Project: Black Friday Purchase Prediction

By Aishwary Shukla

ACKNOWLEDGMENT

The aim of this project was to analyse factors related to the purchase amount in Black Friday sale and predict purchase amount customers are wearing to spend during Black Friday sale using Machine Learning. A data set was provided. From this dataset intense data analysis was done using different visualization techniques using pandas library. univariate bivariate and multivariate analysis to analyse the relationship between the features and the label that is purchase. with the help of which we were able to present report and conclude the factors that affect the purchase amount during Black Friday sale and further models were built in Jupyter notebook using Python and ML libraries. The problem fell under regression category. Thus, we have used regression algorithms; linear regression, decision tree, Random Forest, KNeighbours Regressor, Support Vector Machine, AdaBoostRegressor respectively. The major discovery is that the machine learning approach can be suitable for these types of problems due to many aspects. Python programming language and its libraries namely Pandas, Numpy, Matplotlib, Seaborn etc are also a good choice for a first step, not the least because of the easily grasped user interface, as well as the wide availability of algorithms within machine learning. Advanced methods such as hyper parameter tuning of best models is also available.

Data References:

Training dataset was provided with which data analysis was done and model was prepared. Further built model was tested on the provided test data set.

I would like to thank my trainer and my internship manager for giving me this opportunity to test my skills:

Mr. Shankar Gaud Tegimanni, DataTrained

Mr. Shwetank Mishra, FlipRobo

Mr. Prateeek Rajvanshi, Clevered Institute

And all colleagues, mentors, trainers from DataTrained and FlipRobo for training me and giving me the opportunity to be an intern and expand my knowledge in the field of Data Science and Artificial Intelligence.

INTRODUCTION

Problem Statement

A retail company “ABC Private Limited” wants to understand the customer purchase behaviour (specifically, purchase amount) against various products of different categories. They have shared purchase summary of various customers for selected high volume products from last month. The data set also contains customer demographics (age, gender, marital status, city_type, stay_in_current_city), product details (product_id and product category) and Total purchase_amount from last month. Now, they want to build a model to predict the purchase amount of customer against various products which will help them to create personalized offer for customers against different products.

Abstract

Abstract - Black Friday marks the beginning of the Christmas shopping festival across the US. On Black Friday big shopping giants like Amazon, Flipkart, etc. lure customers by offering discounts and deals on different product categories. The product categories range from electronic items, Clothing, kitchen appliances, Décor. Research has been carried out to predict sales by various researchers. The analysis of this data serves as a basis to provide discounts on various product items. With the purpose of analyzing and predicting the sales, we have used three models. The dataset Black Friday Sales Dataset available on Kaggle has been used for analysis and prediction purposes. The models used for prediction are linear regression, lasso regression, ridge regression, Decision Tree Regressor, and Random Forest Regressor. Mean Squared Error (MSE) is used as a performance evaluation measure. Random Forest Regressor outperforms the other models with the least MSE score.

The shopping sector has greatly evolved due to the Internet revolution. Most of the population takes into consideration online shopping more than the traditional method of shopping. The biggest perks of online shopping are convenience, better prices, more variety, easy price comparisons, no crowds, etc. The pandemic has boosted online shopping. Though online shopping keeps growing every year, the total sales for the year 2021 are expected to be much higher [16]. Black Friday originated in the USA and is also referred to as Thanksgiving Day. This sale is celebrated on the fourth Thursday of November once every year. This day is marked as the busiest day in terms of shopping. The purpose of organizing this sale is to promote customers to buy more products online to boost the online shopping sector. The prediction model built will help to analyze the relationship among various attributes. Black Friday Sales Dataset is used for training and prediction. Black Friday Sales Dataset is the online biggest dataset and the dataset is also accepted by various e-commerce websites [1]. The prediction model built will provide a prediction based on the age of the customer, city category, occupation, etc. The prediction model is implemented based on models like linear regression, ridge regression, lasso regression, Decision Tree Regressor, Random Forest Regressor. The paper further walks through various sections. Section I introduces the problem, section II illustrates the prior research done in this field, section III provides the data set description, section IV presents the proposed model, with the conclusion in the last section.

LITERATURE REVIEW

Ample research is carried out on the analysis and prediction of sales using various techniques. There are many methods proposed to do so by various researchers. In this section, we will summarize a few of the machine learning approaches. I will be implementing the techniques that I have learned during my course and internship to analyse the customer's past spending and predict the future spending of the customer. The dataset referred is Black Friday Sales Dataset from FlipRobo technologies. Models such as Decision Tree, and Decision Tree with bagging, and XGBoost were used. The performance evaluation measure Root Mean Squared Error (RMSE) is used to evaluate the models used. Simple problems like regression can be solved by the use of simple models like linear regression instead of complex neural network models. Additional machine learning models used for implementation are K-Nearest Neighbor, Random Forest, svm and Ada Boost.

The performance evaluation measures used are Mean Absolute Error (MAE). XGBoost outperformed the other algorithms with a MAE rate of 0.409178. I have analysed and visually represented the sales data provided in the complex dataset from which we ample clarity about how it works, which helps the investors and owners of an organization to analyse and visualize the sales data, which will outcome in the form of a proper decision and generate revenue. The data visualization is based on different parameters and dimensions. The result of which will enable the end-user to make better decisions, ability to predict future sales, increase the production dependencies on the demand, and also regional sales can be calculated. I have also analysed and compared the performance of K-Fold cross-validation and hold-out validation method. The result of the experimentations where k-fold cross-validation gives more accurate results. The accuracy results of K - Fold cross-validation were around 0.1 - 3% more accurate as compared to hold-out validation for the same set of algorithms. The algorithms used for experimentations are Linear Regression, K-Nearest Neighbors algorithm, XGBoost, and Random Forest. The result precision is based on Root Mean Squared Error (RMSE), Variance Score, Training, and Testing Accuracies.

Steps followed

Data Analysis:

- Exploratory data analysis
- Feature Engineering
- Relation between different features/amenities of house and target column i.e., sales price.
- Grouped Data Analysis to get more insights.
- Correlation and identification of multicollinearity problem.
- Identifying and selecting best features that affect fare price.

Model Building:

- Data pre-processing.
- Model Initialization and evaluation.
- Model Validation
- Model Testing

Deployement:

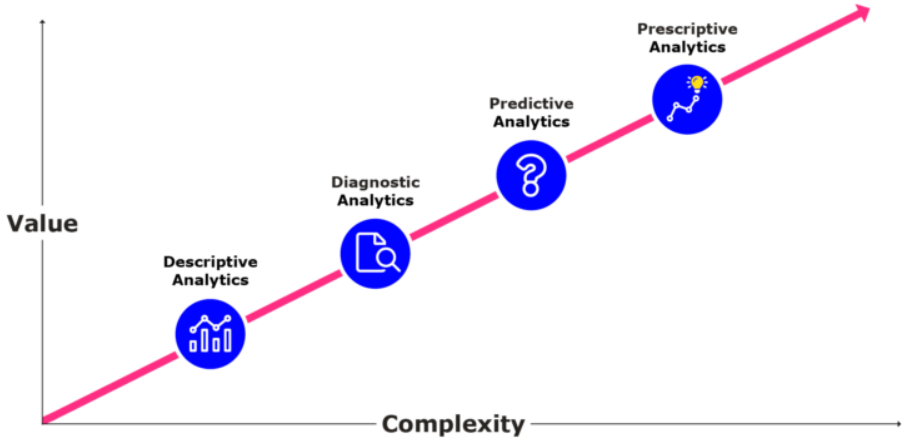
- Pipelining.
- Testing phase

Motivation for the Problem Undertaken

My objective behind making this project is to implement techniques and methods I have learned and practiced during my PG programme in Data Science. Further I continuously

strive to improve my skills, adapt new techniques and methods in data analysis and modelling. Ecommerce industry interest me thus working on this project will give me domain knowledge and technical expertise in deploying ml models for solving real world problems.

Analytical Problem Framing



Analytical Modeling and Machine Learning are typically regarded as two alternative methodologies to model performance of computer systems and applications. In this project we have overviewed three different hybrid modeling methodologies, which leverage on both AM and ML to get the best of the two worlds, namely reducing the model's training time and increasing its accuracy as new training data become available. While research on AM and ML has already reached maturity, investigation on hybrid methodologies is still at its infancy.

Recent work has shown that none of such techniques outperforms the others in terms of accuracy for every application and for every training data set. An interesting research line to pursue, in the light of this result, is to identify which characteristics of the applications being modeled, or of the AM and ML techniques employed for modeling may lead a given hybrid methodology to outperform the others. Another interesting issue to investigate is whether it is possible to fruitfully combine further the described methodologies, with the goal of building a unique, more accurate, meta-hybrid model.

Data Sources and their formats

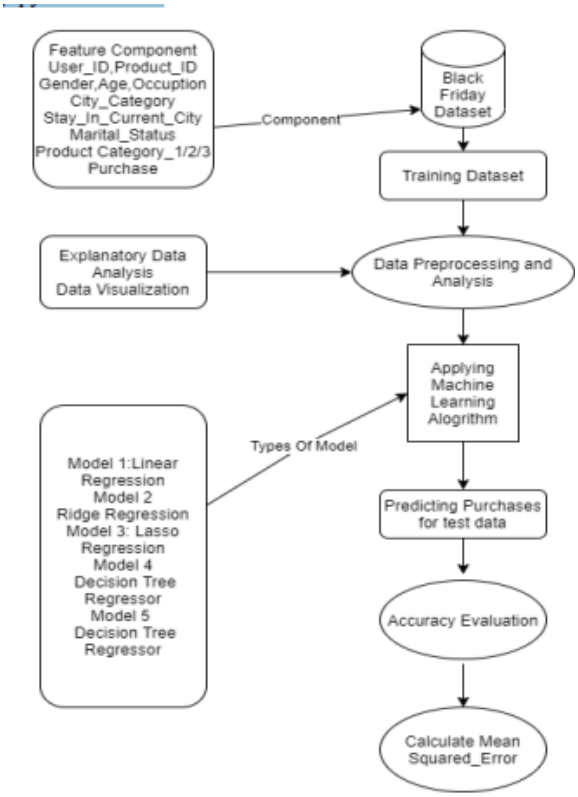
Black Friday Sales Dataset is provided to for analysis and training. The dataset consists of sales transaction data. The dataset consists of 5,50,069 rows. The dataset consists of attributes such as user_id, product_id, martial_status, city_category, occupation, etc.

Sr No	Variable	Definition
1	User_ID	Unique ID of customer
2	Product_ID	Unique product ID
3	Gender	Sex of customer
4	Age	Customer age
5	Occupation	Occupation of customer
6	City_Category	City category of customer
7	Stay_In_Current_City	Number of years customer stays in city
8	Maritial_Status	Customer marital status
9	Product_Category_1	Product Categor
10	Product_Category_2	Product Categor
11	Product_Category_3	Product Categor
12	Purchase	Amount of customer purchas

The dataset definition is mentioned above table.

The Black Friday Sales dataset is used for training various machine learning models and for predicting the purchase amount of customers on black friday sales [1]. The purchase prediction made will provide an insight to retailers to analyse and personalize offers for more customer's preferred products.

The Purchase Variable will be the predictor variable. The Purchase Variable will predict the amount of purchase made by a customer on the occasion of black friday sales. As mentioned in the introduction, the proposed approach tries to implement the machine learning models such as Linear Regression, Ridge Regression, Lasso Regression, Decision Tree Regressor, and Random Forest. Regressor to forecast sales. Figure 1 depicts the flow of data through the proposed model. Exploratory Data Analysis has been performed on the dataset [5]. The tools used for the data analysis are python, pandas, matplotlib, NumPy, array, seaborn and jupyter notebook.



The Black Friday Sales Dataset is the input dataset. Data visualization of the various attributes of this dataset is performed.

Data preprocessing which mainly includes filling missing values is performed. The categorical values are label encoded to numeric form. The categories such as Gender where F represents female and M represents Male is converted to numerical form as 0 and 1 also other categorical values such as City_Category, Stay_In-Current_City, Age are converted to numerical form by applying Label Encoding. The attributes such as User_id and Product_id are removed to train the model with no bias based on user_id or product_id and to achieve better performance. The algorithms used for implementing the system are linear regression, Ridge Regression [19], Lasso Regression [19], Decision Tree Regressor, and RandomForest Regressor. The models are trained using 5 fold cross-validation [4][12]. The performance evaluation measure used is Mean Squared Error (MSE). Random Forest Regressor performs better than the other algorithms with a MSE score of 3062.719.

Hardware for Machine Learning

- Computer/Intel's CPU/Powerful GPU enabled CPU

Hardware for Machine Learning

- Jupyter Notebook/Anaconda
- Scikit learn
- Pandas
- Numpy
- Matplotlib
- Seaborn

Exploratory Data Analysis

Total rows and columns: the data set provided to us for prior analysis and base model building contains 55,000 plus rows and 12 columns.

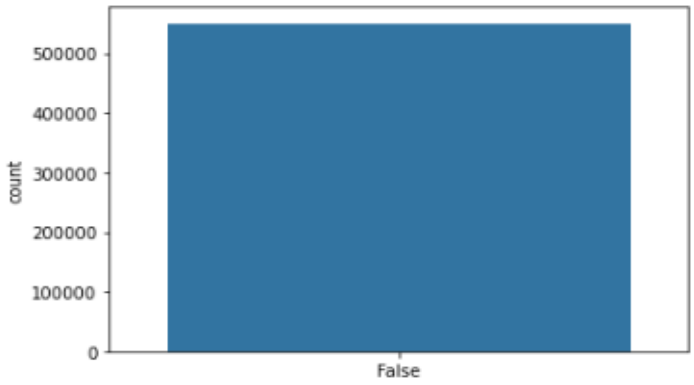
	User_ID	Product_ID	Gender	Age	Occupation	City_Category	Stay_In_Current_City_Years	Marital_Status	Product_Category_1	Product_Category_2	Pr
0	1000001	P00069042	F	0-17	10	A	2	0	3	NaN	
1	1000001	P00248942	F	0-17	10	A	2	0	1	6.0	
2	1000001	P00087842	F	0-17	10	A	2	0	12	NaN	
3	1000001	P00085442	F	0-17	10	A	2	0	12	14.0	
4	1000002	P00285442	M	55+	16	C	4+	0	8	NaN	
...	...	...	...	...	...	...	...	...	...	...	
550063	1006033	P00372445	M	51-55	13	B	1	1	20	NaN	
550064	1006035	P00375436	F	26-35	1	C	3	0	20	NaN	
550065	1006036	P00375436	F	26-35	15	B	4+	1	20	NaN	
550066	1006038	P00375436	F	55+	1	C	2	0	20	NaN	
550067	1006039	P00371644	F	46-50	0	B	4+	1	20	NaN	

550068 rows × 12 columns

Analysis of duplicate values: no duplicate values are present in data set.

```
In [71]: sns.countplot(df.duplicated())
```

```
Out[71]: <AxesSubplot:ylabel='count'>
```



Null values: product category 3 and product category two columns contain null values in them and that are approximately 69% and 31% of complete data, thus from it we can conclude that having such null values in the data set can be misleading and imputing them without having prior information to them is also not justified so we can delete these two columns and then we will be left with approximately 16,000 rows on which we will be doing the analysis and building our model.

```
: null_data = df.isnull().sum()
  null_data.sort_values(ascending=0)

: Product_Category_3            383247
  Product_Category_2            173638
  User_ID                            0
  Product_ID                         0
  Gender                             0
  Age                                0
  Occupation                         0
  City_Category                      0
  Stay_In_Current_City_Years         0
  Marital_Status                     0
  Product_Category_1                 0
  Purchase                           0
  dtype: int64
```

Percentage of missing values

Product_Category_3	69.672659
Product_Category_2	31.566643
User_ID	0.000000
Product_ID	0.000000
Gender	0.000000
Age	0.000000
Occupation	0.000000
City_Category	0.000000
Stay_In_Current_City_Years	0.000000
Marital_Status	0.000000
Product_Category_1	0.000000
Purchase	0.000000

Statistical data description:

From statistical analysis **of** from continuous variables conclusions made are listed below:

Occupation values range from zero up to 20 with mean value being 8 that means most customers have a mean occupation labeled as eight.

mean value of product category one column is 5 where minimum and maximum values are one and 20 respectively.

mean value of product category two column is nearly 10 with minimum and maximum values are two and 16.

mean value of product category two column is nearly one where minimum and maximum values in that column are from 3 and 18.

above data indicates the means are skewed.

The purchase column, being the target column, has a mean value of 9263 where the minimum value is 12 and the maximum value is 23,961. standard deviation is 5023 that is quite large in this case.

```
desc = df.describe().T
desc['range']=desc['max']-desc['min']
desc
```

	count	mean	std	min	25%	50%	75%	max	range
User_ID	550068.0	1.003029e+06	1727.591586	1000001.0	1001516.0	1003077.0	1004478.0	1006040.0	6039.0
Occupation	550068.0	8.076707e+00	6.522660	0.0	2.0	7.0	14.0	20.0	20.0
Marital_Status	550068.0	4.096530e-01	0.491770	0.0	0.0	0.0	1.0	1.0	1.0
Product_Category_1	550068.0	5.404270e+00	3.936211	1.0	1.0	5.0	8.0	20.0	19.0
Product_Category_2	376430.0	9.842329e+00	5.086590	2.0	5.0	9.0	15.0	18.0	16.0
Product_Category_3	166821.0	1.266824e+01	4.125338	3.0	9.0	14.0	16.0	18.0	15.0
Purchase	550068.0	9.263969e+03	5023.065394	12.0	5823.0	8047.0	12054.0	23961.0	23949.0

```
df.describe(include='object').T
```

	count	unique	top	freq
Product_ID	550068	3631	P00265242	1880
Gender	550068	2	M	414259
Age	550068	7	26-35	219587
City_Category	550068	3	B	231173
Stay_In_Current_City_Years	550068	5	1	193821

Understanding the value counts in categorical columns:

for gender column there are two unique values namely male and female.

for age column there are seven unique values, and the values are present in age ranges for example 0 to 17.

city categories are divided into three types namely A B&C.

customers who stay in current city years column has five unique values.

```
]: for i in cat_columns:
       print('For column',i,'unique values are: ',df[i].unique())
       print('For column',i,'count of unique values are: ',df[i].nunique(),'\n\n')
```

```
For column Product_ID unique values are:  ['P00069042' 'P00248942' 'P00087842' ... 'P00370293' 'P00371644'
 'P00370853']
For column Product_ID count of unique values are:  3631


For column Gender unique values are:  ['F' 'M']
For column Gender count of unique values are:  2


For column Age unique values are:  ['0-17' '55+' '26-35' '46-50' '51-55' '36-45' '18-25']
For column Age count of unique values are:  7


For column City_Category unique values are:  ['A' 'C' 'B']
For column City_Category count of unique values are:  3


For column Stay_In_Current_City_Years unique values are:  ['2' '4+' '3' '1' '0']
For column Stay_In_Current_City_Years count of unique values are:  5
```

Further detailed description of value counts is mentioned below:

Value Counts:

For column -- Product_ID -- value counts are:
P00265242 1880
P00025442 1615
P00110742 1612
P00112142 1562
P00057642 1470
...
P00314842 1
P00298842 1
P00231642 1
P00204442 1
P00066342 1
Name: Product_ID, Length: 3631, dtype: int64

For column -- Gender -- value counts are:
M 414259
F 135809

Name: Gender, dtype: int64

For column -- Age -- value counts are:
26-35 219587
36-45 110013
18-25 99660
46-50 45701
51-55 38501
55+ 21504
0-17 15102
Name: Age, dtype: int64

For column -- City_Category -- value counts are:
B 231173
C 171175
A 147720
Name: City_Category, dtype: int64

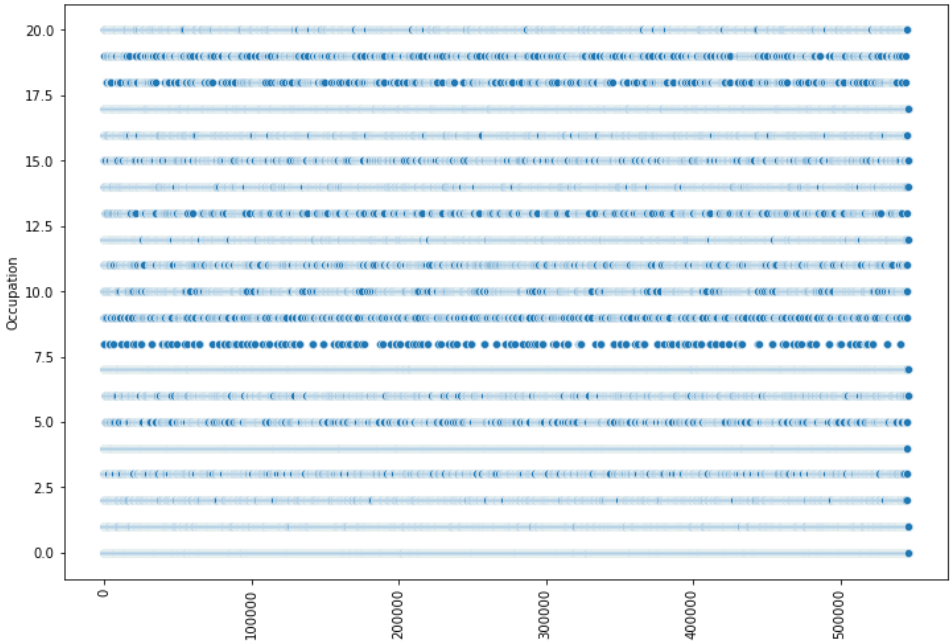
For column -- Stay_In_Current_City_Years -- value counts are:
1 193821
2 101838
3 95285
4+ 84726
0 74398
Name: Stay_In_Current_City_Years, dtype: int64

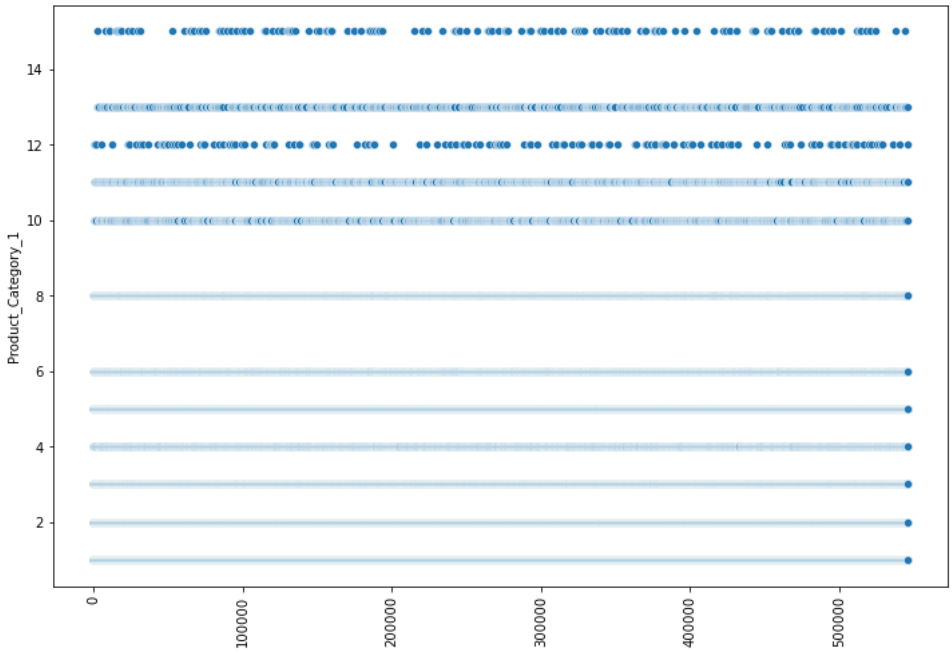
DATASET VISUALIZATION

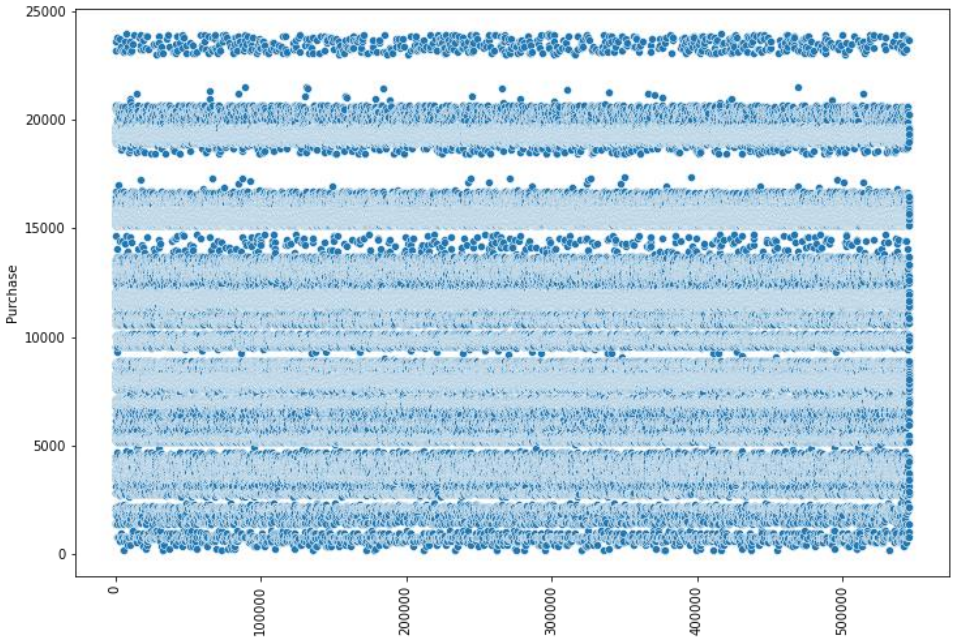
Univariate Analysis

- **Scatterplots**

Scatter plot between index values and features for doing univariate analysis is good for understanding data distribution and identifying high density values.




Product_Category_1
14
12
10
8
6
4
2
0
100000
200000
300000
400000
500000

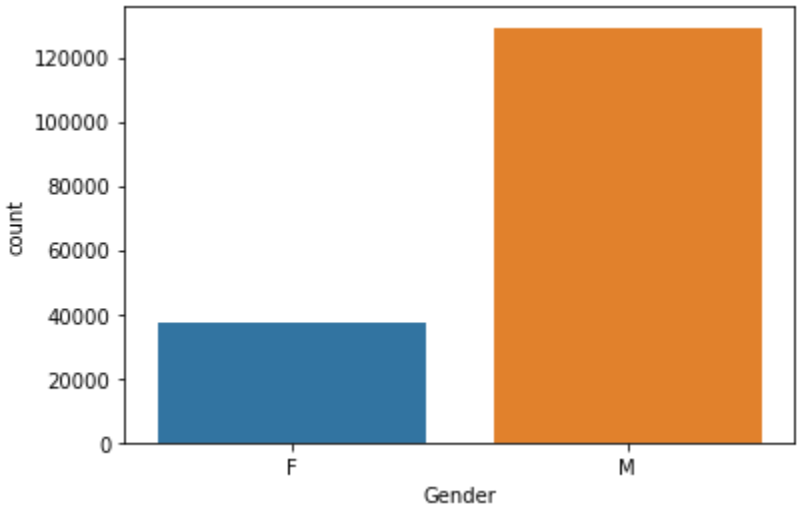

Purchase
25000
20000
15000
10000
5000
0
0
100000
200000
300000
400000
500000

- **Count plots:**

Here data is represented in percentage out of total.

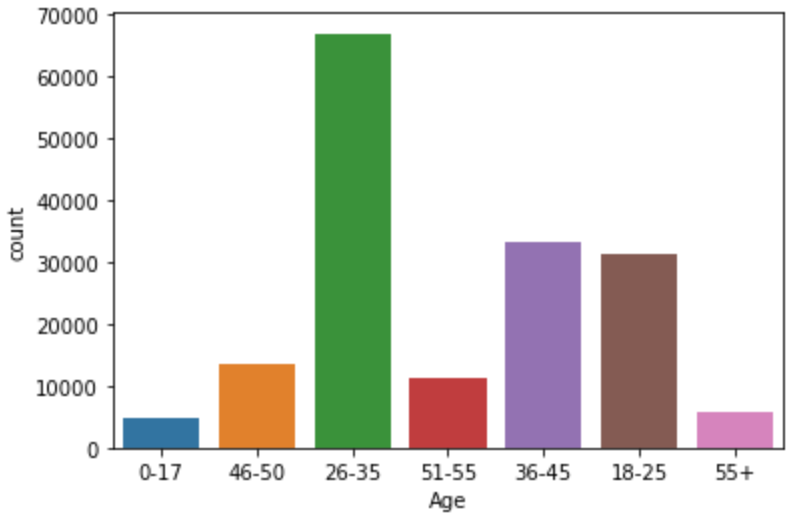
From the whole data set 77 percentage of customers are male and nearly 23 percentage of customers are female.

M 77.464468
F 22.535532



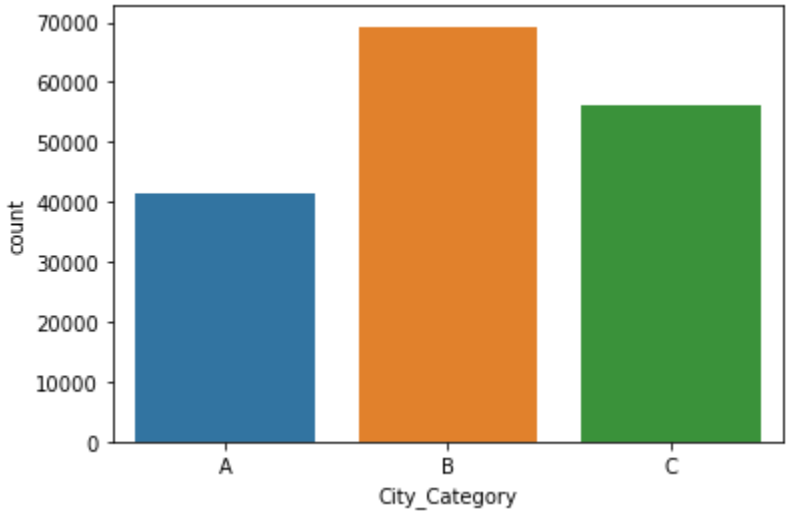
Most customers belong to age range between 26 to 35 years of age and then from 36 to 45 years of age.

26-35 40.128041
36-45 19.952524
18-25 18.772217
46-50 8.016976
51-55 6.693402
55+ 3.515744
0-17 2.921095



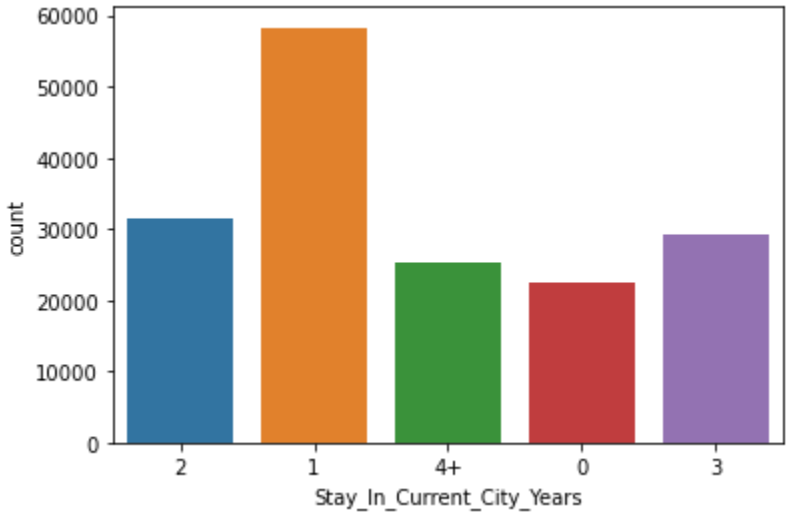
Most customers belong to city with category B.

B 41.507364
C 33.604282
A 24.888353



most customers have stayed for one year in the current city where they are living and from where they are buying.

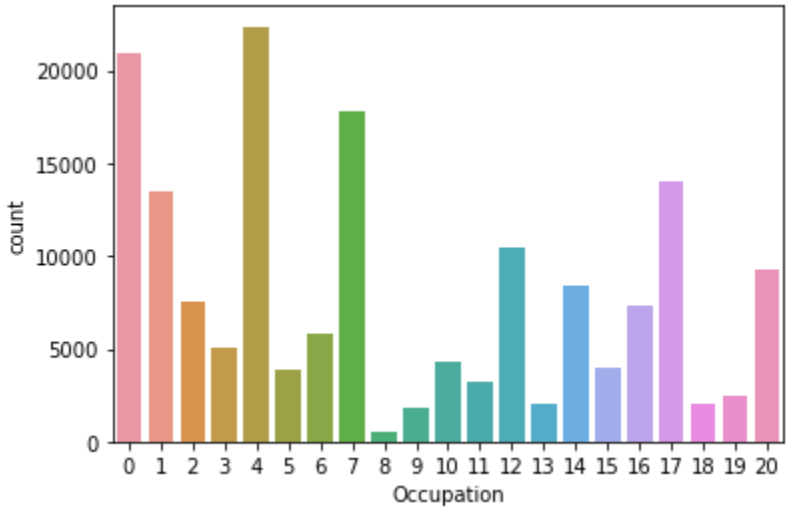
1 34.939846
2 18.891506
3 17.544554
4+ 15.203122
0 13.420972



most customers who are buying belong to occupation category 4. then major contribution is given by people from occupation category 0,7,17 and 1 respectively.

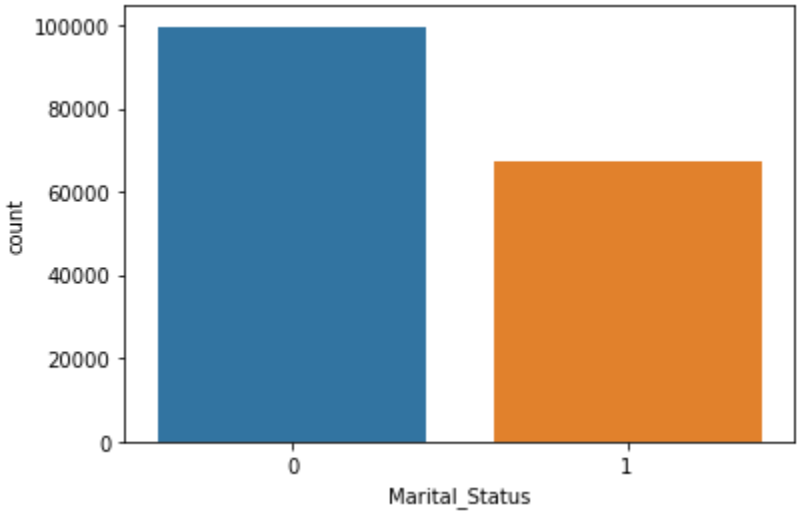
4 13.405986
0 12.570959
7 10.671318
17 8.433590
1 8.099100
12 6.283382
20 5.528680
14 5.027544
2 4.520414
16 4.395730
6 3.486372
3 3.006216
10 2.596196
15 2.389387
5 2.322250
11 1.958986
19 1.444662
13 1.227064
18 1.188699

9 1.118564
8 0.324899



most customers that is approximately 60% of them belong to marital status category 0 and 40% belong to marital status category one.

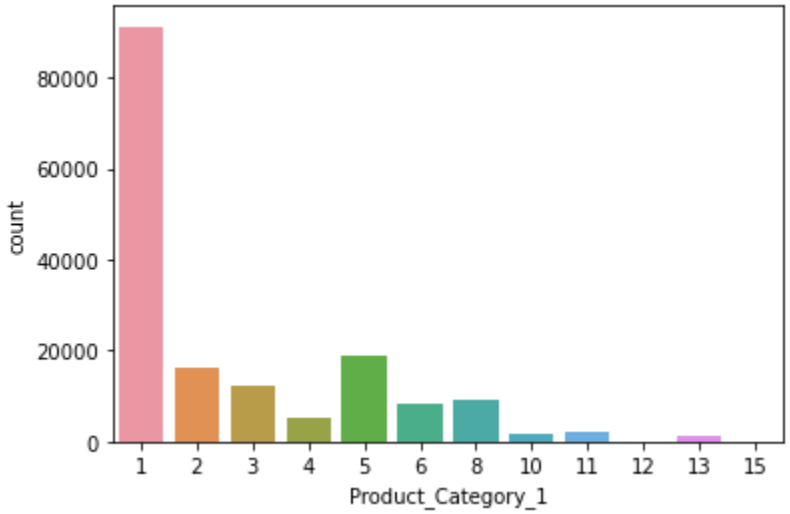
0 59.716103
1 40.283897



- 54% of the customers were categorized into category 1 in product category one.

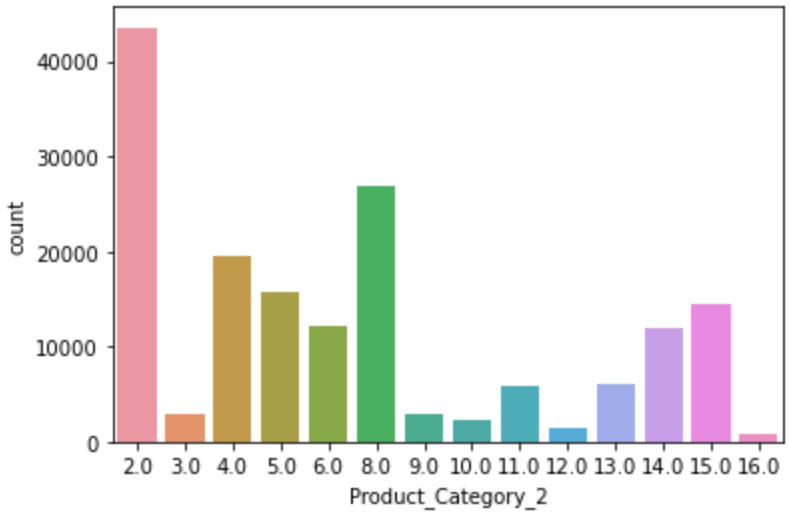
1 54.634608
5 11.396047
2 9.776347
3 7.470283
8 5.435766
6 5.034138
4 3.206431

11 1.175511
10 1.052026
13 0.633014
12 0.095312
15 0.090516
Name: Product_Category_1, dtype: float64



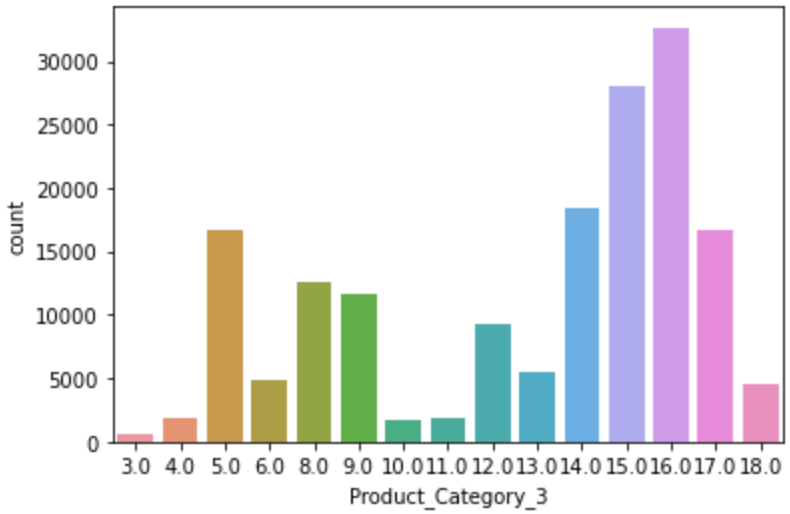
- 26% of the customers were categorized into Category 2 in product category 2.

2.0 26.096235
8.0 16.153242
4.0 11.721546
5.0 9.492210
15.0 8.688355
6.0 7.275463
14.0 7.196336
13.0 3.669202
11.0 3.473184
9.0 1.789343
3.0 1.728799
10.0 1.355345
12.0 0.831430
16.0 0.529310
Name: Product_Category_2, dtype: float64

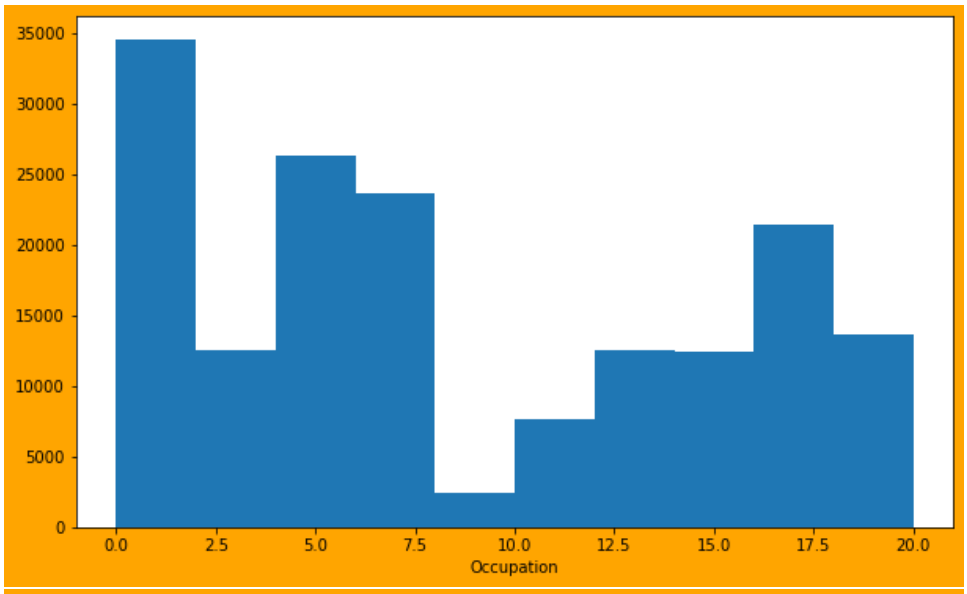


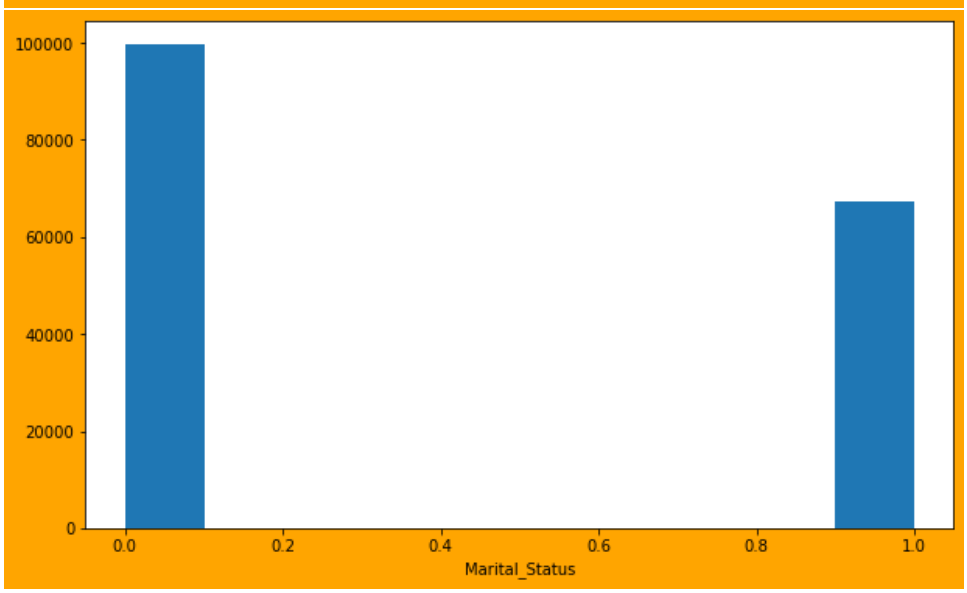
- Approximately 20% of the customers were categorized into 16 in product category 3.

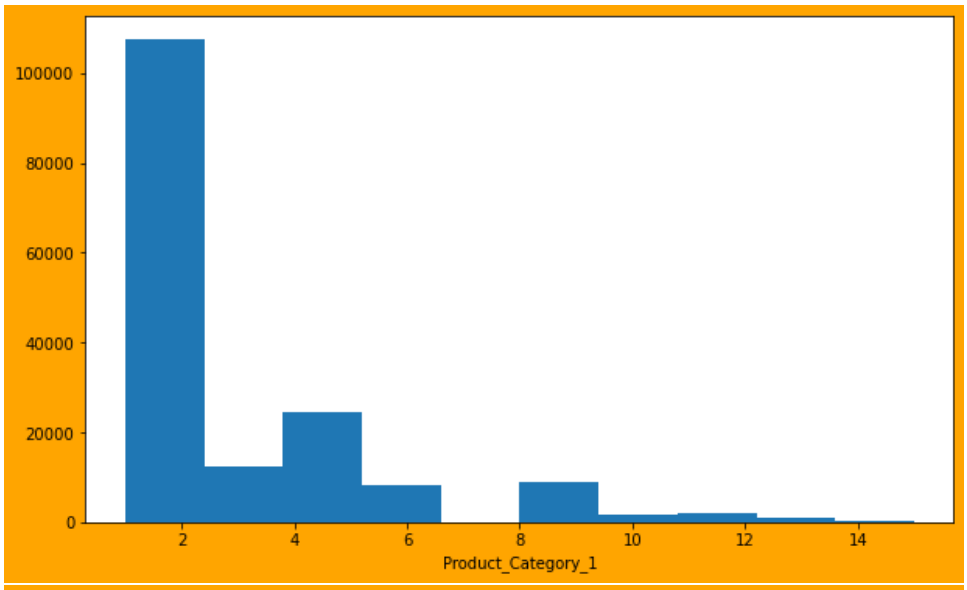
16.0 19.563484
15.0 16.792250
14.0 11.046571
17.0 10.011929
5.0 9.985553
8.0 7.530227
9.0 6.940973
12.0 5.542468
13.0 3.272370
6.0 2.931286
18.0 2.774831
4.0 1.123959
11.0 1.081998
10.0 1.034642
3.0 0.367460
Name: Product_Category_3, dtype: float64

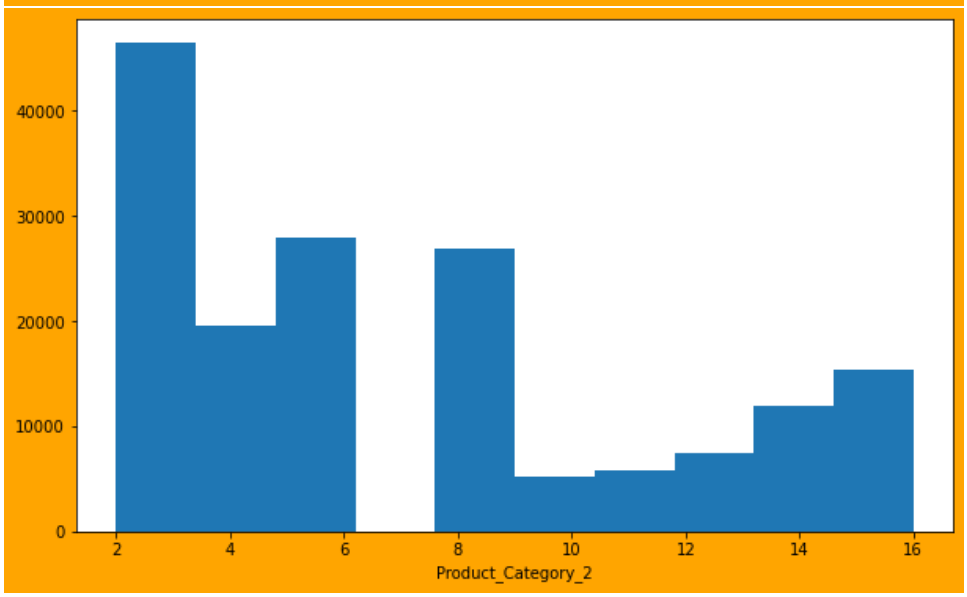


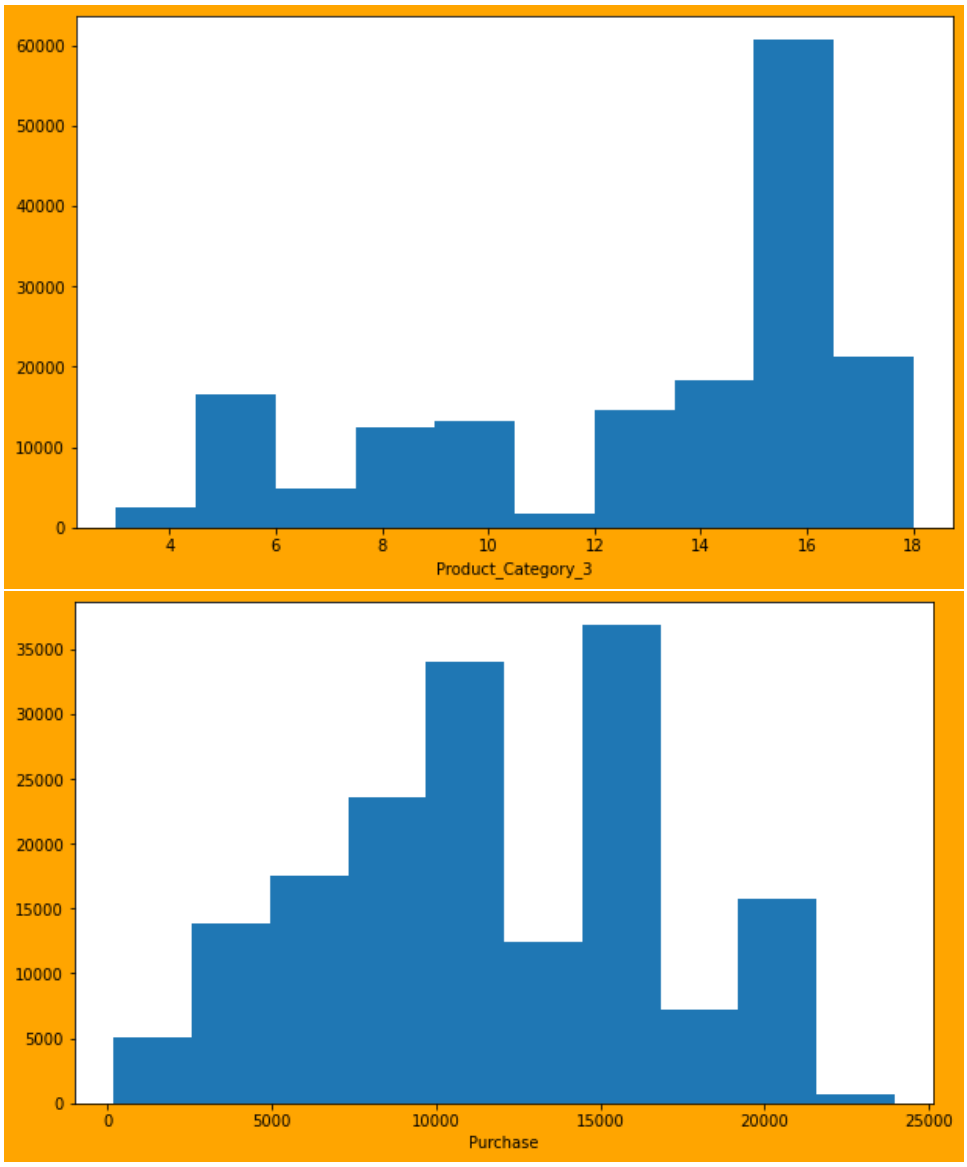
- **Univariate Summary plots**


35000
30000
25000
20000
15000
10000
5000
0
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
Occupation


100000
80000
60000
40000
20000
0
0.0
0.2
0.4
0.6
0.8
1.0
Marital_Status

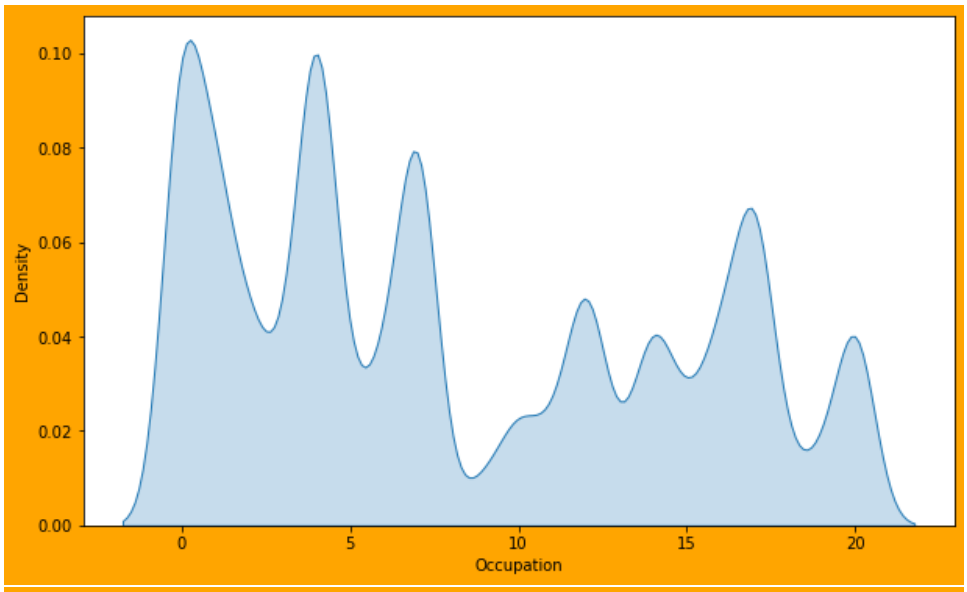

100000
80000
60000
40000
20000
0
2
4
6
8
10
12
14
Product_Category_1

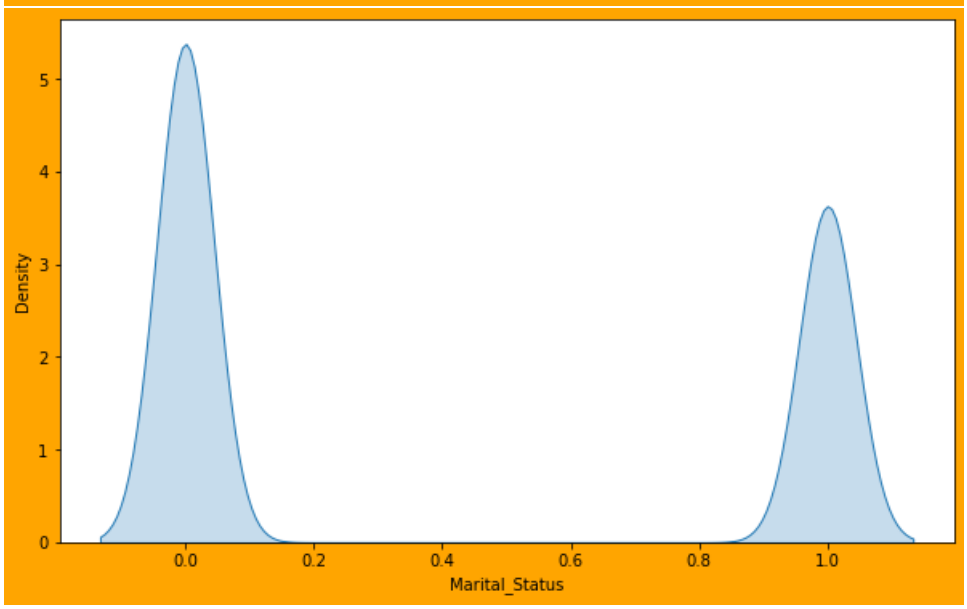

40000
30000
20000
10000
0
2
4
6
8
10
12
14
16
Product_Category_2

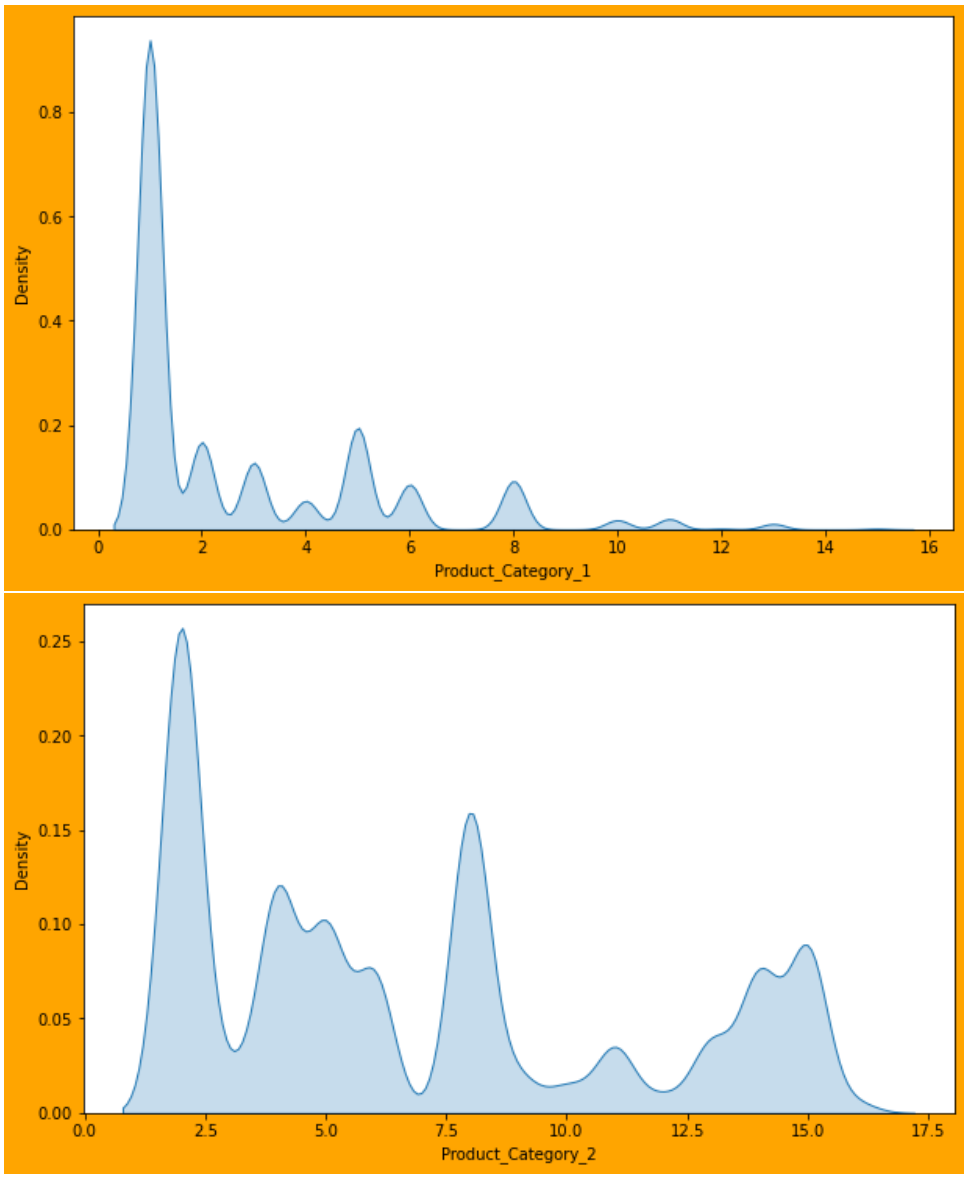


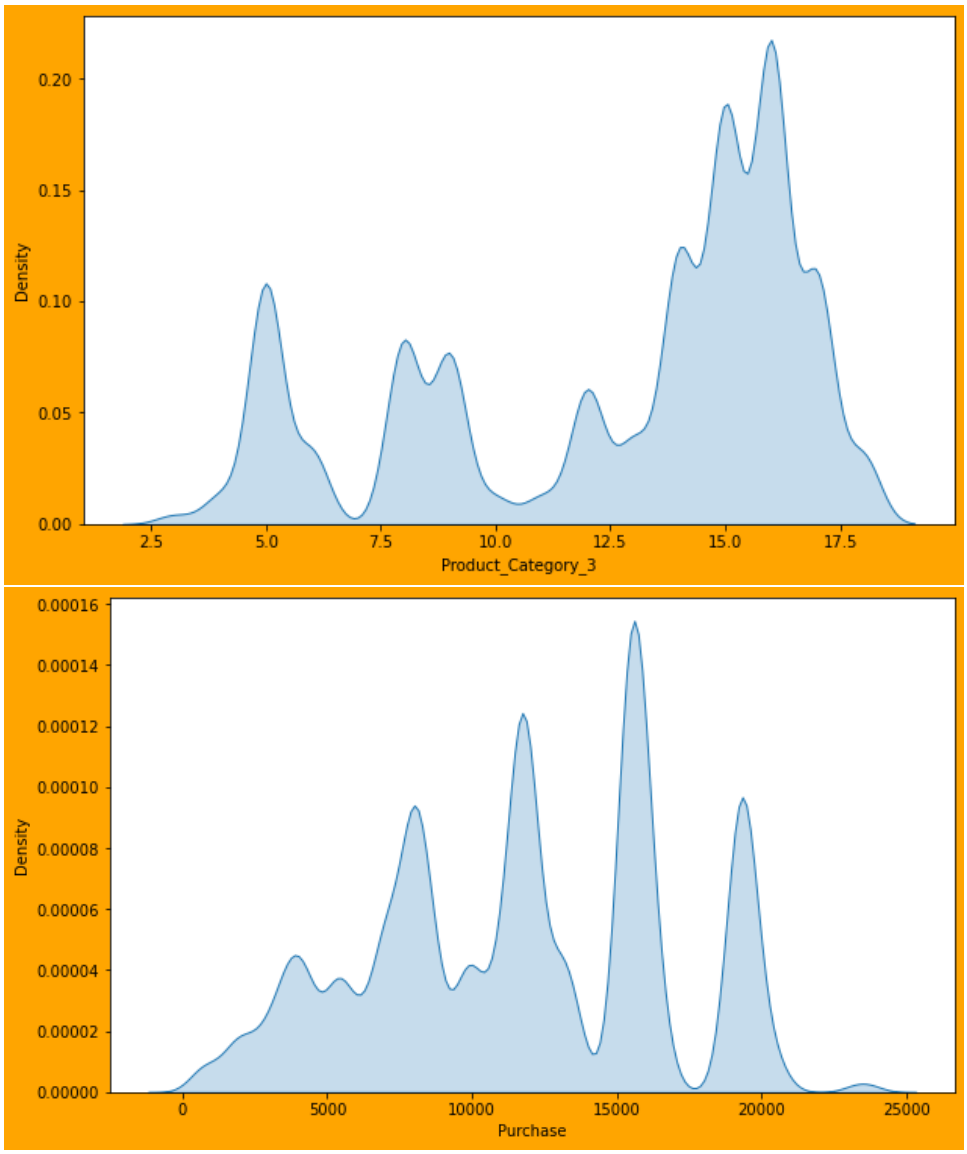
Value counts and histogram data represent the same thing and has helped us in understanding the highest frequency points present in continuous data.

- **KDE plots:**


0.10
0.08
0.06
0.04
0.02
0.00
Density
0
5
10
15
20
Occupation

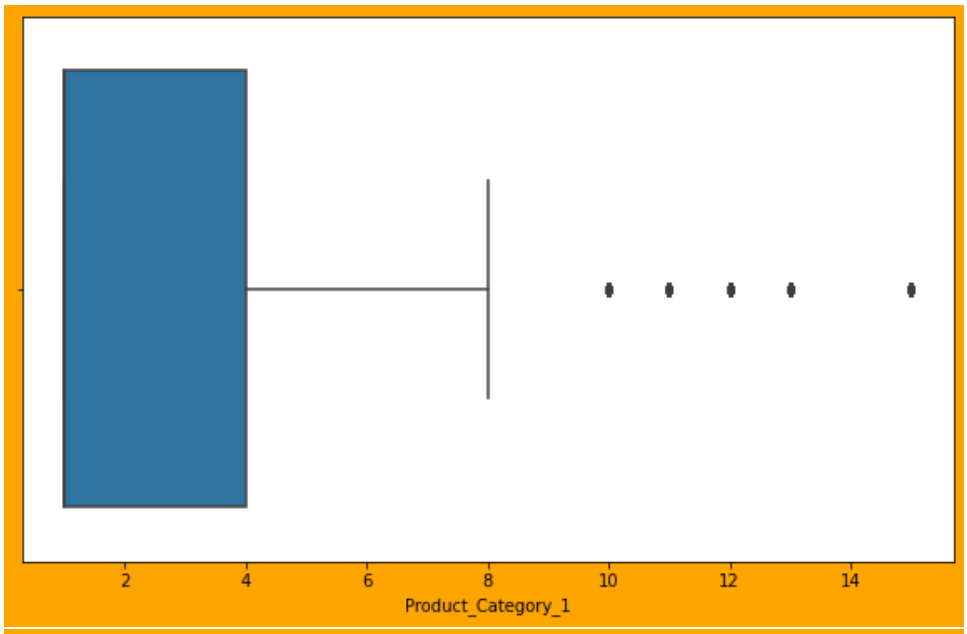

5
4
3
2
1
0
Density
0.0
0.2
0.4
0.6
0.8
1.0
Marital_Status

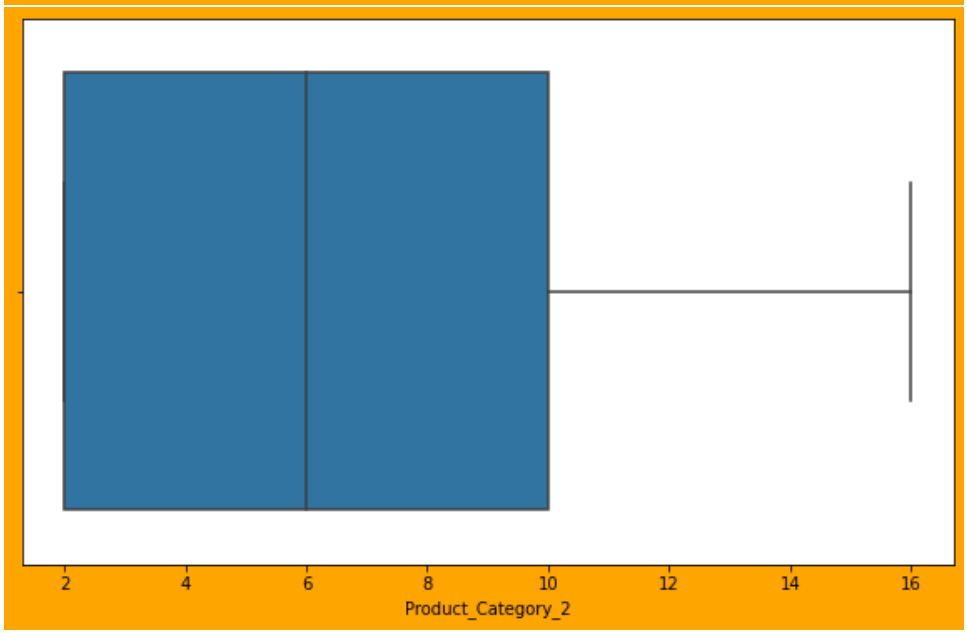

Density
0.0
0.2
0.4
0.6
0.8
0
2
4
6
8
10
12
14
16
Product_Category_1
Density
0.00
0.05
0.10
0.15
0.20
0.25
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
Product_Category_2

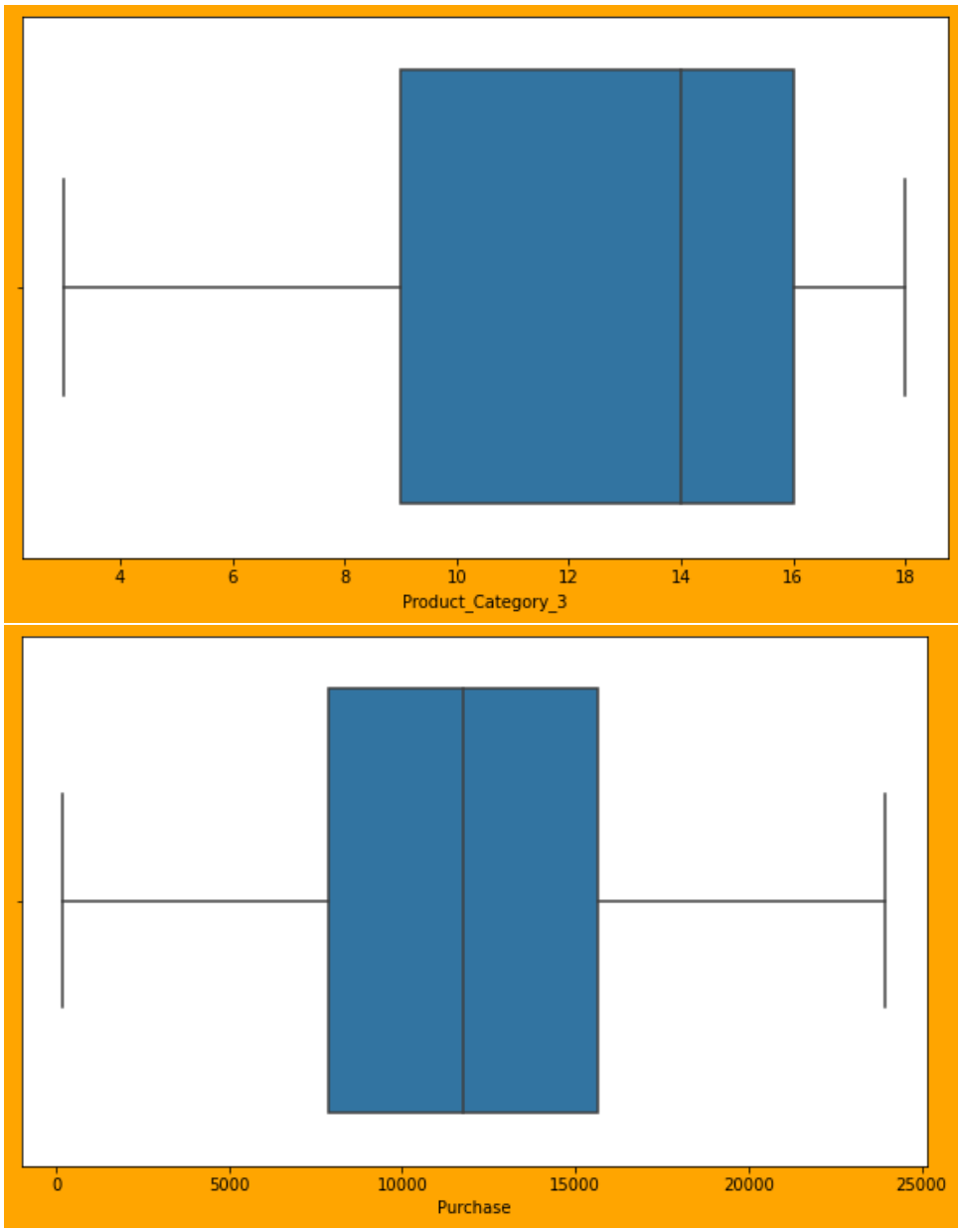


Here using the kernel Density estimate plots all the features and the target have skewed data and have middle peaks that reflect to multimodal nature. For product category one product category 2 the maximum frequency lies add category one and for product category three maximum 3 our frequency lies at 15 unit category. People with medical status labeled as zero had more frequency as compared to Labeled with one. For occupation teacher maximum density lies at label 0.Purchase column had maximum frequency at 15,000.

- **Boxplots:**


2
4
6
8
10
12
14
Product_Category_1

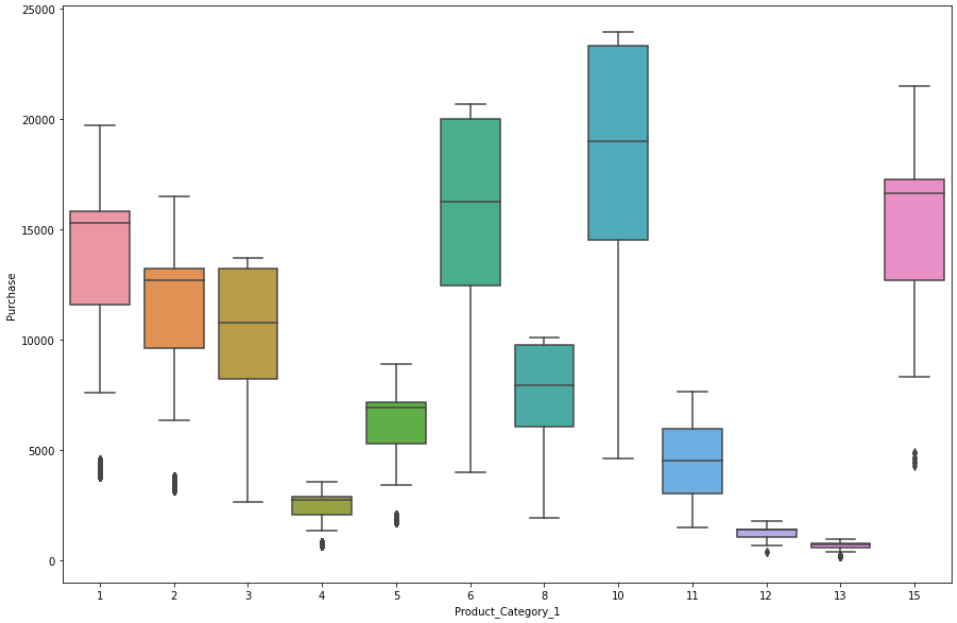

2
4
6
8
10
12
14
16
Product_Category_2



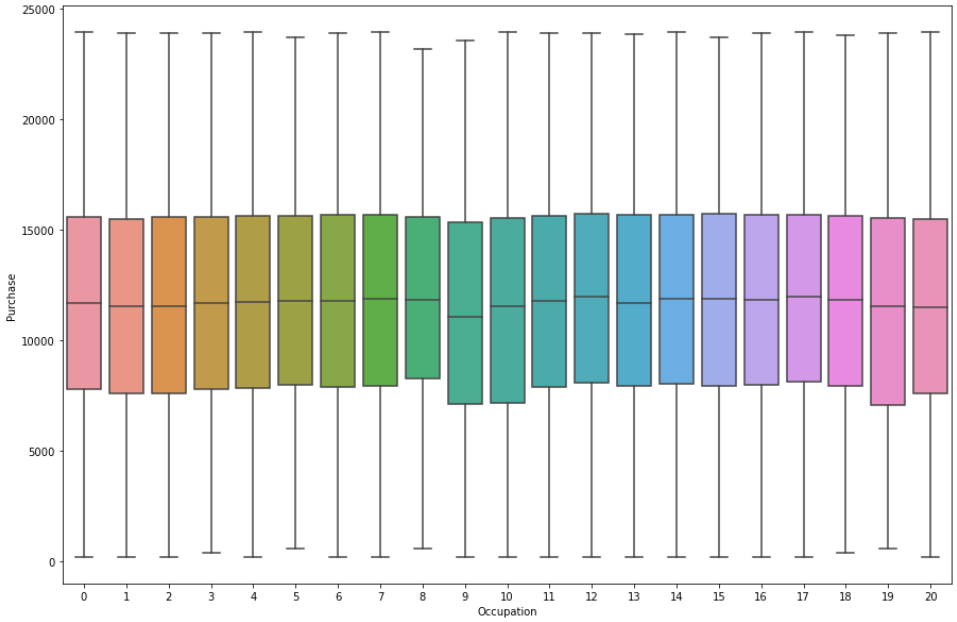
Using box plot that indicates the presence of outliers in different columns in the datasets we have product category one column that is having presence of outliers.

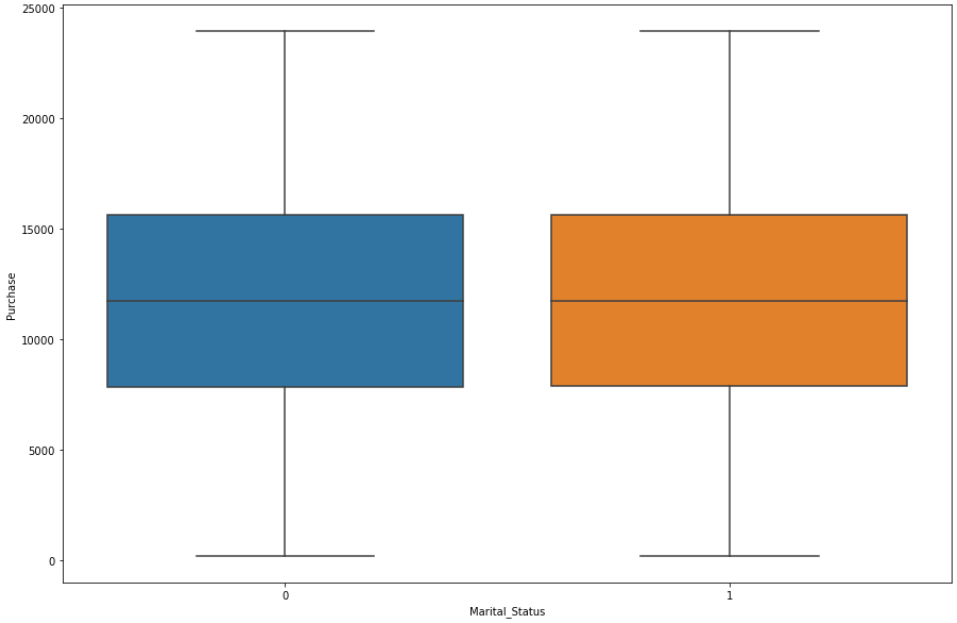
Bivariate Analysis

- **Bibariate Boxplots:**

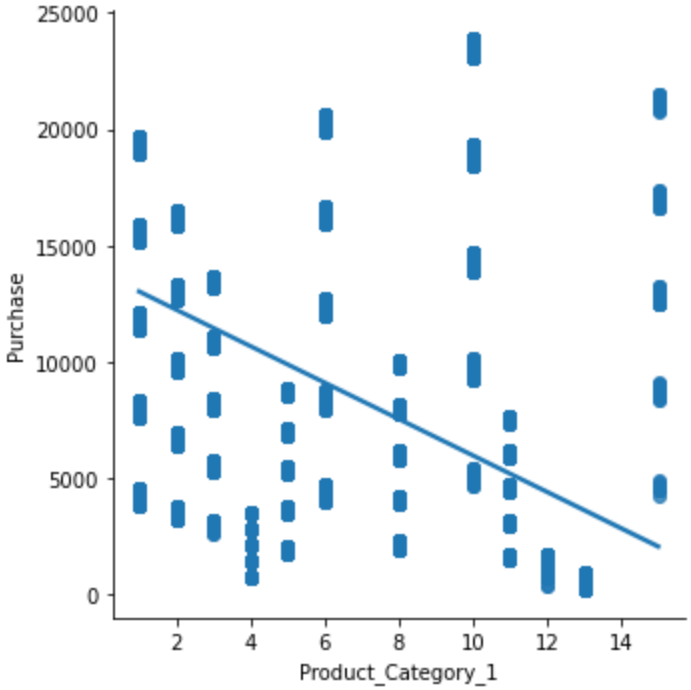


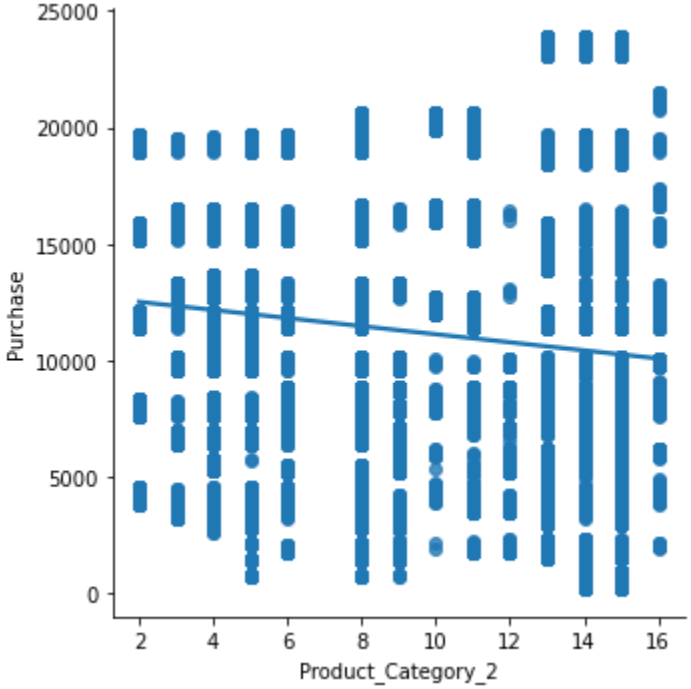
Categorical boxplot between product category one and purchase column indicate that except product categories 3 6 8 10 11 all other categories contain outliers. for product category one with the label 10 the purchase amount had maximum range median value near to 20,000. This indicates people with product category 1 label 10 were willing to spend more as compared to other labels. people from product category one will label 13 12 and 4 were going to spend less very less with purchase amount below 5000.

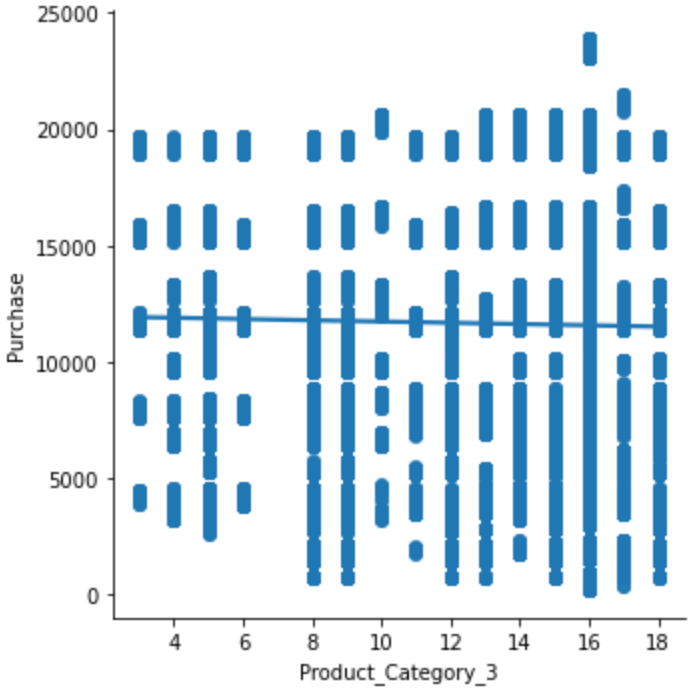

Purchase
Occupation
0
5000
10000
15000
20000
25000
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20


Purchase
Marital_Status
0
5000
10000
15000
20000
25000
0
1

- **LMplots:**

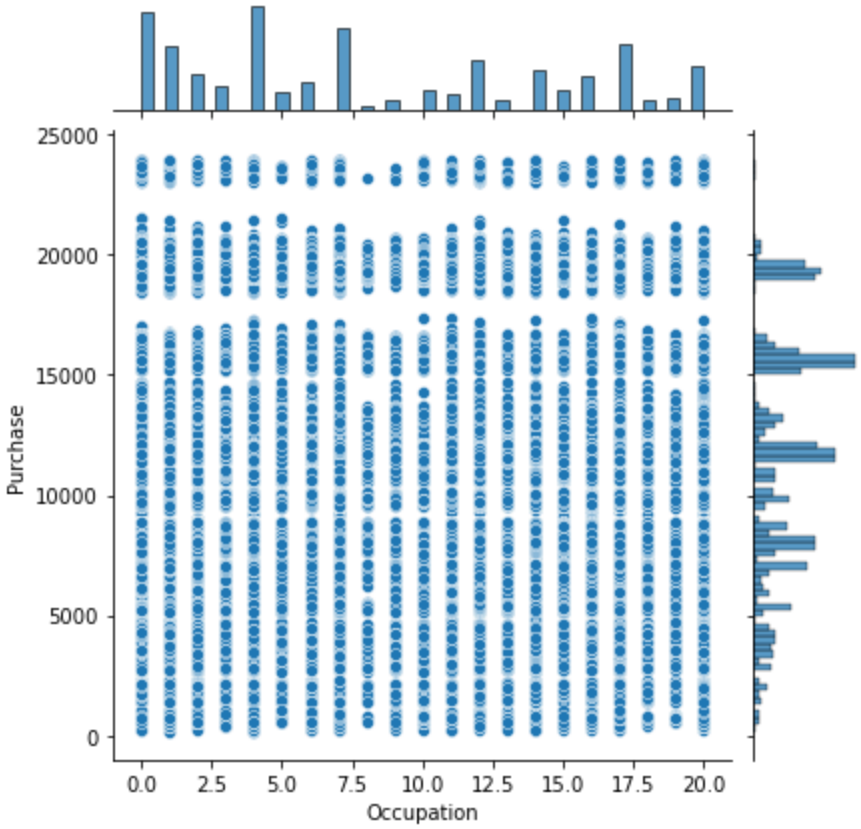

25000
20000
15000
10000
5000
0
Purchase
2
4
6
8
10
12
14
Product_Category_1


25000
20000
15000
10000
5000
0
Purchase
2
4
6
8
10
12
14
16
Product_Category_2

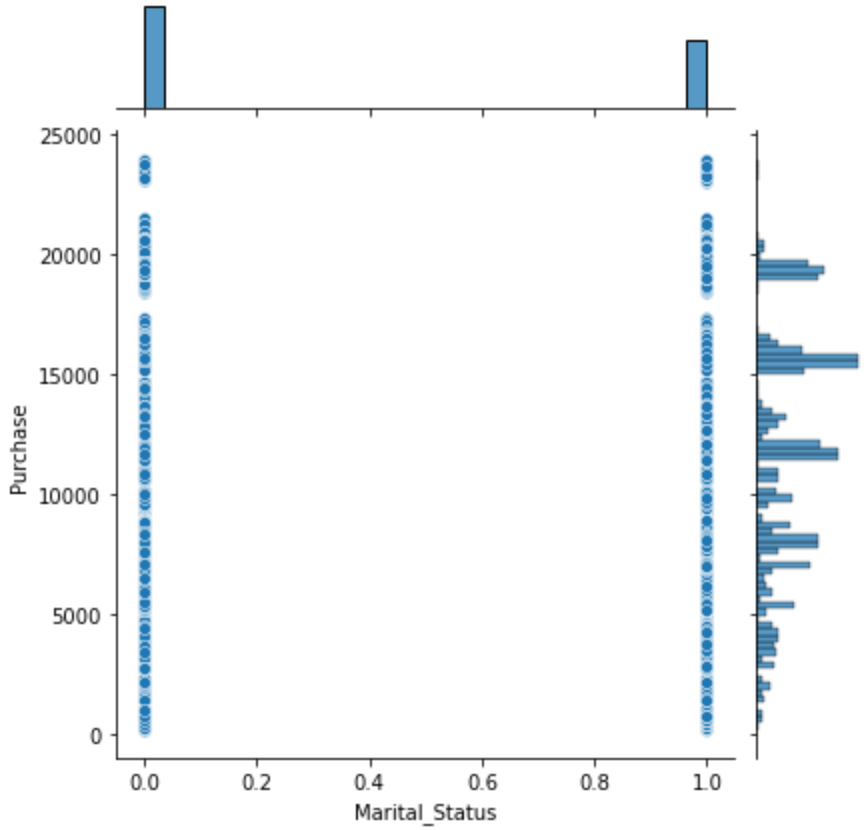


Using regression plot between feature and the label And with the data we have, no positive relationship can be seen between Product category one is 1/2 and three and between label.

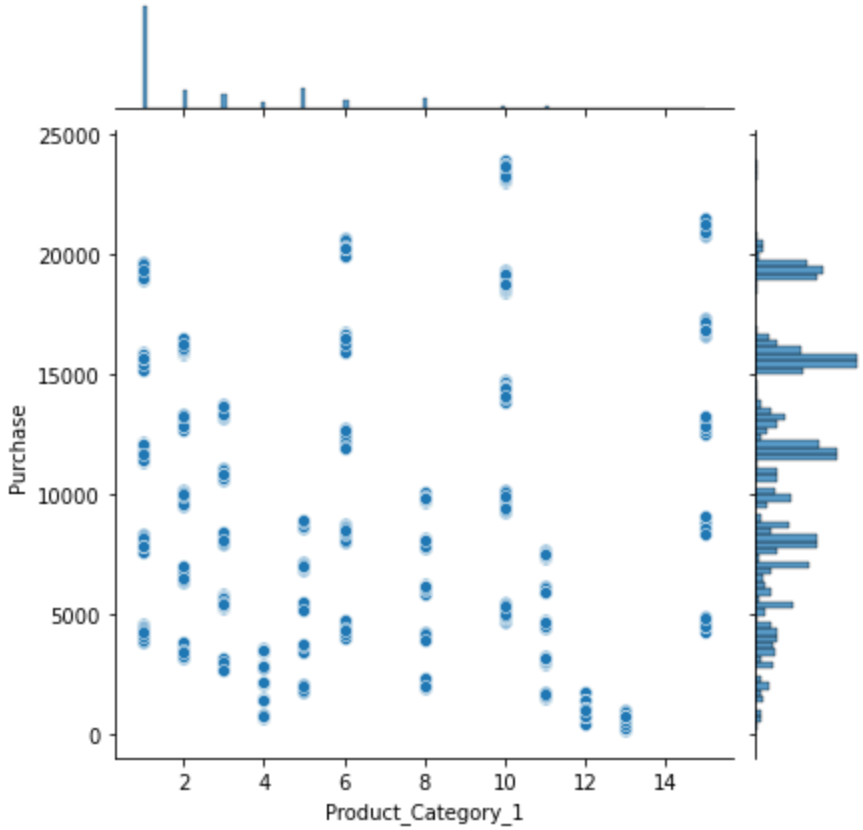
- **Joint plots**



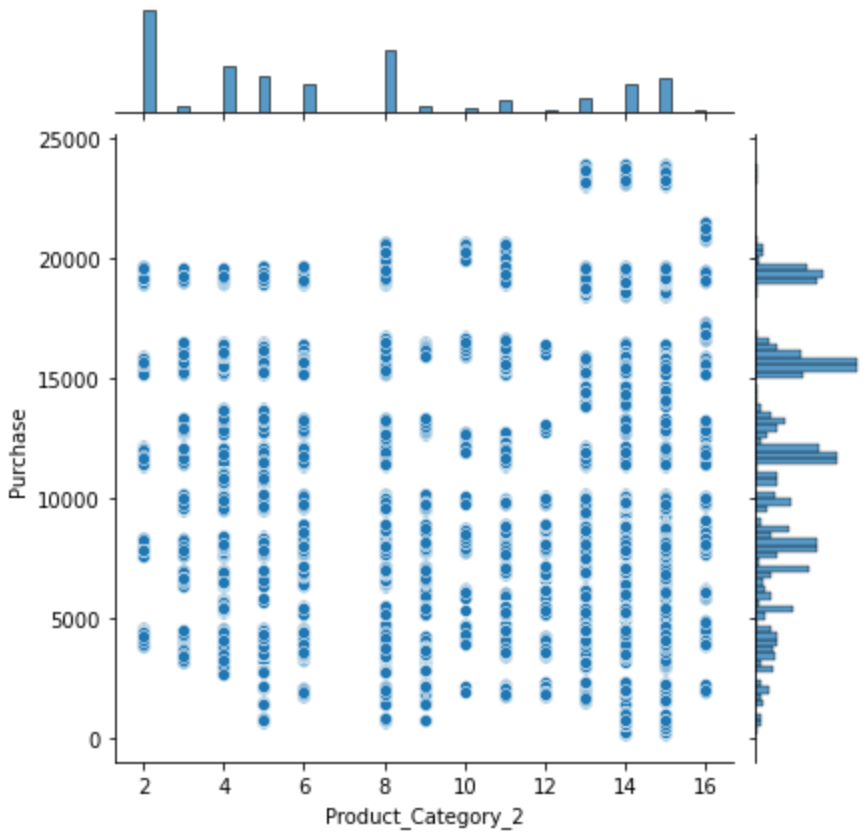
The joint plot between occupation and purchase column indicate that most people were willing to spend an average of 15000 in purchasing. For the occupation feature label 0, 7 and five contributed the most.



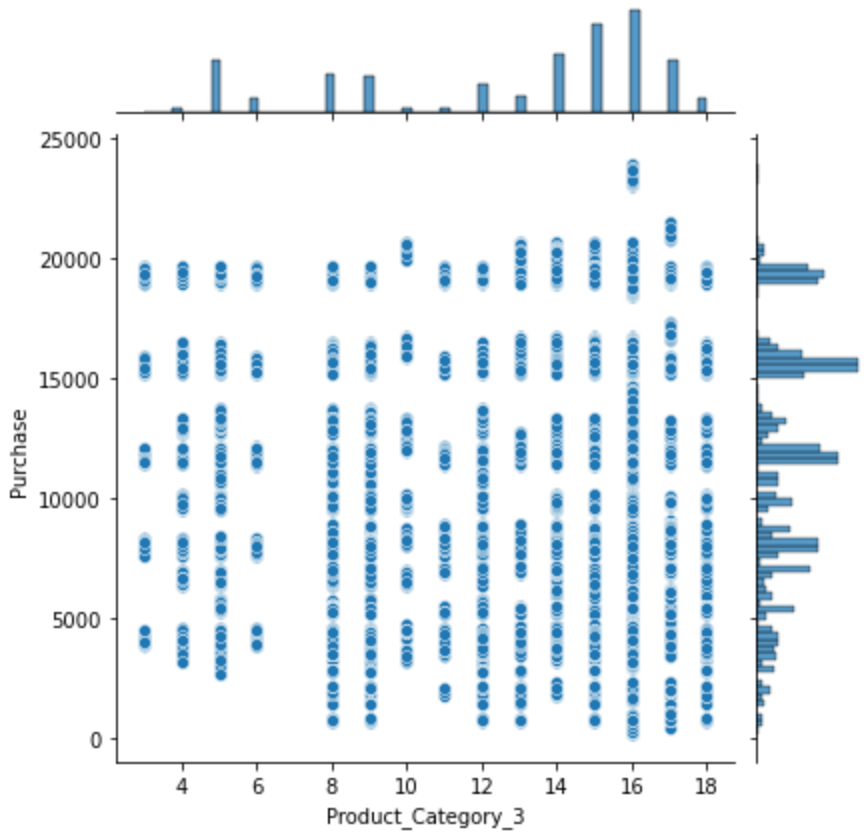
In our data set, the contribution of customers with marital status as 0 was move compared to the customers with marital status labeled as one. This plot also indicates for the purchase column, Maximum contribution is from 15 thousand and around 10,000 respectively.



Customers with product category one and label 0 contributed the most in purchasing. For customers 131 and labeled with 12 14 and four had the least purchase intent and contributed least.

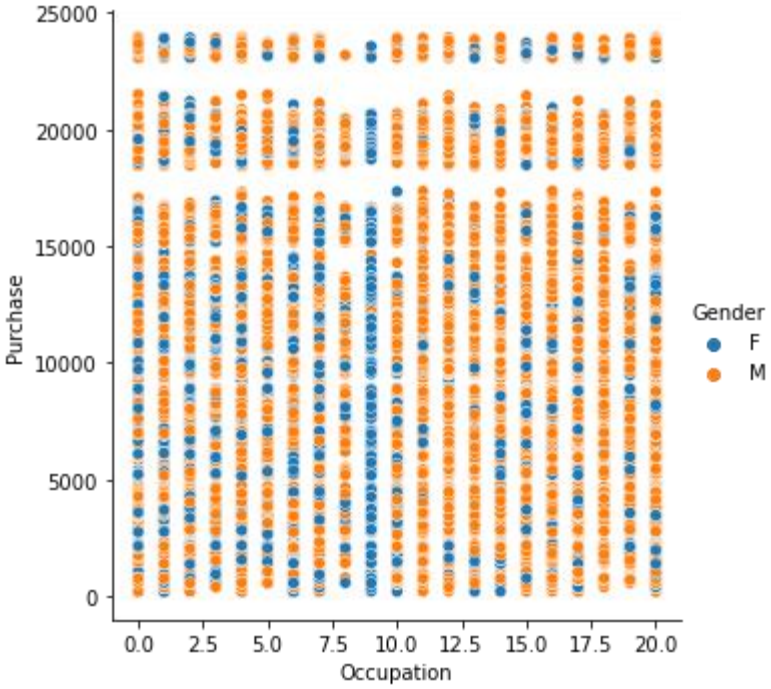


For customers in product category 2 with label 8 and 14 had the widest purchase range from lowest to maximum. And for customers end product category two with label 2 had maximum contribution density in purchasingfrom all other labels in pc 2.

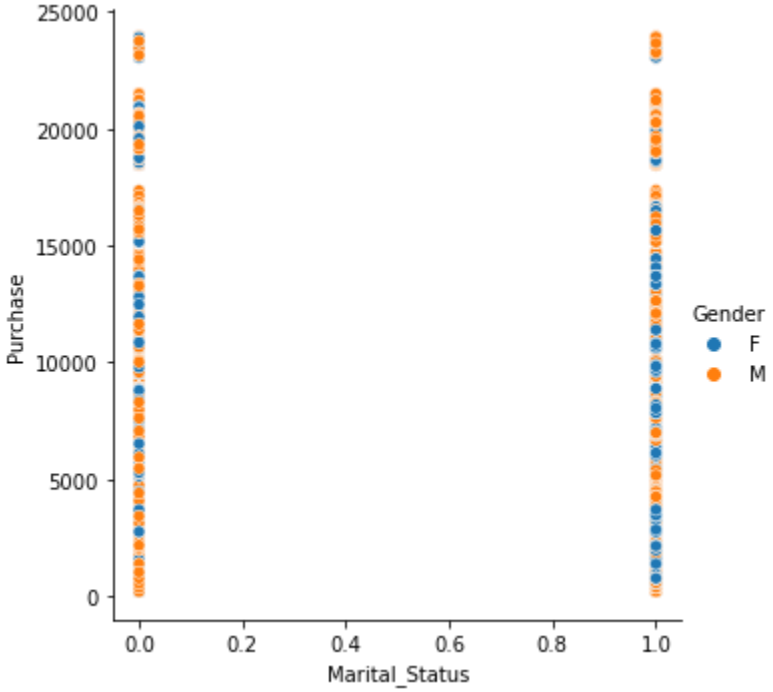


- **Multivariate Analysis**

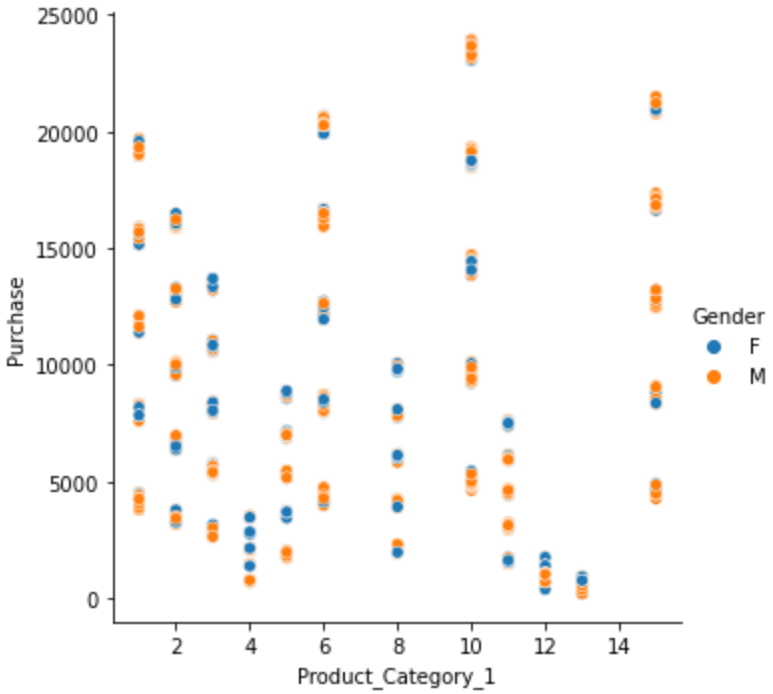
Hue analysis on gender:



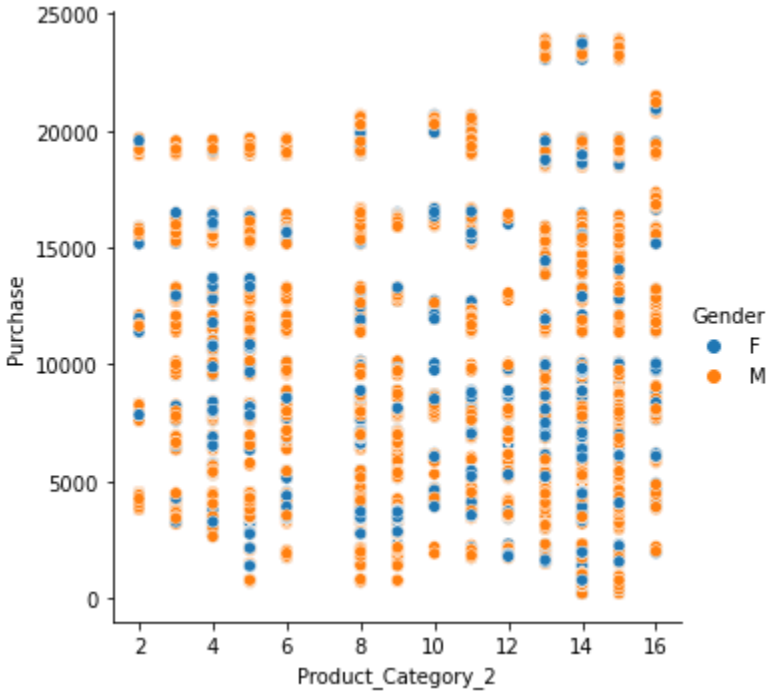
By analyzing the relationship between purchase occupation and gender for occupation label 10 most of the customers were female and the range in which they were willing to spend was from you up to 25,000.



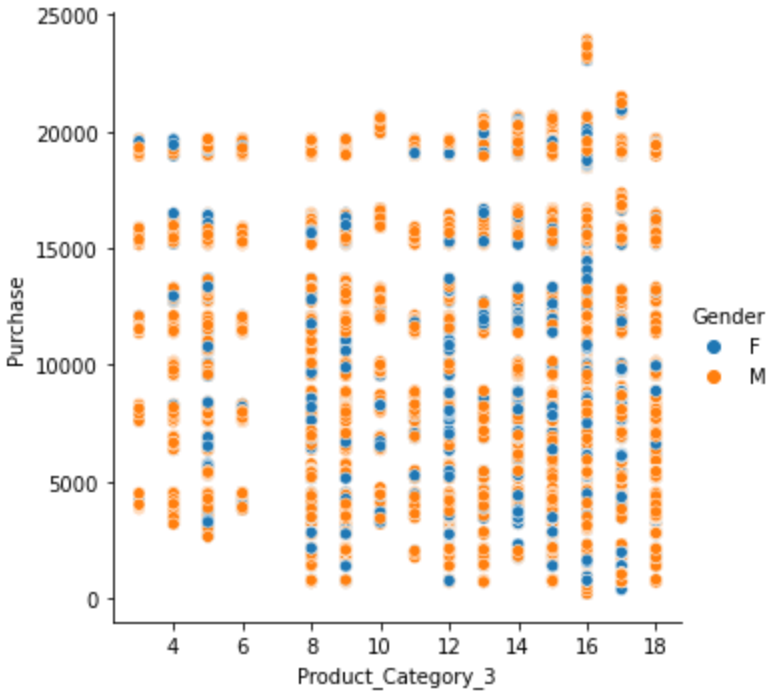
Most of the males were labeled with zero in marital status as compared to the females.



In product category one customers with label 12 and 13 contributed the least with purchase amounts less than 2500 and in them females had more frequency. Most of the males with the label 6 10 and 16 add purchase amount more than 20,000



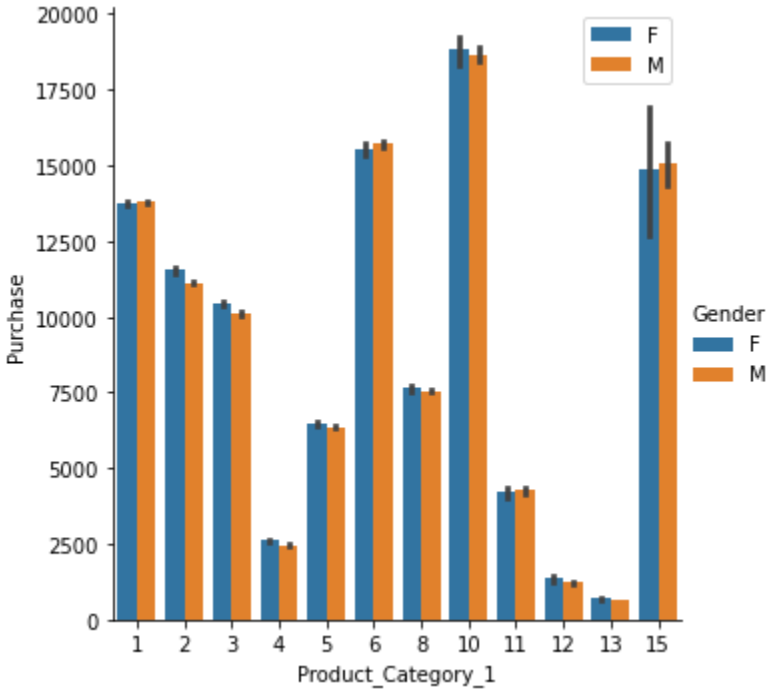
The flowers with product category 2 and labels 13,14 and 15 spent the most above 20,000.



Customers in product categorY 3 with the label 16 spent above 20,000 that is maximum amongst others.

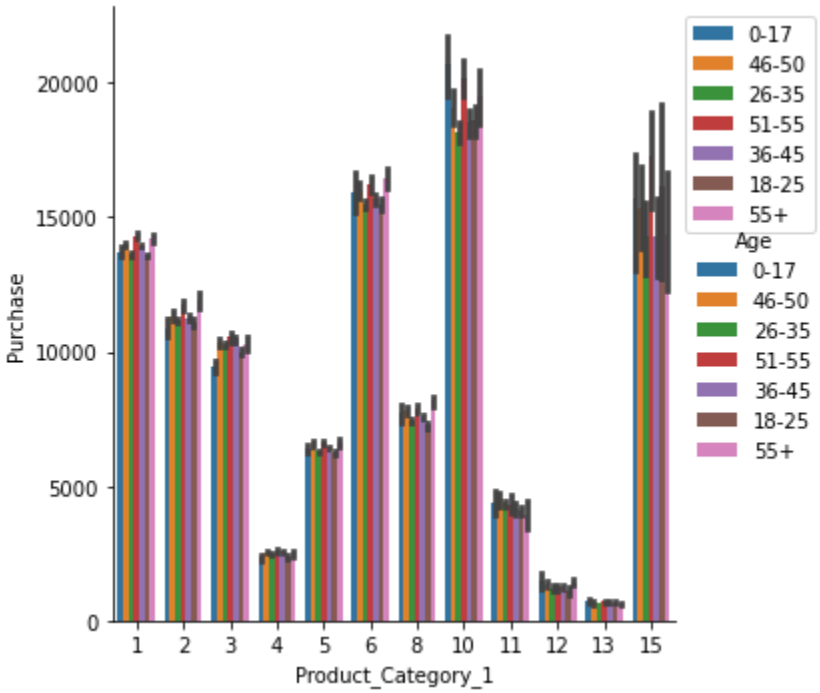
Hue analysis with Catplots for Product category 1:

For hue==> Gender

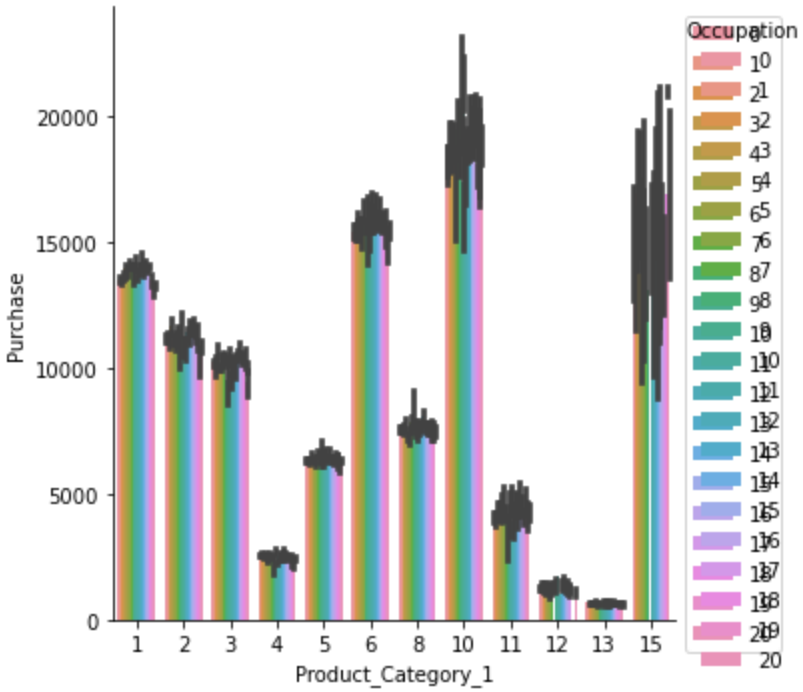


For customers in product category one with the labels 4, 12, 11 and 13 Spent least and purchases Of maximum amounts were done by customers labeled with 10 in product category one.

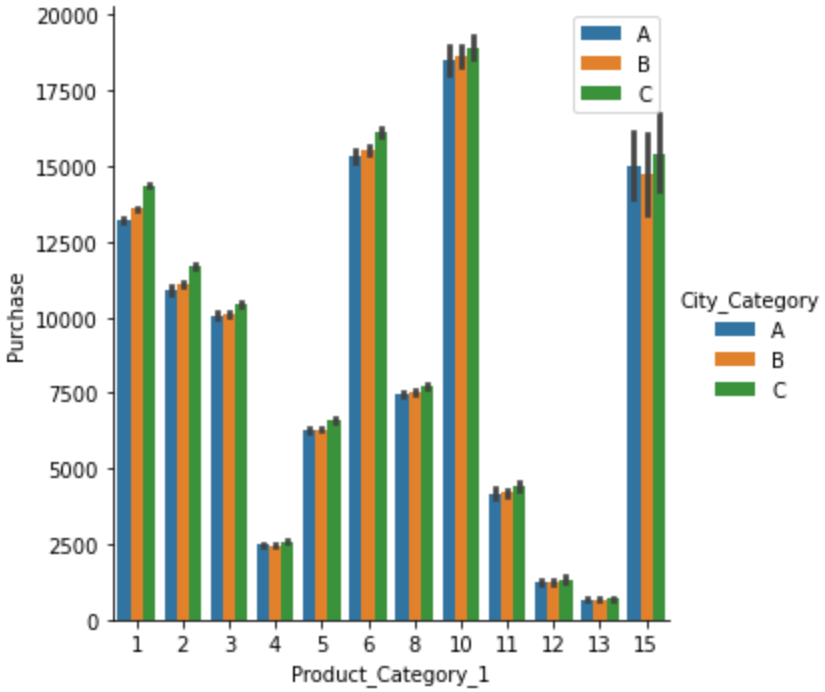
For hue==> Age



For hue==> Occupation

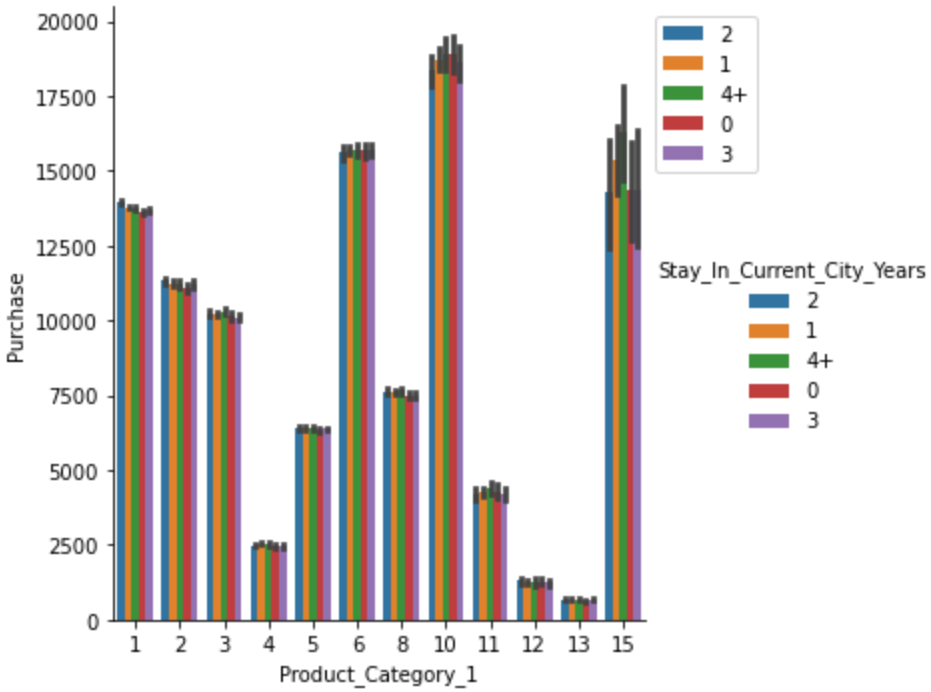


For hue==> City_Category

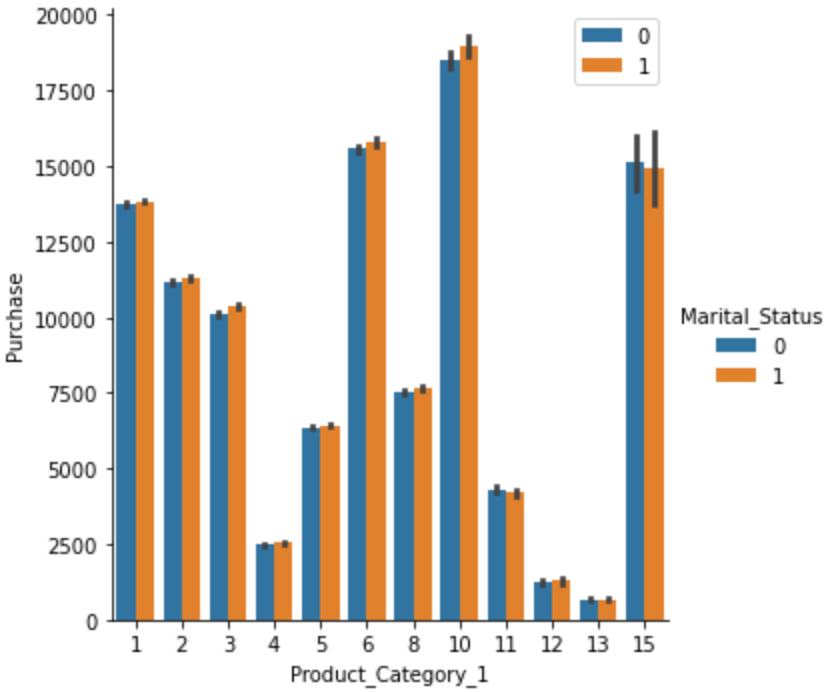


For hue==> Stay_In_Current_City_Years

Customers with city_category contributed more with maximum purchase amounts.



For hue==> Marital_Status

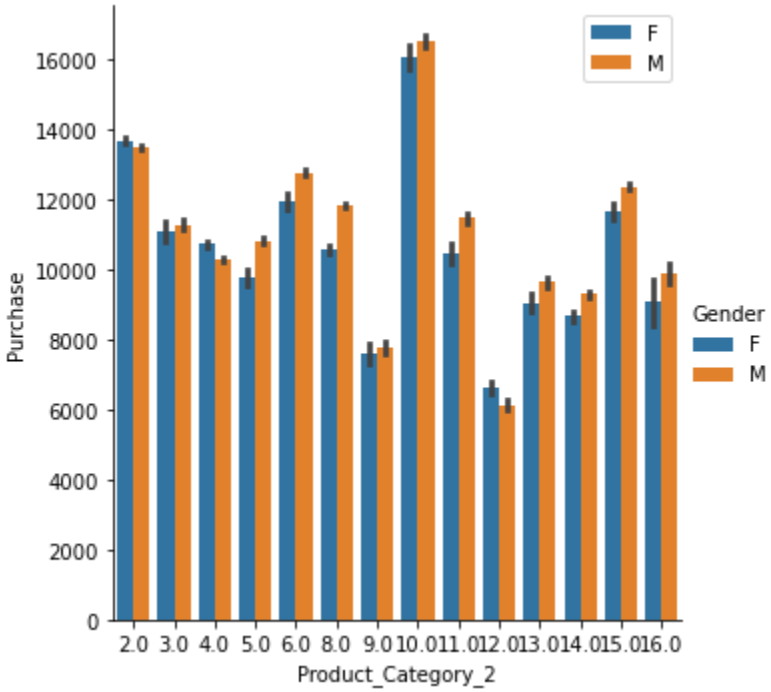


On an average customers with marital status hmm spent more during Black Friday.

Hue analysis with Catplot for Product category 2:

For hue==> Gender

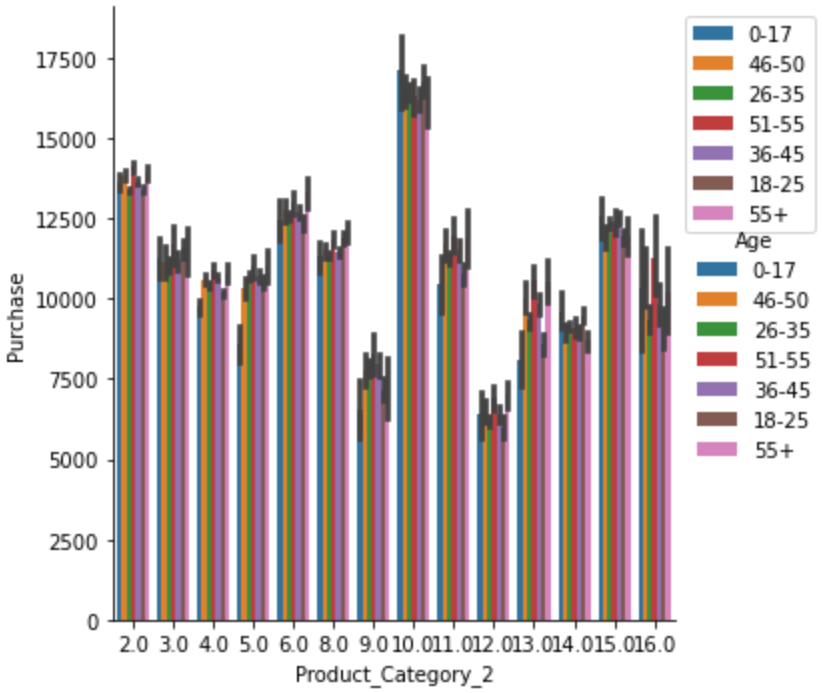
<Figure size 720x432 with 0 Axes>



Male customers in product category 2 in all labels spent more during Black Friday.

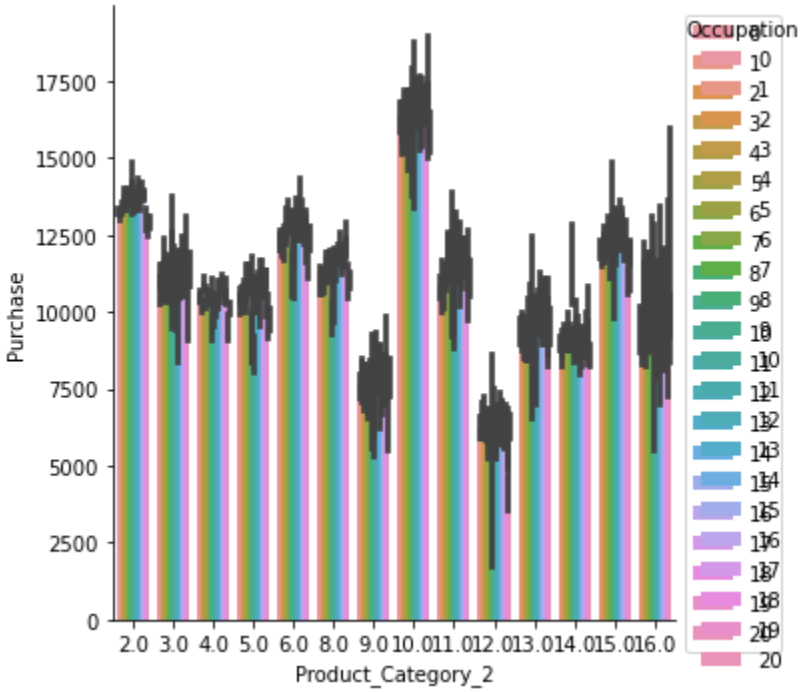
For hue==> Age

<Figure size 720x432 with 0 Axes>



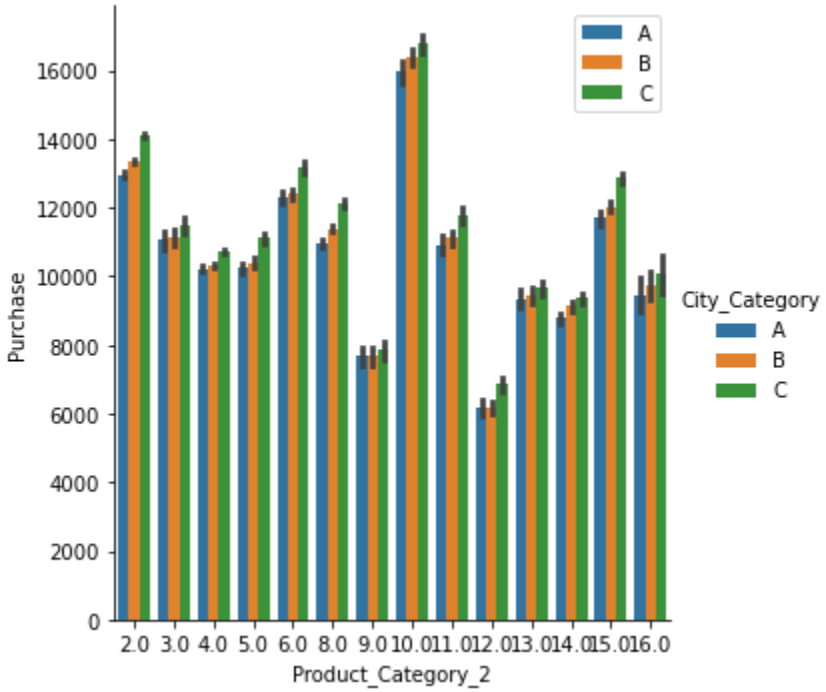
For hue==> Occupation

<Figure size 720x432 with 0 Axes>



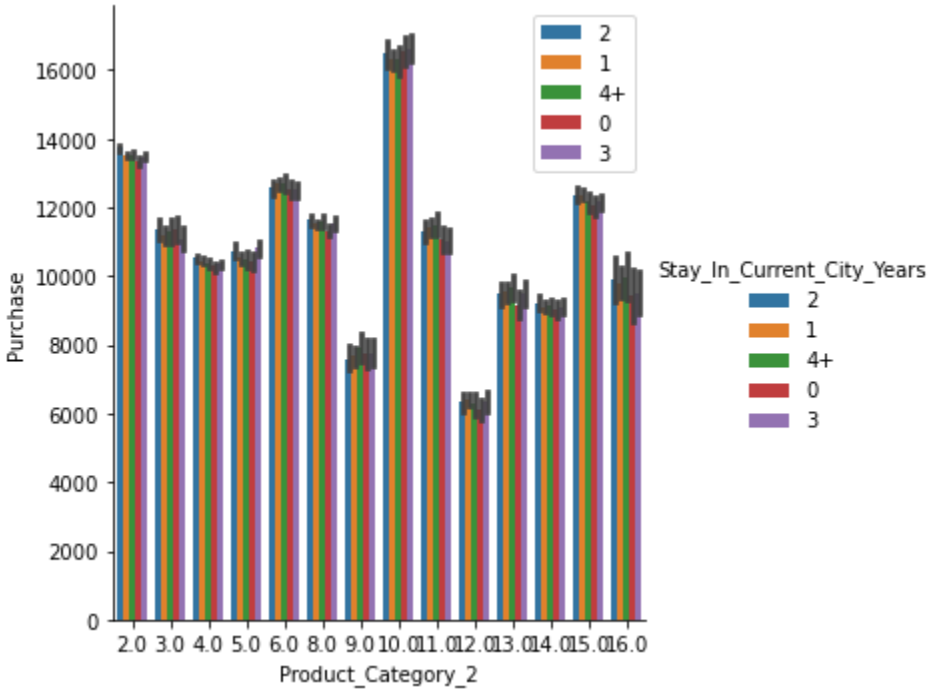
For hue==> City_Category

<Figure size 720x432 with 0 Axes>



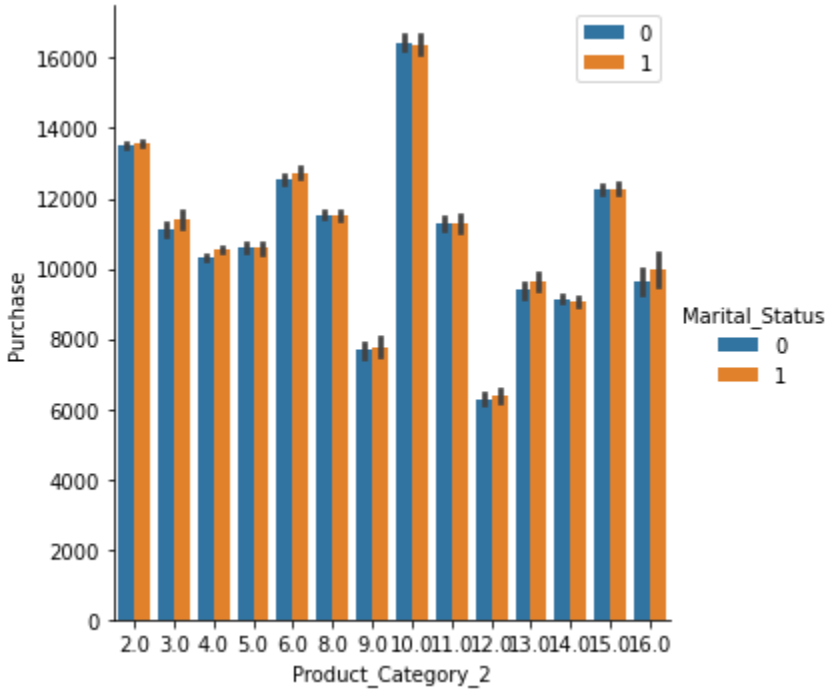
For hue==> Stay_In_Current_City_Years

<Figure size 720x432 with 0 Axes>



For hue==> Marital_Status

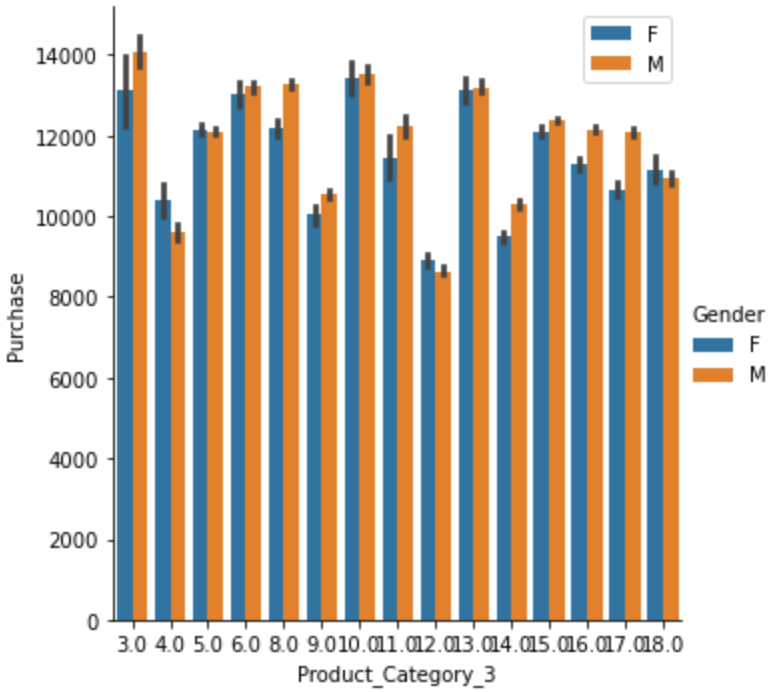
<Figure size 720x432 with 0 Axes>



Hue analysis with Catplot for product category 3:

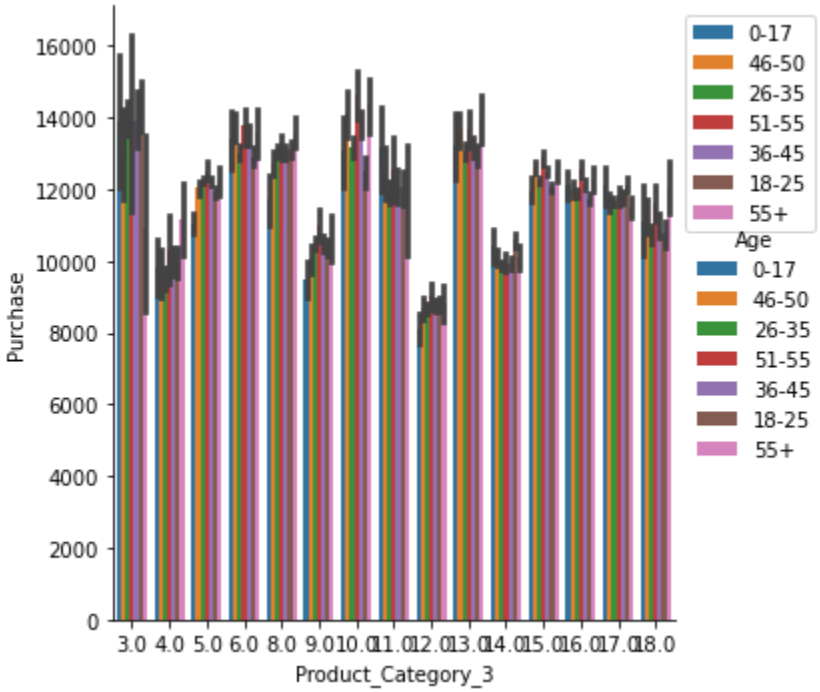
For hue==> Gender

<Figure size 720x432 with 0 Axes>



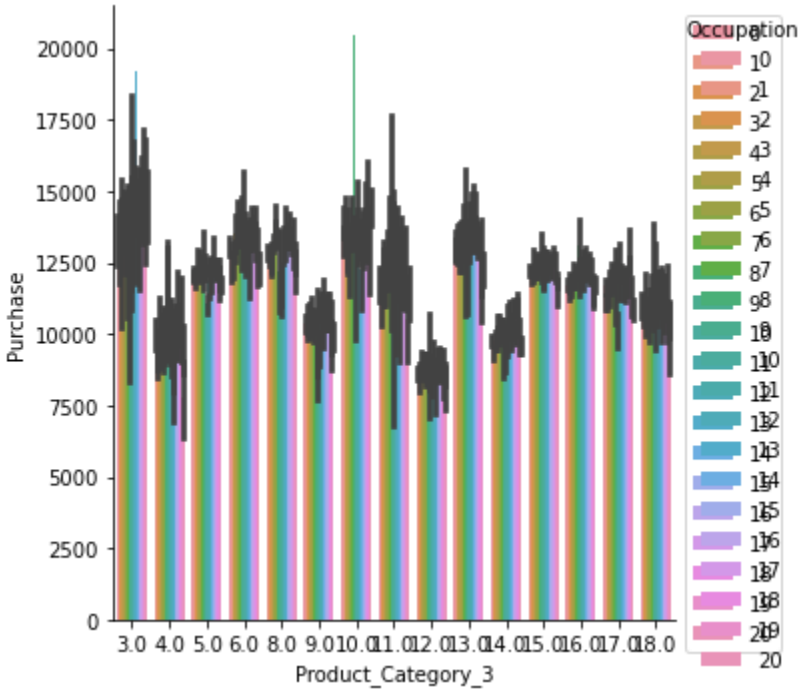
For hue==> Age

<Figure size 720x432 with 0 Axes>



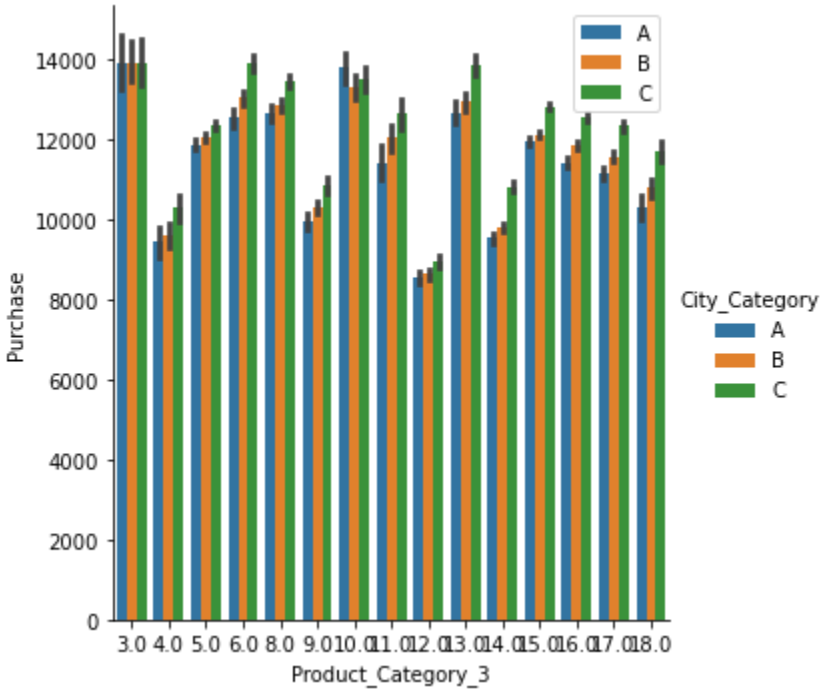
For hue==> Occupation

<Figure size 720x432 with 0 Axes>



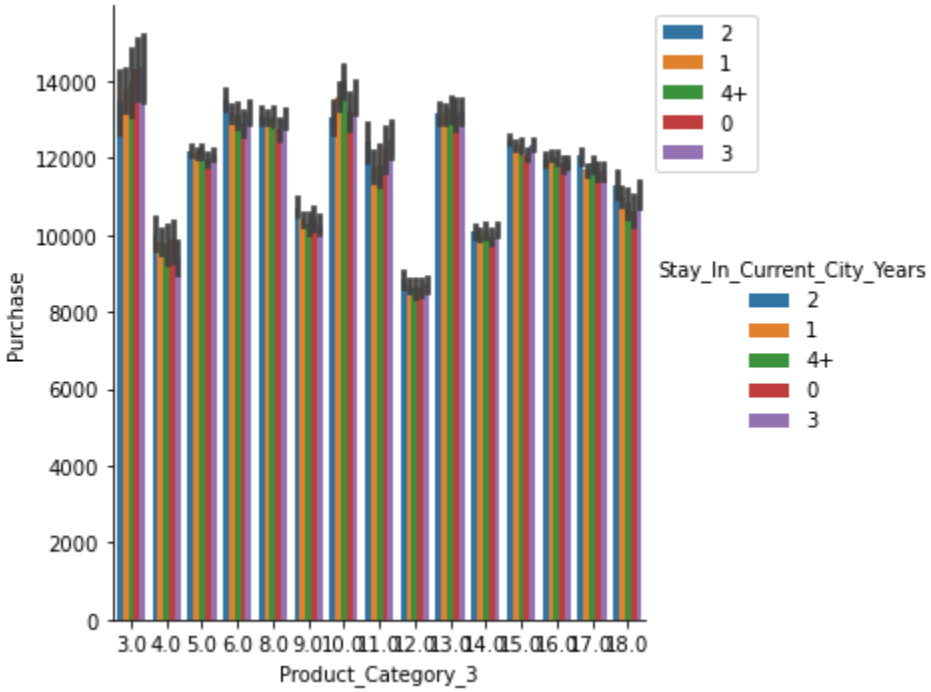
For hue==> City_Category

<Figure size 720x432 with 0 Axes>



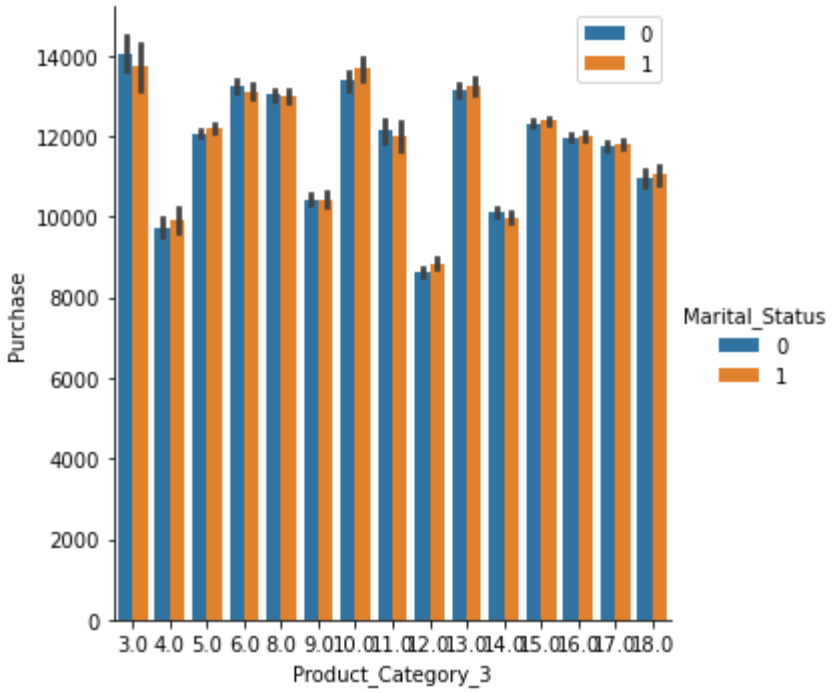
For hue==> Stay_In_Current_City_Years

<Figure size 720x432 with 0 Axes>

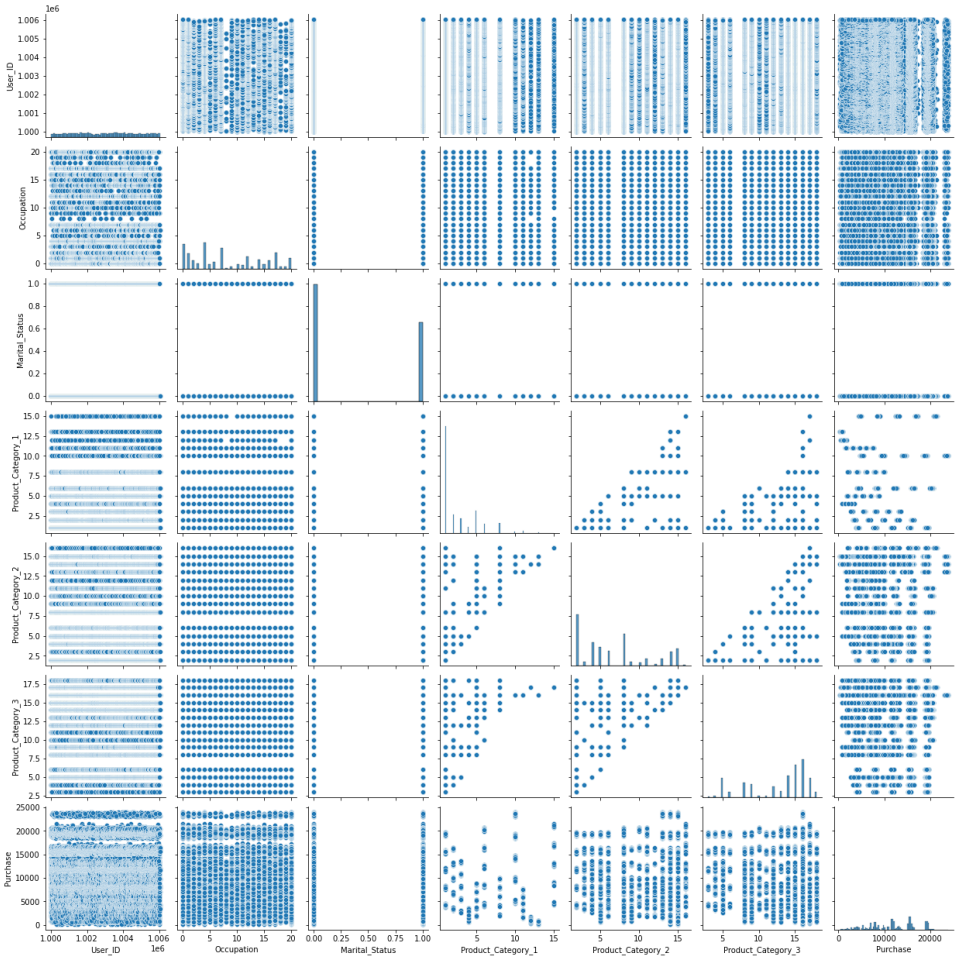


For hue==> Marital_Status

<Figure size 720x432 with 0 Axes>


0
1
14000
12000
10000
8000
6000
4000
2000
0
Purchase
3.0 4.0 5.0 6.0 8.0 9.0 10.0 11.0 12.0 13.0 14.0 15.0 16.0 17.0 18.0
Product_Category_3
Marital_Status
0
1

Pairplot:



Pair plot shows No sign of strong relation between features. As the data is pattered roughly.

Grouped Data Analysis

```
Gender_groups = df1.groupby('Gender')
```

```
Gender_groups['Purchase'].mean().sort_values(ascending=False)
```

```
Gender
M    11824.922756
F    11084.723786
Name: Purchase, dtype: float64
```

```
Age_groups = df1.groupby('Age')
```

```
Age_groups['Purchase'].mean().sort_values(ascending=False)
```

```
Age
51-55    12035.504299
55+      11861.524638
36-45    11729.364398
46-50    11663.978017
26-35    11612.248065
18-25    11580.858539
0-17     11172.358711
Name: Purchase, dtype: float64
```

```
City_Category_groups = df1.groupby('City_Category')
```

```
City_Category_groups['Purchase'].mean().sort_values(ascending=False
```

```
City_Category
C    12207.516991
B    11488.090522
A    11199.868783
Name: Purchase, dtype: float64
```

```
Stay_In_Current_City_Years_groups = df1.groupby('Stay_In_Current_City_Years')
Stay_In_Current_City_Years_groups['Purchase'].mean().sort_values(ascending=False)
```

```
Stay_In_Current_City_Years
2     11773.217230
4+    11695.298439
1     11652.926896
3     11630.236333
0     11503.925678
Name: Purchase, dtype: float64
```

```
# ['Gender', 'Age', 'City_Category','Stay_In_Current_City_Years']:
```

```
for i in ['Occupation', 'Marital_Status', 'Product_Category_1','Product_Category_2', 'Product_Category_3','Purchase']:
    for j in ['Gender', 'Age', 'City_Category','Stay_In_Current_City_Years']:
        print('Grouping by',j,'on',i,':\n')
        print(df1.groupby(j)[i].mean().sort_values(ascending=False))
        print('\n')
```

```
Grouping by Gender on Occupation :

Gender
M    8.570508
F    6.832713
Name: Occupation, dtype: float64


Grouping by Age on Occupation :

Age
55+      9.698721
51-55    9.043973
36-45    8.913264
```

Below, mean values are indicated for different groups:

Grouping by Occupation on Purchase:

Occupation	
17	**12142.126235**
12	**12112.541214**
15	**12022.898144**
14	**11984.760463**
16	**11885.026319**
7	**11878.530165**
8	**11867.909594**
5	**11849.773361**
18	**11811.815431**
13	**11806.197851**
6	**11782.109525**
11	**11701.880049**
4	**11583.881864**
3	**11580.922433**
0	**11476.844118**
2	**11321.644609**
1	**11216.963511**
10	**11201.459940**
20	**11170.454407**
19	**11034.046058**
9	**10931.098607**

Name: Purchase, dtype: float64

Grouping by Marital_Status on Purchase:

```
Marital_Status
1    11686.600295
0    11638.899065
Name: Purchase, dtype: float64
```

Grouping by Product_Category_1 on Purchase:

```
Product_Category_1
10    18690.721937
6     15673.219219
15    15037.178808
1     13768.781978
2     11229.584095
3     10213.919836
8      7574.562748
5      6385.485456
11     4267.861805
4      2507.065433
12     1269.094340
13      678.442235
Name: Purchase, dtype: float64
```

Grouping by Product_Category_2 on Purchase:

```
Product_Category_2
10.0    16413.823087
2.0     13530.118298
6.0     12616.938617
15.0    12257.943011
8.0     11528.251011
11.0    11294.279945
```

3.0 11235.359570
5.0 10592.127629
4.0 10416.845914
16.0 9754.296716
13.0 9483.936285
14.0 9121.196835
9.0 7720.844556
12.0 6320.329488
Name: Purchase, dtype: float64

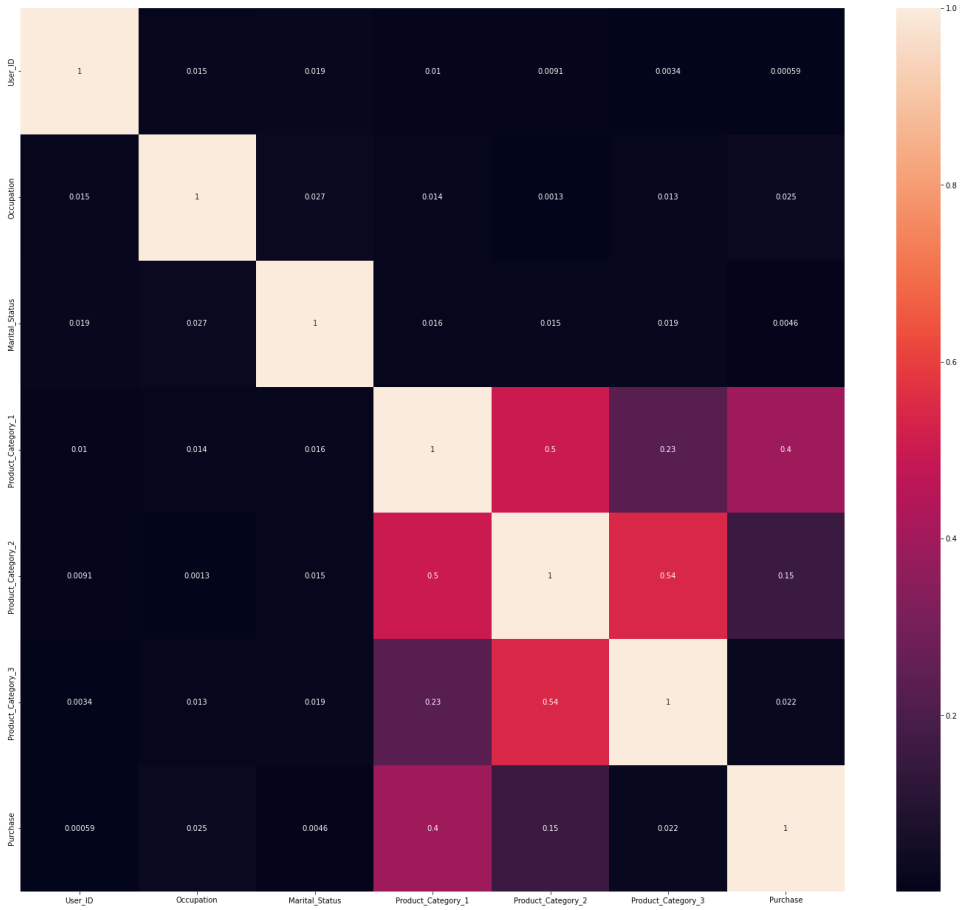
Grouping by Product_Category_3 on Purchase:

Product_Category_3
3.0 13939.696574
10.0 13505.813441
6.0 13194.311043
13.0 13185.118703
8.0 13024.918882
15.0 12339.369900
5.0 12117.786889
11.0 12091.437673
16.0 11981.890642
17.0 11769.943001
18.0 10993.980773
9.0 10431.697210
14.0 10052.594530
4.0 9794.386667
12.0 8715.512762
Name: Purchase, dtype: float64

Using group data analysis, we have come to a conclusion that occupation category 17, marital status category one, for product category 1 label 10, for product category 2 label 10 and product category 3 label 3 spent more during black friday sale. Whereas labels 13 and 12 spent the least.

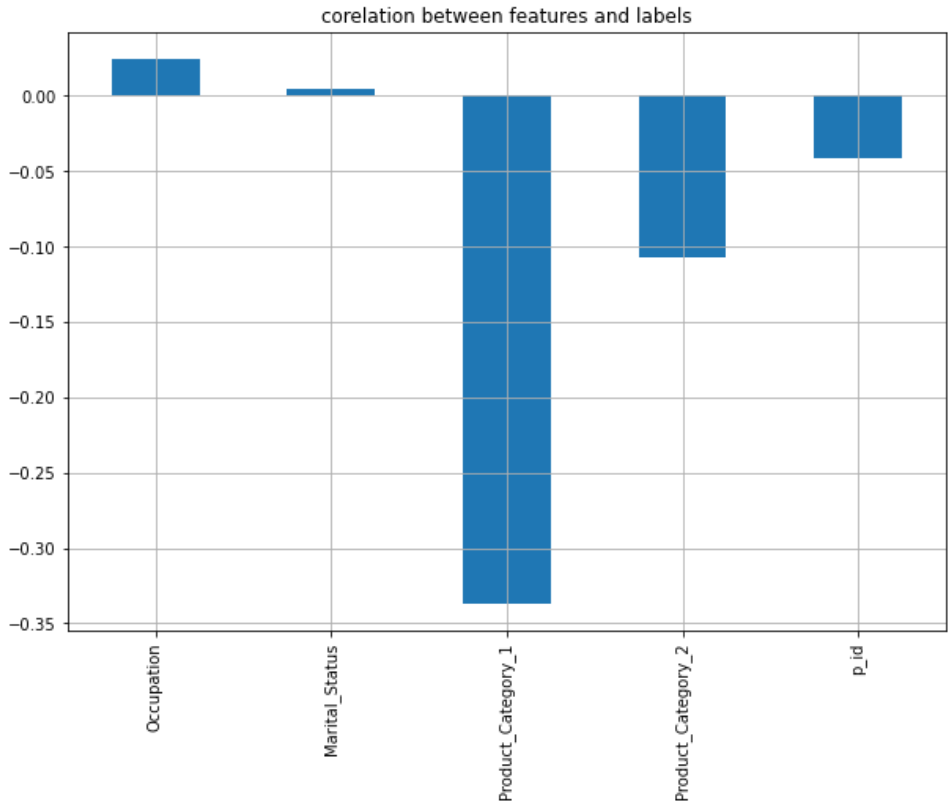
Correlation and Multicolinearity

- **Analysis of correlation using Corr and Heatmap:**



correlation and heat map show product category features have a problem of multicollinearity and that needs to be treated.

- **<u>Analysis of Correlation between features and label using Corrwith:</u>**



Occupation and marital status feature have slight positive correlation between the target that is the purchase amount.

- **Analysis of multicollinearity using Variance Inflation Factor:**

```
]: from statsmodels.stats.outliers_influence import variance_inflation_factor


vif = pd.DataFrame()
vif['vif'] = [variance_inflation_factor(df1[cont_columns],i) for i in range(df1[cont_columns].shape[1])]
vif['features'] = df1[cont_columns].columns
vif.sort_values(by='vif',ascending=False)
```

]:

	vif	features
0	22.823305	User_ID
5	14.918952	Product_Category_3
6	7.517701	Purchase
4	6.022380	Product_Category_2
7	3.581712	p_id
3	3.311656	Product_Category_1
1	2.593746	Occupation
2	1.677069	Marital_Status

We have got an idea that there is a presence of multicollinearity through the heatmap analysis so further we have done variance inflation factor analysis and with this we can firmly conclude that the columns user ID product category 3 purchase and product category two columns have multicollinearity issue.

Feature selection using Chi square test

```
# Using SelectKBest feature selection method: # It is one of the feature selection method:
# when there are lot of features and you cant graphically analyse , short way , selectKbest can be
# used:

from sklearn.feature_selection import SelectKBest, f_classif

# SelectKBest uses f_classif function to select best features where f_classif uses ANOVA test

# Using SelectKBest feature selection method: # It is one of the feature selection method:
# when there are lot of features and you cant graphically analyse , short way , selectKbest can be
# used:

from sklearn.feature_selection import SelectKBest, f_classif

# SelectKBest uses f_classif function to select best features where f_classif uses ANOVA test

best_features = SelectKBest(score_func = f_classif,k = 5)
fit = best_features.fit(X[cont_columns],y)

data_scores = pd.DataFrame(fit.scores_)
data_columns = pd.DataFrame(X[cont_columns].columns)

features_score = pd.concat([data_columns,data_scores],axis=1)

features_score.columns = ['Features','Scores']
print(features_score.nlargest(5,'Scores'),'\n') # print 5 best features

# Here we are getting top 5 features we got based on f_classify that uses ANOVA test of statistics.
```

```
             Features     Scores
2  Product_Category_1  27.240240
3  Product_Category_2   3.041683
4              p_id   1.366393
0         Occupation   1.027642
1     Marital_Status   0.983043
```

Since it is better to understand the feature importance in the analysis process, we have chosen the Chi Square test and found that the product category one feature is the most important in predicting the purchase amount of customers during Black Friday sale.

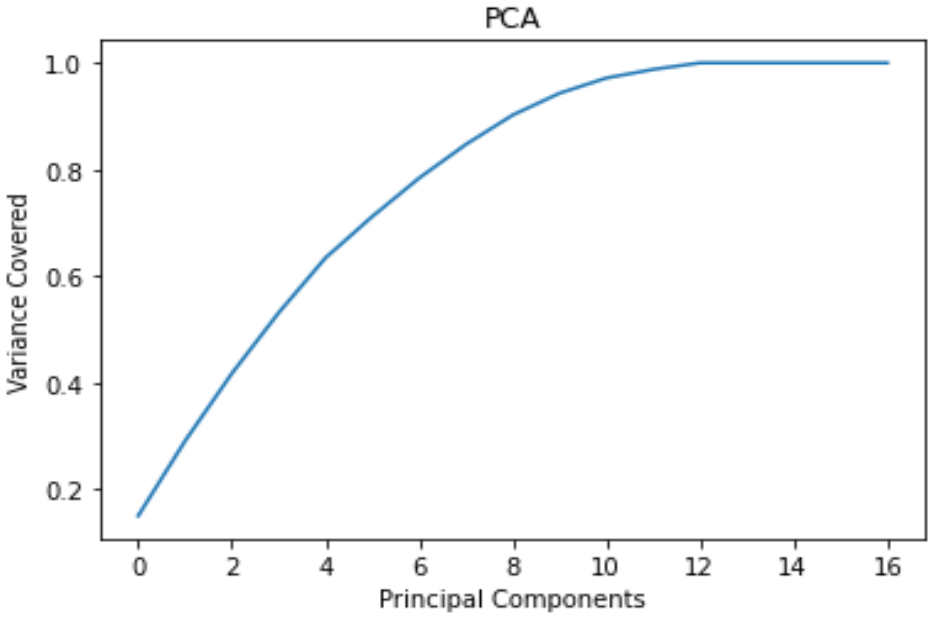
Data preprocessing and Principal Component Analysis

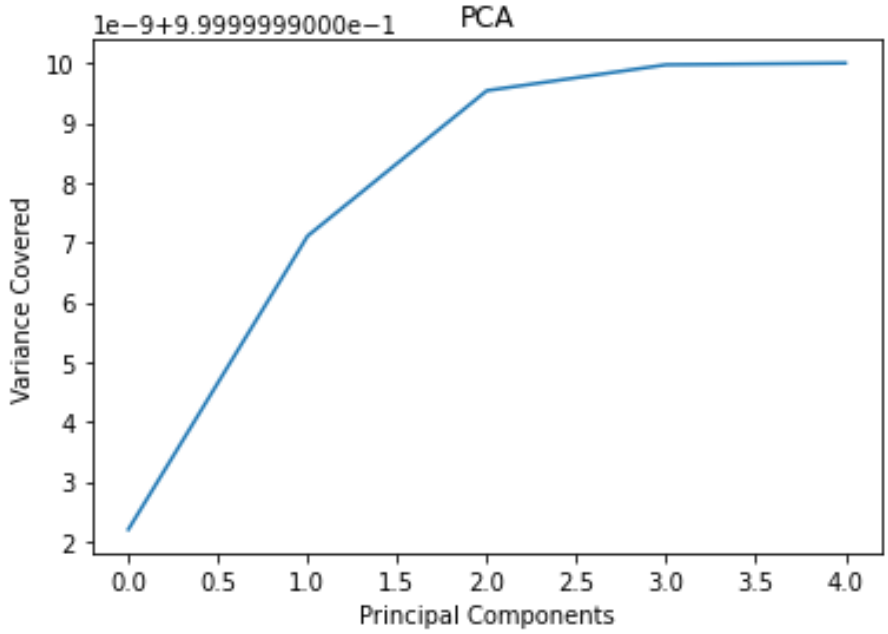
- **Converting categorical data to numerical data**
- **Selecting best components using PCA**
- **Scaling data for final model building**

```
X_cat = pd.get_dummies(X[cat_columns])
X_cat
```

	Gender_F	Gender_M	Age_0-17	Age_18-25	Age_26-35	Age_36-45	Age_46-50	Age_51-55	Age_55+	City_Category_A	City_Category_B	City_Category_C	Stay_In_C
1	1	0	1	0	0	0	0	0	0	1	0	0	
6	0	1	0	0	0	0	1	0	0	0	1	0	
13	0	1	0	0	1	0	0	0	0	1	0	0	
14	1	0	0	0	0	0	0	1	0	1	0	0	
16	1	0	0	0	0	0	0	1	0	1	0	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...
545902	1	0	0	0	0	0	1	0	0	0	1	0	
545904	0	1	0	0	1	0	0	0	0	0	1	0	
545907	0	1	0	0	1	0	0	0	0	0	1	0	
545908	0	1	0	0	1	0	0	0	0	0	1	0	
545914	0	1	0	0	1	0	0	0	0	0	1	0	

163494 rows × 17 columns





As our data it's a mix of categorical and continuous features so before proceeding with the model development process we are going to first convert our categorical features into continuous once using encoding techniques and further we will do the principal component analysis so that we can understand the components that are most responsible and important in covering the variance present in the data set and then we will select them and drop any unnecessary components and this will be done using principal component analysis and by plotting the scree plot.

So here we have plotted scree plot for both categorical features and continuous features separately and selected the best components for further analysis.

Model Building Stage

- **Algorithms used**

1. Linear regression
2. Ridge and lasso regression
3. decision tree regression
4. random forest regressor
5. xg boost regressor
6. ada boost regressor
7. support vector regressor
8. K neighbours regressor

Comparative Analysis

Comparative Analysis The comparison between the MSE rates of all algorithms is depicted in Table below. Based on below data it can be observed that Random Forest Regressor gives better performance with comparison to other machine learning models namely linear regression and Decision tree regressor. The RMSE rate of XG Boost Regressor is 3548 and hence it is more suitable for the prediction model to be implemented.

Model MSE Linear Regression 4617.99

Ridge Regression 4687.75

Lasso Regression 4694.14

Decision Tree Regressor 3363.87

Random Forest Regressor 3548

XG Boost 3258

AdaBoost 2538

KNN Model 2703

Models (Best)

Random Forest Regressor

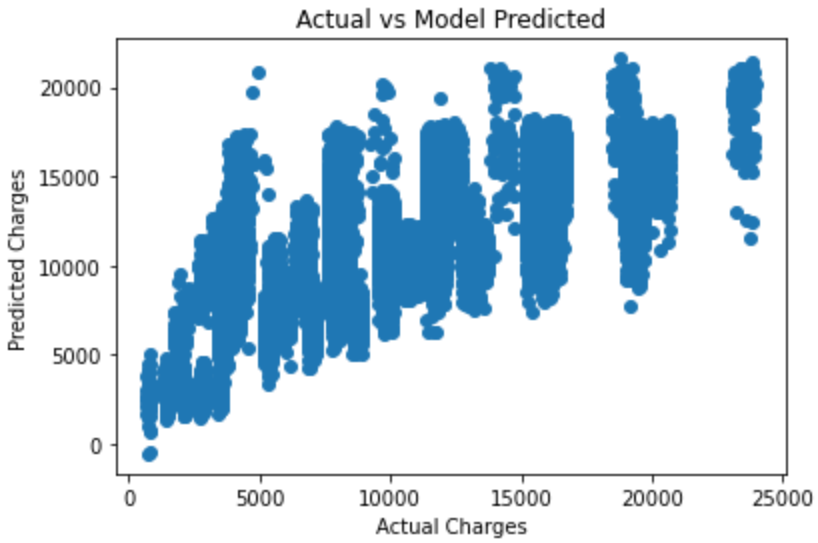
Training result: 88.77343014511078

MAE: 2713.0085636438903
MSE: 12589976.860212602
RMSE: 3548.2357390980383
R2: 0.49153417116195774
Score: 1.0

XG Boost Regressor

Training score: 0.6028498580163906
R2 score: 0.5711187448028863

MAE: 2538.0027214224583
MSE: 10619406.010134248
RMSE: 3258.7430107534174
R2: 0.5711187448028863
Score: 1.0



Knearest neighbors regressor model is also to understand our training data well with an accuracy of 66 as compared to the other models but that testing accuracy is less that indicates that the model is overfitting so there is high hope that doing hyperparameter tuning we'll definitely improve the accuracy off knn model.

Cross Validation

```
from sklearn.model_selection import KFold , cross_val_score
k_f = KFold(n_splits =3)
k_f
```

```
KFold(n_splits=3, random_state=None, shuffle=False)
```

```
for train,test in k_f.split([1,2,3,4,5,6,7,8,9,10]):
    print('train:',train,'test:',test)
```

```
train: [4 5 6 7 8 9] test: [0 1 2 3]
train: [0 1 2 3 7 8 9] test: [4 5 6]
train: [0 1 2 3 4 5 6] test: [7 8 9]
```

```
# Cross Validation score to check if model is overfitting:

cross_val_score(xgb_reg,X_best,y,cv=5) #
```

```
array([0.56283492, 0.56625692, 0.56549797, 0.557862  , 0.55477552])
```

```
cross_val_score(xgb_reg,X_best,y,cv=5).mean()
```

```
0.5614454637984495
```

We have got uniform cross validation scores for different samples thus we can conclude that our model is performing well and is not overfitting or underfitting.

Pipelining

Pipeline helps to streamline and speed up the process by automating these workflows and linking them together.

```
from sklearn.pipeline import Pipeline

pipe1 = Pipeline([('pca',PCA(n_components=17)),('base_model1',xgb.XGBRegressor(n_estimators=200,learning_rate=0.1))])

pipe1.fit(x_train,y_train)

y_predx = xgb_reg.predict(x_train)
y_pred = pipe1.predict(x_test)

print('Training accuracy: ', r2_score(y_train,y_predx)*100)
print('Testing accuracy: ', r2_score(y_test,y_pred)*100)
print("MAE: ", metrics.mean_absolute_error(y_test, y_pred))
print("MSE: ", metrics.mean_squared_error(y_test, y_pred))
print("RMSE: ", metrics.mean_squared_error(y_test, y_pred, squared=False))
print("R2: ", metrics.r2_score(y_test, y_pred), "\n")
print("Score: ", pipe1.score(x_test, y_pred))

# Saving regression model to pickle string

import pickle
saved_model1 = pickle.dumps(pipe1)
pipe_pickle1 = pickle.loads(saved_model1)
pipe_pickle1.predict(x_test) # predicting testing data
```

```
Training accuracy:  60.28498580163906
Testing accuracy:  56.62258985314824
MAE:  2559.1852021096165
MSE:  10740556.376282398
RMSE:  3277.2788066141698
R2:  0.5662258985314824

Score:  1.0
```

Prediction of New Data:

Finally, we have used our model for predicting new unseen data. for this same process was followed, first rose containing missing values were removed and then data pre-

processing was done. at the end we used our xg boost model to predict purchase amount of customers during Black Friday sale.

```
df_test
```

	Gender	Age	Occupation	City_Category	Stay_In_Current_City_Years	Marital_Status	Product_Category_1	Product_Category_2	p_id	Purchase
4	F	26-35	1	C	1	0	4	5.0	53842	2875.656494
5	M	46-50	1	C	3	1	2	3.0	350442	11581.344727
6	M	46-50	1	C	3	1	1	11.0	155442	11933.896484
7	M	46-50	1	C	3	1	2	4.0	94542	11371.570312
8	M	26-35	7	A	1	0	10	13.0	161842	13001.013672
...	...	...	...	...	...	...	...	...	...	...
233584	M	26-35	0	C	2	1	1	11.0	262242	12299.335938
233586	M	36-45	6	C	1	1	1	2.0	110742	16578.753906
233588	M	26-35	17	C	1	1	6	8.0	129842	14053.278320
233591	M	51-55	13	B	1	1	1	2.0	127642	13567.117188
233596	F	26-35	15	B	4+	1	1	5.0	31842	11816.208008

69677 rows × 10 columns

CONCLUSION

With traditional methods not being of much help to business growth in terms of revenue, the use of Machine learning approaches proves to be an important point for the shaping of the business plan taking into consideration the shopping pattern of consumers. Projection of sales concerning several factors including the sale of last year helps businesses take on suitable strategies for increasing the sales of goods that are in demand. The models used are Linear Regression, Lasso Regression, Ridge Regression, Decision Tree Regressor, and Random Forest Regressor. The evaluation measure used is Mean Squared Error (RMSE). XG Boost Regressor is best suitable for the prediction of sales based on a given dataset. Thus, the proposed model will predict the customer purchase on Black Friday and give the retailer insight into customer choice of products. This will result in a discount based on customer-centric choices thus increasing the profit to the retailer as well as the customer

Future Work

As future research, we can perform hyperparameter tuning and apply different machine learning algorithms.

REFERENCES

- www.kaggle.com
- www.Sklearn.com
- www.researchgate.com