Project Note-1

Name: Preeti Singh

Batch Name: PGPDSBA Online Feb20_A

Capstone Project: Tourism

Table of Content

Problem: Tourism

A reputed tourism company is planning to launch a long term travel package. The Product Manager has access to the existing customers' data and information. He wishes to analyse the trend of existing customers to figure out which customer is going to purchase the long term travel package.

Problem Understanding:

Defining Problem Statement:

This data is basically about of tourism based company. It's objective is to launch long term travel package and offered the product to customers which belongs to probably subscription based customers (the customers who had paid money to get membership of the organisation to buy a product).There is total 4888 rows and 20 columns in the data set. To check the viability of market they have gone out to certain no of customers and calculated all features of the data that is mentioned in the data .On the behalf of this we have to predict whether a customer is taken a long term travel product or not.

Need of study/Project:

Tourism is a favorite leisure activity. The motivation which causes someone to choose certain activities and a destination for vacation is an interesting issue, which allows for a better understanding of people's behavior in the area of leisure spending.

Understanding Business and Social Opportunity:

Social tourism improves the well-being of people and reduces stress, improves physical and mental health, increases self-esteem and confidence, enables families to develop positive relationships, provides new skills, and even helps increase employment **opportunities**.

Data Report:

First we will import all necessary libraries. Then load, view and get high level understanding of data set.

Checking the quantum of data:

```
the number of rows 4888
the number of columns 20
```

Checking the data types:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4888 entries, 0 to 4887
Data columns (total 20 columns):
 #   Column                  Non-Null Count  Dtype  
---  ------                  --------------  -----  
 0   CustomerID              4888 non-null   int64  
 1   ProdTaken               4888 non-null   int64  
 2   Age                     4662 non-null   float64
 3   PreferredLoginDevice    4863 non-null   object 
 4   CityTier                4888 non-null   int64  
 5   DurationOfPitch         4637 non-null   float64
 6   Occupation              4888 non-null   object 
 7   Gender                  4888 non-null   object 
 8   NumberOfPersonVisited   4888 non-null   int64  
 9   NumberOfFollowups       4843 non-null   float64
 10  ProductPitched          4888 non-null   object 
 11  PreferredPropertyStar   4862 non-null   float64
 12  MaritalStatus           4888 non-null   object 
 13  NumberOfTrips           4748 non-null   float64
 14  Passport                4888 non-null   int64  
 15  PitchSatisfactionScore  4888 non-null   int64  
 16  OwnCar                  4888 non-null   int64  
 17  NumberOfChildrenVisited 4822 non-null   float64
 18  Designation             4888 non-null   object 
 19  MonthlyIncome           4655 non-null   float64
dtypes: float64(7), int64(7), object(6)
memory usage: 763.9+ KB
```

Observations:

Here, the features ProdTaken(target variable),CityTier,OwnCar and Passport ,PreferredPropertyStar are actually categorical in nature but in the data set all are in numerical (int/flaot) type. We need to convert these into object type for further analysis. ProdTaken is target variable and rest of all are predictor(input variables).

Checking the descriptive statistics of data:

	count	mean	std	min	25%	50%	75%	max
CustomerID	4888.0	202443.500000	1411.188388	200000.0	201221.75	202443.5	203665.25	204887.0

	count	mean	std	min	25%	50%	75%	max
ProdTaken	4888.0	0.188216	0.390925	0.0	0.00	0.0	0.00	1.0
Age	4662.0	37.622265	9.316387	18.0	31.00	36.0	44.00	61.0
CityTier	4888.0	1.654255	0.916583	1.0	1.00	1.0	3.00	3.0
DurationOfPitch	4637.0	15.490835	8.519643	5.0	9.00	13.0	20.00	127.0
NumberOfPersonVisited	4888.0	2.905074	0.724891	1.0	2.00	3.0	3.00	5.0
NumberOfFollowups	4843.0	3.708445	1.002509	1.0	3.00	4.0	4.00	6.0
PreferredPropertyStar	4862.0	3.581037	0.798009	3.0	3.00	3.0	4.00	5.0
NumberOfTrips	4748.0	3.236521	1.849019	1.0	2.00	3.0	4.00	22.0
Passport	4888.0	0.290917	0.454232	0.0	0.00	0.0	1.00	1.0
PitchSatisfactionScore	4888.0	3.078151	1.365792	1.0	2.00	3.0	4.00	5.0
OwnCar	4888.0	0.620295	0.485363	0.0	0.00	1.0	1.00	1.0
NumberOfChildrenVisited	4822.0	1.187267	0.857861	0.0	1.00	1.0	2.00	3.0
MonthlyIncome	4655.0	23619.853491	5380.698361	1000.0	20346.00	22347.0	25571.00	98678.0

Observations:

- At least 50% customers are in age of 35 to 36(younger age grup) that is closer to average age of customers also.
- At least 50% customers belong to Tier-1 city. It means 50% customers belong to metropolitan city.
- At least 75% customers come up with 3 persons to visit the company.
- At least 50% customers preferred to stay in 3 star hotels.
- At least 50% customers are having no passport. It means they are local traveller.
- At least 50% customers are having own car, may be they use their own car for travelling.
- At least 50% customers are done total no of trips 3.It means these customers can do travelling most frequently.
- An average monthly income of customers is 23619.
- Out of 4888 customers on an average total 920(18 %) customers are taken long term travel package.

Let's convert the features that are actually in categorical nature but in data set in numerical nature, into appropriate data type for further analysis.

The features ProdTaken,Passport,OwnCar are having binary values. We will convert all these variables into object type by assigning 1 == Yes and 0==No with the lambda( ) and the features CityTier and preferredPropertyStar are having ordered values, so we will labelled different name for each different values. For CityTier feature we will assign 1==Tier-1,2===Tier-2 and 3==Tier-3 and for feature PreferredPropertyStar, we will replace all nan values==Unknown,4==4 Star,3==3 Star,2==2 Star,1==1Star.

Checking the info of data set df_tourism1:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4888 entries, 0 to 4887
Data columns (total 20 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   CustomerID               4888 non-null   int64
 1   ProdTaken                4888 non-null   object
 2   Age                      4662 non-null   float64
 3   PreferredLoginDevice     4863 non-null   object
 4   CityTier                 4888 non-null   object
 5   DurationOfPitch          4637 non-null   float64
 6   Occupation               4888 non-null   object
```

```
 7   Gender                     4888 non-null   object
 8   NumberOfPersonVisited      4888 non-null   int64
 9   NumberOfFollowups          4843 non-null   float64
 10  ProductPitched             4888 non-null   object
 11  PreferredPropertyStar      4862 non-null   object
 12  MaritalStatus              4888 non-null   object
 13  NumberOfTrips              4748 non-null   float64
 14  Passport                   4888 non-null   object
 15  PitchSatisfactionScore     4888 non-null   int64
 16  OwnCar                     4888 non-null   object
 17  NumberOfChildrenVisited    4822 non-null   float64
 18  Designation                4888 non-null   object
 19  MonthlyIncome              4655 non-null   float64
dtypes: float64(6), int64(3), object(11)
memory usage: 763.9+ KB
```

Now ,all the int/float type categorical variable is in object type.

Exploratory Data analysis:

Checking missing values: We can check missing values by using df_tourism1.isnull().sum().

```
DurationOfPitch            251
MonthlyIncome              233
Age                        226
NumberOfTrips              140
NumberOfChildrenVisited     66
NumberOfFollowups           45
PreferredPropertyStar       26
PreferredLoginDevice        25
Passport                     0
MaritalStatus                0
ProductPitched               0
Designation                  0
NumberOfPersonVisited        0
Gender                       0
Occupation                   0
PitchSatisfactionScore       0
CityTier                     0
OwnCar                       0
ProdTaken                    0
CustomerID                   0
dtype: int64
```

There are so many missing values present in data. We need to take care of this for future analysis.

DurationOfPitch,MonthlyIncome,Age,NumberOfTrips,NumberOfChildrenVisited,NumberOfFollwups are numerical variable wherein missing values are present.

PreferredPropertyStar,PreferredLoginDevice are categorical variable wherein missing values are present.

Checking of total no of missing values:

df_tourism1.isnull().sum().sum()

```
1012
```

Let's see count plot of missing values:

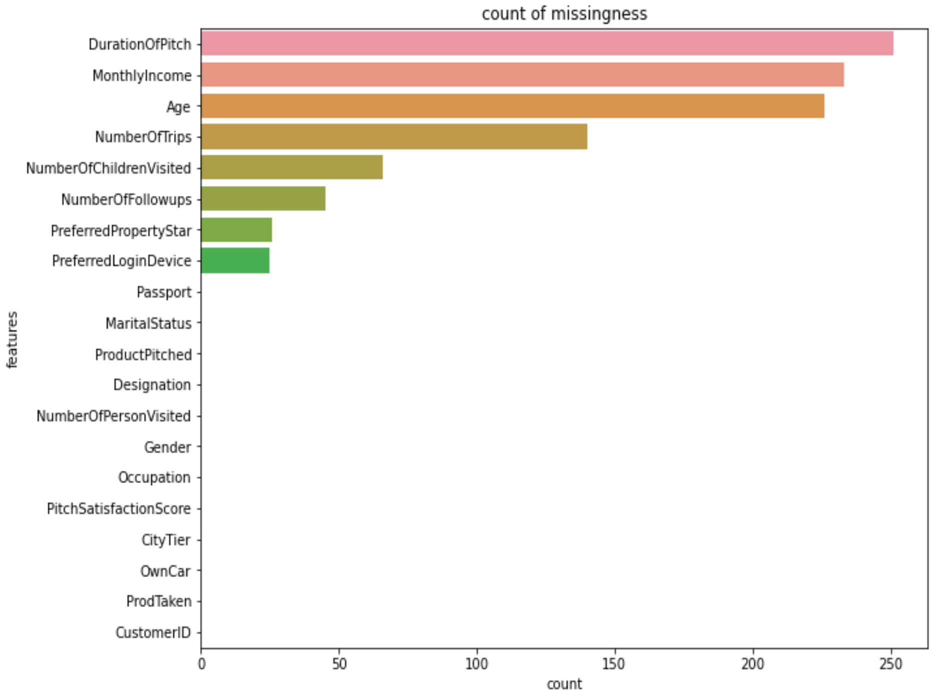

Fig-1

From the above plot we can see feature DurationOfPitch has highest count of missing values.

Calculating propensity of missing values:

```
DurationOfPitch            0.051350
MonthlyIncome              0.047668
Age                        0.046236
NumberOfTrips              0.028642
NumberOfChildrenVisited    0.013502
NumberOfFollowups          0.009206
PreferredPropertyStar      0.005319
PreferredLoginDevice       0.005115
Passport                   0.000000
MaritalStatus              0.000000
ProductPitched             0.000000
Designation                0.000000
NumberOfPersonVisited      0.000000
```

```
Gender                      0.000000
Occupation                  0.000000
PitchSatisfactionScore      0.000000
CityTier                    0.000000
OwnCar                      0.000000
ProdTaken                   0.000000
CustomerID                  0.000000
dtype: float64
```

Observations:

We can observe from the above output, there are some missing values present in numerical variable and categorical variable as well and the extent of missing ness is not so high. It is varying from 0.5% to 5.1%.We can opt removing these observation because variation of missing ness is not high but we will try and impute these missing values to best extent as possible.

Treating of Missing values by median and mode:

Let's separate the numerical and categorical variable first. We can treat the missing values that are present in numerical variable by median().and we can use mode() for categorical variable. We are referring df_tourism2_num data frame for numerical variable and df_tourism2_cat for categorical variable.

Missing values imputation for numerical variable by using median().

df_tourism2_num["Age"]=df_tourism2_num["Age"].fillna(df_tourism2_num["Age"].median())

df_tourism2_num["DurationOfPitch"]=df_tourism2_num["DurationOfPitch"].fillna(df_tourism2_num["DurationOfPitch"].median())

df_tourism2_num["NumberOfFollowups"]=df_tourism2_num["NumberOfFollowups"].fillna(df_tourism2_num["NumberOfFollowups"].median())

df_tourism2_num["NumberOfTrips"]=df_tourism2_num["NumberOfTrips"].fillna(df_tourism2_num["NumberOfTrips"].median())

df_tourism2_num["NumberOfChildrenVisited"]=df_tourism2_num["NumberOfChildrenVisited"].fillna(df_tourism2_num["NumberOfChildrenVisited"].median())

df_tourism2_num["MonthlyIncome"]=df_tourism2_num["MonthlyIncome"].fillna(df_tourism2_num["MonthlyIncome"].median())

Now we are going to check missing values only for numerical variable by df_tourism2_num.isnull().sum()

```
CustomerID                  0
Age                         0
DurationOfPitch             0
NumberOfPersonVisited       0
```

```
NumberOfFollowups          0
NumberOfTrips              0
PitchSatisfactionScore     0
NumberOfChildrenVisited    0
MonthlyIncome              0
dtype: int64
```

There are no missing values present in data after imputation.

Missing values imputation for categorical variable by mode().

df_mode=df_tourism2_cat["PreferredLoginDevice"].mode()[0]

df_mode

```
'Self Enquiry'
```

df_mode1=df_tourism2_cat["PreferredPropertyStar"].mode()[0]

df_mode1

```
'3 Star'
```

df_tourism2_cat["PreferredLoginDevice"]=df_tourism2_cat["PreferredLoginDevice"].replace(np.nan,df_mode)

df_tourism2_cat["PreferredPropertyStar"]=df_tourism2_cat["PreferredPropertyStar"].replace(np.nan,df_mode1)

Let’s check missing values for categorical variable after imputation:

```
ProdTaken                0
PreferredLoginDevice     0
CityTier                 0
Occupation               0
Gender                   0
ProductPitched           0
PreferredPropertyStar    0
MaritalStatus            0
Passport                 0
OwnCar                   0
Designation              0
dtype: int64
```

There are no missing values present in the data after imputation.

After treating the missing values we will concate the numerical and categorical variables and create a new data frame df_tourism2.

df_tourism2 = pd.concat([df_tourism2_cat,df_tourism2_num],axis=1)

Checking of unique values present in categorical columns:

```
ProdTaken
No     3968
Yes     920
```

```
Name: ProdTaken, dtype: int64

PreferredLoginDevice
Self-Enquiry       3469
Company Invited    1419
Name: PreferredLoginDevice, dtype: int64

CityTier
Tier-1    3190
Tier-3    1500
Tier-2     198
Name: CityTier, dtype: int64

Occupation
Salaried          2368
Small Business    2084
Large Business     434
Free Lancer          2
Name: Occupation, dtype: int64

Gender
Male       2916
Female     1817
Fe Male     155
Name: Gender, dtype: int64

ProductPitched
Multi           1842
Super Deluxe    1732
Standard         742
Deluxe           342
King             230
Name: ProductPitched, dtype: int64

PreferredPropertyStar
3 Star    3019
5 Star     956
4 Star     913
Name: PreferredPropertyStar, dtype: int64

MaritalStatus
Married      2340
Divorced      950
Single        916
Unmarried     682
Name: MaritalStatus, dtype: int64

Passport
No     3466
Yes    1422
Name: Passport, dtype: int64

OwnCar
Yes    3032
No     1856
Name: OwnCar, dtype: int64

Designation
```

```
Executive          1842
Manager            1732
Senior Manager      742
AVP                 342
VP                  230
Name: Designation, dtype: int64
```

Observations:

Here we can see, total count of each labelled categorical variable. **Something that we found here, there is unstructured label Fe Male in Gender column seems like bad data with 155 records**. We need to take care of this. We should replace Fe Male by Female for further analysis. Only 2 records of Free Lancer. They are 100 % probability that they sell their product.

```
df_tourism2['Gender'] = df_tourism2['Gender'].apply(lambda x: 'Female' if x == 'Fe Male' else x)
```

Checking of propensity in target variable:

```
No      0.811784
Yes     0.188216
Name: ProdTaken, dtype: float64
```

Only 18% customers is going to opt long term travel package. Also, this is an imbalance data set because no of 1's is more than 0's.

Checking of duplicates rows: checking of duplicates rows by df_tourism2.duplicated().

```
total no of duplicates rows 0
```

Removal of unwanted variable: Here , no need of CustomerID column for analysis.

```
df_tourism2=df_tourism2.drop(["CustomerID"],axis=1)
```

Also features NumberOfTrips,NumberOfChildrenVisited,NumberOfPersonVisited,NumberOfFollowUps are having fraction values. So we should take it as round number before doing visualization for better understanding.

Let's separate out all the numerical variables and categorical variables from df_tourism2 data set.

df_tourism2_num

```
Index(['Age', 'DurationOfPitch', 'NumberOfPersonVisited',
       'NumberOfFollowups', 'NumberOfTrips', 'PitchSatisfactionScore',
       'NumberOfChildrenVisited', 'MonthlyIncome'],
      dtype='object')
```

df_tourism2_cat

```
Index(['ProdTaken', 'PreferredLoginDevice', 'CityTier', 'Occupation', 'Gender',
```

```
       'ProductPitched', 'PreferredPropertyStar', 'MaritalStatus', 'Passp
ort','OwnCar', 'Designation'], dtype='object')
```

Let’s do univariate analysis for all numerical and categorical variables. The data set that are used in below univariate analysis is df_tourism2.

Univarate analysis for Numerical Variable:

Univariate analysis by using distplot() and boxplot() for each and every numerical variables.

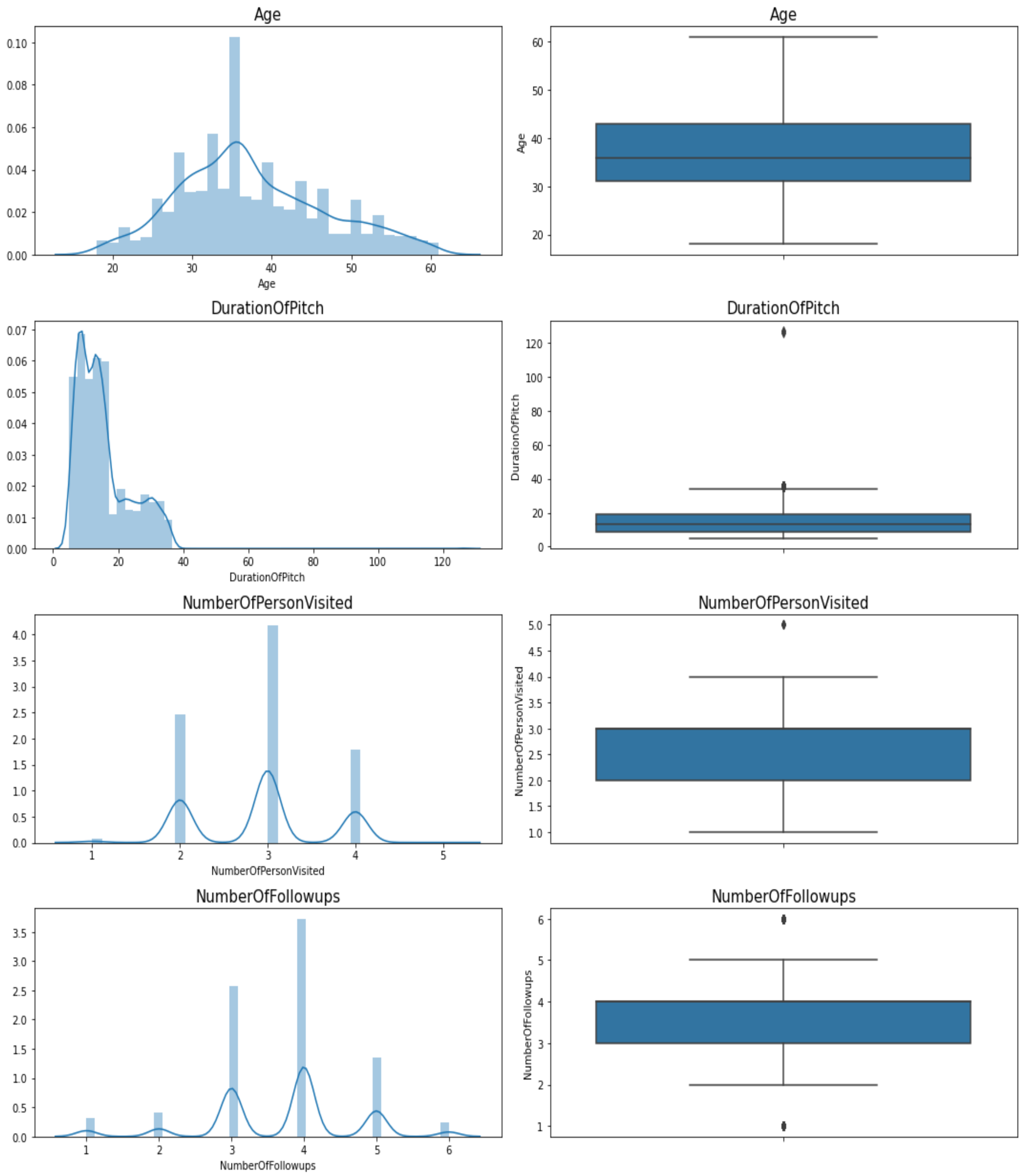

Fig-2

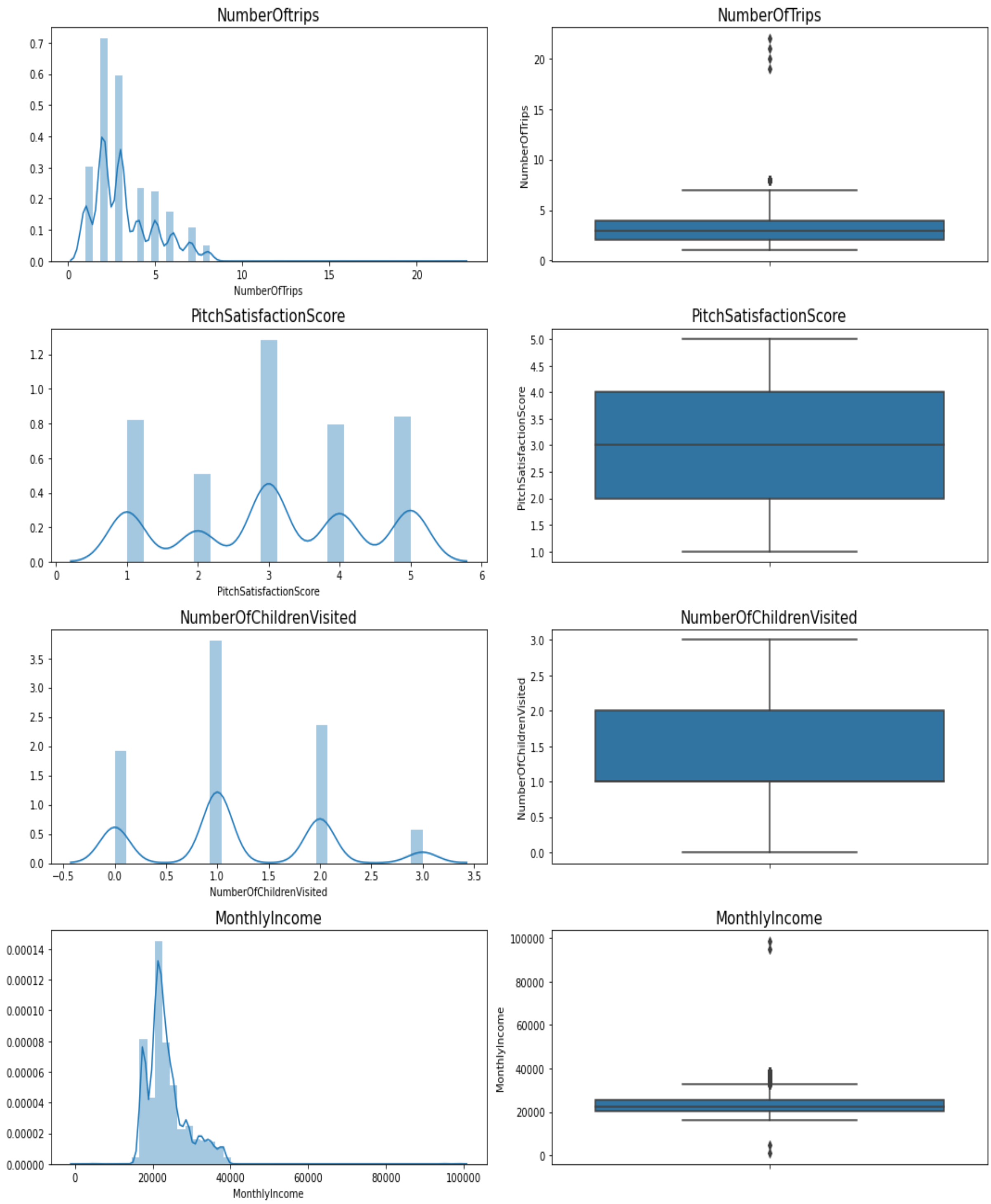

Fig-3

Observations:

- Among all the above numeric variables, only Age is having unimodal distr-ibution (single peak) .So we can say Age is normally distrusted. Others nu merical variables are having multimodal distribution(Multiple peak). Since this is a classification problem, we can choose to leave such variables as they are. To get rid of such multimodal distribution, we can use Binning approach wherein we can create buckets.

- Also there are outliers present in variables NumberOfFollowUps,NumberOfPersonVisited,NumberOfTrips,MonthlyIncome.There is different approaches to handle outlier. We can remove outlier, retain outlier and can do imputation also. This totally depends upon business problem that we are dealing. We will do it later for further analysis.
- 50% customers are in age 35-36(younger age group) and their monthly income in rage of 21000 to 23000.

Univariate Analysis of all Categorical Variables:

Univariate analysis of all categorical variable by countplot().

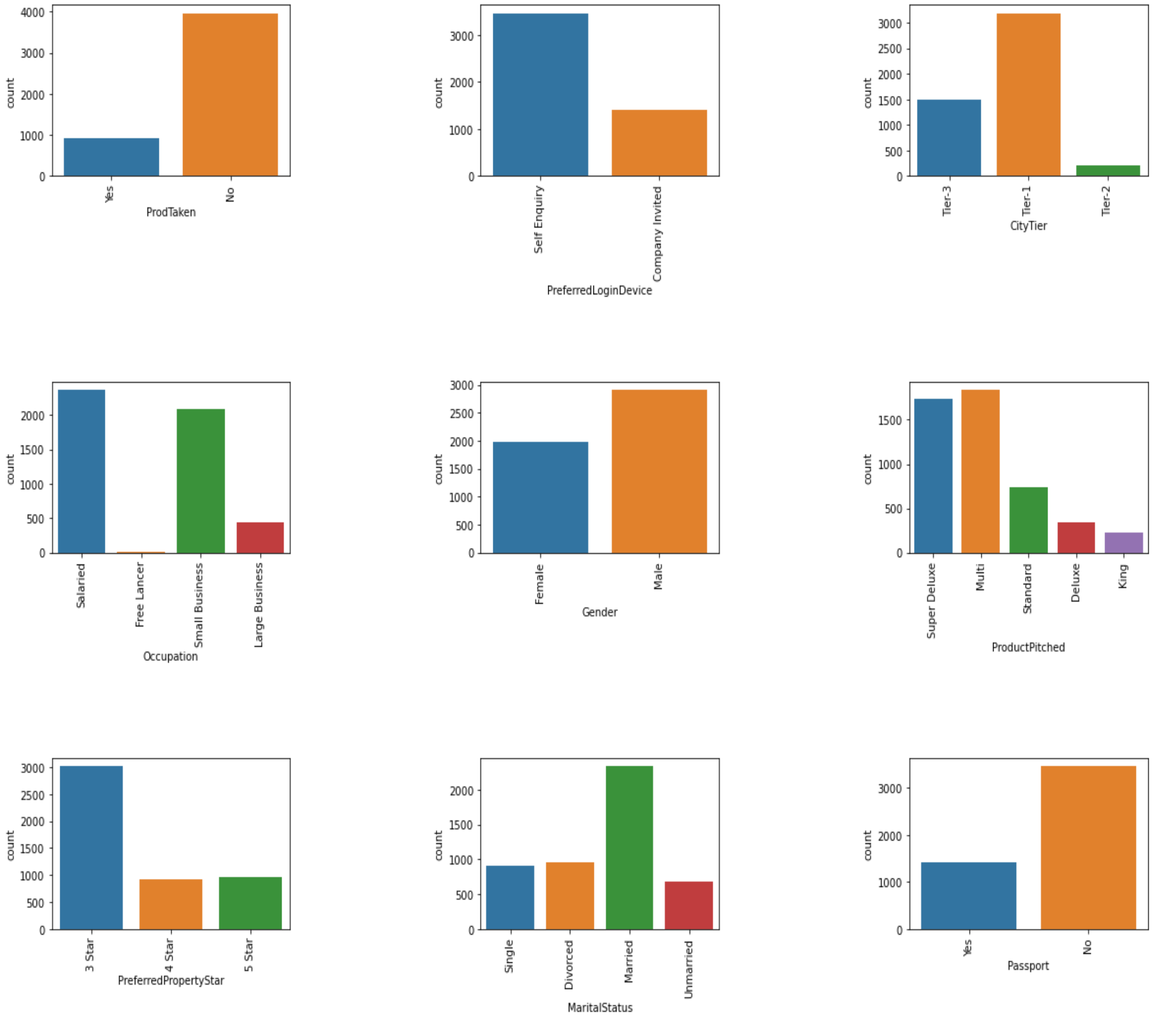

Fig-4

Observations:

- Most of the customers are not taken product.
- Most of the customers come up by themselves.
- Most of the customers do not have passport.
- Most of the customers are gender.
- Most of the customers are taken super deluxe and multi package.
- Most of the customers prefer to stay in 3-Star.
- Most of the customers belong to Tier-1.
- Most of the customer's occupations are salaried and small business.

Conclusion:

- From the above inferences of the categorical variable, we can conclude th at most of customers live in metropolitan city.
- Since the customers belong to small occupation (salaried and small busin es), hence we can conclude that, they have small monthly income, they c annot afford more no of trips, may be they will buy cheaper product and Super Deluxe and multi product.
- Since most of the customers do not have passport, so we can conclude m ost of the customers are domestic traveller.

Feature Engineering:

To get rid from multimodal distribution from that is present in numerical variables in df_tourism2 data set, we are going to use **Binning.** For the sake of further analysis we have taken df_tourism3 data set. This comes under **feature engineering and it is itself divide into two parts:**

1. Variable Transformation

2. Variable Creations

There are many approaches that are used in variable transformation and variable creation. Binning is one of the approach that I have used in variable transformation.

Binning Approach: Binning method is used to smoothing data or to handle noisy data. In this method, the data is first sorted and then the sorted values are distributed into a number of buckets or bins. As binning methods consult the nei-ghborhood of values, they perform local smoothing.

When dealing with continuous numeric data, it is often helpful to bin the data into multiple buckets for further analysis. There are several different terms for binning including bucketing, discrete binning, discretization or quantization. Pandas supports these approaches using the cut and qcut functions.

Binning using quartiles: durationOfPitch:

This approach describes as a "Quantile-based discretization function." This basically means that qcut tries to divide up the underlying data into equal sized bins. The function defines the bins using percentiles based on the distribution of the data, not the actual numeric edges of the bins.

Let's check descriptive statistics of variable DurationOfPitch:

```
count    4888.000000
mean       15.362930
std         8.316166
min         5.000000
25%         9.000000
50%        13.000000
75%        19.000000
max       127.000000
Name: DurationOfPitch, dtype: float64
```

After that we will create binning variable durationOfPitch_bins:

```
Really Low    1471
High          1199
Low           1118
Medium        1100
Name: DurationOfPitch_bins, dtype: int64
```

Here we have done labelling on the behalf of quartiles like from range min to25% named as Really Low, from range 25% to 50% named as Low, from 50% to 75% named as Medium and from 75% up to max named as High.

Binning using quartile:NumberOfFollowups:

```
Medium    2081
Low       1903
High       904
Name: NumberOfFollowups_bins, dtype: int64
```

Binning using quartiles: NumberOfTrips:

```
Low          2084
Very High    1114
Medium       1081
High          609
Name: NumberOfTrips_bins, dtype: int64
```

Binning using map function: PitchSatisfactionScore:

```
Good            1478
Excellent        970
Bad              942
Very Good        912
OK               586
Name: PitchSatisfactionScore_bins, dtype: int64
```

Binning using lambda function: NumberOfPersonVisited

```
Three and above    3431
One or Two         1457
Name: NumberOfPersonVisited_bins, dtype: int64
```

Checking of data types after binning:

After bucketing, we have to drop all the variables that are used for binning. Now, we will check info() of data to check the new variables that are created in binning and also data types of each new variables.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4888 entries, 0 to 4887
Data columns (total 19 columns):
 #   Column                         Non-Null Count  Dtype  
---  ------                         --------------  -----  
 0   ProdTaken                      4888 non-null   object 
 1   PreferredLoginDevice           4888 non-null   object 
 2   CityTier                       4888 non-null   object 
 3   Occupation                     4888 non-null   object 
 4   Gender                         4888 non-null   object 
 5   ProductPitched                 4888 non-null   object 
 6   PreferredPropertyStar          4888 non-null   object 
 7   MaritalStatus                  4888 non-null   object 
 8   Passport                       4888 non-null   object 
 9   OwnCar                         4888 non-null   object 
 10  Designation                    4888 non-null   object 
 11  Age                            4888 non-null   float64
 12  MonthlyIncome                  4888 non-null   float64
 13  DurationOfPitch_bins           4888 non-null   object 
 14  NumberOfPersonVisited_bins     4888 non-null   object 
 15  NumberOfFollowups_bins         4888 non-null   object 
 16  NumberOfTrips_bins             4888 non-null   object 
 17  PitchSatisfactionScore_bins    4888 non-null   object 
 18  NumberOfChildrenVisited_bins   4888 non-null   object 
dtypes: float64(2), object(17)
memory usage: 725.7+ KB
```

Observations:

- Here, we can see, DurationOfPitch_bins,NumberOfPersonVisited_bins,NumberOfFollowups_bins,PitchSatisfactionScore_bins,NumberOfChildrenVi sited_bins are the new variables names that we have created while binning and also all these variables are object type.

- We are left with two numerical variables Age and MonthlyIncome.

Let's do univariate analysis again for numerical variables.

Univariate analysis for all Numerical Variables:

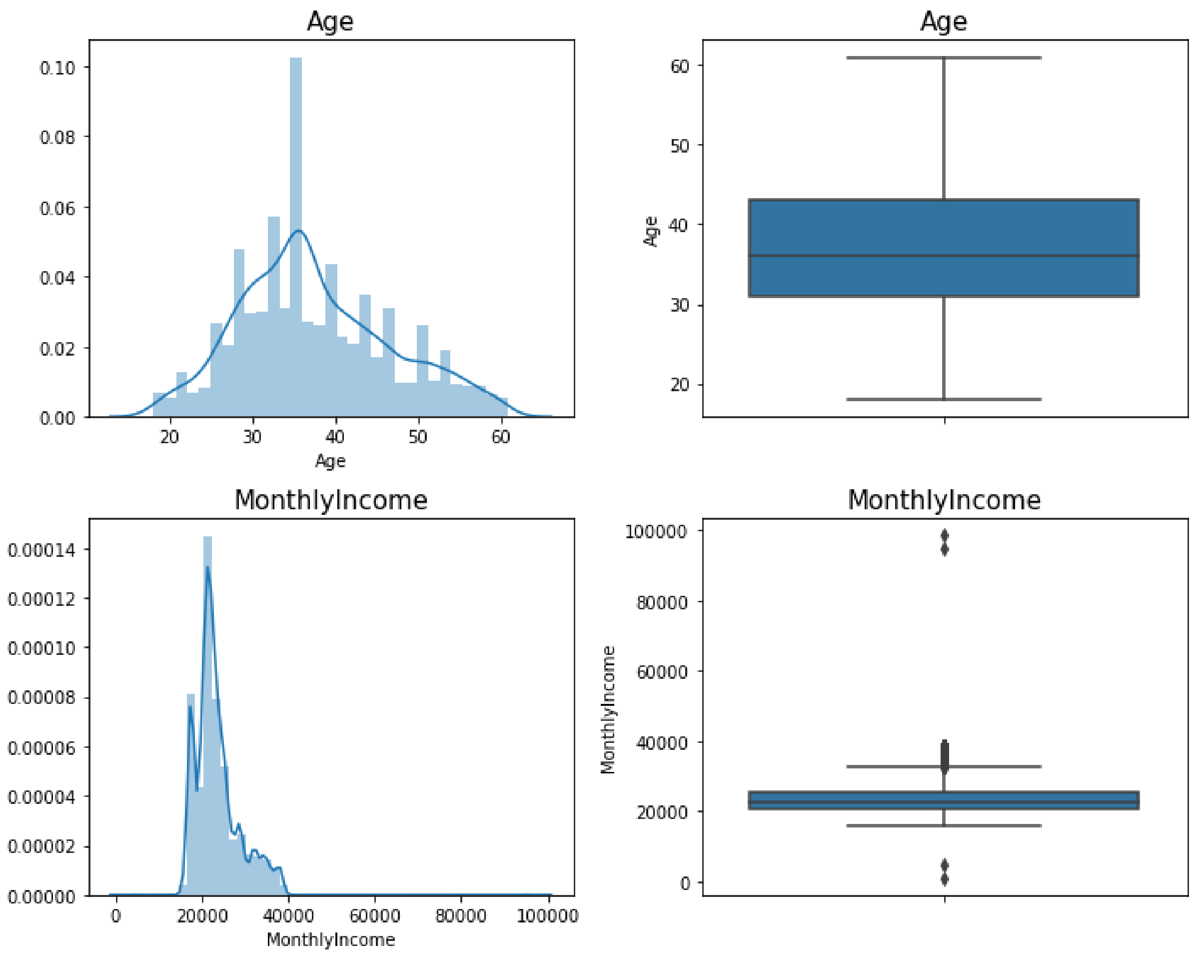

Fig-5

- Age is normally distributed and 50% of customers are in age group35-36(young age group).
- MonthlyIncome is normally distributed.50 % customers are having Mont hlyIncome range between 2000-2200.There are some outliers also in Mo nthlyIncome variables that demonstrate that some customers are having v ery high MonthlyIncome , they might be those customers whose designati on is high and some of customers have very low MonthlyIncome , they might be those customers whose belong to small occupation.

Univariate Analysis of all categorical variables:

Here is Univariate analysis for all categorical variables by using count plot.

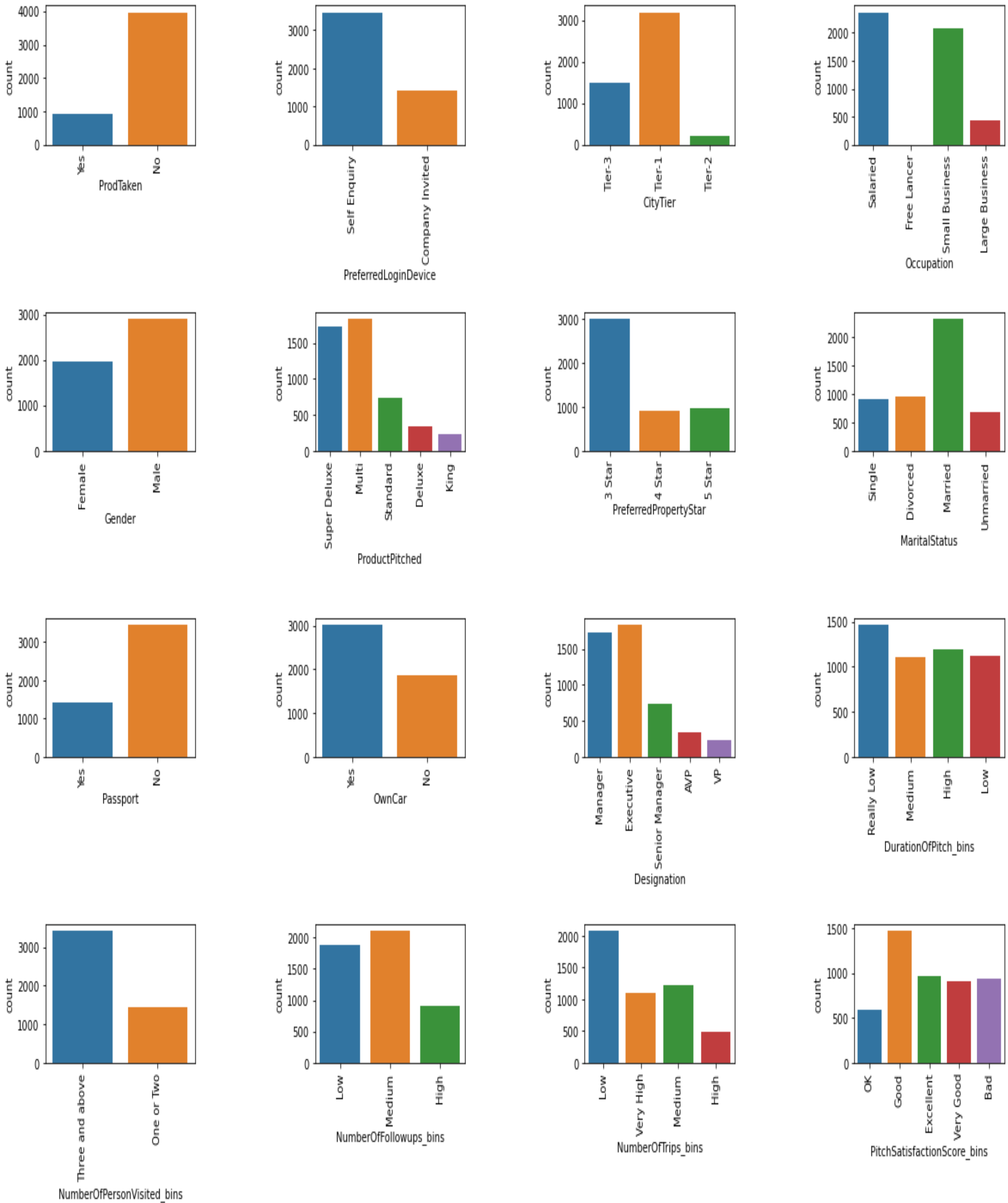

Fig-6

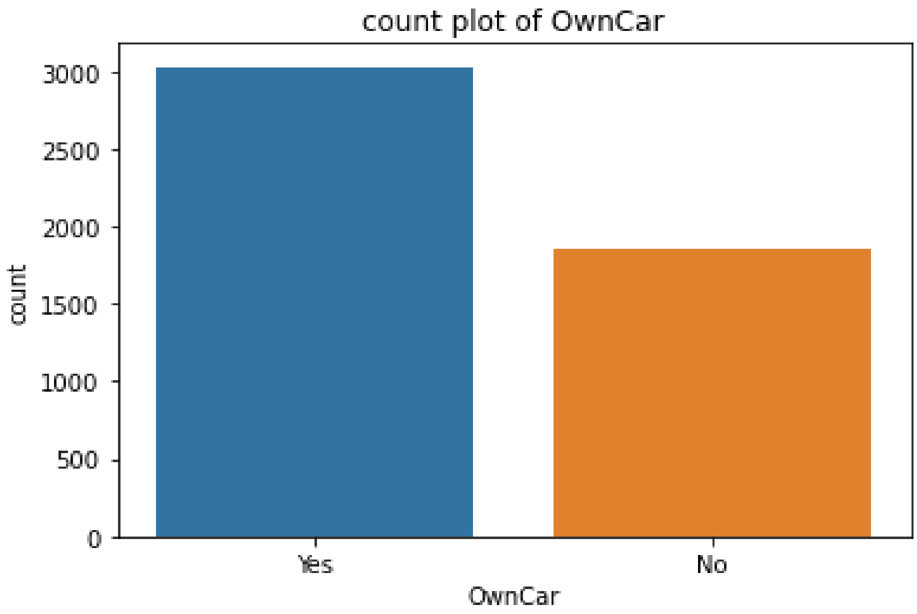

Fig-7

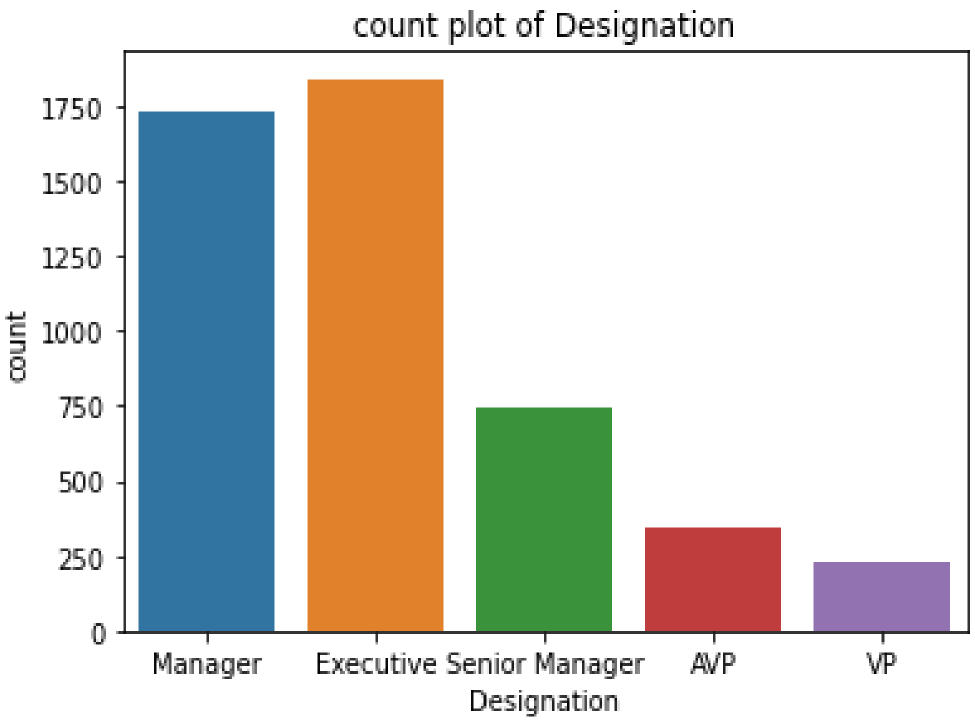

Fig-8

Observations:

- Most of the customers are not taken product.
- Most of the customers come up by themselves.
- Most of the customers belong to city Tier-1(metropolitan city).
- Most of the customers are salaried and have small business.
- Most of the customers are Male.

- Most of the customers are opted multi and Super Deluxe package.
- Most of the customers are preferred to stay in 3-Star.
- Most of the customers are married.
- Most of the customers do not have passport.
- Most of the customers have own car.
- Most of the customer's designations are Executive, Manager.
- Duration of pitch by salesman to customers is really low.
- Most of the customers bring two or three children along with.
- No of follow up is done by sales persons, are medium.
- Most of the customers are done less no of trips in a year.
- Pitch satisfactory score is given by most of the customers, are good.

Conclusion:

- From the above inferences of the categorical variable, we can conclude that most of customers live metropolitan city and they belong to middle /upper middle family and they have probably kids and family and own car as well.
- Since the customers belong to small occupation (salaried and small business), hence we can conclude that, they have small monthly income, they can not afford more no of trips, may be they will buy cheaper product and Super Deluxe and multi product.
- Most of the customer do not have passport, so we can conclude they all are domestic traveller.

Looking at proportion of labelled categorical variable:

```
Proportion of Customers as per ProdTaken

No      0.811784
Yes     0.188216
Name: ProdTaken, dtype: float64

Proportion of Customers as per PreferredLoginDevice

Self Enquiry        0.709697
Company Invited     0.290303
Name: PreferredLoginDevice, dtype: float64

Proportion of Customers as per CityTier

Tier-1    0.652619
Tier-3    0.306874
Tier-2    0.040507
Name: CityTier, dtype: float64
```

```
Proportion of Customers as per Occupation

Salaried          0.484452
Small Business    0.426350
Large Business    0.088789
Free Lancer       0.000409
Name: Occupation, dtype: float64

Proportion of Customers as per Gender

Male      0.596563
Female    0.403437
Name: Gender, dtype: float64

Proportion of Customers as per ProductPitched

Multi           0.376841
Super Deluxe    0.354337
Standard        0.151800
Deluxe          0.069967
King            0.047054
Name: ProductPitched, dtype: float64

Proportion of Customers as per PreferredPropertyStar

3 Star    0.617635
5 Star    0.195581
4 Star    0.186784
Name: PreferredPropertyStar, dtype: float64

Proportion of Customers as per MaritalStatus

Married      0.478723
Divorced     0.194354
Single       0.187398
Unmarried    0.139525
Name: MaritalStatus, dtype: float64

Proportion of Customers as per Passport

No     0.709083
Yes    0.290917
Name: Passport, dtype: float64

Proportion of Customers as per OwnCar
```

```
Yes    0.620295
No     0.379705
Name: OwnCar, dtype: float64

Proportion of Customers as per Designation

Executive         0.376841
Manager           0.354337
Senior Manager    0.151800
AVP               0.069967
VP                0.047054
Name: Designation, dtype: float64

Proportion of Customers as per DurationOfPitch_bins

Really Low    0.300941
High          0.245295
Low           0.228723
Medium        0.225041
Name: DurationOfPitch_bins, dtype: float64

Proportion of Customers as per NumberOfPersonVisited_bins

Three and above    0.701923
One or Two         0.298077
Name: NumberOfPersonVisited_bins, dtype: float64

Proportion of Customers as per NumberOfFollowups_bins

Medium    0.432283
Low       0.382774
High      0.184943
Name: NumberOfFollowups_bins, dtype: float64

Proportion of Customers as per NumberOfTrips_bins

Low          0.426350
Medium       0.249386
Very High    0.226473
High         0.097791
Name: NumberOfTrips_bins, dtype: float64

Proportion of Customers as per PitchSatisfactionScore_bins

Good         0.302373
Excellent    0.198445
Bad          0.192717
```

```
Very Good     0.186579
OK            0.119885
Name: PitchSatisfactionScore_bins, dtype: float64

Proportion of Customers as per NumberOfChildrenVisited_bins

One           0.660393
2 or more     0.339607
Name: NumberOfChildrenVisited_bins, dtype: float64
```

Observations:

- Only 18% customers are taken product.
- 70% customers come up themselves.
- 65% customers belong to Tier-1 (Urban city).
- 33% customers have 2 or more children.
- 70% customers don't have passport.
- 37% customers are working as executive.

Bi-Variate Analysis and Multivariate Analysis:

This is the bivariate analysis across all numerical variables by pairplot() and heatmap().

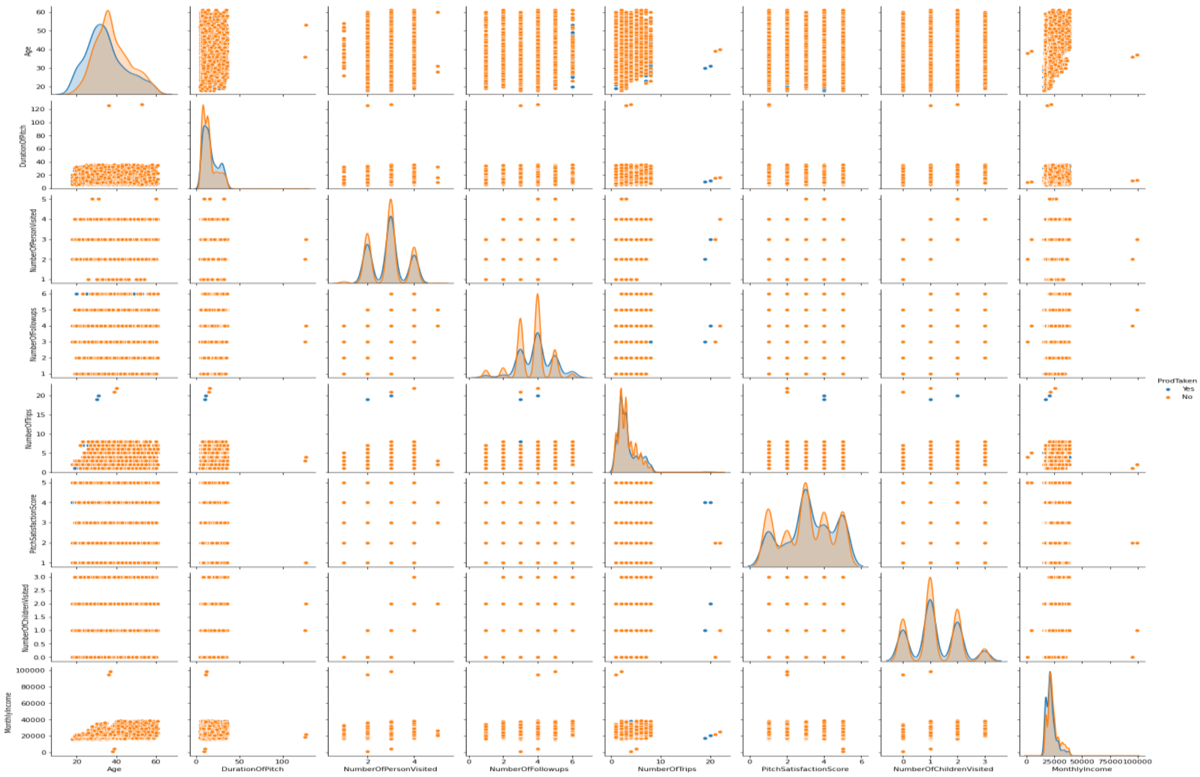

Fig-9

Observations:

There is hardly any correlation.

Bivariate analysis by heatmap().

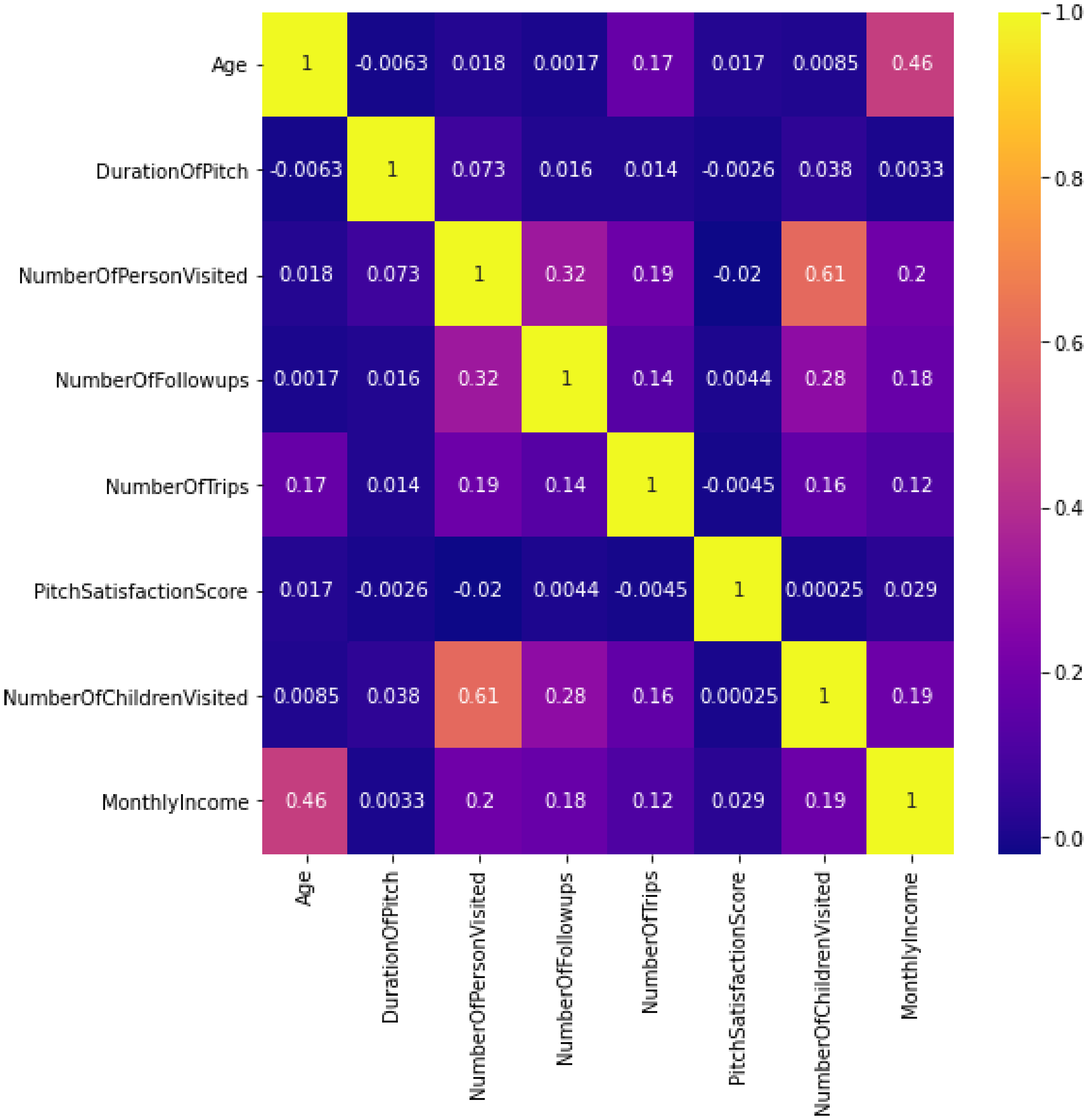

Fig: 10

Observations:

There is only correlation between NumberOfChildrenVisited and NumberOfPersonVisited. As the Number of children visited increases number of person visited also increases.

Let’s see scatterplot b/w numerical variables:

Scatterplot between Age and NumberOfTrips regarding ProdTaken:

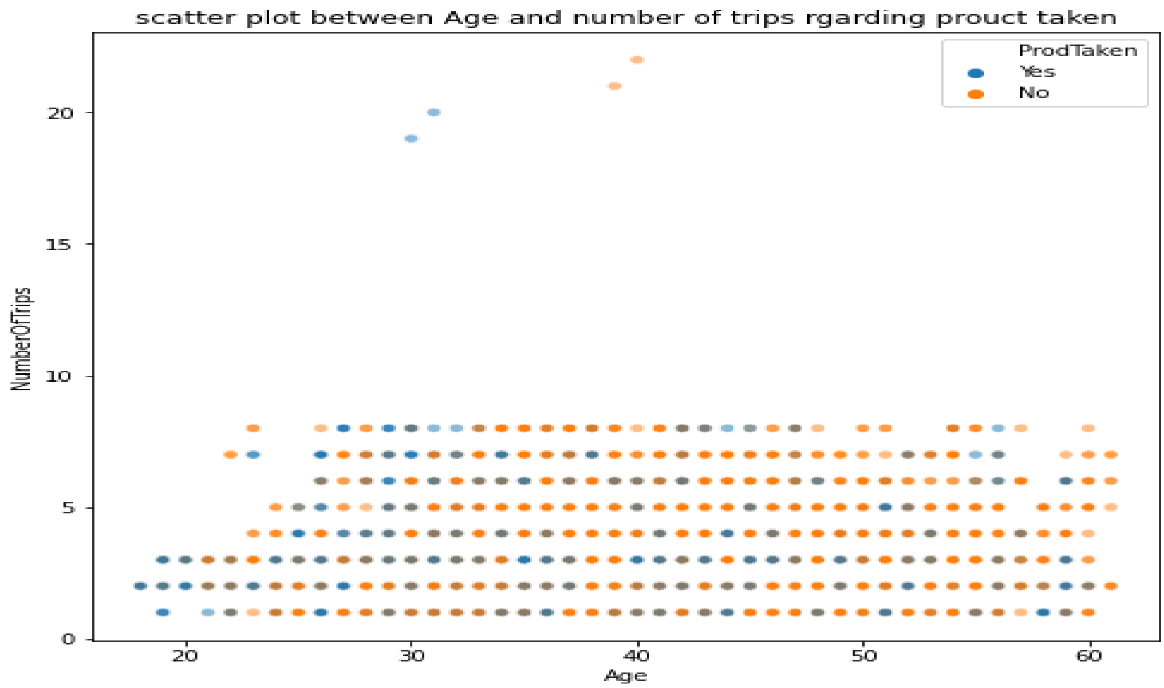

Fig: 11

- We cannot find any correlation.

Scatterplot between MonthlyIncome and Age regarding ProdTaken:

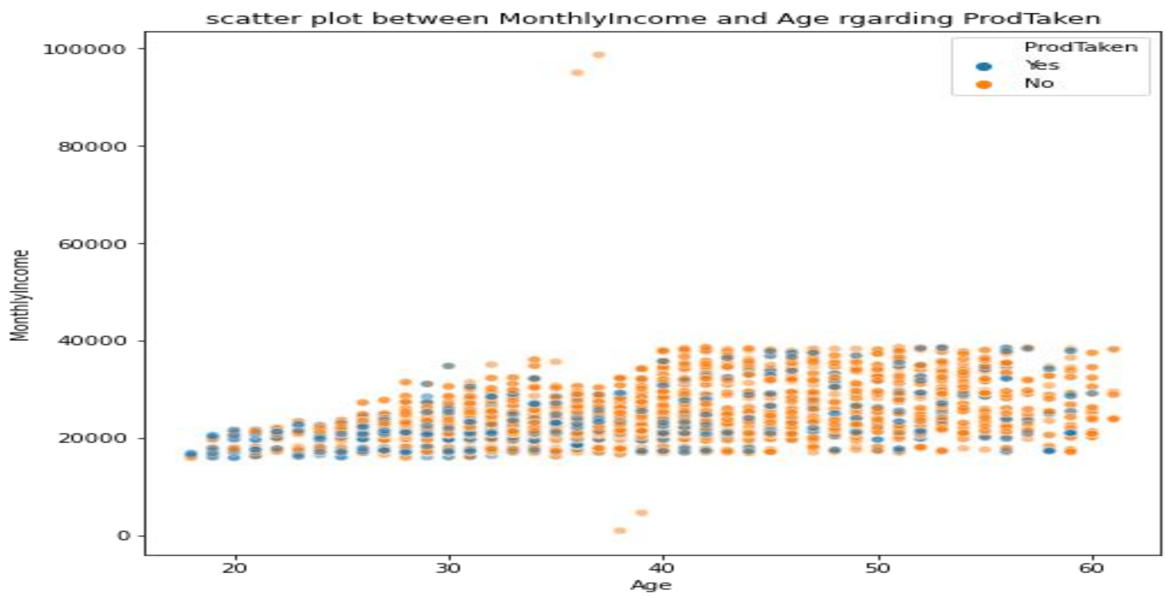

Fig: 12

Observations:

- Concentration of blue dots are high at lower age group people and inco me range 20000-40000.
- Most of the customers that are taken product belong to young age and m iddle age group and they have monthly income 2000 to 40000. After 40 years of age monthly income of the customers are saturated. We can con clude that they all are the customers who belong to higher designation a nd their monthly income is high as well.

Scatterplot between NumberOfPersonVisited and Age regarding ProdTaken:

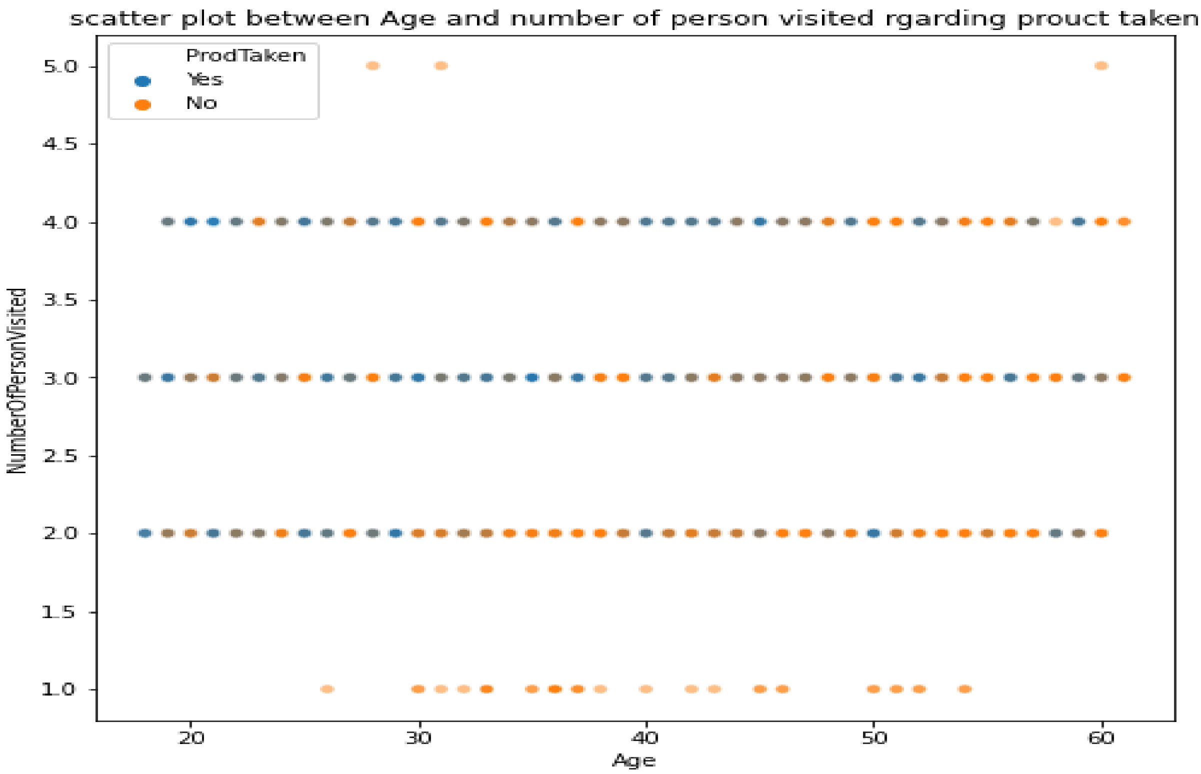

Fig: 13

Observations:

Most of the customers who are taken product belong to younger and middle age group and they belong to small and middle family as well.

Distribution of age across all categorical variable and binned variable:

Let's see distribution of age across categorical and binned variable by boxplot().

Age across all categorical variable and binned variable

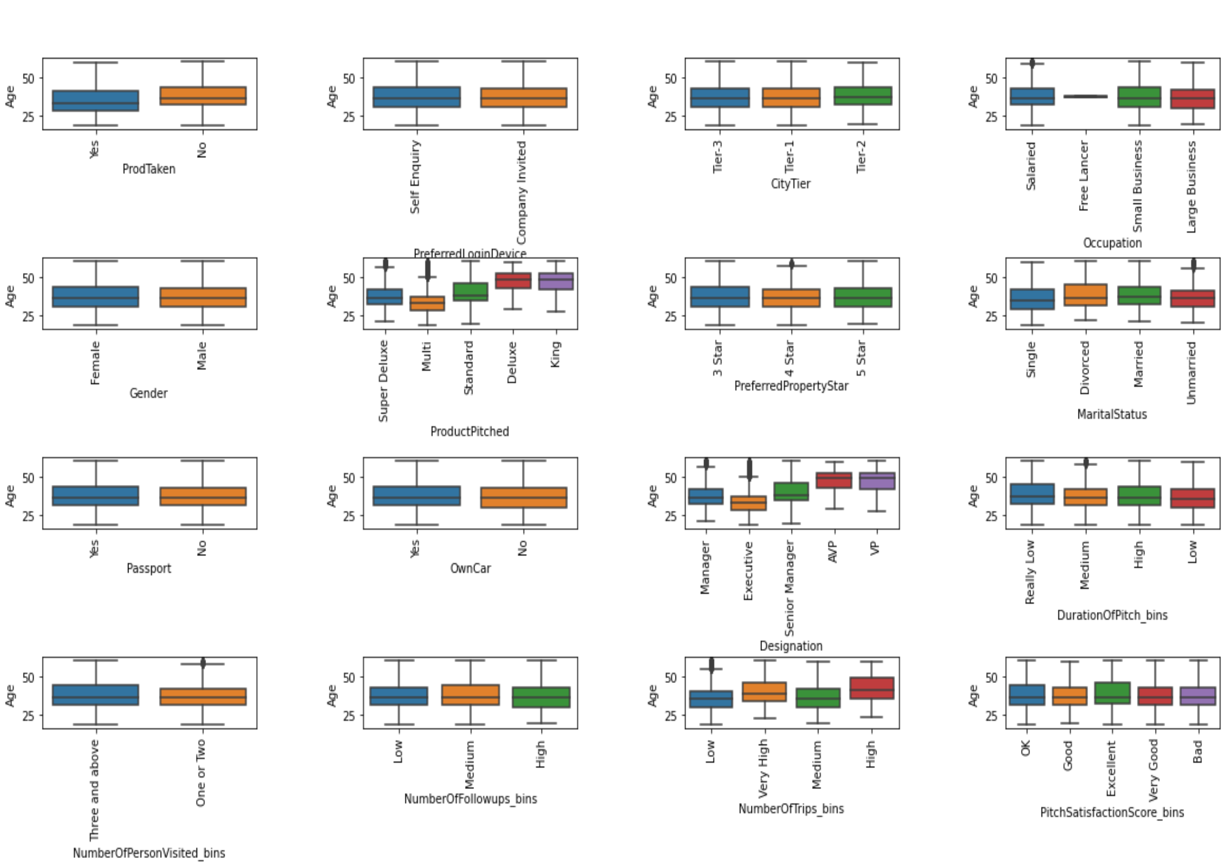

Fig: 14

Observations:

- Median age of the people who has taken product is lesser than who ha s no taken. It means, the people, who has taken product are lower age group.
- Customers, who are younger group and middle age group, both come up by themselves and by company invitation.
- 50% of customers whose age above 36+ belong to Tier-2(urban city).
- The customers those occupations are salaried, small business and larg e business, belong to almost same age group that is middle age group.
- The people who are of higher age group have little more income and t heir designation is also high. They are working as AVP, VP. That's w hy they are pitching the product Deluxe and king.
- The people who are of higher age group, their number of trips are mo re because their monthly income and designation is high.
- 50% of customers who belong to middle age group are married.
- The people, who belong to old age group and middle age group they a re buying expensive product and average range of product like Standa rd, Deluxe and King. Also there are some outliers are also present in

SuperDeluxe and multi product that depicts that some of the older age customers are also buying cheaper product.

Distribution of monthly income across categorical and binned variables:

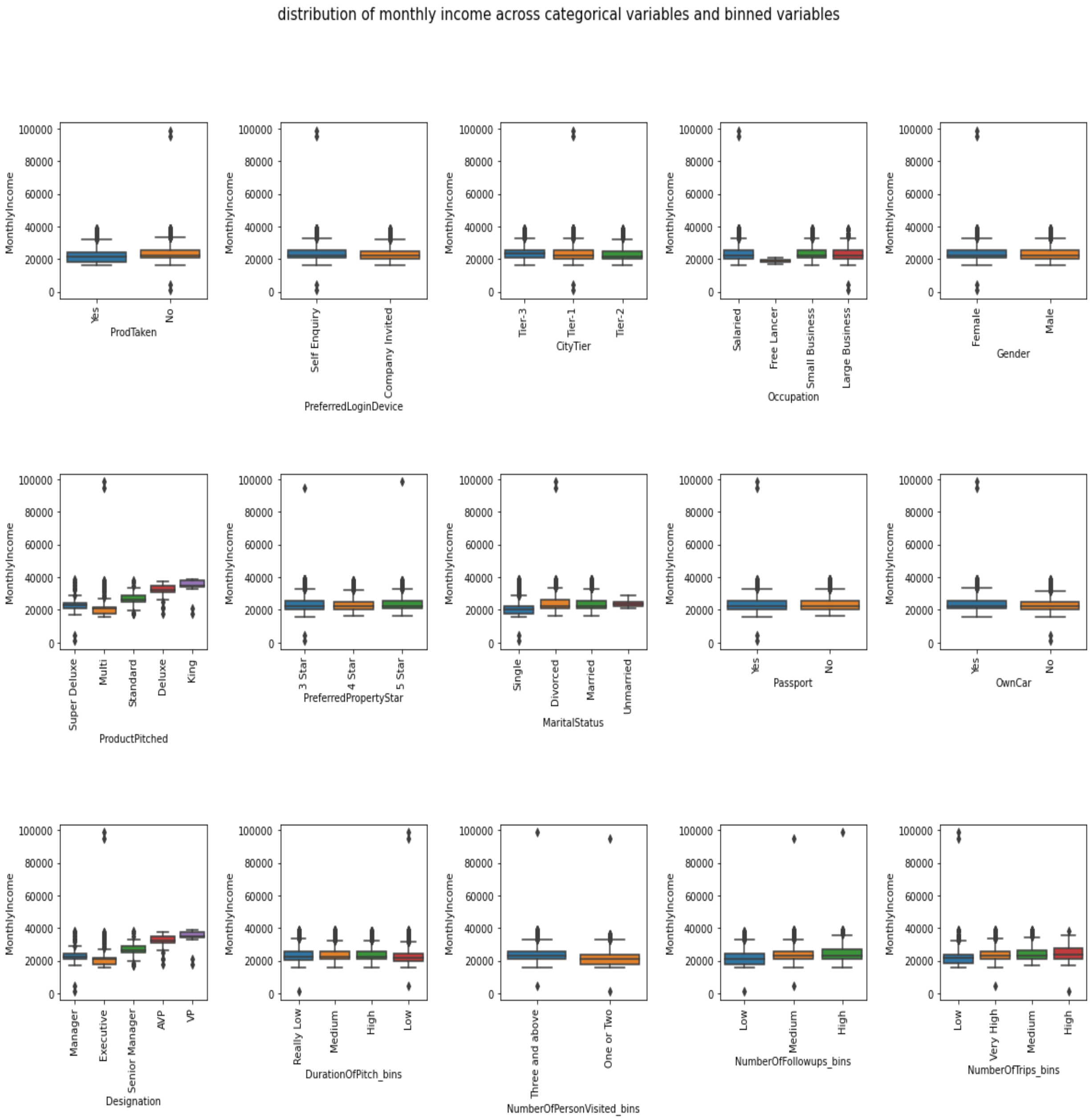

Fig: 15

Observations:

- ➢ Monthly income is higher of those customers who have taken produc t Deluxe and King.
- ➢ Monthly income is higher of those customers who are working as A VP and VP.
- ➢ Monthly income is higher of those customers who are doing more

no of trips or more travel.

- Monthly income is slightly lower of the customers who have taken pr oduct than the customers who have not taken.

Distribution of ProdTaken across all categorical and binned variables:

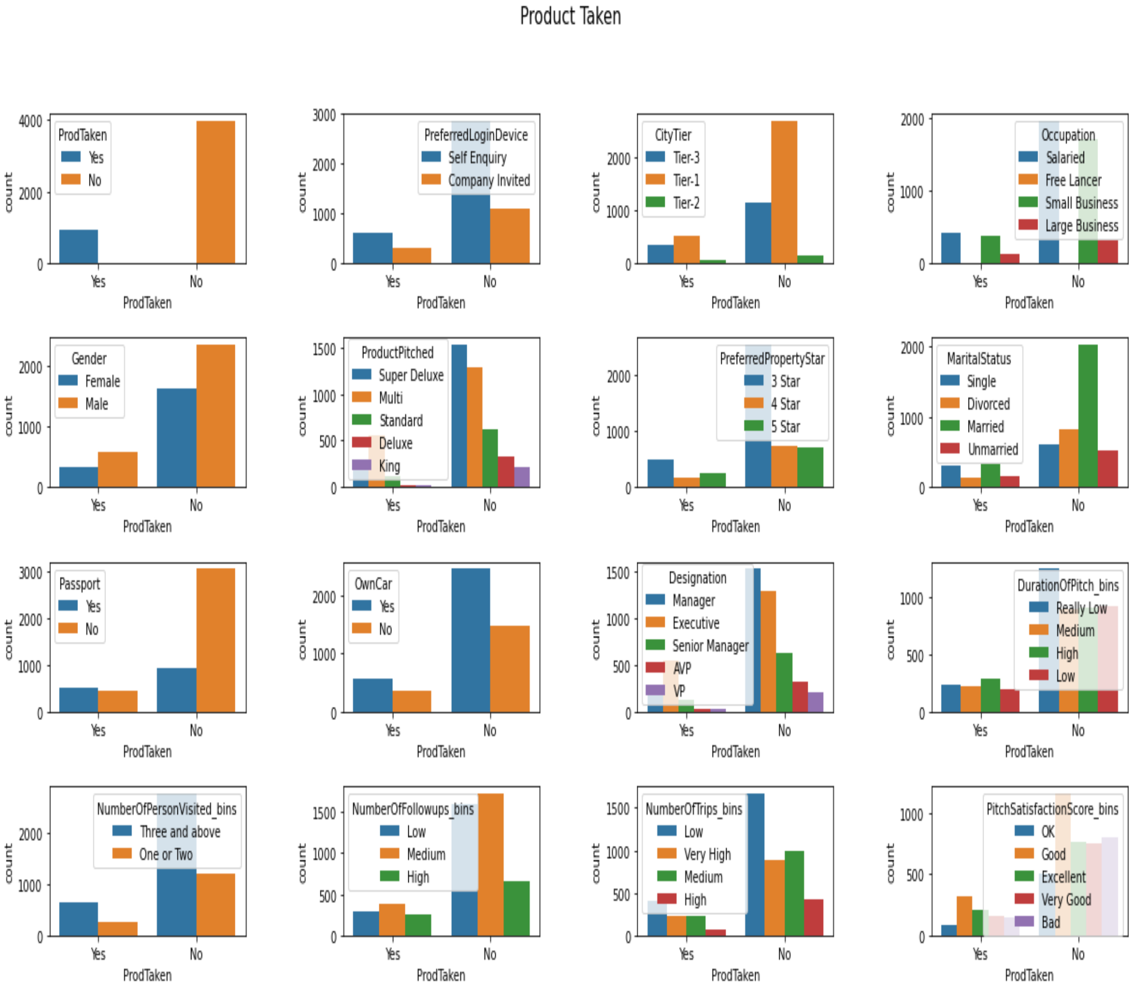

Fig: 16

Observations:

- Product taken is more by the customers who have passport, may be t hey can travel outside of the country. The customers who are taken p roduct and do not have passport, are local traveller.
- Product taken is more by those customer who has own car
- Product taken is more by those customers who are working as execut ive.
- Product taken is more by the customers who live in Tier-1 city(Metro politan city)

- Product taken is more by the customers who is visiting with three an d above three people.
- The customers who have taken product, their DurationOfPitch_bins is high.
- The product taken is more by the customers who stay in 3-Star hotel.

Distribution of Age and ProdTaken across different categorical variables:

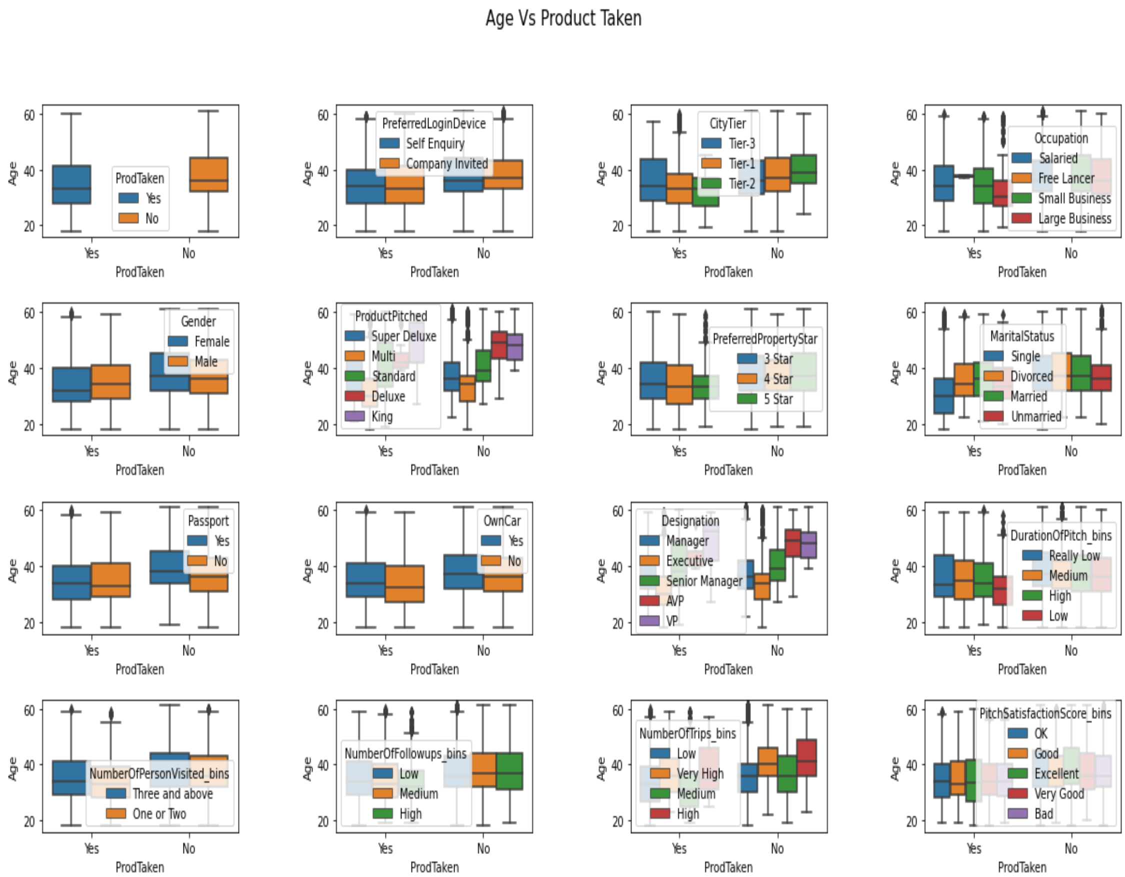

Fig: 17

Observations:

- The customers who are single and belong to age rage 50 to 60 have h igh income. They can spend money to buy Deluxe and King type of product.
- The product taken by the customers is more who preferred to stay in 3 star hotel and their average age is 35 to 36.
- The most of the product is taken by younger and middle age group of customers who belong to small occupation.

- They are 2 Free Lancer and they are taken product also. They will definitely sell their product to customers.
- The most of the product is taken by those middle age group customer whose travelling is more.
- The product taken is more of the customers who are younger and have passport.
- The product taken is more, of the customers who belong to middle age group and married.

In next step, we are going to do variable cluster analysis. In which, we will do feature creation also. This is also a part of feature engineering.

.

Cluster Analysis:

1.First take the df_tourism4 data set .This is the copy of df_tourism2 data set.

2. Let's check data types and variables by info().

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4888 entries, 0 to 4887
Data columns (total 19 columns):
 #   Column                  Non-Null Count  Dtype  
---  ------                  --------------  -----  
 0   ProdTaken               4888 non-null   object 
 1   PreferredLoginDevice    4888 non-null   object 
 2   CityTier                4888 non-null   object 
 3   Occupation              4888 non-null   object 
 4   Gender                  4888 non-null   object 
 5   ProductPitched          4888 non-null   object 
 6   PreferredPropertyStar   4888 non-null   object 
 7   MaritalStatus           4888 non-null   object 
 8   Passport                4888 non-null   object 
 9   OwnCar                  4888 non-null   object 
 10  Designation             4888 non-null   object 
 11  Age                     4888 non-null   float64
 12  DurationOfPitch         4888 non-null   float64
 13  NumberOfPersonVisited   4888 non-null   int64  
 14  NumberOfFollowups       4888 non-null   float64
 15  NumberOfTrips           4888 non-null   float64
 16  PitchSatisfactionScore  4888 non-null   int64  
 17  NumberOfChildrenVisited 4888 non-null   float64
 18  MonthlyIncome           4888 non-null   float64
dtypes: float64(6), int64(2), object(11)
memory usage: 725.7+ KB
```

There are two types of categorical variable in the data set wherein some are or ordinal like ProductPitched,PreferredPropertyStar, Designation which is ranked based and rest of all are categorical where weightage are equal for all different label .

3. For sake of clustering, we need to convert all categorical variables into numerical. For ordinal categorical variable we will use map and lambda function or Categorical().code and other categorical variable we will use one hot encoding and or dummy variable creation.

```
df_tourism4['ProductPitched_codes'] = df_tourism4['ProductPitched'].map({'Multi':1,'Standard':2,'Deluxe':3,'Super Deluxe':4,'King':5})
df_tourism4.drop('ProductPitched',inplace=True,axis=1)
df_tourism4['PreferredPropertyStar_codes'] = df_tourism4['PreferredPropertyStar'].map({'3 Star':1,'4 Star':2,'5 Star':3})
df_tourism4.drop('PreferredPropertyStar',inplace=True,axis=1)
df_tourism4['Designation_codes'] = df_tourism4['Designation'].map({'Executive':1,'Manager':2,'Senior Manager':3,'AVP':4,'VP':5})
df_tourism4.drop('Designation',inplace=True,axis=1)

df_tourism4_cat = df_tourism4[categorical3]
df_tourism4_dummies = pd.get_dummies(df_tourism4_cat)
```

Let’s check info() of data:

```
0   Age                                  4888 non-null   float64
 1   DurationOfPitch                     4888 non-null   float64
 2   NumberOfPersonVisited               4888 non-null   int64
 3   NumberOfFollowups                   4888 non-null   float64
 4   NumberOfTrips                       4888 non-null   float64
 5   PitchSatisfactionScore              4888 non-null   int64
 6   NumberOfChildrenVisited             4888 non-null   float64
 7   MonthlyIncome                       4888 non-null   float64
 8   ProductPitched_codes                4888 non-null   int64
 9   PreferredPropertyStar_codes         4888 non-null   int64
 10  Designation_codes                   4888 non-null   int64
 11  ProdTaken_No                        4888 non-null   uint8
 12  ProdTaken_Yes                       4888 non-null   uint8
 13  PreferredLoginDevice_Company Invited  4888 non-null   uint8
 14  PreferredLoginDevice_Self Enquiry   4888 non-null   uint8
 15  CityTier_Tier-1                     4888 non-null   uint8
 16  CityTier_Tier-2                     4888 non-null   uint8
 17  CityTier_Tier-3                     4888 non-null   uint8
 18  Occupation_Free Lancer              4888 non-null   uint8
 19  Occupation_Large Business           4888 non-null   uint8
 20  Occupation_Salaried                 4888 non-null   uint8
 21  Occupation_Small Business           4888 non-null   uint8
 22  Gender_Female                       4888 non-null   uint8
 23  Gender_Male                         4888 non-null   uint8
 24  MaritalStatus_Divorced              4888 non-null   uint8
 25  MaritalStatus_Married               4888 non-null   uint8
 26  MaritalStatus_Single                4888 non-null   uint8
 27  MaritalStatus_Unmarried             4888 non-null   uint8
 28  Passport_No                         4888 non-null   uint8
 29  Passport_Yes                        4888 non-null   uint8
 30  OwnCar_No                           4888 non-null   uint8
 31  OwnCar_Yes                          4888 non-null   uint8
dtypes: float64(6), int64(5), uint8(21)
memory usage: 520.4 KB
```

Here we can see there are lot of new variables.

4. For clustering analysis we need to scale data because features are in different scale, is not allowed in clustering. So, here we will do scaling by StandarScaler() that are available in scikitlearn library.

5. Next, we will apply k_means algorithm for clustering. K-Means clustering is non-hierarchical clustering wherein initially we have to pre specified how many clusters we require before the model run.

6. Now, we will calculate WSS (within sum of square) for n number of clusters. Here we define range of clusters from 1 to 31.Then calculate inertia for each n number of clusters.

7. Let's see elbow curve: WSS plot for n number of clusters.

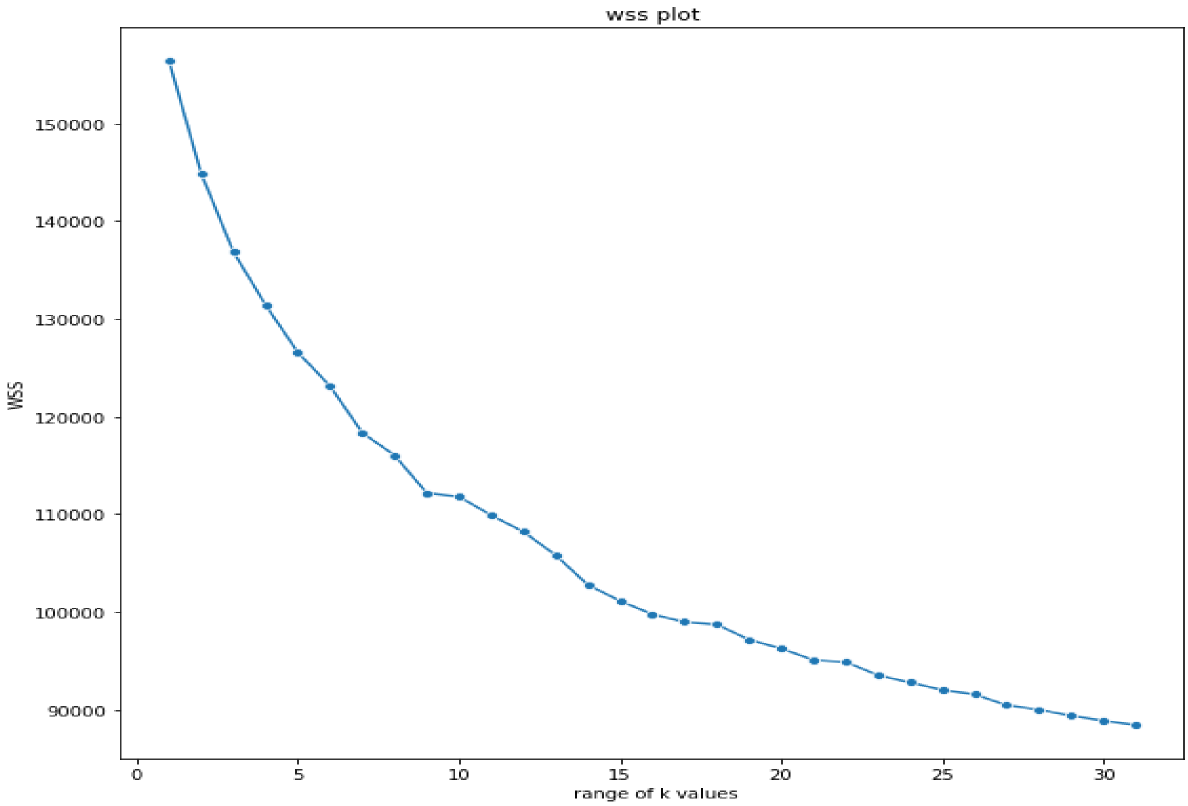

Fig: 18

Here we can see there is significant drop from 1 to 2, 2to3, 3 to4 and 4to5.After 5 very less. We can conclude, 4 and 5 could be optimal number of clusters.

8. Now, we will check silhoutte score for each n number of clusters. This is an indirect model evaluation technique that helps us to analyse whether each and every observation that is mapped to cluster1,clauster2 and cluster3 is actually correct or not based on the distance criteria. Now we will check for what number of clusters

silhouette score is better. Is it 3 or 4? For which we will get silhouette score is bet ter , consider as an optimal number of cluster.

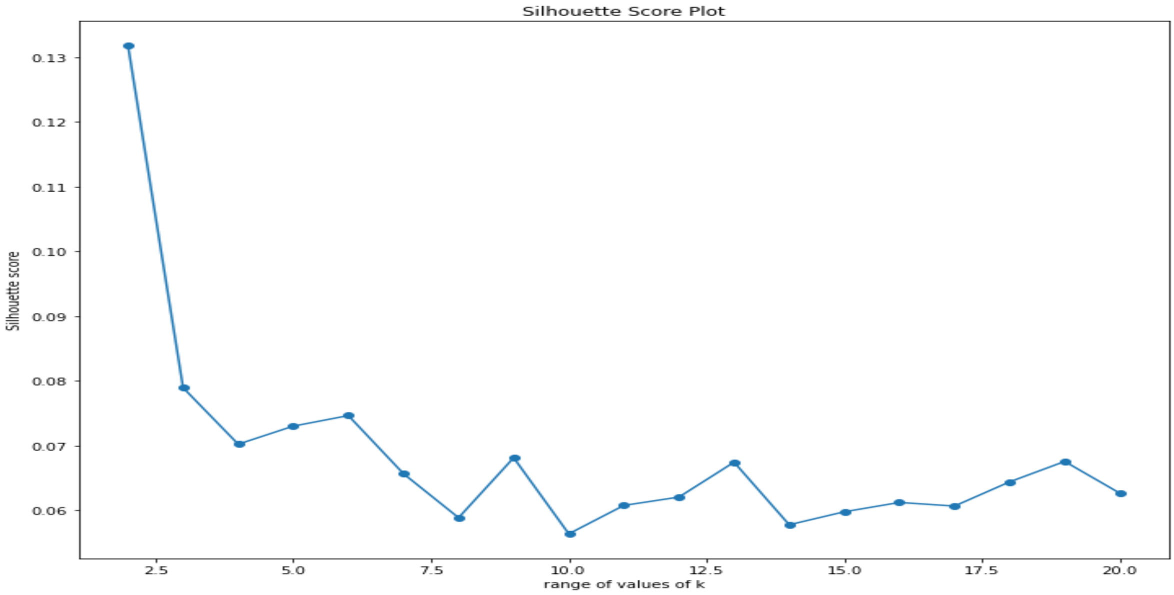

Fig: 19

From the above plot we can say silhoutte score for k=4 is0.07 (approximate) whic h is better than k= 5 is (0.075).Hence k=5 is optimal number of clusters.

9.After getting the optimal number of clusters, we will append the clusters into df _tourism2 and df_tourism3 data set.

Univariate analysis for all clusters across numerical variable:

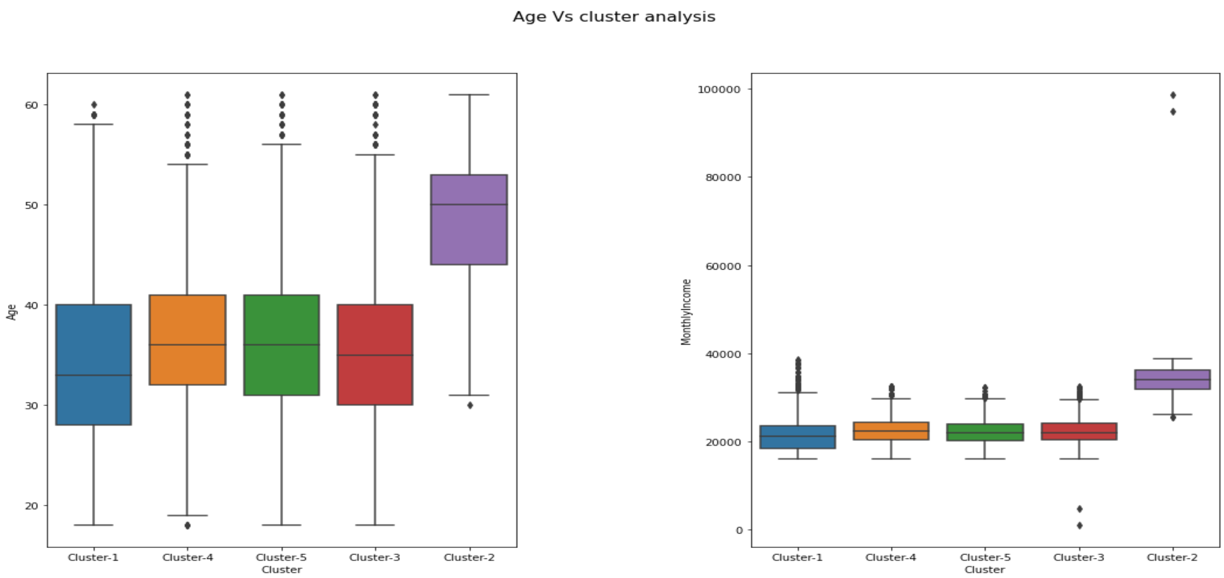

Fig: 20

Observations:

- Cluster-2 is the group of those customers who belong to age 50+(older age group people)
- Cluster-4 is the group of those customers who belong to age group 36 to 40(middle people)
- Cluster-1 is the group of younger to middle age group of customers.
- Cluster-5 is group of younger and middle age group of customers.
- Cluster-2 is the group of those customers whose monthly income is hi gh.
- Cluster-2, Cluster-3, Cluster-4 and cluster-5 are group of those custom ers whose monthly income is in range of 21000-23000.

.

Univariate analysis of clusters across all categorical variable:

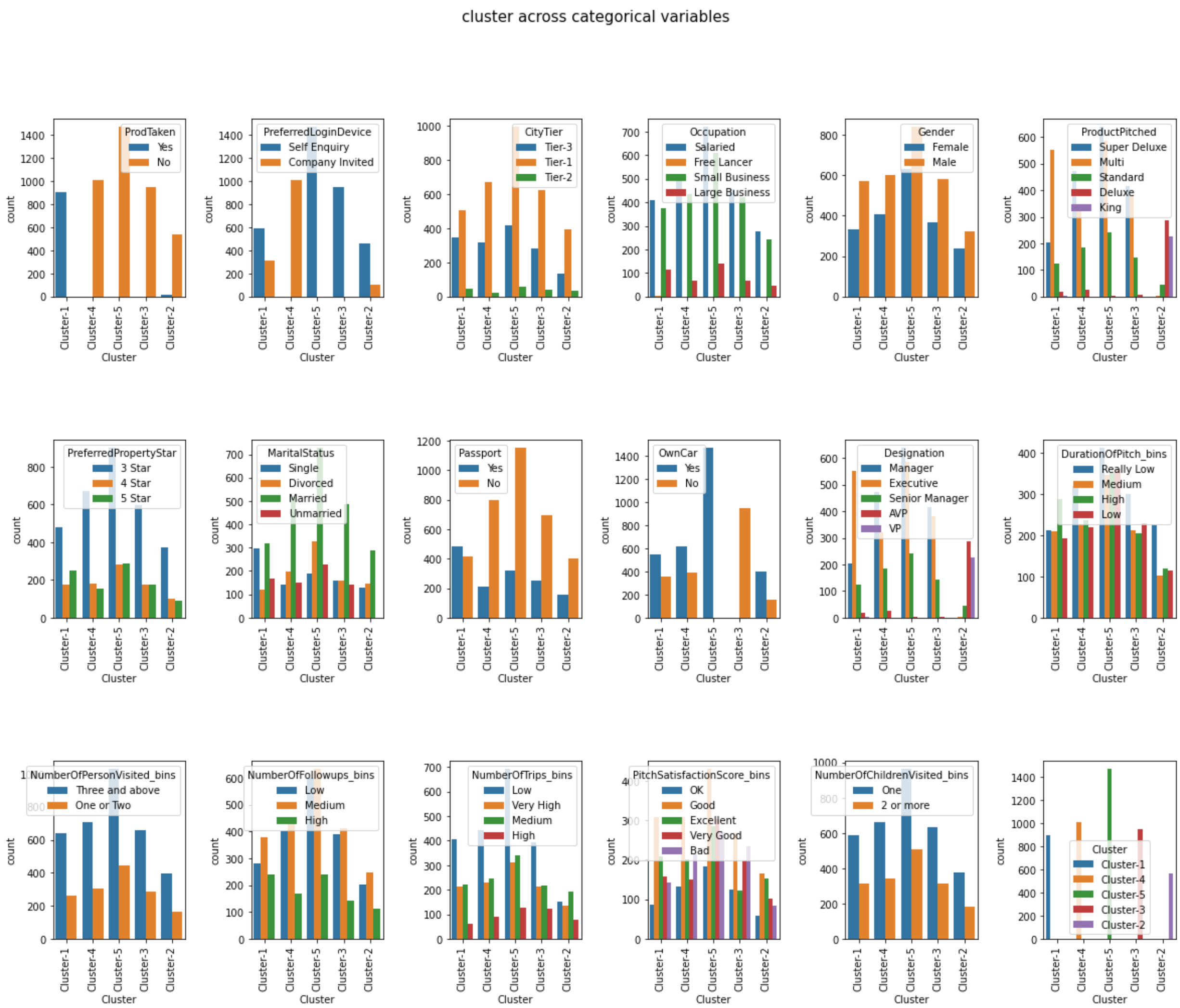

Fig: 21

Let's see the plots of cluster vs Product taken and cluster vs Passport.

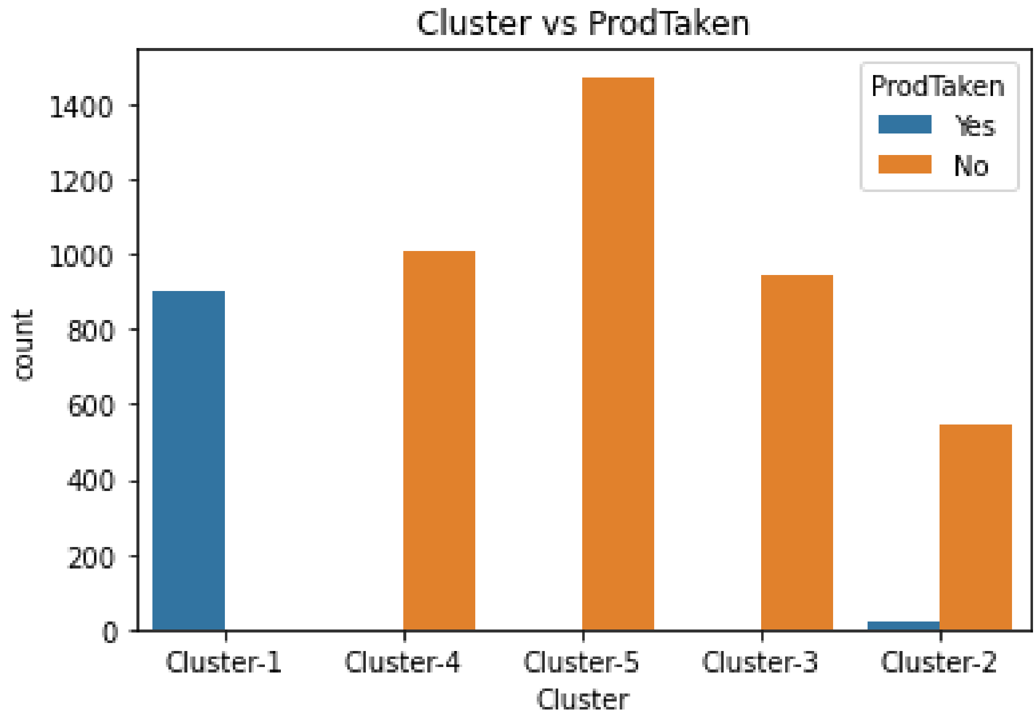

Fig: 22

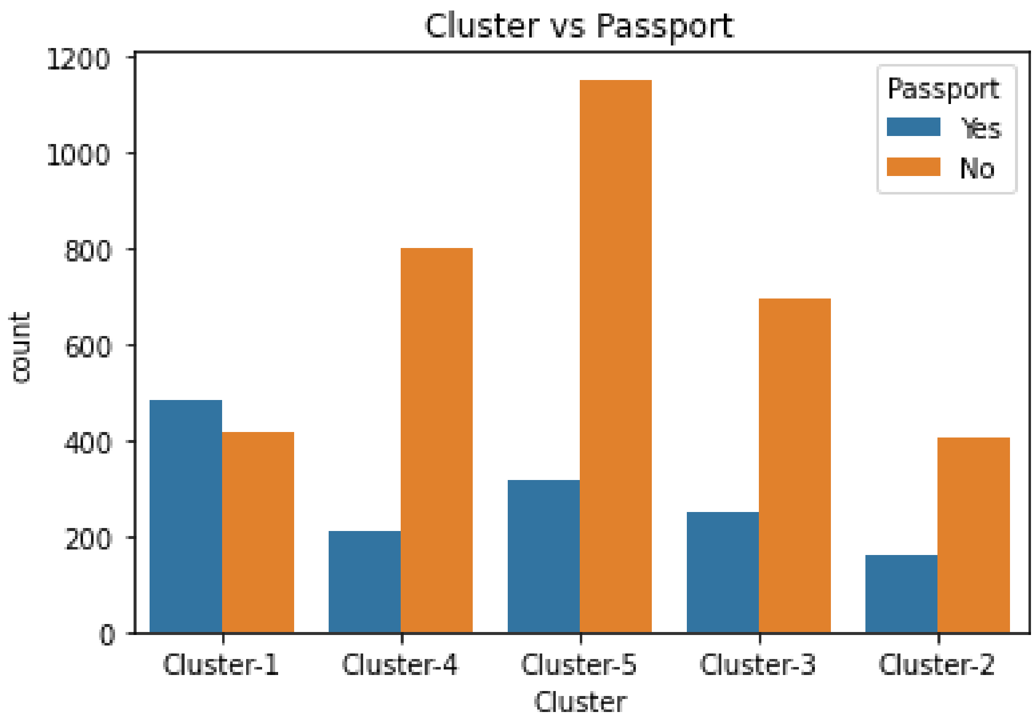

Fig: 23

Observations:

- Cluster-1 is group of those customers who have taken product .
- Maximum numbers of customers who have passport are in Cluster-1.
- Maximum numbers of customers, who are working as executive, belon g to Cluster-1.
- Maximum number of customers belong to Tier-1(metropolitan city) are in Cluster-1.
- The product that are pitched maximum number of times, are multi(chea per product)
- Most of the customers have travelled very less number of trips in Cluste r-1.
- Most of the customers come by themselves in Cluster-1.
- Very few customers are taken product in Cluster-2.
- Very few customers have passport that belong to Cluster-2.

Conclusion:

- Cluster-1 is the group of younger people who have passport. So, t heir propensity of travel will be more.
- Most of the customers in Cluster-1 are working as executive. So t heir monthly income will be low, most probably they will buy ch eaper product like Multi or Super Deluxe.
- Cluser-2 is the group of older people and very few people have p assport. Hence, their propensity of buying product is very low ev en though all they have high monthly income.
- Cluster-4, Cluster-3 and Cluster-5 are the group of younger and middle age group and some of them have passport and most of th em are working as manager. So their monthly income is low. He nce, propensity of buying product is very- very low.

Business Insights from EDA:

1. The data set is imbalanced because the no of 0's (No) is more than 1's(Yes) in the target variables ProdTaken. We should do under sampling and oversampling technique. We can also use SMOTE to remove imbalance data set problem. All these techniques are very important for handling imbalance data set problem. If the model is predicting more 0's then 1's , then model performance will decrease.
2. Cluster-1 is the group of younger people who have passport also. For business perspective Travel Company should target these people for selling the product. Also thought of some strategy so that non passport holder can get passport so that propensity of product could increase.