Final Capstone Project Submission



- by Shivaprasad Shetti

Capstone Project

|  |  |
| --- | --- |
| **Review Parameters** | **Review Points** |
| **1. Introduction** | 1-2 |
| - Brief introduction about the problem statement and the need of solving it. |  |
|  |  |
| **2. EDA and Business Implication** | 3-19 |
| - Uni-variate / Bi-variate / Multi-variate analysis to understand relationship b/w variables. How your analysis is impacting the business? |  |
| - Both visual and non-visual understanding of the data. |  |
|  |  |
| **3. Data Cleaning and Pre-processing** | 19-45 |
| - Approach used for identifying and treating missing values and outlier treatment (and why) |  |
| - Need for variable transformation (if any) |  |
| - Variables removed or added and why (if any) |  |
|  |  |
| **4. Model building** | 40-46 |
| - Clear on why was a particular model(s) chosen. |  |
| - Effort to improve model performance. |  |
|  |  |
| **5. Model validation** | 47-48 |
| - How was the model validated? Just accuracy, or anything else too? |  |
|  |  |
| **6. Final interpretation / recommendation** | 8 |
| - Detailed recommendations for the management/client based on the analysis done. |  |

**1. Introduction**

a) Defining problem statement

**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 tele calling. Hence, they have collaborated with a social networking platform, so they can learn the digital and social behaviour 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.

Brief introduction about the problem statement and the need of solving it.

b) Need of the study/project

* **This is a classification Problem and can be solved using Clustering Technique(K-Means)**
* **The need of this Project is to assist company on boosting their sales of Booking tickets online by checking the Behaviour of the Customers and building a model such that the company can take decision on posting the digital advertisement for the Targeted Customers.**

c) Understanding business/social opportunity

Social networking platform can help us understanding digital and social behaviour 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.]

**This is a classification Problem and can be solved using Clustering Technique(K-Means)**

**2. EDA and Business Implication**

Count of missing values means the columns have null values

A screenshot of a computer program

Description automatically generated with low confidence

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

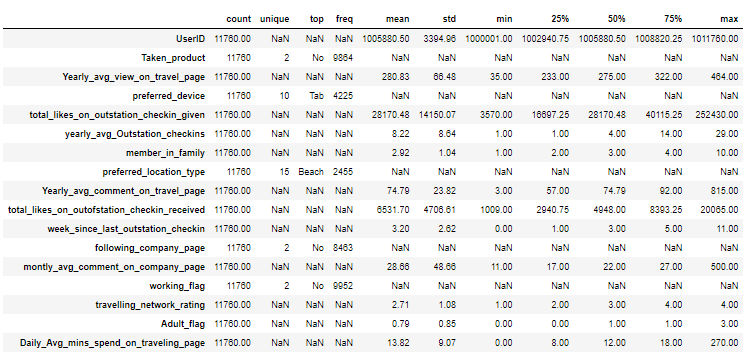
* This gives information at count and Frequency level
* There are11760 Row entries starting from 0 to11759
* There are 17 Column entries.

**Information of data before Pre-Processing**

A screenshot of a computer program

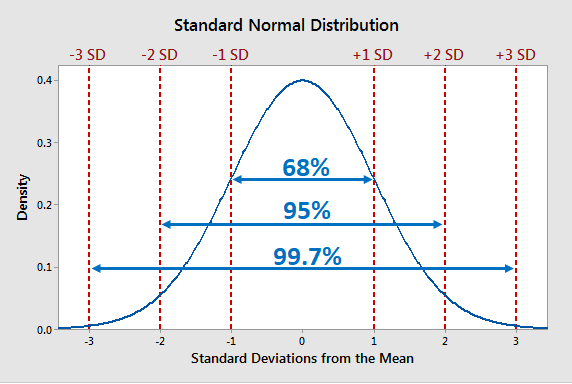
Description automatically generated with medium confidence

**Description of data**



* We have included Object type as well along with Numerical values for Describing the data
* As per Empirical rule 68% of **Yearly\_avg\_view\_on\_travel page** lies between (280-67)=212 to (280+67)=348

**Note:** Mean+/-Std Deviation



* Total Count of **Taken \_Product** is 11760 we have 2 unique values. Top repeated is “No” which occurs 9864 times.
* **Preferred device** to book product most preferred device was ‘**Tab**’
* **Preferred location type** mostselected were  **‘Beach’**
* There are 10 Different types of **Preferred Devices** used.
* **25 % of Total\_likes\_on\_oustation checkin**\_given ranges from 3570 to 16697
* Yearly Average View on Travel Page total count is 11760 mean is 280.83.
* minimum value in Yearly Average View on Travel Page is 35
* 25% viewers Yearly Average View on Travel Page ranges from 35 to 233
* 50% viewers Yearly Average View on Travel Page is 35 to 275
* 75% viewers Yearly Average View on Travel Page is 35 to 322
* Max value in Yearly Average View on Travel Page is 464

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

**Unique values for categorical variables**

|  |
| --- |
| TAKEN\_PRODUCT : 2 |
| Yes 1896 |
| No 9864 |
| Name: Taken\_product, dtype: int64 |
| PREFERRED\_DEVICE : 10 |
| Other 2 |
| Others 2 |
| ANDROID 134 |
| Android OS 145 |
| Android 315 |
| Mobile 600 |
| iOS 1095 |
| Laptop 1108 |
| iOS and Android 4134 |
| Tab 4172 |
| Name: preferred\_device, dtype: int64 |
| YEARLY\_AVG\_OUTSTATION\_CHECKINS : 30 |
| \* 1 |
| 27 96 |
| 21 143 |
| 13 150 |
| 22 152 |
| 12 159 |
| 17 160 |
| 14 167 |
| 19 176 |
| 28 180 |
| 25 198 |
| 20 199 |
| 26 199 |
| 15 206 |
| 18 208 |
| 23 215 |
| 29 215 |
| 24 223 |
| 11 229 |
| 6 236 |
| 16 255 |
| 4 256 |
| 5 261 |
| 8 320 |
| 3 336 |
| 7 336 |
| 9 340 |
| 10 682 |
| 2 844 |
| 1 4543 |
| Name: yearly\_avg\_Outstation\_checkins, dtype: int64 |

|  |
| --- |
| MEMBER\_IN\_FAMILY : 7 |
| 10 11 |
| Three 15 |
| 5 384 |
| 1 1349 |
| 2 2256 |
| 4 3184 |
| 3 4561 |
| Name: member\_in\_family, dtype: int64 |

|  |
| --- |
| PREFERRED\_LOCATION\_TYPE : 15 |
| Movie 5 |
| OTT 7 |
| Game 12 |
| Tour and Travel 47 |
| Tour Travel 60 |
| Hill Stations 108 |
| Entertainment 516 |
| Trekking 528 |
| Social media 633 |
| Big Cities 636 |
| Other 643 |
| Medical 1845 |
| Historical site 1856 |
| Financial 2409 |
| Beach 2424 |
| Name: preferred\_location\_type, dtype: int64 |

|  |
| --- |
| FOLLOWING\_COMPANY\_PAGE : 4 |
| 0 5 |
| 1 12 |
| Yes 3285 |
| No 8355 |
| Name: following\_company\_page, dtype: int64 |

|  |
| --- |
| WORKING\_FLAG : 2 |
| Yes 1808 |
| No 9952 |
| Name: working\_flag, dtype: int64 |

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

A picture containing text, screenshot, number, font

Description automatically generated

A screenshot of a computer program

Description automatically generated with medium confidence

**yearly\_avg\_Outstation\_checkins** Dtype should be **Float64 and not Object since its Numerical continuous data**

Uni-variate / Bi-variate / Multi-variate analysis to understand relationship b/w variables. How your analysis is impacting the business?

**3) Exploratory data analysis**

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

Description of Yearly\_avg\_view\_on\_travel\_page

----------------------------------------------------------------------------

count 11760.000000

mean 280.345153

std 66.511330

min 35.000000

25% 233.000000

50% 271.000000

75% 322.000000

max 464.000000

Name: Yearly\_avg\_view\_on\_travel\_page, dtype: float64 Distribution of Yearly\_avg\_view\_on\_travel\_page

A picture containing diagram, screenshot, plot, text

Description automatically generated

BoxPlot of Yearly\_avg\_view\_on\_travel\_page

----------------------------------------------------------------------- A picture containing diagram, rectangle, line, screenshot

Description automatically generated

**Observation for the Box Plot:**

Minimun Yearly Average view on Travel page is 35 which can be seen as Outlier.

Maximum Yearly Average view on Travel page 464 which is alos an outlier

BoxPlot of montly\_avg\_comment\_on\_company\_page

count 11760.000000

mean 28.661565

std 48.660504

min 11.000000

25% 17.000000

50% 22.000000

75% 27.000000

max 500.000000

Name: montly\_avg\_comment\_on\_company\_page, dtype: float64 Distribution of montly\_avg\_comment\_on\_company\_page

A picture containing text, screenshot, display, rectangle

Description automatically generated

A picture containing text, screenshot, line, rectangle

Description automatically generated

Monthly Average comment on company Page Minimum is 11

Monthly Average comment on company Page Maximum is 500 which is between 300 to 500 range there are outlier

Most of the value are in the range of 0-100 seen is Histogram

Description of travelling\_network\_rating

----------------------------------------------------------------------------

count 11760.000000

mean 2.712245

std 1.080887

min 1.000000

25% 2.000000

50% 3.000000

75% 4.000000

max 4.000000

Name: travelling\_network\_rating, dtype: float64 Distribution of travelling\_network\_rating

A picture containing text, screenshot, diagram, line

Description automatically generated

BoxPlot of travelling\_network\_rating

----------------------------------------------------------------------------

A picture containing display, screenshot, rectangle, text

Description automatically generated

* From the Box plot and His plot we can see that Most of the Customer rating is in range of 3 to 4

**A picture containing screenshot, text, rectangle, diagram

Description automatically generated**

A picture containing text, screenshot, diagram, rectangle

Description automatically generated

For the above Bar Graph we can describe that

* Mobile Users are more in Number than Laptop users

Mobile 10652

Laptop 1108

* Ratio of Product not taken is more than the Product taken as this is out Traget variable and data looks imbalanced

No 9864

Yes 1896

* Users Not following Company page is more that the User who folloe xompany page

**- Both visual and non-visual understanding of the data.**

1. **Bivariate analysis (relationship between different variables , correlations)**

**Outliers are present in the below data**

Monthly average comment on Company page vs Following Company \_Page is ranging from 11 to 500.

**A picture containing text, screenshot, line, diagram

Description automatically generated**

**Outliers Box plot after Treating Outliers**

Monthly average comment on Company page vs Taken\_Product is ranging from 11 to 500.

**A picture containing text, screenshot, display, rectangle

Description automatically generated**

**A picture containing text, screenshot, diagram, rectangle

Description automatically generated**

For the above Bar Graph we can describe that

* Non working Mobile Users are more in munber than working Mobile Users
* Non working Laptop Users are more in munber than working Laptop Users

Multivariate Analysis on Members in Family vs Taken Product

**A picture containing text, screenshot, font, number

Description automatically generated**

**A picture containing screenshot, text, diagram, plot

Description automatically generated**

From the above Bar Graph for the given data we can see that :

* Most of the Product taken by the family members are 3.
* Least of the product taken by family members are 10

A picture containing text, screenshot, number, font

Description automatically generated

From the above cross tab for the given data we can see that:

* Most of the Users taken Product and not taken product their most Preferred travel Destination are “**Beaches**”
* Most of the Users taken Product and not taken product their second most preferred travel Destination are “**Big Cities**”

**Bivariate Analysis(Numeric vs Nummeric):**

**A picture containing screenshot, text, plot, line

Description automatically generated**

Observation:

* total\_likes\_on\_outofstation\_checkin\_received increases as the Daily\_Avg\_mins\_spend\_on\_traveling\_page is increased.

**Multivariate Analysis:**

**A picture containing diagram, screenshot, rectangle, line

Description automatically generated**

**Observation:**

* **More than 3 members in family prefer to take the product using preferred device as Mobile.**

**Correlations:**

A screenshot of a computer

Description automatically generated with medium confidence

* **Total\_likes\_on outstation checkins recieved is highly correlated with**

**Daily\_Avg\_mins\_spend\_on\_travelling\_page Correlation is 67%**

* **Yearly\_avg\_view\_on travel\_page is Highly correleated with Daily\_Avg\_mins\_spend\_on\_travelling\_page Correlation is 58%**
* **Yearly\_avg\_view\_on travel\_page is highly corelated with Total\_likes\_on\_outstaion\_checkins\_recieved Correlation is 48%**

**After dropping parameters like Daily\_Avg\_mins\_spend\_on\_travelling\_page** and **Yearly\_avg\_view\_on travel\_page which are highly correlated the plot looks like below:**

**A screenshot of a computer

Description automatically generated with medium confidence**

**In the above plot now Highly, correlated attributes are eliminated for further computation**

**3. Data Cleaning and Pre-processing**

a) Removal of unwanted variables (if applicable)

**Below are the Bad Data**

preferred\_location\_type has Missing values

['Financial', 'Other', 'Medical', nan, 'Game', 'Social media',

'Entertainment', 'Tour and Travel', 'Movie', 'OTT', 'Tour Travel',

'Beach', 'Historical site', 'Big Cities', 'Trekking',

'Hill Stations']

yearly\_avg\_Outstation\_checkins has Missing values

['1', '24', '23', '27', '16', '15', '26', '19', '21', '11', '10',

'25', '12', '18', '29', nan, '22', '14', '20', '28', '17', '13',

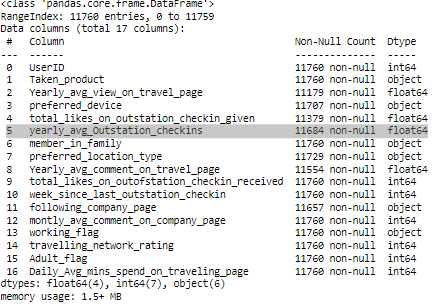
'\*', '5', '8', '2', '3', '9', '7', '6', '4']

following\_company\_page has Missing values and replace values 1 to Yes and 0 to No

['Yes', 'No', nan, '1', '0']

Treating of Bad data

Now we Observe that.



Bad data has been treated for **yearly\_avg\_Outstation\_checkins** datatype has been converted to **float64** as this is the numerical continuous data

member\_in\_family had Bad data as ‘Three’ which has been replaced to 3.

following\_company\_page has Missing values and bad data as 1 and 0 replace values 1 to Yes and 0 to No

b) Missing Value treatment (if applicable)

# Treatment of missing values for Numerical variables

* **yearly\_avg\_Outstation\_checkins** had bad data ‘\*’ which was replaced to null values and then null values was replaced to the value which is mean of total **yearly\_avg\_Outstation\_checkins**
* **Yearly\_avg\_view\_on\_travel\_page** had null values which was replaced to the value which is mean of total **Yearly\_avg\_view\_on\_travel\_page**
* **total\_likes\_on\_outstation\_checkin\_given** had null values which was replaced to the value which is mean of total **total\_likes\_on\_outstation\_checkin\_given**
* **Yearly\_avg\_comment\_on\_travel\_page** had null values which was replaced to the value which is mean of total **Yearly\_avg\_comment\_on\_travel\_page**

**Dataset after treating Null values for Numerical variables**

**A screenshot of a computer program

Description automatically generated with medium confidence**

# Treatment of missing values for Categorical variables

* **following\_company\_page** had null values which was replaced to the value which is mode of highest repeated value in **following\_company\_page**
* **preferred\_device** had null values which was replaced to the value which is mode of highest repeated value in **preferred\_device**
* **preferred\_location\_type** had null values which was replaced to the value which is mode of highest repeated value in **preferred\_location\_type**

**Dataset after treating Null values for Categorical variables**

**A screen shot of a computer program

Description automatically generated with low confidence**

**Note:**

* **Mean imputation is often used** when the missing values are numerical and the **distribution of the variable is approximately normal.**
* **Median imputation is preferred** when **the distribution is skewed**, as the median is less sensitive to outliers than the mean.
* **Mode imputation** is suitable for **categorical variables** or numerical variables with a small number of unique values.

**4) Business insights from EDA**

Data had missing values and Bad data which has been treated by imputation technique,

Dataset also has Outliers. Since this is a classification problem so outlier treatment is not required

Outliers can bee seen in the given dataset.

A picture containing line, screenshot, plot, parallel

Description automatically generated

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

Data is unbalanced

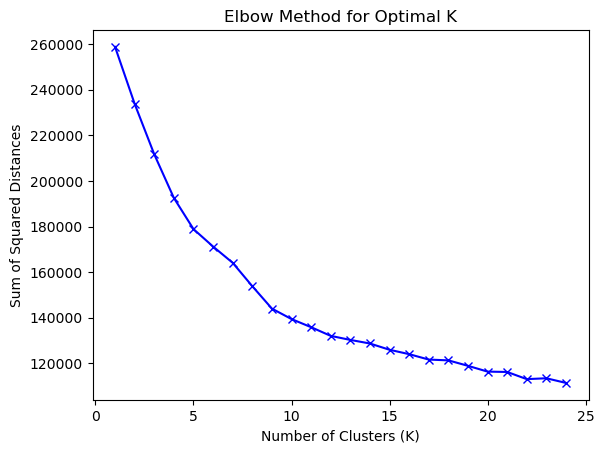
* When check the Target variable distribution In this case it is Taken\_Product
* From the below inference Our variable of Interest is 1 that is ‘Yes’ but its has very less count compared with 0 that is ‘No’ This indicates the data Imbalance

The number of classes before fit Counter({0: 6905, 1: 1327})

The number of classes after fit Counter({0: 6905, 1: 5178})

**Note: Balancing of the data is done using SMOTE**

b) Any business insights using clustering (if applicable)



From the above plot we can consider Cluster 9 as optimal and that is the elbow point( point at which it starts decreasing linearly ).

|  |  |
| --- | --- |
| **Within Sum of Squares** | **Cluster** |
| 258720 | 1 |
| 233731.3694 | 2 |
| 211611.2308 | 3 |
| 192500.9083 | 4 |
| 178859.5827 | 5 |
| 171208.5154 | 6 |
| 164079.4409 | 7 |
| 153828.3094 | 8 |
| 144011.0775 | 9(Elbow point) |
| 139424.9702 | 10 |
| 135863.559 | 11 |
| 132055.2968 | 12 |
| 130286.4412 | 13 |
| 128716.5029 | 14 |
| 125966.2502 | 15 |
| 124044.228 | 16 |
| 121669.6465 | 17 |
| 121320.9659 | 18 |
| 118932.1024 | 19 |
| 116390.8101 | 20 |
| 116236.9169 | 21 |
| 113096.0389 | 22 |
| 113474.2954 | 23 |
| 111434.8268 | 24 |

|  |  |
| --- | --- |
| Column Number | Column names in Clustering for scaled data |
| 0 | total\_likes\_on\_outstation\_checkin\_given |
| 1 | yearly\_avg\_Outstation\_checkins |
| 2 | member\_in\_family |
| 3 | Yearly\_avg\_comment\_on\_travel\_page |
| 4 | total\_likes\_on\_outofstation\_checkin\_received |
| 5 | week\_since\_last\_outstation\_checkin |
| 6 | montly\_avg\_comment\_on\_company\_page |
| 7 | travelling\_network\_rating |
| 8 | Adult\_flag |
| 9 | Taken\_product\_No |
| 10 | Taken\_product\_Yes |
| 11 | preferred\_device\_Laptop |
| 12 | preferred\_device\_Mobile |
| 13 | preferred\_location\_type\_Beach |
| 14 | preferred\_location\_type\_Financial |
| 15 | preferred\_location\_type\_Historical site |
| 16 | preferred\_location\_type\_Medical |
| 17 | preferred\_location\_type\_Other |
| 18 | following\_company\_page\_No |
| 19 | following\_company\_page\_Yes |
| 20 | working\_flag\_No |
| 21 | working\_flag\_Yes |



Observation of Cluster Distribution on categorical variable above table and plot:

Considering the attribute Members in Family

Cluster 1 has Highest Accumulation of Sum of members in family.

Cluster 4 has second highest accumulation Sum of members in family.

Cluster 5 has least accumulation Sum of members in family.

|  |  |
| --- | --- |
| **Cluster** | **Sum of total\_likes\_on\_outstation\_checkin\_given** |
| 0 | 30768788 |
| 1 | 54069848 |
| 2 | 42626906 |
| 3 | 38724871 |
| 4 | 49606085 |
| 5 | 5518878 |
| 6 | 40676474 |
| 7 | 32188448 |
| 8 | 37068570 |

Observation of Cluster Distribution on Continuous Variable above table and plot:

Cluster 1 has Highest Accumulation of **Sum of total\_likes\_on\_outstation\_checkin\_given**.

Cluster 4 has second highest accumulation of of **Sum of total\_likes\_on\_outstation\_checkin\_given**

Cluster 5 has least accumulation of **Sum of total\_likes\_on\_outstation\_checkin\_given**.

Clustering Inference for **Taken Product\_Yes** vs **Sum\_of\_members in Family**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Clusters** | **Sum of Taken\_product\_Yes** | **Rank Sum of Taken\_product\_Yes** | **Sum of member\_in\_family** | **Rank Sum of member\_in\_family** |
| 0 | 276 | 2 | 3208 | 7 |
| 1 | 0 | 5 | 6035 | 1 |
| 2 | 0 | 5 | 4425 | 3 |
| 3 | 0 | 5 | 3802 | 5 |
| 4 | 0 | 5 | 5147 | 2 |
| 5 | 33 | 4 | 551 | 9 |
| 6 | 1517 | 1 | 4323 | 4 |
| 7 | 0 | 5 | 3136 | 8 |
| 8 | 70 | 3 | 3728 | 6 |

Cluster 6: Product taken in Cluster 1 has the Highest count Ranking is 1 for which the Sum of members in Family are 4323 Ranking is 4.

Cluster 1:Rank Sum of members\_in\_family has the Highest count Ranking is 1 for which the Taken product is 0 means nobody taken product.

c) Any other business insights

The below table we understand that.

1. 18 % of non-Working people travel to **Beaches.**
2. Non-working people travel more than working people.
3. 4% of non-Working people travel to **Trekking.**

A picture containing text, screenshot, font, number

Description automatically generated

* More people tend to travel to **beaches** and **Financial Institution**
* Least number of people travel to **Movie,OTT,Game,Hill station,Tour and Travel**

A picture containing text, screenshot, line, plot

Description automatically generated

Tab ,IOS Andriod Users are more engaged in Travel stuff.

A picture containing screenshot, text, rectangle, line

Description automatically generated

- Approach used for identifying and treating missing values and outlier treatment (and why)

For Logistic Regression, LDA model, Kmeans are sensitive to outliers so outliers will have to be treated.

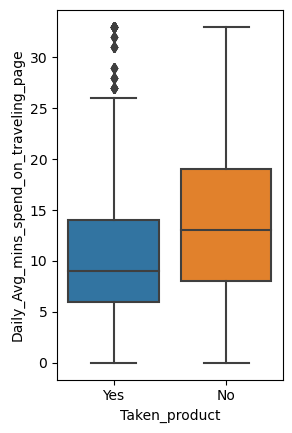
After treating Outliers

Customer buying the product after spending Daily Average minutes on travelling page is less.

Customer buying or not buying the product after spending monthly avg comment on company page is 50-50 chance

Customer buying or not buying the product after spending Yearly\_avg\_view\_on\_travel\_page is 50-50 chance

Customer buying the product even after total\_likes\_on\_outstation\_checkin\_given is less.

A picture containing diagram, rectangle, line, screenshot

Description automatically generated

A picture containing diagram, text, screenshot, rectangle

Description automatically generated A picture containing diagram, rectangle, screenshot, text

Description automatically generatedLDA Model on Balanced Data For Mobile Devices

Note: Outliers are not treated while this model was computed as we know that LDA Model are sensitive to Outliers

Tabular Form:

----------------------Train Model After SMOTE-------------------------

**LinearDiscriminantAnalysis**

Model performance for Training set

- Accuracy: 0.7570

- F1 score: 0.7561

-Precision: 0.7286

-Recall: 0.6900

-Roc Auc Score: 0.7486

precision recall f1-score support

0 0.78 0.81 0.79 6905

1 0.73 0.69 0.71 5178

accuracy 0.76 12083

macro avg 0.75 0.75 0.75 12083

weighted avg 0.76 0.76 0.76 12083

----------------------Confusion Matrix Train Model After SMOTE---------

[[5574 1331]

[1605 3573]]

----------------------Test Model After SMOTE-------------------------

Model performance for testing set

- Accuracy: 0.7710

- F1 score: 0.7905

-Precision: 0.3691

-Recall: 0.5923

-Roc Auc Score: 0.6988

precision recall f1-score support

0 0.91 0.81 0.86 2959

1 0.37 0.59 0.45 569

accuracy 0.77 3528

macro avg 0.64 0.70 0.65 3528

weighted avg 0.82 0.77 0.79 3528

----------------------Confusion Matrix Test Model After SMOTE---------

[[2383 576]

[ 232 337]]

Inferences

* In this model precision(0.37) and recall(0.59) are very poor for predicted test data hence this model is not good for evaluation
* Seeing the AUC Curve: The training AUC (0.809) is slightly higher than the test AUC (0.753).

This difference suggests that the model may have slightly overfit the training data.

* AUC for the Training Data: 0.809
* AUC for the Test Data: 0.753
* A picture containing text, plot, screenshot, line

  Description automatically generated

LDA Model on Balanced Data For Laptop Devices

Note: Outliers are not treated while this model was computed as we know that LDA Model are sensitive to Outliers

Tabular Form:

----------------------Train Model Aftre SMOTE-------------------------

LinearDiscriminantAnalysis

Model performance for Training set

- Accuracy: 0.7520

- F1 score: 0.7511

-Precision: 0.7224

-Recall: 0.6844

-Roc Auc Score: 0.7436

precision recall f1-score support

0 0.77 0.80 0.79 6905

1 0.72 0.68 0.70 5178

accuracy 0.75 12083

macro avg 0.75 0.74 0.75 12083

weighted avg 0.75 0.75 0.75 12083

----------------------Confusion Matrix Train Model After SMOTE-------------------------

[[5543 1362]

[1634 3544]]

----------------------Test Model After SMOTE-------------------------

Model performance for testing set

- Accuracy: 0.7599

- F1 score: 0.7821

-Precision: 0.3549

-Recall: 0.5975

-Roc Auc Score: 0.6943

precision recall f1-score support

0 0.91 0.79 0.85 2959

1 0.35 0.60 0.45 569

accuracy 0.76 3528

macro avg 0.63 0.69 0.65 3528

weighted avg 0.82 0.76 0.78 3528

----------------------Confusion Matrix Test Model After SMOTE-------------------------

[[2341 618]

[ 229 340]]

Inferences

* In this model precision(0.35) and recall(0.60) are very poor for predicted test data hence this model is not good for evaluation
* Seeing the AUC Curve: The training AUC (0.802) is slightly higher than the test AUC (0.750).

This difference suggests that the model may have slightly overfit the training data.

* AUC for the Training Data: 0.802
* AUC for the Test Data: 0.750

A picture containing text, plot, screenshot, line

Description automatically generated

Logistic Regression Model on Balanced Data For Mobile Devices

Logistic Regression model on Balanced Data

Note: Outliers are not treated while this model was computed as we know that Logistic Model are sensitive to Outliers

**----------------------Train Model After SMOTE-------------------------**

**Logistic Regression**

Model performance for Training set

- Accuracy: 0.6186

- F1 score: 0.5943

-Precision: 0.5942

-Recall: 0.3472

-Roc Auc Score: 0.5847

precision recall f1-score support

0 0.63 0.82 0.71 6905

1 0.59 0.35 0.44 5178

accuracy 0.62 12083

macro avg 0.61 0.58 0.57 12083

weighted avg 0.61 0.62 0.59 12083

**----------------------Confusion Matrix Train Model After SMOTE---------**

[[5677 1228]

[3380 1798]]

**----------------------Test Model After SMOTE-------------------------**

Model performance for testing set

- Accuracy: 0.7406

- F1 score: 0.7531

-Precision: 0.2630

-Recall: 0.3374

-Roc Auc Score: 0.5778

precision recall f1-score support

0 0.87 0.82 0.84 2959

1 0.26 0.34 0.30 569

accuracy 0.74 3528

macro avg 0.56 0.58 0.57 3528

weighted avg 0.77 0.74 0.75 3528

**----------------------Confusion Matrix Test Model After SMOTE--------**

[[2421 538]

[ 377 192]]

A picture containing text, plot, screenshot, line

Description automatically generated

Inference:

* Training vs. Test AUC Comparison: The training AUC (0.662) is slightly higher than the test AUC (0.656), but the difference is minimal. This suggests that the model's performance is relatively consistent between the training and test datasets. However, the similarity in AUC values also indicates that the model may not be significantly overfitting or underfitting the training data.
* In this model precision (0.26) and recall(0.34) are very poor for predicted test data hence this model is not good for evaluation

Logistic Regression Model on Balanced Data For Laptop Devices

Logistic Regression model on Balanced Data

Note: Outliers are not treated while this model was computed as we know that Logistic Model are sensitive to Outliers

----------------------Train Model After SMOTE-------------------------

Logistic Regression

Model performance for Training set

- Accuracy: 0.6219

- F1 score: 0.5985

-Precision: 0.5993

-Recall: 0.3550

-Roc Auc Score: 0.5885

precision recall f1-score support

0 0.63 0.82 0.71 6905

1 0.60 0.35 0.45 5178

accuracy 0.62 12083

macro avg 0.61 0.59 0.58 12083

weighted avg 0.62 0.62 0.60 12083

----------------------Confusion Matrix Train Model After SMOTE---------

[[5676 1229]

[3340 1838]]

----------------------Test Model After SMOTE-------------------------

Model performance for testing set

- Accuracy: 0.7489

- F1 score: 0.7605

-Precision: 0.2808

-Recall: 0.3568

-Roc Auc Score: 0.5905

precision recall f1-score support

0 0.87 0.82 0.85 2959

1 0.28 0.36 0.31 569

accuracy 0.75 3528

macro avg 0.58 0.59 0.58 3528

weighted avg 0.77 0.75 0.76 3528

----------------------Confusion Matrix Test Model After SMOTE----------

[[2439 520]

[ 366 203]]

AUC Curve:

AUC for the Training Data: 0.670

AUC for the Test Data: 0.679

A picture containing text, plot, screenshot, line

Description automatically generated

Inference:

* Training vs. Test AUC Comparison: The training AUC (0.670) is slightly higher than the test AUC (0.679), but the difference is minimal. This suggests that the model's performance is relatively consistent between the training and test datasets. However, the similarity in AUC values also indicates that the model may not be significantly overfitting or underfitting the training data.
* In this model precision(0.28) and recall(0.36) are very poor for predicted test data hence this model is not good for evaluation

Random Forest Model on Balanced Data For Mobile Devices

Random Forest model on Balanced Data

**----------------------Train Model After SMOTE-------------------------**

Random Forest

Model performance for Training set

- Accuracy: 1.0000

- F1 score: 1.0000

-Precision: 1.0000

-Recall: 1.0000

-Roc Auc Score: 1.0000

precision recall f1-score support

0 1.00 1.00 1.00 6905

1 1.00 1.00 1.00 5178

accuracy 1.00 12083

macro avg 1.00 1.00 1.00 12083

weighted avg 1.00 1.00 1.00 12083

**----------------------Confusion Matrix Train Model After SMOTE---------**

[[6905 0]

[ 0 5178]]

**----------------------Test Model After SMOTE-------------------------**

Model performance for testing set

- Accuracy: 0.9872

- F1 score: 0.9871

-Precision: 0.9799

-Recall: 0.9402

-Roc Auc Score: 0.9683

precision recall f1-score support

0 0.99 1.00 0.99 2959

1 0.98 0.94 0.96 569

accuracy 0.99 3528

macro avg 0.98 0.97 0.98 3528

weighted avg 0.99 0.99 0.99 3528

**----------------------Confusion Matrix Test Model After SMOTE----------**

[[2948 11]

[ 34 535]]

AUC for the Training Data: 1.000

AUC for the Test Data: 0.998

AUC Curve

A picture containing text, line, screenshot, plot

Description automatically generated

**Inference**:

* model has made no false positive predictions during training
* A precision score of 100 in the training set could potentially be a sign of overfitting. Overfitting occurs when a model performs extremely well on the training data but fails to generalize well to unseen data.
* model has memorized the training data instead of learning the underlying patterns. This model is not recommended
* precision score of 100 in a training model should be viewed with caution
* AUC values of 1.000 is rare in practice and may indicate potential issues such as overfitting or data leakage.

Random Forest Model on Balanced Data For Laptop Devices

Random Forest model on Balanced Data

----------------------Train Model After SMOTE-------------------------

Random Forest

Model performance for Training set

- Accuracy: 1.0000

- F1 score: 1.0000

-Precision: 1.0000

-Recall: 1.0000

-Roc Auc Score: 1.0000

precision recall f1-score support

0 1.00 1.00 1.00 6905

1 1.00 1.00 1.00 5178

accuracy 1.00 12083

macro avg 1.00 1.00 1.00 12083

weighted avg 1.00 1.00 1.00 12083

----------------------Confusion Matrix Train Model After SMOTE-------------------------

[[6905 0]

[ 0 5178]]

----------------------Test Model After SMOTE-------------------------

Model performance for testing set

- Accuracy: 0.9858

- F1 score: 0.9857

-Precision: 0.9761

-Recall: 0.9350

-Roc Auc Score: 0.9653

precision recall f1-score support

0 0.99 1.00 0.99 2959

1 0.98 0.93 0.96 569

accuracy 0.99 3528

macro avg 0.98 0.97 0.97 3528

weighted avg 0.99 0.99 0.99 3528

----------------------Confusion Matrix Test Model After SMOTE----------

[[2946 13]

[ 37 532]]

AUC for the Training Data: 1.000

AUC for the Test Data: 0.998

**A picture containing text, line, plot, screenshot

Description automatically generated**

**Inference**:

* model has made no false positive predictions during training
* A precision score of 100 in the training set could potentially be a sign of overfitting. Overfitting occurs when a model performs extremely well on the training data but fails to generalize well to unseen data.
* model has memorized the training data instead of learning the underlying patterns. This model is not recommended
* precision score of 100 in a training model should be viewed with caution
* AUC values of 1.000 is rare in practice and may indicate potential issues such as overfitting or data leakage.

Decision Tree Model on Balanced Data For Mobile Devices

Decision Tree Model on Balanced Data

**----------------------Train Model After SMOTE-------------------------**

**Decision Tree**

Model performance for Training set

- Accuracy: 1.0000

- F1 score: 1.0000

-Precision: 1.0000

-Recall: 1.0000

-Roc Auc Score: 1.0000

precision recall f1-score support

0 1.00 1.00 1.00 6905

1 1.00 1.00 1.00 5178

accuracy 1.00 12083

macro avg 1.00 1.00 1.00 12083

weighted avg 1.00 1.00 1.00 12083

**----------------------Confusion Matrix Train Model After SMOTE---------**

[[6905 0]

[ 0 5178]]

**----------------------Test Model After SMOTE-------------------------**

Model performance for testing set

- Accuracy: 0.9660

- F1 score: 0.9663

-Precision: 0.8748

-Recall: 0.9209

-Roc Auc Score: 0.9478

precision recall f1-score support

0 0.98 0.97 0.98 2959

1 0.87 0.92 0.90 569

accuracy 0.97 3528

macro avg 0.93 0.95 0.94 3528

weighted avg 0.97 0.97 0.97 3528

**----------------------Confusion Matrix Test Model After SMOTE----------**

[[2884 75]

[ 45 524]]

**Inference**:

* model has made no false positive predictions during training
* A precision score of 100 in the training set could potentially be a sign of overfitting. Overfitting occurs when a model performs extremely well on the training data but fails to generalize well to unseen data.
* model has memorized the training data instead of learning the underlying patterns. This model is not recommended
* precision score of 100 in a training model should be viewed with caution
* AUC values of 1.000 is rare in practice and may indicate potential issues such as overfitting or data leakage.

Decision Tree Model on Balanced Data For Laptop Devices

Decision Tree Model on Balanced Data

----------------------Train Model After SMOTE-----------------

Decision Tree

Model performance for Training set

- Accuracy: 1.0000

- F1 score: 1.0000

-Precision: 1.0000

-Recall: 1.0000

-Roc Auc Score: 1.0000

precision recall f1-score support

0 1.00 1.00 1.00 6905

1 1.00 1.00 1.00 5178

accuracy 1.00 12083

macro avg 1.00 1.00 1.00 12083

weighted avg 1.00 1.00 1.00 12083

----------------------Confusion Matrix Train Model After SMOTE---------

[[6905 0]

[ 0 5178]]

----------------------Test Model After SMOTE--------------------

Model performance for testing set

- Accuracy: 0.9600

- F1 score: 0.9607

-Precision: 0.8463

-Recall: 0.9192

-Roc Auc Score: 0.9435

precision recall f1-score support

0 0.98 0.97 0.98 2959

1 0.85 0.92 0.88 569

accuracy 0.96 3528

macro avg 0.92 0.94 0.93 3528

weighted avg 0.96 0.96 0.96 3528

----------------------Confusion Matrix Test Model After SMOTE----------

[[2864 95]

[ 46 523]]

AUC Curve:

AUC for the Training Data: 1.000

AUC for the Test Data: 0.944

A picture containing text, line, screenshot, plot

Description automatically generated

**Inference**:

* model has made no false positive predictions during training
* A precision score of 100 in the training set could potentially be a sign of overfitting. Overfitting occurs when a model performs extremely well on the training data but fails to generalize well to unseen data.
* model has memorized the training data instead of learning the underlying patterns. This model is not recommended
* precision score of 100 in a training model should be viewed with caution
* AUC values of 1.000 is rare in practice and may indicate potential issues such as overfitting or data leakage.

**- Effort to improve model performance.**

Random Forest Model on Balanced Data For Mobile Devices on Hypertuning

**We have did Hyper Parameter Tuning to improve the performance of the model**

**Hyper Parameter Tuning to improve model performance**

**Best Parameters for tuning RANDOM Forest Classifier.**

Fitting 3 folds for each of 100 candidates, totalling 300 fits

----------------------Best Params for RF-------------------------

{'n\_estimators': 1000, 'min\_samples\_split': 2, 'max\_features': 8, 'max\_depth': 15}

----------------------Train Model Aftre SMOTE-------------------------

RandomForestClassifier(max\_depth=15, max\_features=8, n\_estimators=1000)

Model performance for Training set

- Accuracy: 1.0000

- F1 score: 1.0000

-Precision: 1.0000

-Recall: 1.0000

-Roc Auc Score: 1.0000

precision recall f1-score support

0 1.00 1.00 1.00 6905

1 1.00 1.00 1.00 5178

accuracy 1.00 12083

macro avg 1.00 1.00 1.00 12083

weighted avg 1.00 1.00 1.00 12083

----------------------Confusion Matrix Train Model After SMOTE---------

[[6905 0]

[ 0 5178]]

----------------------Test Model After SMOTE-------------------------

Model performance for testing set

- Accuracy: 0.9889

- F1 score: 0.9889

-Precision: 0.9715

-Recall: 0.9596

-Roc Auc Score: 0.9771

precision recall f1-score support

0 0.99 0.99 0.99 2959

1 0.97 0.96 0.97 569

accuracy 0.99 3528

macro avg 0.98 0.98 0.98 3528

weighted avg 0.99 0.99 0.99 3528

----------------------Confusion Matrix Test Model After SMOTE----------

[[2943 16]

[ 23 546]]

AUC Curve:

AUC for the Training Data: 1.000

AUC for the Test Data: 0.999

A picture containing text, screenshot, line, plot

Description automatically generated

**Inference**:

* model has made no false positive predictions during training
* A precision score of 100 in the training set could potentially be a sign of overfitting. Overfitting occurs when a model performs extremely well on the training data but fails to generalize well to unseen data.
* model has memorized the training data instead of learning the underlying patterns. This model is not recommended
* precision score of 100 in a training model should be viewed with caution
* AUC values of 1.000 is rare in practice and may indicate potential issues such as overfitting or data leakage.

Random Forest Model on Balanced Data For Laptop Devices on Hyper parameter tuning

**We have did Hyper Parameter Tuning to improve the performance of the model**

**Hyper Parameter Tuning to improve model performance**

**Best Parameters for tuning RANDOM Forest Classifier.**

Fitting 3 folds for each of 100 candidates, totalling 300 fits

----------------------Best Params for RF-------------------------

{'n\_estimators': 1000, 'min\_samples\_split': 2, 'max\_features': 8, 'max\_depth': 15}

----------------------Train Model After SMOTE-----------------

Random Forest

Model performance for Training set

- Accuracy: 0.9999

- F1 score: 0.9999

-Precision: 1.0000

-Recall: 0.9998

-Roc Auc Score: 0.9999

precision recall f1-score support

0 1.00 1.00 1.00 6905

1 1.00 1.00 1.00 5178

accuracy 1.00 12083

macro avg 1.00 1.00 1.00 12083

weighted avg 1.00 1.00 1.00 12083

----------------------Confusion Matrix Train Model After SMOTE----------------

[[6905 0]

[ 1 5177]]

----------------------Test Model After SMOTE--------------------

Model performance for testing set

- Accuracy: 0.9872

- F1 score: 0.9872

-Precision: 0.9679

-Recall: 0.9525

-Roc Auc Score: 0.9732

precision recall f1-score support

0 0.99 0.99 0.99 2959

1 0.97 0.95 0.96 569

accuracy 0.99 3528

macro avg 0.98 0.97 0.98 3528

weighted avg 0.99 0.99 0.99 3528

----------------------Confusion Matrix Test Model After SMOTE----------

[[2941 18]

[ 27 542]]

**AUC Curve**

AUC for the Training Data: 1.000

AUC for the Test Data: 0.998

**A picture containing text, screenshot, line, plot

Description automatically generated**

**Inference**:

* model has made no false positive predictions during training
* A precision score of 100 in the training set could potentially be a sign of overfitting. Overfitting occurs when a model performs extremely well on the training data but fails to generalize well to unseen data.
* model has memorized the training data instead of learning the underlying patterns. This model is not recommended
* precision score of 100 in a training model should be viewed with caution
* AUC values of 1.000 is rare in practice and may indicate potential issues such as overfitting or data leakage.

Decision Tree Model on Balanced Data For Mobile Devices on Hypertuning

Decision Tree Hyper Parameter Tuning

Fitting 3 folds for each of 80 candidates, totalling 240 fits

----------------------Best Params for DT-------------------------

{'min\_samples\_split': 2, 'max\_features': 8, 'max\_depth': None}

----------------------Train Model Aftre SMOTE-------------------------

Decision Tree criterion ='gini', max\_features= 8, max\_depth= 15, random\_state=1

Model performance for Training set

- Accuracy: 0.9880

- F1 score: 0.9880

-Precision: 0.9891

-Recall: 0.9828

-Roc Auc Score: 0.9874

precision recall f1-score support

0 0.99 0.99 0.99 6905

1 0.99 0.98 0.99 5178

accuracy 0.99 12083

macro avg 0.99 0.99 0.99 12083

weighted avg 0.99 0.99 0.99 12083

**----------------------Confusion Matrix Train Model After SMOTE---------**

[[6849 56]

[ 89 5089]]

**----------------------Test Model After SMOTE-------------------------**

Model performance for testing set

- Accuracy: 0.9342

- F1 score: 0.9356

-Precision: 0.7662

-Recall: 0.8524

-Roc Auc Score: 0.9012

precision recall f1-score support

0 0.97 0.95 0.96 2959

1 0.77 0.85 0.81 569

accuracy 0.93 3528

macro avg 0.87 0.90 0.88 3528

weighted avg 0.94 0.93 0.94 3528

**----------------------Confusion Matrix Test Model After SMOTE----------**

[[2811 148]

[ 84 485]]

AUC Curve:

AUC for the Training Data: 0.999

AUC for the Test Data: 0.914

A picture containing text, line, plot, screenshot

Description automatically generated

Inference:

* For Mobile Users Precision on Training in 99.89% for and Recall is 99.11%
* Precision score of 99 in a training model is generally considered excellent
* There is no Overfitting seen in this model as the accuracy score between train and test is within the accepted limit +/-10%
* Recall:(TP) The number of correctly predicted positive instances is 485 out of 569
* The number of incorrectly predicted negative instances is 84 out of 569
* Mobile Users Recall on test data:TP/TP+FN=485/485+84=85.23%
* Mobile Users Precision on test data:TP/TP+FP=485/485+148=76.61%
* Accuracy: 93.42% for Mobile Users on test data predictions
* So far Best Model For Mobile Devices

Decision Tree Model on Balanced Data For Laptop Devices on Hypertuning

Decision Tree Hyper Parameter Tuning

Fitting 3 folds for each of 80 candidates, totalling 240 fits

----------------------Best Params for DT-------------------------

criterion ='gini', max\_features= 7, max\_depth= 15, random\_state=1

----------------------Train Model After SMOTE-----------------

Decision Tree

Model performance for Training set

- Accuracy: 0.9815

- F1 score: 0.9814

-Precision: 0.9878

-Recall: 0.9687

-Roc Auc Score: 0.9799

precision recall f1-score support

0 0.98 0.99 0.98 6905

1 0.99 0.97 0.98 5178

accuracy 0.98 12083

macro avg 0.98 0.98 0.98 12083

weighted avg 0.98 0.98 0.98 12083

----------------------Confusion Matrix Train Model After SMOTE----------------

[[6843 62]

[ 162 5016]]

----------------------Test Model After SMOTE--------------------

Model performance for testing set

- Accuracy: 0.9226

- F1 score: 0.9248

-Precision: 0.7249

-Recall: 0.8383

-Roc Auc Score: 0.8886

precision recall f1-score support

0 0.97 0.94 0.95 2959

1 0.72 0.84 0.78 569

accuracy 0.92 3528

macro avg 0.85 0.89 0.87 3528

weighted avg 0.93 0.92 0.92 3528

----------------------Confusion Matrix Test Model After SMOTE----------

[[2778 181]

[ 92 477]]

AUC Curve:

AUC for the Training Data: 0.999

AUC for the Test Data: 0.905

A picture containing text, line, plot, screenshot

Description automatically generated

Inference:

* For Laptop Users Precision on Training in 99.87% for and Recall is 96.87%
* Precision score of 98.78% in a training model is generally considered excellent
* There is no Overfitting seen in this model as the accuracy score between train and test is within the accepted limit +/-10%
* Recall:(TP) The number of correctly predicted positive instances is 477 out of 569
* The number of incorrectly predicted negative instances is 181 out of 569
* Laptop Users Recall on Test Data: TP/TP+FN=477/477+92=83.83%
* Laptop Users Precision Test data: TP/TP+FP=477/477+181=72.49%
* Accuracy: 99% for Laptop Users on Test Predictions
* So far Best Model For Laptop Devices

**5. Model validation**

- How was the model validated? Just accuracy, or anything else too?

Inference on Best Model:

For predicting **Taken Product** used (Label 1 ):

**Mobile Devices**

**Precision (77%) – 77% of Users predicted are actually taken product out of all Mobile Users who are using Mobile devices.**

**Recall (85%) –For mobile Users, Number of people who actually have taken product has been identified as not taken product correctly is 85%**

**Accuracy (93%) – Decision Tree Model is 86% accurate for Users who are using Mobile devices**

**Laptop Devices:**

**Precision (72%) – 72% of Users predicted are actually taken product out of all Laptop Users who are using Laptop devices.**

**Recall (84%) –For Laptop Users, Number of people who actually have taken product has been identified as not taken product correctly is 84%**

**Accuracy (92%) – Decision Tree Model is 92% accurate for Users who are using Laptop devices**

Train and test performance is within the accepted limit +/-10%

(Difference in accuracy of Train and Test)

Model running information is good Model Running Information and noise is bad

**Variable Importance found after Regularizing Decision Tree for the users who have Taken product in the past and those use laptop/Mobile for booking.**

|  |  |
| --- | --- |
| **Feature importance for Mobile Users** | **Comments** |
| total\_likes\_on\_outofstation\_checkin\_received 0.195485 | This Feature is of High Importance to conisder the Customer will get the product |
| total\_likes\_on\_outstation\_checkin\_given 0.127739 | This Feature is of High Importance to conisder the Customer will get the product |
| Yearly\_avg\_view\_on\_travel\_page 0.101221 | This Feature is of High Importance to conisder the Customer will get the product |
| yearly\_avg\_Outstation\_checkins 0.082618 | This Feature is of High Importance to conisder the Customer will get the product |
|  |  |
|  |  |
| **Feature importance for Laptop Users** | **Comments** |
| total\_likes\_on\_outofstation\_checkin\_received 0.189112 | This Feature is of High Importance to conisder the Customer will get the product |
| total\_likes\_on\_outstation\_checkin\_given 0.120158 | This Feature is of High Importance to conisder the Customer will get the product |
| Yearly\_avg\_view\_on\_travel\_page 0.093885 | This Feature is of High Importance to conisder the Customer will get the product |
| following\_company\_page\_Yes 0.078319 | This Feature is of High Importance to conisder the Customer will get the product |

**6. Final interpretation / recommendation**

* **Considering the Training model with 99.4% Precision for Mobile and 98.26% on laptop. This model has maximized True positives with very minimal FP for the balanced data set .we can recommend to use the Decision Tree Model with Hyper Tuning parameters gives most accurate results.**
* ***Total Likes on outstation Check in received*, *Total Like on Outstation check in given*, *Yearly average view on travel page*, *Following company page* these 4 features would be of High importance and are most like to buy the product who has booked using laptop if recommended by Advertisements and campaign**
* ***Total Likes on outstation Check in received*, *Total Like on Outstation check in given*, *Yearly average view on travel page*, *Yearly out station check-ins* these 4 features would be of High importance and are most like to buy the product who has booked using mobile if recommended by Advertisements and campaign**

Interpretation of the most optimum model and its implication on the business

**Out of all the models we found that Decision Tree with Hyper Parameter Tuning model has high precision when compared with other models and this model is not overfitting with good Accuracy, Precision and recall for both Mobile and laptop devices for Class of interest of Taken Product “Yes”**

**In marketing kind of Problem whether the User will buy the product or not we take Precision value most considered for predictions**