BLACK FRIDAY SALES PREDICTION

CSP 571

Submitted By :-
Radhika Malhotra (A20491601)
Aman Singh (A20491333)
Aniruddh Suresh Pillai (A20488146)

Exploratory Data Analysis

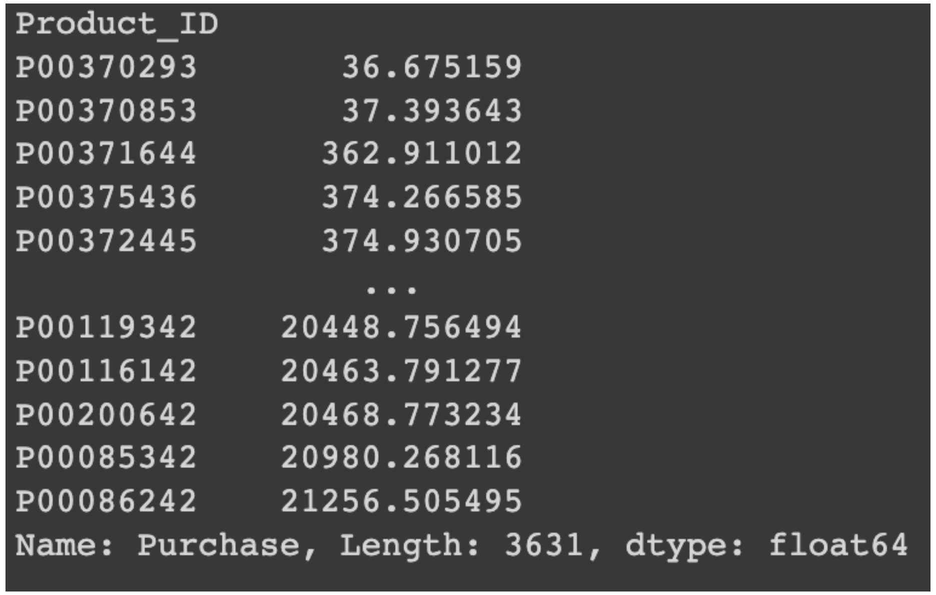
Exploratory Data Analysis (EDA) is the critical process of doing early investigations on data utilizing summary statistics and graphical representations in order to identify patterns, detect anomalies, test hypotheses, and check assumptions.

In a nutshell, EDA is a data exploration approach used to comprehend the many features of the data. EDA is frequently used to learn more about the variables in a data collection and how they interact outside of formal modeling. It may also assist us in determining if the statistical approaches we are contemplating for data analysis are adequate. It provides insight into all of the data and the multiple relationships between the data parts prior to modeling the data.

Objectives of performing EDA:

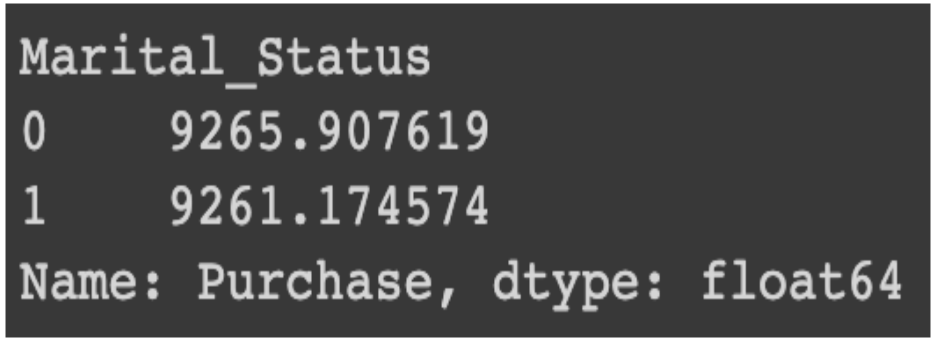
- ❑ Outliers list
- ❑ Parameter estimations
- ❑ Uncertainties about those estimations
- ❑ List of all significant factors
- ❑ Conclusions or assumptions about whether or not certain individual factors are statistically significant
- ❑ Optimal configurations
- ❑ A good forecasting

GROUPBY() is a very powerful function with a lot of variations. It makes it incredibly simple and effective to divide the data frame among a few criteria. It helps to aggregate the data efficiently.



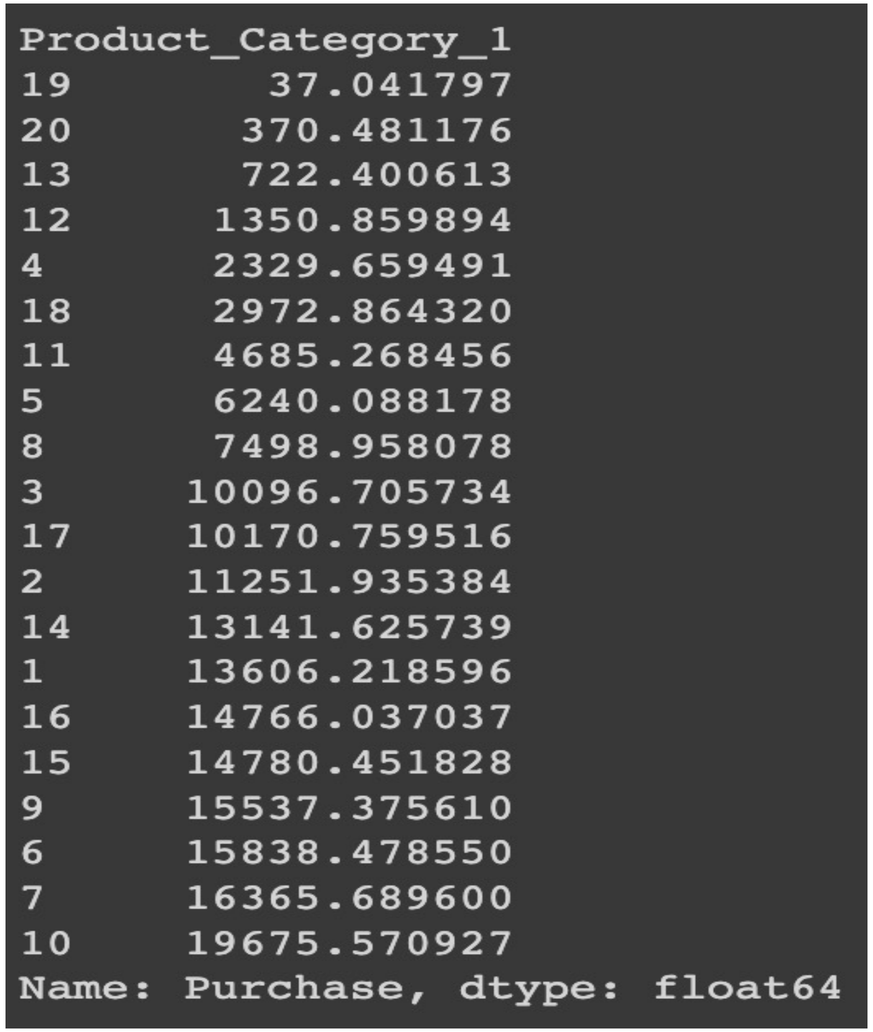
```
Product_ID
P00370293        36.675159
P00370853        37.393643
P00371644       362.911012
P00375436       374.266585
P00372445       374.930705
                   ...
P00119342     20448.756494
P00116142     20463.791277
P00200642     20468.773234
P00085342     20980.268116
P00086242     21256.505495
Name: Purchase, Length: 3631, dtype: float64
```

Grouping by the product ids' purchase amount (mean purchase)



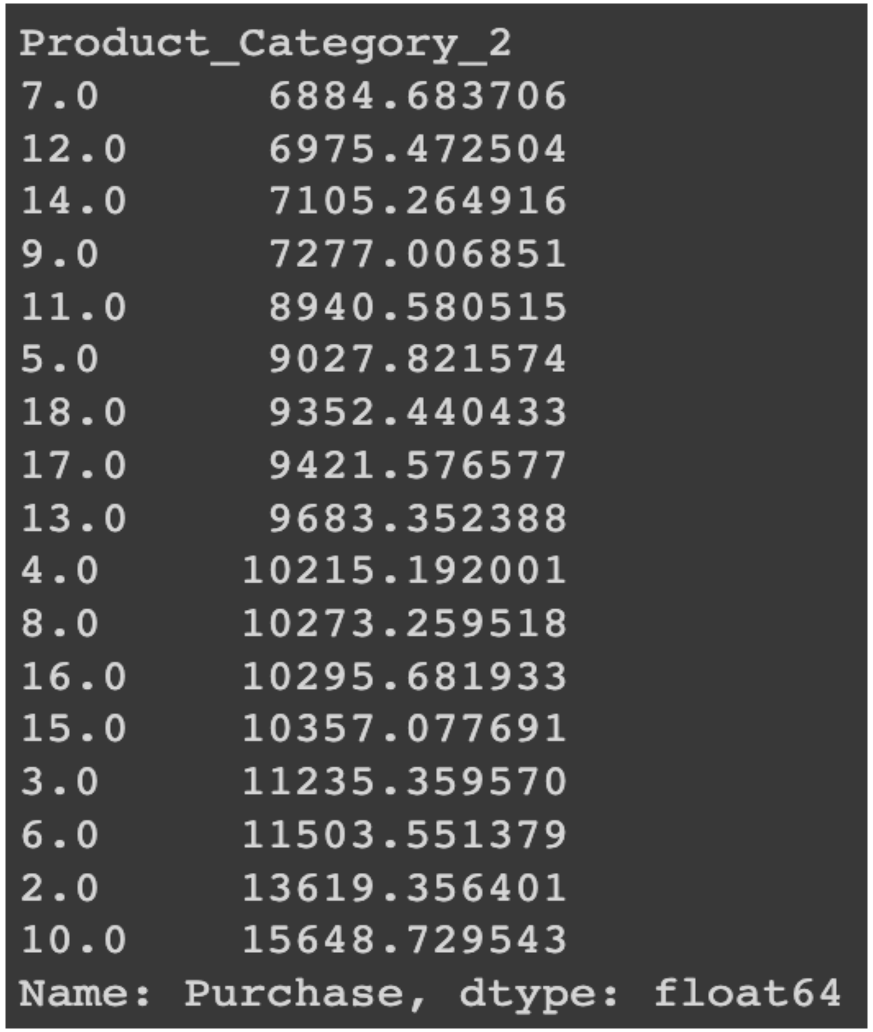
```
Marital_Status
0    9265.907619
1    9261.174574
Name: Purchase, dtype: float64
```

Grouping by marital status and their purchase



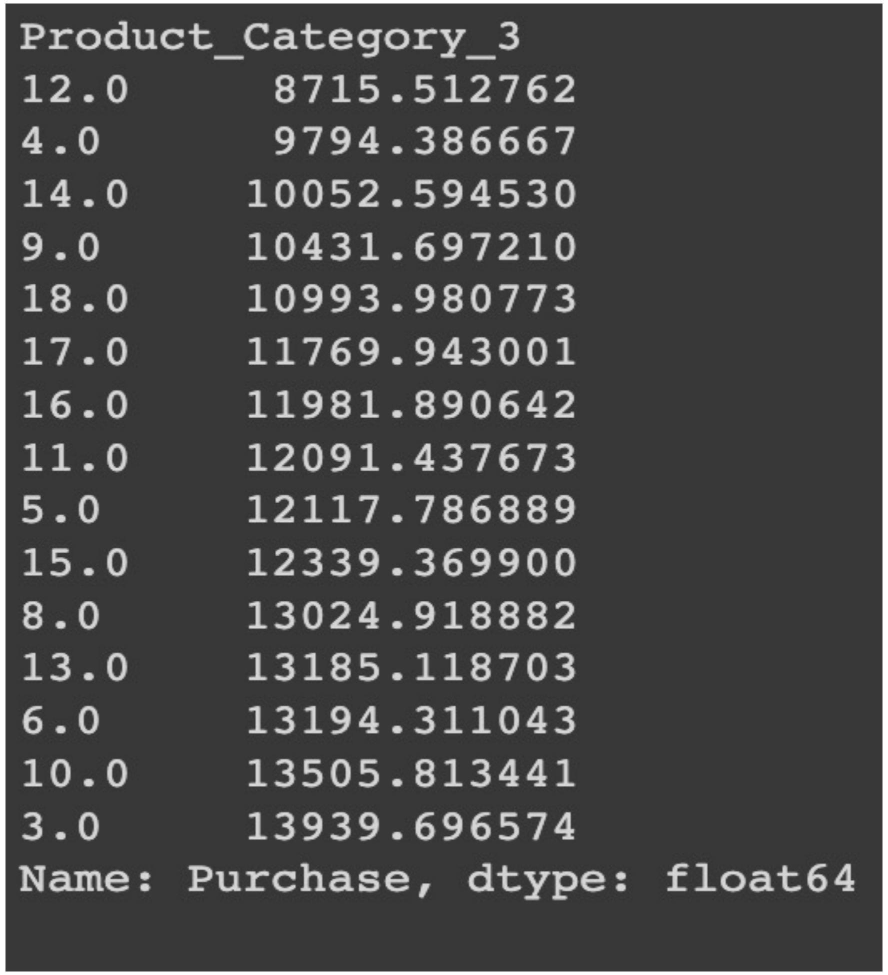
```
Product_Category_1
19          37.041797
20         370.481176
13         722.400613
12        1350.859894
4         2329.659491
18        2972.864320
11        4685.268456
5         6240.088178
8         7498.958078
3        10096.705734
17       10170.759516
2        11251.935384
14       13141.625739
1        13606.218596
16       14766.037037
15       14780.451828
9        15537.375610
6        15838.478550
7        16365.689600
10       19675.570927
Name: Purchase, dtype: float64
```

Grouping by product category 1 and their purchase



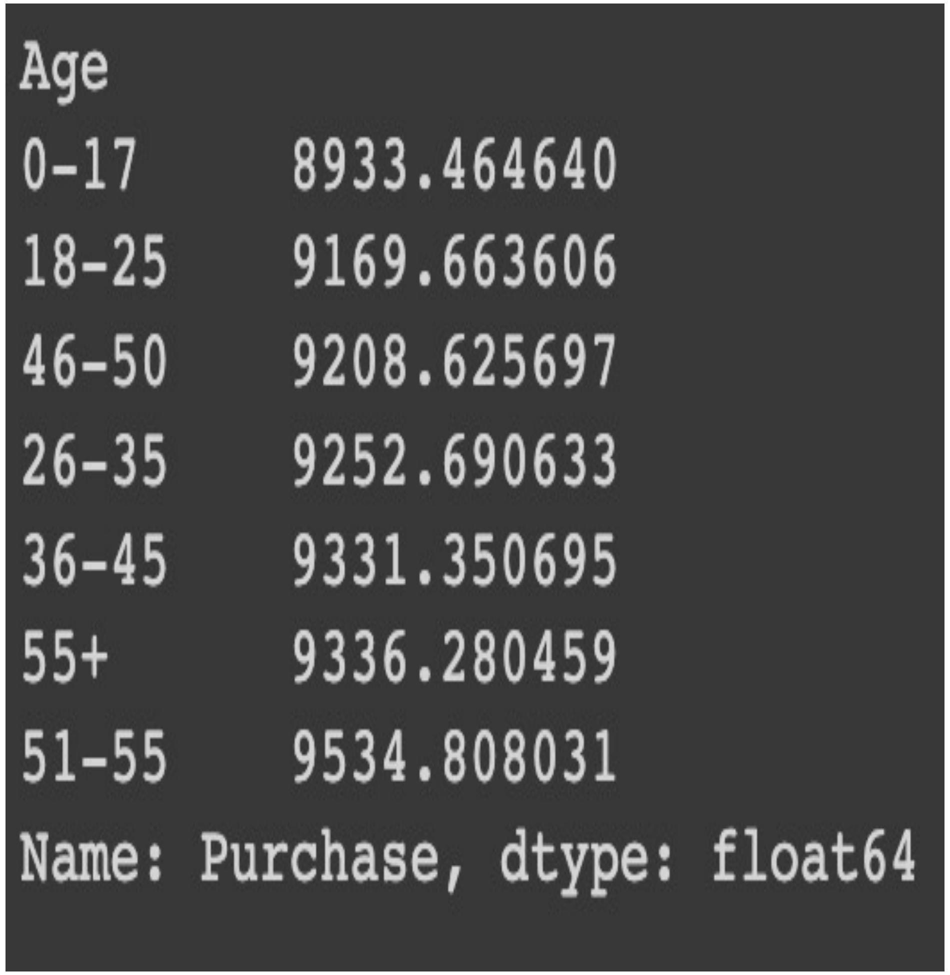
```
Product_Category_2
7.0          6884.683706
12.0         6975.472504
14.0         7105.264916
9.0          7277.006851
11.0         8940.580515
5.0          9027.821574
18.0         9352.440433
17.0         9421.576577
13.0         9683.352388
4.0         10215.192001
8.0         10273.259518
16.0        10295.681933
15.0        10357.077691
3.0         11235.359570
6.0         11503.551379
2.0         13619.356401
10.0        15648.729543
Name: Purchase, dtype: float64
```

Grouping by product category 2 and their purchase



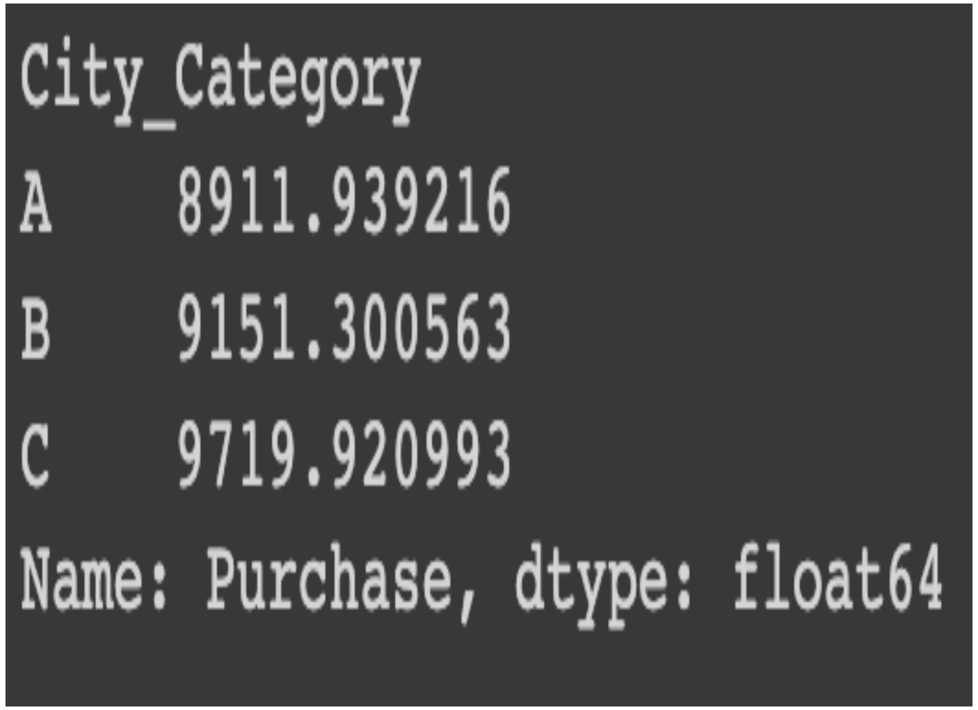
```
Product_Category_3
12.0       8715.512762
4.0        9794.386667
14.0      10052.594530
9.0       10431.697210
18.0      10993.980773
17.0      11769.943001
16.0      11981.890642
11.0      12091.437673
5.0       12117.786889
15.0      12339.369900
8.0       13024.918882
13.0      13185.118703
6.0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

Grouping by product category 3 and their purchase



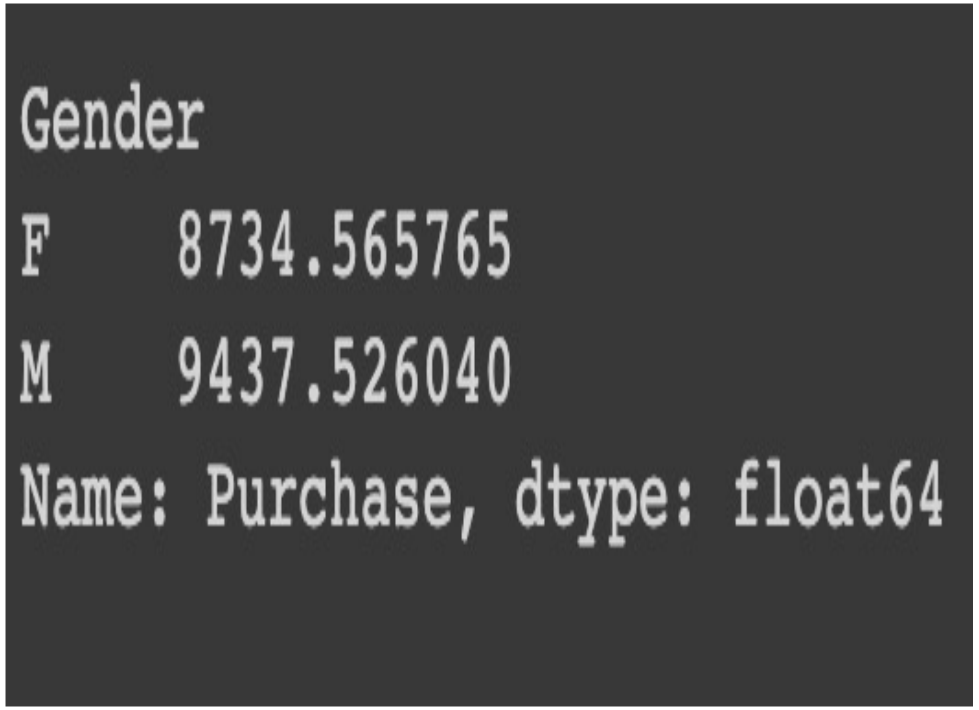
```
Age
0-17     8933.464640
18-25    9169.663606
46-50    9208.625697
26-35    9252.690633
36-45    9331.350695
55+      9336.280459
51-55    9534.808031
Name: Purchase, dtype: float64
```

Grouping by age and look their purchase



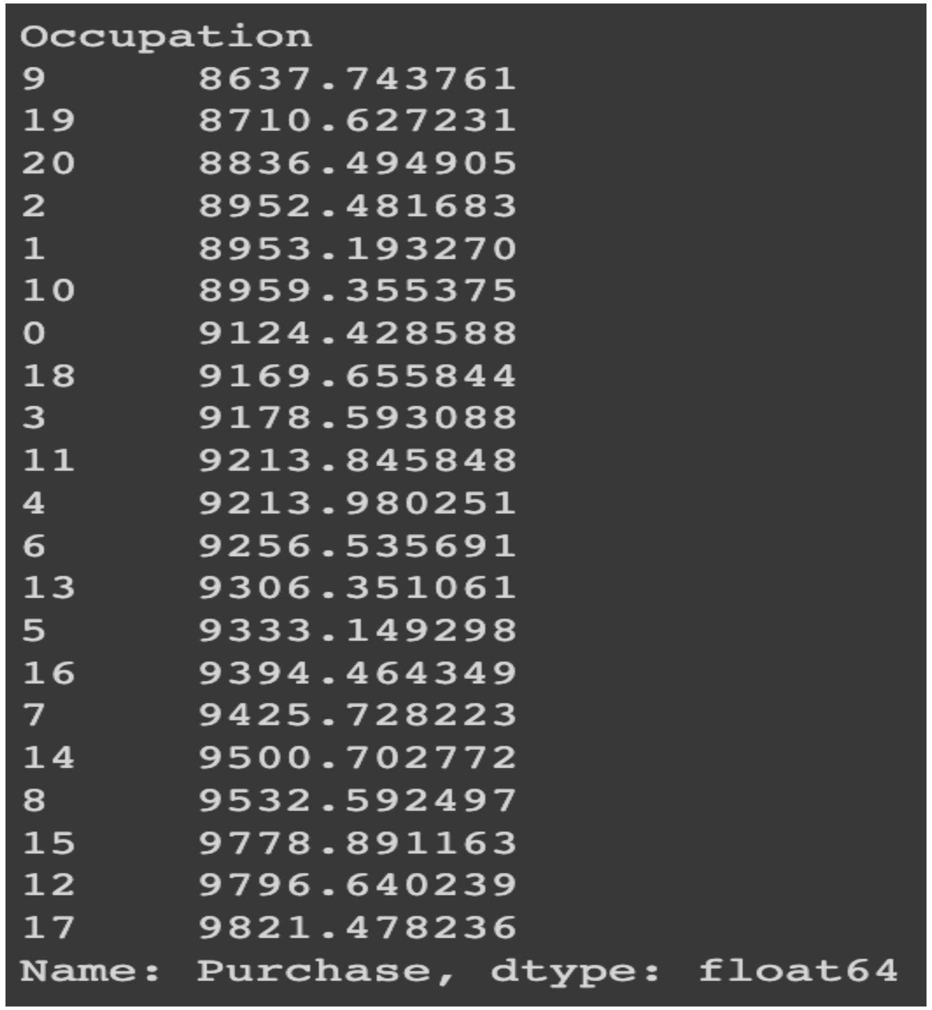
```
City_Category
A    8911.939216
B    9151.300563
C    9719.920993
Name: Purchase, dtype: float64
```

Group by their city category and look their purchase



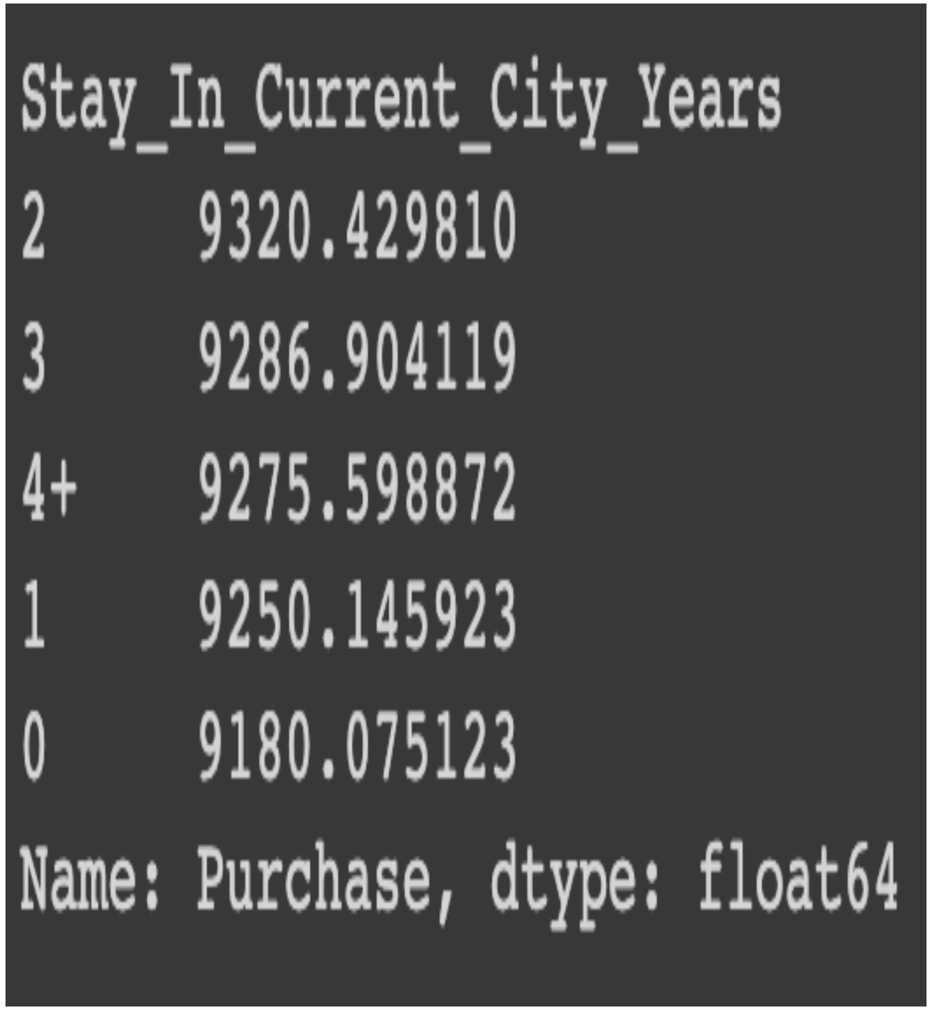
```
Gender
F    8734.565765
M    9437.526040
Name: Purchase, dtype: float64
```

Grouping by gender and looking at purchase



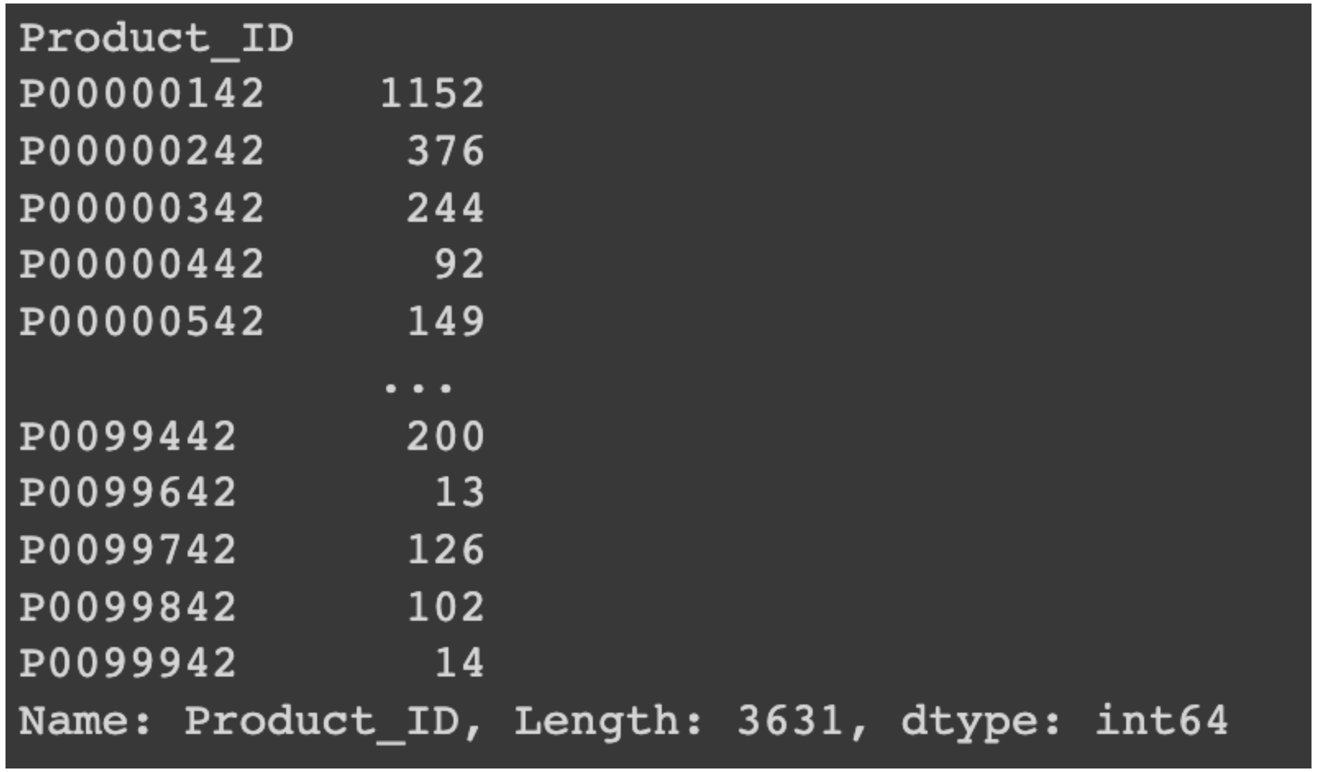
```
Occupation
9       8637.743761
19      8710.627231
20      8836.494905
2       8952.481683
1       8953.193270
10      8959.355375
0       9124.428588
18      9169.655844
3       9178.593088
11      9213.845848
4       9213.980251
6       9256.535691
13      9306.351061
5       9333.149298
16      9394.464349
7       9425.728223
14      9500.702772
8       9532.592497
15      9778.891163
12      9796.640239
17      9821.478236
Name: Purchase, dtype: float64
```

Grouping by occupation and looking at purchase



```
Stay_In_Current_City_Years
2     9320.429810
3     9286.904119
4+    9275.598872
1     9250.145923
0     9180.075123
Name: Purchase, dtype: float64
```

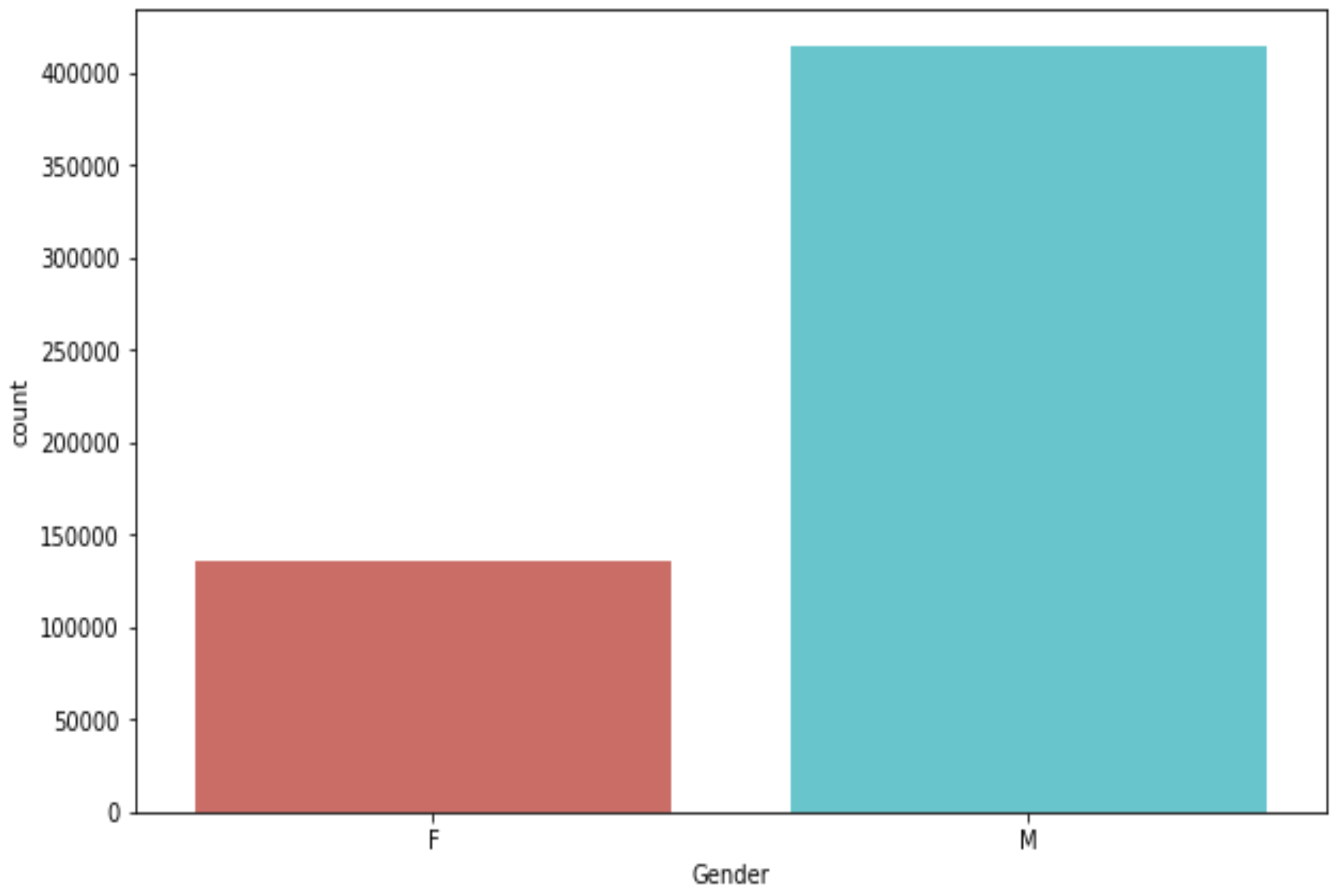
Grouping by stay in city and looking at purchase



```
Product_ID
P00000142    1152
P00000242     376
P00000342     244
P00000442      92
P00000542     149
             ...
P0099442      200
P0099642       13
P0099742      126
P0099842      102
P0099942       14
Name: Product_ID, Length: 3631, dtype: int64
```

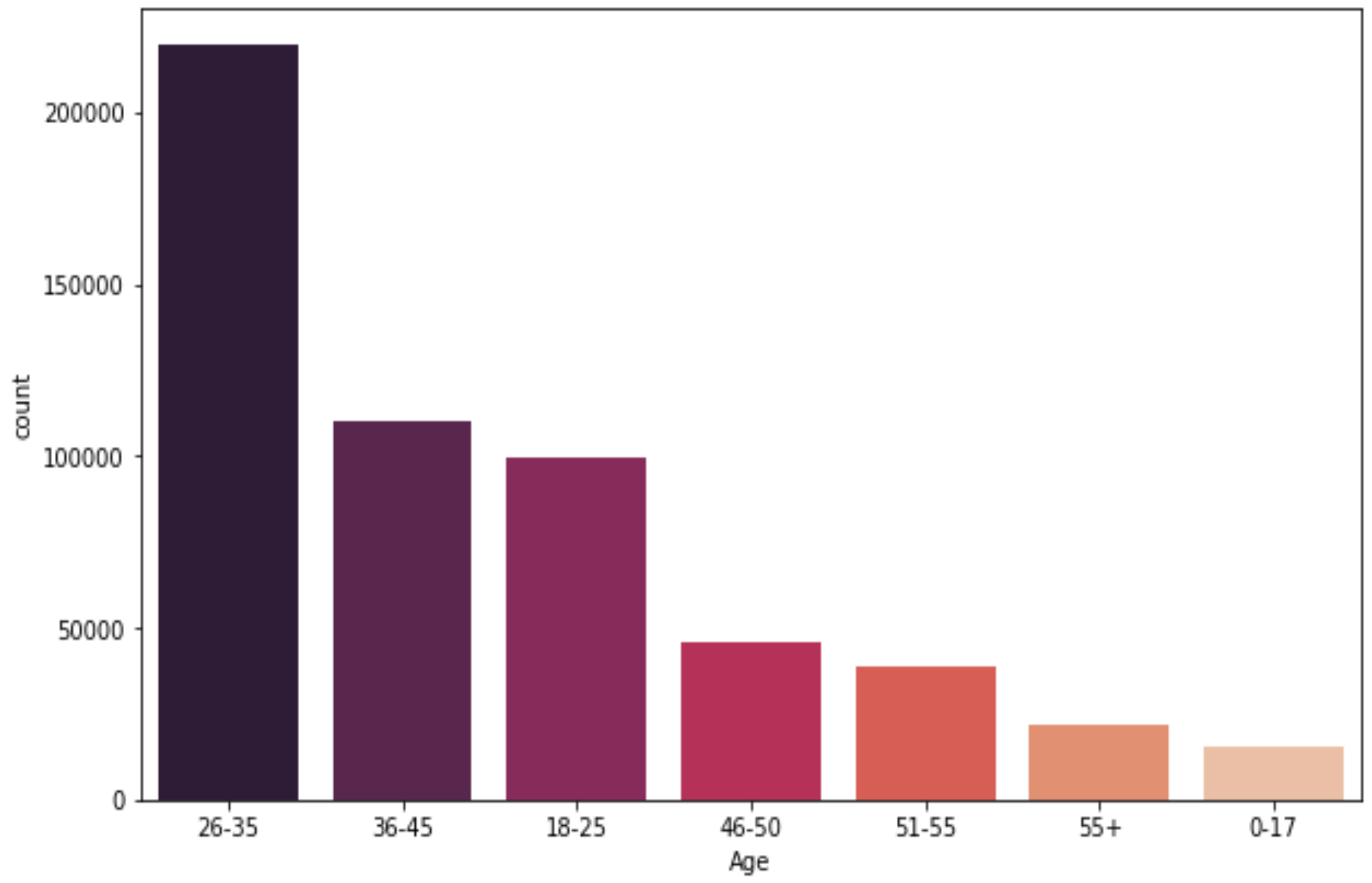
Counting the frequency of the product id

The countplot shows the occurrence (counts) of the categorical variable in the given dataset upon which it is applied.
The graphic representation makes use of the idea of a bar chart.

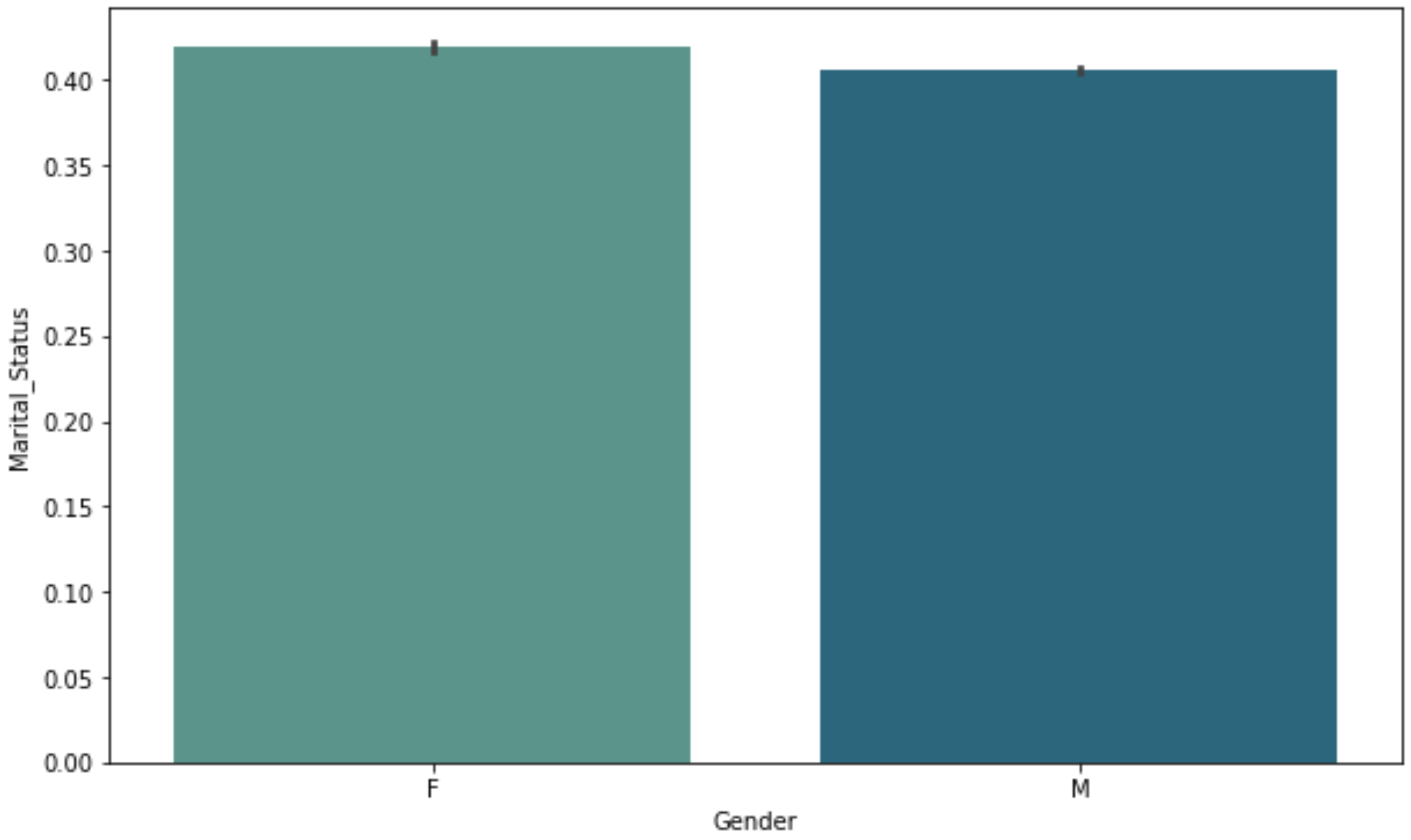


The count of Male gender is higher as compared to the female.

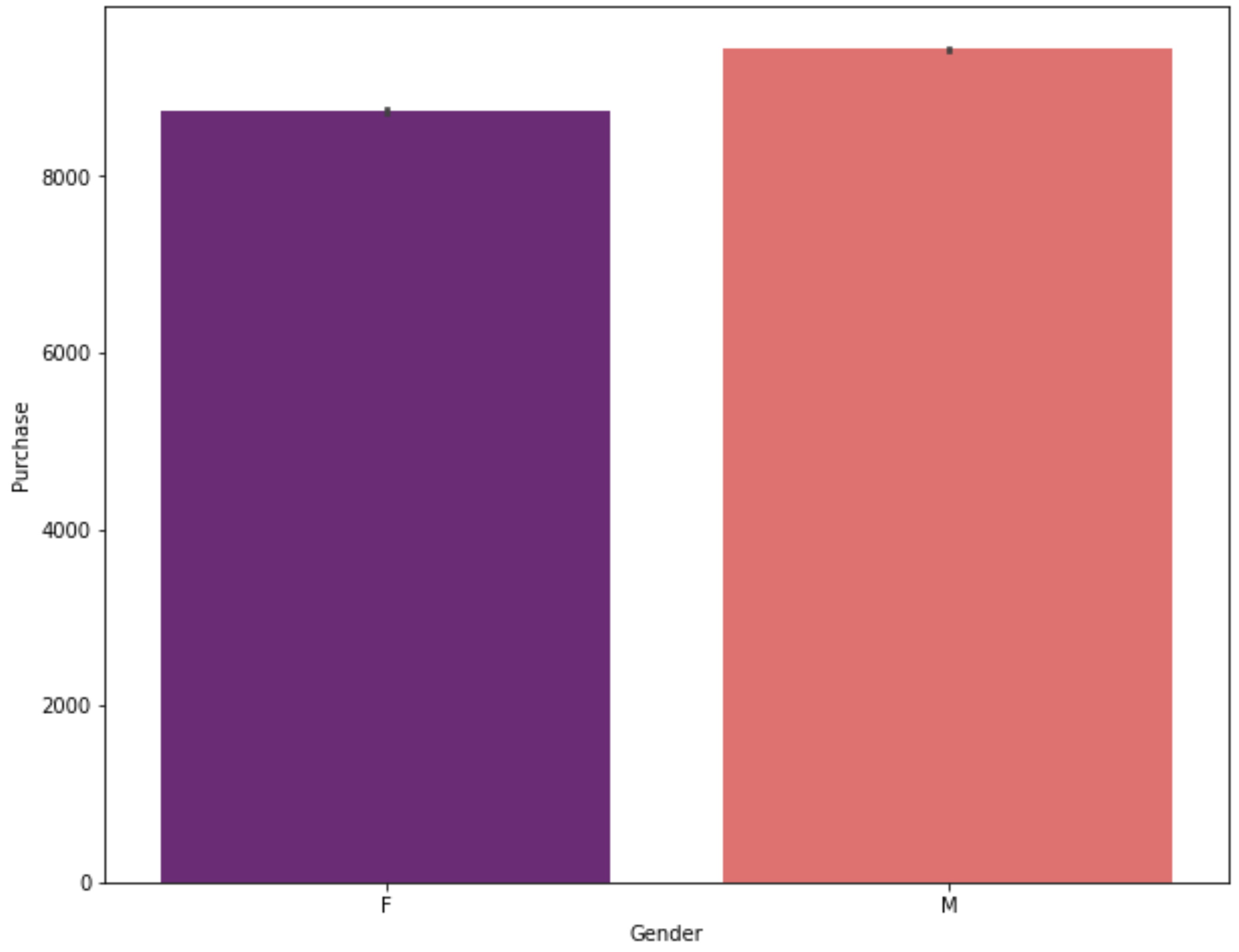
A BARPLOT is used to show how a numerical and a categorical variable are related.



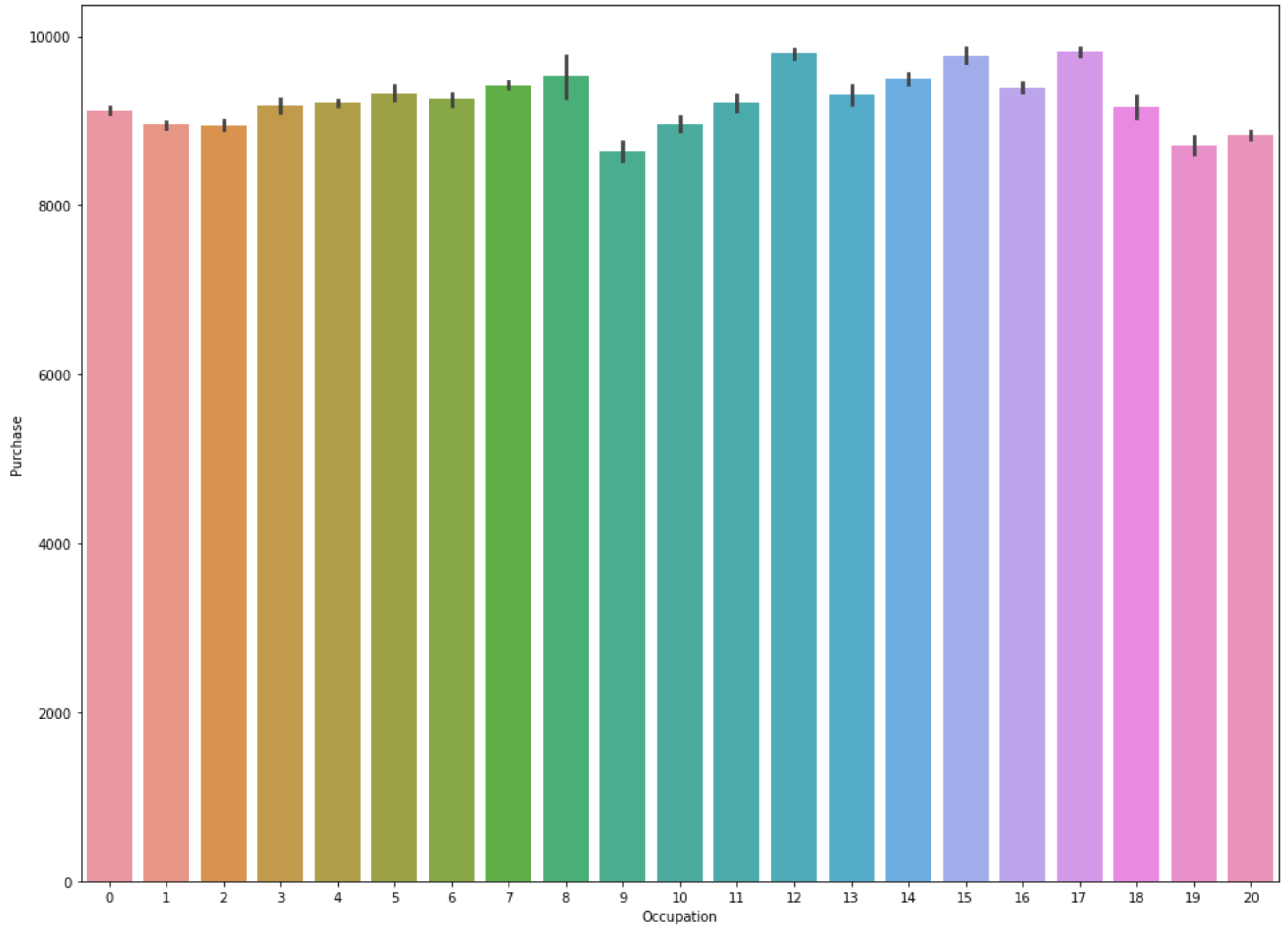
The Age Category of people 26-35 is the highest when compared to the other age groups.



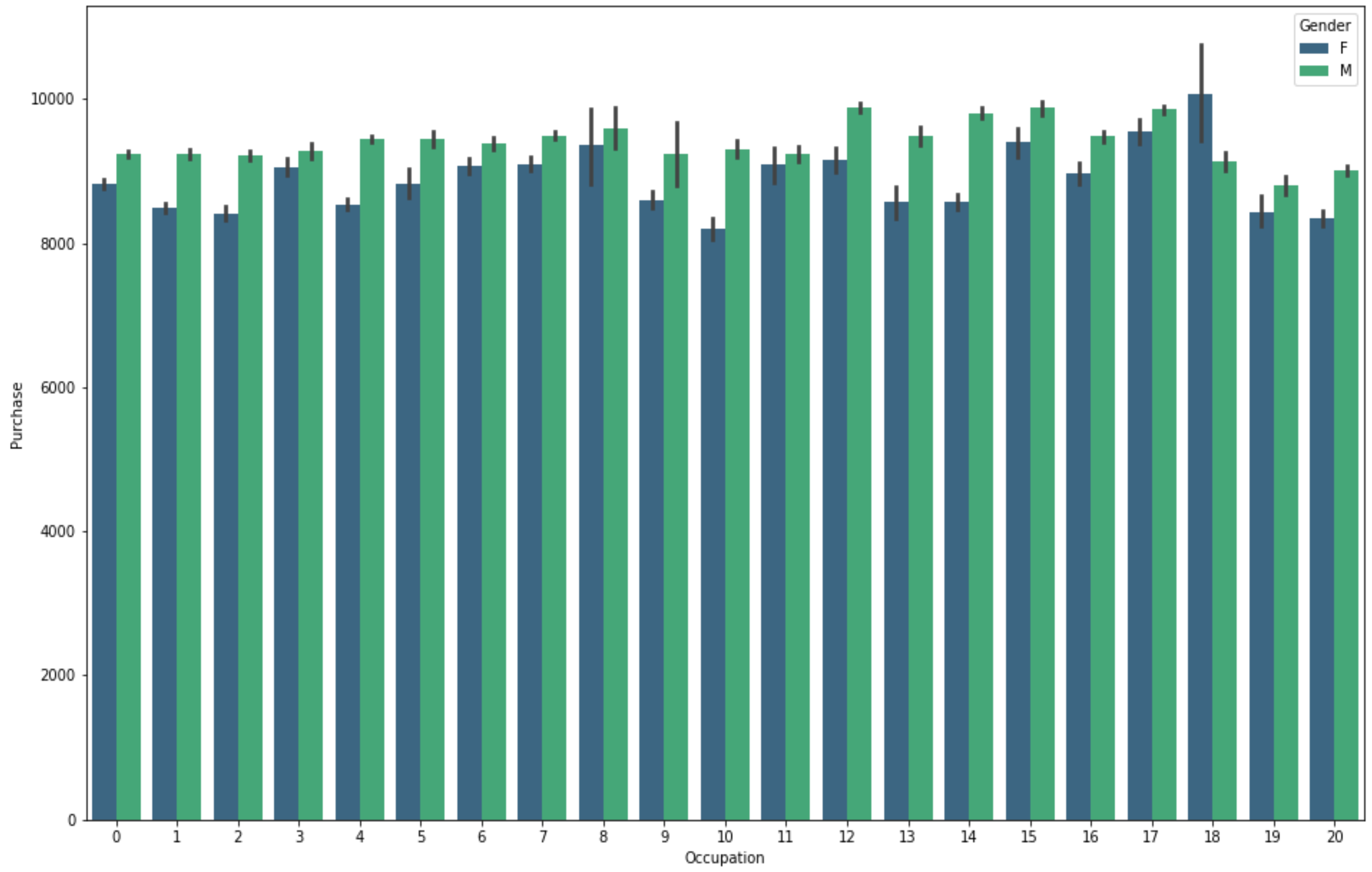
The given visualization shows us that the number of married females making purchases is slightly higher as compared to the number of married males.



Higher purchases have been done by the male gender as compared to the female gender.



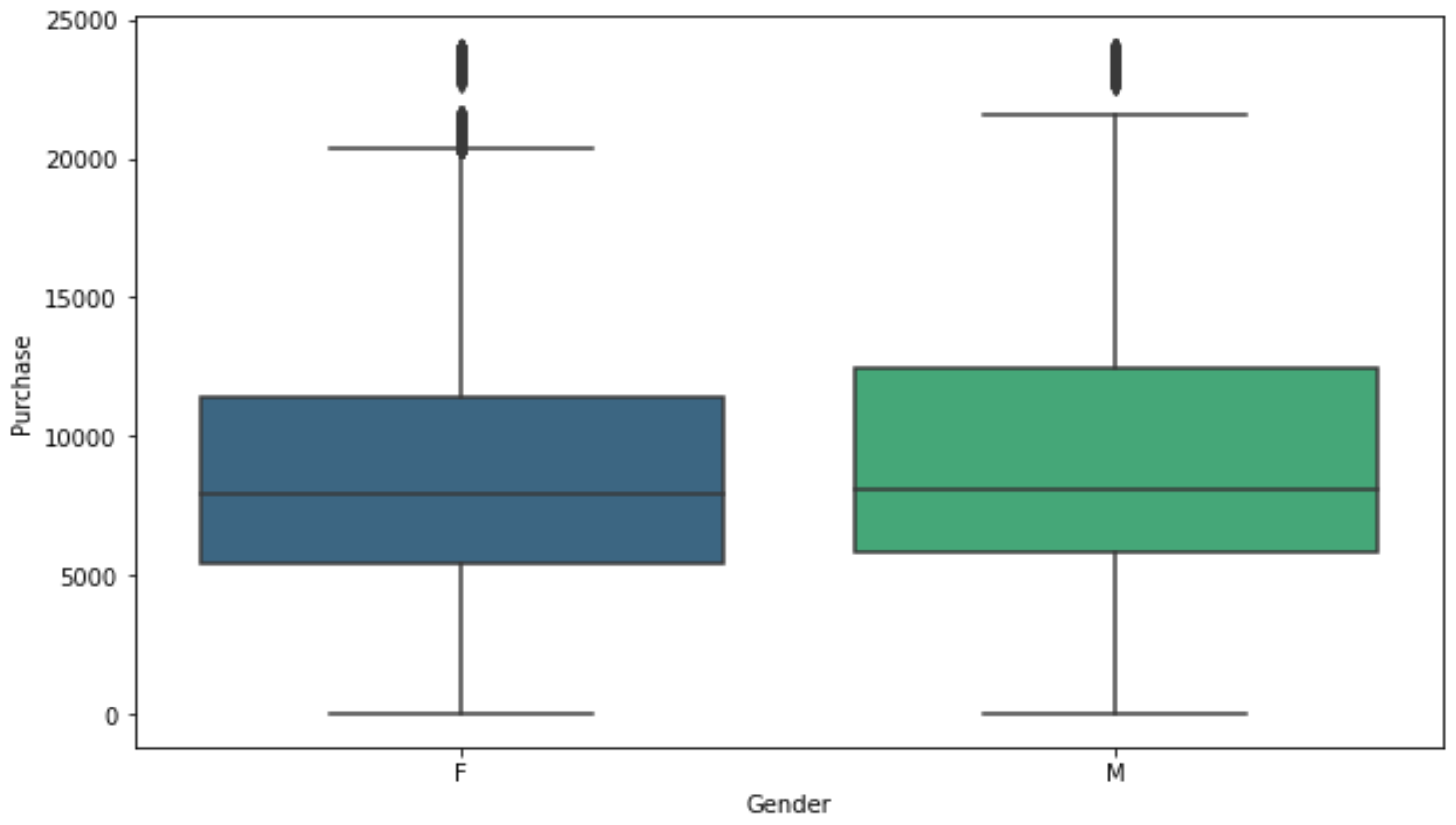
Occupation has a direct effect on the purchases done by the customers and the occupation codes: 12, 15, 17 have higher purchase rates.



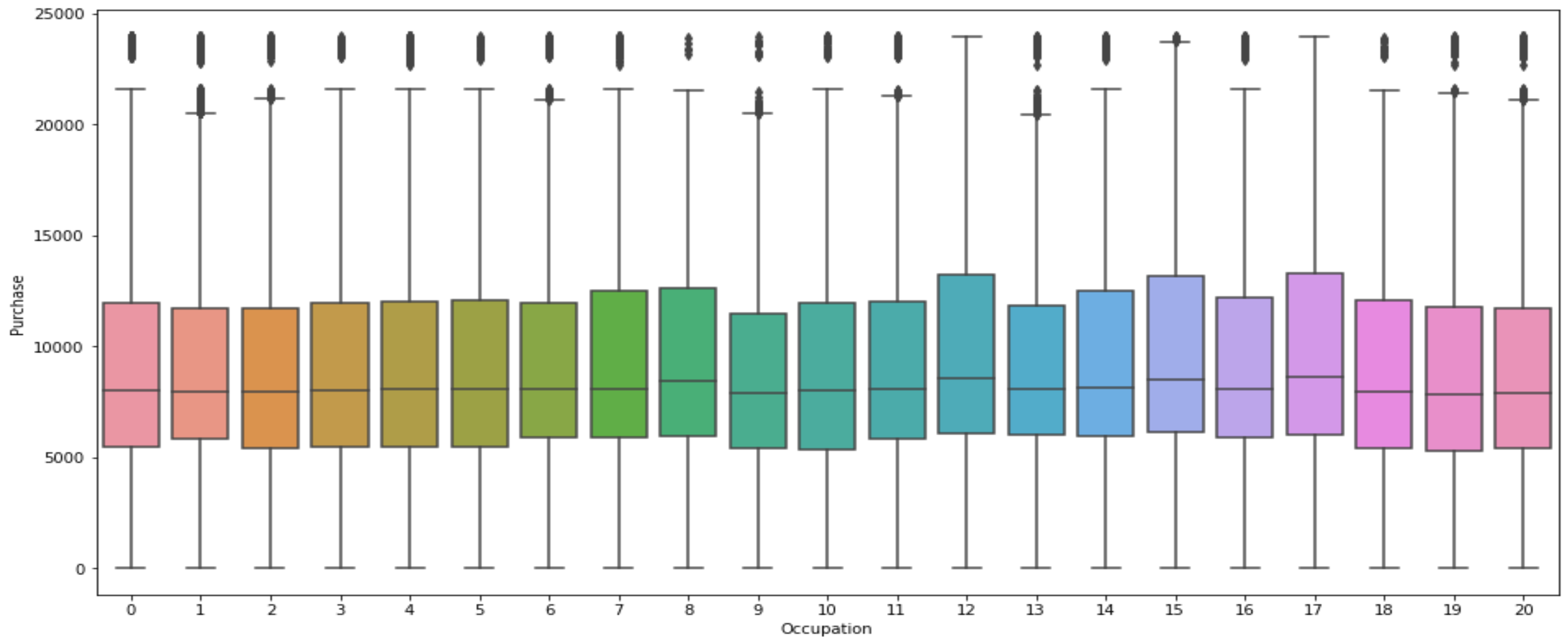
In this graph it can be seen that the female gender in the occupation 18 with higher purchases compared to others.

Outlier Detection: Checking the presence of outliers using BoxPlot.

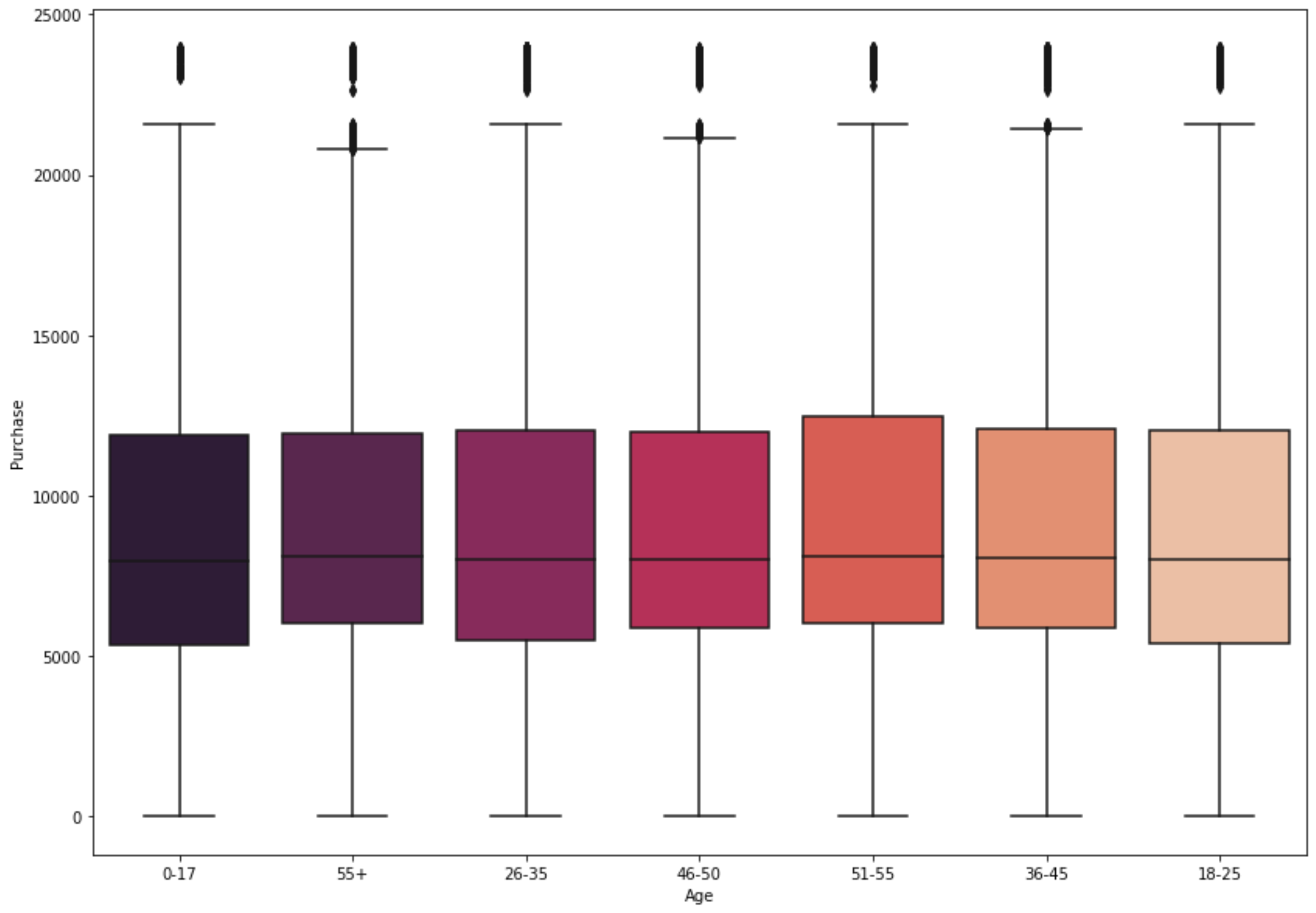
The distribution of data is shown using a boxplot, which is a standardized method based on a five-number summary (minimum, Q1, median, Q3, and maximum). It reveals information about the outliers' values. They show us how closely our data is clustered, whether or not our data is skewed, and whether or not your data is symmetrical.



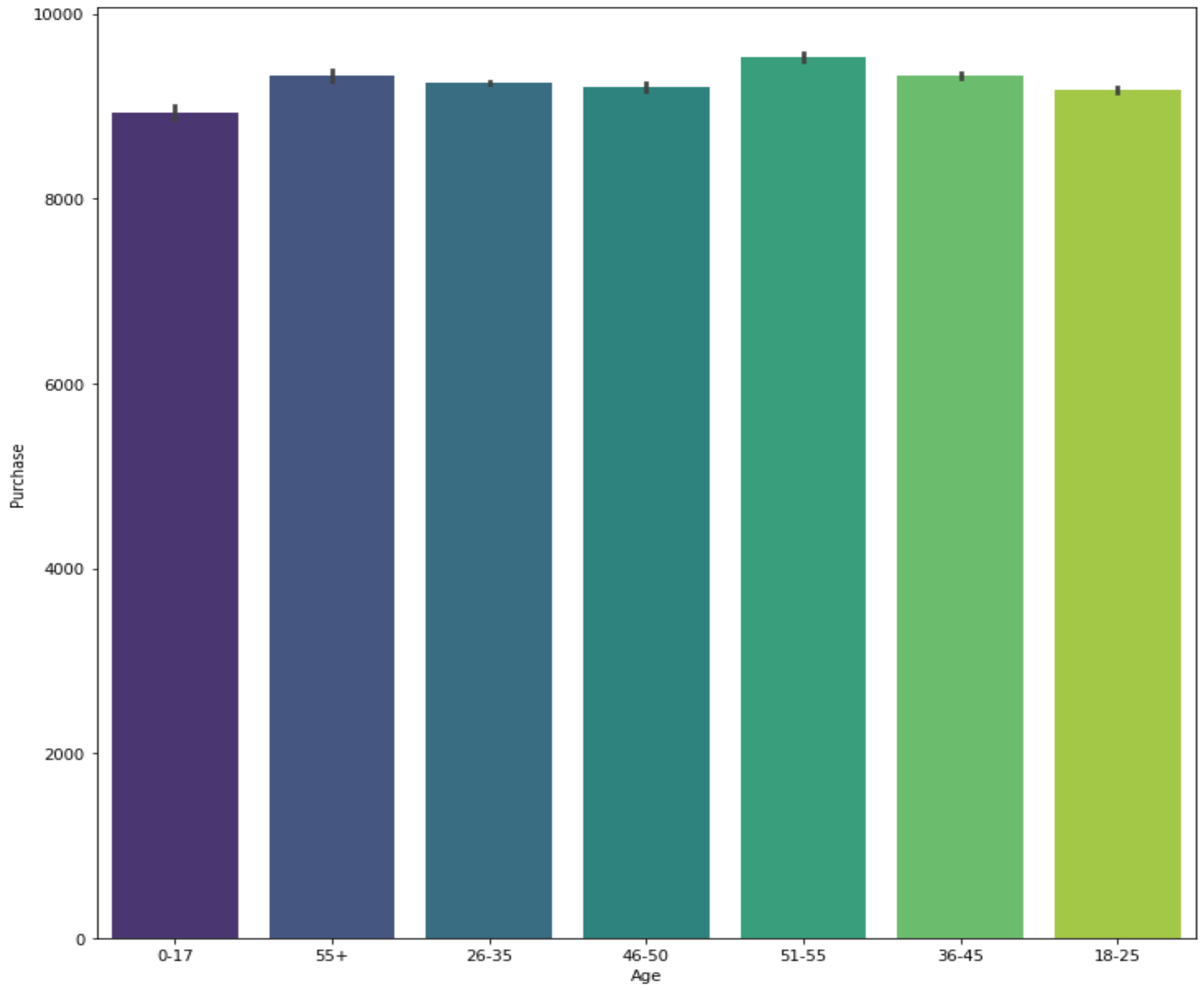
We can see that the presence of outliers in the data for the males and females making purchases have outliers present in them.



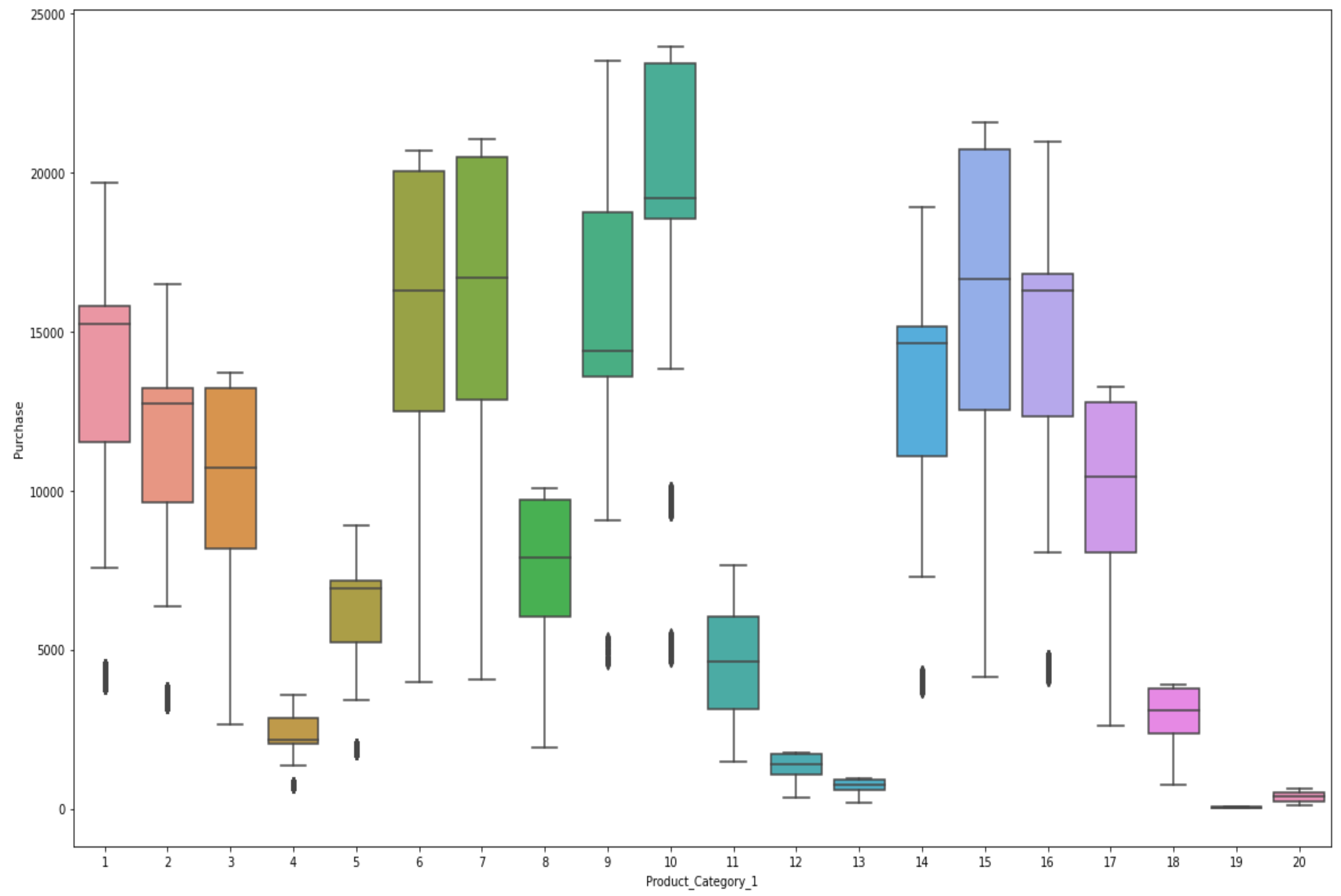
The purchase column has outliers (these can possibly effect the performance of the ML models we're going to apply to our data further on).
And as we can see above there are outliers present in the Purchase column.



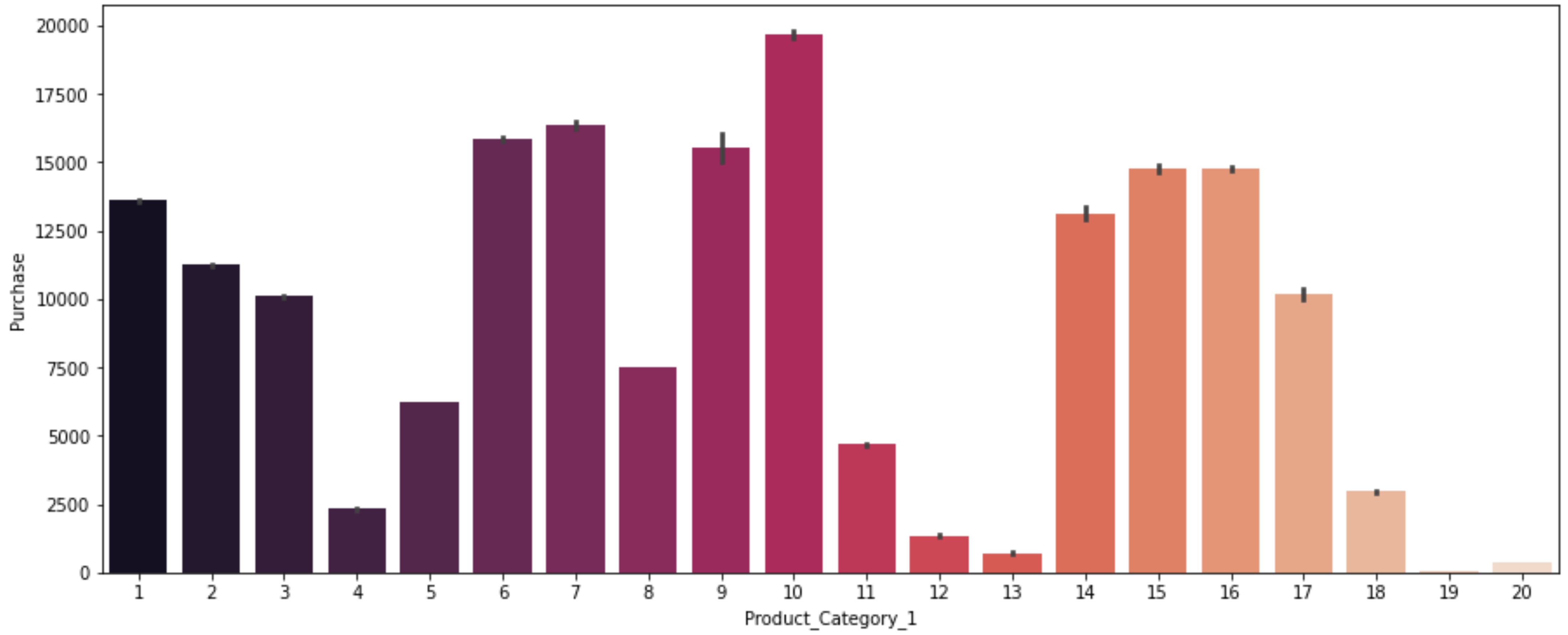
We can see below that the Age with Purchases again have some outliers present in them.



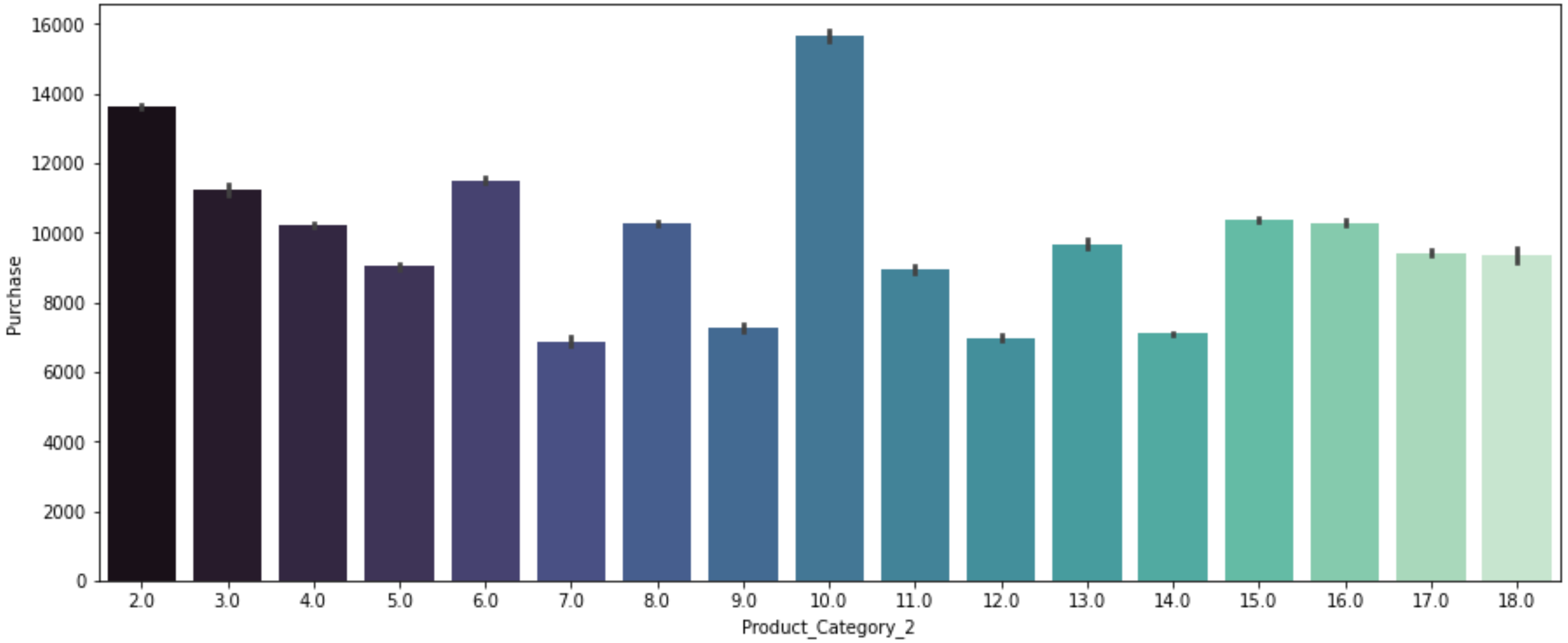
We can see here that people aged 51 – 55 have made the most purchases.



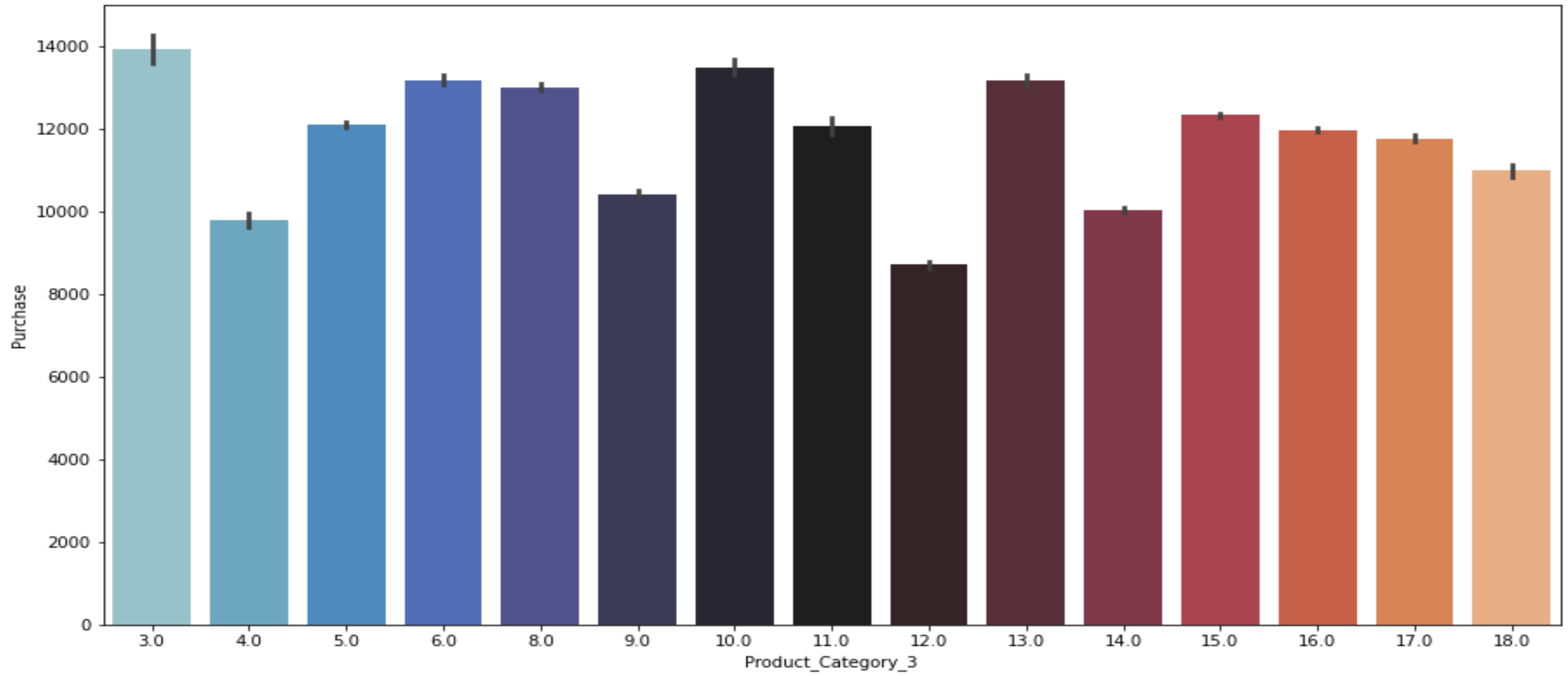
There are outliers present in the Product Category as well.



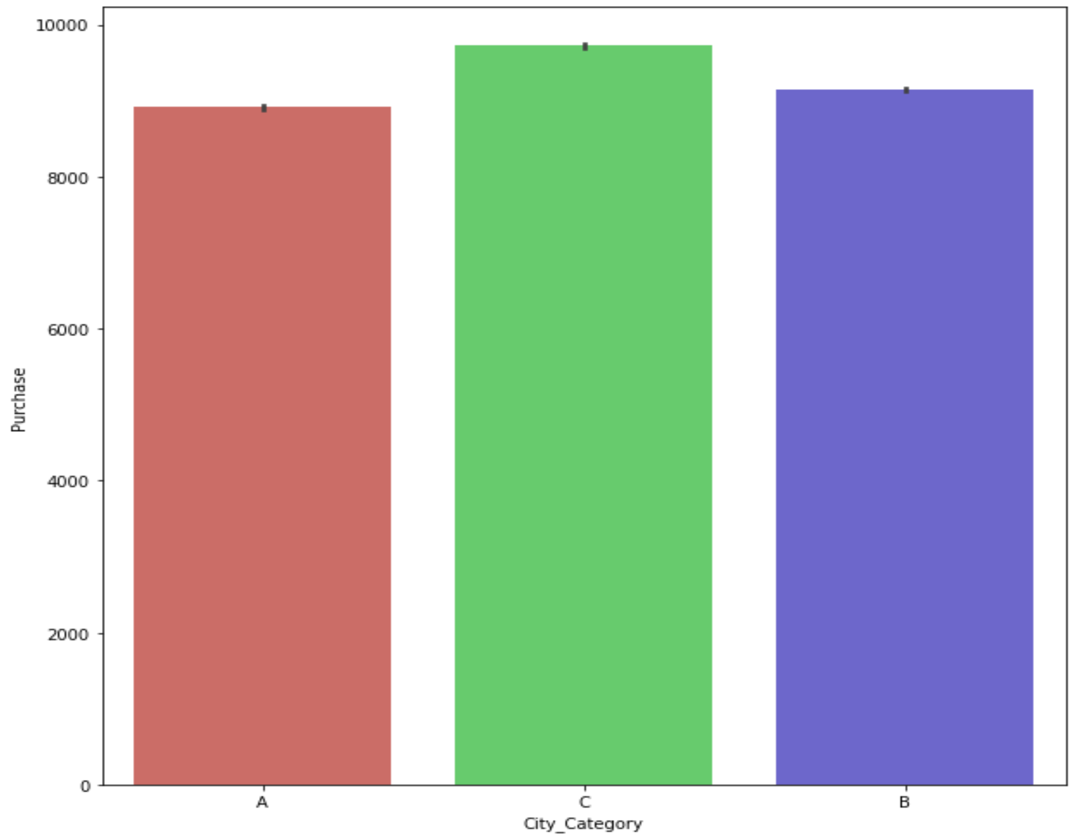
Here we can see that the Product Category 1 = 10 has been purchased the most amongst all other products, followed by 7 and 6.



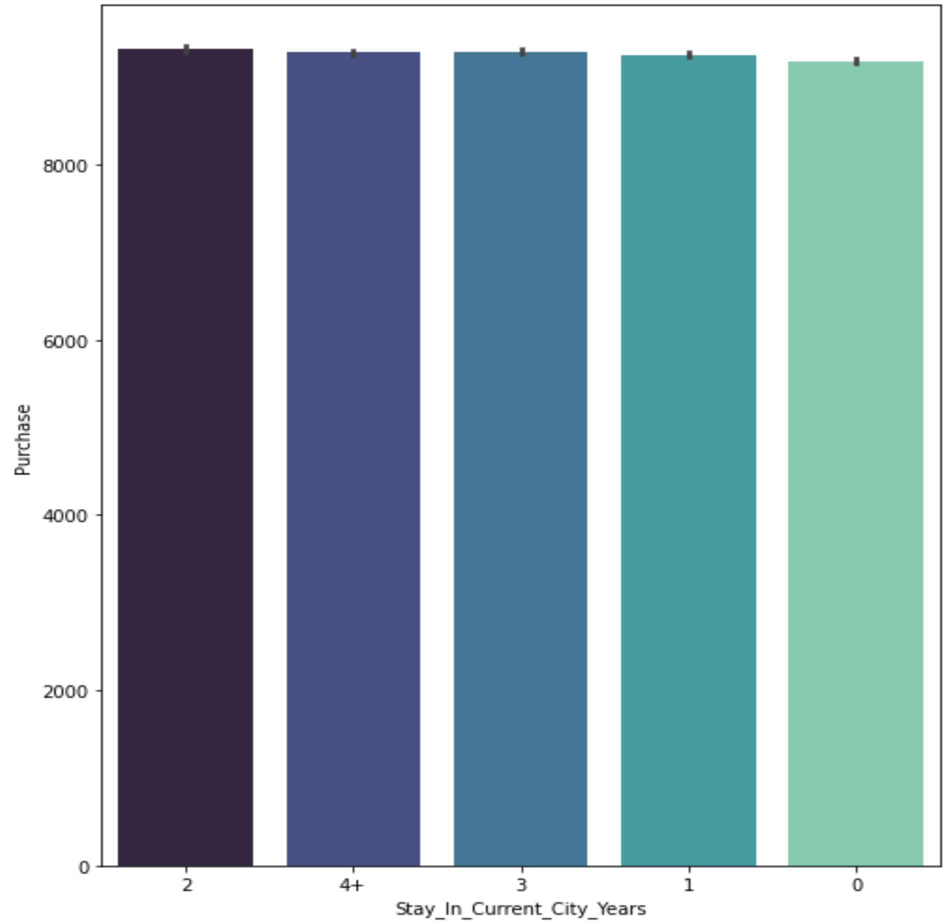
Here we can see that the Product Category 2 = 10.0 has been purchased the most amongst all other products followed by 2.0 and 6.0 .



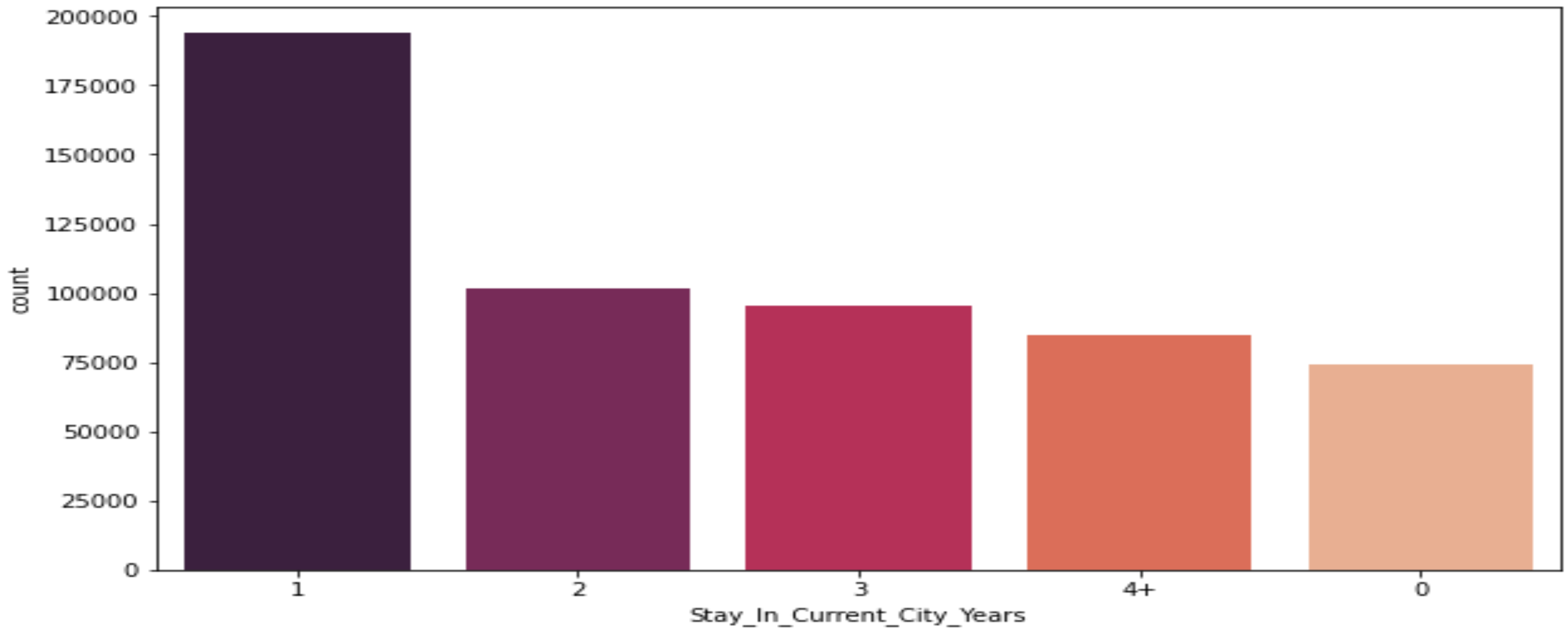
Here we can see that the Product Category 2 = 3.0 has been purchased the most amongst all other products followed by 10.0 and 13.0 .



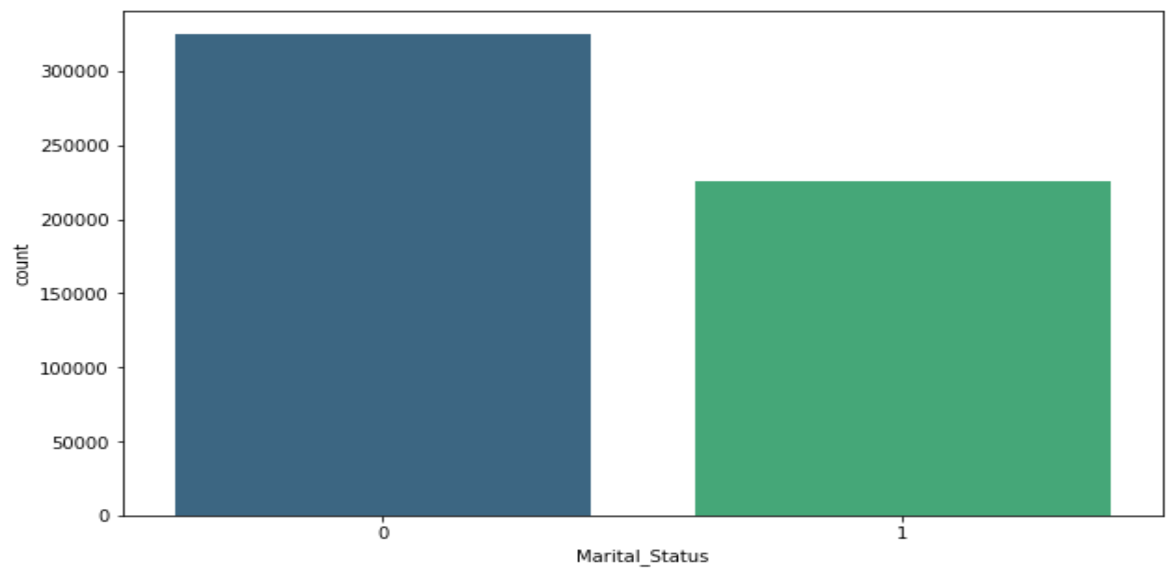
The graph produced here represents that the City Category C has made the most number of purchases followed by B and finally A.



This graph clearly depicts that number of years that people have spent, living in any city, does not seem to have much of an effect on the amount of purchases they make.

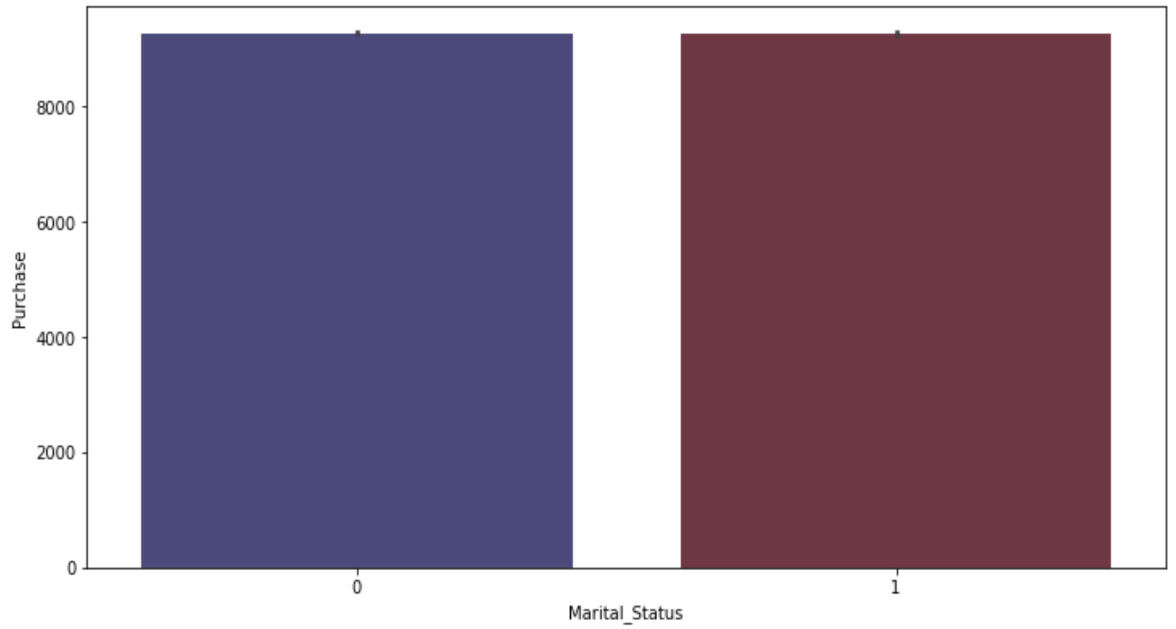


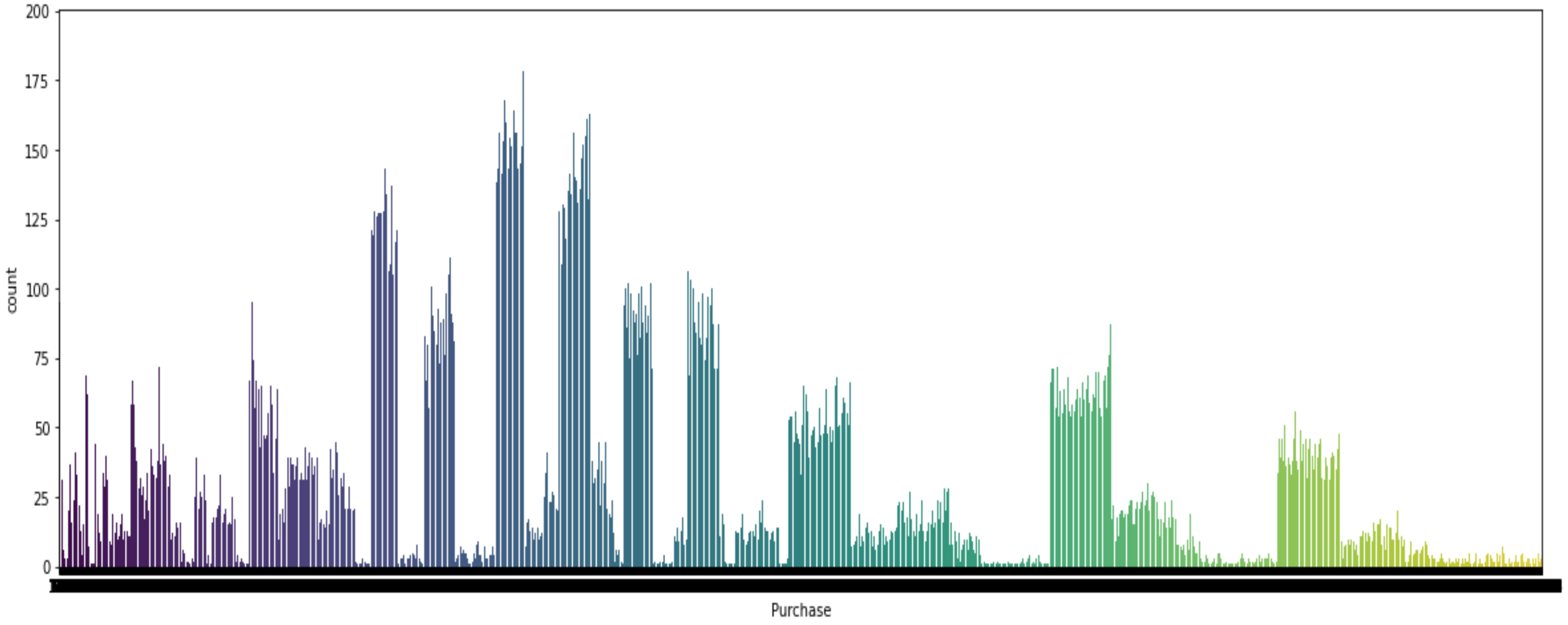
Here we can see that the maximum number of people (making any purchases) have stayed in their respective current cities for around 1 year only.



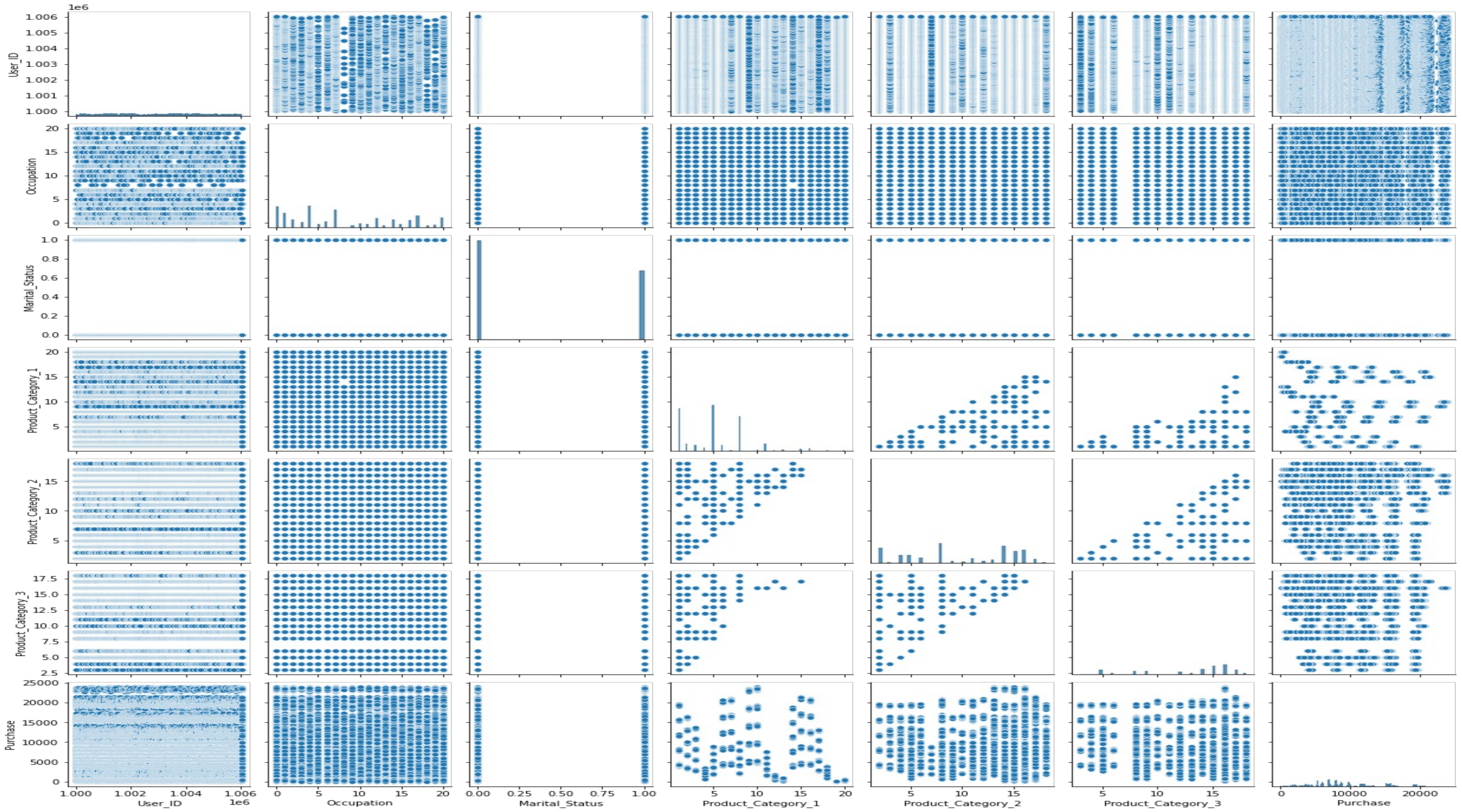
This graph represents that the number of people who are married, in our dataset , is lower than the number of people who are married.

This graph here represents that the amount of purchases made by the people is independent of whether they are married or unmarried.



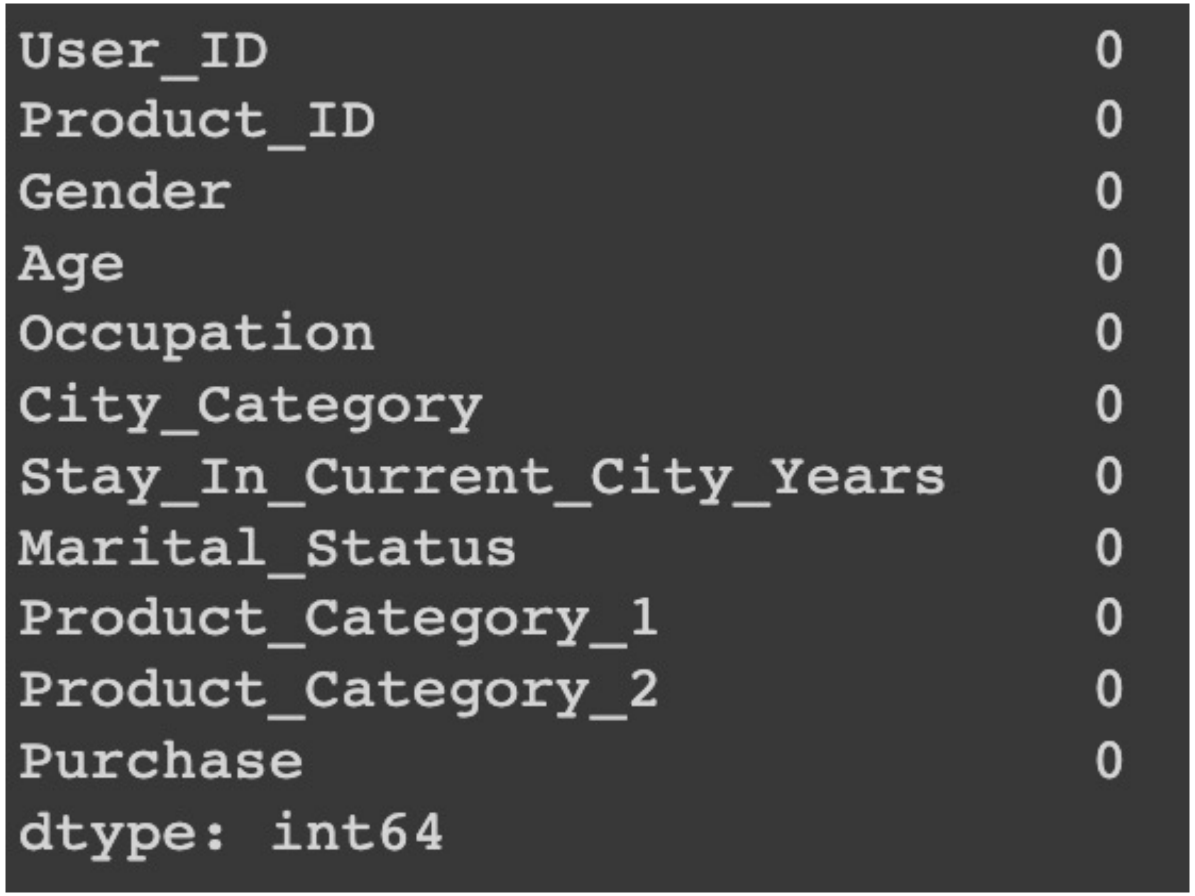


This graph here represents the variation of the purchase amounts across our entire dataset.



These here are the paired plots between each column of the dataset with every other column.

Data preprocessing is a data mining technique which is used to transform the raw data in a useful and efficient format.



```
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
Product_Category_2            0
Purchase                      0
dtype: int64
```

As we can see the missing values have been successfully estimated and now there are no null values present in the dataset anymore.

Data Pre-Processing

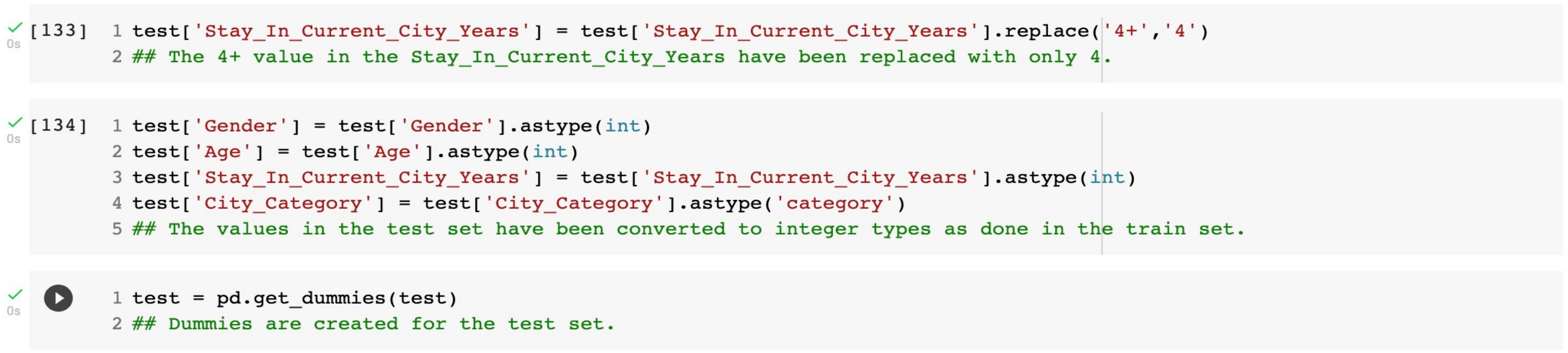
- The steps Involved in the Data Preprocessing we have performed:

1. Data Cleaning: The data may contain many useless and missing pieces. Data cleansing is performed to handle this section. It entails dealing with missing data, noisy data, and so on:
   - ➢ Missing Data: This situation arises when some data is missing in the dataset. We have handled it by:
     - ✓ Filling in the Missing values: It can either be chosen to fill the missing values manually, by the attribute mean or the most probable value.

2. Data Transformation: This stage is used to transform data into appropriate forms for the mining process. This entails the following steps:
   - ➢ Normalization is the process of scaling data values within a specified range (-1.0 to 1.0 or 0.0 to 1.0)
   - ➢ Attribute Selection: In this method, new attributes are created from a given collection of characteristics in order to aid the mining process.

3. Encoding of Categorical Variables: is the process of turning categorical data into integer format so that the data with converted categorical values can be delivered to models for prediction and improvement. We have accomplished this by

   ➢ Label encoding: which is the process of translating labels into a numeric format so that they may be read by machines. Machine learning algorithms can then make better decisions about how those labels should be used. In supervised learning, it is a crucial pre-processing step for the structured dataset.

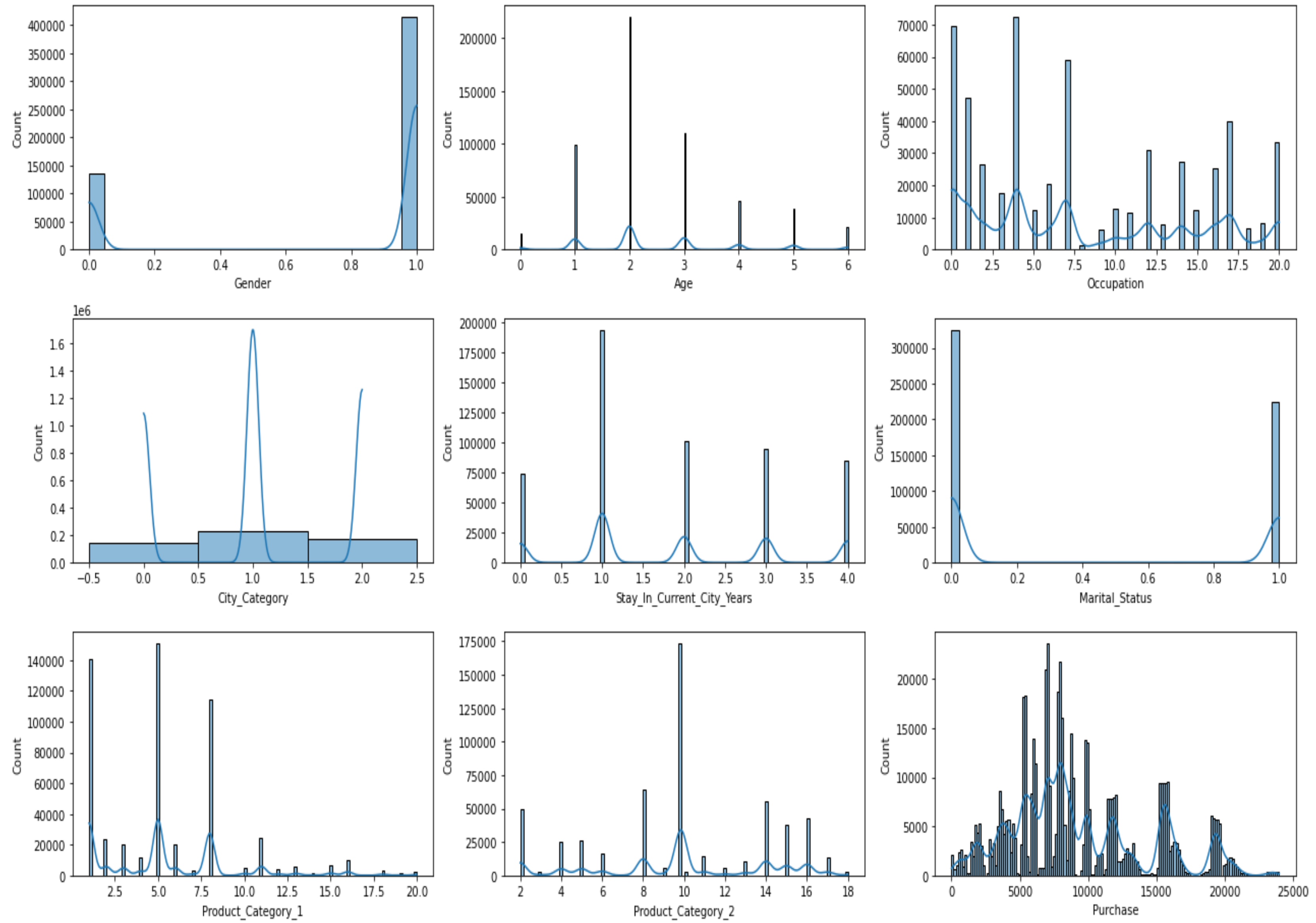


```
[133]
test['Stay_In_Current_City_Years'] = test['Stay_In_Current_City_Years'].replace('4+','4')
## The 4+ value in the Stay_In_Current_City_Years have been replaced with only 4.
```

```
[134]
test['Gender'] = test['Gender'].astype(int)
test['Age'] = test['Age'].astype(int)
test['Stay_In_Current_City_Years'] = test['Stay_In_Current_City_Years'].astype(int)
test['City_Category'] = test['City_Category'].astype('category')
## The values in the test set have been converted to integer types as done in the train set.
```

```
test = pd.get_dummies(test)
## Dummies are created for the test set.
```

As it is shown here, label encoding is being performed in the code as part of the Data Preprocessing steps, and we are creating Dummy Variables for the data.

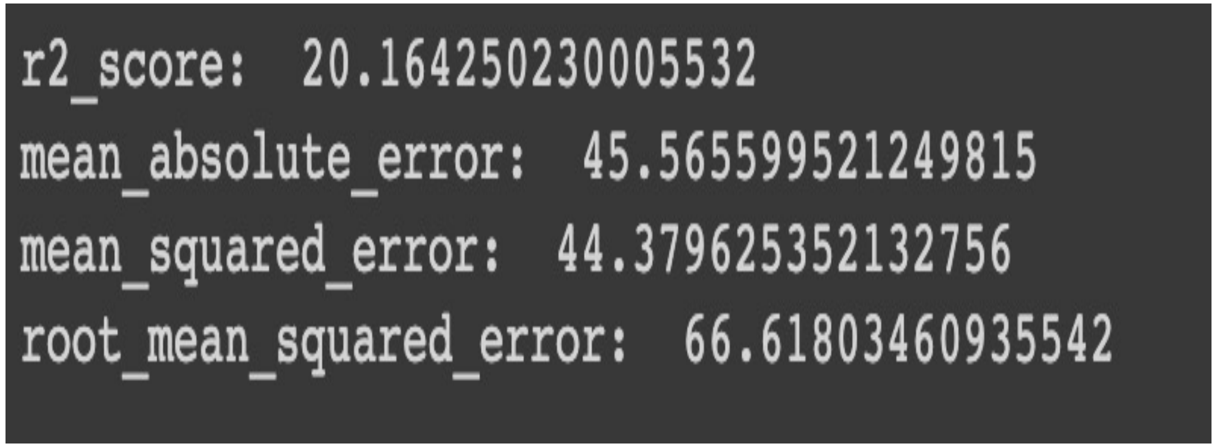


The distribution plot helps us to detect the skewness of the data as can be seen in the purchase column.

Linear Regression

- Linear regression: A variable's value can be predicted using linear regression analysis based on the value of another variable.
- Linear Regression fits a linear model with coefficients w = (w1, …, wp) to minimize the residual sum of squares between the observed targets in the dataset, and the targets predicted by the linear approximation.
- To locate the target variable amongst the dependent and independent variables, linear regression is an approach that can be classified as supervised learning in machine learning.
- In our dataset the target variable is Purchase attribute.

Evaluation Metrics for our Linear Regression Model



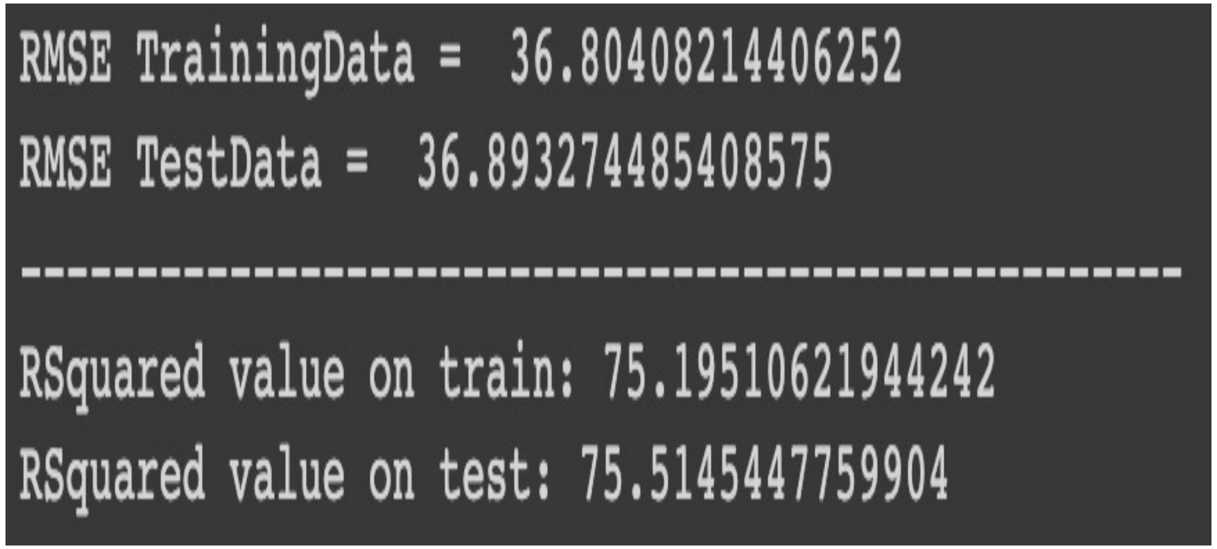
```
r2_score:  20.164250230005532
mean_absolute_error:  45.565599521249815
mean_squared_error:  44.379625352132756
root_mean_squared_error:  66.61803460935542
```

The given evaluation metrics help us to find how well our model is performing. As we can see the r2_score is only 0.20 and as the Root mean square error is high the model is not very accurate to predict the purchases or the target column.

Decision Tree Regressor

- A decision tree creates tree-like models for classification or regression. It incrementally develops an associated decision tree while segmenting a dataset into smaller and smaller sections.
- The outcome is a tree containing leaf nodes and decision nodes.
- Decision trees are useful for dealing with non-linear data sets. Categorical variable and continuous variable decision trees are the two categories into which decision trees may be divided.

Evaluation Metrics for our Decision Tree Model



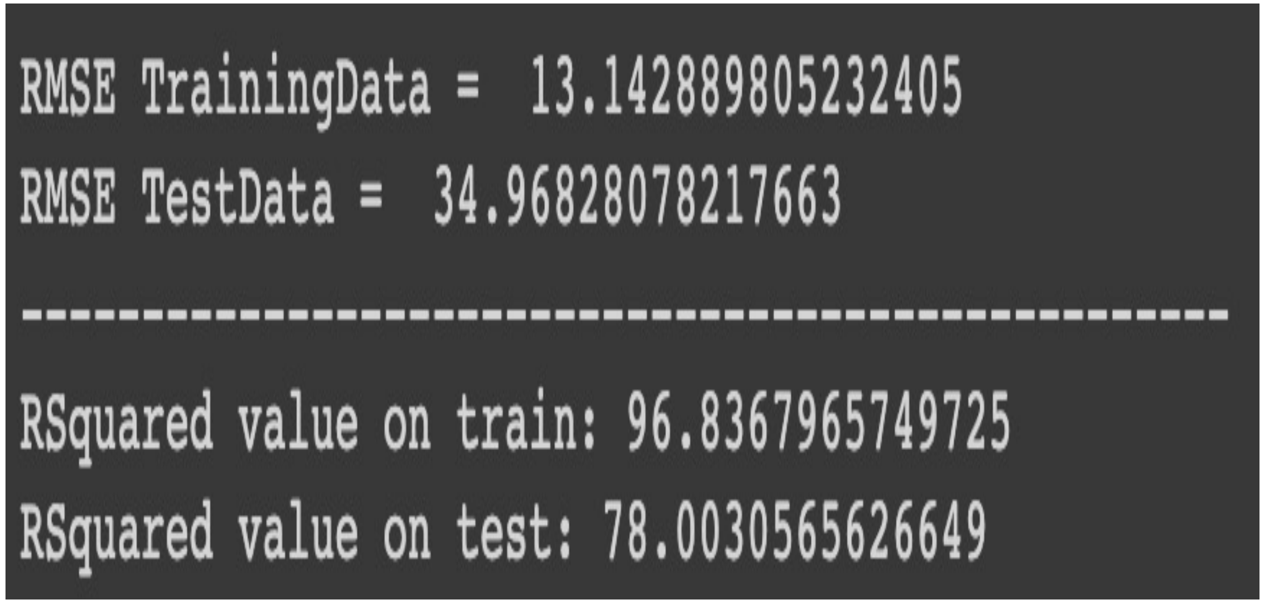
```
RMSE TrainingData =  36.80408214406252
RMSE TestData =  36.893274485408575
-----------------------------------------------
RSquared value on train: 75.19510621944242
RSquared value on test: 75.5145447759904
```

The Decision Tree Regressor is better compared to Linear regression as it can be observed that the root mean square error is less as compared to the previous model and the RSquared value is higher in this model.

Random Forest Regressor

- As a supervised learning algorithm, Random Forest Regression does regression using the ensemble learning approach.
- The ensemble learning method combines predictions from various machine learning algorithms to provide predictions that are more accurate than those from a single model.
- The Random Forest Algorithm's ability to handle data sets with both continuous variables, as in regression, and categorical variables, as in classification, is one of its most crucial qualities.

Evaluation Metrics for our Random Forest Model



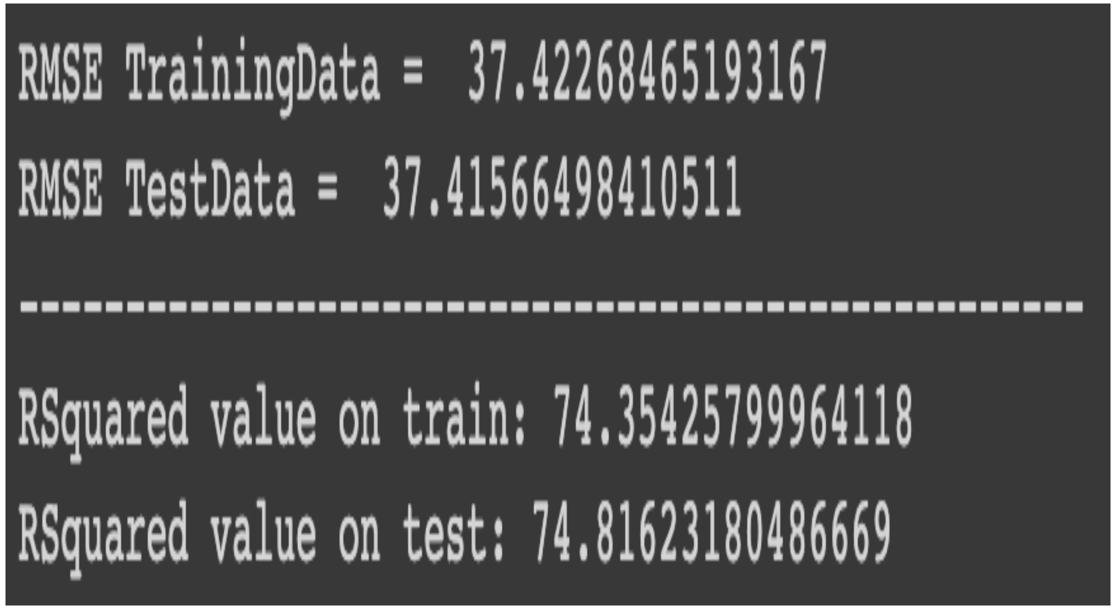
```
RMSE TrainingData =  13.142889805232405
RMSE TestData =  34.96828078217663
--------------------------------------------
RSquared value on train: 96.8367965749725
RSquared value on test: 78.003056562649
```

The Random Forest regressor model is again better than the previous model as we have a lower root mean square error value and the RSquared value is higher than the previous model.

XGBoost Regressor

- A well-liked supervised-learning technique for regression and classification on sizable datasets is called XGBoost (eXtreme Gradient Boosting).
- In order to produce reliable results, it employs shallow decision trees that are formed sequentially and a highly scalable training technique that prevents overfitting.
- When it comes to classification and regression predictive modeling issues, XGBoost dominates structured or tabular datasets.

Evaluation Metrics for our XGBoost Model



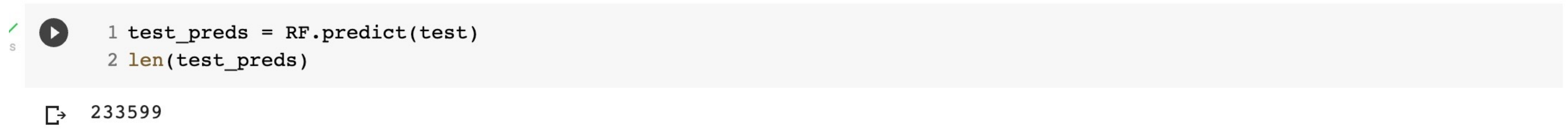
```
RMSE TrainingData =  37.42268465193167
RMSE TestData =  37.41566498410511
---------------------------------------------
RSquared value on train: 74.35425799964118
RSquared value on test: 74.81623180486669
```

The XGBoost regressor model is not better than the previous model as we have a bit higher root mean square error value and the RSquared value is lower than the previous model.

Conclusion

After applying the 4 ML modes i.e., Linear Regression, Decision Tree Regressor, Random Forest Regressor and XGBoost Regressor, we have reached to the conclusion that the Random Forest Regressor model performs the best.
The reason for the above conclusion is that Random Forest Regressor has a lower root mean square error value and the RSquared value is higher than the previous model.
This analysis can be further improved by adding more features to the data and we may also be able to add consumer behavior analysis so that we can predict more accurately the black Friday sales in the coming years, along with judging people’s sentiments regarding the ongoing sales.

Finalizing Results



```
test_preds = RF.predict(test)
len(test_preds)
```

```
233599
```

We chose to use the Random Forest Regressor on our final test data as it had performed the best, on the Training Data.