**Project - Data Mining**

**Problem 1: Clustering**

A leading bank wants to develop a customer segmentation to give promotional offers to its customers. They collected a sample that summarizes the activities of users during the past few months. You are given the task to identify the segments based on credit card usage.

Solutions-

* 1. **Read the data, do the necessary initial steps, and exploratory data analysis (Univariate, Bi-variate, and multivariate analysis).**

Solution-

* Shape of the data-After reading the data we can see that there are 210 rows and 7 columns.
* Data Information-

Data columns (total 7 columns):

# Column Non-Null Count Dtype

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

0 spending 210 non-null float64

1 advance\_payments 210 non-null float64

2 probability\_of\_full\_payment 210 non-null float64

3 current\_balance 210 non-null float64

4 credit\_limit 210 non-null float64

5 min\_payment\_amt 210 non-null float64

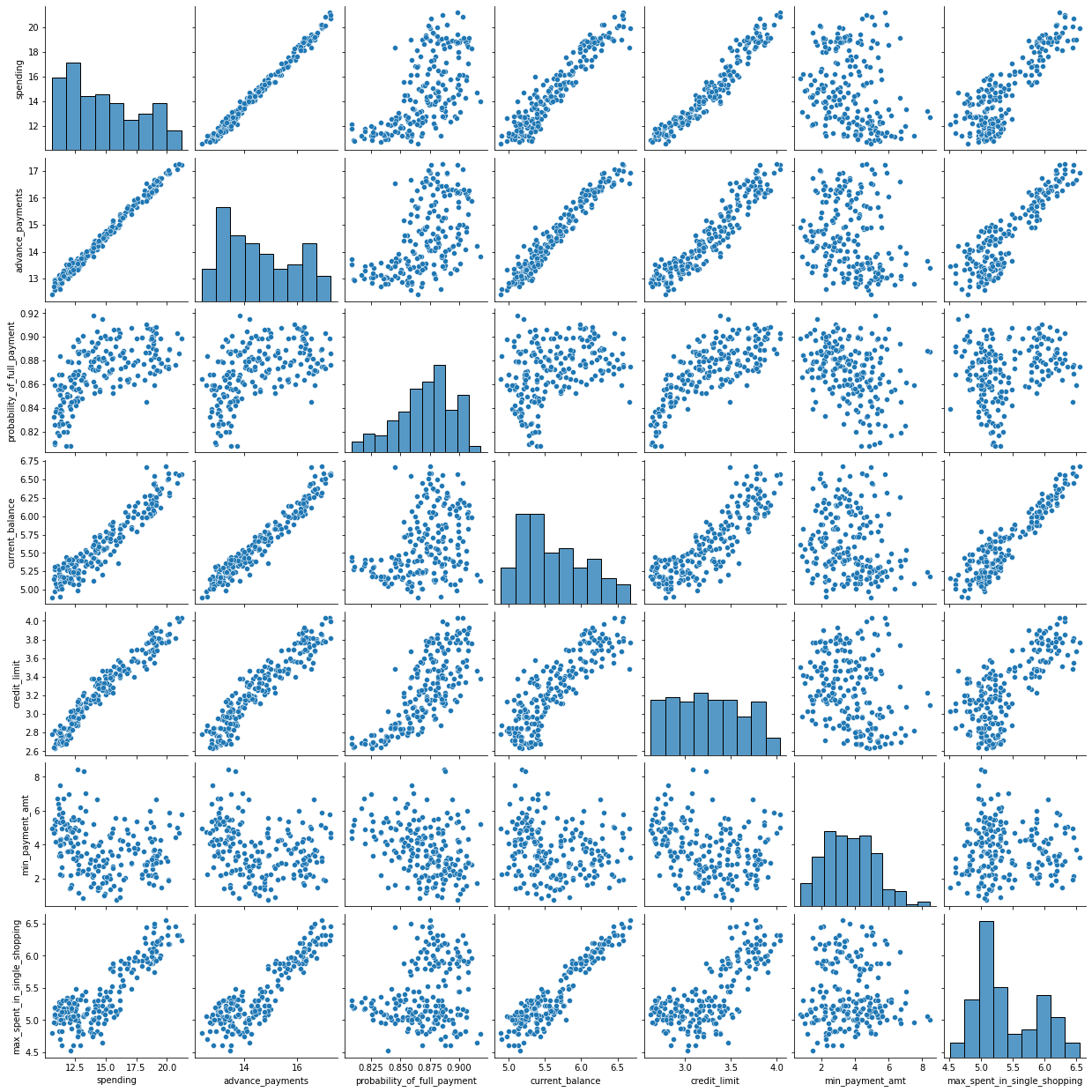
6 max\_spent\_in\_single\_shopping 210 non-null float64

dtypes: float64(7)

* Data Describtion-

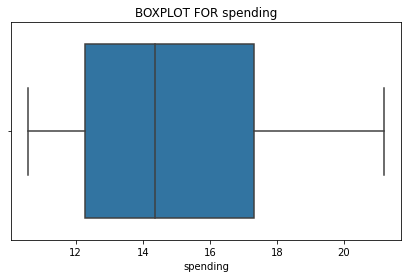
| **count** | **mean** | **std** | **min** | **25%** | **50%** | **75%** | **max** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **spending** | 210.0 | 14.847524 | 2.909699 | 10.5900 | 12.27000 | 14.35500 | 17.305000 | 21.1800 |
| **advance\_payments** | 210.0 | 14.559286 | 1.305959 | 12.4100 | 13.45000 | 14.32000 | 15.715000 | 17.2500 |
| **probability\_of\_full\_payment** | 210.0 | 0.870999 | 0.023629 | 0.8081 | 0.85690 | 0.87345 | 0.887775 | 0.9183 |
| **current\_balance** | 210.0 | 5.628533 | 0.443063 | 4.8990 | 5.26225 | 5.52350 | 5.979750 | 6.6750 |
| **credit\_limit** | 210.0 | 3.258605 | 0.377714 | 2.6300 | 2.94400 | 3.23700 | 3.561750 | 4.0330 |
| **min\_payment\_amt** | 210.0 | 3.700201 | 1.503557 | 0.7651 | 2.56150 | 3.59900 | 4.768750 | 8.4560 |
| **max\_spent\_in\_single\_shopping** | 210.0 | 5.408071 | 0.491480 | 4.5190 | 5.04500 | 5.22300 | 5.877000 | 6.5500 |

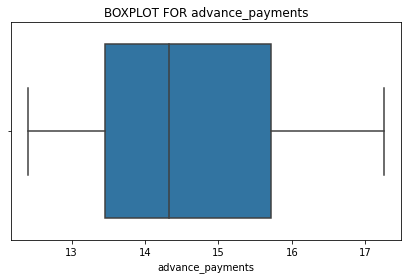
* Pairplot Analysis of data set-

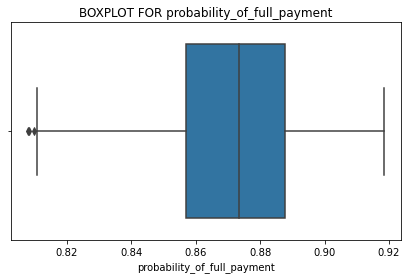


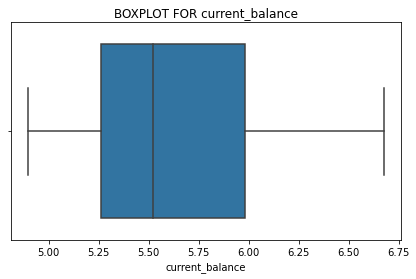
* Correlation Matrix analysis for data set-

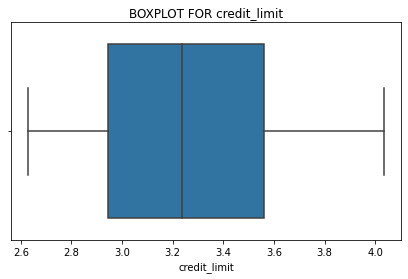
****

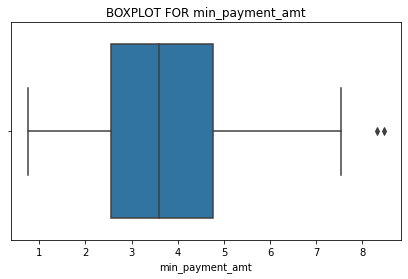
****

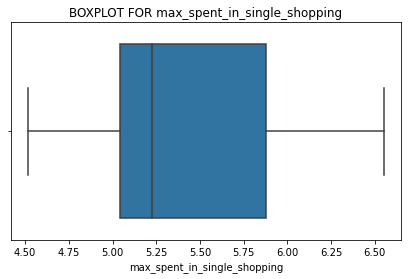
****

****

****

****

****

****

* 1. **Do you think scaling is necessary for clustering in this case? Justify**

**Solution-**

Yes scaling is important in data set, although most of the variables mean value is around 5-6 but to variables spending,advance\_payments has a mean score between 14-15 which tells us its better to scale the data set so that all the variables gets the equal weightage.

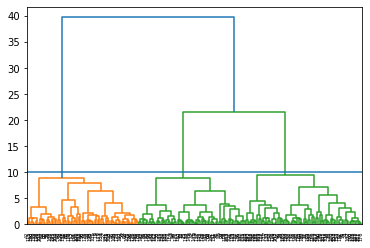
* 1. **Apply hierarchical clustering to scaled data. Identify the number of optimum clusters using Dendrogram and briefly describe them?**

Solution-

Here I have performed hierarchical clustering using wards method.

Dendogram-

We can infer that I have chosen 3 clusters as we can infer from dendo gram 3 cluster can be the optimal in our case.



While labeling the cluster I have used Max cluster critera.

| **clusters** | **spending** | **advance\_payments** | **probability\_of\_full\_payment** | **current\_balance** | **credit\_limit** | **min\_payment\_amt** | **max\_spent\_in\_single\_shopping** | **Frequency** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 1 | 18.371429 | 16.145429 | 0.884400 | 6.158171 | 3.684629 | 3.639157 | 6.017371 | 70 |
| **1** | 2 | 11.872388 | 13.257015 | 0.848072 | 5.238940 | 2.848537 | 4.949433 | 5.122209 | 67 |
| **2** | 3 | 14.199041 | 14.233562 | 0.879190 | 5.478233 | 3.226452 | 2.612181 | 5.086178 | 73 |

In [71]:



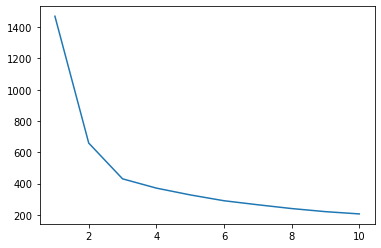
* 1. **Apply K-Means clustering on scaled data and determine optimum clusters. Apply elbow curve. Explain the results properly. Interpret and write inferences on the finalized clusters.**

Solution-

Here also we will choose 3 clusters as optimam cluster.

K\_means.inertia\_ value will be 430.65

Elbow curve-



We can infer from above elbow curve that to coose 3 clusters is a good decision as curve get flatted after it reaches to 3 clusters.

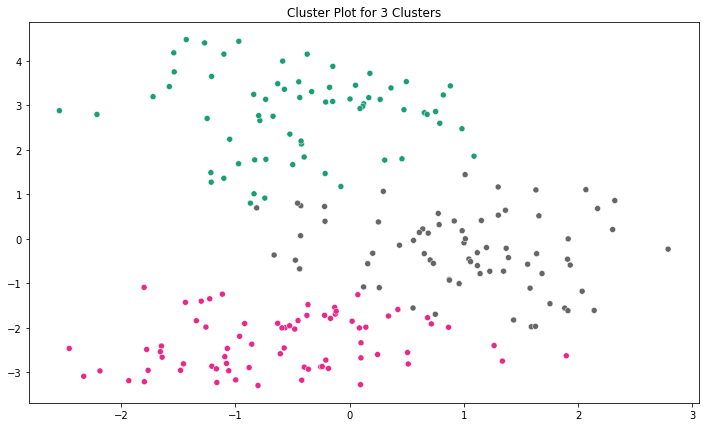
Cluster profiling with K means method-

| **Clus\_kmeans** | **spending** | **advance\_payments** | **probability\_of\_full\_payment** | **current\_balance** | **credit\_limit** | **min\_payment\_amt** | **max\_spent\_in\_single\_shopping** | **clusters** | **Kmeans\_clusters** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 0 | 11.856944 | 13.247778 | 0.848253 | 5.231750 | 2.849542 | 4.742389 | 5.101722 | 2.083333 | 2.083333 |
| **1** | 1 | 18.495373 | 16.203433 | 0.884210 | 6.175687 | 3.697537 | 3.632373 | 6.041701 | 1.029851 | 1.029851 |
| **2** | 2 | 14.437887 | 14.337746 | 0.881597 | 5.514577 | 3.259225 | 2.707341 | 5.120803 | 2.873239 | 2.873239 |

In [97]:



1. The graphical presentation of 3 clusters with scatter plot-



**Silhouette score(DF\_Kmeans,clust\_3)-**

**Silhouette score** of the data set is 0.50 which is fine to infer results from clustring.

**1.5 Describe cluster profiles for the clusters defined. Recommend different promotional strategies for different clusters.**

Solution-

Hierarchical clustering

| **clusters** | **spending** | **advance\_payments** | **probability\_of\_full\_payment** | **current\_balance** | **credit\_limit** | **min\_payment\_amt** | **max\_spent\_in\_single\_shopping** | **Frequency** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 1 | 18.371429 | 16.145429 | 0.884400 | 6.158171 | 3.684629 | 3.639157 | 6.017371 | 70 |
| **1** | 2 | 11.872388 | 13.257015 | 0.848072 | 5.238940 | 2.848537 | 4.949433 | 5.122209 | 67 |
| **2** | 3 | 14.199041 | 14.233562 | 0.879190 | 5.478233 | 3.226452 | 2.612181 | 5.086178 | 73 |

Cluster profiling with K means method-

.

.

| **Clus\_kmeans** | **spending** | **advance\_payments** | **probability\_of\_full\_payment** | **current\_balance** | **credit\_limit** | **min\_payment\_amt** | **max\_spent\_in\_single\_shopping** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 0 | 11.856944 | 13.247778 | 0.848253 | 5.231750 | 2.849542 | 4.742389 | 5.101722 |
| **1** | 1 | 18.495373 | 16.203433 | 0.884210 | 6.175687 | 3.697537 | 3.632373 | 6.041701 |
| **2** | 2 | 14.437887 | 14.337746 | 0.881597 | 5.514577 | 3.259225 | 2.707341 | 5.120803 |

In [ ]:



​

**Conclusion-**

As we can infer from hierarchical clustering and K means clustering both the methods are showing similar result and we are getting 3 clusters-

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 18.371429 | 16.145429 | 0.884400 | 6.158171 | 3.684629 | 3.639157 | 6.017371 |
| 2 | 11.872388 | 13.257015 | 0.848072 | 5.238940 | 2.848537 | 4.949433 | 5.122209 |
| 3 | 14.199041 | 14.233562 | 0.879190 | 5.478233 | 3.226452 | 2.612181 | 5.086178 |

**Cluster 1-** Elite customer- Highest spending, high **probability\_of\_full\_payment, high balances, min payment value is high, Maximum spend in one shopping is also highest.**

**Cluster2-** Good Customers- In All the above parameter’s they are in Mid level

**Cluster3-** Average Customers- In All the above parameter’s they are lower in comparison to other two cluster customers.

**Promotional strategies-**

**Cluster 1-** For cluster one client we can see that their purchasing capacity is good as well they have good payback history so there is good cross selling opportunity with them, and based on their record we can give a credit card limit enhancement as well us upgrade opportunity on higher variant card.

**Cluster 2-** We can see that although their advance payment and probability of payment is good but their max spent is low so we can give opportunity to enhance their credit limit, also there is small concern for low minimum payment in this cluster of customers.

**Cluster 3-** For cluster three card holders we can infer that the average of minimum payment amount and

Their max spent in single shopping is high so we can infer that the potential is good for these card holders.We just have to increase the spending for this cluster for that we can use some sales promotion techniques in which these card holders will get the cash back on multiple websites and stores.,

**Problem 2: CART-RF-ANN**

* 1. **Read the data, do the necessary initial steps, and exploratory data analysis (Univariate, Bi-variate, and multivariate analysis).**

Solution-

df.shape- (3000, 10)

Information-

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 3000 entries, 0 to 2999

Data columns (total 10 columns):

# Column Non-Null Count Dtype

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

0 Age 3000 non-null int64

1 Agency\_Code 3000 non-null object

2 Type 3000 non-null object

3 Claimed 3000 non-null object

4 Commision 3000 non-null float64

5 Channel 3000 non-null object

6 Duration 3000 non-null int64

7 Sales 3000 non-null float64

8 Product Name 3000 non-null object

9 Destination 3000 non-null object

dtypes: float64(2), int64(2), object(6)

memory usage: 234.5+ KB

Describe-

|  | **count** | **mean** | **std** | **min** | **25%** | **50%** | **75%** | **max** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Age** | 3000.0 | 38.091000 | 10.463518 | 8.0 | 32.0 | 36.00 | 42.000 | 84.00 |
| **Commision** | 3000.0 | 14.529203 | 25.481455 | 0.0 | 0.0 | 4.63 | 17.235 | 210.21 |
| **Duration** | 3000.0 | 70.001333 | 134.053313 | -1.0 | 11.0 | 26.50 | 63.000 | 4580.00 |
| **Sales** | 3000.0 | 60.249913 | 70.733954 | 0.0 | 20.0 | 33.00 | 69.000 | 539.00 |

Information for the data set-

RangeIndex: 3000 entries, 0 to 2999

Data columns (total 10 columns):

# Column Non-Null Count Dtype

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

0 Age 3000 non-null int64

1 Agency\_Code 3000 non-null object

2 Type 3000 non-null object

3 Claimed 3000 non-null object

4 Commision 3000 non-null float64

5 Channel 3000 non-null object

6 Duration 3000 non-null int64

7 Sales 3000 non-null float64

8 Product Name 3000 non-null object

9 Destination 3000 non-null object

dtypes: float64(2), int64(2), object(6)

memory usage: 234.5+ KB

Distribution of Agency code-

EPX 1365

C2B 924

CWT 472

JZI 239

Distribution of Type-

Travel Agency 1837

Airlines 1163

Distribution of Claimed-

No 2076

Yes 924

In percentage value-

No 0.692

Yes 0.308

Distribution of channel-

Online 2954

Offline 46

In percentage value-

Online 0.984667

Offline 0.015333

Distribution of product name-

Customised Plan 1136

Cancellation Plan 678

Bronze Plan 650

Silver Plan 427

Gold Plan 109

Null values in data set-

Age 0

Agency\_Code 0

Type 0

Claimed 0

Commision 0

Channel 0

Duration 0

Sales 0

Product Name 0

Destination 0

* 1. **Data Split: Split the data into test and train, build classification model CART, Random Forest**

Solution-

Classification Model Cart-

|  | **Age** | **Type** | **Claimed** | **Commision** | **Channel** | **Duration** | **Sales** | **Product Name** | **Destination** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 48 | Airlines | No | 0.70 | Online | 7 | 2.51 | Customised Plan | ASIA |
| **1** | 36 | Travel Agency | No | 0.00 | Online | 34 | 20.00 | Customised Plan | ASIA |
| **2** | 39 | Travel Agency | No | 5.94 | Online | 3 | 9.90 | Customised Plan | Americas |
| **3** | 36 | Travel Agency | No | 0.00 | Online | 4 | 26.00 | Cancellation Plan | ASIA |
| **4** | 33 | Airlines | No | 6.30 | Online | 53 | 18.00 | Bronze Plan | ASIA |

As we can infer from above table there are dtypes: float64(2), int64(2), object(6)

So to fit this data type into Model we have to transform 6 Object type data into float or intiger.

I have used here replace and pd.categorical method to transform the data.

New transformed data set info-

# Column Non-Null Count Dtype

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

0 Age 3000 non-null int64

1 Type 3000 non-null int8

2 Claimed 3000 non-null int64

3 Commision 3000 non-null float64

4 Channel 3000 non-null int64

5 Duration 3000 non-null int64

6 Sales 3000 non-null float64

7 Product Name 3000 non-null int8

8 Destination 3000 non-null int8

dtypes: float64(2), int64(4), int8(3)

Now data set is ready to fit into modal-

Spliting the data Into train & Test data-

I have used the split ratio of 70:30 for train & Test data set.

Accuracy score for Train Data after fitting the data into model is- 99.57%

Accuracy score for Test Data after fiting the data into model is- 69.55%

**2.3 Performance Metrics: Comment and Check the performance of Predictions on Train and Test sets using Accuracy, Confusion Matrix, Plot ROC curve and get ROC\_AUC score, classification reports for each model.**

**confusion matrix(y\_test,test\_predictions)-**

array([[484, 139],

[135, 142]], dtype=int64)

**Precesion, Recall and F1 score for the model-**

precision recall f1-score support

0 0.78 0.78 0.78 623

1 0.51 0.51 0.51 277

accuracy 0.70 900

macro avg 0.64 0.64 0.64 900

weighted avg 0.70 0.70 0.70 900

We can infer from above result that modal is overfitted now we have to do Pruning & Regularization Model-

For decision tree classifier we will take following criteria-

criterion='gini', random\_state=1, max\_depth= 10 ,min\_samples\_split=75, min\_samples\_leaf=60

Modal Score for train data set- 0.792

Modal Score for test data set- 0.76

Lets try to find more optimum score by using grid search-

parameters = {'max\_depth' : [2,5,7,9,11,13,15,17,20],

'min\_samples\_split' : [50,100,200,300],

'min\_samples\_leaf' : [50,100,200,400],

'criterion' : ['gini', 'entropy']}

We get to find the best combination-

'criterion': 'gini',

'max\_depth': 7,

'min\_samples\_leaf': 50,

'min\_samples\_split': 50

Modal Score for train data set- 0.76

Modal Score for test data set- 0.79

Let find the classification matrix-

precision recall f1-score support

0 0.81 0.85 0.83 623

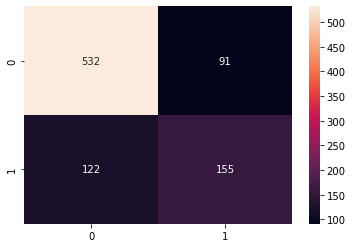
1 0.63 0.56 0.59 277

accuracy 0.76 900

macro avg 0.72 0.71 0.71 900

weighted avg 0.76 0.76 0.76 900

Graphical representation of confusion matrix by heat map-



Random Forest Classifier-

Parameter taken for Random forest model-

n\_estimators=100,

criterion='gini',

oob\_score=True,

max\_samples=0.9,

max\_features=8,

max\_depth=10)

Modal Score for train data set- 0.91

Modal Score for test data set- 0.74

Out of bag score-

0.77

Lets find the importance of each variable in the model-

Product Name 0.302790

Sales 0.191459

Duration 0.190590

Age 0.162360

Commision 0.123800

Destination 0.015994

Type 0.008548

Channel 0.004461

dtype: float64

We can infer from above result that modal is overfitted now we have to do Pruning & Regularization Model with the help of Grid Search-

Parameter taken for Random forest model-

'max\_depth': [7, 15],

'max\_features': [4, 6],

'min\_samples\_leaf': [50, 100, 150, 200],

'min\_samples\_split': [150, 200, 250],

'n\_estimators': [101, 201, 301]

}

Lets find the best Paramiteters from Grid search-

'max\_depth': 15,

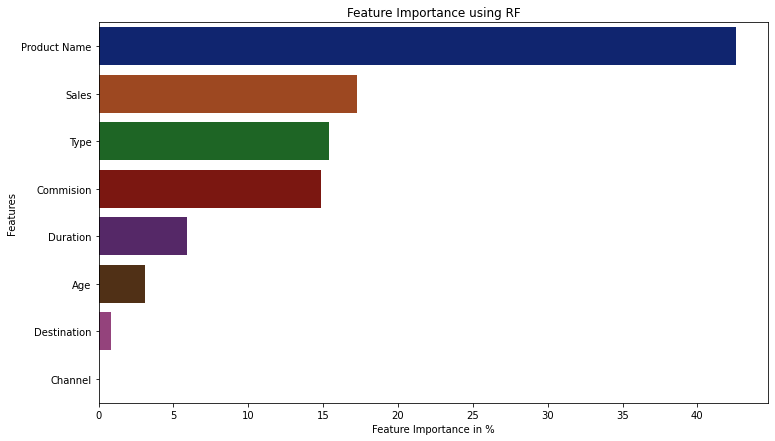
'max\_features': 4,

'min\_samples\_leaf': 50,

'min\_samples\_split': 150,

'n\_estimators': 201}

Lets make the graphical representation of feature importance-



Confusion matrix for train data-

array([[1330, 123],

[ 327, 320]]

Confusion matrix for test data-

array([[558, 65],

[167, 110]]

classification\_report for train data-

precision recall f1-score support

0 0.80 0.92 0.86 1453

1 0.72 0.49 0.59 647

accuracy 0.79 2100

macro avg 0.76 0.70 0.72 2100

weighted avg 0.78 0.79 0.77 2100

classification report for test data-

precision recall f1-score support

0 0.77 0.90 0.83 623

1 0.63 0.40 0.49 277

accuracy 0.74 900

macro avg 0.70 0.65 0.66 900

weighted avg 0.73 0.74 0.72 900

* 1. **Final Model: Compare all the models and write an inference which model is best/optimized.**

Solution-



From the above table Model 2 turns out to be the most effective one among the three.



Although due to class imbalance in the data for Once score is pretty low although model is doing preety good to predict the 0’s for the variable K.

* 1. **Inference: Based on the whole Analysis, what are the business insights and recommendations.**

**Solution-**

We can infer from the above analysis that to predict for the clients who are not going to claim model is pretty effective but for to forecast about the claim model is not reliable.