CAPSTONE PROJECT: SOCIAL MEDIA TOURISM

PGP-DSBA 18 FEB 2022

By: THILAK RAJ

Table of Contents

Tables

Figures

1. Introduction of the business problem

Problem statement:

An aviation company that provides domestic as well as international trips to the customers now wants to apply a targeted approach instead of reaching out to each of the customers. This time they want to do it digitally instead of telecalling. Hence they have collaborated with a social networking platform, so they can learn the digital and social behavior of the customers and provide the digital advertisement on the user page of the targeted customers who have a high propensity to take up the product.
Propensity of buying tickets is different for different login devices. Hence, you have to create 2 models separately for Laptop and Mobile. [Anything which is not a laptop can be considered as mobile phone usage.]
The advertisements on the digital platform are a bit expensive; hence, you need to be very accurate while creating the models.

a. Defining the problem statement.

Create two different models for each type of device customers use to access social networking platforms to learn the customer behavior on travel-related pages, their recent travel check-ins and also their influence on others. Model should be created for two types of device, Laptops and Mobiles (anything which is not a laptop shall be considered a mobile device)

b. Need of the study/project.

Nowadays, people's digital footprint is increasing as they are sucked into the culture of social media. Thus, targeting a customer through digital marketing is much more beneficial to a company than a traditional method like telecalling, which might reach a larger group of people, but we are not sure of a potential customer. To find out the customer's interest in travel, which could help the aviation company pitch their product correctly and on time to the right customer, by accessing the customer's digital and social behaviour via social networking platforms? And provide a digital ad only to customers who have a higher propensity to plan a trip soon.

c. Understanding business/social opportunity.

Business Opportunity:

As we can target only the potential customers who have a good chance of buying the product, the return on investment (ROI) on marketing spending could be higher. Also, there is a reduction in telecalling, which translates into less spending on call centers and more control over marketing spend. This will help businesses concentrate more on interested customers and increase the customer retention rate.

Social Opportunity:

As we are avoiding calling everyone to advertise the product, even if the chances of customers buying the product are very low or null, we can channel the company's potential towards the right group of customers through social media campaigns. As we are targeting the right group of customers, the chance of annoying, uninterested customers will be low.

Variable description:

Variable	Description
UserID	Unique ID of user
Buy_ticket	Buy ticket in next month
Yearly_avg_view_on_travel_page	Average yearly views on any travel related page by user
preferred_device	Through which device user preferred to do login
total_likes_on_outstation_checkin_given	Total number of likes given by a user on out of station checkings in last year
yearly_avg_Outstation_checkins	Average number of out of station check-in done by user
member_in_family	Total number of relationship mentioned by user in the account
preferred_location_type	Preferred type of the location for travelling of user
Yearly_avg_comment_on_travel_page	Average yearly comments on any travel related page by user
total_likes_on_outofstation_checkin_received	Total number of likes received by a user on out of station checkings in last year
week_since_last_outstation_checkin	Number of weeks since last out of station check-in update by user
following_company_page	Weather the customer is following company page (Yes or No)
montly_avg_comment_on_company_page	Average monthly comments on company page by user
working_flag	Weather the customer is working or not
travelling_network_rating	Does user have close friends who also like travelling. 1 is highs and 4 is lowest
Adult_flag	Weather the customer is adult or not
Daily_Avg_mins_spend_on_traveling_page	Average time spend on the company page by user on daily basis

Table 1: Variable description

2. Data Report

a. Understanding how data was collected in terms of time, frequency and methodology.

The digital and social behavior of 11,760 unique customers has been collected by a third party; in this case, it is a social networking site. The data collected regarding their travel interests. The data consists of their

- Likes, comments, and reviews on travel-related pages.
- Outstation check-ins, their frequency, likes, and interactions with others' check-ins.
- Personal information such as their family, work status, whether they are adults, and average time spent on travel-related pages.
- Finally, the target columns state whether each customer has bought a ticket for their next trip from the aviation company.

b. Visual inspection of data (rows, columns, descriptive details).

Below is the dataset sample.

UserID	Taken_product	Yearly_avg_view_on_travel_page	preferred_device	total_likes_on_outstation_checkin_given	yearly_avg_Outstation_checkins	member_in_family	preferred_location_type	Yearly_avg_comment_on_travel_page	total_likes_on_outofstation_checkin_received	week_since_last_outstation_checkin	following_company_page	montly_avg_comment_on_company_page	working_flag	travelling_network_rating	Adult_flag	Daily_Avg_mins_spend_on_traveling_page
1000001	Yes	307	iOS and A	38570	1	2	Financial	94	5993	8	Yes	11	No	1	0	8
1000002	No	367	iOS	9765	1	1	Financial	61	5130	1	No	23	Yes	4	1	10
1000003	Yes	277	iOS and A	48055	1	2	Other	92	2090	6	Yes	15	No	2	0	7
1000004	No	247	iOS	48720	1	4	Financial	56	2909	1	Yes	11	No	3	0	8
1000005	No	202	iOS and A	20685	1	1	Medical	40	3468	9	No	12	No	4	1	6
1000006	No	240	iOS	35175	1	2	Financial	79	3068	0	No	13	No	3	0	8
1000007	No		iOS and A	46340	1	Three	Medical	81	2670	4	Yes	20	Yes	1	3	12
1000008	No	225	iOS and Android		24	1	Financial	67	2693	1	No	22	Yes	2	1	1
1000009	No	285	iOS	7560	23	3	Financial	44	9526	0	No	21	Yes	2	0	10
1000010	No	270	iOS and A	45465	27	3		94	5237	6	No	13	No	2	2	17

Table 2: Sample of given dataset

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11760 entries, 0 to 11759
Data columns (total 17 columns):
 #   Column                                    Non-Null Count  Dtype
---  ------                                    --------------  -----
 0   UserID                                    11760 non-null  int64
 1   Taken_product                             11760 non-null  object
 2   Yearly_avg_view_on_travel_page            11179 non-null  float64
 3   preferred_device                          11707 non-null  object
 4   total_likes_on_outstation_checkin_given   11379 non-null  float64
 5   yearly_avg_Outstation_checkins            11685 non-null  object
 6   member_in_family                          11760 non-null  object
 7   preferred_location_type                   11729 non-null  object
 8   Yearly_avg_comment_on_travel_page         11554 non-null  float64
 9   total_likes_on_outofstation_checkin_received  11760 non-null  int64
 10  week_since_last_outstation_checkin        11760 non-null  int64
 11  following_company_page                    11657 non-null  object
 12  montly_avg_comment_on_company_page        11760 non-null  int64
 13  working_flag                              11760 non-null  object
 14  travelling_network_rating                 11760 non-null  int64
 15  Adult_flag                                11760 non-null  int64
 16  Daily_Avg_mins_spend_on_traveling_page    11760 non-null  int64
dtypes: float64(3), int64(7), object(7)
memory usage: 1.5+ MB
```

Table 3: Dataset info

Dataset observation:

- **Number of Rows (Observations): 11760**
- **The number of Variables (columns): 17**

- **Out of 17 columns, 7 are object data type and the remaining variables are of integer and float.**
- **Here we observe 'yearly_avg_Outstation_checkins' and 'member in family' are in object type this means some character are present in data which is a bad data So, we have to clean this and convert these data into int64/float.**
- **There are some missing values in some features we have to treat them as well.**
- **Also 'travelling_network_rating', 'week_since_last_outstation_checkin' and 'Adult_flag' are of datatype int64. We can validate and change these to object**

Duplicate value observation:

- **No duplicate values found in the dataset**

Missing Values:

Below is the missing values count.

	Variables	Missing_values
1	Yearly_avg_view_on_travel_page	581
3	total_likes_on_outstation_checkin_given	381
7	Yearly_avg_comment_on_travel_page	206
10	following_company_page	103
4	yearly_avg_Outstation_checkins	75
2	preferred_device	53
6	preferred_location_type	31
0	Taken_product	0
5	member_in_family	0
8	total_likes_on_outofstation_checkin_received	0
9	week_since_last_outstation_checkin	0
11	montly_avg_comment_on_company_page	0
12	working_flag	0
13	travelling_network_rating	0
14	Adult_flag	0
15	Daily_Avg_mins_spend_on_traveling_page	0

Table 4: Missing value count

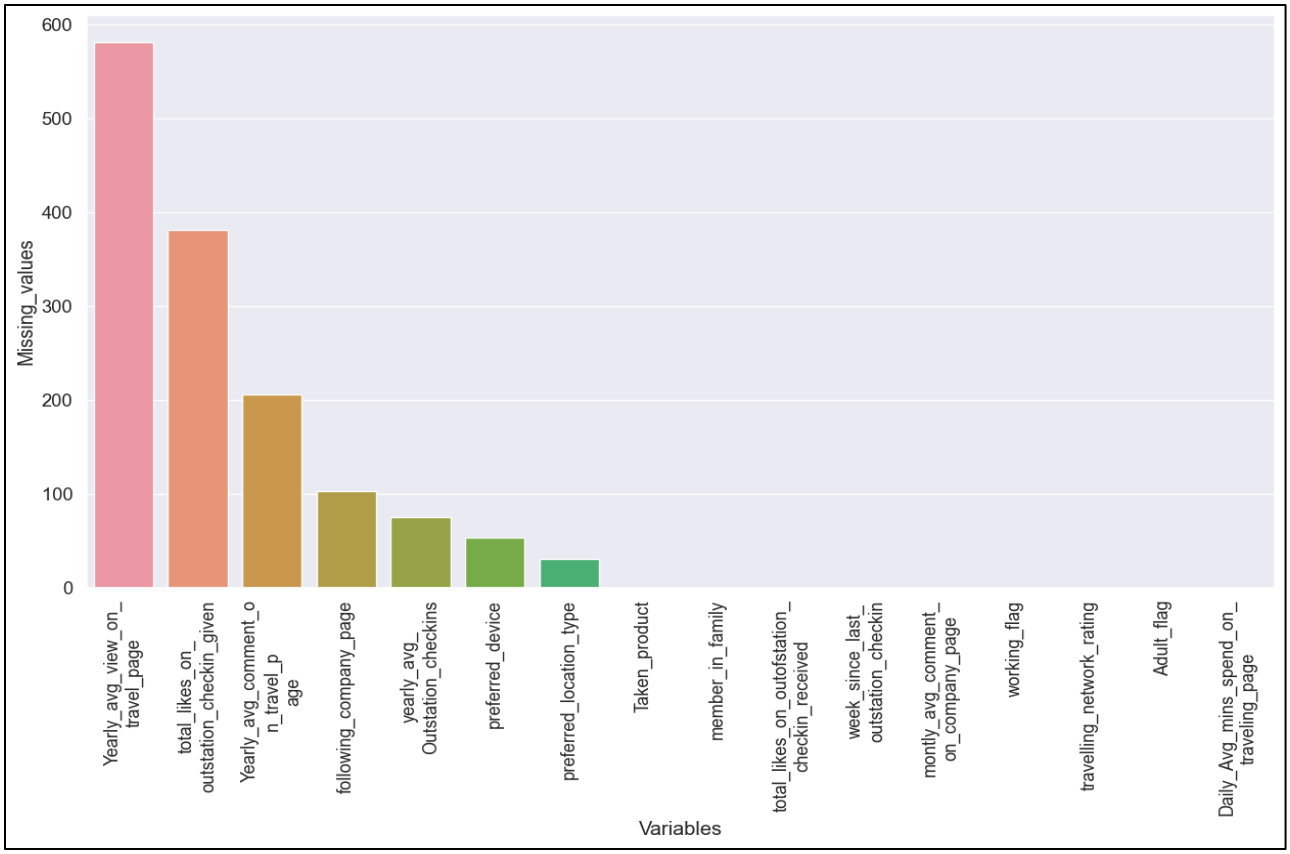


Figure 1: Missing values

Yearly_avg_view_on_travel_page	0.0494
total_likes_on_outstation_checkin_given	0.0324
Yearly_avg_comment_on_travel_page	0.0175
following_company_page	0.0088
yearly_avg_Outstation_checkins	0.0064
preferred_device	0.0045
preferred_location_type	0.0026
Taken_product	0.0000
member_in_family	0.0000
total_likes_on_outofstation_checkin_received	0.0000
week_since_last_outstation_checkin	0.0000
montly_avg_comment_on_company_page	0.0000
working_flag	0.0000
travelling_network_rating	0.0000
Adult_flag	0.0000
Daily_Avg_mins_spend_on_traveling_page	0.0000
dtype: float64	

Table 5: Distribution of missing values.

Missing values observation.

- **7 columns have missing values. However missing values in each column are lies between 0.26% and 4.9% of the total missing values in the dataset i.e., not more than 5% of total missing values in each column observations.**

- **The total number of missing value is 1430 which less than 1% in the dataset (0.76%).**
- **1304 rows have at least 1 missing value, which is ~11% of the total observations.**
- **Except 'following_company_page' variable all other variables are of continuous data type.**

Description of data:

	count	mean	std	min	25%	50%	75%	max
Yearly_avg_view_on_travel_page	11179.0	280.831	68.183	35.0	232.00	271.0	324.00	464.0
total_likes_on_outstation_checkin_given	11379.0	28170.482	14385.032	3570.0	16380.00	28076.0	40525.00	252430.0
Yearly_avg_comment_on_travel_page	11554.0	74.790	24.027	3.0	57.00	75.0	92.00	815.0
total_likes_on_outofstation_checkin_received	11760.0	6531.699	4706.614	1009.0	2940.75	4948.0	8393.25	20065.0
week_since_last_outstation_checkin	11760.0	3.204	2.616	0.0	1.00	3.0	5.00	11.0
montly_avg_comment_on_company_page	11760.0	28.662	48.661	11.0	17.00	22.0	27.00	500.0
travelling_network_rating	11760.0	2.712	1.081	1.0	2.00	3.0	4.00	4.0
Adult_flag	11760.0	0.794	0.852	0.0	0.00	1.0	1.00	3.0
Daily_Avg_mins_spend_on_traveling_page	11760.0	13.817	9.071	0.0	8.00	12.0	18.00	270.0

Table 6: Description of data-Numeric

	count	unique	top	freq
Taken_product	11760	2	No	9864
preferred_device	11707	10	Tab	4172
yearly_avg_Outstation_checkins	11685	30	1	4543
member_in_family	11760	7	3	4561
preferred_location_type	11729	15	Beach	2424
following_company_page	11657	4	No	8355
working_flag	11760	2	No	9952

Table 7: Description of data- Categorical

- **Data type is integer; we are converting this to categorical values as we are not performing any type of numerical operations.**
- **We can keep 'Adult_flag' as binary value (categorical value) i.e., whether the user is an adult: Yes or No. However, it has 2 and 3 as their values.**
- **Values can be imputed as such 0 is not adult and anything other than that shall be adult.**

	count	mean	std	min	25%	50%	75%	max
Yearly_avg_view_on_travel_page	11179.0	280.831	68.183	35.0	232.00	271.0	324.00	464.0
total_likes_on_outstation_checkin_given	11379.0	28170.482	14385.032	3570.0	16380.00	28076.0	40525.00	252430.0
Yearly_avg_comment_on_travel_page	11554.0	74.790	24.027	3.0	57.00	75.0	92.00	815.0
total_likes_on_outofstation_checkin_received	11760.0	6531.699	4706.614	1009.0	2940.75	4948.0	8393.25	20065.0
montly_avg_comment_on_company_page	11760.0	28.662	48.661	11.0	17.00	22.0	27.00	500.0
Daily_Avg_mins_spend_on_traveling_page	11760.0	13.817	9.071	0.0	8.00	12.0	18.00	270.0

Table 8: Description of data after data-type conversion-Numeric

	count	unique	top	freq
Taken_product	11760	2	No	9864
preferred_device	11707	10	Tab	4172
yearly_avg_Outstation_checkins	11685	30	1	4543
member_in_family	11760	7	3	4561
preferred_location_type	11729	15	Beach	2424
week_since_last_outstation_checkin	11760	12	1	3070
following_company_page	11657	4	No	8355
working_flag	11760	2	No	9952
travelling_network_rating	11760	4	3	3672
Adult_flag	11760	4	0	5048

Table 9: Description of data after data-type conversion- Categorical

Description of data- Observation:

Continuous variables:

- Few of the variables have larger difference in mean and median (50%) thus only few variables are containing the outliers.

Categorical Variables:

- There are more number of customer who are not taken the products that customer who took the product as per target variable 'Taken_product'
- Majority of people's preferred device is 'Tab'.
- Majority of the family having 3 members as per given dataset.
- Most preferred location shall be 'Beach'.
- Most of the users are with the outstation checkins within 1 week
- With the given data we can see it consist of more non-working people.

c. Understanding of attributes (variable info, renaming if required).

Value count of all categorical attributes:

```
Value count of "Taken_product"
No     9864
Yes    1896
Name: Taken_product, dtype: int64
```

```
Value count of "preferred_device"
Tab                4172
iOS and Android    4134
Laptop             1108
iOS                1095
Mobile              600
Android             315
Android OS          145
ANDROID             134
Other                 2
Others                2
Name: preferred_device, dtype: int64
```

```
Value count of "yearly_avg_Outstation_checkins"
1     4543
2      844
10     682
9      340
7      336
3      336
8      320
5      261
4      256
16     255
6      236
11     229
24     223
29     215
23     215
18     208
15     206
26     199
20     199
25     198
28     180
19     176
14     167
17     160
12     159
22     152
13     150
21     143
27      96
*        1
Name: yearly_avg_Outstation_checkins, dtype: int64
```

```
Value count of "member_in_family"
3        4561
4        3184
2        2256
1        1349
5         384
Three      15
10         11
Name: member_in_family, dtype: int64
```

```
Value count of "preferred_location_type"
Beach              2424
Financial          2409
Historical site    1856
Medical            1845
Other               643
Big Cities          636
Social media        633
Trekking            528
Entertainment       516
Hill Stations       108
Tour  Travel         60
Tour and Travel      47
Game                 12
OTT                   7
Movie                 5
Name: preferred_location_type, dtype: int64
```

```
Value count of "week_since_last_outstation_checkin"
1     3070
3     1766
2     1700
4     1118
0     1032
5      728
6      654
7      594
9      472
8      428
10     138
11      60
Name: week_since_last_outstation_checkin, dtype: int64
```

```
Value count of "following_company_page"
No     8355
Yes    3285
1        12
0         5
Name: following_company_page, dtype: int64
```

```
Value count of "working_flag"
No     9952
Yes    1808
Name: working_flag, dtype: int64
```

```
Value count of "travelling_network_rating"
3    3672
4    3456
2    2424
1    2208
Name: travelling_network_rating, dtype: int64
```

```
Value count of "Adult_flag"
0    5048
1    4768
2    1264
3     680
Name: Adult_flag, dtype: int64
```

Table 10: Value count for categorical variables

Target variable class proportion:

```
No     0.838776
Yes    0.161224
Name: Taken_product, dtype: float64
```

Table 11: Target variable class proportion

There are few redundancies and improper data present in the few columns

- **preferred_device**: Has three different 'Androids', two different 'others' present. We are going to change all the values other than 'Laptop' to 'Mobile'
- **yearly_avg_Outstation_checkins**: There is a special character (*) along with the numerical values.
- **member_in_family**: We should maintain the uniformity but we can find some string values along with numeric values.
- **following_company_page**: Has both 'yes / no' and 0/1 so we should clean up the data.
- **Adult_Flag**: We are going to keep binary values for this attribute.
- **preferred_location_type**: we found 'Tour Travel' and 'Tour and Travel' as different values. We will maintain the uniformity by replacing 'Tour Travel' with 'Tour and Travel'.

Data wrangling:

- We have converted the **Preferred_device** categories into two values, First one is Laptop and anything other than Laptop are converted as Mobile devices
- We have replaced * character in **yearly_avg_outstation_checkins** with column most frequent (Mode) value.
- We have replaced the string values in **member_in_family** with the numeric values.
- We wanted to keep only binary values for this **Adult_flag** variable, so we have converted values 2 and 3 into 1. No the column containing only 0s and 1s.
- To maintain the uniformity we have replaced 'Tour Travel' with 'Tour and Travel' in the variable **Preffered_location_type** .

3. Exploratory Data Analysis(EDA).

a. Univariate analysis (distribution and spread for every continuous attribute, distribution of data in categories for categorical ones)

Number of Numerical columns: 7
Number of Categorical Columns: 9

Distribution of Numerical and categorical variables.

Numerical variable distribution:

```
Yearly_avg_view_on_travel_page                0.414409
total_likes_on_outstation_checkin_given       0.489638
yearly_avg_Outstation_checkins                0.968297
member_in_family                              0.001205
Yearly_avg_comment_on_travel_page             4.868225
total_likes_on_outofstation_checkin_received  1.368578
week_since_last_outstation_checkin            0.915334
montly_avg_comment_on_company_page            7.684150
travelling_network_rating                    -0.302557
Adult_flag                                   -0.285906
Daily_Avg_mins_spend_on_traveling_page        4.480682
dtype: float64
```

Table 12: Skewness of the numerical values

Histogram and Boxplot for all numerical value columns:

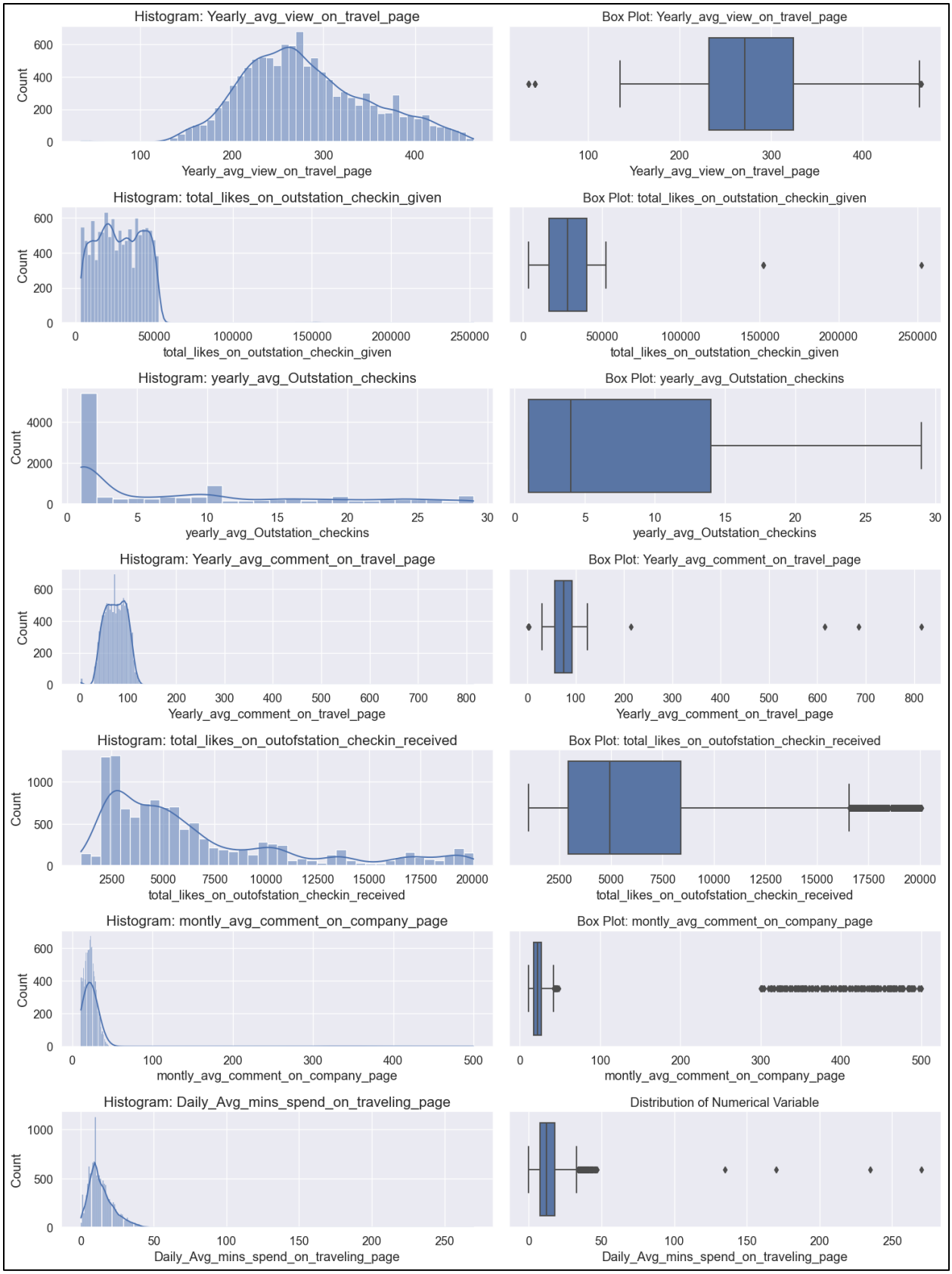


Figure 2: Distribution of Numerical Variable- Histogram and Baxplot.

Observation:

Histogram

- Yearly average views on Travel pages are normally distributed with most number of views are around 250 to 300 views. Remaining all other variables is right skewed with outliers.
- total_likes_on_outstation_checkins_given, yearly_avg_comment_on_travel_page, montly_avg_comment_on_company_page and Daily_avg_mins_spend_on_traveling_page has very few extreme values at the upper band thus all are right skewed.
- Majority user has the yearly average outstation checkins below 5.
- Likewise, monthly average comments on company pages are also less than 50 for most of the users.
- Likes given on outstation check in has even distribution of users across different bucket of likes.

Boxplot:

- yearly_avg_view_on_travel_page has very few outliers and normally distributed.
- total_likes_on_outstation_checkin_given also evenly distributed with only very few outliers at the upper limit which are significantly higher than the other values.
- yearly_avg_comment_on_travel_page is also has only few outliers which can be treated.
- monthly_avg_comment_on_company_page and Daily_Avg_mins_spend_on_travelling_page has whole group of outliers at the upper range**.**

Categorical variable distribution.

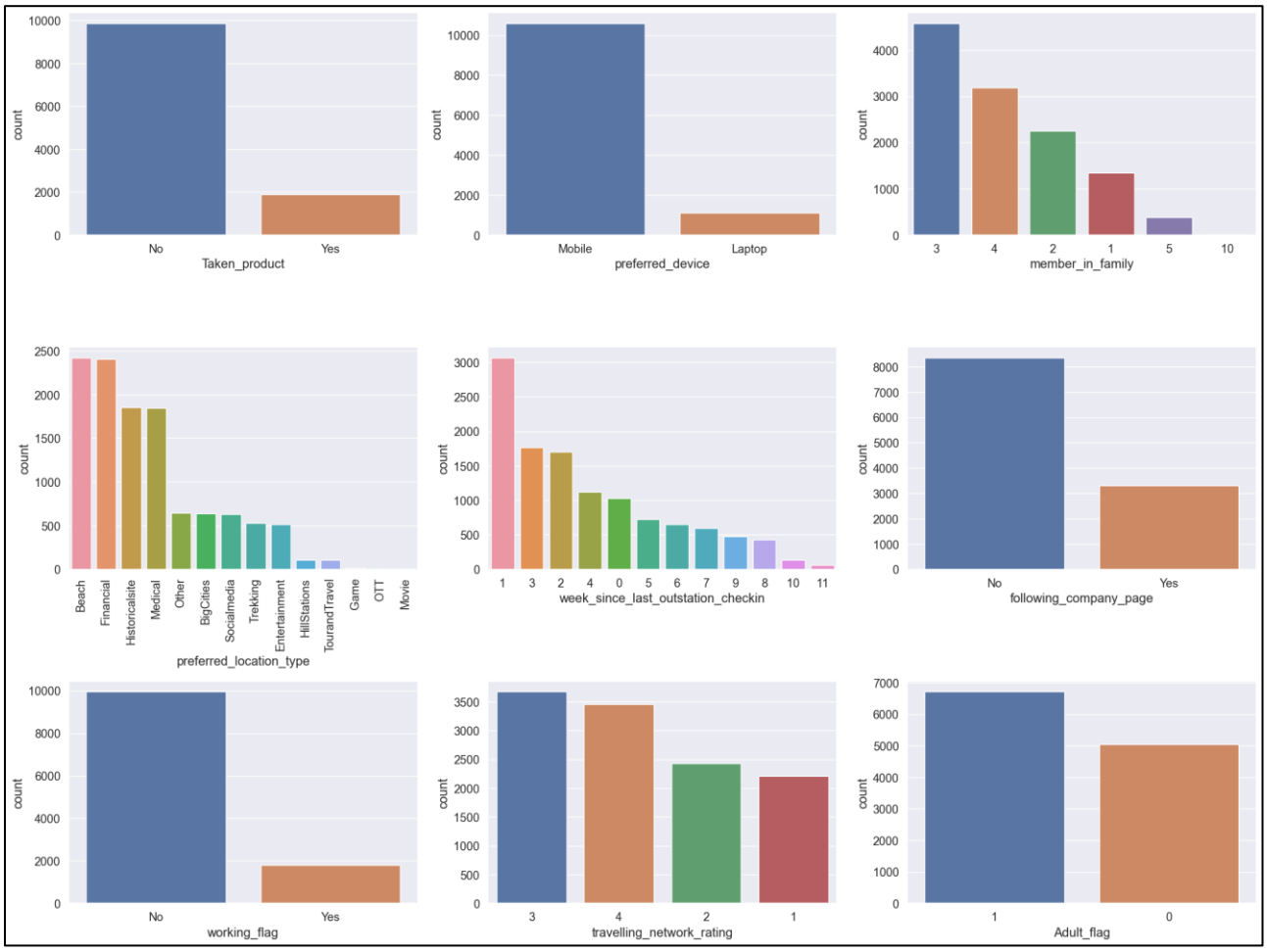


Figure 3: Categorical variable distribution- Countplot

Observation:

- User's most preferred device is mobile devices (Tab, iOS, Android)
- Most of the users have a medium size family with 3 members.
- User's most preferred location type is Beach and Financial, which means that many may went for vacation or business trip, which then followed by Historical site and Medical. The least preferred location type would be Movie, OTT and game.
- Many of the users have very recent outstation check-ins, within in 1 week or less. Indicates that many of the given users had a recent travel.
- Given data has more number of users who are not following the company's social medial page.
- Surprisingly we have larger number of users with no-work flag, which needs to be further analyzed with preferred location type, adult flag and Product taken to understand better who are out targeted customers.
- Majority of the users with travelling network rating as 3 and 4 which indicates that many of the users doesn't have network of travelers.

b. Bivariate analysis (relationship between different variables, correlations)

Relation between target variable and the categorical variables:

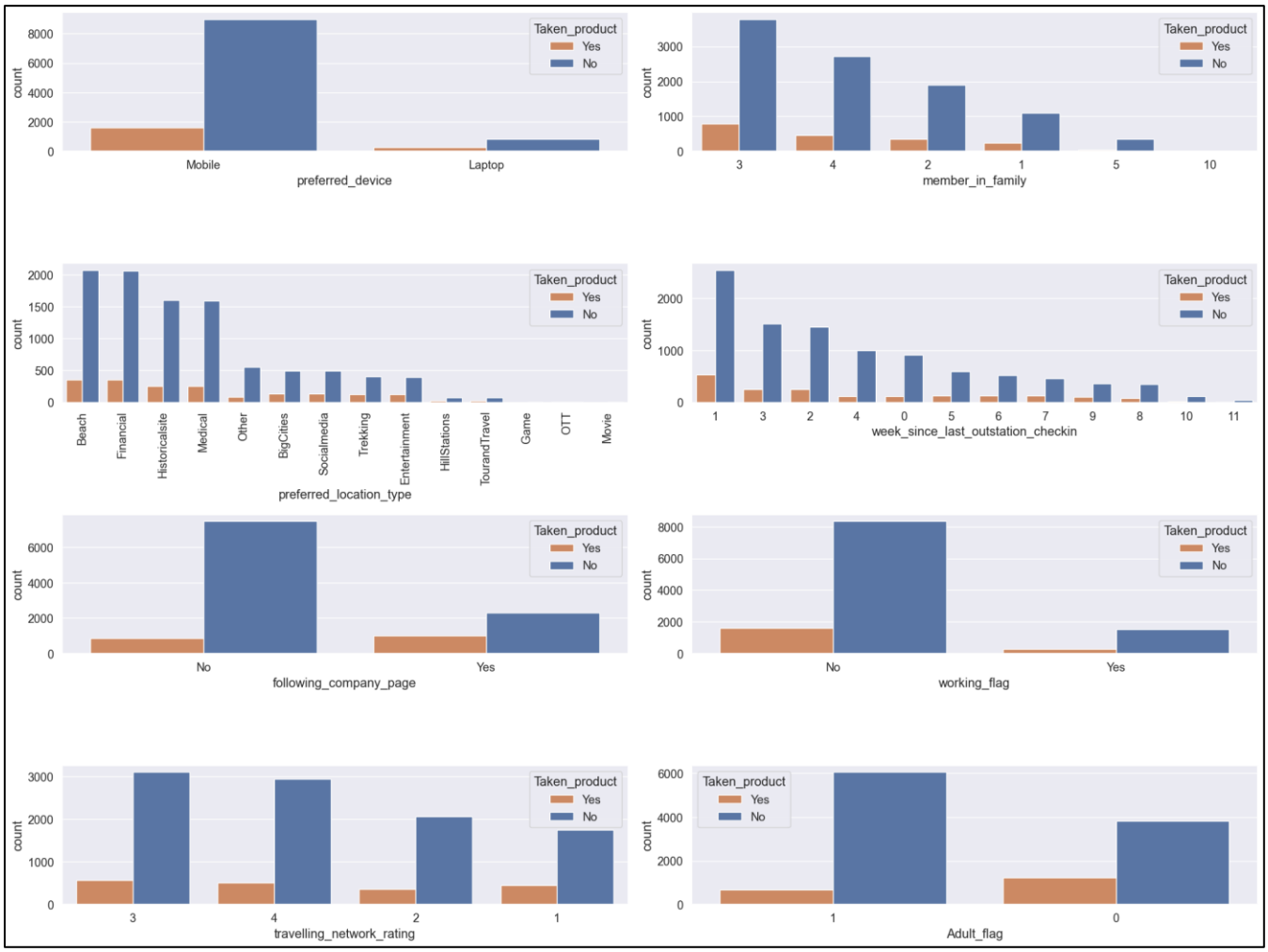


Figure 4: Target variables and Categorical variables

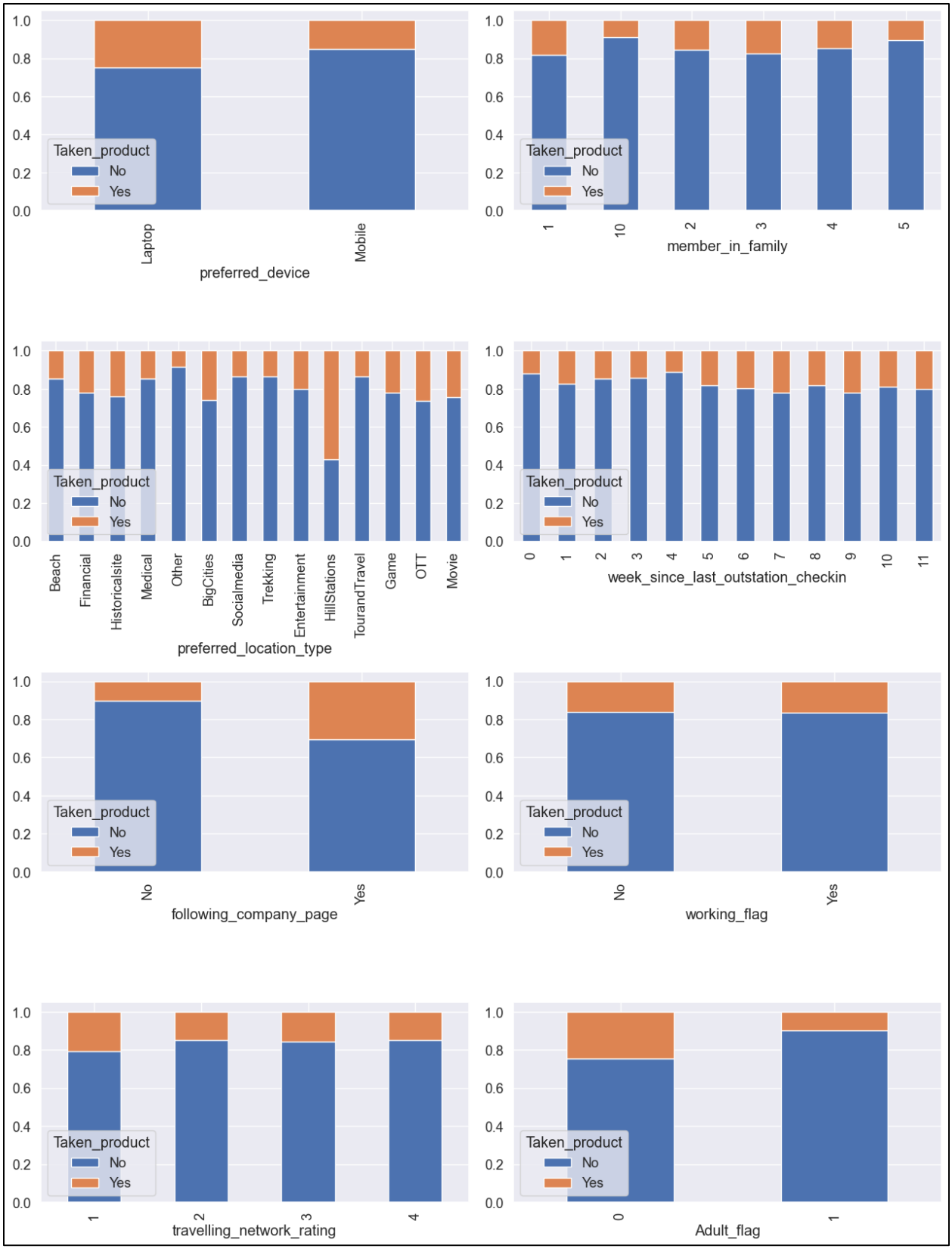


Figure 5: Relationship of target variable with categorical variables (Normalized)

Percentage of taken product in each categorical value level:

Taken_product	No	Yes	All
preferred_device			
Laptop	7.11	2.36	9.46
Mobile	76.78	13.75	90.54
All	83.89	16.11	100.00

Taken_product	No	Yes	All
member_in_family			
1	9.37	2.10	11.47
10	0.09	0.01	0.09
2	16.19	2.99	19.18
3	32.18	6.73	38.91
4	23.13	3.95	27.07
5	2.93	0.34	3.27
All	83.88	16.12	100.00

Taken_product	No	Yes	All
preferred_location_type			
Beach	17.63	3.04	20.67
Big Cities	4.23	1.19	5.42
Entertainment	3.35	1.05	4.40
Financial	17.52	3.02	20.54
Game	0.09	0.01	0.10
Hill Stations	0.68	0.24	0.92
Historical site	13.68	2.15	15.82
Medical	13.58	2.15	15.73
Movie	0.03	0.01	0.04
OTT	0.03	0.03	0.06
Other	4.74	0.74	5.48
Social media	4.21	1.19	5.40
Tour and Travel	0.67	0.24	0.91
Trekking	3.41	1.09	4.50
All	83.86	16.14	100.00

Taken_product	No	Yes	All
week_since_last_outstation_checkin			
0	7.72	1.05	8.78
1	21.56	4.54	26.11
2	12.31	2.14	14.46
3	12.87	2.14	15.02
4	8.45	1.05	9.51
5	5.07	1.12	6.19
6	4.47	1.09	5.56
7	3.95	1.11	5.05
8	2.98	0.66	3.64
9	3.13	0.88	4.01
10	0.95	0.22	1.17
11	0.41	0.10	0.51
All	83.88	16.12	100.00

Taken_product	No	Yes	All
following_company_page			
No	64.23	7.49	71.72
Yes	19.65	8.63	28.28
All	83.88	16.12	100.00

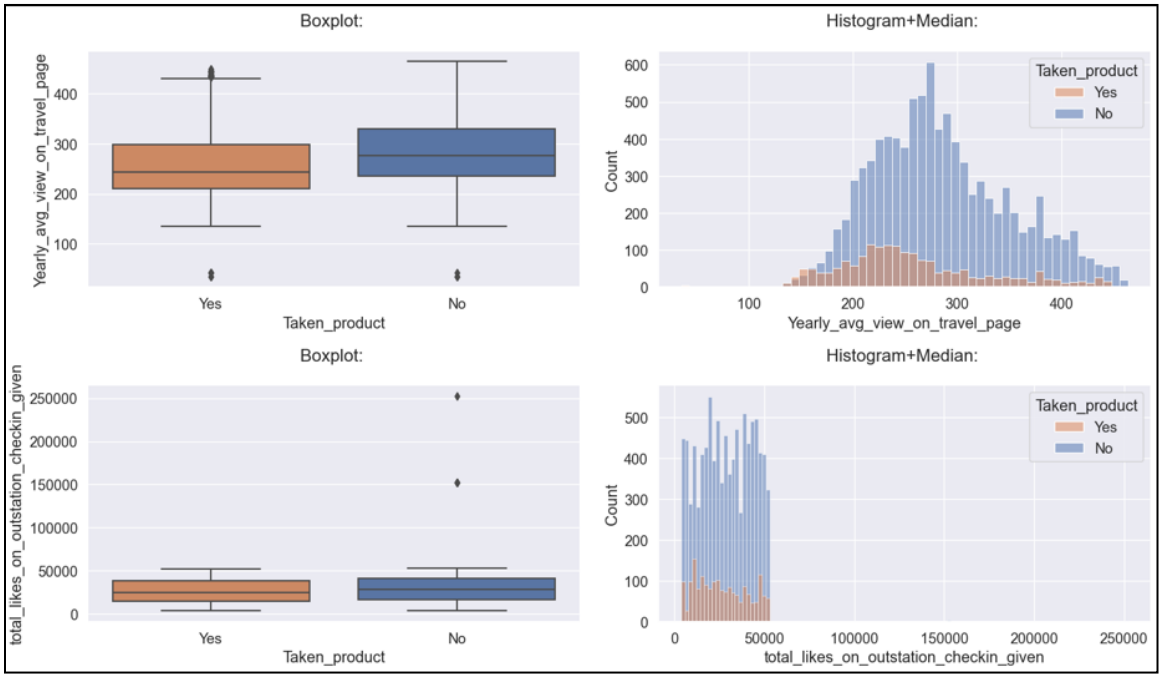
Taken_product	No	Yes	All
working_flag			
No	71.02	13.61	84.63
Yes	12.86	2.52	15.37
All	83.88	16.12	100.00

Taken_product	No	Yes	All
travelling_network_rating			
1	14.90	3.88	18.78
2	17.55	3.06	20.61
3	26.39	4.83	31.22
4	25.03	4.35	29.39
All	83.88	16.12	100.00

Table 13: Percentage of taken product in each categorical value level.

Observation:

- 90% of the customers prefer mobile and 14% of customers have taken the product.
- Number of family member doesn't give any pattern on predicting the target variable but we can see that 39% of the customers are with 3 members in the family and ~7% of the customers chosen the products are belong to this group.
- In preferred_location_type, beach and Financial got highest hits of purchase (apprx 3%).
- As the recency of the last outstation check-ins increases chance of taking the product decreases.
- People who follow company's page don't give any significant difference in to predict the customer's who preferred the product. The percentage of choosing the product in both the segments is around 7.5-8.5%.
- However, people who are not working have better difference in target class prediction. The value is almost 14%.



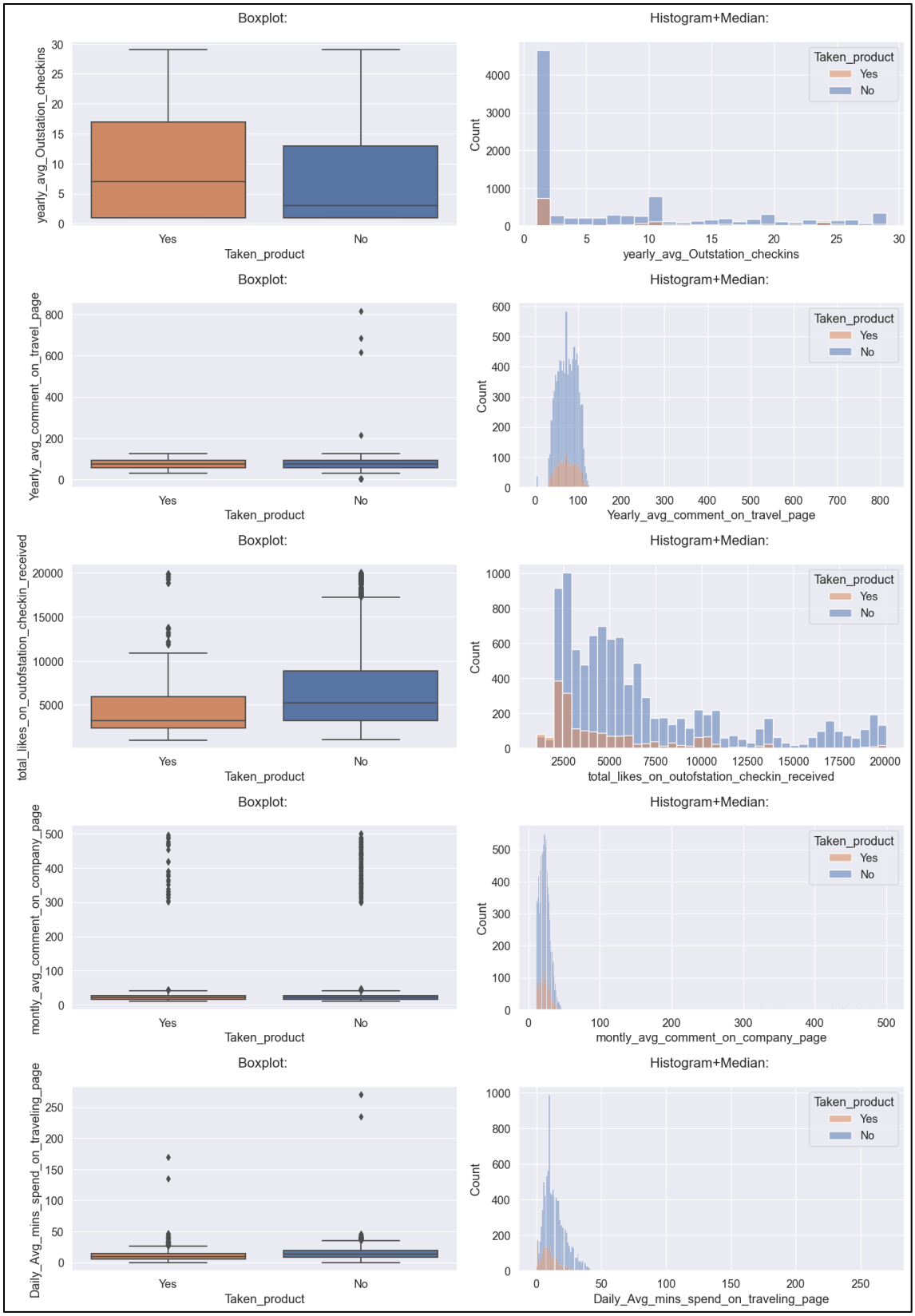


Figure 6: Histogram and Boxplot representation of target and categorical variables.

Observation:

- Yearly_avg_view_on_travel_page, yearly_avg_outstation_checkins, total_likes_on_outstation_checkin_received makes a significant difference between the two classes of target variable.
- However, same is not the case with other continuous variables. They are very good at separating the classes.

Relation between numerical variables.

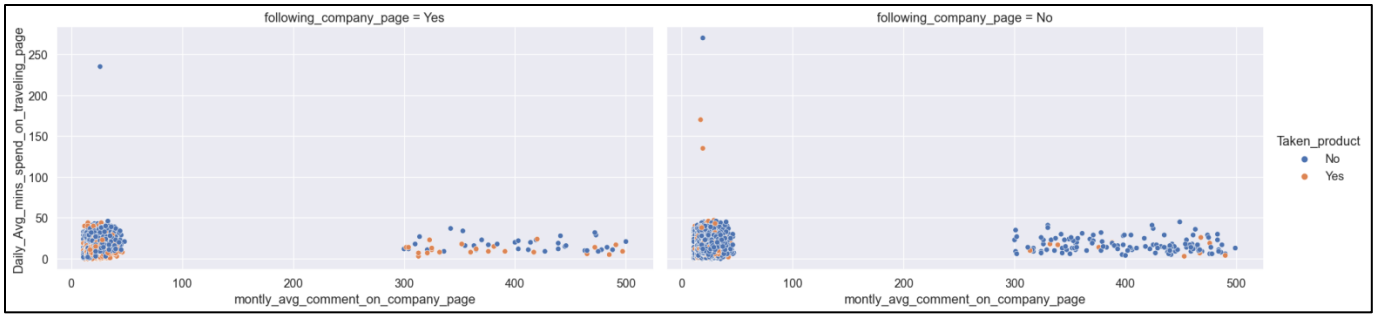


Figure 7: montly_avg_comment_on_company_page vs Daily_Avg_mins_spend_on_traveling_page

Figure 8: Heatmap of numerical variables.

Insights on correlation between independent variables:

- We cannot see any strong correlation between the variables.
- Daily_avg_mins_spend_on_traveling_page and total_likes_on_outofstation_checkin_received has moderate positive correlation suggest that person who spends more time on traveling page has higher likes for their check-ins.
- Monthly_avg_comment_on_company_page and yearly_average_view_on_travel_page is negatively correlated
- Suprisingly, yearly_avg_outstation_checkins and total_likes_on_outofstation_checkin_received is negatively correlated. Thus, the number of check-ins are high tendency to receive likes is low.
- However, the negative correlation is very weak correlation and doesn't show any strong relationship.
- The user who interact more (comments on company page) and also follows company page has higher density on taking the product.

a. Removal of unwanted variables (if applicable).

We have dropped the **UserID** columns as it doesn't provide any information about the class of target variable. We are not dropping any other variables.

b. Missing Value treatment (if applicable)

Below are the missing value counts in data.

	Variables	Missing_values
1	Yearly_avg_view_on_travel_page	581
3	total_likes_on_outstation_checkin_given	381
7	Yearly_avg_comment_on_travel_page	206
10	following_company_page	103
4	yearly_avg_Outstation_checkins	75
2	preferred_device	53
6	preferred_location_type	31

Table 14: Missing values in the data

We imputed the missing values in the dataset. Categorical values are imputed with the mode values and median for continuous variables.

d. Outlier treatment (if required)

As many of the continuous variables has outliers and extreme values which shall be removed as many of the Machine learning algorithm such as Logistic Regression are sensitive to outliers.

Any values above 1.5 x IQR from Q3 shall be floored to that limit, likewise any values below 1.5 x IQR from Q1 shall be capped to that lower limit. IQR shall be calculated as difference between Q3 and Q1.

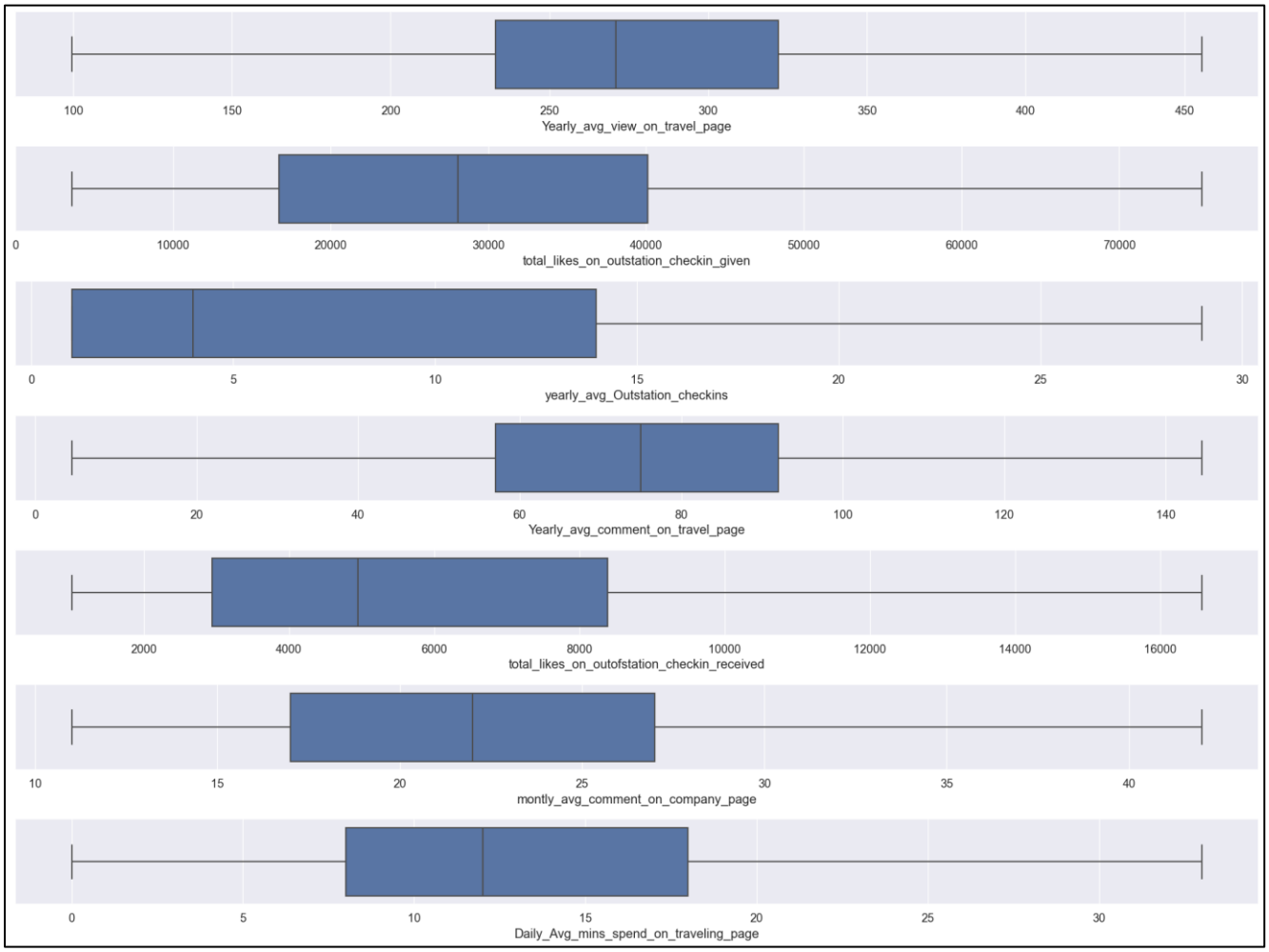


Figure 9: Boxplot after outlier treatment.

e. Variable transformation (if applicable)

It's necessary to convert the entire variable with numerical values as Machine learning algorithms will work only with the numerical values. Thus it is important to map the entire categorical variable into their appropriate numerical values.

preferred_location_type:

- It has different categories of customer preferred locations which is not of any order and doesn't have any priority. Thus we shall encode it will dummy variables and drop the original column to avoid any multicolinearity.

Taken_product , following_company_page , working_flag:

- These variables only has binary values which shall be converted to 0s and 1s mapped to 'No' and 'Yes' respectively.
- And all other object data type variables are converted to integer except 'preferred_device' which shall be used to differentiate the data set into based on the user preferred device to create two different models.

Below is the data info after all updates.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11760 entries, 0 to 11759
Data columns (total 28 columns):
 #   Column                                    Non-Null Count  Dtype
---  ------                                    --------------  -----
 0   Taken_product                             11760 non-null  int32
 1   Yearly_avg_view_on_travel_page            11760 non-null  float64
 2   preferred_device                          11760 non-null  object
 3   total_likes_on_outstation_checkin_given   11760 non-null  float64
 4   yearly_avg_Outstation_checkins            11760 non-null  float64
 5   member_in_family                          11760 non-null  int64
 6   Yearly_avg_comment_on_travel_page         11760 non-null  float64
 7   total_likes_on_outofstation_checkin_received  11760 non-null  float64
 8   week_since_last_outstation_checkin        11760 non-null  int64
 9   following_company_page                    11760 non-null  int32
 10  montly_avg_comment_on_company_page        11760 non-null  float64
 11  working_flag                              11760 non-null  int32
 12  travelling_network_rating                 11760 non-null  int64
 13  Adult_flag                                11760 non-null  int64
 14  Daily_Avg_mins_spend_on_traveling_page    11760 non-null  float64
 15  preferred_location_type_Big Cities        11760 non-null  uint8
 16  preferred_location_type_Entertainment     11760 non-null  uint8
 17  preferred_location_type_Financial         11760 non-null  uint8
 18  preferred_location_type_Game              11760 non-null  uint8
 19  preferred_location_type_Hill Stations     11760 non-null  uint8
 20  preferred_location_type_Historical site   11760 non-null  uint8
 21  preferred_location_type_Medical           11760 non-null  uint8
 22  preferred_location_type_Movie             11760 non-null  uint8
 23  preferred_location_type_OTT               11760 non-null  uint8
 24  preferred_location_type_Other             11760 non-null  uint8
 25  preferred_location_type_Social media      11760 non-null  uint8
 26  preferred_location_type_Tour and Travel   11760 non-null  uint8
 27  preferred_location_type_Trekking          11760 non-null  uint8
dtypes: float64(7), int32(3), int64(4), object(1), uint8(13)
memory usage: 1.4+ MB
```

Table 15: Data info after updates.

f. Addition of new variables (if required)

We are not adding any new variables as of now. But we noted that we can use variables like **yearly_avg_Outstation_checkins** and **total_likes_on_outofstation_checkin_received** to create new variables to improve the model building process. We will create it if it is necessary.

4. Business insights from EDA

a. Is the data unbalanced? If so, what can be done? Please explain in the context of the business.

Before checking the balance of the data, let bifurcate the dataset into two based on the User preferred device - 'Laptop' and 'Mobile'.

Checking for class imbalance:

Laptop dataset:

```
0    0.750903
1    0.249097
Name: Taken_product, dtype: float64
```

Mobile dataset:

```
0    0.847916
1    0.152084
Name: Taken_product, dtype: float64
```

As the Target class:1 of the Dependent variable is more than 10% of the given data in both of the data sets there is no need for any data balancing.

c. Business insights.

yearly_avg_Outstation_checkins and **total_likes_on_outofstation_checkin_received**

- The attribute **total_likes_on_outofstation_checkin_received** directly in relation with the popularity f the user. Popularity of the user give the significance on the Target variable however, it is opposite that user who is highly popular tend to take the company's product. Thus company has higher scope of approaching these customers as they have potential of influence people in the social media since their average likes per check-in is high.

- Basically the **yearly_avg_Outstation_checkins** value will be high for travelers. We have noticed that the travel frequency of the user also has significance in predication the class '0' or '1' of target variable. Also, user who is a frequent traveler tends to take the company products. Even then we still have many customers to convert into the class '1' by making them to prefer the company's product.

Model selection:

Model building and interpretation.

Addition of new variable 'Traveller':

New variable 'Traveller' based on yearly average number of outstation check-ins can be created in order to find the most frequent travellers buy dividing the values into 3 bukets

- <=2 average check-ins per year as 1 (Not a traveller),
- 3-10 average check-ins per year as 2 (Moderate traveller),
- 11-29 average check-ins per year as 3 (Frequent traveller)

Based on this user's travel frequency we can check whether their frequent travel and popularity status influence in the converting them as a customer to taken the company's product.

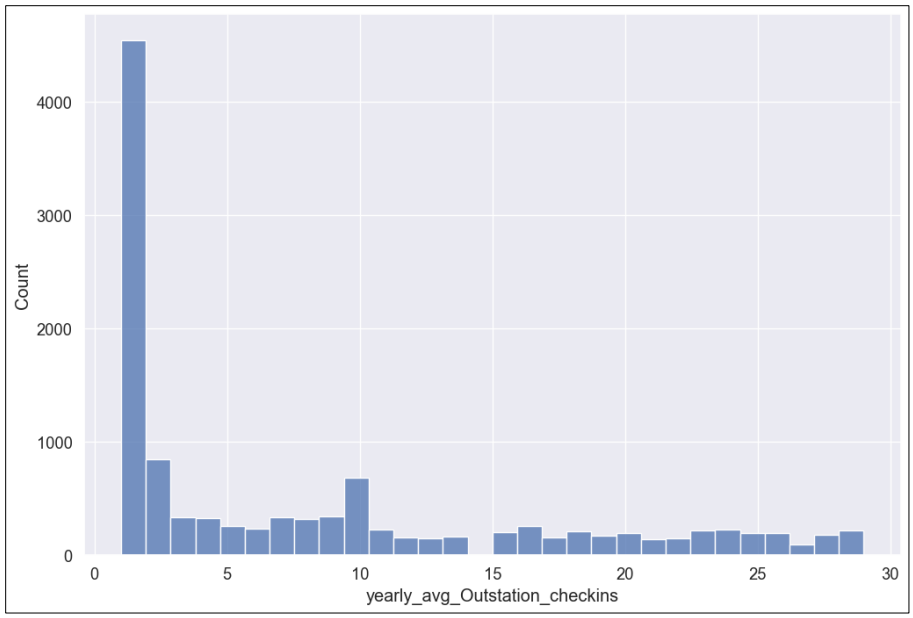


Figure 10: Yearly average outstation check-ins.

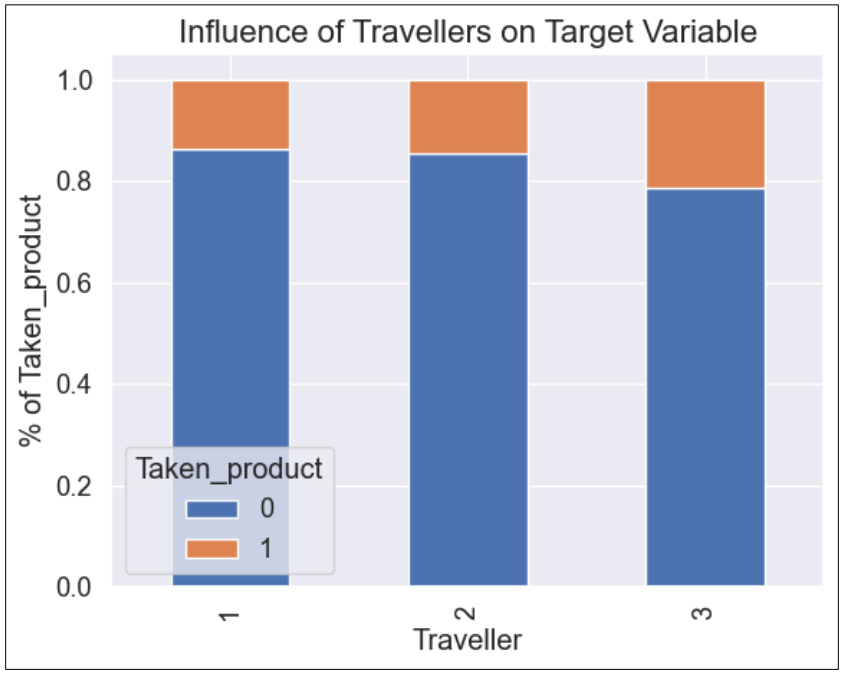


Figure 11: Influence of travelers on target variables.

Traveler value count in laptop and mobile datasets:

Laptop dataset:

```
2    448
1    344
3    316
Name: Traveller, dtype: int64
```

Mobile dataset:

```
1    5044
3    2985
2    2623
Name: Traveller, dtype: int64
```

Table 16: Traveller value count in laptop and mobile datasets.

Observation:

- We noticed good number of frequent travellers and moderate travellers in both the datasets.
- Thus by grouping the users based on the Travel frequency based on the check-ins show significance in predicting the target of 'Taken_product'.
- As the user with higher travel frequency has more proportion of opting for the company's product

- 'Traveller' frequency of the user also has significance in predication the class '0' or '1' of target variable. Also, user who is a frequent traveller tends to take the company products. Even then we still have many customers to convert into the class '1' by making them to prefer the company's product.

Splitting the data set into 'Predictor' and 'Target' variables:

Both datasets for 'Laptop' and 'Mobile' devices are split into Predictor and Target variables (X and Y) respectively before further split into train and test data.

Laptop Data:
Size of Predictor variables (X_laptop): (1108, 26)
Size of target variable (Y_laptop): (1108)

Mobile data:
Size of Predictor variables (X_mobile): (10652, 26)
Size of target variable (Y_mobile): (10652)

- Scale the 'predictor' variables as few models are sensitive to scales of different variables
- Scale the numerical data into common scale using StandardScaler from Sklearn.

Divide the data into Test and Train dataset:

Let divide each dataset into Train and Test data in order to train the model and validate it for its performance on the unknown data. We shall keep 30% of the data as test data and the remaining 70% as train data.

Laptop data:
Size of X_train for laptop: (775, 26)
Size of X_test for laptop: (333, 26)
Size of y_train for laptop: (775)
Size of y_test for laptop: (333)

Mobile data:
Size of X_train for mobile: (7456, 26)
Size of X_test for mobile: (3196, 26)
Size of y_train for mobile: (7456)
Size of y_test for mobile: (3196)

Choice of Models:

Our choice of models for the particular business problem, which is a classification problem, with the Target variable as binary class ('Yes' – 1 & 'No' – 0):

- Tree based models (Decision Tree, Random Forest)
- Ensemble Model (XGBoost Classifier)
- Logistic Regression
- Artificial Neural Network.

Choice of model Evaluation Metrics:

- For the first iteration, let us build the models with default parameters. Depending upon the performance of the model, whether it overfit or underfit, we shall do the hyper parameter tuning to achieve a better results on both train and test data set for the necessary parameters.
- Here, we shall consider **Recall** of the positive class as one of the important metric in evaluating the model performance. Higher Recall shows that the model is good at predicting the entire customer who is likely to buy the product by reducing the False Positives.

Models for Laptop:

Detecting Multicollinearity with VIF (Variance Inflation Factor):

Multicollinearity occurs when there are two or more independent variables in a multiple regression model, which have a high correlation among themselves. When some features are highly correlated, we might have difficulty in distinguishing between their individual effects on the dependent variable. Multicollinearity can be detected using various techniques, one such technique being the Variance Inflation Factor (VIF).

Let us check for the Variance Inflation factor to ensure that there is not multicollinearity between the predictor variables before training the model. Turns out that the Predictor variables doesn't have any multicollinearity.

	variables	VIF
6	member_in_family	7.907532
25	Traveller	6.324587
10	travelling_network_rating	6.307655
5	Daily_Avg_mins_spend_on_traveling_page	3.025064
7	week_since_last_outstation_checkin	2.887609
11	Adult_flag	2.255310
3	total_likes_on_outofstation_checkin_received	2.090047
0	Yearly_avg_view_on_travel_page	1.786551
17	preferred_location_type_Historical site	1.691398
9	working_flag	1.569327
8	following_company_page	1.508582
4	montly_avg_comment_on_company_page	1.447612
12	preferred_location_type_Big Cities	1.287143
24	preferred_location_type_Trekking	1.205035
21	preferred_location_type_Other	1.190352
16	preferred_location_type_Hill Stations	1.088956
2	Yearly_avg_comment_on_travel_page	1.051710
1	total_likes_on_outstation_checkin_given	1.044346
13	preferred_location_type_Entertainment	NaN
14	preferred_location_type_Financial	NaN
15	preferred_location_type_Game	NaN
18	preferred_location_type_Medical	NaN
19	preferred_location_type_Movie	NaN
20	preferred_location_type_OTT	NaN
22	preferred_location_type_Social media	NaN
23	preferred_location_type_Tour and Travel	NaN

Table 17: VIF laptop data.

Performance metrics for Different models:

Confusion matrix:

Laptop training data:

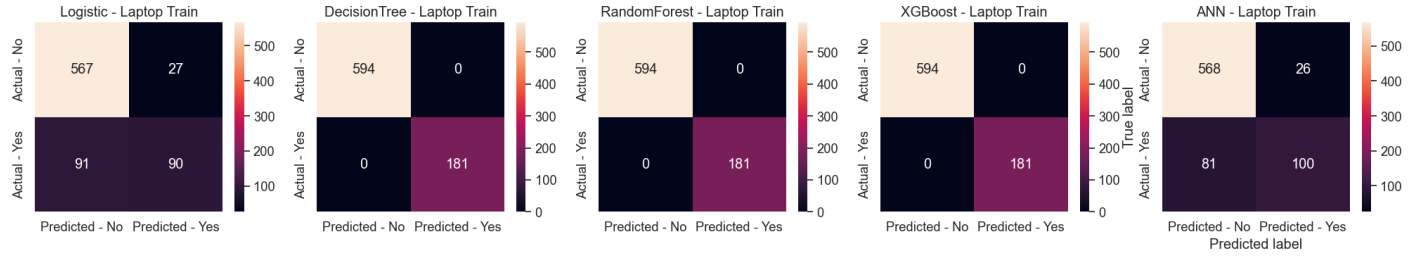


Figure 12: Confusion Matrix - Laptop Train (Default Model)

Laptop testing data:

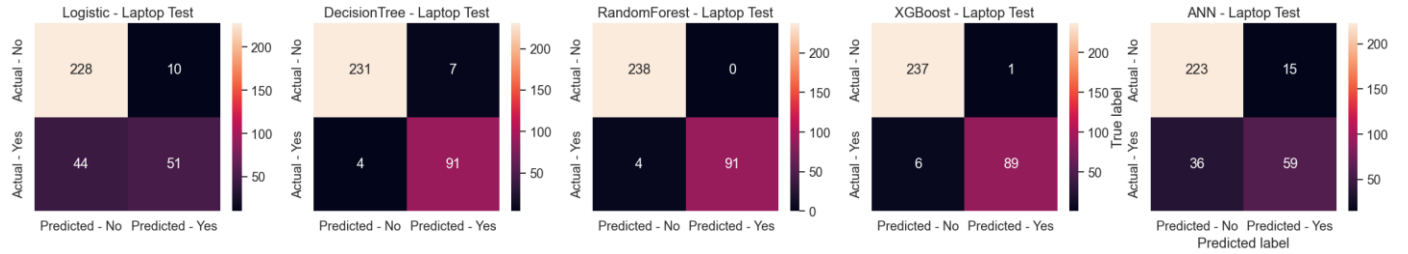


Figure 13: Confusion Matrix - Laptop Test (Default Model)

Classification report:

Model metrics of laptop data (Default model):

Model	Accuracy_train	Accuracy_test	Precision_train	Precision_test	Recall_train	Recall_test	F1_Score_train	F1_score_test
Logistic	0.847742	0.837838	0.769231	0.836066	0.497238	0.536842	0.604027	0.653846
DecisionTree	1.000000	0.966967	1.000000	0.928571	1.000000	0.957895	1.000000	0.943005
RandomForest	1.000000	0.987988	1.000000	1.000000	1.000000	0.957895	1.000000	0.978495
XGBoost	1.000000	0.978979	1.000000	0.988889	1.000000	0.936842	1.000000	0.962162
ANN	0.861935	0.846847	0.793651	0.797297	0.552486	0.621053	0.651466	0.698225

Table 18: Model Metrics (Default Model) - Laptop Train and Test data

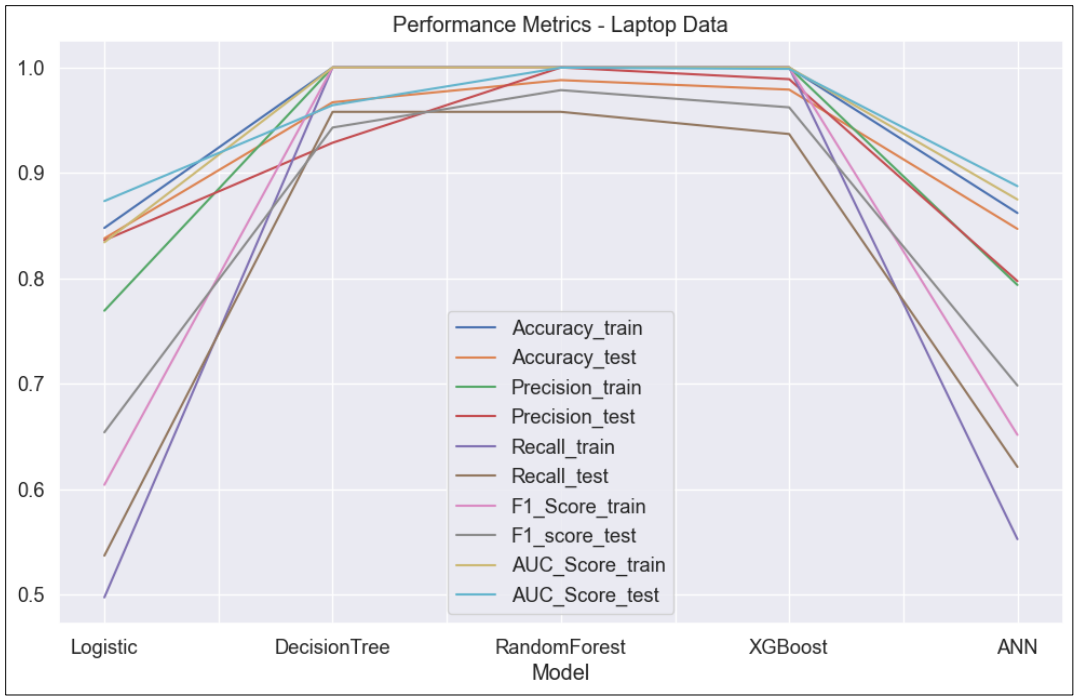


Figure 14: Performance metric (Defaul model)-Laptop

Insights for the default Model:

• Random Forest and XGBoost has the least number of False Negatives (Type II error) in the Positive Class (1)

• Logistic regression is the least performing model followed by ANN.

• However, out of 5 models 3 are overfitting which achieve perfect accuracy of 100% in Train data thus the model is not generalized enough. Hence, we shall tune the hyperparameters for each model to achieve an optimal performance on both Train and Test data.

Hyperparameter Tuning – Laptop

Tune the hyper-parameter which can prune the trees and avoid over-fitting and reduce the complexity of the cost function in non-tree based models.

XGBoost Classifier:

- **Number of Estimators (Trees)** – Number of Trees used for Ensemble, higher the number of trees gives more robust model and a consistent result.
- **Max Depth** – How far the tree can grow, by default is it set to grown until no further possible split which tends to overfit the model. Too high number usually tends to overfit.
- **Min Child Weight** – Minimum weight required in the node for further split. If weights are lesser than the value given tree stop growing further.

- **Learning Rate** – Amount of steps taken by the model function to achieve the minimal optimal point. Lower the number more accurate the results are but take up more resource and time.
- **Subsample** – How much of a sample to be considered for each iteration of the model.

Logistic Regression:

- **Regularization(C)** – Strength of Regularization higher the value lesser the effect of Penalty.
- **Penalty** – Type of Regularization used to calculate the best fit model my attending the global minima.
- **Solver function** – Type of Loss function used to calculate the best fit model my attending the global minima.

Decision Tree:

- **Max Depth** – How far the tree can grow, by default is it set to grown until no further possible split which tends to overfit the model. Too high number usually tends to overfit.
- **Max Feature** – No of features need to be considered for each split. It's good practice to start with half of the features available.
- **Min sample leaf** – Minimum number of samples required to considered the node as leaf node. Higher the value lesser the model over fits by quickly attaining the leaf node.
- **Min sample Split** – Minimum number of Samples required to further split the child node. Higher the value lesser the model over fits by pruning the tree.
- **Criterion**: It determines the measure used for splitting nodes, such as "gini" for Gini impurity or "entropy" for information gain.

Random Forest:

- **N-estimators** – Number of Trees used for Ensemble, higher the number of trees gives more robust model and a consistent result.
- **Max depth** – How far the tree can grow, by default is it set to grown until no further possible split which tends to overfit the model. Too high number usually tends to overfit.
- **Max features** – No of features need to be considered for each split. It's good practice to start with half of the features available.
- **Min sample leaf** – Minimum number of samples required to consider the node as leaf node. Higher the value lesser the model over fits by quickly attaining the leaf node.
- **Min sample Split** – Minimum number of Samples required to further split the child node. Higher the value lesser the model over fits by pruning the tree.
- **Criterion**: It determines the measure used for splitting nodes, such as "gini" for Gini impurity or "entropy" for information gain.

Artificial Neural Network:

- **Hidden layer sizes** – Describes number of neurons and hidden layer to be used for the neural network. Higher the value more complex the network is and consumes more resource & time.
- **Tolerance** – Error value between each step of learning rate, smaller the value more complex the model is and take more time to converge.
- **Max iter** – Maximum number of iteration unless the convergence is achieve.

Performance Metric after Hyperparameter Tuning – Laptop.

After multiple iteration with different parameter values for the hyperparameter, below are the performance metrics of the tuned models.

Model	Tuned_Logistic	Tuned_DecisionTree	Tune_RandomForest	Tuned_XGBoost	Tuned_ANN
Accuracy_train	0.850323	0.874839	0.976774	0.979355	0.845161
Accuracy_test	0.828829	0.855856	0.93994	0.924925	0.840841
Precision_train	0.821782	0.776316	1.0	0.994012	0.842697
Precision_test	0.851852	0.797468	0.962963	0.926829	0.903846
Recall_train	0.458564	0.651934	0.900552	0.917127	0.414365
Recall_test	0.484211	0.663158	0.821053	0.8	0.494737
F1_Score_train	0.588652	0.708709	0.947674	0.954023	0.555556
F1_score_test	0.61745	0.724138	0.886364	0.858757	0.639456
auc_train	0.830031	0.900232	0.999572	0.998493	0.845927
auc_test	0.876426	0.917382	0.990668	0.978328	0.873198

Table 19: Performance Metrics After Tuning – Laptop

After fine tuning them model to avoid over-fitting we can see that the XGBoost performance better that other model in terms of evaluation metrics. Also, it has a good consist performance between train and test data. Thus we will choose this our model of choice.

Models for Mobile

Let us check for the Variance Inflation factor to ensure that there is not multicollinearity between the predictor variables before training the model. Turns out that the Predictor variables don't have any multicollinearity.

	variables	VIF
6	member_in_family	6.667257
10	travelling_network_rating	5.776853
25	Traveller	4.789532
5	Daily_Avg_mins_spend_on_traveling_page	2.964863
7	week_since_last_outstation_checkin	2.727697
14	preferred_location_type_Financial	2.304867
11	Adult_flag	2.266178
3	total_likes_on_outofstation_checkin_received	2.247741
18	preferred_location_type_Medical	2.042216
4	montly_avg_comment_on_company_page	1.864958
0	Yearly_avg_view_on_travel_page	1.756391
17	preferred_location_type_Historical site	1.654804
9	working_flag	1.565909
22	preferred_location_type_Social media	1.385883
8	following_company_page	1.383910
21	preferred_location_type_Other	1.278301
13	preferred_location_type_Entertainment	1.277200
12	preferred_location_type_Big Cities	1.253730
24	preferred_location_type_Trekking	1.178806
23	preferred_location_type_Tour and Travel	1.076392
2	Yearly_avg_comment_on_travel_page	1.073568
16	preferred_location_type_Hill Stations	1.046718
1	total_likes_on_outstation_checkin_given	1.012446
15	preferred_location_type_Game	1.008974
20	preferred_location_type_OTT	1.007420
19	preferred_location_type_Movie	1.004588

Table 20: VIF Mobile data.

Performance metrics for Different models:

Confusion matrix:

Mobile training data:

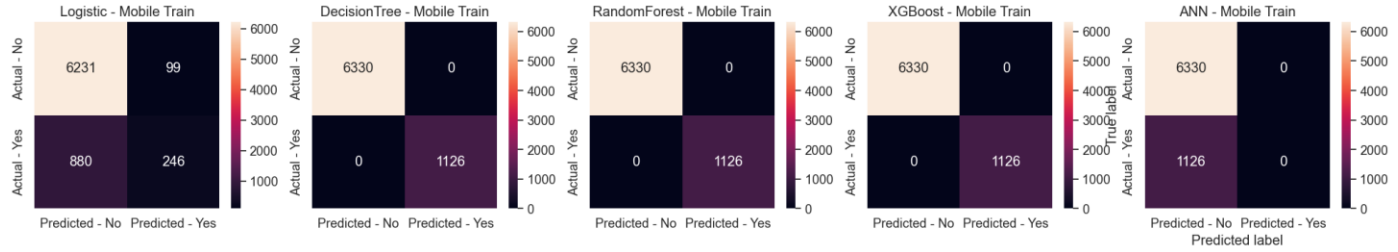


Figure 15: Confusion Matrix - Mobile Train (Default Model)

Mobile testing data:

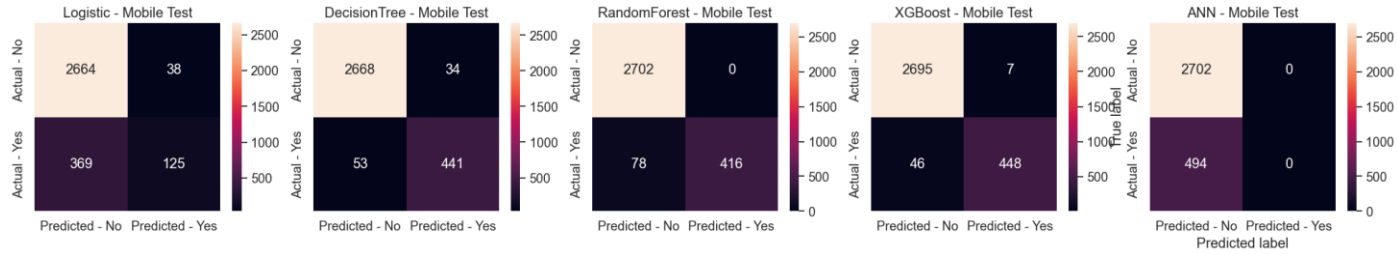


Figure 16: Confusion Matrix - Mobile Test (Default Model)

Classification report:

Model metrics of mobile data (Default model):

Model	Accuracy_train	Accuracy_test	Precision_train	Precision_test	Recall_train	Recall_test	F1_Score_train	F1_score_test
Logistic	0.868696	0.872653	0.713043	0.766871	0.218472	0.253036	0.334466	0.380518
DecisionTree	1.000000	0.972778	1.000000	0.928421	1.000000	0.892713	1.000000	0.910217
RandomForest	1.000000	0.975594	1.000000	1.000000	1.000000	0.842105	1.000000	0.914286
XGBoost	1.000000	0.983417	1.000000	0.984615	1.000000	0.906883	1.000000	0.944152
ANN	0.848981	0.845432	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000

Table 21: Model Metrics (Default Model) - Mobile Train and Test data

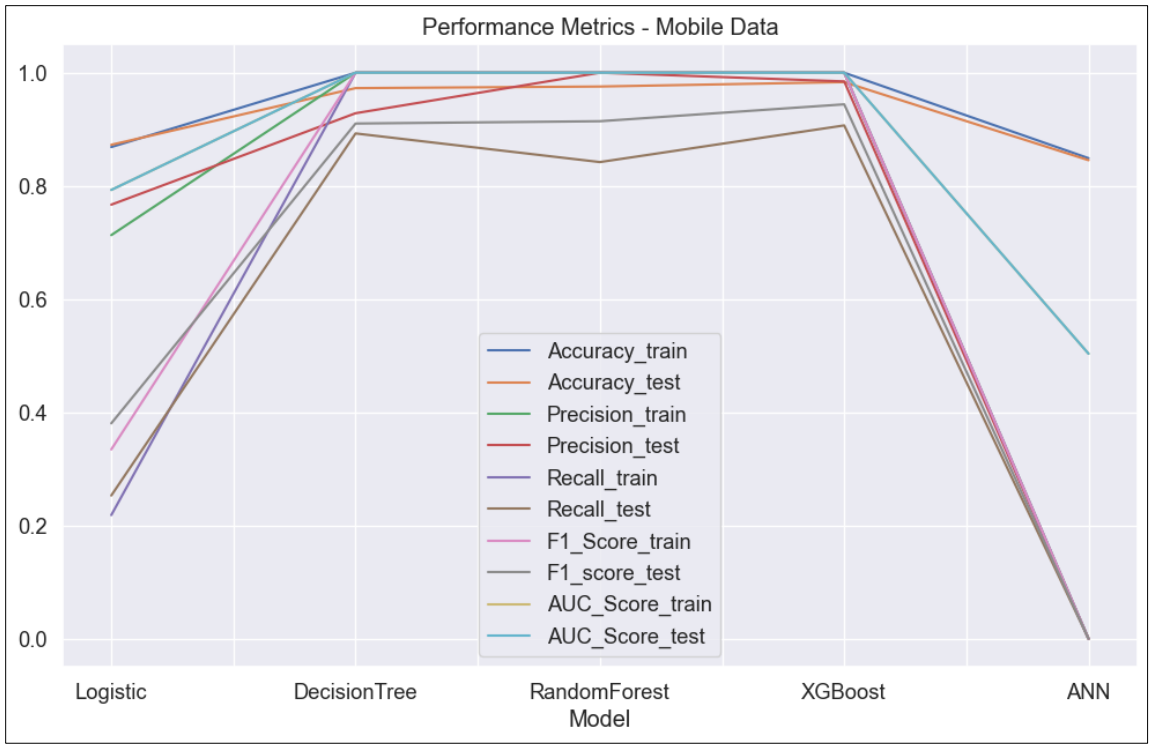


Figure 17: Performance metric (Default model)-Mobile

Insights for the default Model:

• Random Forest and XGBoost has the least number of False Negatives (Type II error) in the Positive Class (1)

• ANN is the least performing model followed by Logistic regression.

• However, out of 5 models 3 are overfitting which achieve perfect accuracy of 100% in Train data thus the model is not generalized enough. Hence, we shall tune the hyper-parameters for each model to achieve an optimal performance on both Train and Test data.

Hyperparameter Tuning – Mobile

Tune the hyper-parameter which can prune the trees and avoid over-fitting and reduce the complexity of the cost function in non-tree based models.

Performance Metric after Hyperparameter Tuning – Mobile.

After multiple iteration with the different parameter values for the hyperparameter, below are the performance metrics of the tuned models.

Model	Tuned_Logistic	Tuned_DecisionTree	Tune_RandomForest	Tuned_XGBoost	Tuned_ANN
Accuracy_train	0.869367	0.875	0.960569	0.994099	0.877548
Accuracy_test	0.87234	0.87985	0.938673	0.96214	0.873592
Precision_train	0.714689	0.677007	0.998801	0.998158	0.734066
Precision_test	0.75	0.741228	0.986928	0.967419	0.706422
Recall_train	0.224689	0.329485	0.739787	0.9627	0.296625
Recall_test	0.261134	0.342105	0.611336	0.781377	0.311741
F1_Score_train	0.341892	0.44325	0.85	0.980108	0.422517
F1_score_test	0.387387	0.468144	0.755	0.864502	0.432584
auc_train	0.793334	0.810145	0.997979	0.999734	0.828542
auc_test	0.800955	0.804845	0.985175	0.988815	0.822636

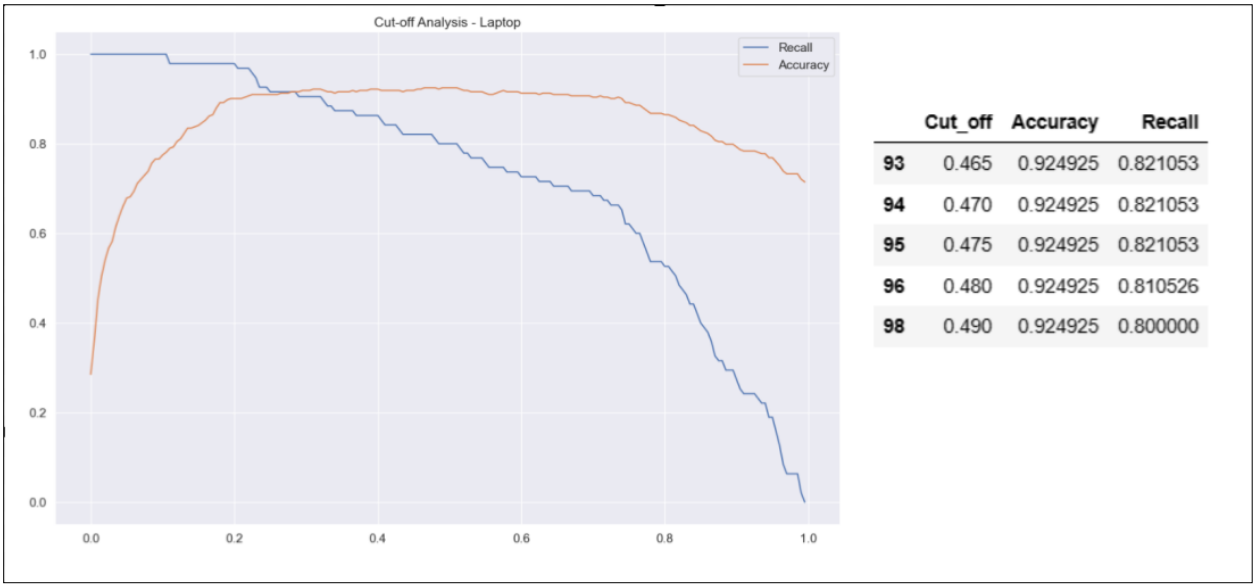
Table 22: Performance Metrics after Tuning – Mobile

After fine tuning them model to avoid over-fitting we can see that the XGBoost performance better that other model in terms of evaluation metrics. Also, it has a good consist performance between train and test data. Thus we will choose this our model of choice.

Cut Off Analysis

To find the optimal threshold of probability above which the prediction is consider as positive class of prediction. We shall find such optimal threshold to achieve the maximum accuracy and Recall trade-off.

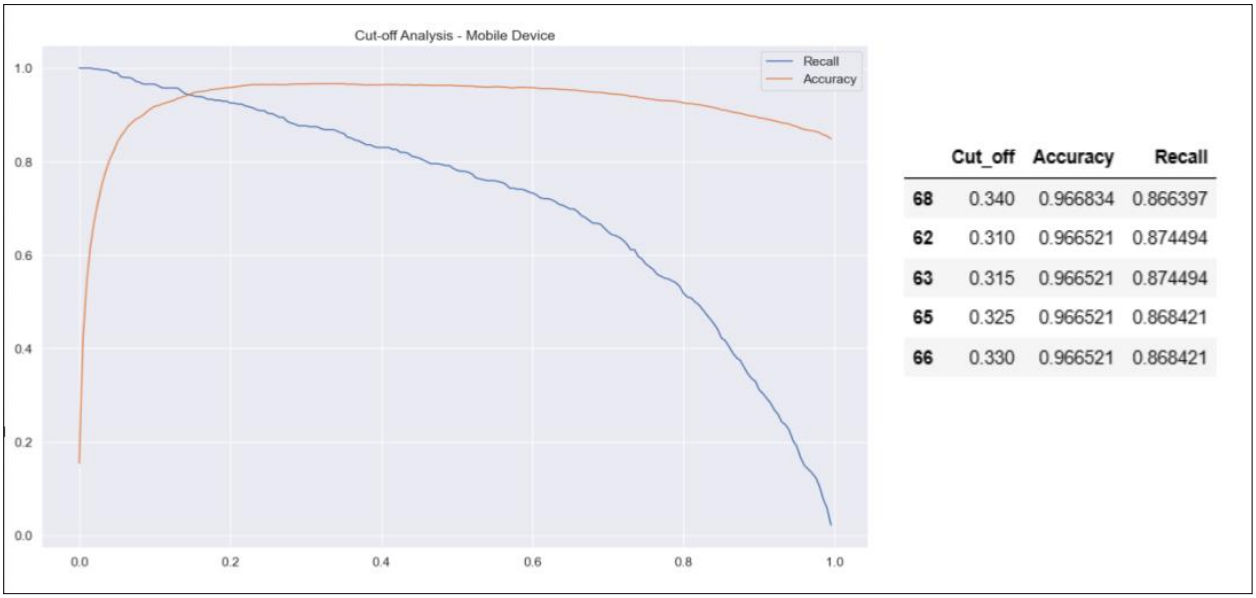
Laptop Data:



	Cut_off	Accuracy	Recall
93	0.465	0.924925	0.821053
94	0.470	0.924925	0.821053
95	0.475	0.924925	0.821053
96	0.480	0.924925	0.810526
98	0.490	0.924925	0.800000

Figure 18: Cut Off Analysis-Laptop

Mobile Data:



	Cut_off	Accuracy	Recall
68	0.340	0.966834	0.866397
62	0.310	0.966521	0.874494
63	0.315	0.966521	0.874494
65	0.325	0.966521	0.868421
66	0.330	0.966521	0.868421

Figure 19: Cut Off Analysis-Mobile

Let us consider 0.4 for Laptop and 0.3 for mobile as a cut-off and predict the class as 1 when it is above the given threshold.

Feature Importance:

The feature importance is calculated for a single decision tree by the amount that each attribute's split point improves the performance measure, weighted by the number of observations the node is responsible for.

Feature Importance for Laptop (XGB Model):

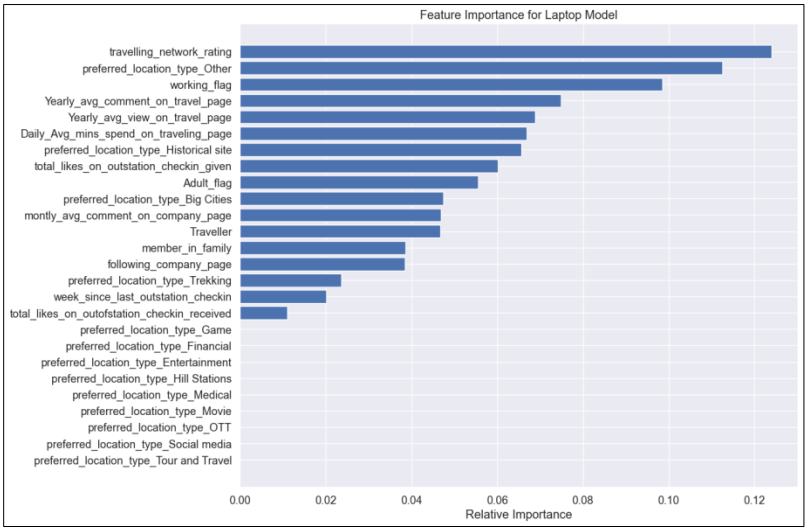


Figure 20: Feature Importance - Laptop

Feature Importance for Mobile (XGB Model):

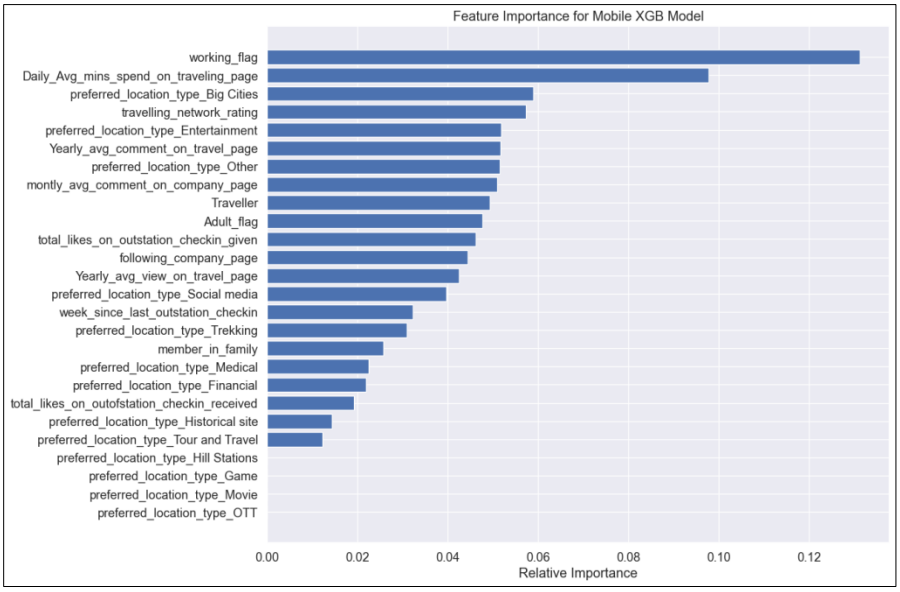


Figure 21: Feature Importance – Mobile

With the Importance feature of both the models from XGBoost we shall consider the top 10 feature to find the insights and which feature as good significance on the Target variable.

Significance on Target variable-Laptop:

Numerical variables:

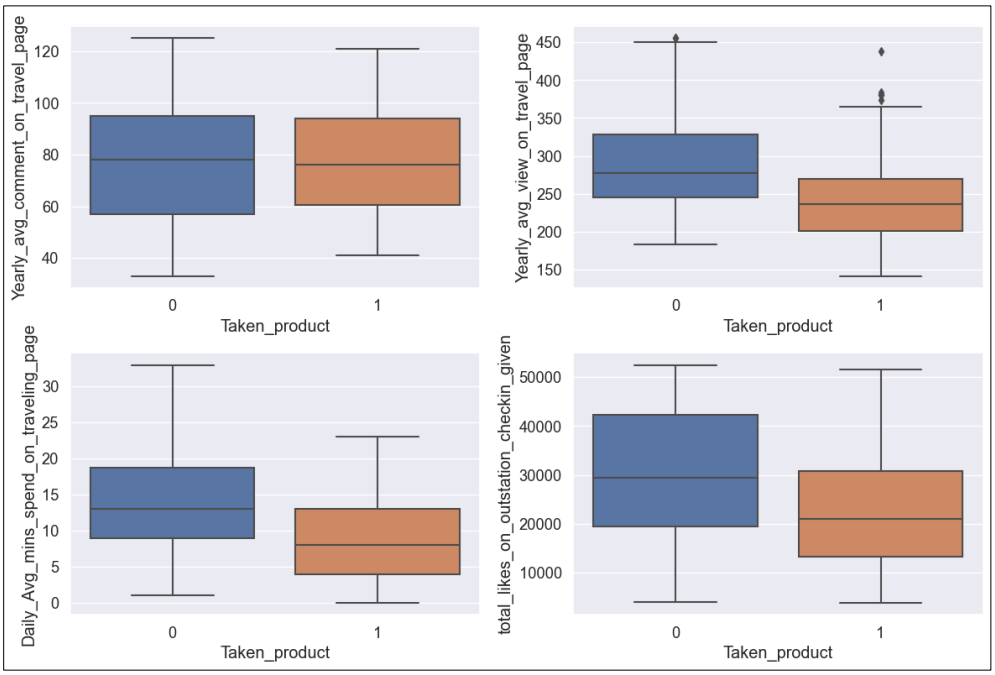


Figure 22: Significance on Target variable -Numerical-Laptop

Categorical variables:

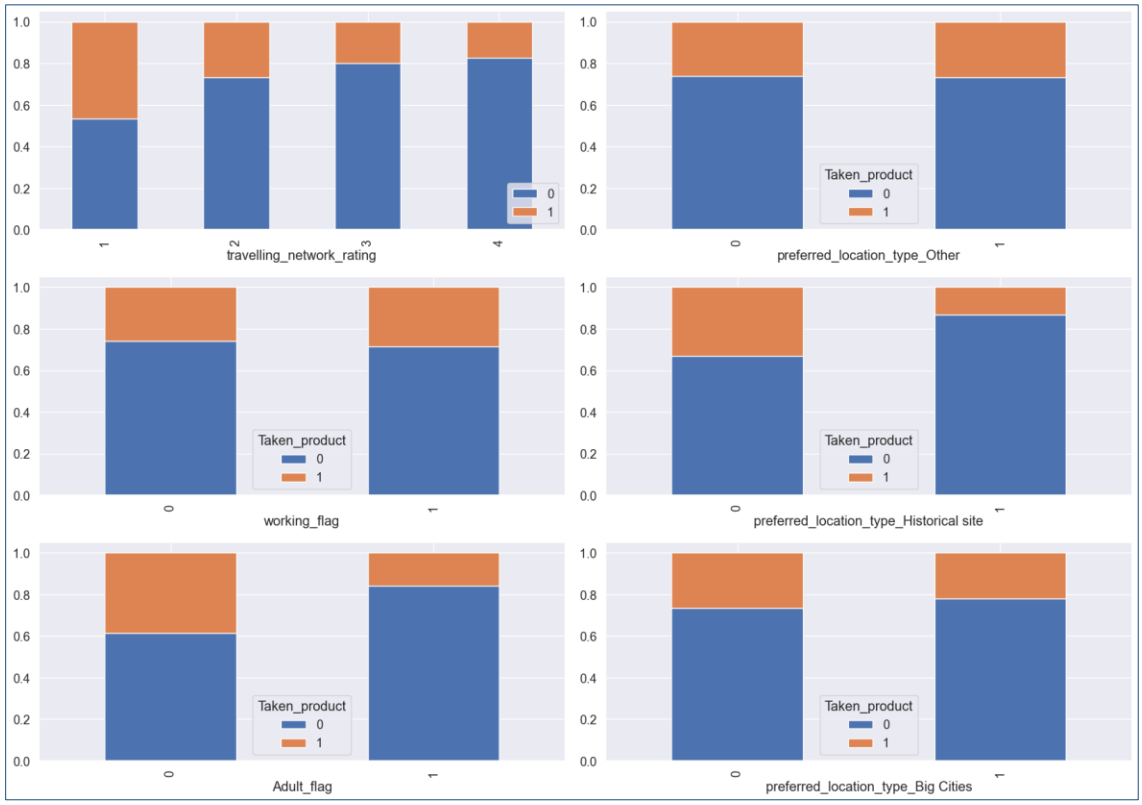


Figure 23: Significance on Target variable -Categorical-Laptop

Insights for Laptop as Preferred Device:

- Travelling network rating has the highest percentage of converting to prefer the company's product. Out of the four class of Travel rating user who with the least Travel network rating (rating 1) has high ~50% chance of buying the product than other ratings. but overall taking the product is more than 12%.
- User, who is working, has ~10% more chance of buying the company's product.
- The chances of buying of product on the avg comments on the travel page are upto 7.5%.
- If the customer is not Adult, then the chances of purchasing the product is more.

Significance on Target variable-Mobile:

Numerical variables:

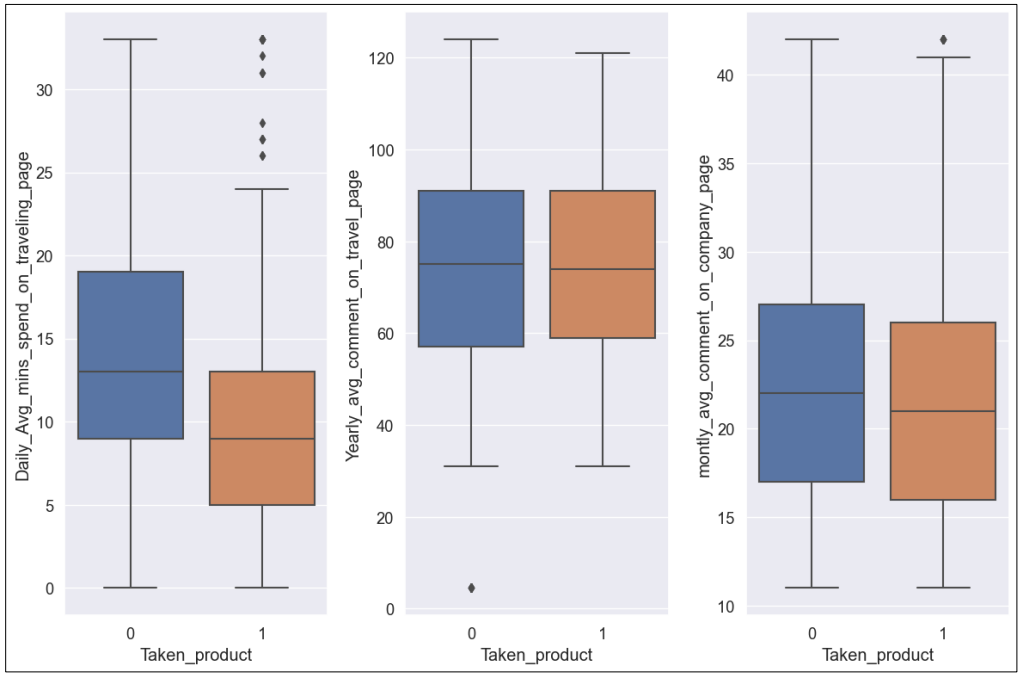


Figure 24: Significance on Target variable -Numerical-Mobile.

Categorical variables:

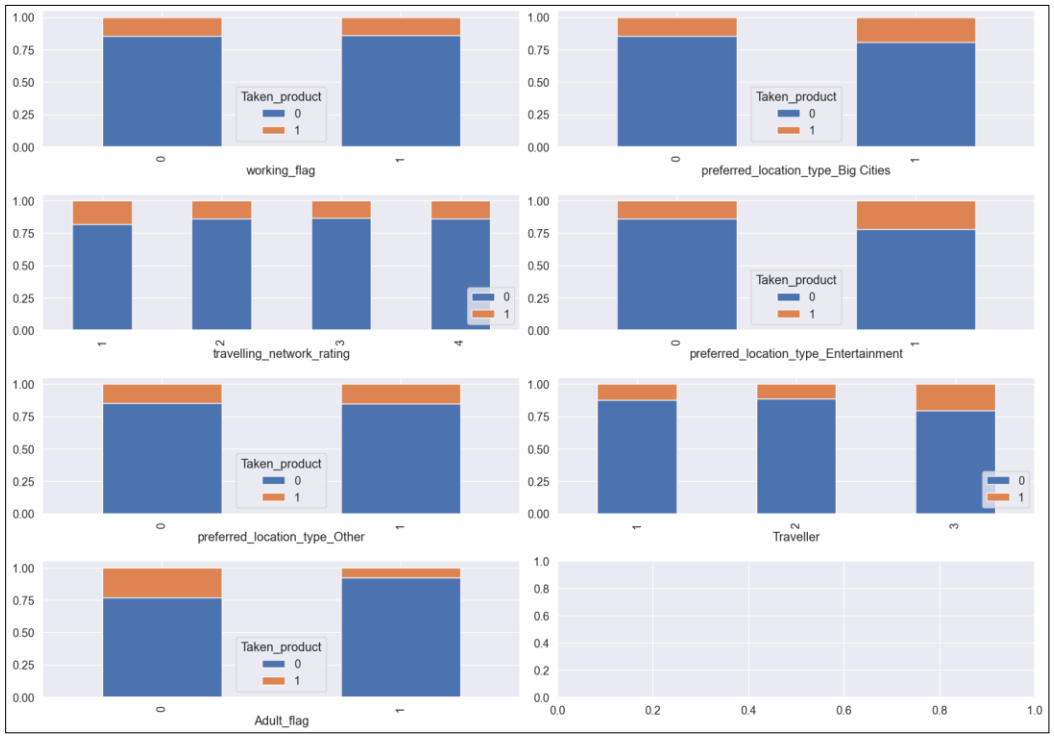


Figure 25: Significance on Target variable -Categorical-Mobile.

Insights for Mobile as Preferred Device:

- It does not matter if the customer is working or not, the percentage of taking the product lies between 15-20%. The overall percentage is 13%.
- Users who are not 'Adult' have ~20% more chance of buying the product.
- User who has Entertainment as their preferred location has a ~20% more chance in converting.
- Irrespective of Traveler network rating, the chances of taking the product lies between 10-15%.
- Frequent traveler is having up to 20% chances of taking product.

User Profiling for Targeted Digital Marketing:

After finding the significant variables we shall now cluster the users based on their behaviour on these particularly importance variables, which then help us to target the user group who as better metrics in the importance features we have selected.

By doing K-means Clustering with the optimal cluster of 4 we shall group the users into 4 clusters for both Laptop and Mobile device.

Profiling for Laptop users:

Value count in each cluster:

Cluster 2: 600
Cluster 4: 344
Cluster 1: 108
Cluster 3: 56

Percentage of 'Taken product' in each customer:

cluster	Taken_product	
Cluster-1	0	0.703704
	1	0.296296
Cluster-2	0	0.700000
	1	0.300000
Cluster-3	0	0.785714
	1	0.214286
Cluster-4	0	0.848837
	1	0.151163

Table 23: Percentage of taken product in each clusters- Laptop

Cluster	Adult_flag	preferred_location_type_Big Cities	preferred_location_type_Historical site	preferred_location_type_Other	travelling_network_rating	working_flag	Taken_product
1	1	1	0	0	3	0	29.60%
2	1	0	0	0	3	0	30%
3	1	0	0	1	3	0	21.40%
4	1	0	1	0	3	0	15.10%

Table 24: Profiling for Laptop Users

Profiling for Mobile users:

Value count in each cluster:

Cluster 1: 4838

Cluster 2: 3741

Cluster 4: 1557

Cluster 3: 516

Percentage of 'Taken product' in each customer:

cluster	Taken_product	
Cluster-1	0	0.841112
	1	0.158888
Cluster-2	0	0.858700
	1	0.141300
Cluster-3	0	0.872232
	1	0.127768
Cluster-4	0	0.761628
	1	0.238372

Table 25: Percentage of taken product in each clusters-Mobile

Cluster	Adult_flag	Traveller	preferred_location_type_Big Cities	preferred_location_type_Entertainment	preferred_location_type_Other	travelling_network_rating	working_flag	Taken_product
1	1	3	0	0	0	3	0	15.88%
2	0	1	0	0	0	3	0	14%
3	1	2	0	0	1	3	0	12.77%
4	1	2	0	1	0	3	0	23.8%

Table 26: Profiling for Mobile Users

Business Insights:

Business insights for Laptop users.

For the users who prefer Laptop devices, we shall target the below customers with the digital advertisement as they may tend to convert as a potential customer:

- Users who are **adults, non-working** and **travelling_network_rating as 3** has a very significant toward preferring the company's product.
- User who prefer the location type as **big cities, Historical sites** or as **others** also has significance in taking the company's product.

Business insights for mobile users.

For the users who prefer Mobile devices, we shall target the below customers with the digital advertisement as they may tend to convert as a potential customer:

- Users who are **adults, non-working** and **travelling_network_rating as 3** has a very significant toward preferring the company's product.
- User who is **moderate traveller** has the more chances of taking the product compare the frequent travellers and non-travellers.
- Users who are taking the products are who do not **prefer the big cities**.
- User who prefers the location type as **Entertainment** or as **others** also has significance in taking the company's product.

General Business Insights.

- It is also important to target the customer who is wrongly predicted as converted to improve the sale. But, this shall be give second priority only when there is still a budget to spend.
- Overall the most significant variables to concentrate shall be Adult flag, working flag, Travel frequency, Travelling network rating and the types of preferred locations by the users. i.e. high propensity to buy ticket for their next trip. Thus those user clusters can be profiled and Target for digital Marketing.
- It is a good idea to promote the business through paid promotions over frequent travelers' social media profiles as it can influence other travelers.
- Be active in Business social media handles, post contents on latest plans and offers and detailed videos on locations.
- Conduct social media contests by collaborating with influencers to reach more people. Discount coupons and reference benefits can enhance the sale.

---------- END ----------