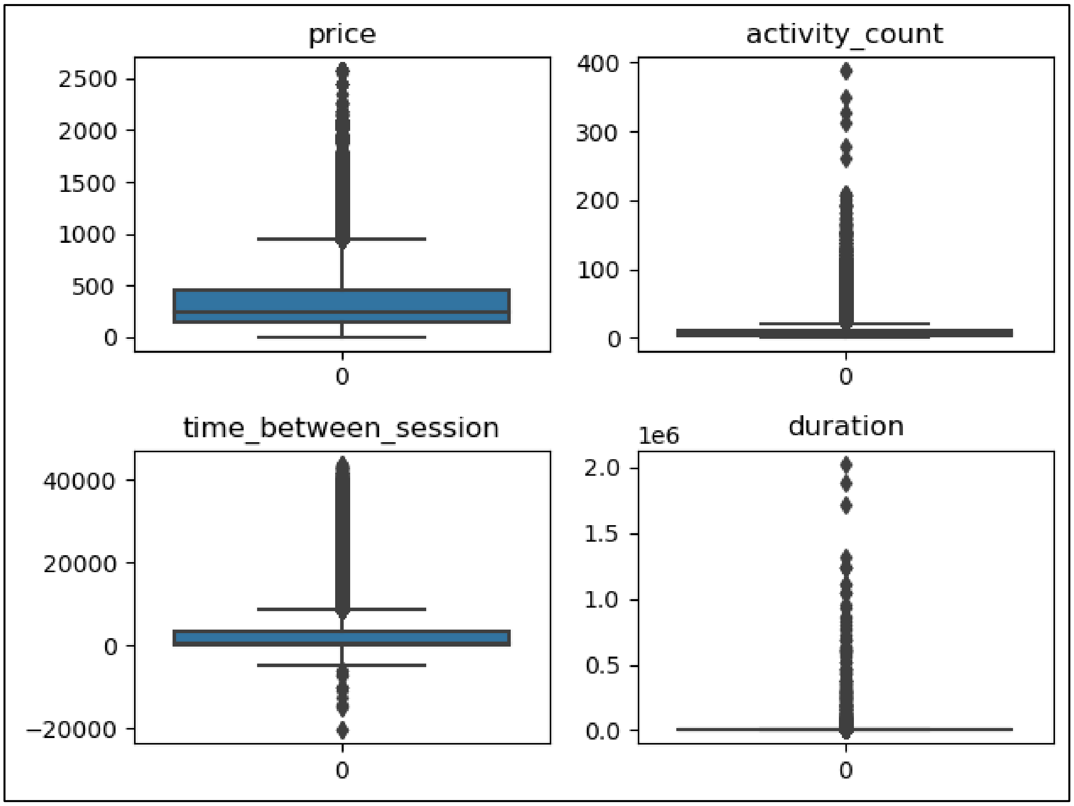
Distribution of Variables – Returning Users:



Outliers – First Time Users:

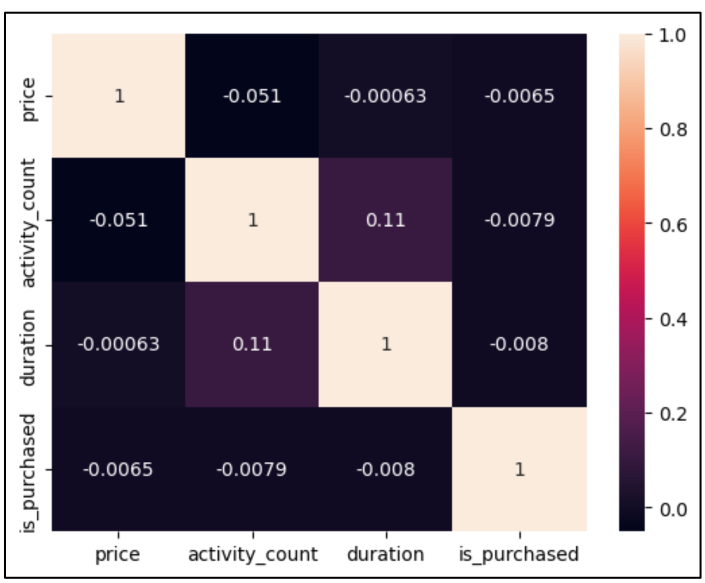


Outliers – Returning Users:

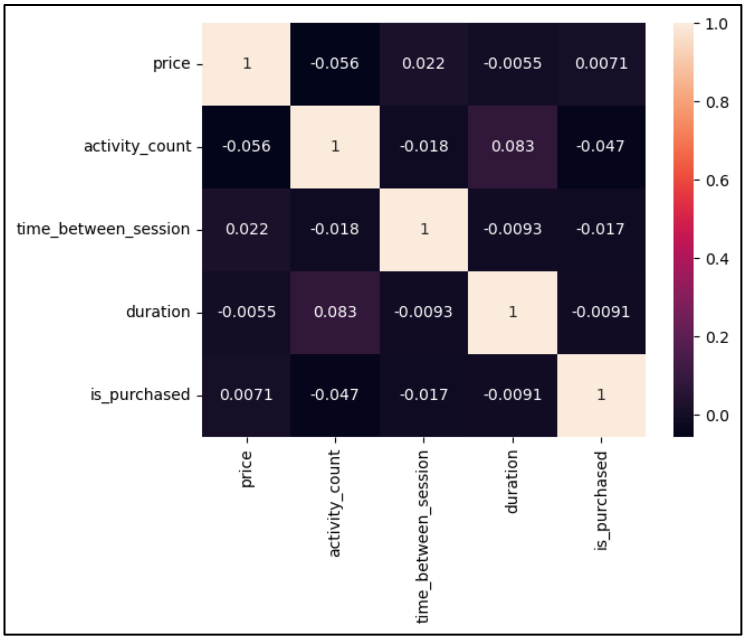


There are lot of outliers present in both the dataset. We are not treating the variables for outliers, since we think that those outliers have effect on purchase decision of the customers.

Checking for Relationship between Variables– First Time Users:

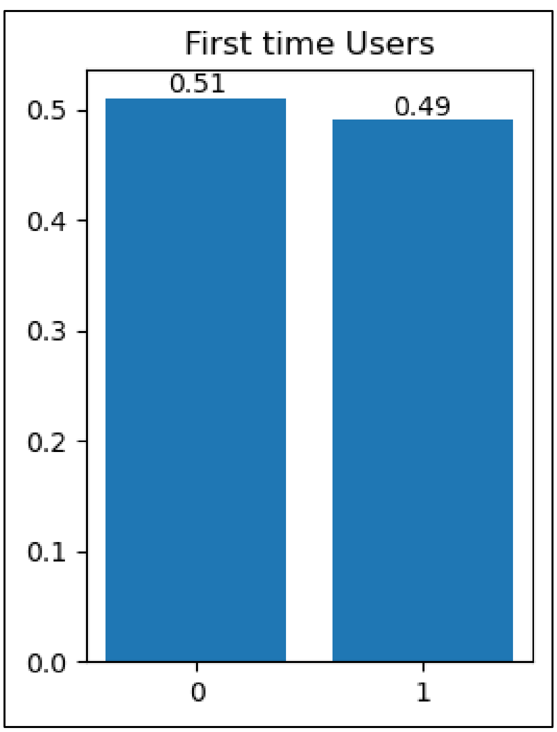


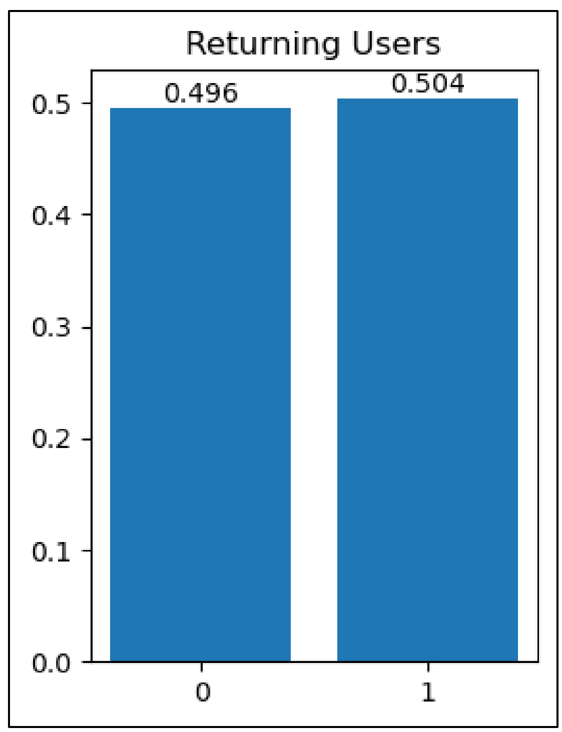
Checking for Relationship between Variables– Returning Users:



There is almost no relation among the independent variables hence multicollinearity is not present in both datasets.

Target Imbalance:





The datasets are balanced hence we can proceed withour any technique to treat the imbalanced datasets. Since Imabalanced datasets will affect the recall precision score after ML algorithm.

Statistical test for Variable Significance:

We perform statistical tests for variables in order find that the variables have a significant effect on the target variable.

i. For Categorical columns, we perform Chi square test of Independence with one categorical variable and target variable which is also a categorical variable.
ii. Day Variables has Relationship with the Purchase column after the Statistical Tests.
iii. For Numerical columns, we perform two sample tests. Since the data is not normal, we should perform nonparametric test like Mann-Whitney U test. One Sample is product being purchased and another sample is products being not purchased.
iv. The Variables Price, Duration, Activity count, Time between sessions has significant relation with the Purchase columns after the Statistical Tests.

Encoding the Categorical variables:

Before model building, we should ensure that all the columns are numeric in nature. Hence, it's imperative to change the categorical columns into numeric through any one of the encoding techniques.

One hot encoding introduces multicollinearity to my dataset which in turn reflects in the model summary. Label Encoding also adds ordinality to the data. Hence, we use frequency encoding which helps us when there is high cardinality. Brand and Category columns are frequency encoded and Day variable is N-1 dummy encoded.

We have also converted the purchase column into abandoned column by changing 0 to 1 and 1 to 0. Since our focus of prediction is whether the customer has abandoned the cart or purchased the product.

Transforming of Numerical variables:

All the numerical variables have high skewness due to presence of outliers. Hence it is necessary to perform transformation like Box-cox, Yeo-Johnson to make the data more closely approximate a normal distribution. Here, we are using Yeo-Johnson to reduce the skewness because this technique handles both zero and positive values.

```
price              1.920088
activity_count     4.493521
duration          70.044596
dtype: float64
```

Before Transfromation

```
price             0.008872
activity_count    0.061177
duration          0.011869
dtype: float64
```

After Transformation

BASE MODEL BUILDING

We are using Logistics Regression as our base model since this is a Binary classification Problem and due to the very high explainability of the model. It is easy to understand and interpret the prediction made by the model. It is also faster and easier to train models for large datasets than complex algorithms. The model will provide us with probability rather than classes which will be useful in cases where probability of that class may be needed.

Before building the model, we should split the data into training and test datasets. Training dataset is used for training the model for prediction while test data is used to measure the performance of the model. Here test data will act as unseen data.

Then we build the model from the stats model library because we will be able to see the summary of the model. Once we fit the model using train data, we can see the model's summary. From the model summary we can see which variable will increase the odds of Purchase the most and the significance of variables.

<u>The Base Model Summary – First Time Users</u>:

From the summary, we can see that Variable Saturaday is not a significant variable and should be discared for our model building.All the other variables are significant for our model.The scoefficient of the variables is in log odds. Therefore, we need to convert it into Odds to interpret the result and find the feature importance based on the coefficients.

```
Optimization terminated successfully.
         Current function value: 0.674908
         Iterations 4
                           Logit Regression Results
==============================================================================
Dep. Variable:            is_purchased   No. Observations:                94792
Model:                           Logit   Df Residuals:                    94779
Method:                            MLE   Df Model:                           12
Date:                 Wed, 24 Jan 2024   Pseudo R-squ.:                 0.02615
Time:                         10:21:28   Log-Likelihood:                -63976.
converged:                        True   LL-Null:                       -65694.
Covariance Type:             nonrobust   LLR p-value:                     0.000
=================================================================================
                    coef    std err          z      P>|z|      [0.025      0.975]
---------------------------------------------------------------------------------
const             0.7255      0.025     28.969      0.000       0.676       0.775
brand            -0.9038      0.047    -19.316      0.000      -0.995      -0.812
price             0.0372      0.007      5.301      0.000       0.023       0.051
category_1       -0.0790      0.033     -2.378      0.017      -0.144      -0.014
category_2       -0.5850      0.030    -19.539      0.000      -0.644      -0.526
activity_count   -0.0564      0.010     -5.586      0.000      -0.076      -0.037
duration         -0.2464      0.010    -23.949      0.000      -0.267      -0.226
Day_Monday       -0.1397      0.026     -5.419      0.000      -0.190      -0.089
Day_Saturday     -0.0320      0.024     -1.318      0.187      -0.080       0.016
Day_Sunday       -0.0501      0.025     -2.026      0.043      -0.098      -0.002
Day_Thursday     -0.1657      0.024     -6.854      0.000      -0.213      -0.118
Day_Tuesday      -0.1235      0.023     -5.308      0.000      -0.169      -0.078
Day_Wednesday    -0.1993      0.024     -8.463      0.000      -0.245      -0.153
=================================================================================
```

Change in Odds of Purchase due to each variables:

- The price of the product is the most significant variable for predicting the abandonment of the cart.
- One unit increase in the price will increase the odds of abandonment of the cart by 1.037867 units.
- Since Saturday is not a significant variable, we will drop that in the feature model building.
- If the day of adding to the cart is Sunday, then odds of abandonment will increase by 0.9511.
- One unit increase in the activity count will increase the odds of abandonment of the cart by 0.945123 units.
- Category_2 and Brand are the least significant variable for prediction of abandonment.

	Variable	Odds
0	const	2.065834
2	price	1.037862
8	Day_Saturday	0.968470
9	Day_Sunday	0.951177
5	activity_count	0.945134
3	category_1	0.924044
11	Day_Tuesday	0.883808
7	Day_Monday	0.869587
10	Day_Thursday	0.847296
12	Day_Wednesday	0.819336
6	duration	0.781607
4	category_2	0.557125
1	brand	0.405040

THE EVALUATION METRICS-First Time Users:

Selecting the best Threshold value using Youden Index.

Threshold value is based on the formula = Max (TPR- FPR). The threshold value is selected if the difference between the TPR and FPR is maximum.

fpr	tpr	threshold	diff	difference
0.262631	0.444924	0.5357	0.182293	0.182293

Hence Optimal Threshold is 0.5357

- Since our dataset is balanced, we can use accuracy for the main evaluation metrics.
- For our problem statement, we should focus on Sensitivity (Recall score) because that indicates that our customer will abandon the cart.
- We must decrease the false negative rate since the model will predict that those who will abandon the cart as they will buy the product.
- This will make the company miss those who will abandon their cart for any targeted marketing.

```
The Evaluation metrics for Test Data

Accuracy   : 0.5872347757593659
Precision  : 0.6411195577055978
Recall     : 0.4448760370210521
F1 score   : 0.5252668233162529
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.56      0.74      0.63     19773
         1.0       0.64      0.44      0.53     20853

    accuracy                           0.59     40626
   macro avg       0.60      0.59      0.58     40626
weighted avg       0.60      0.59      0.58     40626
```

```
The Evaluation metrics for Training Data:

Accuracy  : 0.5897544096548232
Precision : 0.6361185185185185
Recall    : 0.4465637740244613
F1 score  : 0.5247476352259673
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.56      0.74      0.64     46716
         1.0       0.64      0.45      0.52     48076

    accuracy                           0.59     94792
   macro avg       0.60      0.59      0.58     94792
weighted avg       0.60      0.59      0.58     94792
```

- The accuracy of the model is 0.59 which means 59 percent of the data are correctly predicted. This is not an acceptable level for a prediction model that can be used in the industry.
- The training and test data both show the same level of accuracy meaning that the model is underfit.

- This may be due to a lot of reasons like bias in the data, the need for more and better predictor variables or the model may not be able to learn the complex patterns.
- Since the model is not overfitted and has less variance, boosting algorithms may give better performance.
- We can also try different weak learners, since each algorithm uses different assumptions, underlying approach.
- Our Focus metrics – Recall score is very low for this model (0.45). We try to increase the recall score in further models.
- Recall score for class 0 is 0.74 which means 74 percent of those who have purchased are predicted correctly. This will be useful if our Problem Statement is focused on predicting the purchase event.

<u>The Base Model Summary – Returning Users</u>:

Similarly, we built Logistics Regression model for returning users. In this model, except for day Saturday and Tuesday all the other variables are significant.

```
Optimization terminated successfully.
         Current function value: 0.684408
         Iterations 4
                           Logit Regression Results
==============================================================================
Dep. Variable:           is_purchased   No. Observations:               287826
Model:                          Logit   Df Residuals:                   287812
Method:                           MLE   Df Model:                           13
Date:                Wed, 24 Jan 2024   Pseudo R-squ.:                 0.01255
Time:                        12:15:29   Log-Likelihood:            -1.9699e+05
converged:                       True   LL-Null:                   -1.9949e+05
Covariance Type:            nonrobust   LLR p-value:                     0.000
=========================================================================================
                            coef    std err          z      P>|z|      [0.025      0.975]
-----------------------------------------------------------------------------------------
const                     0.4717      0.014     34.516      0.000       0.445       0.498
brand                    -1.1705      0.033    -35.770      0.000      -1.235      -1.106
price                     0.0386      0.004      9.349      0.000       0.030       0.047
category_1                0.0506      0.019      2.705      0.007       0.014       0.087
category_2               -0.4819      0.018    -26.925      0.000      -0.517      -0.447
activity_count           -0.0626      0.006    -10.630      0.000      -0.074      -0.051
time_between_session      0.0403      0.004     10.040      0.000       0.032       0.048
duration                 -0.0755      0.006    -12.818      0.000      -0.087      -0.064
Day_Monday               -0.1047      0.014     -7.399      0.000      -0.132      -0.077
Day_Saturday             -0.0106      0.014     -0.760      0.447      -0.038       0.017
Day_Sunday               -0.0402      0.014     -2.903      0.004      -0.067      -0.013
Day_Thursday             -0.1496      0.014    -10.761      0.000      -0.177      -0.122
Day_Tuesday              -0.0087      0.014     -0.630      0.529      -0.036       0.018
Day_Wednesday            -0.1553      0.014    -11.173      0.000      -0.183      -0.128
=========================================================================================
```

	Variable	Odds
0	const	1.602650
3	category_1	1.051941
6	time_between_session	1.041147
2	price	1.039315
12	Day_Tuesday	0.991334
9	Day_Saturday	0.989463
10	Day_Sunday	0.960561
5	activity_count	0.939293
7	duration	0.927318
8	Day_Monday	0.900577
11	Day_Thursday	0.861016
13	Day_Wednesday	0.856154
4	category_2	0.617636
1	brand	0.310197

- Category_1, Time between session and price are the most Significant Variables.
- Category_2 and brand are the least significant variables.
- One unit increase in Time between session will increase the odd of abandonment by 1.041147.
- One unit increase in price will increase the odds of abandonment by 1.039315.

THE EVALUATION METRICS – Returning Users:

	fpr	tpr	threshold	difference
26160	0.372197	0.502289	0.500516	0.130093

The Optimal Threshold values for the Logistics Regression model is 0.500516.

```
The Evaluation metrics for Test Data

Accuracy   : 0.5654006728547688
Precision  : 0.5714881358502009
Recall     : 0.49308303626909117
F1 score   : 0.5293983391562352
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.56      0.64      0.60     62201
         1.0       0.57      0.49      0.53     61154

    accuracy                           0.57    123355
   macro avg       0.57      0.56      0.56    123355
weighted avg       0.57      0.57      0.56    123355
```

```
The Evaluation metrics for Training Data:

Accuracy  : 0.5614398977159811
Precision : 0.5669388522432832
Recall    : 0.48637523841579716
F1 score  : 0.5235760574596813
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.56      0.64      0.59    145218
         1.0       0.57      0.49      0.52    142608

    accuracy                           0.56    287826
   macro avg       0.56      0.56      0.56    287826
weighted avg       0.56      0.56      0.56    287826
```

- The overall accuracy is 0.57, which is lower than first time users and needs to be improved.
- Our Focus Metrics Recall is 0.49 which is better than the first-time users. Hence, we can predict with more accuracy for returning users.

ML Models:

Naïve Bayes:

The "naive" in Naive Bayes comes from the assumption of feature independence. The main assumption of Naive Bayes is that all features used to describe an observation are independent of each other given the class label.

Our Variables are independent of each other based on the correlation coefficients. Hence we use Gaussian Naïve Bayes model because of the presence of continuous variables.

For First Time Users:

```
The Evaluation metrics for Test Data

Accuracy   : 0.6199478166691281
Precision  : 0.6764226582360994
Recall     : 0.49762624082865775
F1 score   : 0.5734099574515114
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.59      0.75      0.66     19773
         1.0       0.68      0.50      0.57     20853

    accuracy                           0.62     40626
   macro avg       0.63      0.62      0.62     40626
weighted avg       0.63      0.62      0.61     40626
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.6234175879821082
Precision  : 0.671392321270962
Recall     : 0.5058077922077923
F1 score   : 0.5769545276780318
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.59      0.74      0.66     46667
         1.0       0.67      0.51      0.58     48125

    accuracy                           0.62     94792
   macro avg       0.63      0.63      0.62     94792
weighted avg       0.63      0.62      0.62     94792
```

For Returning Users:

```
The Evaluation metrics for Test Data

Accuracy   : 0.606007052815046
Precision  : 0.6277191054677167
Recall     : 0.5044314353926154
F1 score   : 0.5593624486613417
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.59      0.71      0.64     62201
         1.0       0.63      0.50      0.56     61154

    accuracy                           0.61    123355
   macro avg       0.61      0.61      0.60    123355
weighted avg       0.61      0.61      0.60    123355
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.6027634751551284
Precision  : 0.623634566778321
Recall     : 0.5000210366879838
F1 score   : 0.5550284297662181
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.59      0.70      0.64    145218
         1.0       0.62      0.50      0.56    142608

    accuracy                           0.60    287826
   macro avg       0.61      0.60      0.60    287826
weighted avg       0.61      0.60      0.60    287826
```

KNeighbors Classifier:

KNeighbors classification is a supervised machine learning algorithm used for classification tasks. It works by assigning a data point to the majority class among its k-nearest neighbors, determined based on a predefined distance metric. The algorithm is simple yet effective, making decisions based on the proximity of data points in the feature space.

Model built are sensitive to outliers and noise. Hence it is not preferred for our model. It is also hard to predict when the number of dimensions is very high. Interpretability is not possible in this model.

```
The Evaluation metrics for Test Data

Accuracy   : 0.5871609314232265
Precision  : 0.6025954198473282
Recall     : 0.5691693905018266
F1 score   : 0.5854056459188213
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.57      0.61      0.59     19822
         1.0       0.60      0.57      0.59     20804

    accuracy                           0.59     40626
   macro avg       0.59      0.59      0.59     40626
weighted avg       0.59      0.59      0.59     40626
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.7322031395054435
Precision  : 0.7489817369596636
Recall     : 0.7107116883116883
F1 score   : 0.7293450331055217
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.72      0.75      0.74     46667
         1.0       0.75      0.71      0.73     48125

    accuracy                           0.73     94792
   macro avg       0.73      0.73      0.73     94792
weighted avg       0.73      0.73      0.73     94792
```

For Returning Users:

```
The Evaluation metrics for Test Data

Accuracy   : 0.5826192695877751
Precision  : 0.5852406982895433
Recall     : 0.5427118422343592
F1 score   : 0.5631745062105477
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.58      0.62      0.60     62201
         1.0       0.59      0.54      0.56     61154

    accuracy                           0.58    123355
   macro avg       0.58      0.58      0.58    123355
weighted avg       0.58      0.58      0.58    123355
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.7308790727731338
Precision  : 0.7443624251699149
Recall     : 0.6957884550656345
F1 score   : 0.7192562792214853
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.72      0.77      0.74    145218
         1.0       0.74      0.70      0.72    142608

    accuracy                           0.73    287826
   macro avg       0.73      0.73      0.73    287826
weighted avg       0.73      0.73      0.73    287826
```

After Hyper Parameter Tuning:

```
The Evaluation metrics for Test Data

Accuracy   : 0.5698567419878895
Precision  : 0.5816451660371805
Recall     : 0.5699865410497981
F1 score   : 0.5757568400864266
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.56      0.57      0.56     19822
         1.0       0.58      0.57      0.58     20804

    accuracy                           0.57     40626
   macro avg       0.57      0.57      0.57     40626
weighted avg       0.57      0.57      0.57     40626
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.9966452865220694
Precision  : 0.996799334926738
Recall     : 0.9965922077922078
F1 score   : 0.9966957605985038
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       1.00      1.00      1.00     46667
         1.0       1.00      1.00      1.00     48125

    accuracy                           1.00     94792
   macro avg       1.00      1.00      1.00     94792
weighted avg       1.00      1.00      1.00     94792
```

For Returning Users:

```
The Evaluation metrics for Test Data

Accuracy   : 0.5699160958210044
Precision  : 0.5669958153459369
Recall     : 0.5605520489256631
F1 score   : 0.5637555195579421
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.57      0.58      0.58     62201
         1.0       0.57      0.56      0.56     61154

    accuracy                           0.57    123355
   macro avg       0.57      0.57      0.57    123355
weighted avg       0.57      0.57      0.57    123355
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.9992599695649454
Precision  : 0.9992216916636867
Recall     : 0.9992847526085493
F1 score   : 0.9992532211412043
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       1.00      1.00      1.00    145218
         1.0       1.00      1.00      1.00    142608

    accuracy                           1.00    287826
   macro avg       1.00      1.00      1.00    287826
weighted avg       1.00      1.00      1.00    287826
```

Decision Tree Classifier:

This Algorithm builds a tree-like model by recursively splitting the dataset based on the most significant features, leading to a set of decision rules. The final leaves of the tree represent the predicted classes or values.

```
The Evaluation metrics for Test Data

Accuracy   : 0.5693398316349136
Precision  : 0.5813344962930659
Recall     : 0.5753129046180405
F1 score   : 0.5783080260303688
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.56      0.56      0.56     19773
         1.0       0.58      0.58      0.58     20853

    accuracy                           0.57     40626
   macro avg       0.57      0.57      0.57     40626
weighted avg       0.57      0.57      0.57     40626
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.6890032914169972
Precision  : 0.7029599231642365
Recall     : 0.6698560612363758
F1 score   : 0.6860088616223585
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.68      0.71      0.69     46716
         1.0       0.70      0.67      0.69     48076

    accuracy                           0.69     94792
   macro avg       0.69      0.69      0.69     94792
weighted avg       0.69      0.69      0.69     94792
```

For Returning Users:

```
The Evaluation metrics for Test Data

Accuracy   : 0.5692837744720523
Precision  : 0.5654607463977416
Recall     : 0.566635052490434
F1 score   : 0.5660472903989873
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.57      0.57      0.57     62201
         1.0       0.57      0.57      0.57     61154

    accuracy                           0.57    123355
   macro avg       0.57      0.57      0.57    123355
weighted avg       0.57      0.57      0.57    123355
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.9992599695649454
Precision  : 1.0
Recall     : 0.9985063951531471
F1 score   : 0.9992526394459006
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       1.00      1.00      1.00    145218
         1.0       1.00      1.00      1.00    142608

    accuracy                           1.00    287826
   macro avg       1.00      1.00      1.00    287826
weighted avg       1.00      1.00      1.00    287826
```

After Hyper Parameter Tuning:

```
The Evaluation metrics for Test Data

Accuracy   : 0.6128095308423177
Precision  : 0.6272795031055901
Recall     : 0.6053805207883758
F1 score   : 0.6161354873346674
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.60      0.62      0.61     19773
         1.0       0.63      0.61      0.62     20853

    accuracy                           0.61     40626
   macro avg       0.61      0.61      0.61     40626
weighted avg       0.61      0.61      0.61     40626
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.6145877289222719
Precision  : 0.623096284288213
Recall     : 0.6076212663283135
F1 score   : 0.6152614840245161
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.61      0.62      0.61     46716
         1.0       0.62      0.61      0.62     48076

    accuracy                           0.61     94792
   macro avg       0.61      0.61      0.61     94792
weighted avg       0.61      0.61      0.61     94792
```

For Returning Users:

```
The Evaluation metrics for Test Data

Accuracy   : 0.6229419156094199
Precision  : 0.6602811104299852
Recall     : 0.4931647970696929
F1 score   : 0.5646166807076664
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.60      0.75      0.67     62201
         1.0       0.66      0.49      0.56     61154

    accuracy                           0.62    123355
   macro avg       0.63      0.62      0.62    123355
weighted avg       0.63      0.62      0.62    123355
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.6341122761668508
Precision  : 0.6746492591829472
Recall     : 0.50512593963873
F1 score   : 0.57770809440938
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.61      0.76      0.68    145218
         1.0       0.67      0.51      0.58    142608

    accuracy                           0.63    287826
   macro avg       0.64      0.63      0.63    287826
weighted avg       0.64      0.63      0.63    287826
```

Ensemble Model – Random Forest:

```
The Evaluation metrics for Test Data

Accuracy   : 0.5975237532614582
Precision  : 0.6130701225637935
Recall     : 0.5852874886107514
F1 score   : 0.5988567503250656
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.58      0.61      0.60     19773
         1.0       0.61      0.59      0.60     20853

    accuracy                           0.60     40626
   macro avg       0.60      0.60      0.60     40626
weighted avg       0.60      0.60      0.60     40626
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.9968879230314794
Precision  : 0.9970663504150802
Recall     : 0.9967967384973792
F1 score   : 0.9969315262276495
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       1.00      1.00      1.00     46716
         1.0       1.00      1.00      1.00     48076

    accuracy                           1.00     94792
   macro avg       1.00      1.00      1.00     94792
weighted avg       1.00      1.00      1.00     94792
```

For Returning Users:

```
The Evaluation metrics for Test Data

Accuracy   : 0.6119330387904828
Precision  : 0.6199090121317158
Recall     : 0.5615004742126435
F1 score   : 0.5892608926947299
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.61      0.66      0.63     62201
         1.0       0.62      0.56      0.59     61154

    accuracy                           0.61    123355
   macro avg       0.61      0.61      0.61    123355
weighted avg       0.61      0.61      0.61    123355
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.9992599695649454
Precision  : 0.9993547436859566
Recall     : 0.9991515202513183
F1 score   : 0.9992531216359791
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       1.00      1.00      1.00    145218
         1.0       1.00      1.00      1.00    142608

    accuracy                           1.00    287826
   macro avg       1.00      1.00      1.00    287826
weighted avg       1.00      1.00      1.00    287826
```

After Hyper Parameter Tuning:

```
The Evaluation metrics for Test Data

Accuracy   : 0.6162309850834441
Precision  : 0.6373173277661796
Recall     : 0.5855752169951566
F1 score   : 0.6103516357183916
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.60      0.65      0.62     19773
         1.0       0.64      0.59      0.61     20853

    accuracy                           0.62     40626
   macro avg       0.62      0.62      0.62     40626
weighted avg       0.62      0.62      0.62     40626
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.6191239767068951
Precision  : 0.6335504885993485
Recall     : 0.5906689408436642
F1 score   : 0.6113586944821202
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.61      0.65      0.63     46716
         1.0       0.63      0.59      0.61     48076

    accuracy                           0.62     94792
   macro avg       0.62      0.62      0.62     94792
weighted avg       0.62      0.62      0.62     94792
```

For Returning Users:

```
The Evaluation metrics for Test Data

Accuracy   : 0.6224393012038426
Precision  : 0.6589135457993635
Recall     : 0.49427674395787685
F1 score   : 0.5648428448630265
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.60      0.75      0.67     62201
         1.0       0.66      0.49      0.56     61154

    accuracy                           0.62    123355
   macro avg       0.63      0.62      0.62    123355
weighted avg       0.63      0.62      0.62    123355
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.6215942965541681
Precision  : 0.6574983872928022
Recall     : 0.49315606417592284
F1 score   : 0.5635911223659802
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.60      0.75      0.67    145218
         1.0       0.66      0.49      0.56    142608

    accuracy                           0.62    287826
   macro avg       0.63      0.62      0.61    287826
weighted avg       0.63      0.62      0.62    287826
```

Ensemble Model – AdaBoost Classifier:

```
The Evaluation metrics for Test Data

Accuracy   : 0.6289568256781372
Precision  : 0.6824524846877565
Recall     : 0.5182947297750923
F1 score   : 0.5891523575906241
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.59      0.75      0.66     19773
         1.0       0.68      0.52      0.59     20853

    accuracy                           0.63     40626
   macro avg       0.64      0.63      0.63     40626
weighted avg       0.64      0.63      0.62     40626
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.6320364587728923
Precision  : 0.6779203969366843
Recall     : 0.52292204010317
F1 score   : 0.5904180366369188
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.60      0.74      0.67     46716
         1.0       0.68      0.52      0.59     48076

    accuracy                           0.63     94792
   macro avg       0.64      0.63      0.63     94792
weighted avg       0.64      0.63      0.63     94792
```

For Returning Users:

```
The Evaluation metrics for Test Data

Accuracy   : 0.6246524259251753
Precision  : 0.6584658060386216
Recall     : 0.5046113091539393
F1 score   : 0.5713624454956999
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.60      0.74      0.67     62201
         1.0       0.66      0.50      0.57     61154

    accuracy                           0.62    123355
   macro avg       0.63      0.62      0.62    123355
weighted avg       0.63      0.62      0.62    123355
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.6206597041267988
Precision  : 0.652802413824632
Recall     : 0.5006521373274991
F1 score   : 0.5666923302828025
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.60      0.74      0.66    145218
         1.0       0.65      0.50      0.57    142608

    accuracy                           0.62    287826
   macro avg       0.63      0.62      0.61    287826
weighted avg       0.63      0.62      0.62    287826
```

After Hyper Parameter Tuning:

```
The Evaluation metrics for Test Data

Accuracy   : 0.6304090976222124
Precision  : 0.6788383776497978
Recall     : 0.5313384165347912
F1 score   : 0.5960995292535307
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.60      0.73      0.66     19773
         1.0       0.68      0.53      0.60     20853

    accuracy                           0.63     40626
   macro avg       0.64      0.63      0.63     40626
weighted avg       0.64      0.63      0.63     40626
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.6354122710777281
Precision  : 0.6764583006945533
Recall     : 0.5388759464181713
F1 score   : 0.5998795933961609
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.61      0.73      0.67     46716
         1.0       0.68      0.54      0.60     48076

    accuracy                           0.64     94792
   macro avg       0.64      0.64      0.63     94792
weighted avg       0.64      0.64      0.63     94792
```

For Returning Users:

```
The Evaluation metrics for Test Data

Accuracy   : 0.6258035750476267
Precision  : 0.6568284979187148
Recall     : 0.51347417993917
F1 score   : 0.5763713622305229
-----------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.61      0.74      0.66     62201
         1.0       0.66      0.51      0.58     61154

    accuracy                           0.63    123355
   macro avg       0.63      0.62      0.62    123355
weighted avg       0.63      0.63      0.62    123355
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.6241722429523393
Precision  : 0.6542385939137679
Recall     : 0.512117132278694
F1 score   : 0.5745190511215913
-----------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.61      0.73      0.66    145218
         1.0       0.65      0.51      0.57    142608

    accuracy                           0.62    287826
   macro avg       0.63      0.62      0.62    287826
weighted avg       0.63      0.62      0.62    287826
```

Ensemble Model – Gradient Boost Classifier:

```
The Evaluation metrics for Train Data

Accuracy   : 0.6354122710777281
Precision  : 0.6764583006945533
Recall     : 0.5388759464181713
F1 score   : 0.5998795933961609
-----------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.61      0.73      0.67     46716
         1.0       0.68      0.54      0.60     48076

    accuracy                           0.64     94792
   macro avg       0.64      0.64      0.63     94792
weighted avg       0.64      0.64      0.63     94792
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.636203477086674
Precision  : 0.6814795032714648
Recall     : 0.5307845910641484
F1 score   : 0.5967657066685375
-----------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.61      0.74      0.67     46716
         1.0       0.68      0.53      0.60     48076

    accuracy                           0.64     94792
   macro avg       0.64      0.64      0.63     94792
weighted avg       0.64      0.64      0.63     94792
```

For Returning Users:

```
The Evaluation metrics for Test Data

Accuracy   : 0.627854566089741
Precision  : 0.6597586019026864
Recall     : 0.5148477613892796
F1 score   : 0.5783643778243139
-----------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.61      0.74      0.67     62201
         1.0       0.66      0.51      0.58     61154

    accuracy                           0.63    123355
   macro avg       0.63      0.63      0.62    123355
weighted avg       0.63      0.63      0.62    123355
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.6270003404834865
Precision  : 0.6578676292759829
Recall     : 0.5150131829911365
F1 score   : 0.5777407187442231
-----------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.61      0.74      0.67    145218
         1.0       0.66      0.52      0.58    142608

    accuracy                           0.63    287826
   macro avg       0.63      0.63      0.62    287826
weighted avg       0.63      0.63      0.62    287826
```

After Hyper Parameter Tuning:

```
The Evaluation metrics for Test Data

Accuracy   : 0.6054250972283759
Precision  : 0.6334901743703294
Recall     : 0.5487939385220352
F1 score   : 0.5881083303355774
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.58      0.67      0.62     19773
         1.0       0.63      0.55      0.59     20853

    accuracy                           0.61     40626
   macro avg       0.61      0.61      0.60     40626
weighted avg       0.61      0.61      0.60     40626
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.7580703012912482
Precision  : 0.8013230747105774
Recall     : 0.6953989516598719
F1 score   : 0.7446128489815919
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.72      0.82      0.77     46716
         1.0       0.80      0.70      0.74     48076

    accuracy                           0.76     94792
   macro avg       0.76      0.76      0.76     94792
weighted avg       0.76      0.76      0.76     94792
```

For Returning Users:

```
The Evaluation metrics for Test Data

Accuracy   : 0.632986097037007
Precision  : 0.6614218048016913
Recall     : 0.5320338816757694
F1 score   : 0.5897140785717522
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.61      0.73      0.67     62201
         1.0       0.66      0.53      0.59     61154

    accuracy                           0.63    123355
   macro avg       0.64      0.63      0.63    123355
weighted avg       0.64      0.63      0.63    123355
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.6464009505743054
Precision  : 0.6781466777191222
Recall     : 0.544983451138786
F1 score   : 0.6043162670627067
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.63      0.75      0.68    145218
         1.0       0.68      0.54      0.60    142608

    accuracy                           0.65    287826
   macro avg       0.65      0.65      0.64    287826
weighted avg       0.65      0.65      0.64    287826
```

Ensemble Model – XG Boost Classifier:

```
The Evaluation metrics for Test Data

Accuracy   : 0.6290552847929897
Precision  : 0.6770992833956023
Recall     : 0.5301395482664365
F1 score   : 0.5946745562130178
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.60      0.73      0.66     19773
         1.0       0.68      0.53      0.59     20853

    accuracy                           0.63     40626
   macro avg       0.64      0.63      0.63     40626
weighted avg       0.64      0.63      0.63     40626
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.6733901595071314
Precision  : 0.724902764637864
Recall     : 0.5737582161577502
F1 score   : 0.6405350176481516
---------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.64      0.78      0.70     46716
         1.0       0.72      0.57      0.64     48076

    accuracy                           0.67     94792
   macro avg       0.68      0.67      0.67     94792
weighted avg       0.68      0.67      0.67     94792
```

For Returning Users:

```
The Evaluation metrics for Test Data

Accuracy   : 0.6322078553767582
Precision  : 0.660733560067613
Recall     : 0.530529482944697
F1 score   : 0.5885159218915806
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.61      0.73      0.67     62201
         1.0       0.66      0.53      0.59     61154

    accuracy                           0.63    123355
   macro avg       0.64      0.63      0.63    123355
weighted avg       0.64      0.63      0.63    123355
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.6503651511677194
Precision  : 0.683684007842039
Recall     : 0.5477602939526535
F1 score   : 0.6082206926622649
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.63      0.75      0.68    145218
         1.0       0.68      0.55      0.61    142608

    accuracy                           0.65    287826
   macro avg       0.66      0.65      0.65    287826
weighted avg       0.66      0.65      0.65    287826
```

After Hyper Parameter Tuning:

```
The Evaluation metrics for Test Data

Accuracy   : 0.6307537045241963
Precision  : 0.6754827875734677
Recall     : 0.5401141322591474
F1 score   : 0.6002611453087111
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.60      0.73      0.66     19773
         1.0       0.68      0.54      0.60     20853

    accuracy                           0.63     40626
   macro avg       0.64      0.63      0.63     40626
weighted avg       0.64      0.63      0.63     40626
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.6335977719638788
Precision  : 0.6704388698717622
Recall     : 0.545906481404443
F1 score   : 0.6017977115865264
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.61      0.72      0.66     46716
         1.0       0.67      0.55      0.60     48076

    accuracy                           0.63     94792
   macro avg       0.64      0.63      0.63     94792
weighted avg       0.64      0.63      0.63     94792
```

For Returning Users:

```
The Evaluation metrics for Test Data

Accuracy   : 0.6269466174861174
Precision  : 0.6545941087551579
Recall     : 0.5240049710566765
F1 score   : 0.5820648817524612
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.61      0.73      0.66     62201
         1.0       0.65      0.52      0.58     61154

    accuracy                           0.63    123355
   macro avg       0.63      0.63      0.62    123355
weighted avg       0.63      0.63      0.62    123355
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.6252492825526533
Precision  : 0.6516017557791488
Recall     : 0.5235961516885448
F1 score   : 0.5806276025365374
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.61      0.73      0.66    145218
         1.0       0.65      0.52      0.58    142608

    accuracy                           0.63    287826
   macro avg       0.63      0.62      0.62    287826
weighted avg       0.63      0.63      0.62    287826
```

Best Models:

For First Time Users: Decision Tree after Hyper Parameter

```
The Evaluation metrics for Test Data

Accuracy   : 0.6128095308423177
Precision  : 0.6272795031055901
Recall     : 0.6053805207883758
F1 score   : 0.6161354873346674
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.60      0.62      0.61     19773
         1.0       0.63      0.61      0.62     20853

    accuracy                           0.61     40626
   macro avg       0.61      0.61      0.61     40626
weighted avg       0.61      0.61      0.61     40626
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.6145877289222719
Precision  : 0.623096284288213
Recall     : 0.6076212663283135
F1 score   : 0.6152614840245161
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.61      0.62      0.61     46716
         1.0       0.62      0.61      0.62     48076

    accuracy                           0.61     94792
   macro avg       0.61      0.61      0.61     94792
weighted avg       0.61      0.61      0.61     94792
```

Recall Base Model: **0.44**

Recall Best Model: **0.61**

For Returning Users: XG Boost Classifier

```
The Evaluation metrics for Test Data

Accuracy   : 0.6322078553767582
Precision  : 0.660733560067613
Recall     : 0.530529482944697
F1 score   : 0.5885159218915806
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.61      0.73      0.67     62201
         1.0       0.66      0.53      0.59     61154

    accuracy                           0.63    123355
   macro avg       0.64      0.63      0.63    123355
weighted avg       0.64      0.63      0.63    123355
```

```
The Evaluation metrics for Train Data

Accuracy   : 0.6733901595071314
Precision  : 0.724902764637864
Recall     : 0.5737582161577502
F1 score   : 0.6405350176481516
--------------------------------------------------
The Classification report
              precision    recall  f1-score   support

         0.0       0.64      0.78      0.70     46716
         1.0       0.72      0.57      0.64     48076

    accuracy                           0.67     94792
   macro avg       0.68      0.67      0.67     94792
weighted avg       0.68      0.67      0.67     94792
```

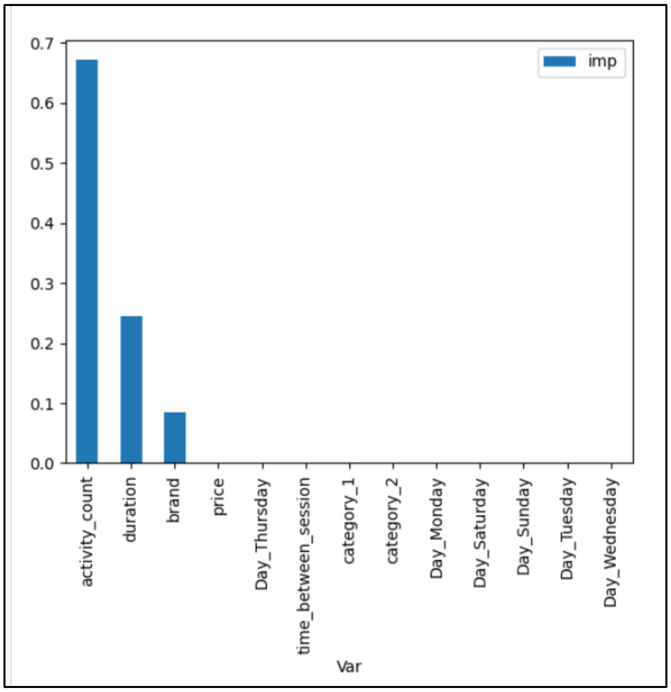
Recall Base Model: **0.49**

Recall Best Model: **0.53**

Importance Features for Prediction:

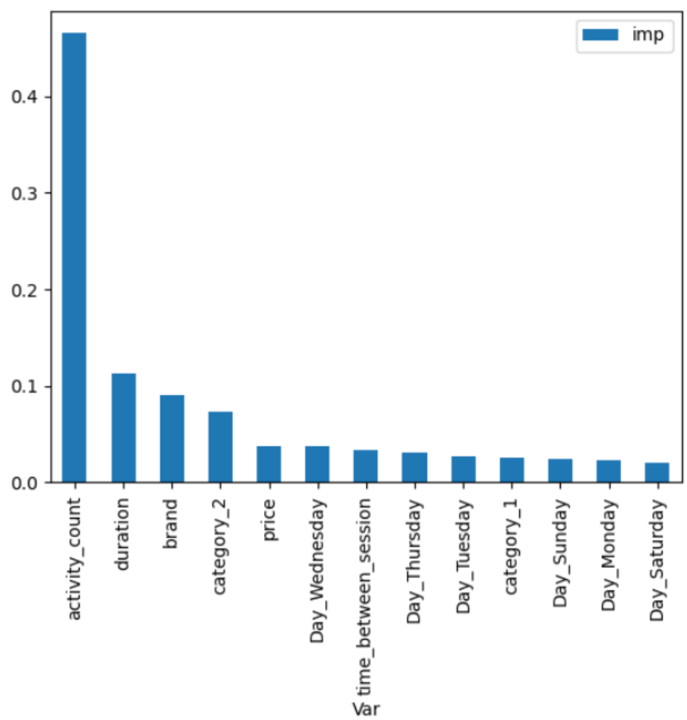
For First Time Users:

From the bar chart, activity count is the most important feature, then duration of the session, brand are the next important features.



For returning Users:

Similarly for returning users, activity count is the most important feature for prediction of cart abandonment. Duration, brand , category_2 are the other important features that needs attention.



Conclusion:

We have devised the problem statement for the business problem that needs to be addressed. Identified the approach to solve the problem, selected the data that can help us understand and finally solve the problem. The data was not usable directly for our prediction. Hence, we have cleaned the data, transformed it into a particular format, extracted new features that help us to understand the customer behavior in Ecommerce website. Using the Feature, we have developed a Classification Model that helps us to predict whether the cart will get abandoned or not. We have built a lot of models, learned to split the dataset according to each scenario and build separate models for each dataset.

Using Machine learning Algorithm's metrics, we have evaluated the performance of different models and choose the best model. Accurately Predicting cart abandonment will help the company with their marketing strategies and increase the potential revenue which will be lost otherwise.

The limitation of our models is that the metrics are not high enough, but it will help us to increase the revenue and will not incur any loss due to the wrong prediction. Hence error in the model is not a critical problem for this business. Due to the large dataset and limited machine capabilities further enhancement was not made. With better capable machines that help us to tune the hyperparameter better, we can build better performing Models.