



## INDIAN E-COMMERCE CUSTOMER RETENTION

**Submitted By:**

**Lakshmi Rajendra Thute**

## Machine Learning Project

# Indian E-commerce Customer Retention



A screenshot of a Microsoft Bing search results page. The main result is a circular diagram titled 'Customer Retention' divided into six segments: Overall Satisfaction (top), Initial overall score for your brand; Emotional commitment to brand (top-right); Future Intentions (bottom-right); Intention or propensity to repurchase, change spend /share (bottom); Advocacy (bottom-left); and Perceived Value for Money (top-left). To the right of the diagram is a sidebar with related content, including a thumbnail for 'Assignment on Sales including Customer Retention and Winback ...'. Below the main result are sections for 'Related content', 'Customer Retention Strategies', 'Traction', and 'Recruitment &amp; Retention Systems'. The bottom of the screen shows a Windows taskbar with various icons and the system tray.

## **ACKNOWLEDGEMENT**

I am very much Thankful to FlipRobo Technologies for giving me the opportunity to work with them and to work on this project and also I am very grateful to Data Trained Education Team for their support and help to understand each and every concept of machine learning which helped me a lot while working on this project. I thought, I am fortunate to become a part of FlipRobo Technology.

### **References:**

Google website

Stackoverflow

Analytics Vidya

Kaggle

Medium

Data trained notes

## **CONTENT**

- Introduction
- Problem Statement
- Importing Necessary Libraries
- Data Collection
- Data Pre-processing
- Exploratory Data Analysis (EDA)
- EDA Observations
- Checking Correlation
- Label and Ordinal Encoding
- Checking Skewness
- Data Scaling
- Checking VIF, Feature Importance
- Model Building and Model Evaluation
- Hyper Parameter tuning
- Conclusions

## INTRODUCTION

**Customer retention** refers to the ability of a company or product to retain its customers over some specified period. High customer retention means customers of the product or business tend to return to, continue to buy or in some other way not defect to another product or business, or to non-use entirely. Selling organizations generally attempt to reduce customer defections. Customer retention starts with the first contact an organization has with a customer and continues throughout the entire lifetime of a relationship and successful retention efforts take this entire lifecycle into account. A company's ability to attract and retain new customers is related not only to its product or services, but also to the way it services its existing customers, the value the customers actually perceive as a result of utilizing the solutions, and the reputation it creates within and across the marketplace.

Successful customer retention involves more than giving the customer what they expect. Generating loyal advocates of the brand might mean exceeding customer expectations. Creating customer loyalty puts 'customer value rather than maximizing profits and shareholder value at the centre of business strategy'. The key differentiation in a competitive environment is often the delivery of a consistently high standard of customer service. Furthermore, in the emerging world of Customer Success, retention is a major objective.

### What is Customer Retention?

In simple words, customer retention is a **brand's ability to keep customers long-term**. As a metric, it's usually measured as the percentage of customers who made a repeat purchase over a specific period of time.

### Why customer retention is Important?

Optimizing the customer experience and listening to their need's leads to lifetime brand loyalty, plus major financial benefits. On average, attracting a new customer costs five times as much as keeping an existing one, plus up to 30 times the marketing cost. However, increasing customer retention rates by just 5% can boost your profits by 25% to 95%! Here are five reasons why customer retention is important:

1. Lower Marketing Costs
2. Repeat Purchases Means Repeat Profit
3. Word-of-Mouth Advertising
4. Gain Valuable Feedback
5. Sell at Premium Prices

## **PROBLEM STATEMENT**

- 1 Gender of respondent,
- 2 How old are you?
- 3 Which cities do you shop online from?
- 4 What is the Pin Code of where you shop online from?
- 5 Since How Long You are Shopping Online?
- 6 How many times you have made an online purchase in the past 1 year?
- 7 How do you access the internet while shopping on-line?
- 8 Which devices do you use to access the online shopping?
- 9 What is the screen size of your mobile device?
- 10 What is the operating system (OS) of your device?
- 11 What browsers do you run on your device to access the website?
- 12 Which channels did you follow to arrive at your favourite online store for the first time?
- 13 After first visit, how do you reach the online retail store?
- 14 How much time do you explore the e- retail store before making a purchase decision?
- 15 What is your preferred payment Option?
- 16 How frequently do you abandon (selecting an item and leaving without making payment) your shopping cart?
- 17 Why did you abandon the "Bag", "Shopping Cart"?
- 18 The content on the website must be easy to read and understand',

- 19 Information on similar product to the one highlighted is important for product comparison',
- 20 Complete information on listed seller and product being offered is important for purchase decision';
- 21 All relevant information on listed products must be stated clearly',
- 22 Ease of navigation in website',
- 23 Loading and processing speed',
- 24 User friendly Interface of the website',
- 25 Convenient Payment methods',
- 26 Trust that the online retail store will fulfil its part of the transaction at the stipulated time',
- 27 Empathy (readiness to assist with queries) towards the customers',
- 28 Being able to guarantee the privacy of the customer',
- 29 Responsiveness, availability of several communication channels (email, online rep, twitter, phone etc.)',
- 30 Online shopping gives monetary benefit and discounts',
- 31 Enjoyment is derived from shopping online',
- 32 Shopping online is convenient and flexible',
- 33 Return and replacement policy of the e-tailer is important for purchase decision',
- 34 Gaining access to loyalty programs is a benefit of shopping online',
- 35 Displaying quality Information on the website improves satisfaction of customers',
- 36 User derive satisfaction while shopping on a good quality website or application',
- 37 Net Benefit derived from shopping online can lead to users satisfaction',

- 38 User satisfaction cannot exist without trust',
- 39 Offering a wide variety of listed product in several category',
- 40 Provision of complete and relevant product information',
- 41 Monetary savings',
- 42 The Convenience of patronizing the online retailer',
- 43 Shopping on the website gives you the sense of adventure',
- 44 Shopping on your preferred e-tailer enhances your social status',
- 45 You feel gratification shopping on your favorite e-tailer',
- 46 Shopping on the website helps you fulfill certain roles',
- 47 Getting value for money spent',
- 48 From the following, tick any (or all) of the online retailers you have shopped from;
- 49 Easy to use website or application',
- 50 Visual appealing web-page layout', 'Wild variety of product on offer',
- 51 Complete, relevant description information of products',
- 52 Fast loading website speed of website and application',
- 53 Reliability of the website or application',
- 54 Quickness to complete purchase',
- 55 Availability of several payment options', 'Speedy order delivery ',
- 56 Privacy of customers' information',
- 57 Security of customer financial information',
- 58 Perceived Trustworthiness',
- 59 Presence of online assistance through multi-channel',

- 60 Longer time to get logged in (promotion, sales period)',
  - 61 Longer time in displaying graphics and photos (promotion, sales period)',
  - 62 Late declarations of price (promotion, sales period)',
  - 63 Longer page loading time (promotion, sales period)',
  - 64 Limited mode of payment on most products (promotion, sales period)',
  - 65 Longer delivery period', 'Change in website/Application design',
  - 66 Frequent disruptions when moving from one page to another',
  - 67 Website is as efficient as before.
- 71 Which of the Indian online retailer would you recommend to a friend

# IMPORTING IMPORTANT LIBRARIES

The screenshot shows a Jupyter Notebook running on a Windows desktop. The notebook title is "Customer Retention Project". The code cell contains imports for pandas, numpy, matplotlib.pyplot, seaborn, and warnings, followed by a warning suppression command. A question is displayed in the text area: "Which of the Indian online retailer would you recommend to a friend?". The desktop taskbar at the bottom shows various icons and the date/time: 17-08-2022, 11:58. The system tray indicates the weather is 84°F and cloudy.

```
In [1]: # Target column.  
71 Which of the Indian online retailer would you recommend to a friend?  
  
Importing Important Libraries.  
  
In [1]: import pandas as pd  
import numpy as np  
import matplotlib.pyplot as plt  
import seaborn as sns  
import warnings  
warnings.filterwarnings('ignore')  
  
Data Collection and Preprocessing
```

# DATA COLLECTION

I have imported the excel file to get the dataset using pandas data frame.

The screenshot shows a Jupyter Notebook window titled "Customer Retention Project". The code cell In [2] contains:

```
# Collecting Data from excel file.  
data=pd.read_excel('C:/Users/91749/Downloads/E-commerce.xlsx')
```

The code cell In [3] contains:

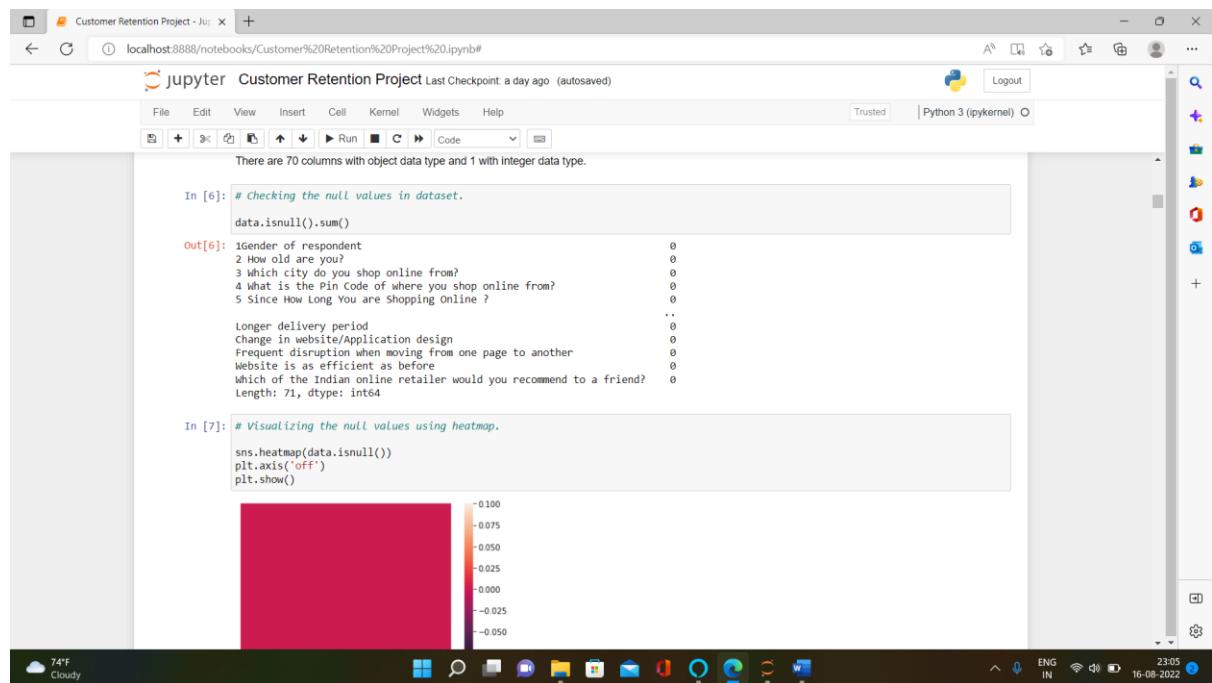
```
# checking the first five rows of dataset.  
data.head()
```

The output cell Out[3] displays the first five rows of the dataset as a table:

1Gender of respondent	2 How old are you?	3 Which city do you shop online from?	4 What is the Pin Code of where you shop online from?	5 Since How long you are shopping Online ?	6 How many times you have made an online purchase in the past 1 year?	7 How do you access the internet while shopping on-line?	8 Which device do you use to access the online shopping?	9 What is the screen size of your mobile device?	10 What is the operating system (OS) of your device?	... Longer time to get logged in (promotion, sales period)	Longer time in download graphics and photos (promotion, sales period)	Late declaration of price (promotion, sales period)	
0 Male	31-40 years	Delhi	110009	Above 4 years	31-40 times	Dial-up	Desktop	Others	Window/windows Mobile	...	Amazon.in	Amazon.in	Flipkart.com
1 Female	21-30 years	Delhi	110030	Above 4 years	41 times and above	Wi-Fi	Smartphone	4.7 inches	iOS/Mac	...	Amazon.in	Flipkart.com	Mynta.com snapdeal.com
2 Female	21-30 years	Greater Noida	201308	3-4 years	41 times and above	Mobile Internet	Smartphone	5.5 inches	Android	...	Mynta.com	Mynta.com	Mynta.com
3 Male	21-30 years	Karnal	132001	3-4 years	Less than 10 times	Mobile Internet	Smartphone	5.5 inches	iOS/Mac	...	Snapdeal.com	Snapdeal.com	Mynta.com
4 Female	21-30 years	Bangalore	530068	2-3 years	11-20 times	Wi-Fi	Smartphone	4.7 inches	iOS/Mac	...	Flipkart.com, Paytm.com	Paytm.com	Paytm.com

## DATA PRE-PROCESSING

In data pre-processing firstly I check the shape of dataset, there are 269 rows and 71 columns then I checked the dataset summary then checked the null values and lastly the statistical summary of dataset.



The screenshot shows a Jupyter Notebook interface running on a Windows desktop. The notebook title is "Customer Retention Project - Ju". The URL in the address bar is "localhost:8888/notebooks/Customer%20Retention%20Project%20.ipynb#". The notebook has two cells:

- In [6]:**

```
# Checking the null values in dataset.  
data.isnull().sum()
```

**Out[6]:** A table showing the count of null values for various columns:

Column	Count
Gender of respondent	0
How old are you?	0
Which city do you shop online from?	0
What is the Pin Code of where you shop online from?	0
Since How Long You are Shopping Online ?	0
Longer delivery period	0
Change in website/Application design	0
Frequent disruption when moving from one page to another	0
Website is as efficient as before	0
Which of the Indian online retailer would you recommend to a friend?	0
Length:	71
dtype:	int64
- In [7]:**

```
# Visualizing the null values using heatmap.  
sns.heatmap(data.isnull())  
plt.axis('off')  
plt.show()
```

A heatmap visualization showing the distribution of null values in the dataset. The color scale ranges from dark red (representing 0 null values) to light yellow (representing approximately -0.100 null values). The heatmap is mostly dark red, indicating that most values are non-null.

The desktop taskbar at the bottom shows various application icons, and the system tray indicates it's 23:05 on 16-08-2022.

The screenshot shows a Jupyter Notebook running on a Windows desktop. The notebook is titled "Customer Retention Project". The code cell In [9] contains the following Python code:

```
# checking the statistical summary of dataset.
data.describe()
```

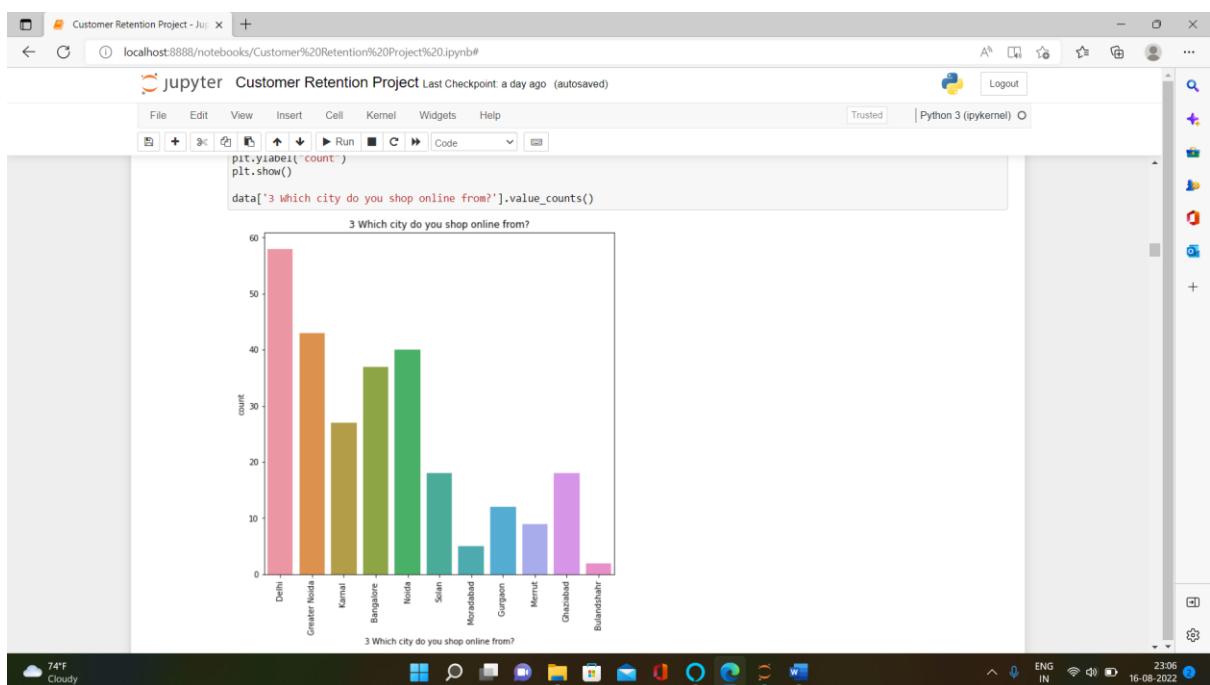
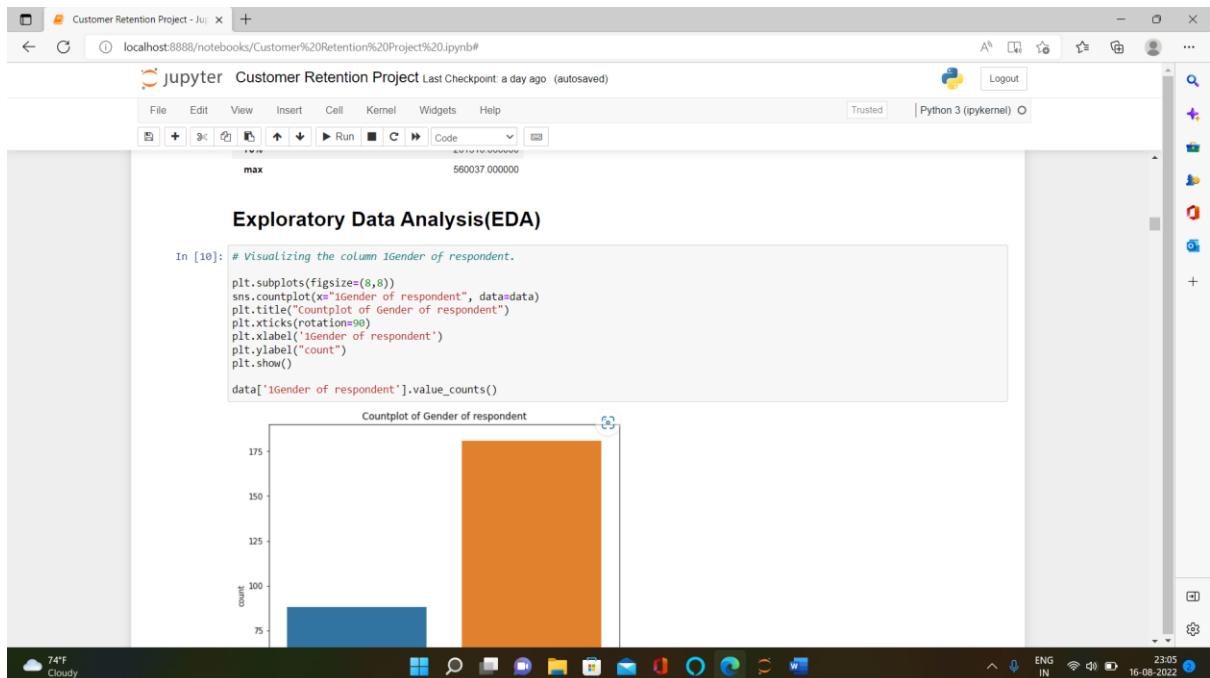
The output cell Out[9] displays the statistical summary of the dataset:

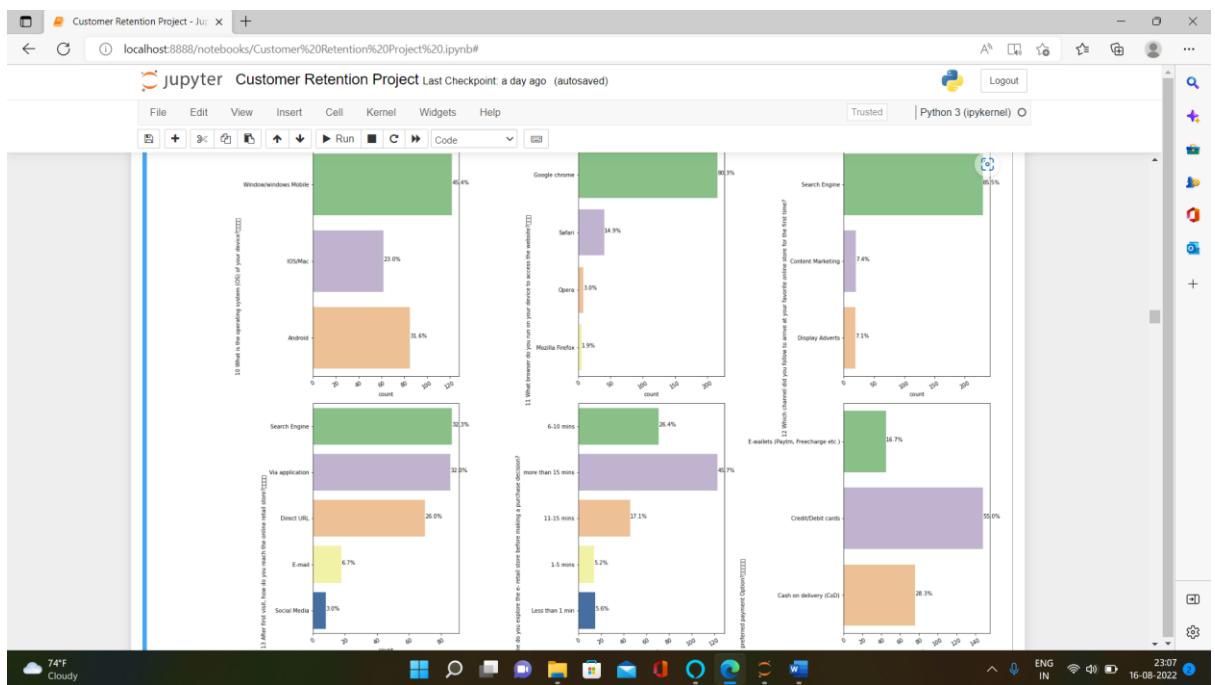
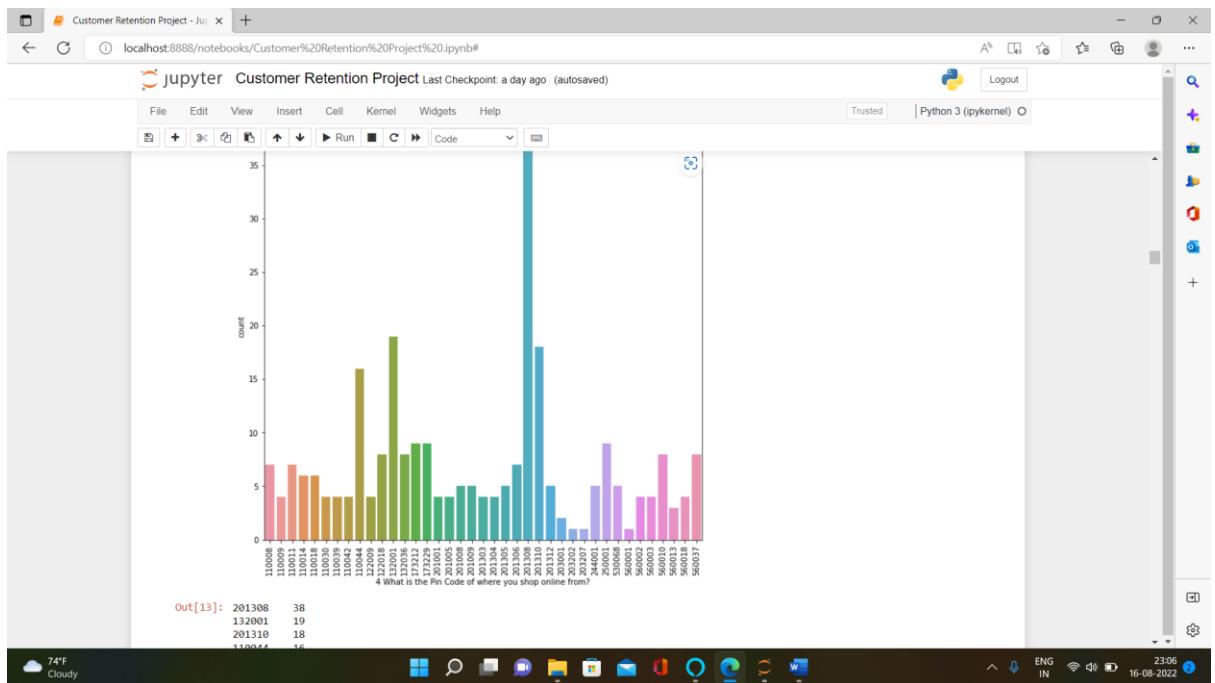
	4 What is the Pin Code of where you shop online from?
count	269.000000
mean	220465.747212
std	140524.341051
min	110008.000000
25%	122018.000000
50%	201303.000000
75%	201310.000000
max	560037.000000

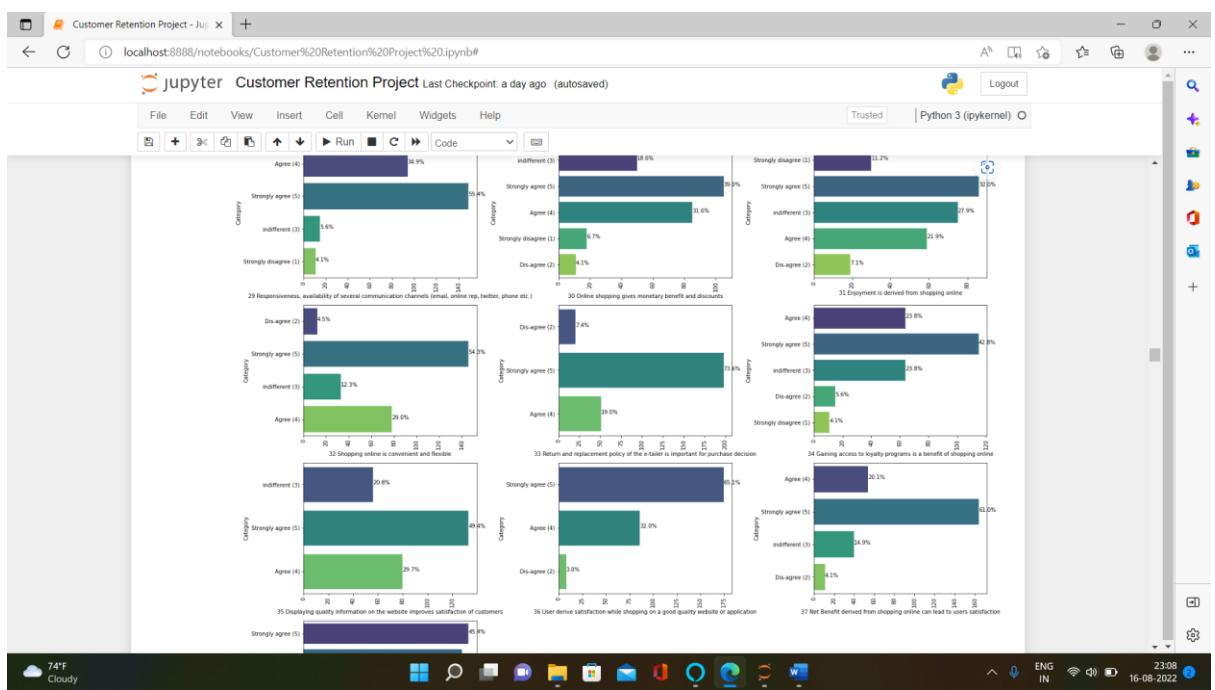
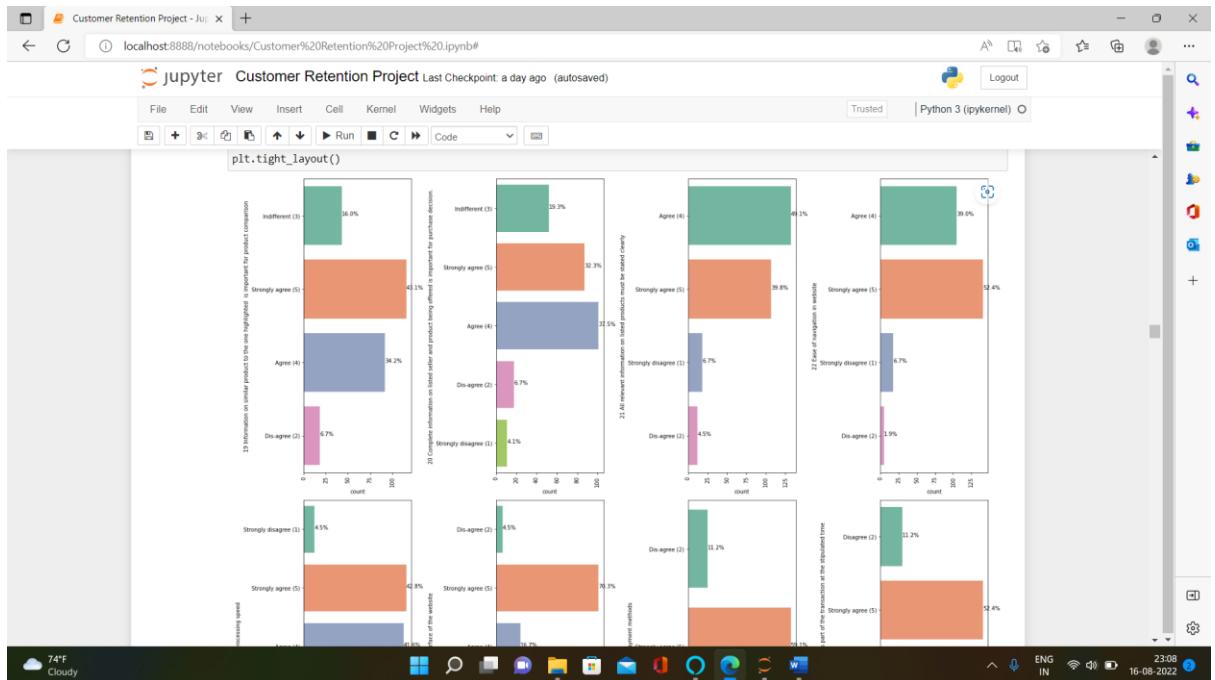
## EXPLORATORY DATA ANALYSIS (EDA)

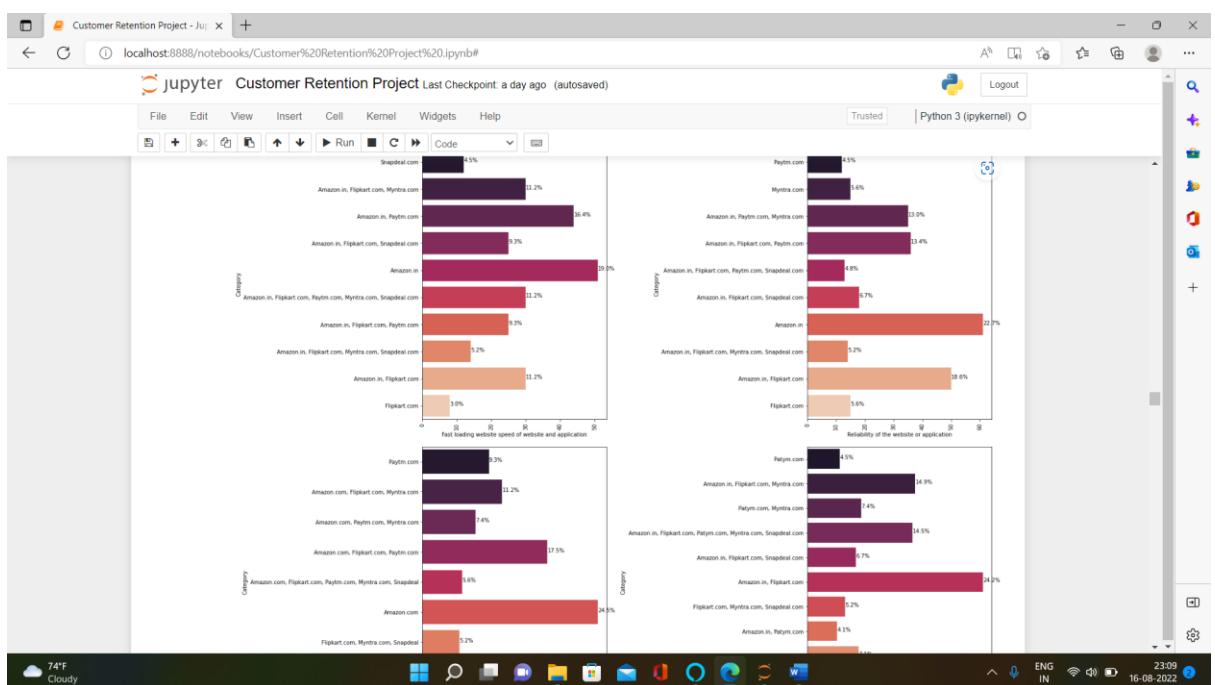
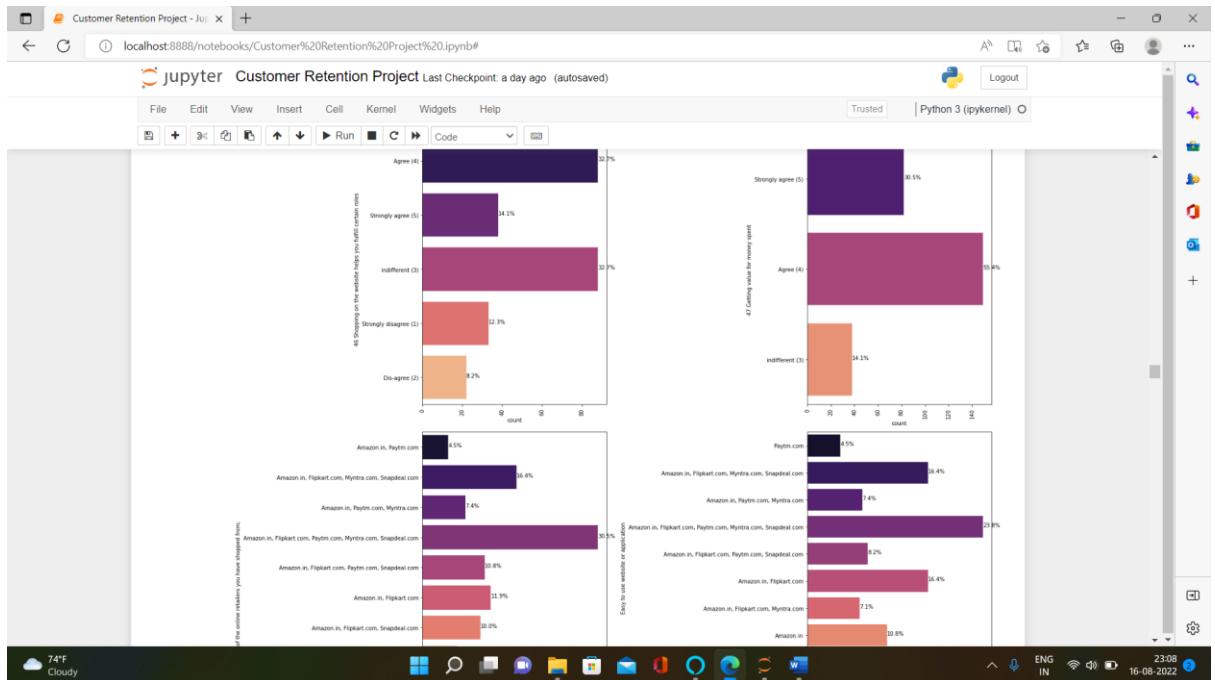
In Exploratory Data Analysis, I Univariate and Multivariate analysis, in which I used count plot for univariate analysis and pair plot for multivariate analysis.

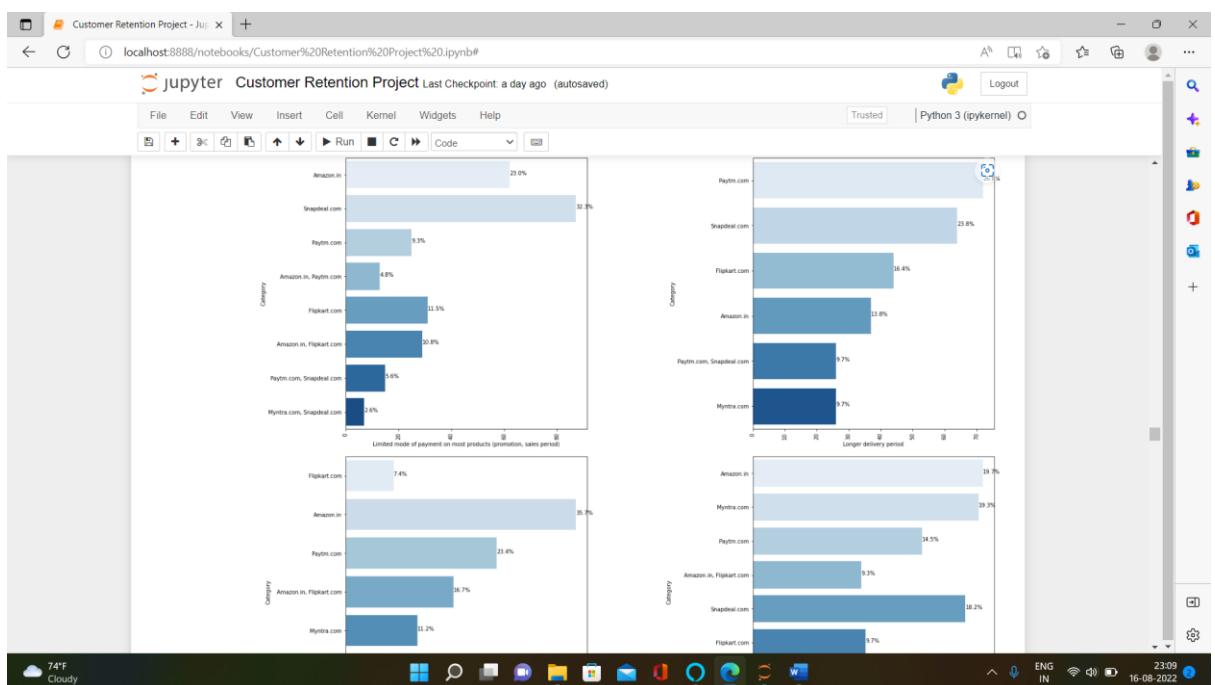
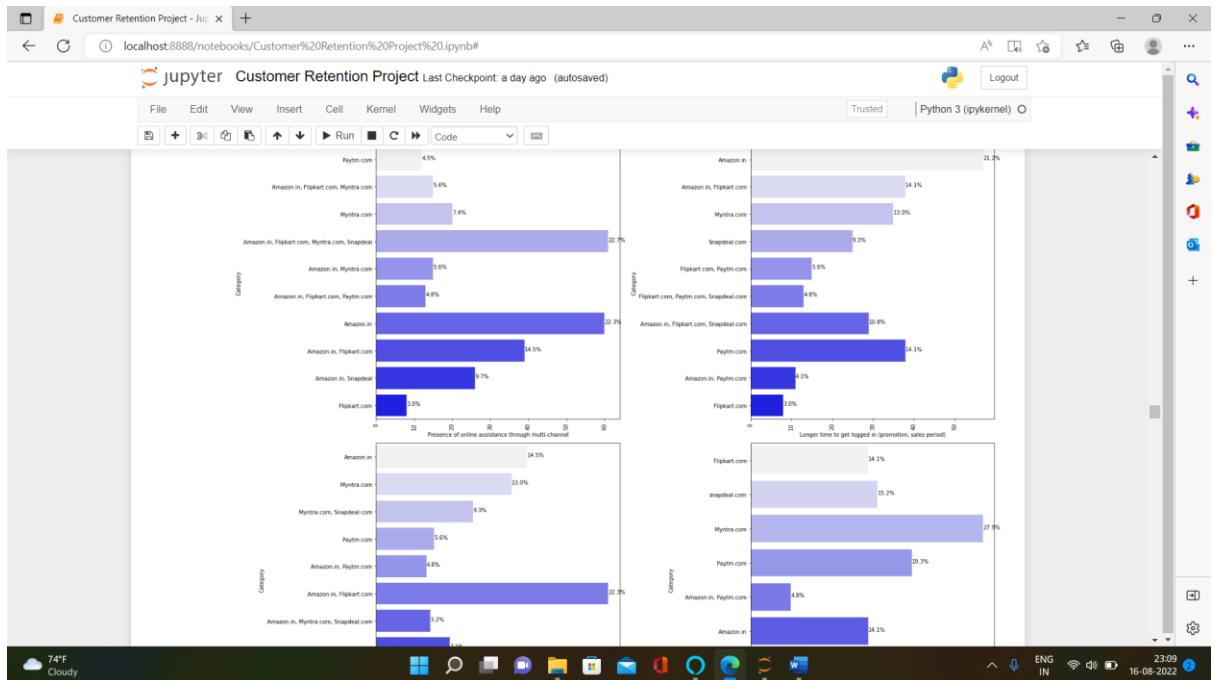
### Univariate Analysis



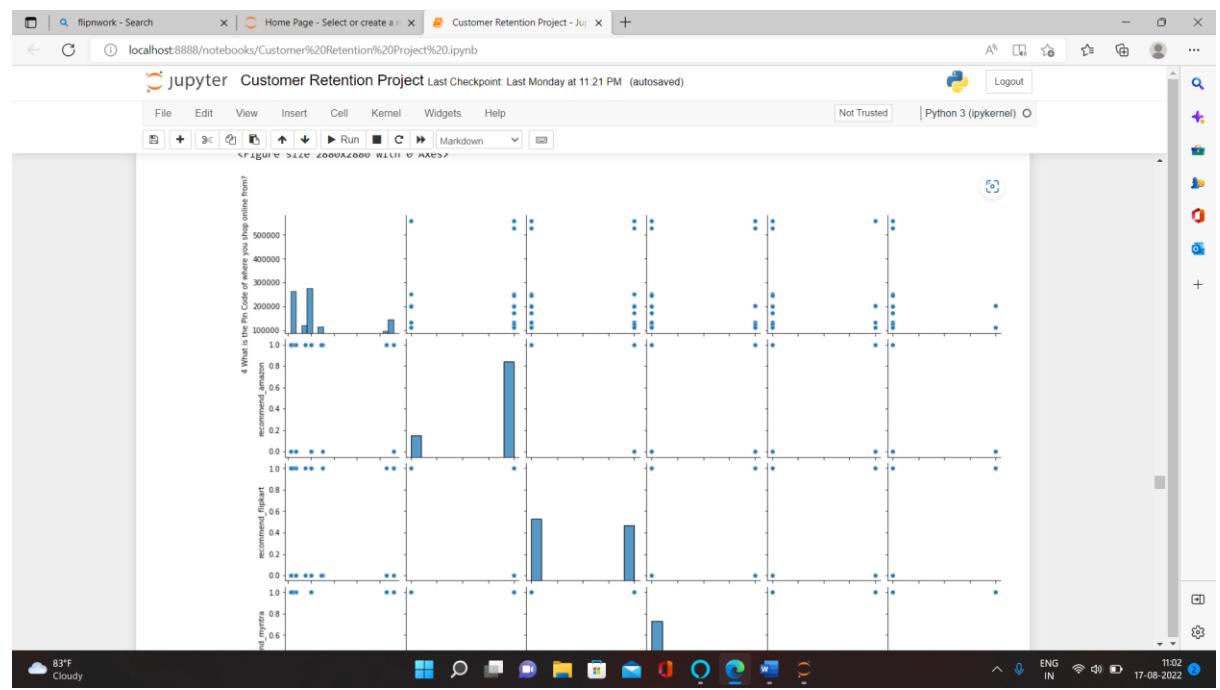








## Multivariate Analysis

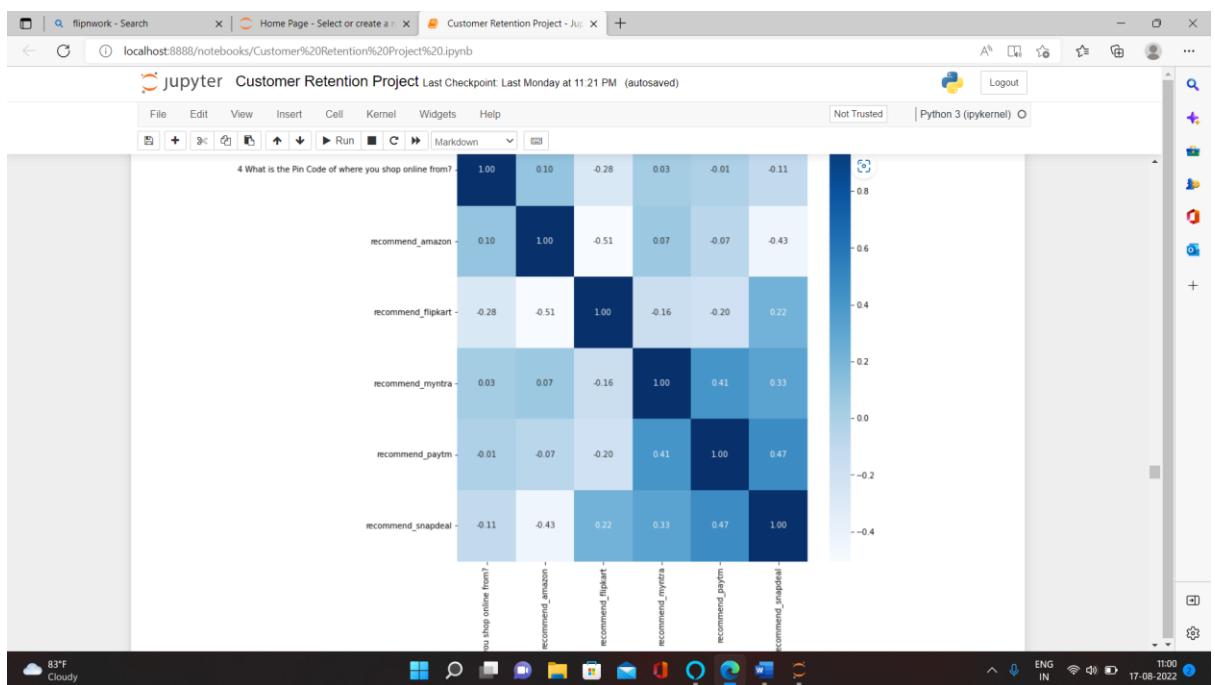


## **OBSERVATIONS OF EDA**

- In Gender of respondent column majority is of females and there are less no. of males.
- In How old are you? column most of people lie in age range 30-41years and less no. of people are in age range 51 and above.
- In Which city do you shop online from? column most of people shop from Delhi city and least no. of people shop from Bulandshar.
- In What is the Pin Code of where you shop online from? column most of people shop with pincode-201308 and least shop from pincode-203202,560001,203207.
- In Since How Long You are Shopping Online? column most of people have shopping duration of above 4 years and least of 1-2 years.
- In How many times you have made an online purchase in the past 1 year? column most of people shopped 10 times in a year and least shopped 42 times in a year.
- In How do you access the internet while shopping on-line? column most of people use mobile internet for shopping and very less use dial-up.
- In Which device do you use to access the online shopping? column most of people use smartphone to access online shopping and very least use Tablet.
- In column 10 What is the operating system (OS) of your device? Most of people use windows operating system and least use ios mac operating system.
- In column 11 What browser do you run on your device to access the website? most of people use google chrome browser for accessing website and least use Mozzarilla Firefox.
- In column 13 After first visit, how do you reach the online retail store? most of people reach online retail store via search engine and least via social media.

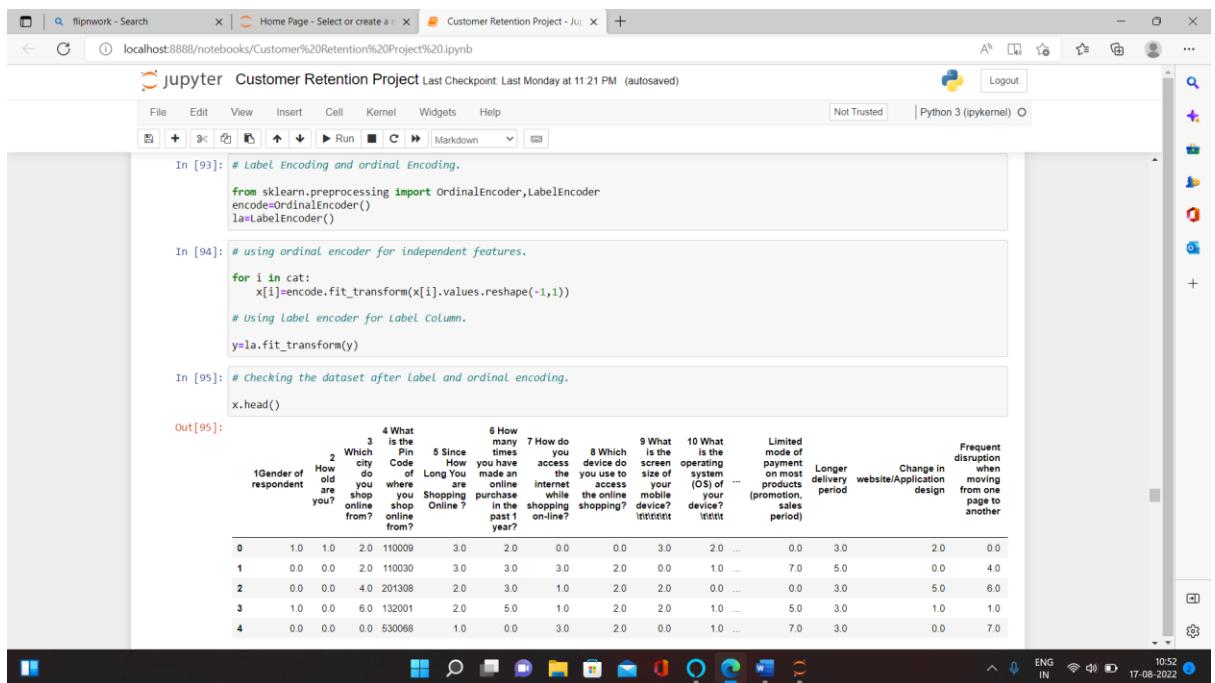
## CHECKING CORRELATION

I checked the correlation in dataset, some of the columns shows correlation with each other.



## LABEL AND ORDINAL ENCODING

Before Model building, I have encoded the categorical columns with label and ordinal encoder as we won't be able to fit model directly to the categorical columns.



The screenshot shows a Jupyter Notebook interface with several code cells and their outputs. The code is used for label and ordinal encoding of categorical variables in a dataset.

```
In [93]: # Label Encoding and ordinal Encoding.
from sklearn.preprocessing import OrdinalEncoder,LabelEncoder
encode=OrdinalEncoder()
la=LabelEncoder()

In [94]: # using ordinal encoder for independent features.
for i in cat:
    x[i]=encode.fit_transform(x[i].values.reshape(-1,1))

# Using label encoder for Label column.

y=la.fit_transform(y)

In [95]: # Checking the dataset after label and ordinal encoding.
x.head()

Out[95]:
```

1Gender of respondent	2 How old are you?	3 Where you shop from?	4 What city you are from?	5 Pin Code	6 Since Long time you shop online?	7 How many times you purchase in the past 1 year?	8 Which device you use to access the online shopping?	9 What is the screen size of your mobile device?	10 What is the operating system (OS) of your device?	Limited mode of payment products (promotion, delivery period)	Longer website/Application period	Change in sales (promotion, website/Applicaiton design)	Frequent disruption when moving from one page to another		
0	1.0	1.0	2.0	110000	3.0	2.0	0.0	0.0	3.0	2.0	...	0.0	3.0	2.0	0.0
1	0.0	0.0	2.0	110030	3.0	3.0	3.0	2.0	0.0	1.0	...	7.0	5.0	0.0	4.0
2	0.0	0.0	4.0	201308	2.0	3.0	1.0	2.0	2.0	0.0	...	0.0	3.0	5.0	6.0
3	1.0	0.0	6.0	132001	2.0	5.0	1.0	2.0	2.0	1.0	...	5.0	3.0	1.0	1.0
4	0.0	0.0	0.0	530068	1.0	0.0	3.0	2.0	0.0	1.0	...	7.0	3.0	0.0	7.0

## CHECKING SKEWNESS

The screenshot shows a Jupyter Notebook interface with the title "Customer Retention Project". The notebook contains the following code:

```
In [96]: # Checking the skewness in dataset.  
x.skew()  
Out[96]: 1Gender of respondent      0.741028  
2 How old are you?                0.680987  
3 Which city do you shop online from? 0.313729  
4 What is the Pin Code of where you shop online from? 1.748322  
5 Since How Long You are Shopping Online ? -0.276968  
...  
recommend_amazon          -1.624097  
recommend_flipkart         0.112325  
recommend_mynttra        0.971477  
recommend_paytm           1.829335  
recommend_snapdeal       4.662545  
Length: 75, dtype: float64  
  
In [ ]: Here we can see some skewness in dataset and we will handle it using powertransformation(method='yeo-johnson').  
  
In [97]: # Skewness removal using Powertransform(method='yeo-johnson')  
from sklearn.preprocessing import PowerTransformer  
scaler = PowerTransformer(method='yeo-johnson')  
  
In [98]: scaler.fit_transform(x)  
Out[98]: array([[ 1.43416114e+00, -6.80588196e-02, -6.05625000e-01, ...,  
   -6.27520823e-01, -4.42216639e-01, -2.06484040e-01],
```

The system tray at the bottom shows the date as 17-08-2022 and the time as 11:07.

I have checked skewness in dataset, there is some skewness in data and I removed it using Power transformation(method='yeo-johnson')

# DATA SCALING

The screenshot shows a Jupyter Notebook interface with the title "Customer Retention Project". The notebook contains the following code:

```
In [100]: # We will scale our dataset using MinMaxScaler.  
from sklearn.preprocessing import MinMaxScaler  
scaler=MinMaxScaler()  
  
In [101]: # Data Scaling.  
x_scaled=scaler.fit_transform(x)  
x=pd.DataFrame(x_scaled,columns=x.columns)  
x
```

Out[101]:

	1Gender of respondent	2 How old are you?	3 Which platform do you shop online from?	4 What is the Code of where you shop Online?	5 Since how long are you shopping Online?	6 How many times have you made an online purchase in the past 1 year?	7 How do you access internet while shopping on-line?	8 Which device do you use to access the online shopping?	9 What is the screen size of your mobile device?	10 What is the operating system (OS) of your device?	Limited mode of payment on most products (promotion, sales period)	Longer delivery period	Change in website/Application design	Frequent disrupt will move from page to another	
0	1.0	0.25	0.2	0.000002	0.75	0.4	0.000000	0.000000	1.000000	1.0	...	0.000000	0.6	0.333333	0.0000
1	0.0	0.00	0.2	0.000049	0.75	0.6	1.000000	0.666667	0.000000	0.5	...	1.000000	1.0	0.000000	0.571
2	0.0	0.00	0.4	0.202876	0.50	0.6	0.333333	0.666667	0.666667	0.0	...	0.000000	0.6	0.833333	0.857
3	1.0	0.00	0.6	0.048870	0.50	1.0	0.333333	0.666667	0.666667	0.5	...	0.714286	0.6	0.166667	0.142
4	0.0	0.00	0.0	0.933407	0.25	0.0	1.000000	0.666667	0.000000	0.5	...	1.000000	0.6	0.000000	1.000
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
264	0.0	0.00	1.0	0.140444	0.00	1.0	0.333333	0.666667	0.666667	0.0	...	0.000000	0.0	0.000000	0.000

Our Dataset needs Standardization so I used MinMaxScaler for standardizing the data.

## CHECKING VIF, FEATURE IMPORTANCE

The screenshot shows a Jupyter Notebook interface with the title "Customer Retention Project". The notebook contains two code cells:

**Checking VIF**

```
In [53]: # Checking the VIF factor again before model prediction.  
from statsmodels.stats.outliers_influence import variance_inflation_factor  
vif = pd.DataFrame()  
vif["VIF values"] = [variance_inflation_factor(x.values,i)  
                     for i in range(len(x.columns))]  
vif["Features"] = x.columns  
  
# Let's check the values  
vif.head()
```

**Out[53]:**

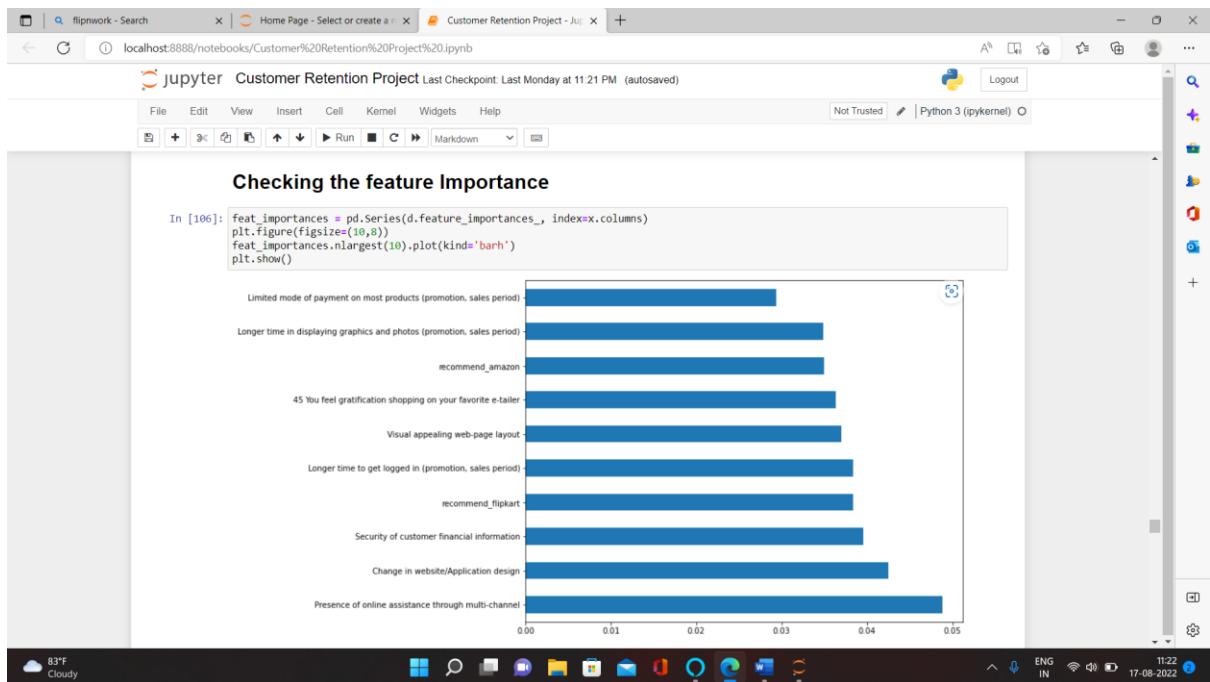
	VIF values	Features
0	2.206812	1Gender of respondent
1	1.882311	2 How old are you?
2	1.976970	3 Which city do you shop online from?
3	1.981758	4 What is the Pin Code of where you shop online?
4	1.715616	5 Since How Long You are Shopping Online ?

Here after checking the VIF factor we can see there is no multicollinearity in dataset.

**Checking the feature Importance**

```
In [106]: feat_importances = pd.Series(d.feature_importances_, index=x.columns)  
plt.figure(figsize=(10,8))  
feat_importances.nlargest(10).plot(kind='barh')  
plt.show()
```

As I want to check the Multicollinearity in data, so I used Variance Inflation factor, as VIF factor is less than 10 so we can say that there is no multicollinearity in dataset.



In the above chart we can see that above features are of most importance in determining which platform will a customer recommend to his friend

## MODEL BUILDING AND EVALUATION

In [67]:

```
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
```

### Logistic Regression

In [69]:

```
# Logistic Regression
lr=LogisticRegression()
lr.fit(x_train,y_train)
pred_lr=lr.predict(x_test)
print("Accuracy:",accuracy_score(y_test,pred_lr)*100)

# Checking the confusion matrix and classification report.

print(confusion_matrix(y_test,pred_lr))
print(classification_report(y_test,pred_lr))
```

Accuracy: 100.0

[15 0 0 0 0 0 0]	[ 0 14 0 0 0 0 0]	[ 0 0 2 0 0 0 0]	[ 0 0 0 13 0 0 0]	[ 0 0 0 0 1 0 0]	[ 0 0 0 0 0 9 0]	[ 0 0 0 0 0 0 12]	[ 0 0 0 0 0 0 2]]
precision	recall	f1-score	support				
0	1.00	1.00	1.00	15			

In [70]:

```
# Applying RandomForest Classifier for model building and evaluation.

from sklearn.ensemble import RandomForestClassifier
rf=RandomForestClassifier()
rf.fit(x_train,y_train)
pred_rf = rf.predict(x_test)
print("Accuracy:",accuracy_score(y_test,pred_rf)*100)

# checking the confusion matrix and classification report.

print(confusion_matrix(y_test,pred_rf))
print(classification_report(y_test,pred_rf))
```

Accuracy: 100.0

[15 0 0 0 0 0 0]	[ 0 14 0 0 0 0 0]	[ 0 0 2 0 0 0 0]	[ 0 0 0 13 0 0 0]	[ 0 0 0 0 1 0 0]	[ 0 0 0 0 0 9 0]	[ 0 0 0 0 0 0 12]	[ 0 0 0 0 0 0 2]]
precision	recall	f1-score	support				
0	1.00	1.00	1.00	15			
1	1.00	1.00	1.00	14			
2	1.00	1.00	1.00	2			
3	1.00	1.00	1.00	13			
4	1.00	1.00	1.00	1			
5	1.00	1.00	1.00	9			
6	1.00	1.00	1.00	12			
7	1.00	1.00	1.00	2			

```
In [71]: # Applying DecisionTree Classifier for model building and evaluation.  
from sklearn.tree import DecisionTreeClassifier  
dt=DecisionTreeClassifier()  
dt.fit(x_train,y_train)  
pred_dt = dt.predict(x_test)  
print("Accuracy:",accuracy_score(y_test,pred_dt)*100)  
  
# checking the confusion matrix and classification report.  
print(confusion_matrix(y_test,pred_dt))  
print(classification_report(y_test,pred_dt))
```

[15 0 0 0 0 0 0]	[ 0 14 0 0 0 0 0]	[ 0 0 2 0 0 0 0]	[ 0 0 0 13 0 0 0]	[ 0 0 0 0 1 0 0]	[ 0 0 0 0 0 9 0]	[ 0 0 0 0 0 0 12]	[ 0 0 0 0 0 0 2]
precision	recall	f1-score	support				
0 1.00	1.00	1.00	15				
1 1.00	1.00	1.00	14				
2 1.00	1.00	1.00	2				
3 1.00	1.00	1.00	13				
4 1.00	1.00	1.00	1				
5 1.00	1.00	1.00	9				

```
In [72]: # Applying Support Vector Classifier for model building and evaluation.  
from sklearn.svm import SVC  
svc=SVC()  
svc.fit(x_train,y_train)  
pred_svc=svc.predict(x_test)  
print("Accuracy:",accuracy_score(y_test,pred_svc)*100)  
  
# checking the confusion matrix and classification report.  
print(confusion_matrix(y_test,pred_svc))  
print(classification_report(y_test,pred_svc))
```

[15 0 0 0 0 0 0]	[ 0 14 0 0 0 0 0]	[ 0 0 2 0 0 0 0]	[ 0 0 0 13 0 0 0]	[ 0 0 0 0 1 0 0]	[ 0 0 0 0 0 9 0]	[ 0 0 0 0 0 0 12]	[ 0 0 0 0 0 0 2]
precision	recall	f1-score	support				
0 1.00	1.00	1.00	15				
1 1.00	1.00	1.00	14				
2 1.00	1.00	1.00	2				
3 1.00	1.00	1.00	13				
4 1.00	1.00	1.00	1				
5 1.00	1.00	1.00	9				
6 1.00	1.00	1.00	12				
7 1.00	1.00	1.00	2				

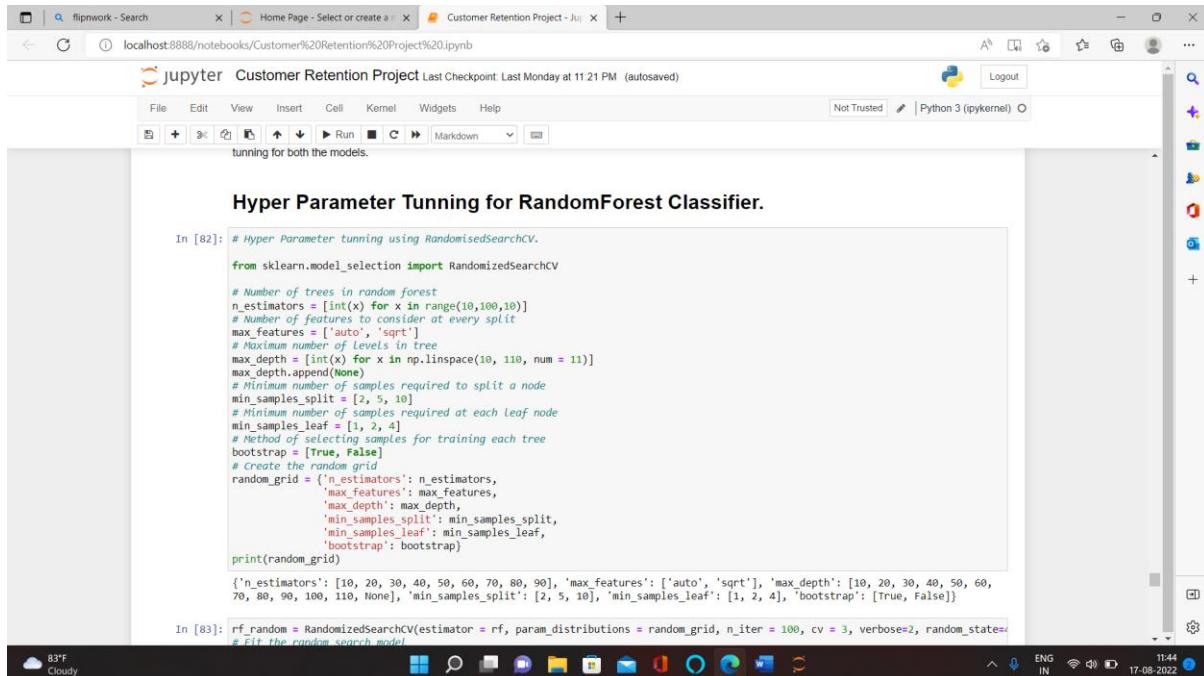
The screenshot shows a Jupyter Notebook interface running on a Windows desktop. The notebook title is "Customer Retention Project". The code in cell [80] applies an XGB Classifier for model building and evaluation. It includes importing XGBoost, fitting the classifier, predicting on test data, printing accuracy, and checking the confusion matrix and classification report. The output shows an accuracy of 98.52941176470588 and a detailed confusion matrix and classification report.

```
In [80]: # Applying XGB Classifier for model building and evaluation.  
from xgboost import XGBClassifier  
xgb=XGBClassifier()  
xgb.fit(x_train,y_train)  
pred_xgb=xgb.predict(x_test)  
print("Accuracy:",accuracy_score(y_test,pred_xgb)*100)  
  
# Checking the confusion matrix and classification report.  
print(confusion_matrix(y_test,pred_xgb))  
print(classification_report(y_test,pred_xgb))  
  
Accuracy: 98.52941176470588  
[[15  0  0  0  0  0]  
 [ 0 14  0  0  0  0]  
 [ 0  0 12  0  0  0]  
 [ 0  0  0 12  1  0]  
 [ 0  0  0  0  1  0]  
 [ 0  0  0  0  0  9]  
 [ 0  0  0  0  0 12]  
 [ 0  0  0  0  0  2]]  
 precision    recall   f1-score   support  
 0       1.00      1.00      1.00      15  
 1       1.00      1.00      1.00      14  
 2       1.00      1.00      1.00       2  
 3       1.00      0.92      0.96      13  
 4       0.50      1.00      0.67       1  
 5       1.00      1.00      1.00       9
```

In model building and evaluation, I used 5 models that are Logistic Regression, Random Forest Classifier, Support Vector Classifier, XBG Classifier. After model training and prediction, the accuracy score of each and every model is very well good but for selecting the best fit model I checked the cross-validation score for each model and 2 models Random Forest and XGboost classifier shows very less or negligible difference in accuracy score and cross validation score.

## HYPER PARAMETER TUNNING

### Hyper parameter tunning for Random Forest Classifier using RandomisedSearchCV



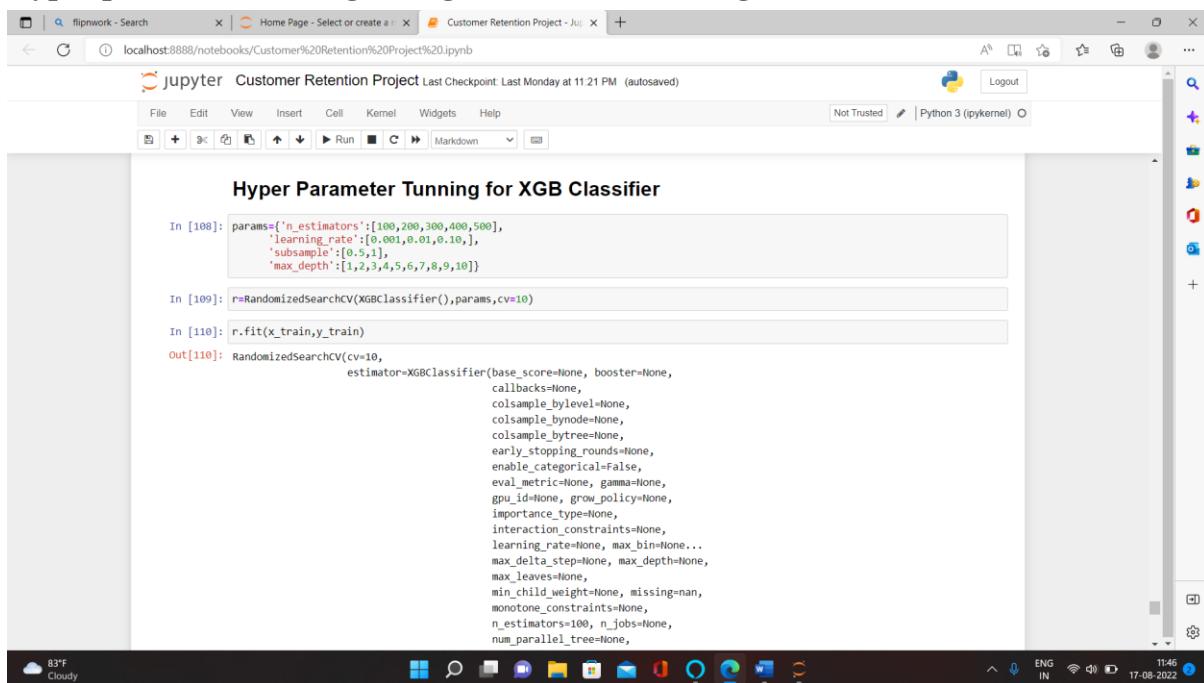
The screenshot shows a Jupyter Notebook interface with the title "Hyper Parameter Tunning for RandomForest Classifier." The code in cell [82] is as follows:

```
# Hyper Parameter tuning using RandomisedSearchCV.  
from sklearn.model_selection import RandomizedSearchCV  
  
# Number of trees in random forest  
n_estimators = [int(x) for x in range(10,100,10)]  
# Number of features to consider at every split  
max_features = ['auto', 'sqrt']  
# Maximum number of levels in tree  
max_depth = [int(x) for x in np.linspace(10, 110, num = 11)]  
max_depth.append(None)  
# Minimum number of samples required to split a node  
min_samples_split = [2, 5, 10]  
# Minimum number of samples required at each leaf node  
min_samples_leaf = [1, 2, 4]  
# Method of selecting samples for training each tree  
bootstrap = [True, False]  
# Create the random grid  
random_grid = {'n_estimators': n_estimators,  
               'max_features': max_features,  
               'max_depth': max_depth,  
               'min_samples_split': min_samples_split,  
               'min_samples_leaf': min_samples_leaf,  
               'bootstrap': bootstrap}  
print(random_grid)  
  
{'n_estimators': [10, 20, 30, 40, 50, 60, 70, 80, 90], 'max_features': ['auto', 'sqrt'], 'max_depth': [10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, None], 'min_samples_split': [2, 5, 10], 'min_samples_leaf': [1, 2, 4], 'bootstrap': [True, False]}
```

In cell [83], the code is:

```
rf_random = RandomizedSearchCV(estimator = rf, param_distributions = random_grid, n_iter = 100, cv = 3, verbose=2, random_state=42, n_jobs = -1)  
# Fit the random search model.
```

### Hyper parameter tunning for XgBoost Classifier using RandomisedSearchCV



The screenshot shows a Jupyter Notebook interface with the title "Hyper Parameter Tunning for XGB Classifier." The code in cell [108] is as follows:

```
params=[{'n_estimators':[100,200,300,400,500],  
         'learning_rate':[0.001,0.01,0.1,0.10],  
         'subsample':[0.5,1],  
         'max_depth':[1,2,3,4,5,6,7,8,9,10]}]
```

In cell [109], the code is:

```
r=RandomizedSearchCV(XGBClassifier(),params,cv=10)
```

In cell [110], the code is:

```
r.fit(x_train,y_train)
```

The output in cell [110] is:

```
Out[110]: RandomizedSearchCV(cv=10,  
                           estimator=XGBClassifier(base_score=None, booster=None,  
                           callbacks=None,  
                           colsample_bylevel=None,  
                           colsample_bynode=None,  
                           colsample_bytree=None,  
                           early_stopping_rounds=None,  
                           enable_categorical=False,  
                           eval_metric=None, gamma=None,  
                           gpu_id=None, grow_policy=None,  
                           importance_type=None,  
                           interaction_constraints=None,  
                           learning_rate=None, max_bin=None...  
                           max_delta_step=None, max_depth=None,  
                           max_leaves=None,  
                           min_child_weight=None, missing='nan',  
                           monotone_constraints=None,  
                           n_estimators=100, n_jobs=None,  
                           num_parallel_tree=None,
```

Both models performing very well and we have to select only one of them, so we will choose Random Forest Classifier model as it is perfectly giving cross validation score and accuracy score.

## **CONCLUSION**

The results of this study suggest following outputs which might be useful for E-commerce websites to extend their business

The cost of the product, the reliability of the E-commerce company and the return policies all play an equally important role in deciding the buying behaviour of online customers.

The cost is an important factor as it was the basic criteria used by online retailers to attract customers. The reliability of the E-commerce company is also important, as it is even required in offline retail.

It is important because customers are paying online, so they need to be sure of security of the online transaction. The return policies are important because in online retail customer does not get to feel the product. Thus, people want to be sure that it will be possible to return the product if he does not like it in real.

Whereas, the logistics factor, which included Cash on delivery option, One day delivery and the quality of packaging plays a secondary role in this process though these are Must-be-quality. This is so because these all does not interfere with the real product and people believe that this is the basic value that E-commerce websites provide.

All the websites were not equally preferred by online customers. Amazon was the most preferred followed by Flipkart. This can be explained easily by previous result that we got. These two companies are most trusted in the industry and hence, have a huge reliability. Also, the sellers listed on these websites are generally from Tier 1 cities as compared to Snapdeal and PayTM which have more sellers from tier 2 and 3 cities. Also, these websites have the most lenient return policies as compared to others and also the time required to process a return is low for these.