

|  |  |
| --- | --- |
| Data Mining | |
|  |  |
|  |  |

***CLUSTERING***

**PROBLEM 1**: 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.

**1.1** Read the data and do exploratory data analysis. Describe the data briefly.

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

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

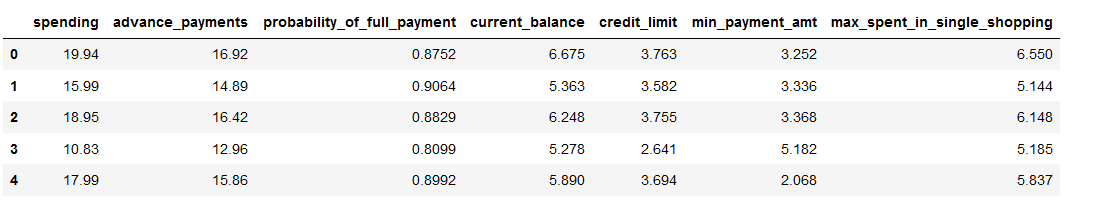
**1.4** Apply K-Means clustering on scaled data and determine optimum clusters. Apply elbow curve and silhouette score.

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

**1.1** Read the data and do exploratory data analysis. Describe the data briefly.

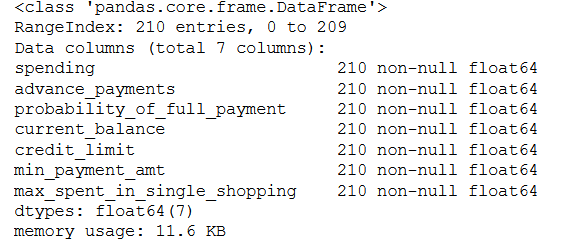
**Viewing the dataset**

* data.head()



**Information of dataset**

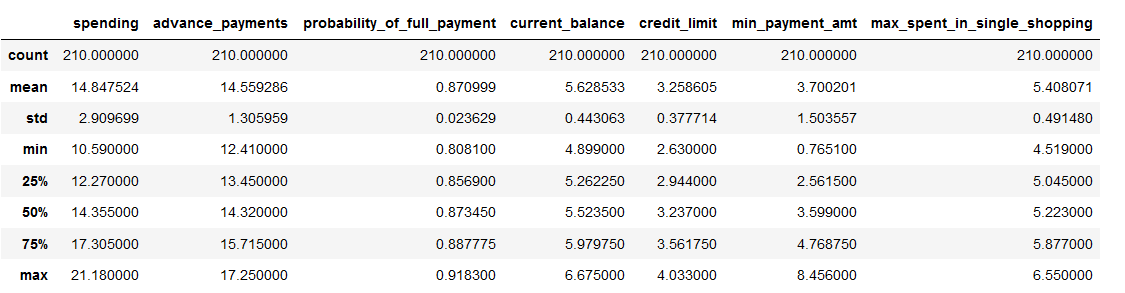
* data.info()



All values are float datatype

**Describe the dataset**

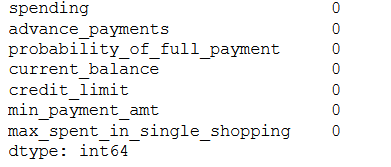
* data.describe(include="all")



The dataset is not scaled and mean is highest in spending and minimum in probability\_of\_full\_payment

**Checking null values**

* data.isnull().sum()



There are no null/missing values present in the dataset

**Checking Duplicates**

* dups = data.duplicated()
* print('Number of duplicate rows = %d' % (dups.sum()))



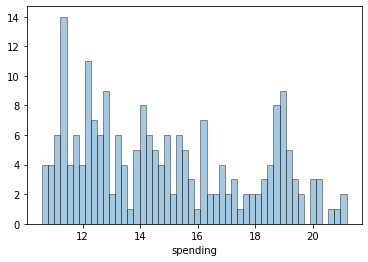
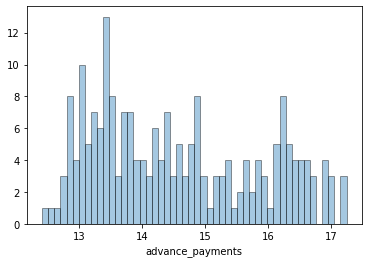
There are no duplicates values present in the dataset

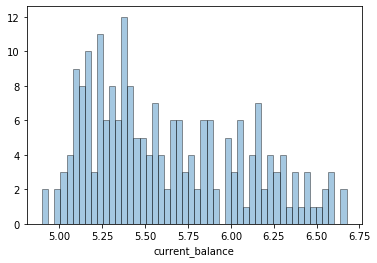
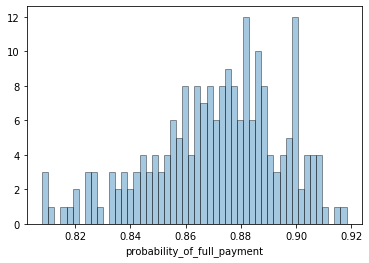
Shape of the dataset

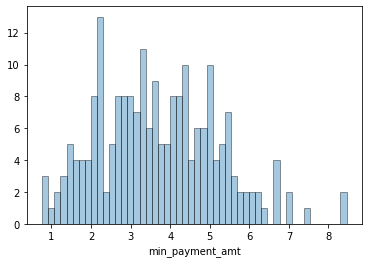
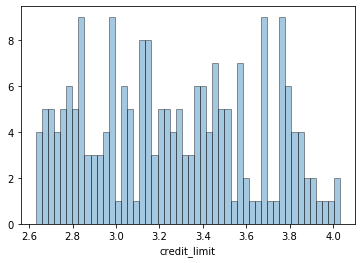
(210, 7)

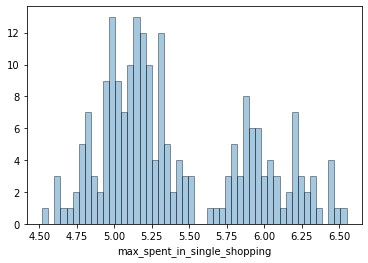
There are 7 columns and 210 rows in the dataset.

**Univariate Analysis**

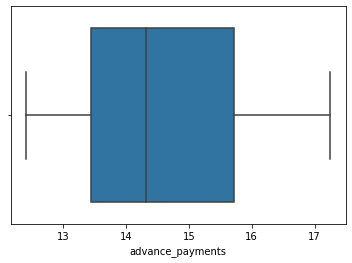
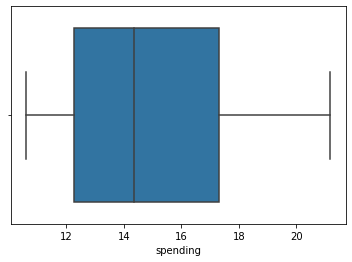
*** ***

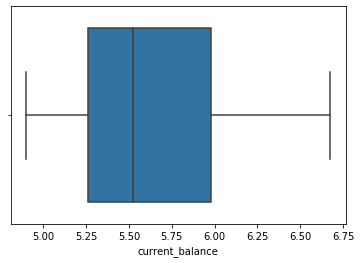
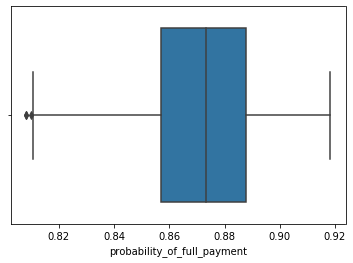
******

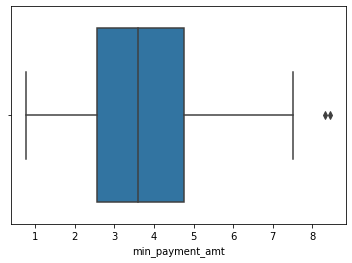
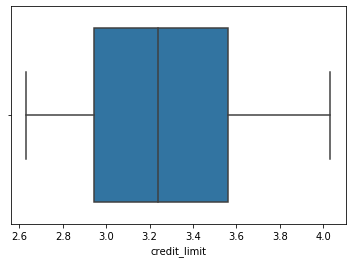
******

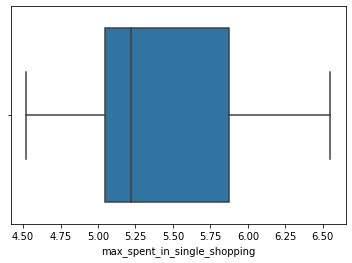
******

***BoxPlot***

******

******

******

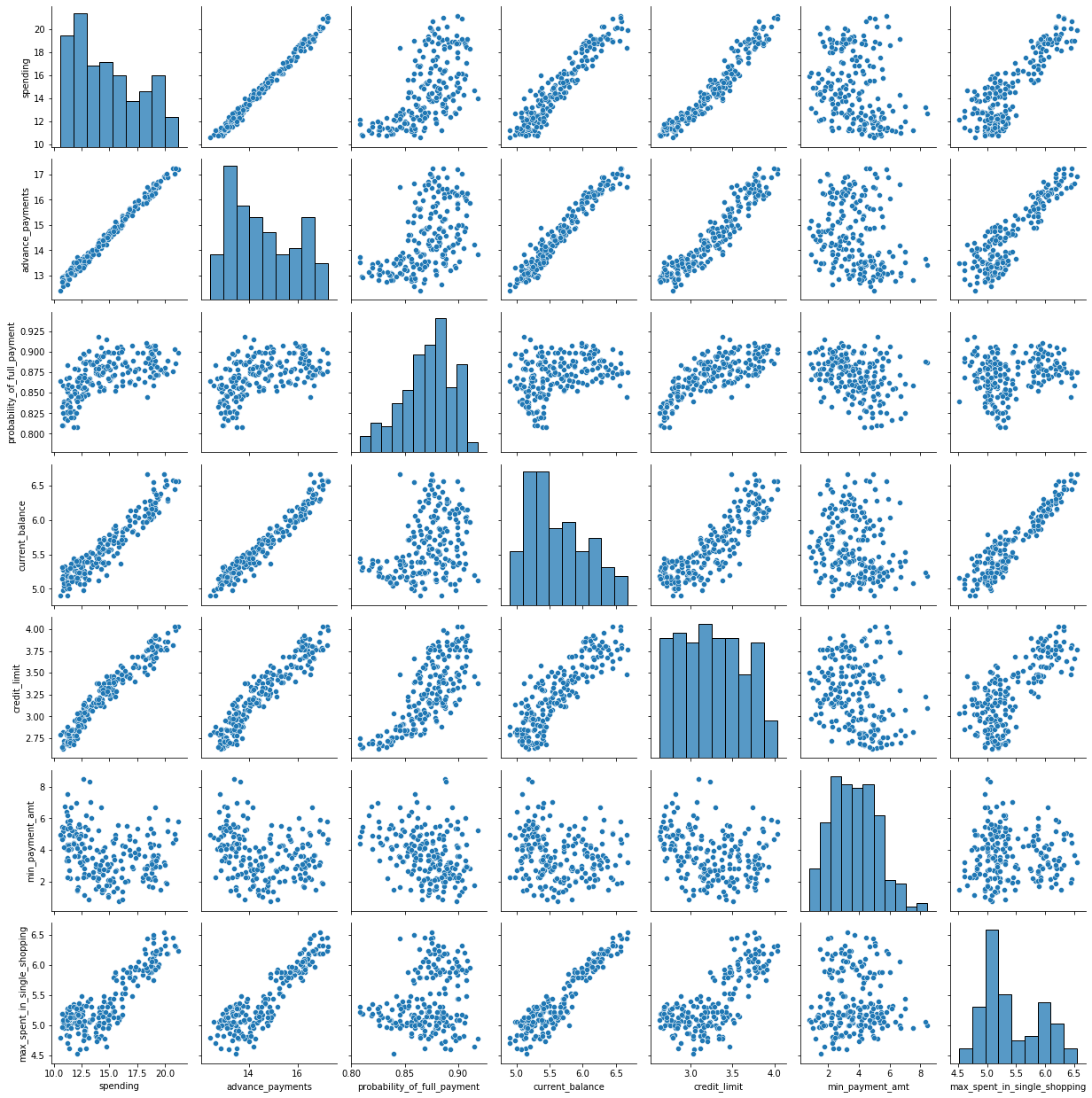
******

There are a few outliers are present i.e, **min\_payment\_amt** which means that there are only a few customers whose minimum payment amount falls on the higher side on an average. Since only one of the seven variable have a very small outlier value, hence there is no need to treat the outliers. This small value will not create any difference in our analysis.

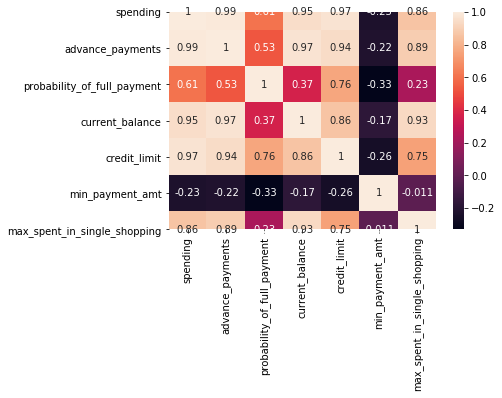
We can conclude from the above graphs that the most of the customers in our data have a higher spending capacity, high current balance in their accounts and these customers spent a higher amount during a single shopping event. Majority of the customers have a higher probability to make full payment to the bank.

**Multivariate Analysis**

**Pair Plot**



**Heatmap**

****

There is high correlation between spending-advance\_payment , spending – current\_balance, advance\_payment-current balance, advance\_payment-credit\_limit and Current\_balance-max\_spent.

The customers who are spending very high have a higher current balance and high credit limit. Advance payments and maximum expenditure done in single shopping are done by majority of those customers who have high current balance in their bank accounts.

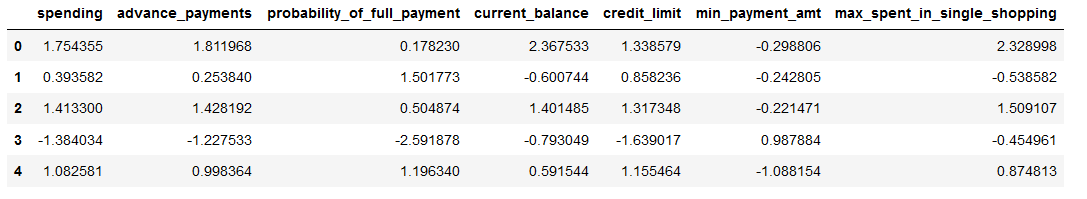
Probability of full payments are higher for those customers who have a higher credit limit.

Minimum payment amount is not correlated to any of the variables, hence, it is not affected by any changes in the current balance or credit limit of the customers.

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

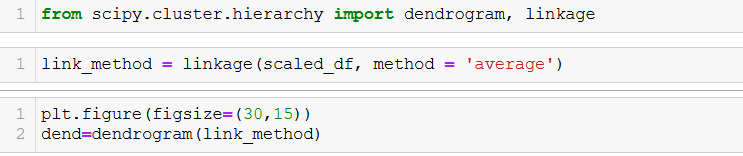
Yes, scaling is required for standardization of values to pre-process the data. It helps to normalize the data in particular range. Some amounts are in 100s, some in 1000s etc., Further, the clustering algorithms use distance measure, it is important that all variables are to the same scale.

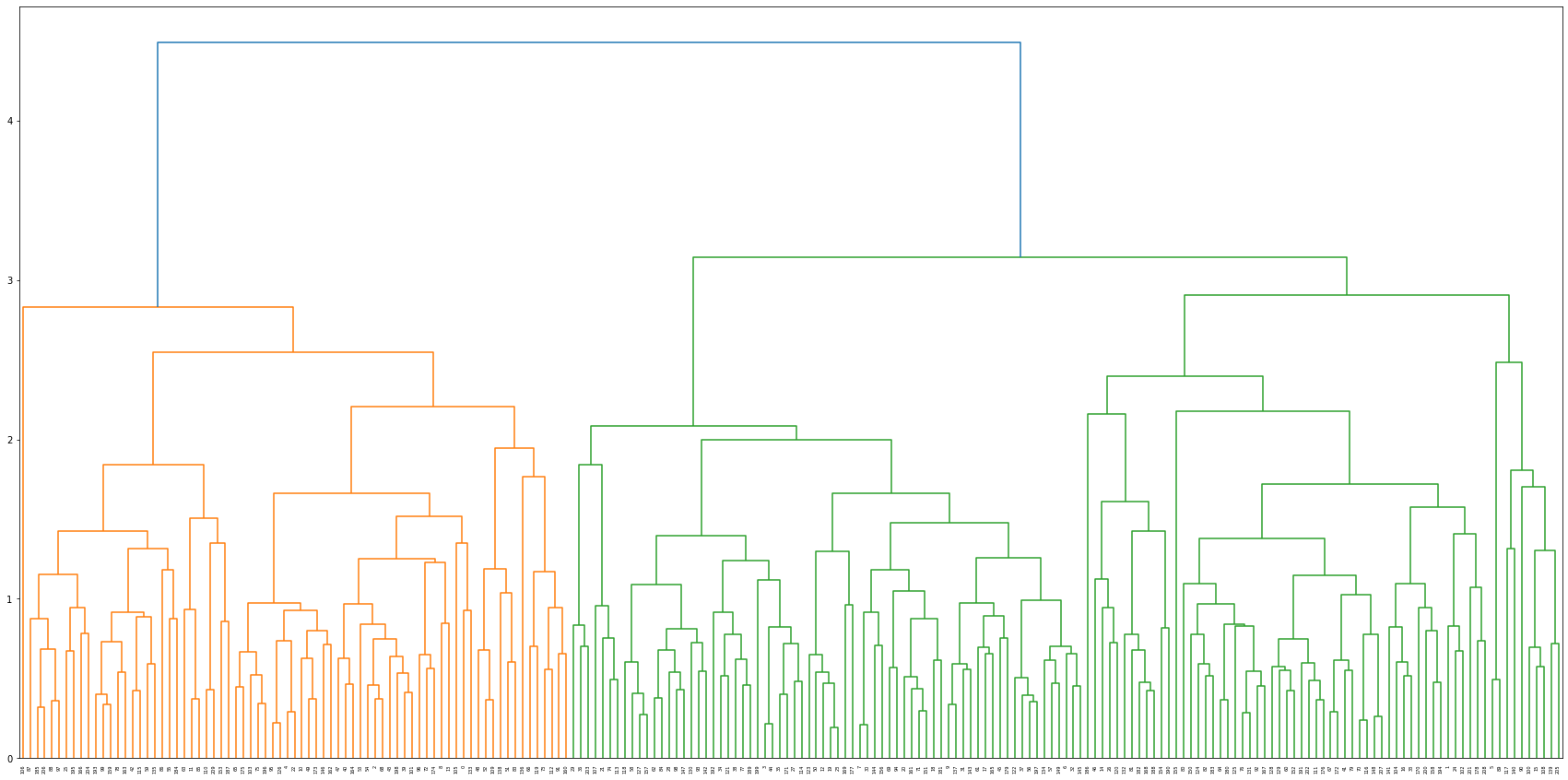
* from sklearn.preprocessing import StandardScaler
* X = StandardScaler()
* scaled\_df = pd.DataFrame(X.fit\_transform(data),columns=data.columns)
* scaled\_df.head()



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

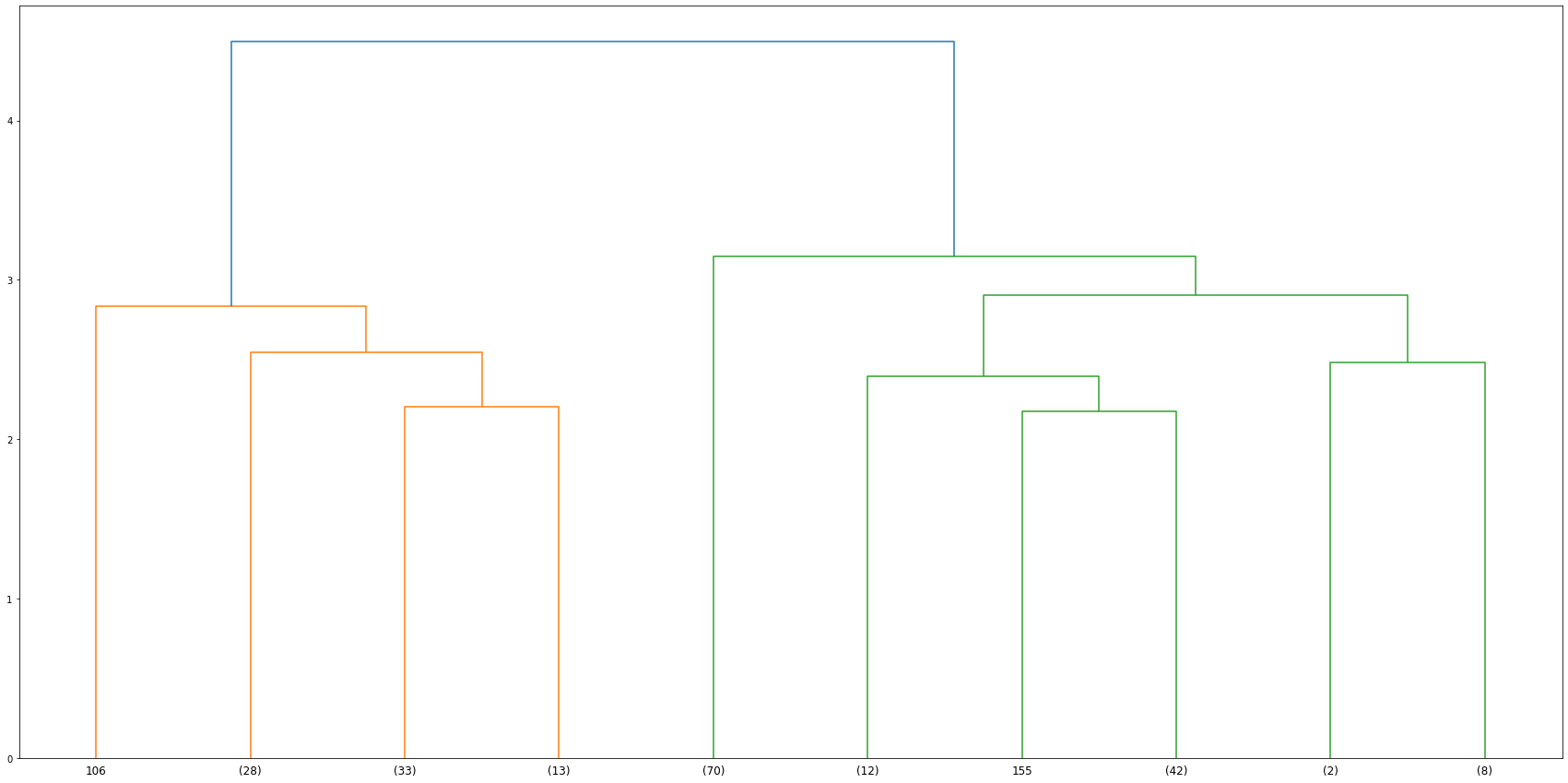
**Average Method**



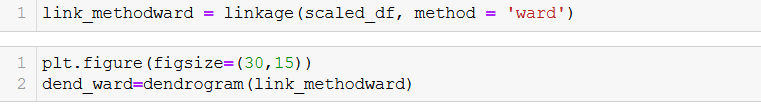


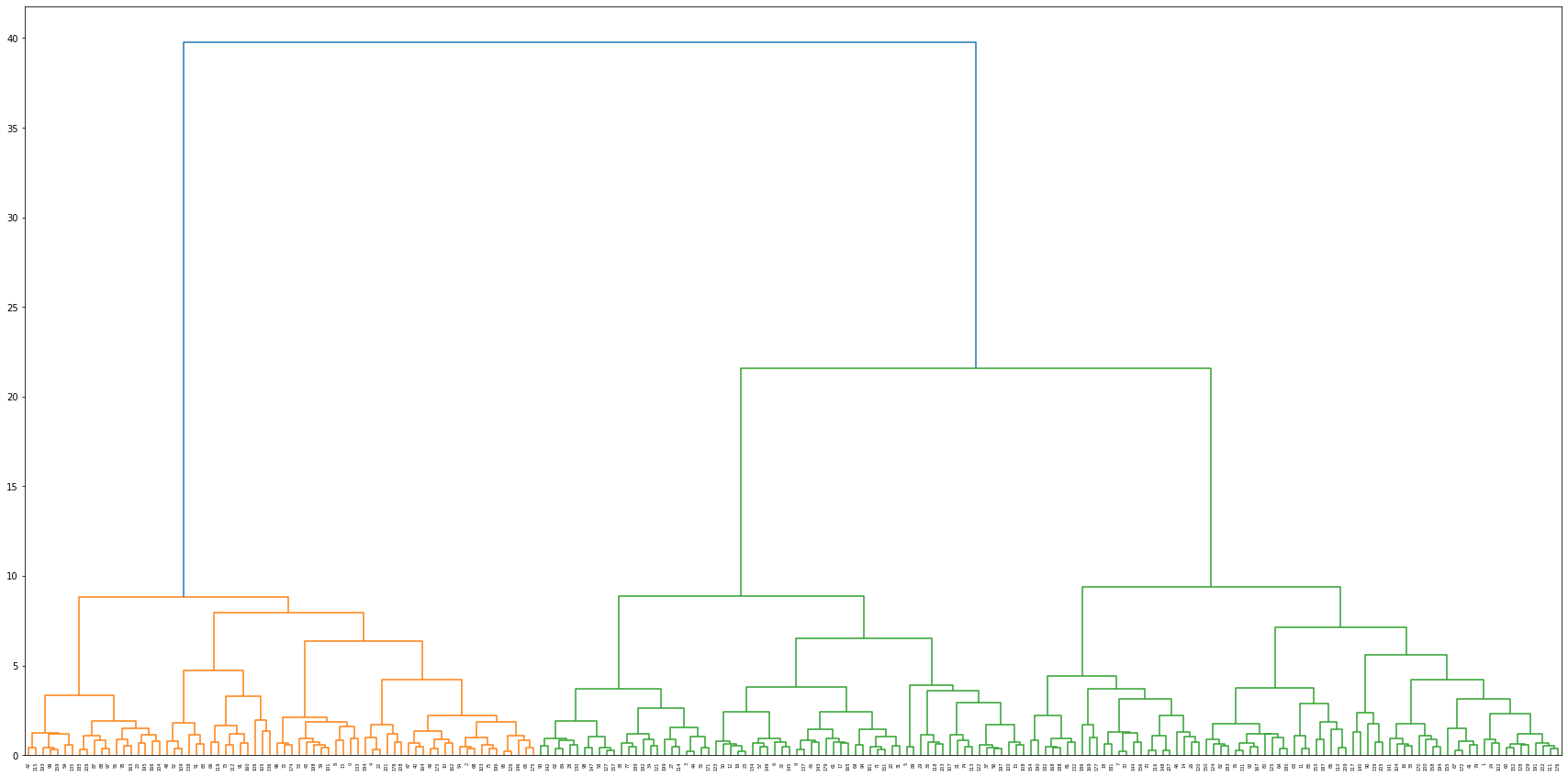
plt.figure(figsize=(30,15))

dend=dendrogram(link\_method,truncate\_mode='lastp',p=10)



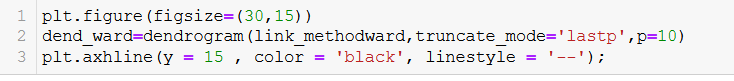
**Ward Method**

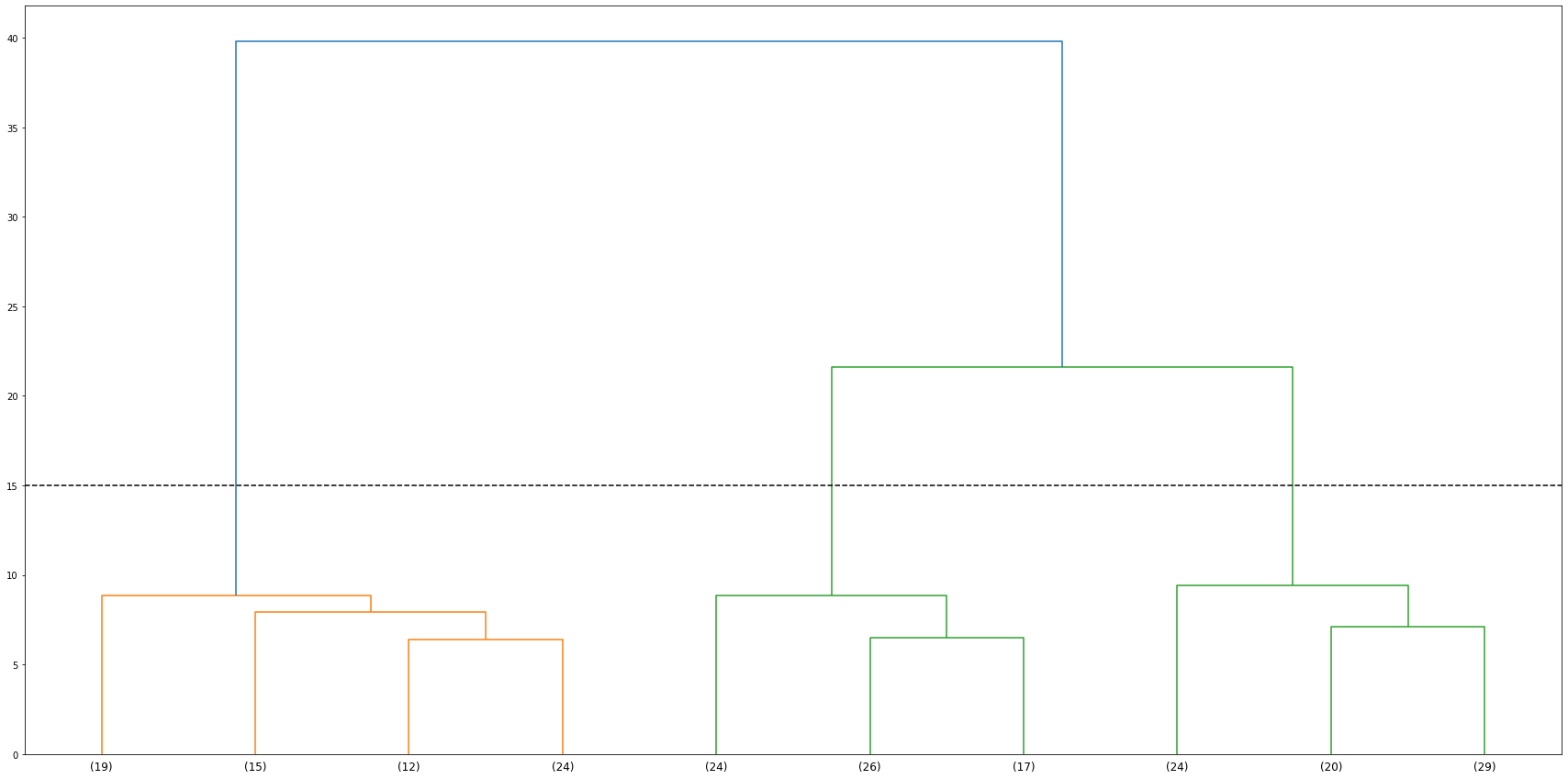


****

Clustering of the data has been done with the average and Ward's method for linkage, part of Agglomerative Hierarchical Clustering method.

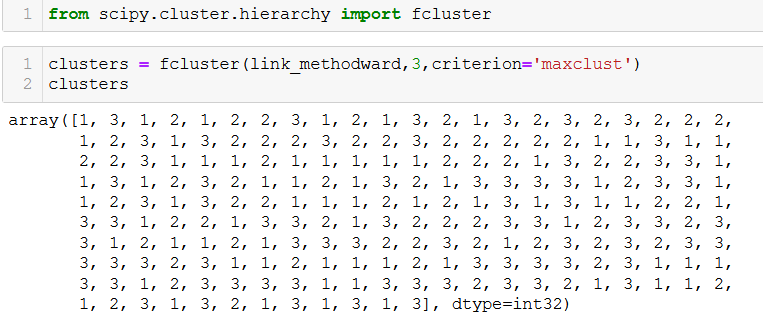
To create a Dendrogram using our scaled data we have firstly imported the package dendrogram, linkage from scipy.cluster.hierarchy. Using this function, we have created a dendrogram which shows two clusters very clearly. Now, we will check the make-up of these two clusters using ‘maxclust’ and ‘distance’. As can be seen from above we will now take 3 clusters for our further analysis.





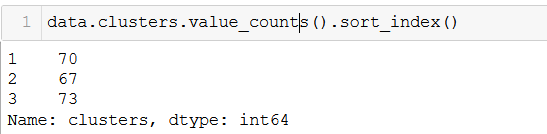
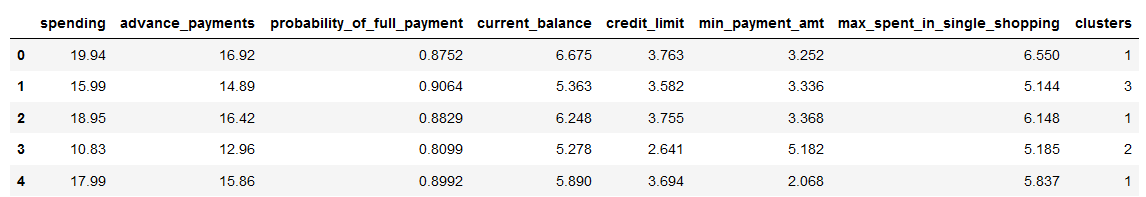
We have used 10 links in the dendrogram.

We use fcluster() to find the optimal number of clusters. With Wardlink method, and criterion as maxclust with value as 3 we find the optimal number of clusters. Also when you look at the dendrogram, it seems that 3 clusters would be optimal. But that is not the only evidence. With distance criterion and value as 20-25, we see that that the optimal number of clusters shall be 3. This cuts a horizontal line at the value mentioned and the number of lines vertical lines cut by this horizontal line gives the optimal number of clusters.



data['clusters']= clusters

data.head()

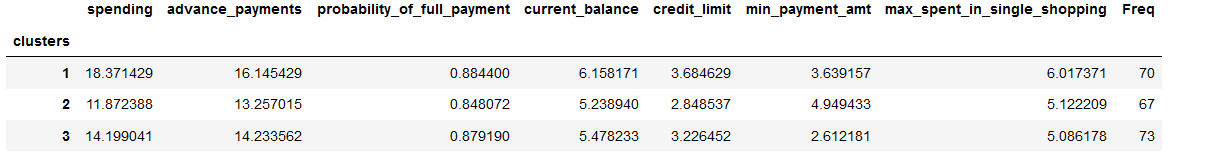


**Using groupby function to find the frequency.**

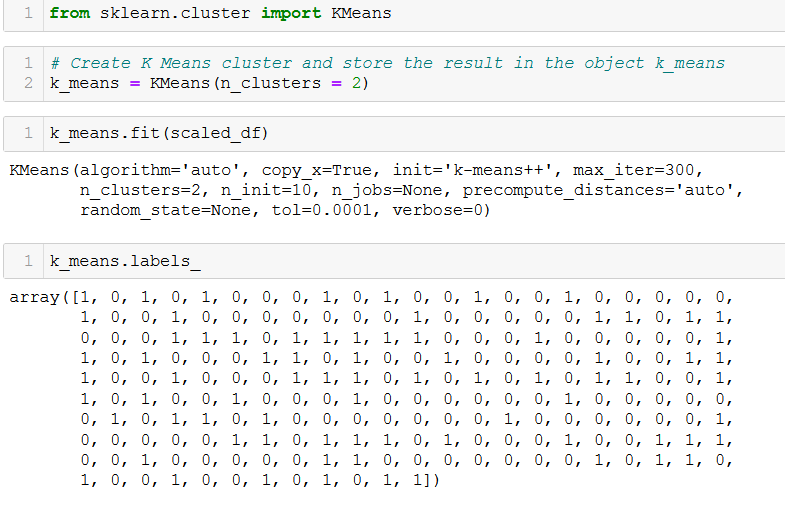
aggdata=data.groupby('clusters').mean()

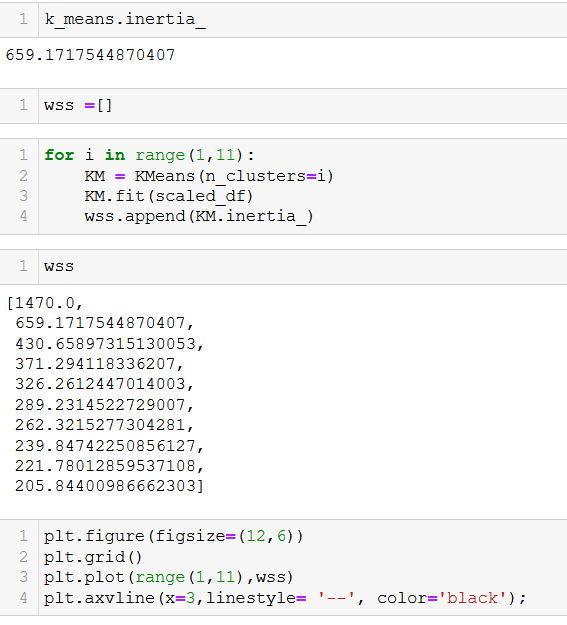
aggdata['Freq']=data.clusters.value\_counts().sort\_index()

aggdata

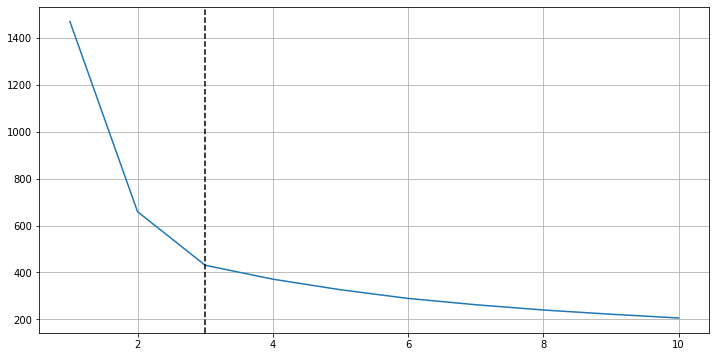


1.4 Apply K-Means clustering on scaled data and determine optimum clusters. Apply elbow curve and silhouette score.

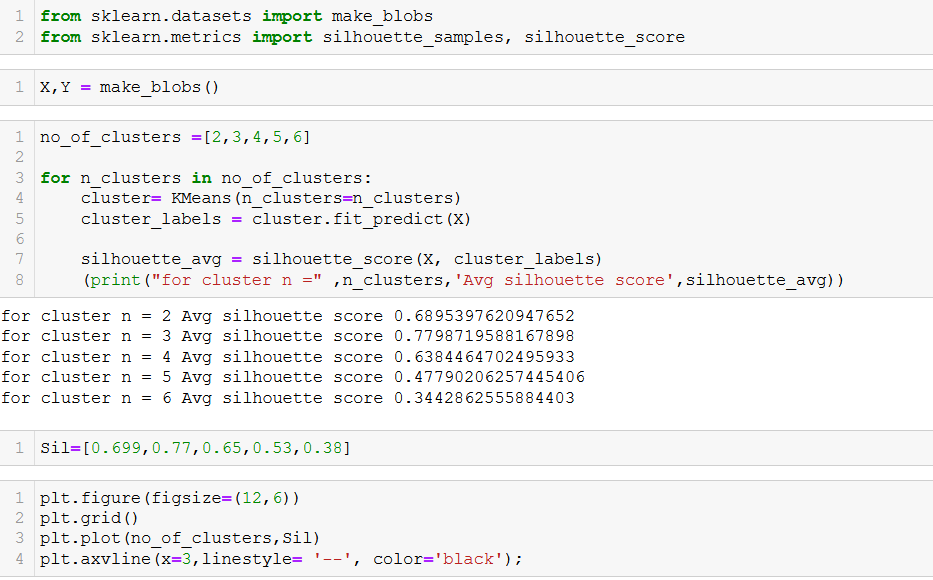


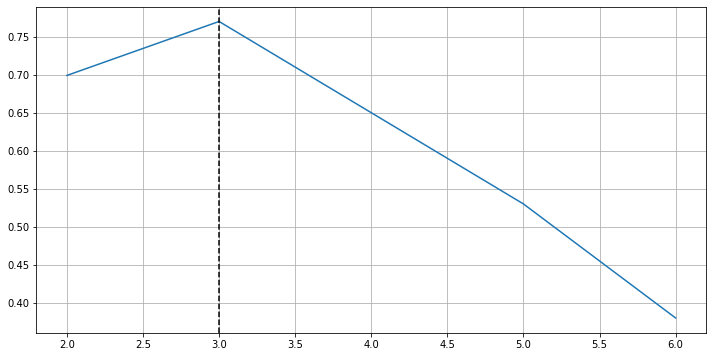


The inertia values are stored in the variable

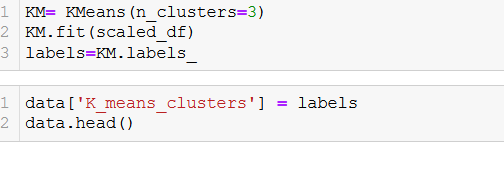


As per the above plot, within sum of squares [wss], we can conclude that optimal no of clusters to be taken for k means is 3.

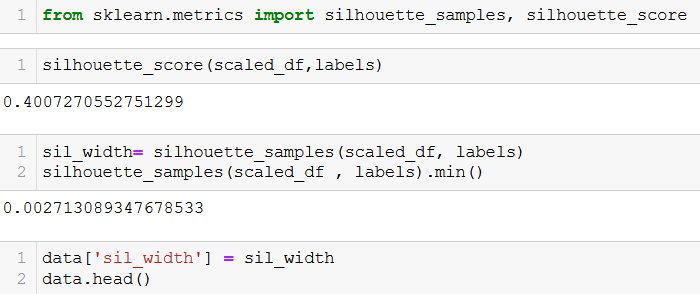




As per the plot silhouette average scores, highest average score is k=3. By both methods we have k=3.





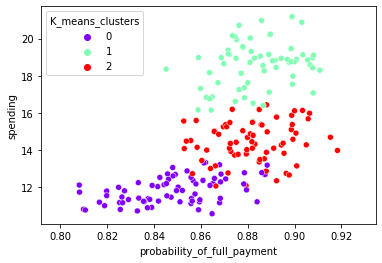




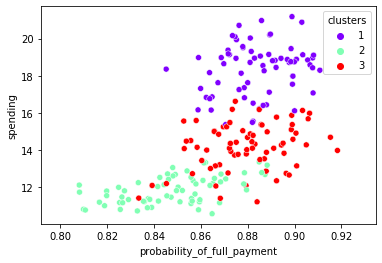
The average silhouettes score is coming to be 0.400 and minimum silhouette score is 0.002. The silhouette score ranges from -1 to +1 and higher the silhouette score better the clustering.

1.5. To describe cluster profiles for the clusters defined and recommend different promotional strategies for different clusters.

* sns.scatterplot(x = 'probability\_of\_full\_payment' ,y = 'spending', hue = 'K\_means\_clusters', palette= 'rainbow' , data = data);



sns.scatterplot(x = 'probability\_of\_full\_payment' ,y = 'spending', hue = 'clusters', palette= 'rainbow' , data = data);



Three different colors for 3 clusters and k means clusters.

aggdata=data.groupby('clusters').mean()

aggdata['Freq']=data.clusters.value\_counts().sort\_index()

aggdata



aggdata1=data.groupby('K\_means\_clusters').mean()

aggdata1['Freq']=data. K\_means.clusters.value\_counts().sort\_index()

aggdata1



**Observations for Hierarchical Clustering:**

* For Cluster 1, Spending is on higher side, averaging 18371. Probability of Full Payment is very high, averaging around 0.8844. It has the Highest advance payments around 1614. Current Balance is around 6158 which is highest among three clusters. Credit Limit is also high for this cluster ranging around 36846. And also, max\_spent\_in\_single\_shopping is around 6017. Which is higher comparatively to other clusters.
* For Cluster 2, this is average Spending cluster is on lower side, averaging 11872. Probability of Full Payment is the least amongst other clusters, averaging around 0.848. Current Balance is around 5238 which is least among three clusters. Credit Limit is least for this cluster ranging around 28485. And also, minimum payment amount is found max in this cluster, it is around 494. Max\_spent\_in\_single\_shopping is around 5122.
* For Cluster 3, Spending is 14199. Probability of Full Payment is little on the higher side, averaging around 0.879. Current Balance is around 5478 which is average among three clusters. Credit Limit is ranging in between for this cluster ranging around 32264. And also, minimum payment amount is found least in this cluster, it is around 261. Max\_spent\_in\_single\_shopping is the least around 5086.

**Observations for K-Means Clustering:**

* For Cluster 1, this is least Spending cluster, averaging 11856. Probability of Full Payment is the least amongst other clusters, averaging around 0.848. Current Balance is around 5231 which is least among three clusters. Credit Limit is least for this cluster ranging around 28495. And also, minimum payment amount is found max in this cluster, it is around 474. Max\_spent\_in\_single\_shopping is around 5101.
* For Cluster 2, Spending is highest amongst other clusters, averaging 18495. Probability of Full Payment is little on the higher side, averaging around 0.884. Current Balance is around 6175 which is highest among three clusters. Credit Limit is ranging in between for this cluster ranging around 36975. It is highest amongst three clusters. And also, minimum payment amount is found least in this cluster, it is around 363. Max\_spent\_in\_single\_shopping is the highest around 6041.
* For Cluster 3, Spending is average spending cluster, averaging 14437. Probability of Full Payment is least among other Clusters, averaging around 0.8815. It has the Highest advance payments around 1433. Current Balance is around 5514. It is average when compared to other two clusters. Credit Limit is for this cluster ranging around 32592. Minimum payment is averaging 270. This is the least amongst others. And also, max\_spent\_in\_single\_shopping is around 5120.

**Recommendations for Hierarchical Clustering:**

Bank should focus on Cluster 1 as the customers in this cluster have higher spending. Bank can think of providing them offers like, for shopping if they spend more than their current maximum spending in single shopping. Cluster 2 spends the least reason might be less Credit limit. There are probabilities that these customers may increase spending if their Credit limit increases. Cluster 3 has average spending. Bank should give customers in this Cluster more promotional offers because there are more chances that these customers may move to Cluster 1 of high spending. Cluster 2 and 3 have lesser maximum one-time spending compared to Cluster 1, which has highest max one-time spending. Hence Bank should promote customers in Cluster 2 & 3 for more effect of the promotional offers.

**Recommendations for K-Means Clustering:**

Bank should focus on Cluster 3 as the customers in this cluster have higher spending. Bank can think of providing them offers like, for shopping if they spend more than their current maximum spending in single shopping. Cluster 2 spends the least reason might be less Credit limit. There are probabilities that these customers may increase spending if their Credit limit increases. Cluster 3 has average spending. Bank should give customers in this Cluster more promotional offers because there are more chances that these customers may move to Cluster 1 of high spending. Cluster 1 and 2 have lesser maximum one-time spending compared to Cluster 3, which has highest max one-time spending. Hence Bank should promote customers in Cluster 1 & 2 for more effect of the promotional offers.

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

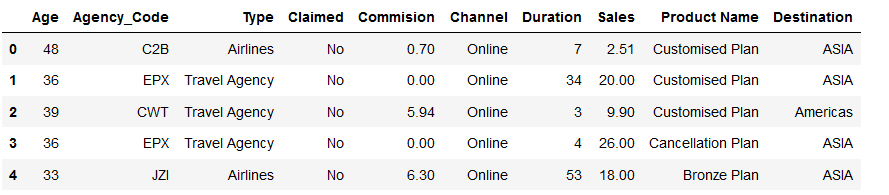
An Insurance firm providing tour insurance is facing higher claim frequency. The management decides to collect data from the past few years. You are assigned the task to make a model which predicts the claim status and provide recommendations to management. Use CART, RF & ANN and compare the models' performances in train and test sets.

**2.1** Data Ingestion: Read the dataset. Do the descriptive statistics and do null value condition check, write an inference on it.   
**2.2** Data Split: Split the data into test and train, build classification model CART, Random Forest, Artificial Neural Network  
**2.3** Performance Metrics: Check the performance of Predictions on Train and Test sets using Accuracy, Confusion Matrix, Plot ROC curve and get ROC\_AUC score for each model  
**2.4** Final Model: Compare all the model and write an inference which model is best/optimized.  
**2.5** Inference: Based on the whole Analysis, what are the business insights and recommendations

**2.1** Data Ingestion: Read the dataset. Do the descriptive statistics and do null value condition check, write an inference on it.

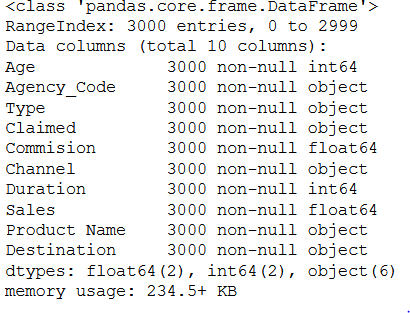
**Viewing the dataset**

* df=pd.read\_csv('insurance.csv')
* df.head()



**Information of dataset**

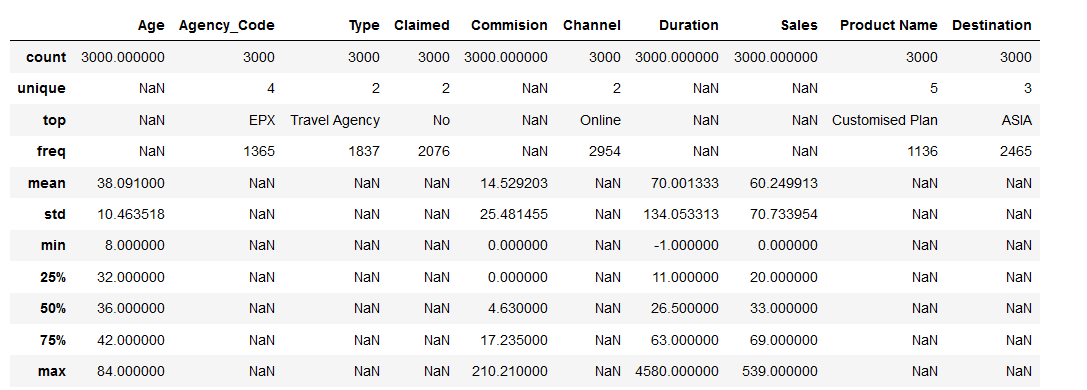
* data.info()



From the above data , there are 6 variables as objects, 2 float and 2 int variables

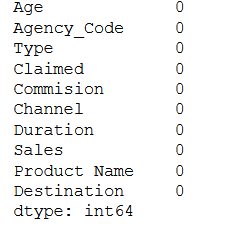
Describe the dataset

* data.describe(include="all")



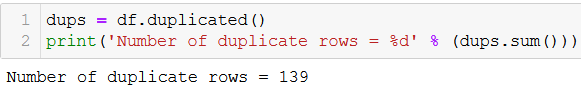
**Checking null values**

* data.isnull().sum()



There are no null/missing values present in the dataset

**Checking Duplicates**



There are 139 duplicates values present in the dataset

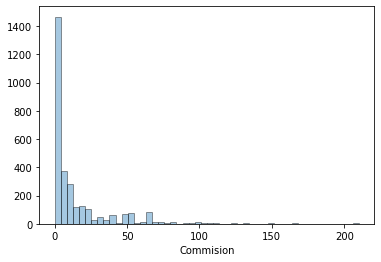
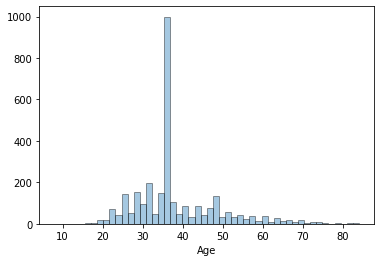
Since there is no customer id, there can be customers of same age to book the travel package, so duplicates is not removed.

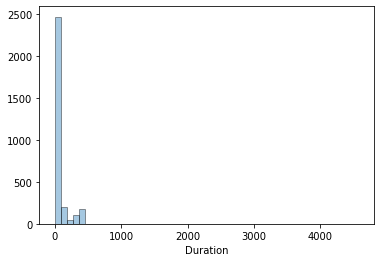
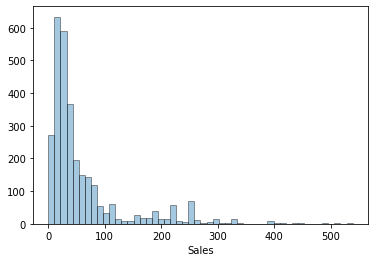
**Shape of the dataset**

(3000, 10)

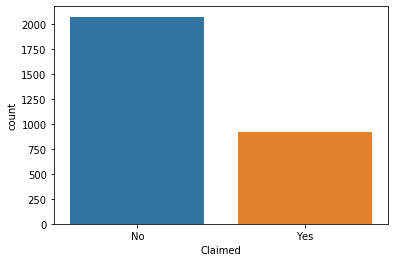
There are 10 columns and 3000 rows in the dataset

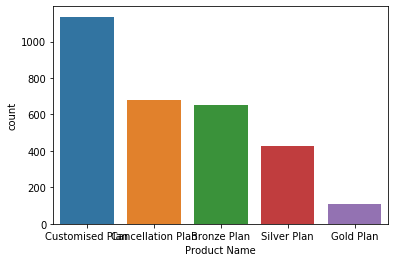
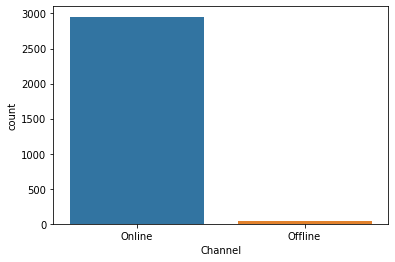
**Univariate Analysis**

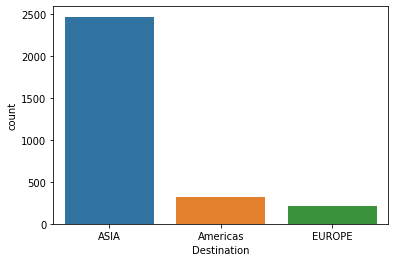


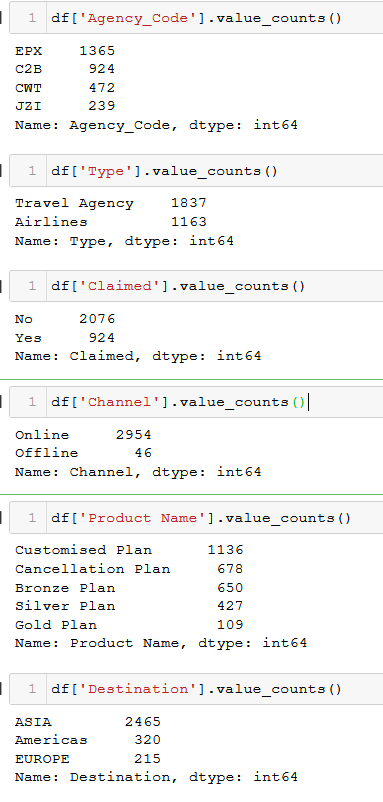
**Countplot for Categorical variables**

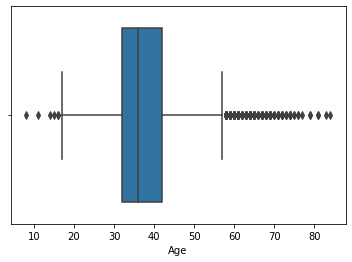
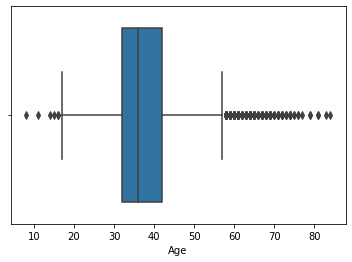


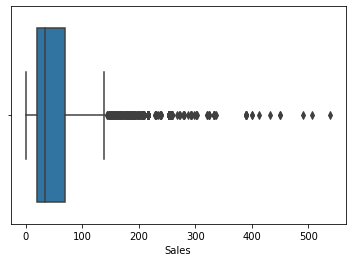


**Valuecount for categorical variables**



**Boxplot**

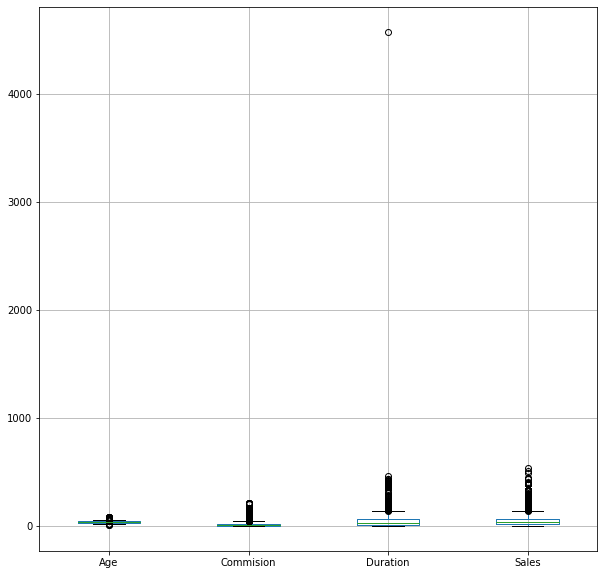
 

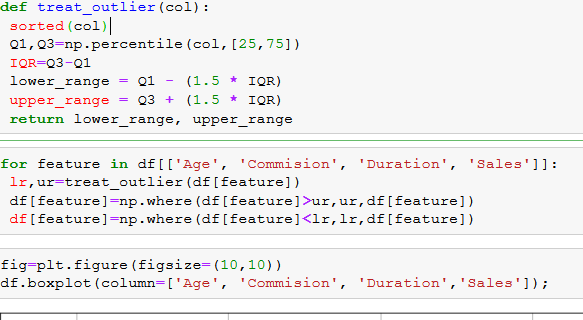
There are outliers to be treated

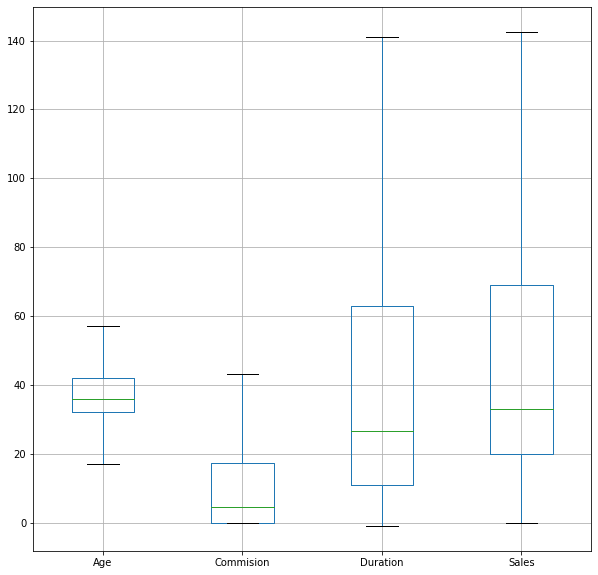
**Outliers**





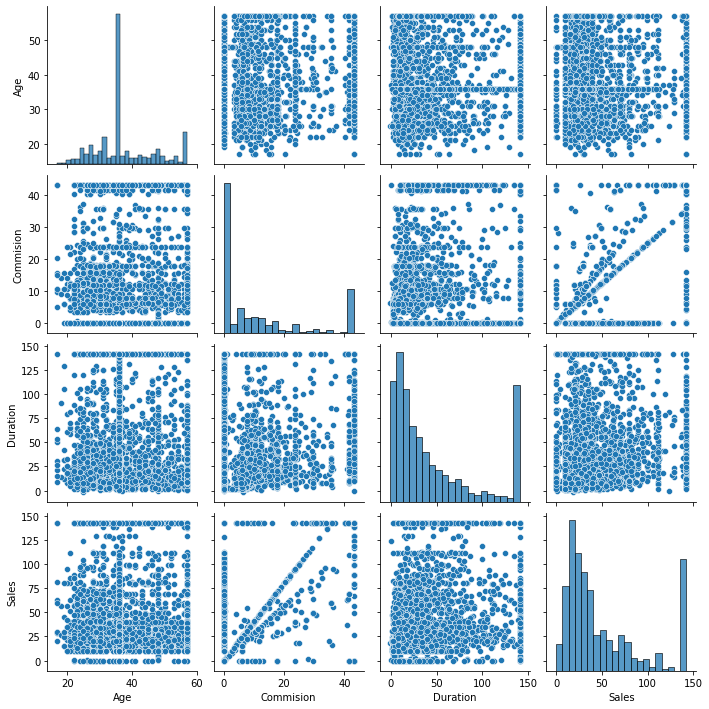
Treating Outliers





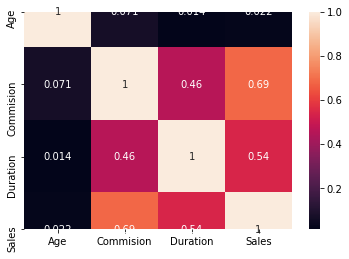
**Multivariate Analysis**

**Pairplot**



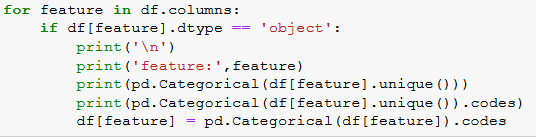
HeatMap

sns.heatmap(df.corr(),annot=True);



There is no extreme low correlation between the variables

**Converting categorical variables to int datatypes**



* feature: Agency\_Code

[C2B, EPX, CWT, JZI]

Categories (4, object): [C2B, CWT, EPX, JZI]

[0 2 1 3]

* feature: Type

[Airlines, Travel Agency]

Categories (2, object): [Airlines, Travel Agency]

[0 1]

* feature: Claimed

[No, Yes]

Categories (2, object): [No, Yes]

[0 1]

* feature: Channel

Online, Offline]

Categories (2, object): [Offline, Online]

[1 0]

* feature: Product Name

[Customised Plan, Cancellation Plan, Bronze Plan, Silver Plan, Gold Plan]

Categories (5, object): [Bronze Plan, Cancellation Plan, Customised Plan, Gold Plan, Silver Plan]

[2 1 0 4 3]

* feature: Destination

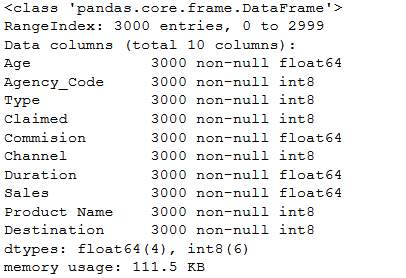
[ASIA, Americas, EUROPE]

Categories (3, object): [ASIA, Americas, EUROPE]

[0 1 2]

Checking dataset

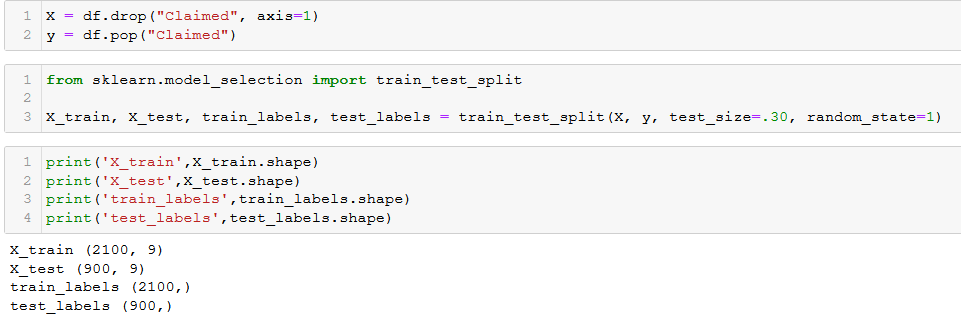
* df.info()



2.2 Data Split: Split the data into test and train, build classification model CART, Random Forest, Artificial Neural Network

**Splitting tarin and test data:**

In the ratio of 30 and 70 for test and training respectively.



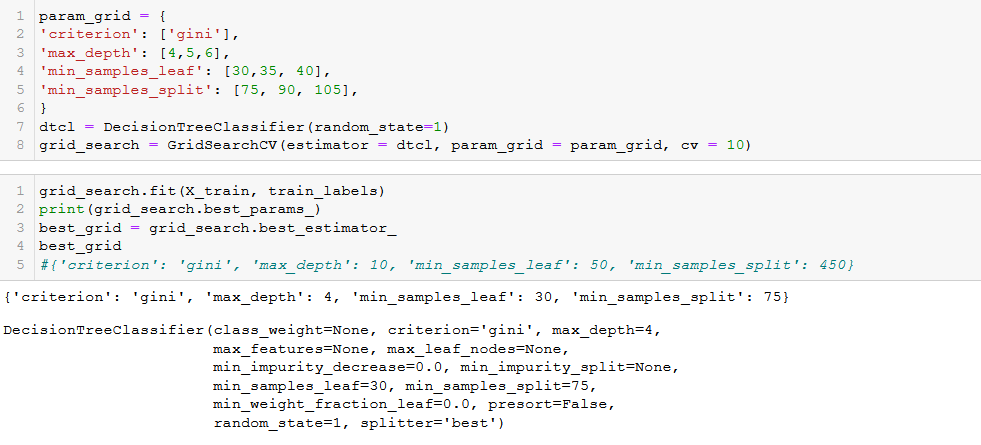
We are using “gini” as the criterion.

The GridSearchCV function in python helps us to determine optimum number of nodes,

minimum sample leaf size, minimum sample split size. For this we have to import the

GridSearchCV from sklearn.model\_selection.

**CART Model:**



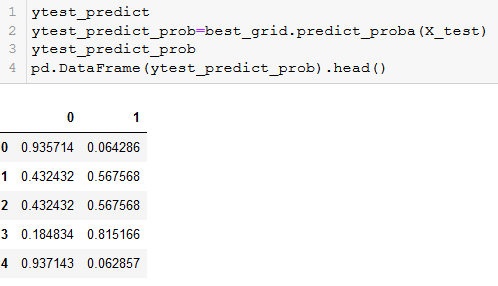
Below are the variable importance values or the feature importance to build the tree



**Predicting on Training and Testing data**

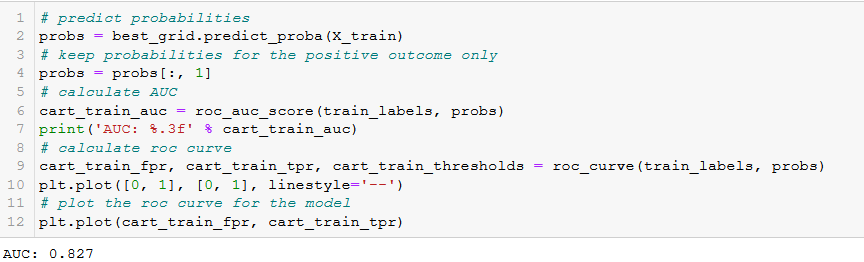


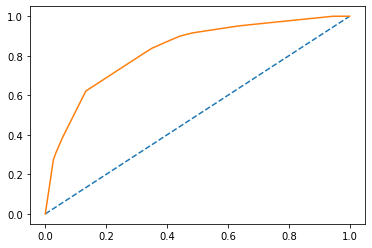
**Getting the Predicted Classes and Probs**



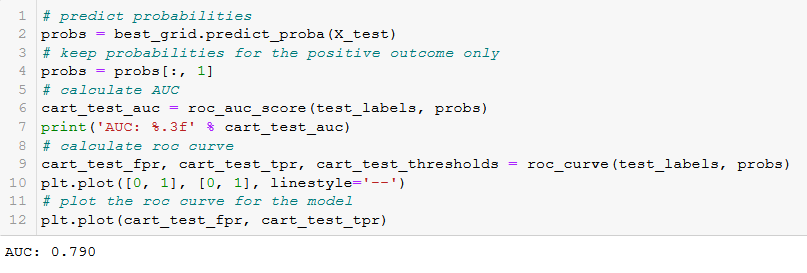
The probabilities of train data, calculate the AUC, calculating the ROC curve and plotting the same.

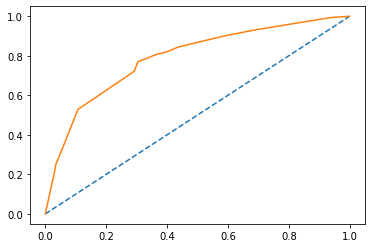
**Training Data:**





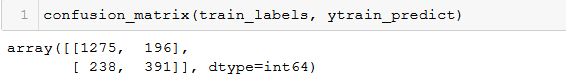
**Test data:**

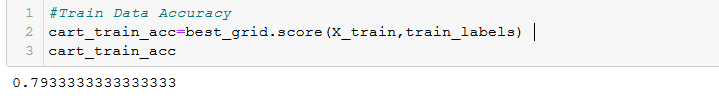




**Confusion matrix:**

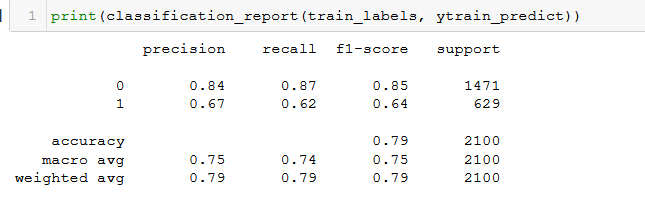
**Training data:**



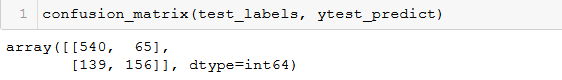


Data accuracy : 0.7933

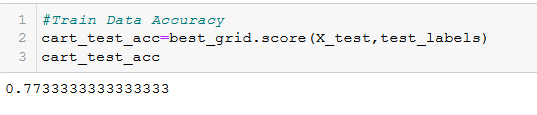
Classification report:



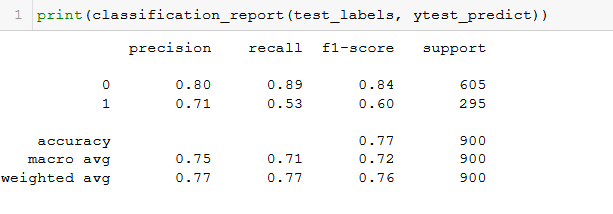
**Testing Data:**

****

**Data accuracy: 0.7733**

****

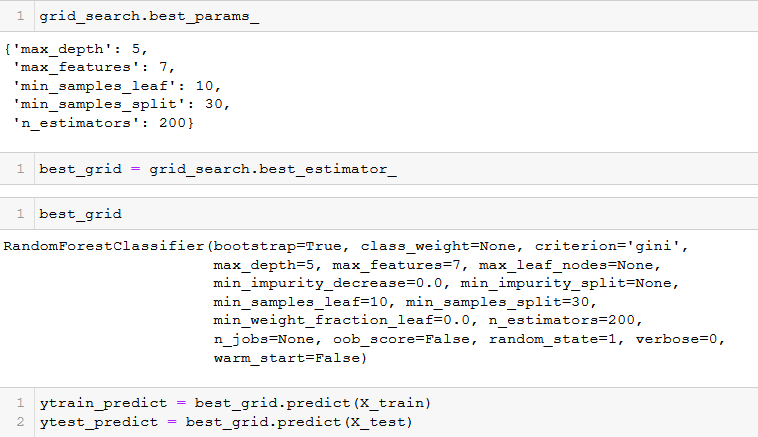
**Classification report :**



**Random Forest**

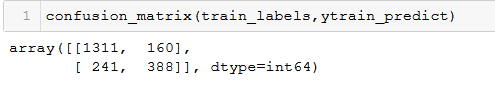


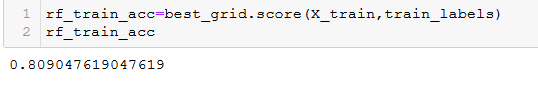
The grid search and fit our data in to it to give us best parameters for the mode



on running the code grid\_search.best\_params\_, we will get the following parameters are best for the given values

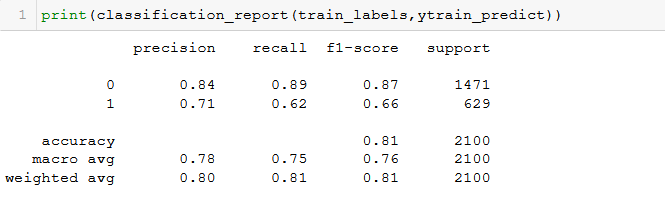
**Confusion Matrix: Training Data**



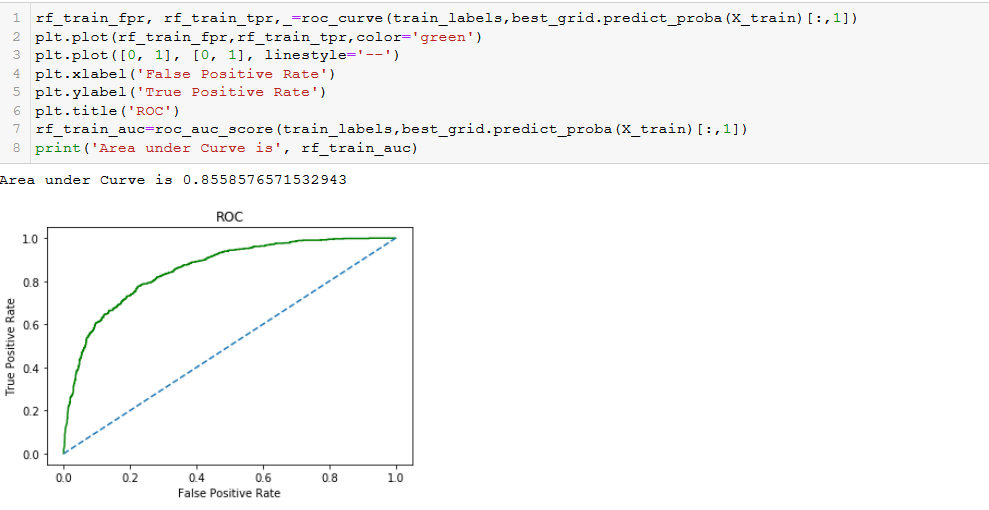


Accuracy Value: 0.8090

**Classification report :**



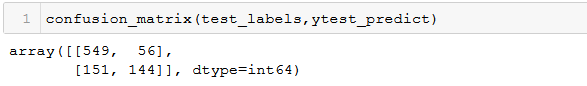
**ROC curve and AUC :**

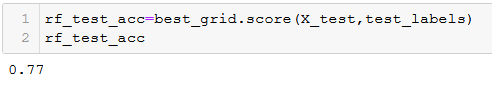


AUC : 0.855

**Test data:**

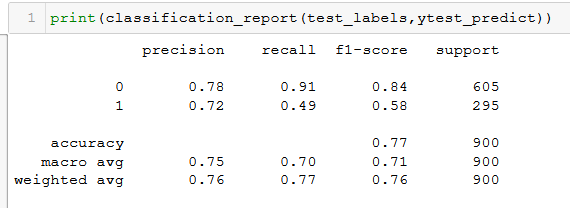
**Confusion Matrix :**



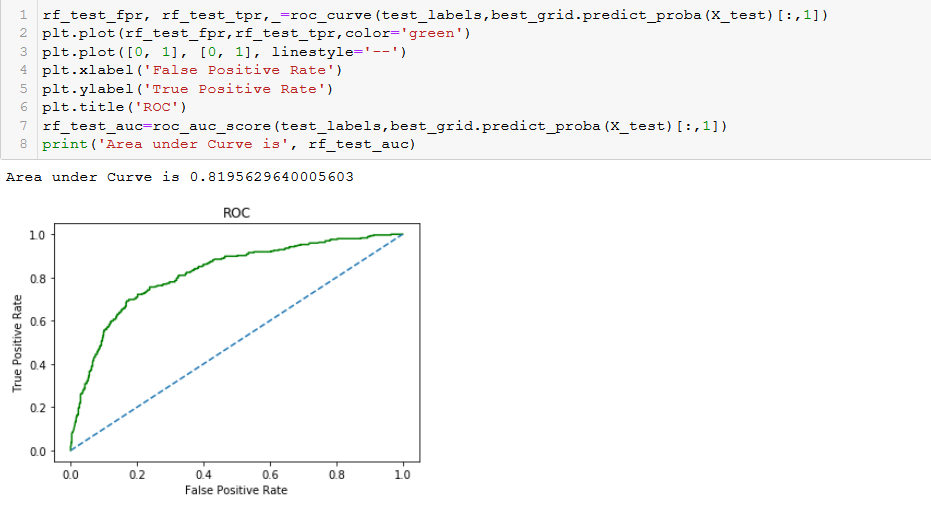


Accuracy value : 0.77

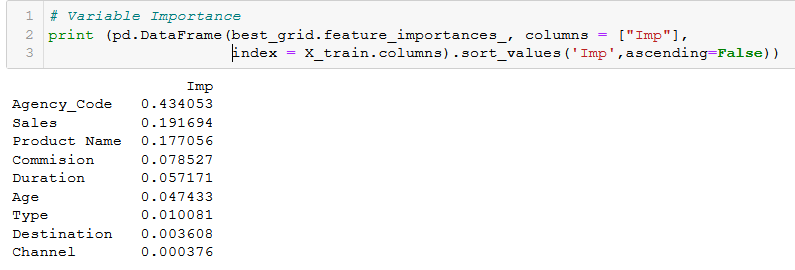
**Classification Report :**



**ROC curve and AUC** :

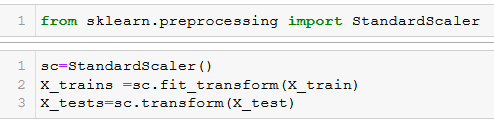


AUC = 0.819

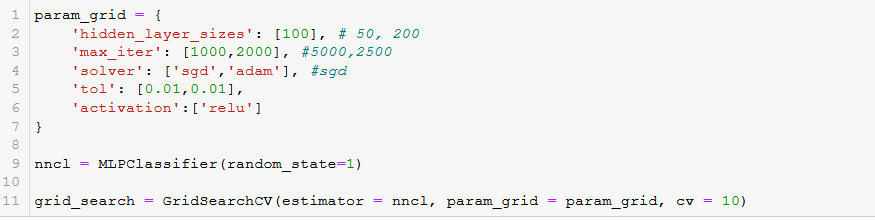


**Artificial Neural Network Model:**

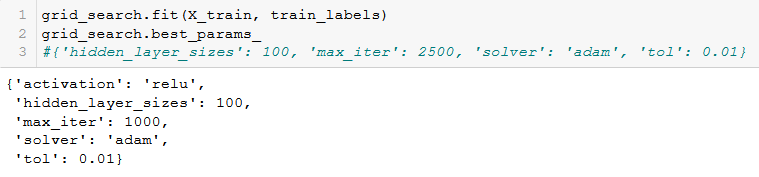
**Scaling of the data:**

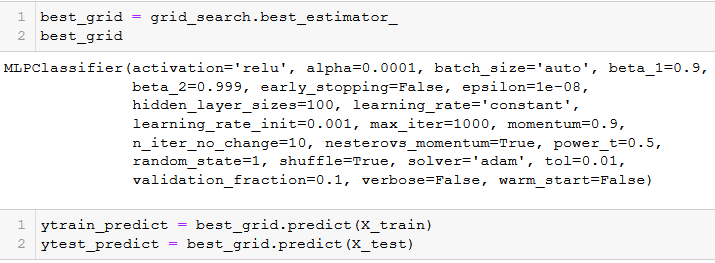


**Grid Search :**

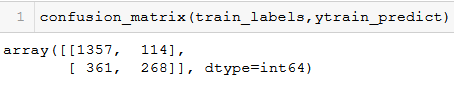


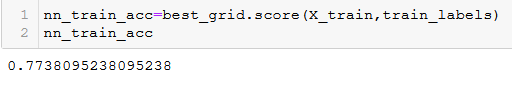
we will fit the train data to estimate the best parameters





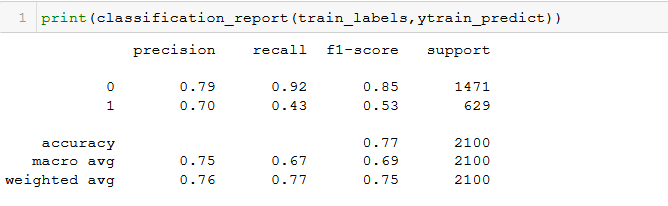
**Confusion matrix : Train data**

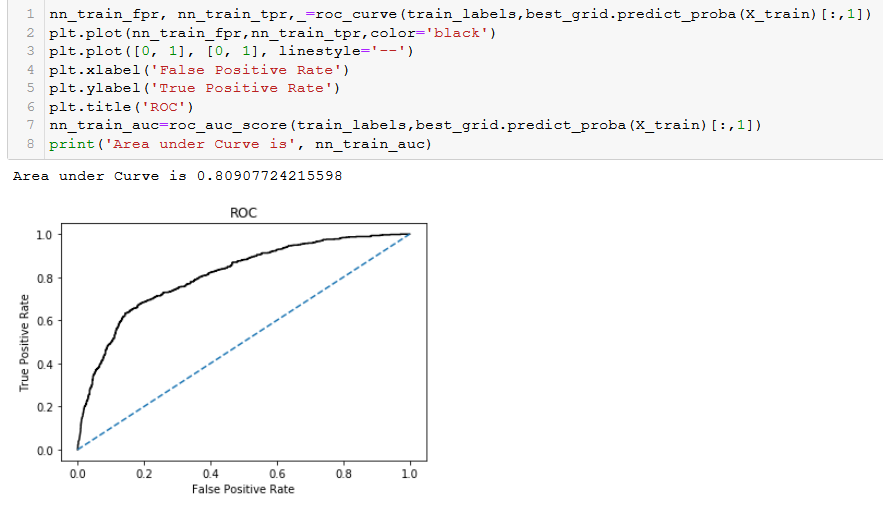




Accuracy value : 0.77

**Classification Report :**

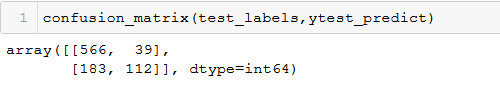
**ROC curve and AUC :**

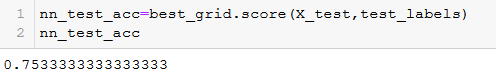


AUC : 0.809

**Test data:**

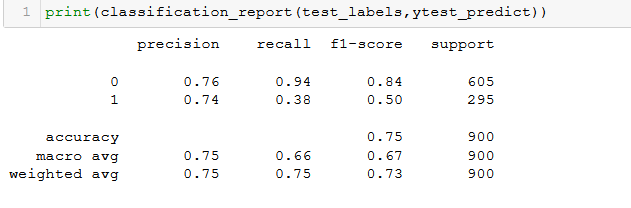
**Confusion Matrix :**



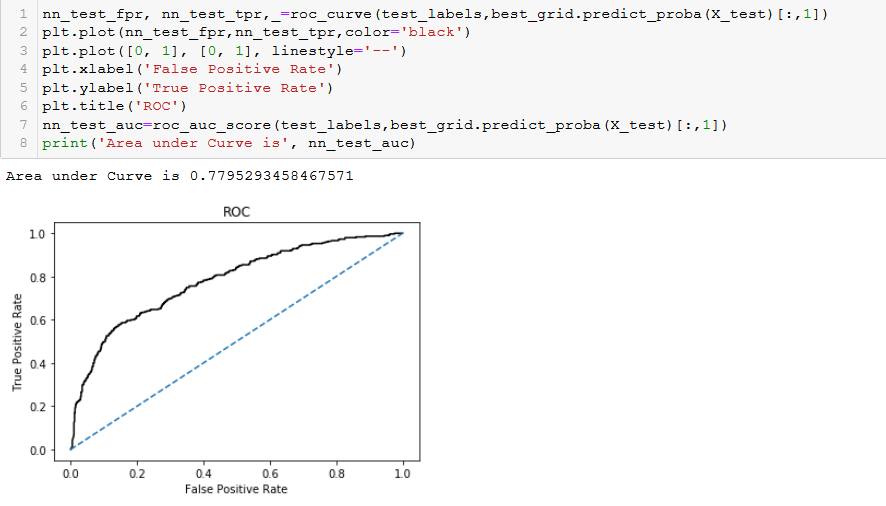


Accuracy value : 0.753

**Classification report :**



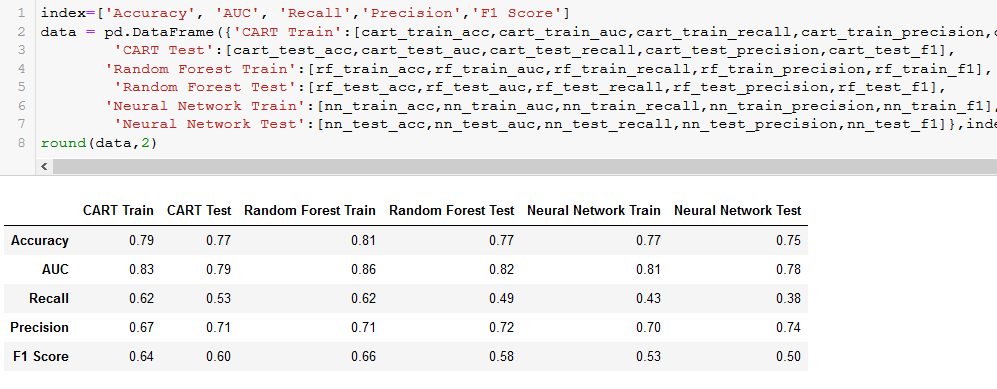
**ROC curve and AUC:**



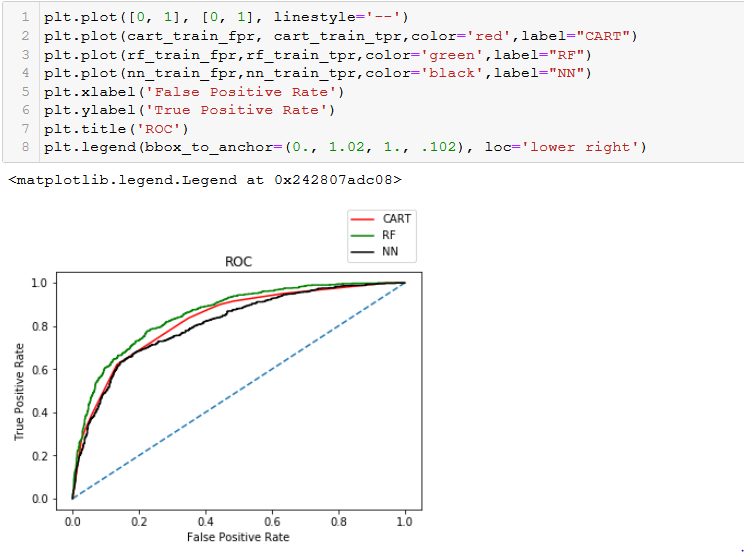
AUC :0.779

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

A comparative table for test and train data for all the tree i.e. CART, RF and ANN models.

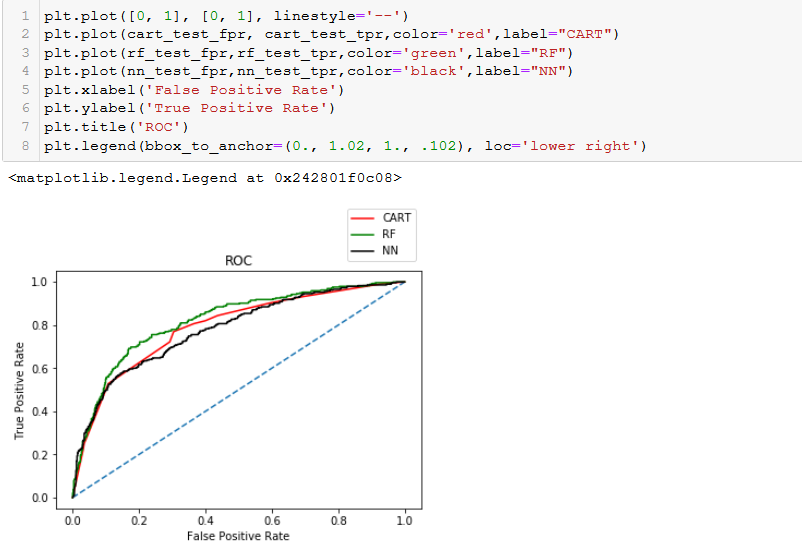


The Accuracy, AUC, Precision and F-1 scores are highest in Random forest train set. We can come to a conclusion on plotting the ROC curve for all Testing data and Training data. For **Training data**, we get the following curve



we can say that the Random Forest model with green colour is having heist AUC.

**Testing data:**



we can say that the Random Forest model with green colour is having heist AUC.

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

* The Accuracy, Precision, F1 Score are computed using Classification Report. The confusion matrix, AUC\_ROC Scores and ROC plot are computed for each model separately and compared. All the three models have performed well but to increase our accuracy in determining the claims made by the customers we can choose the Random Forest Model. Instead of creating a single Decision Tree it can create a multiple decision trees and hence can provide the best claim status from the data.
* For all the models i.e. CART, Random Forest and ANN have performed exceptionally well. Hence, we can choose either of the models but choosing Random Forest Model is a great option as even though they exhibit the same accuracy but choosing Random Forest over Cart model is way better as they have much less variance than a single decision tree.
* By performing the 3 models, we can conclude that
  + the data set is well balanced to conduct the modelling.
  + The data set is containing significant outliers
  + The Agency code has significant importance
* Claims are Higher for Online Distribution channel of tour insurance agencies. Claims are very low for Offline Distribution channel of tour insurance agencies. Reason might be, in recent time many people are preferring Online purchase which is very easier. So, Management can think of promoting offline Distribution channel of tour insurance agencies in order to reduce claims. Offline Purchase can be made more attractive by offering extra discounts or additional benefits.
* Higher Claims are observed for Agency Code C2B. So, Management need to check why claim state is high for this agency. Reason might be lack of knowledge to the insurance representative on insurance policies which leads loop holes. This might be leading to high claims. There is also a possibility that people might be purposely taking insurance from this agency for the reason of easy claims. This agency services can be compared with other agencies with are leading to fewer number of claims.
* Claims are higher for Airlines as Type of tour insurance firms. There are many factors involved in this like flight delay, baggage delay poor service recorded by the Airline Service or connection flight missing. These are usual; hence terms and conditions can be added or delay time frame can be increased. For baggage loss or any other delays, we can increase the premium. Or claim value can be reduced by some percentage.
* Claims are higher for Silver plan which is one of the Name of the tour insurance products. We need to recheck for which reason are we getting claims from this Silver plan, or this plan can be stopped for sometime to confirm if this reduces the claims. Other plans can be promoted in order to reduce the claims.
* Claims are higher for Destination of the tour as ASIA. We need to dig out the reason why there are high claims for ASIA. Several reasons are present for it. People are negligent may be in ASIA. Terms and Conditions need to be increased for ASIA. There is also possibility that we can increase the premium for insurances related to ASIA in order to compensate for the claims. If customers are checking for ASIAN travels, we can give them more interesting options of other places to travel.
* The category for which company is getting more claims can be added as addons at extra cost. And the category for which company is getting less claims can be kept as basic facilities. And this can be varied based on location of travel.
* Management can also think of keeping the purchase procedure simple but we need to increase the complexity of the Claim Procedure.

E.g. Claiming can be done only by offline mode, so as to verify the genuineness of the claim.