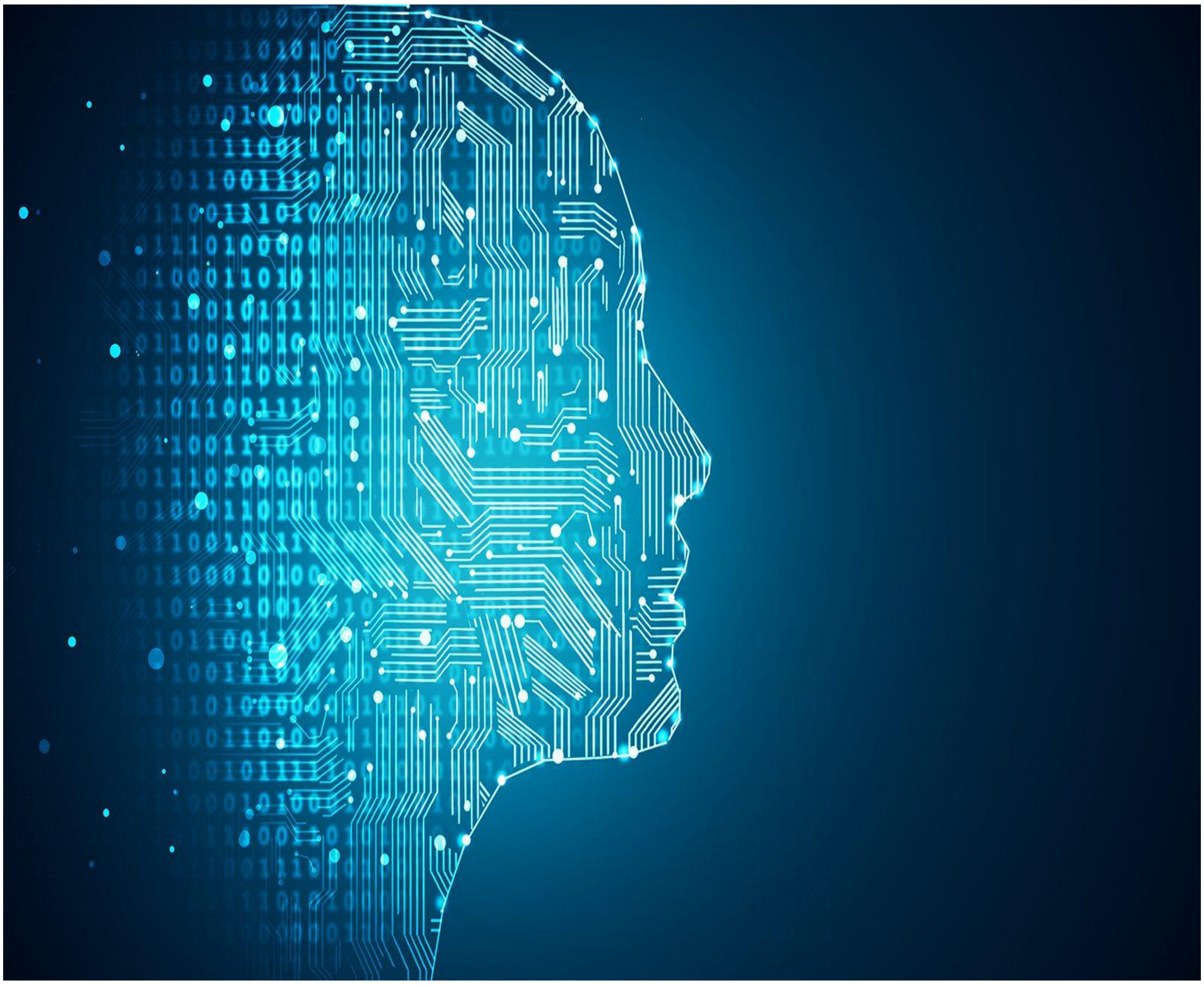
Great Learning

Machine Learning- Project

Clustering, CART, Random Forest & ANN



Sridhar V

8/16/2020

1. Project 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.

Step by Step Approach
We shall follow step by step approach to arrive to the conclusion as follows:

- EDA : Exploratory Data Analysis
- Scaling of the data
- Build Hierarchical clustering model : Agglomerative Hierarchical Clustering
- Interpret optimal number of clusters within dendogram and Visualizing them
- Aggregate of the Hierarchical clusters
- Silhouette score
- Determine optimal number of clusters for K-means using Distant Gradient, WSS , Silhoutee, Gap method.
- Perform K-means clustering with the determined number of clusters
- Visualize the clusters
- Calculate the silhouette score, profile the clusters based on the aggregate

#Q1 Read the data and do exploratory data analysis. Describe the data briefly

1.1. Exploring Data

Data Dictionary:

Data have 7 variable naming

- Spending: Amount spent by the customer per month (in 1000s)
- Advance_payments: Amount paid by the customer in advance by cash (in 100s)
- Probability_of_full_payment: Probability of payment done in full by the customer to the bank
- Current_balance: Balance amount left in the account to make purchases (in 1000s)
- Credit_limit: Limit of the amount in credit card (10000s)
- Min_payment_amt : minimum paid by the customer while making payments for purchases made monthly (in 100s)
- Max_spent_in_single_shopping: Maximum amount spent in one purchase (in 1000s)

Reading of Data

Bank_data <- read.csv(file.choose(),header = TRUE)

Snippet of Data

A data.frame: 6 × 7

	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping
1	19.94	16.92	0.8752	6.675	3.763	3.252	6.550
2	15.99	14.89	0.9064	5.363	3.582	3.336	5.144
3	18.95	16.42	0.8829	6.248	3.755	3.368	6.148
4	10.83	12.96	0.8099	5.278	2.641	5.182	5.185
5	17.99	15.86	0.8992	5.890	3.694	2.068	5.837
6	12.70	13.41	0.8874	5.183	3.091	8.456	5.000

From the given data we have to cluster and classify. Clustering would consists of Agglomerative or Hierarchical Clustering and K Means.

Transforming the data to their original values

A data.frame: 6 × 7

	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping
1	19940	1692	0.8752	6675	37630	325.2	6550
2	15990	1489	0.9064	5363	35820	333.6	5144
3	18950	1642	0.8829	6248	37550	336.8	6148
4	10830	1296	0.8099	5278	26410	518.2	5185
5	17990	1586	0.8992	5890	36940	206.8	5837
6	12700	1341	0.8874	5183	30910	845.6	5000

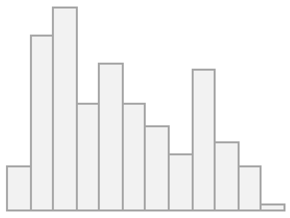
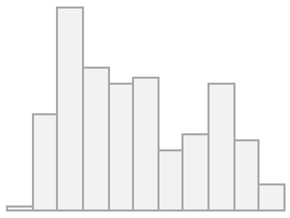
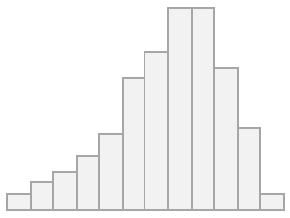
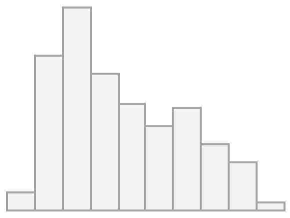
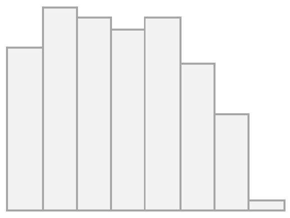
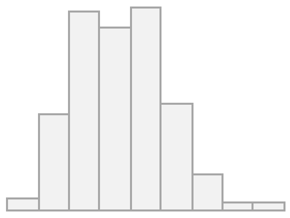
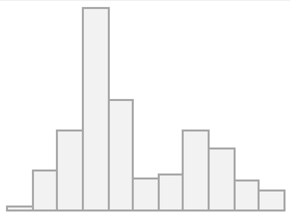
In the earlier data explanation each variable were counted with particular units of 1000 and 100 so to see the original values the given data was multiplied with their respective units.

Data Frame Summary

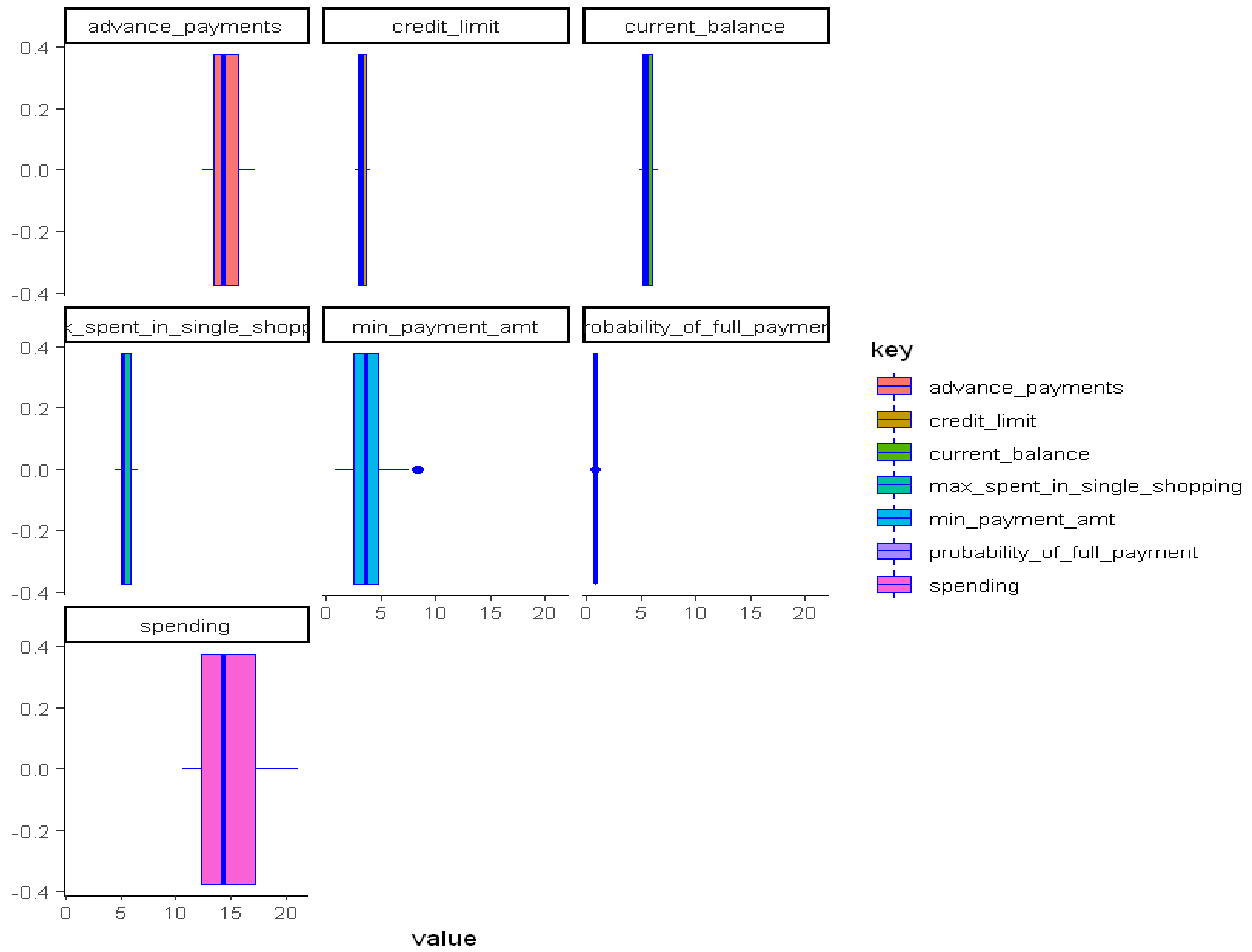
Bank_data

Dimensions: 210 x 7

Duplicates: 0

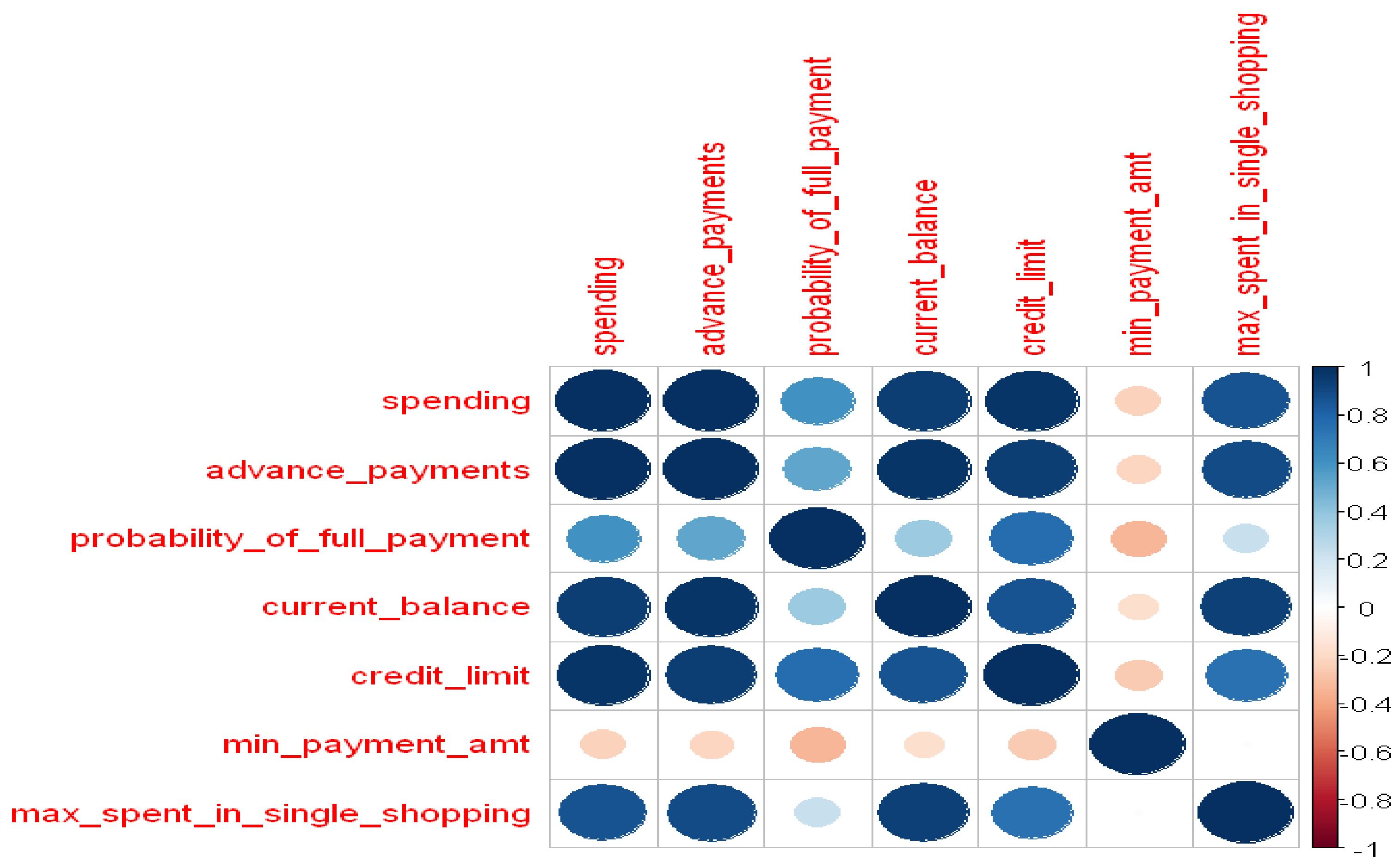
No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Valid	Missing
1	spending [numeric]	Mean (sd) : 14847.5 (2909.7) min < med < max: 10590 < 14355 < 21180 IQR (CV) : 5035 (0.2)	193 distinct values		210 (100%)	0 (0%)
2	advance_payments [numeric]	Mean (sd) : 1455.9 (130.6) min < med < max: 1241 < 1432 < 1725 IQR (CV) : 226.5 (0.1)	170 distinct values		210 (100%)	0 (0%)
3	probability_of_full_payment [numeric]	Mean (sd) : 0.9 (0) min < med < max: 0.8 < 0.9 < 0.9 IQR (CV) : 0 (0)	186 distinct values		210 (100%)	0 (0%)
4	current_balance [numeric]	Mean (sd) : 5628.5 (443.1) min < med < max: 4899 < 5523.5 < 6675 IQR (CV) : 717.5 (0.1)	188 distinct values		210 (100%)	0 (0%)
5	credit_limit [numeric]	Mean (sd) : 32586 (3777.1) min < med < max: 26300 < 32370 < 40330 IQR (CV) : 6177.5 (0.1)	184 distinct values		210 (100%)	0 (0%)
6	min_payment_amt [numeric]	Mean (sd) : 370 (150.4) min < med < max: 76.5 < 359.9 < 845.6 IQR (CV) : 220.7 (0.4)	207 distinct values		210 (100%)	0 (0%)
7	max_spent_in_single_shopping [numeric]	Mean (sd) : 5408.1 (491.5) min < med < max: 4519 < 5223 < 6550 IQR (CV) : 832 (0.1)	148 distinct values		210 (100%)	0 (0%)

Boxplot



Except Min payment amount rest of the variable doesn't have an outlier. In Min payment amount 845.6 is an outlier

Correlation



Except Minimum payment amount and Probability of full payment rest of the variable show high correlation.

#Q2 Do you think scaling is necessary for clustering in this case?

1.2. Scaling

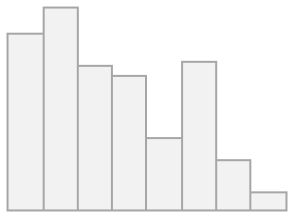
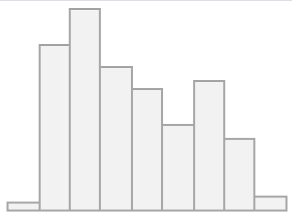
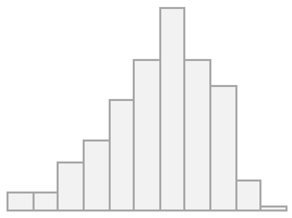
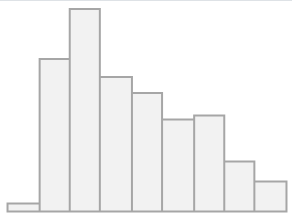
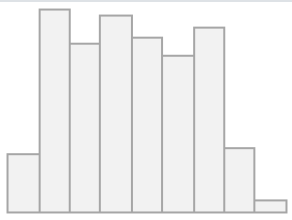
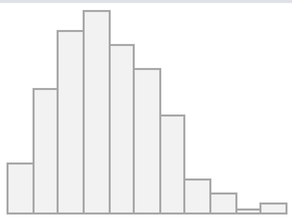
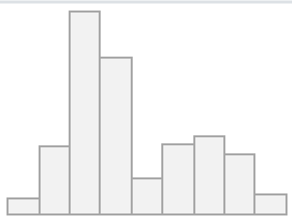
The variables have different magnitude which would create problem when we undergo distance or weight based model like clustering. As the larger magnitude variable would have more effect on the overall calculation of the model than the smaller magnitude variable. To suppress this effect, we need to bring all features to the same level of magnitudes. This can be achieved by scaling.

Snippet of scaled data

A matrix: 6 × 7 of type dbl

spending	advance_ payments	probability_of_ full_payment	current_ balance	credit _limit	min_paym ent_amt	max_spent_in_sing le_shopping
1.75	1.81	0.18	2.36	1.34	-0.30	2.32
0.39	0.25	1.50	-0.60	0.86	-0.24	-0.54
1.41	1.42	0.50	1.40	1.31	-0.22	1.51
-1.38	-1.22	-2.59	-0.79	-1.64	0.99	-0.45
1.08	1.00	1.19	0.59	1.15	-1.09	0.87
-0.74	-0.88	0.69	-1.01	-0.44	3.16	-0.83

Data Summary of Scaled Data

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Valid	Missing
1	spending [numeric]	Mean (sd) : 0 (1) min < med < max: -1.5 < -0.2 < 2.2 IQR (CV) : 1.7 (7950660579516545)	193 distinct values		210 (100%)	0 (0%)
2	advance_payments [numeric]	Mean (sd) : 0 (1) min < med < max: -1.6 < -0.2 < 2.1 IQR (CV) : 1.7 (1310995576611260)	170 distinct values		210 (100%)	0 (0%)
3	probability_of_full_payment [numeric]	Mean (sd) : 0 (1) min < med < max: -2.7 < 0.1 < 2 IQR (CV) : 1.3 (810744751775395)	186 distinct values		210 (100%)	0 (0%)
4	current_balance [numeric]	Mean (sd) : 0 (1) min < med < max: -1.6 < -0.2 < 2.4 IQR (CV) : 1.6 (-1045040369897904)	188 distinct values		210 (100%)	0 (0%)
5	credit_limit [numeric]	Mean (sd) : 0 (1) min < med < max: -1.7 < -0.1 < 2.1 IQR (CV) : 1.6 (5764470273742003)	184 distinct values		210 (100%)	0 (0%)
6	min_payment_amt [numeric]	Mean (sd) : 0 (1) min < med < max: -2 < -0.1 < 3.2 IQR (CV) : 1.5 (12444156865102686)	207 distinct values		210 (100%)	0 (0%)
7	max_spent_in_single_shopping [numeric]	Mean (sd) : 0 (1) min < med < max: -1.8 < -0.4 < 2.3 IQR (CV) : 1.7 (3932289398839753)	148 distinct values		210 (100%)	0 (0%)

Earlier when the data was not scaled the variable Probability of full payment which was in range of 0 to 1 would have the least effect on the model in comparison to values like spending which was in range of 1000.
As we can see in data summary IQR has been scaled between 1.3 to 1.7 making all variable on a single plain

#Q3 Apply hierarchical clustering to scaled data. Identify the number of optimum clusters using Dendrogram and briefly describe them

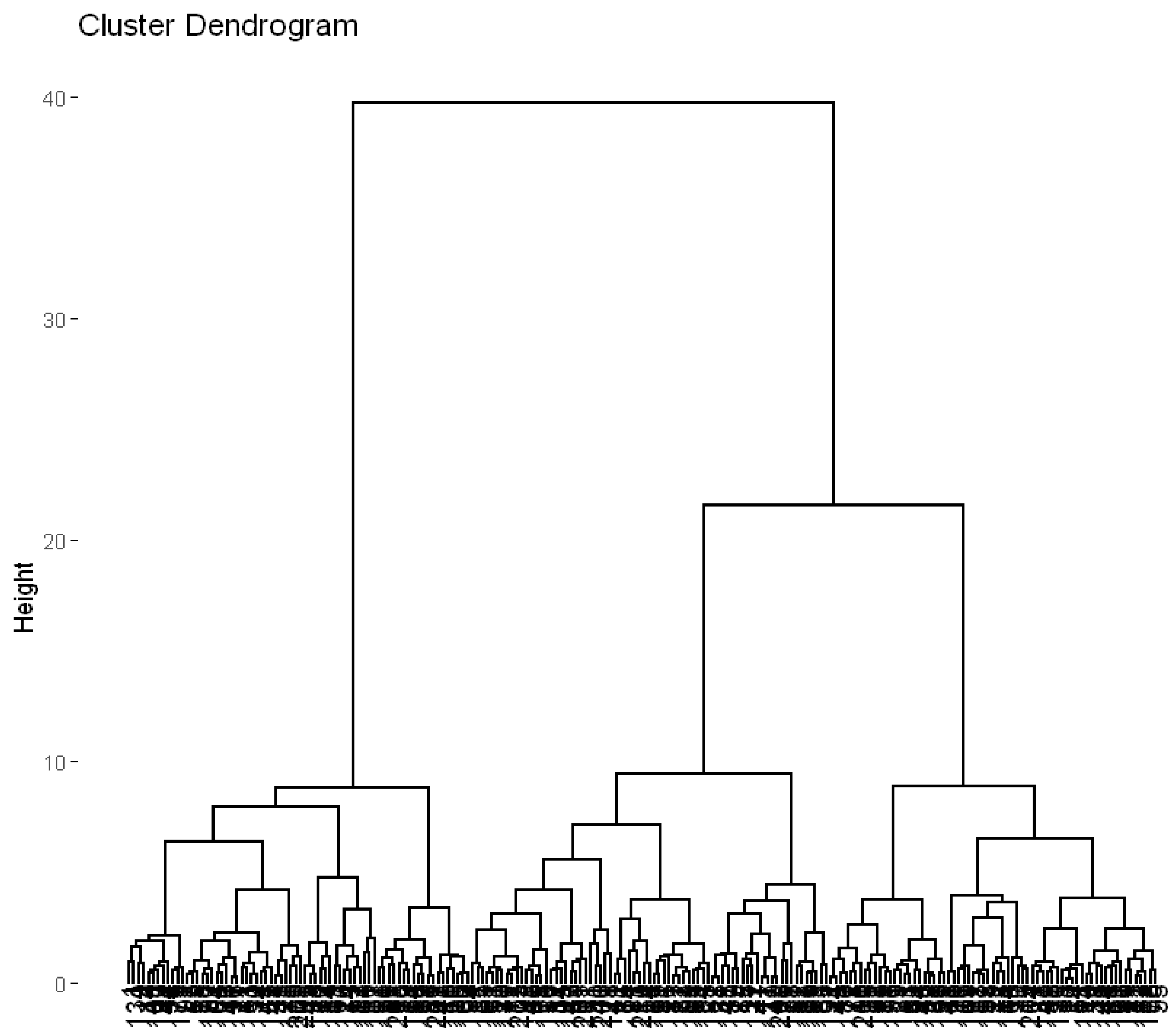
1.3. Hierarchical Clustering

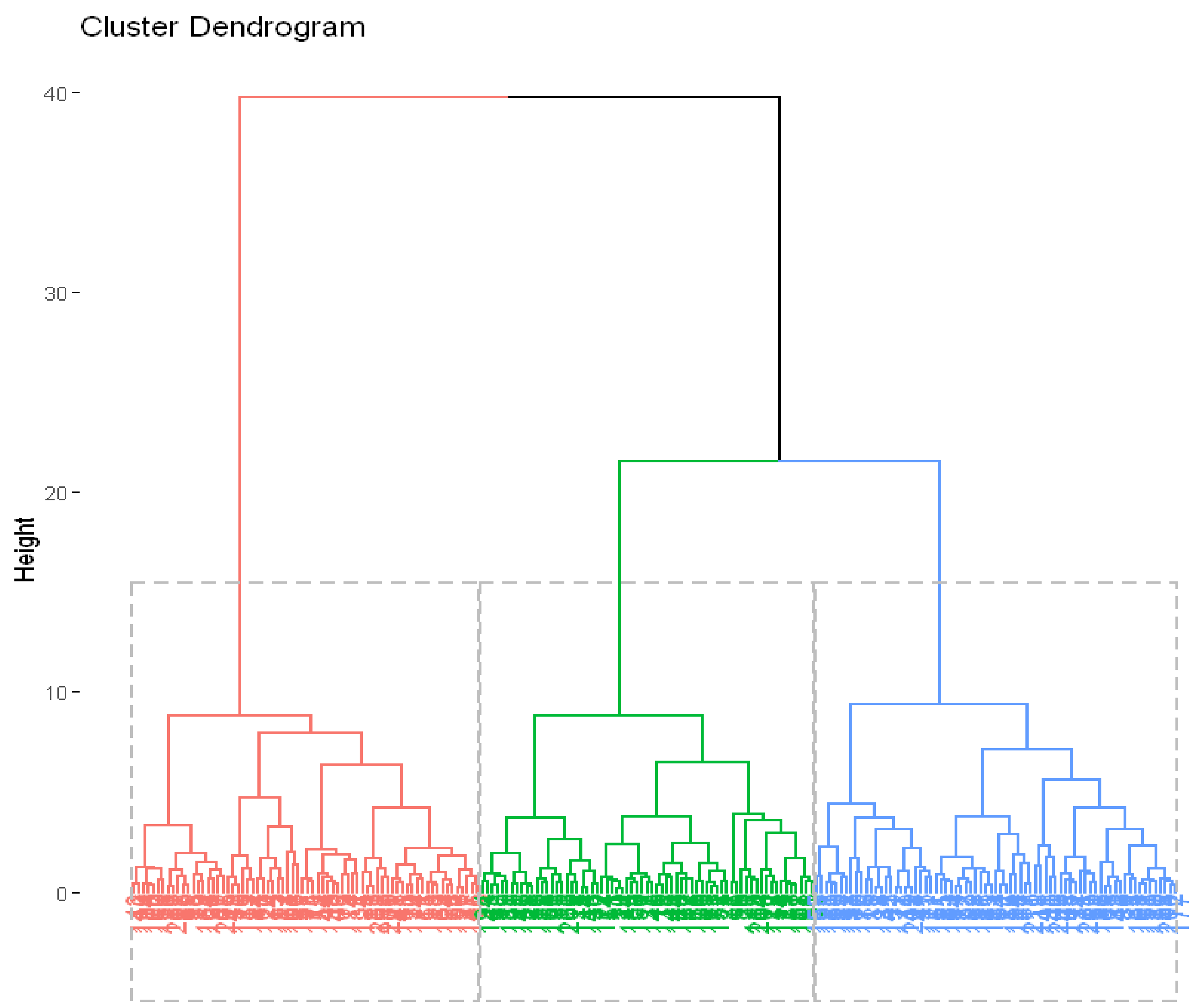
Hierarchical Clustering could be performed on different method for clustering of distance to have a better accuracy which was calculated through this code

```
#1. Which Method to Use
(Complete <- agnes(R_Data, method = "complete"))$ac
(Average <- agnes(R_Data, method = "average"))$ac
(Ward <- agnes(R_Data, method = "ward"))$ac
(Weighted <- agnes(R_Data, method = "weighted"))$ac

0.923151240030351
0.864954716959582
0.984626371916746
0.880694112728998
```

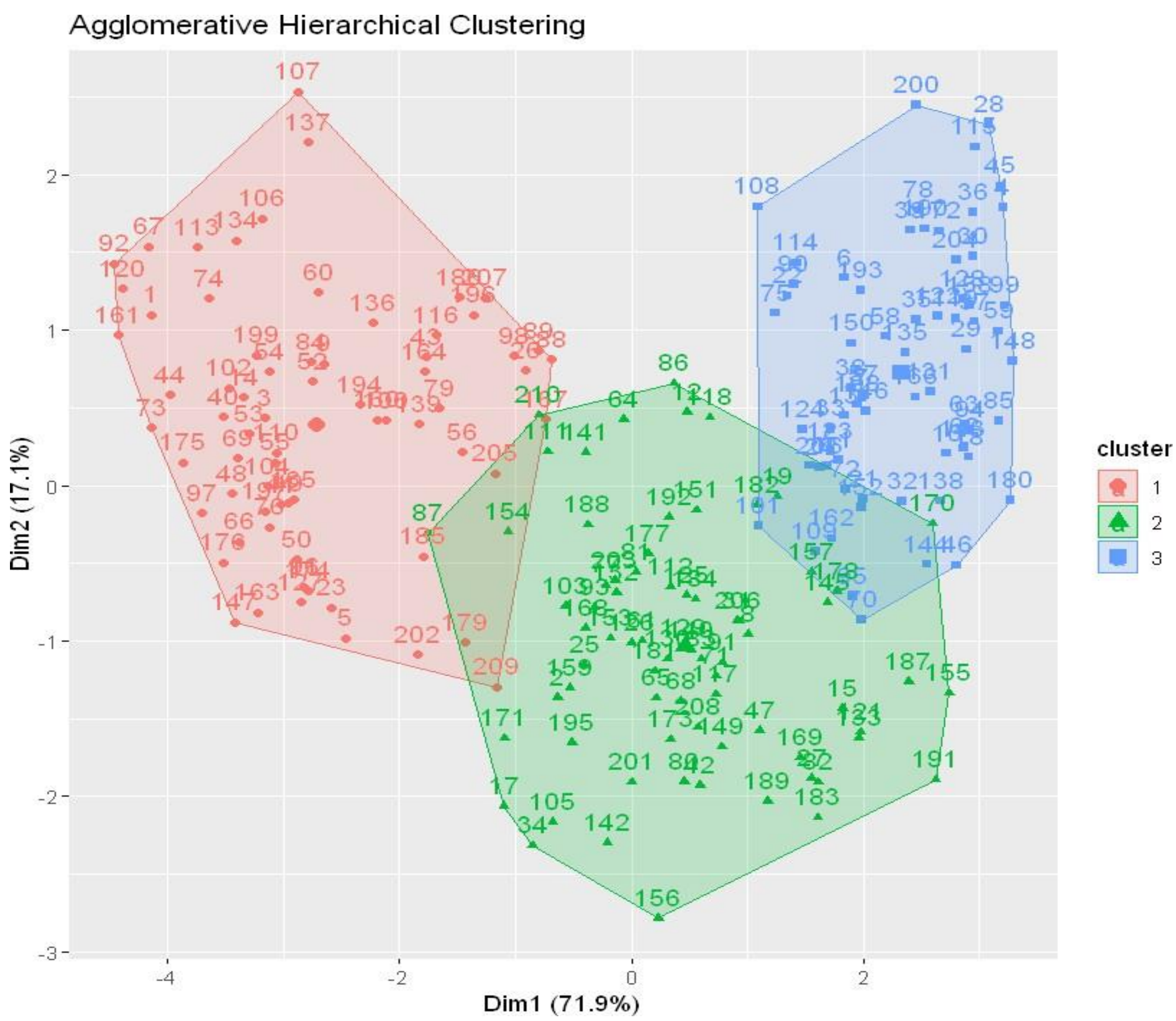
As we could see Ward have better accuracy we went for Ward method had accuracy of 0.98 we went with Ward method.

Agglomerative Hierarchical Clustering (AGNES)





As we could see large 3 groups in the cluster we would cut the tree on the value and the corresponding graph is as follows.



Aggregate of the cluster

A tibble: 3 × 8(Average values)

sub_grp	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping
<int>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	18371.43	1614.543	0.884400	6158.17	36846.29	363.915	6017.371
2	14199.04	1423.356	0.879190	5478.23	32264.52	261.218	5086.178
3	11872.39	1325.701	0.848071	5238.94	28485.37	494.943	5122.209

Silhouette

cluster	size	ave.sil.width
1	70	0.51
2	73	0.27
3	67	0.53

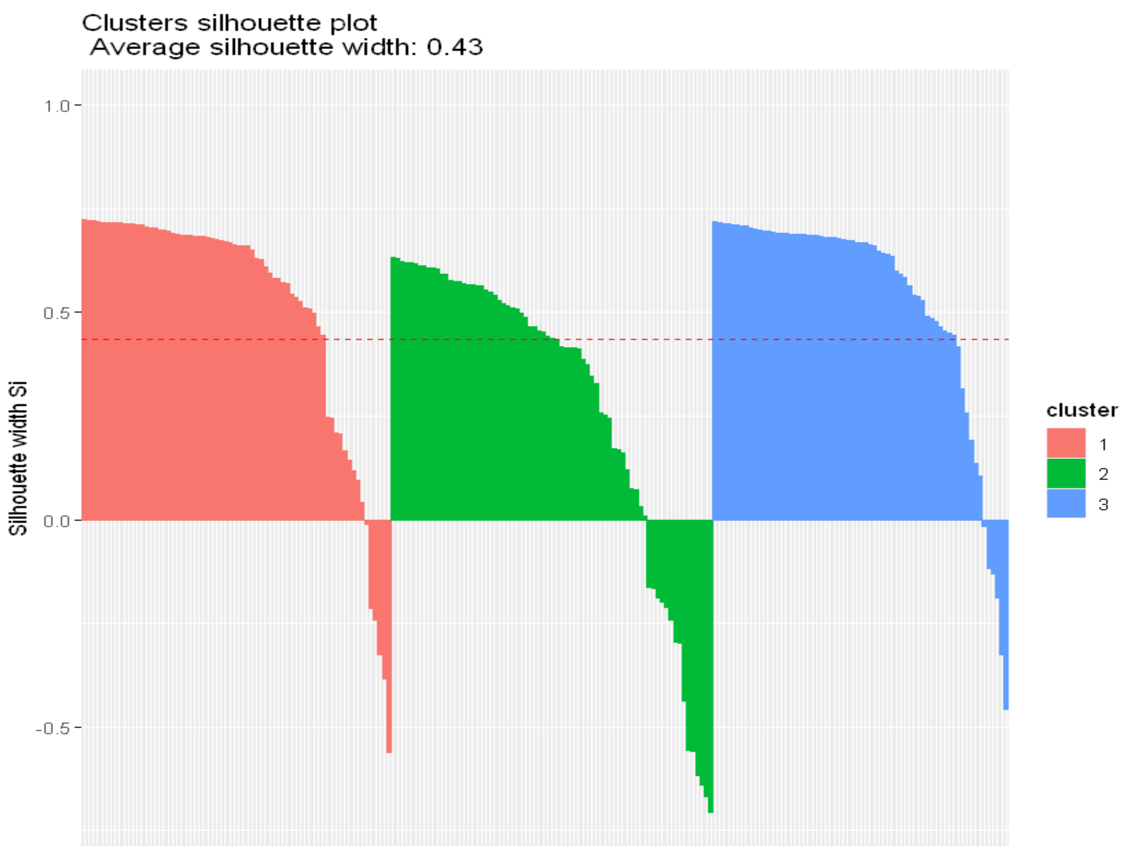
```
Silhouette of 210 units in 3 clusters from silhouette.default(x = Data_With_
AGNES$sub_grp, dist = dist(Data_With_AGNES)) :
 Cluster sizes and average silhouette widths:
       70        73        67
0.5087389 0.2744064 0.5300087
Individual silhouette widths:
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
-0.7065  0.3325  0.5732  0.4341  0.6832  0.7235
```

Negative value showcase the presence of the values in the wrong clusters

8	sub_grp [factor]	1. 1 2. 2 3. 3	6 (22.2%) 15 (55.6%) 6 (22.2%)	

Values having Negative silhouette width

Silhouette Plot

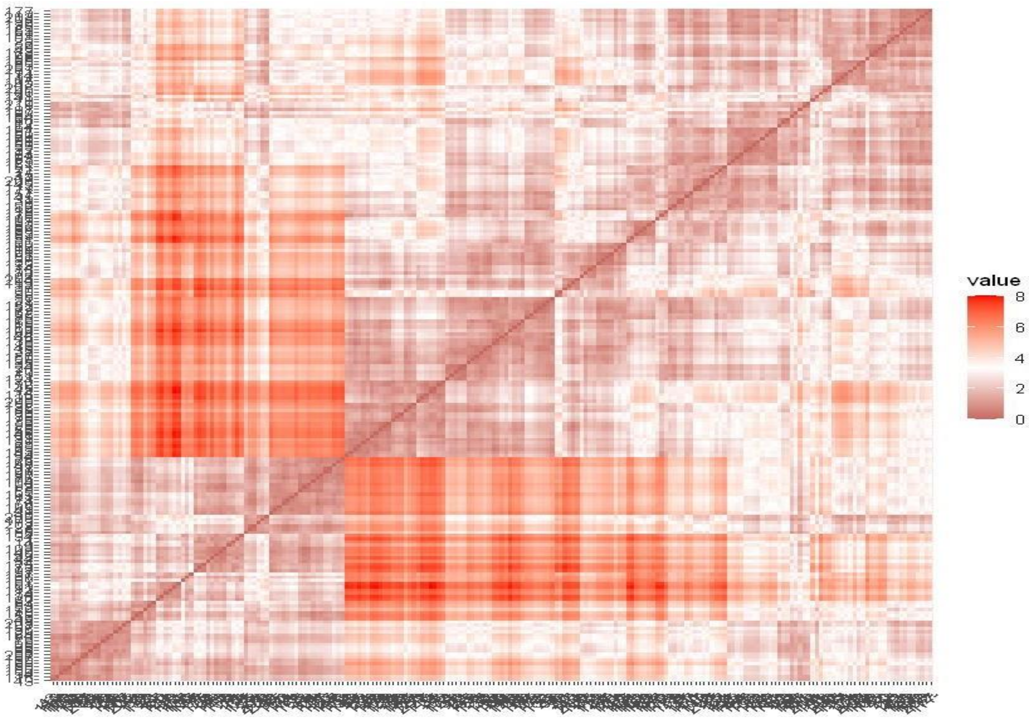


Q 4 : Apply K-Means clustering on scaled data and determine optimum clusters.

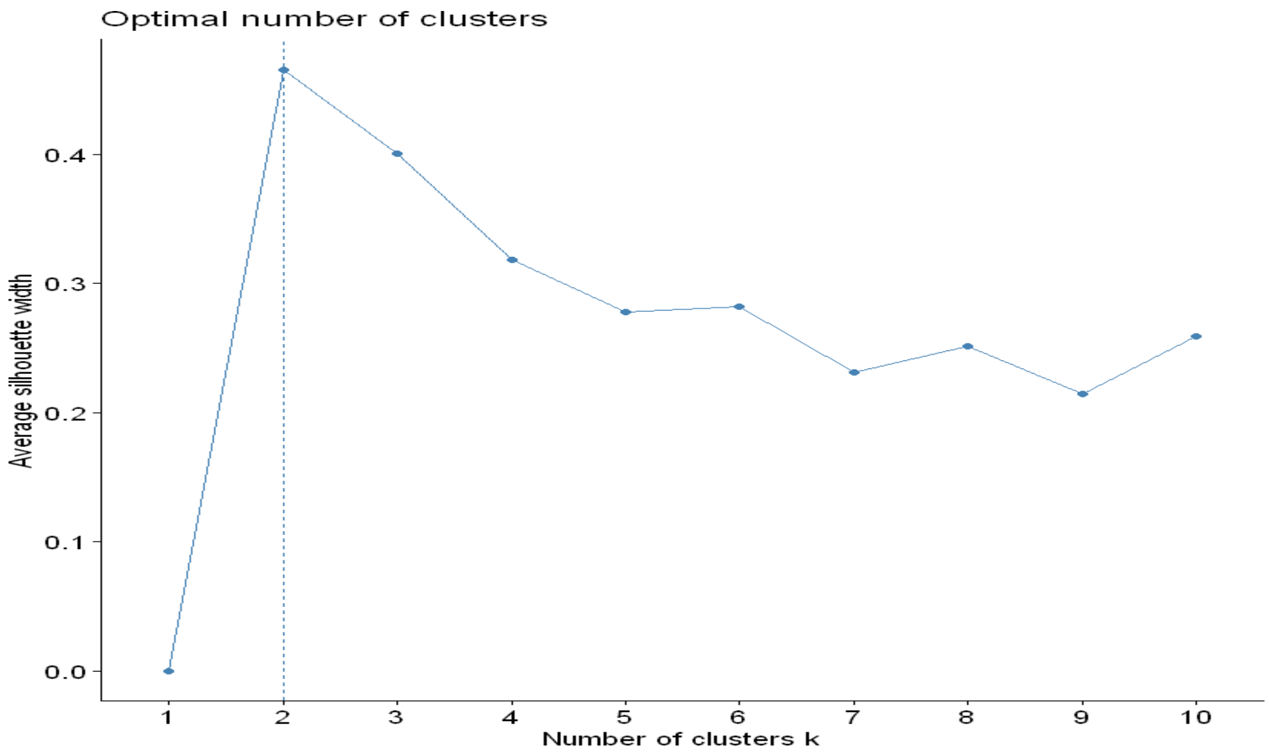
1.4. K-Means

Determining optimal number of clusters

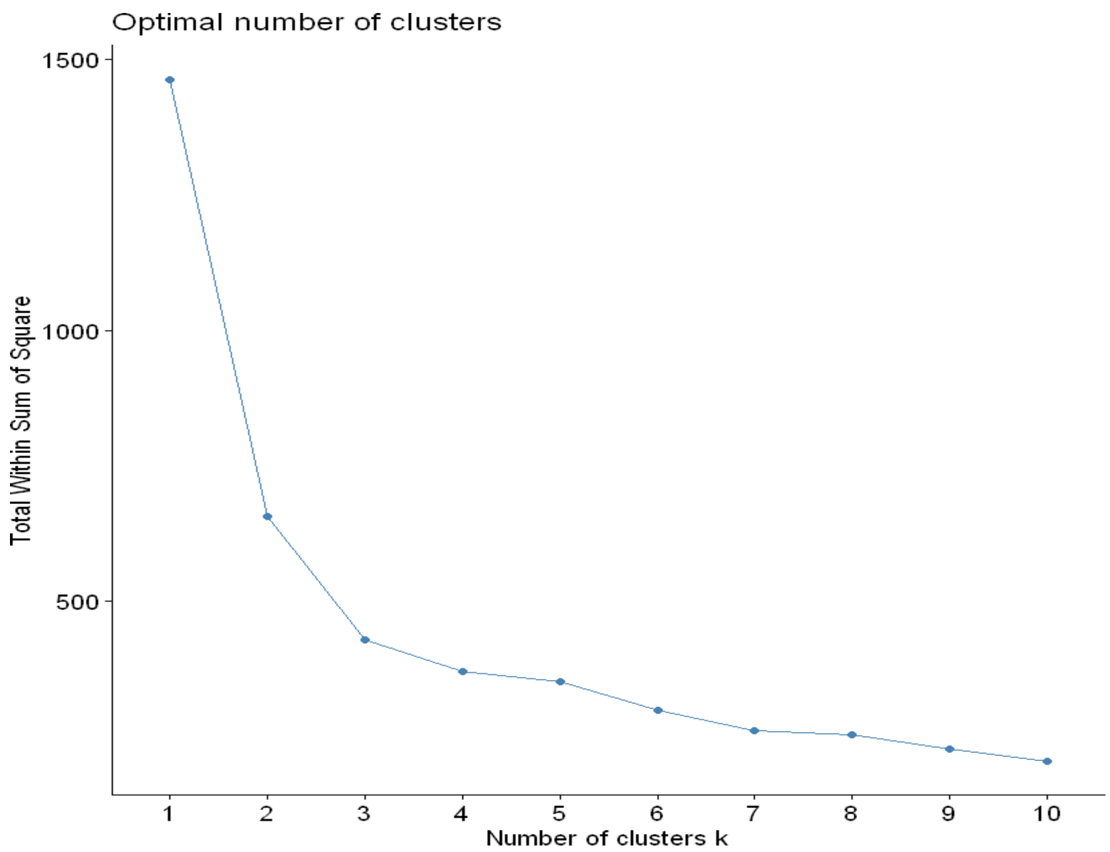
Distance Gradient method



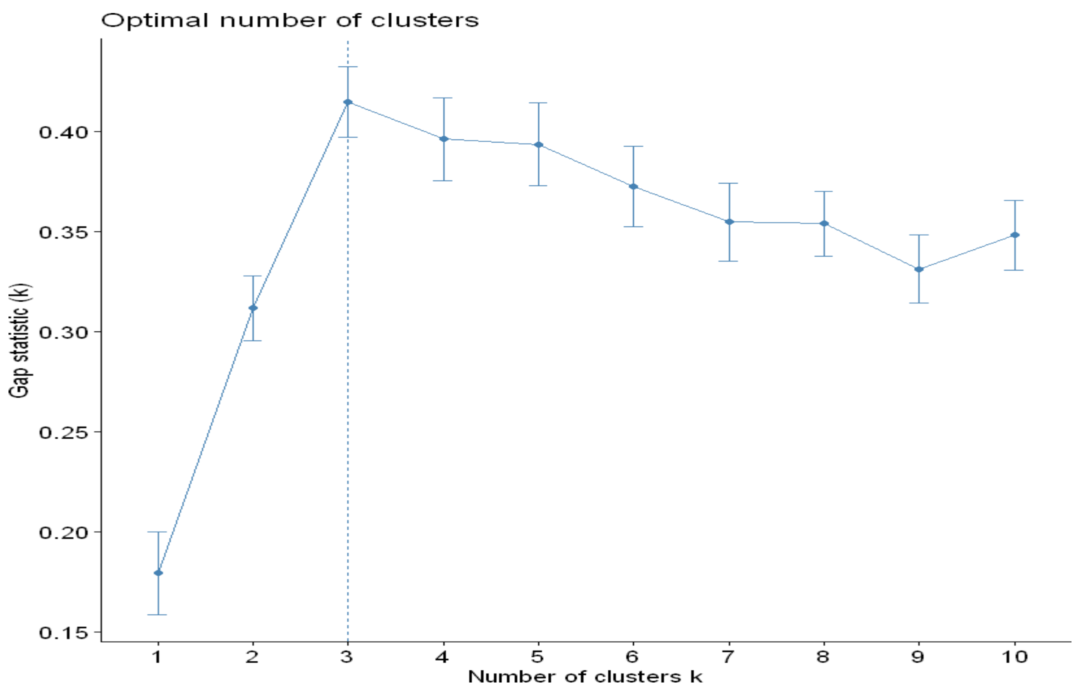
Silhouette Method



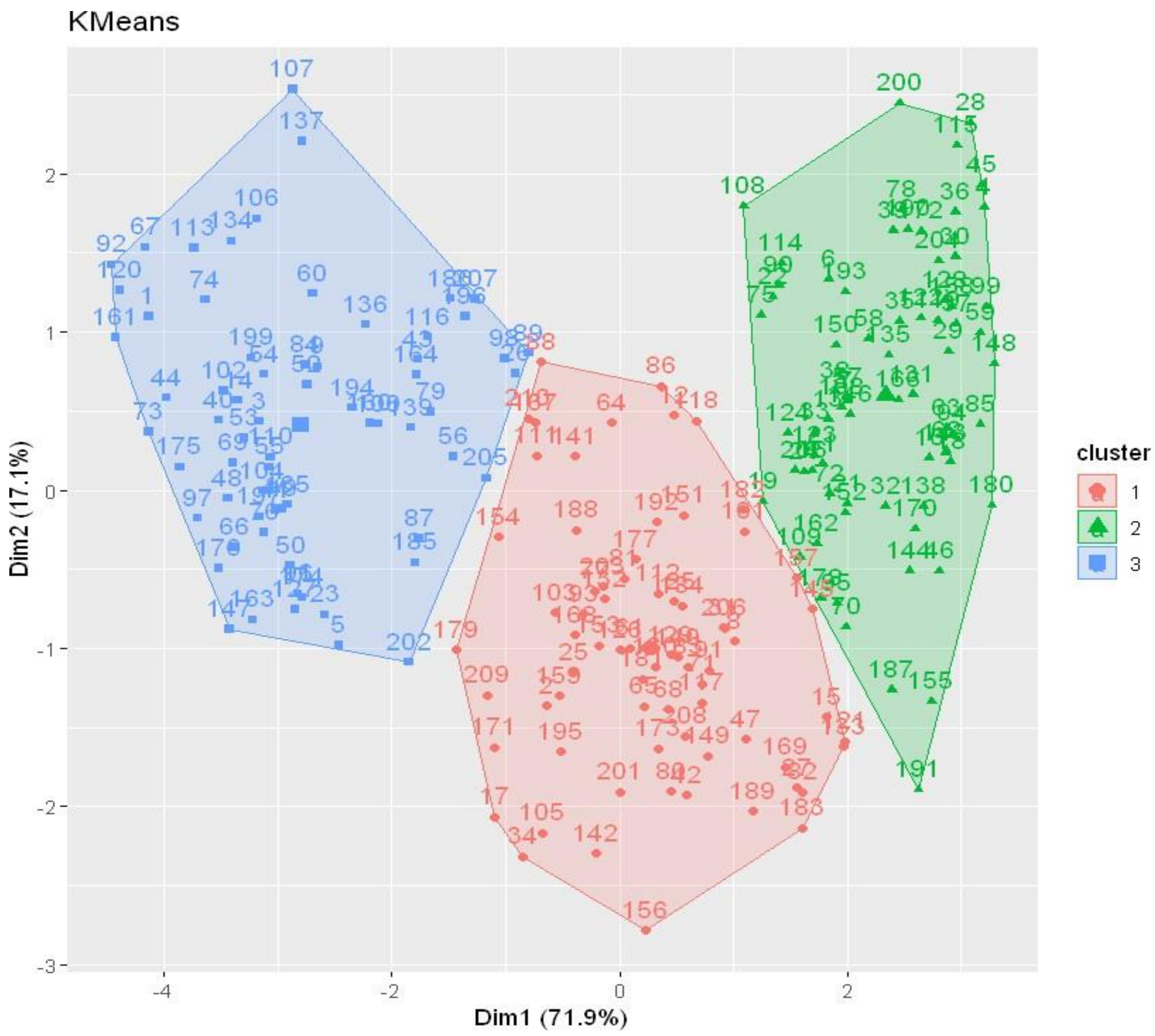
WSS method



Gap method



After referring each method the no of cluster was considered 3 for K Means



<u>Profiling of K-means</u>

A tibble: 3 × 8

Cluster	spending	advance_payments	probability_of_full_payment	current_balance	credit_limit	min_payment_amt	max_spent_in_single_shopping
2	14437.89	1433.775	0.8815972	5514.57	32592.25	270.7341	5120.803
3	11856.94	1324.778	0.8482528	5231.75	28495.42	474.2389	5101.722
1	18495.37	1620.343	0.8842104	6175.68	36975.37	363.2373	6041.701

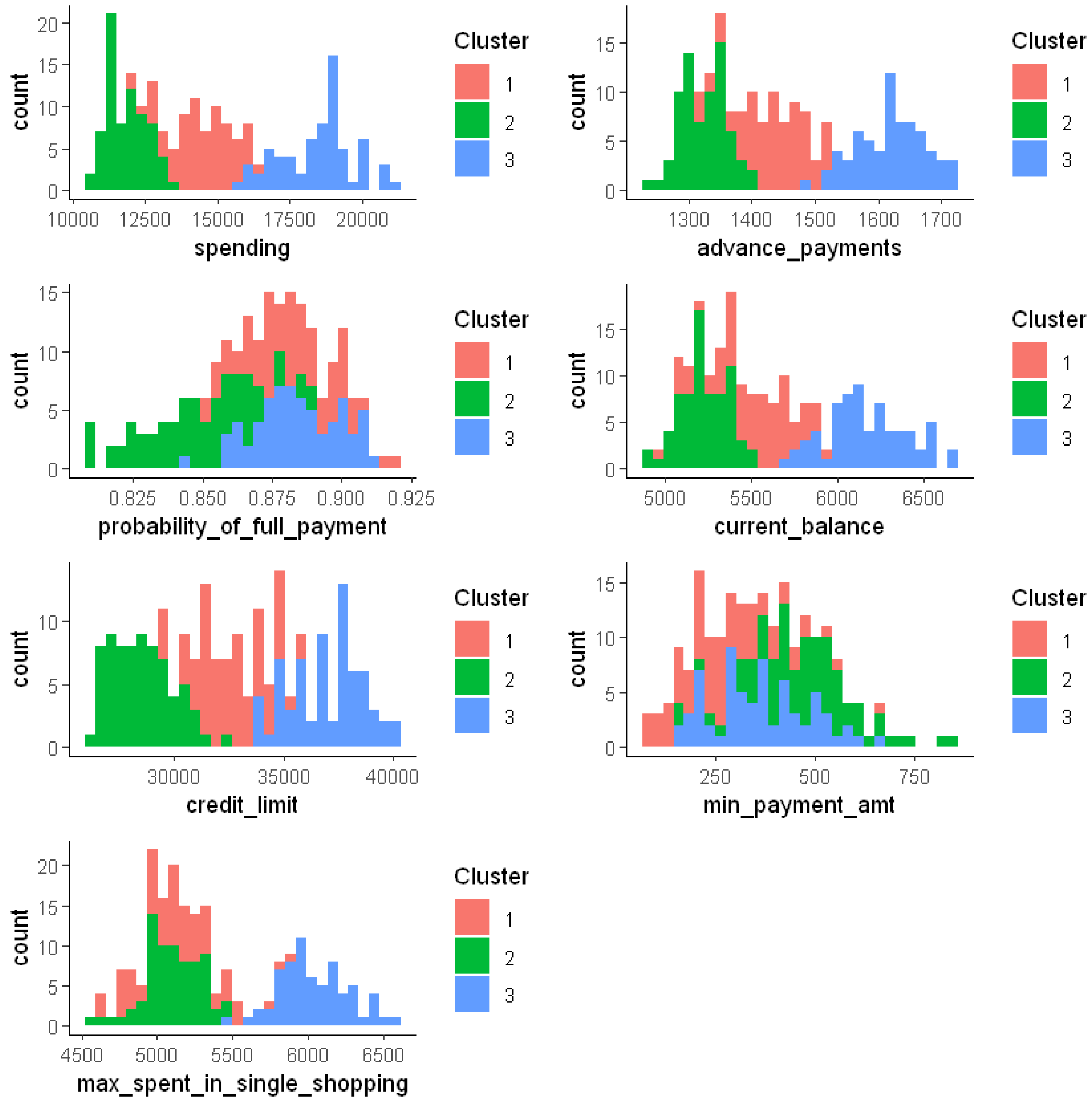
#Q 6 Describe cluster profiles for the clusters defined. Recommend different promotional strategies for different clusters

As we could clearly see through K- Means the clustering

- Spending group (First group): This group has large spending value. As they have a larger current balance their tendency to spend is lot and they pay in advance for their shopping and have larger probability of complete payment.
- Saving group(Second group): This group are middle spenders and average account balance but the reason the group was named saving group because of the min and max spent payment as the min payment is the least and maximum payment is near about third group
- Least spenders (Third group): Having less current balance with least spending habits.The only exception is the min payment amount is more than the earlier 2 groups.

But if we see Hierarchical clustering the cluster are grouped in a order of increasing values resulting in the overlapping of cluster. So the best clustering is provided by K- Means.

Comparision of Diffferent Cluster:



Promotional Strategies:

- Prodigal (Cluster 1): This group has large monthly spending as well as they spend large amount in single go. So the promotional strategies of the bank should be to have them platinum card for their spending and try to make them invest on long term plans ,investments.

- Cost-effective (Cluster 2): This group are average spenders but the reason the group was named saving group because of the min and max spent payment as the min payment is the least and maximum payment is near about third group. Approach them for different saving plans and for mutual investment

- Least spenders (Cluster 3): These group have least spending habits and less account balance which lead them to have less probability of repayment and pay high minimum payment amount. Approach them for loans and provide them with shopping coupons to make them spend money.

1.5. Conclusion

Using K-means and Hierarchical clustering we were able to have market segmentation and differ promotional strategies which made the decision making easier and data classification for further prediction

2. Problem 2

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. Exploring of Data

- Target: Claim Status (Claimed)
- Code of tour firm (Agency_Code)
- Type of tour insurance firms (Type)
- Distribution channel of tour insurance agencies (Channel)
- Name of the tour insurance products (Product)
- Duration of the tour (Duration)
- Destination of the tour (Destination)

Reading the data
Insurance <- read.csv(file.choose(),header = TRUE)

Snippet Data

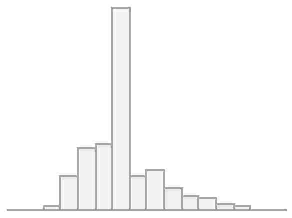
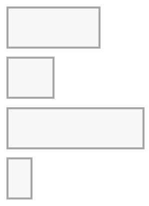
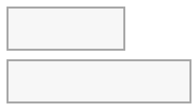
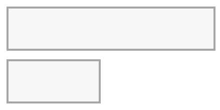
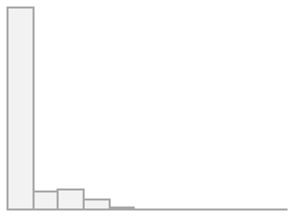
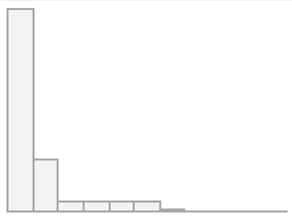
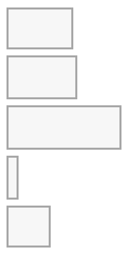
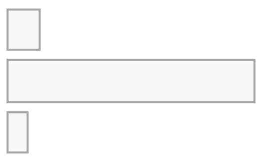
	Age	Agency_Code	Type	Claimed	Commision	Channel	Duration	Sales	Product.Name	Destination
1	48	C2B	Airlines	No	0.70	Online	7	2.51	Customised Plan	ASIA
2	36	EPX	Travel Agency	No	0.00	Online	34	20.00	Customised Plan	ASIA
3	39	CWT	Travel Agency	No	5.94	Online	3	9.90	Customised Plan	Americas
4	36	EPX	Travel Agency	No	0.00	Online	4	26.00	Cancellation Plan	ASIA
5	33	JZI	Airlines	No	6.30	Online	53	18.00	Bronze Plan	ASIA
6	45	JZI	Airlines	Yes	15.75	Online	8	45.00	Bronze Plan	ASIA

Data Frame Summary

Insurance

Dimensions: 3000 x 10

Duplicates: 139

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Valid	Missing
1	Age [integer]	Mean (sd) : 38.1 (10.5) min < med < max: 8 < 36 < 84 IQR (CV) : 10 (0.3)	70 distinct values		3000 (100%)	0 (0%)
2	Agency_Code [character]	1. C2B 2. CWT 3. EPX 4. JZI	924 (30.8%) 472 (15.7%) 1365 (45.5%) 239 (8.0%)		3000 (100%)	0 (0%)
3	Type [character]	1. Airlines 2. Travel Agency	1163 (38.8%) 1837 (61.2%)		3000 (100%)	0 (0%)
4	Claimed [character]	1. No 2. Yes	2076 (69.2%) 924 (30.8%)		3000 (100%)	0 (0%)
5	Commision [numeric]	Mean (sd) : 14.5 (25.5) min < med < max: 0 < 4.6 < 210.2 IQR (CV) : 17.2 (1.8)	324 distinct values		3000 (100%)	0 (0%)
6	Channel [character]	1. Offline 2. Online	46 (1.5%) 2954 (98.5%)		3000 (100%)	0 (0%)
7	Duration [integer]	Mean (sd) : 70 (134.1) min < med < max: -1 < 26.5 < 4580 IQR (CV) : 52 (1.9)	257 distinct values		3000 (100%)	0 (0%)
8	Sales [numeric]	Mean (sd) : 60.2 (70.7) min < med < max: 0 < 33 < 539 IQR (CV) : 49 (1.2)	380 distinct values		3000 (100%)	0 (0%)
9	Product.Name [character]	1. Bronze Plan 2. Cancellation Plan 3. Customised Plan 4. Gold Plan 5. Silver Plan	650 (21.7%) 678 (22.6%) 1136 (37.9%) 109 (3.6%) 427 (14.2%)		3000 (100%)	0 (0%)
10	Destination [character]	1. Americas 2. ASIA 3. EUROPE	320 (10.7%) 2465 (82.2%) 215 (7.2%)		3000 (100%)	0 (0%)

0 Missing Values and except Duration Commission and Age rest of the variable are character.

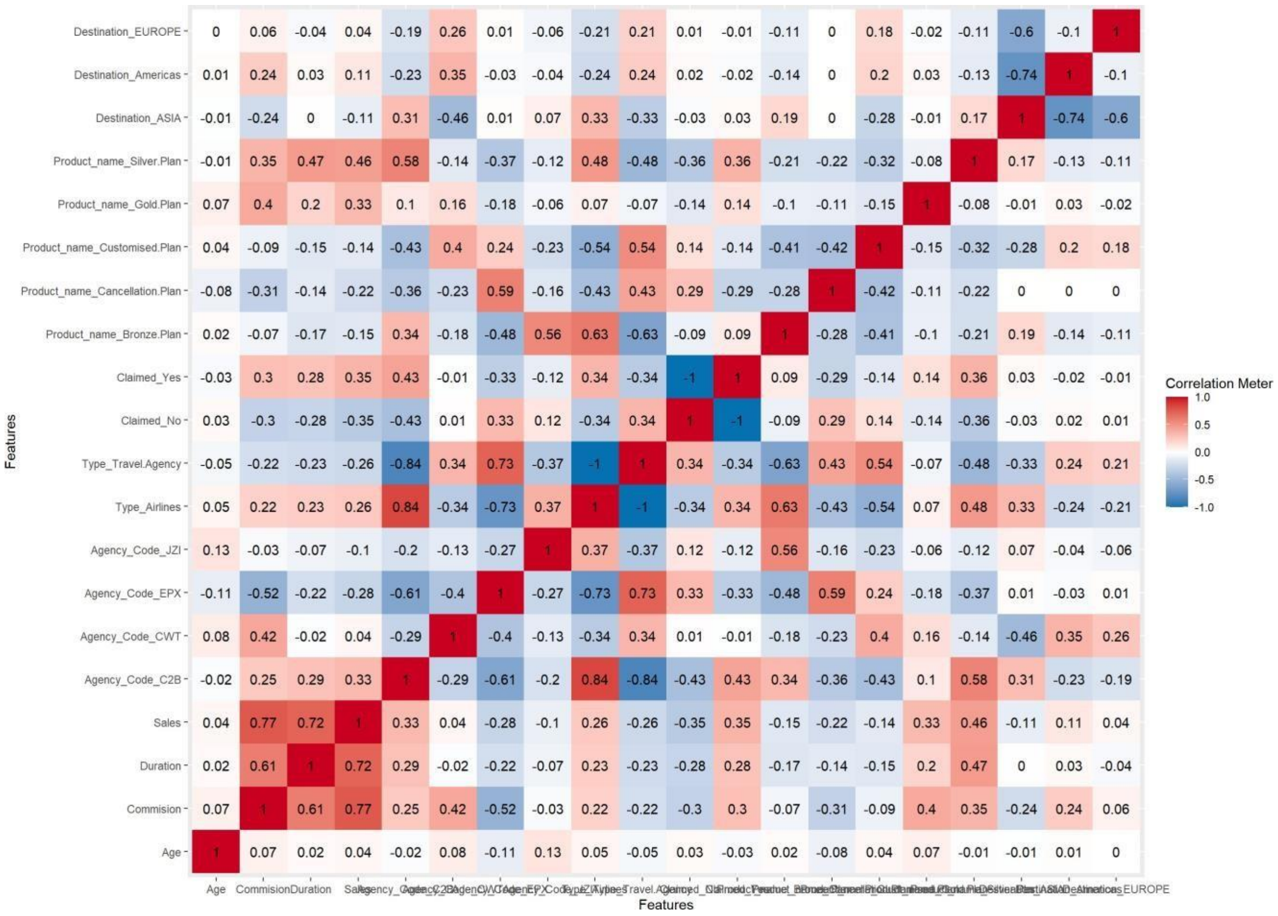
*Outlier

Converting them to factors, changing names and removing outliers

```
names(Insurance)[names(Insurance) == "Product.Name"] <- "Product_name"
Insurance$Agency_Code=as.factor(Insurance$Agency_Code)
Insurance$Claimed=as.factor(Insurance$Claimed)
Insurance$Channel=as.factor(Insurance$Channel)
Insurance$Product_name=as.factor(Insurance$Product_name)
Insurance$Destination=as.factor(Insurance$Destination)
Insurance=Insurance[-c(which.max(Insurance$Duration),which.min(Insurance$Duration)),]
```

The value in Duration have 4580,-1 as outlier removing those values. Removing Channel as only 46 are of offline.

Corrrelation



2.2. Data Split

```
set.seed(1353)
Ins_split <- initial_split(Insurance,prop=0.7,strata ="Claimed")
train_data <- training(Ins_split)
test_data <- testing(Ins_split)
```

Splitting the data in 70 and 30 proportion.

Number of Claimed in Insurance data set

No	Yes
2074	924

Number in test data

No	Yes
1452	647

Number in train data

No	Yes
622	277

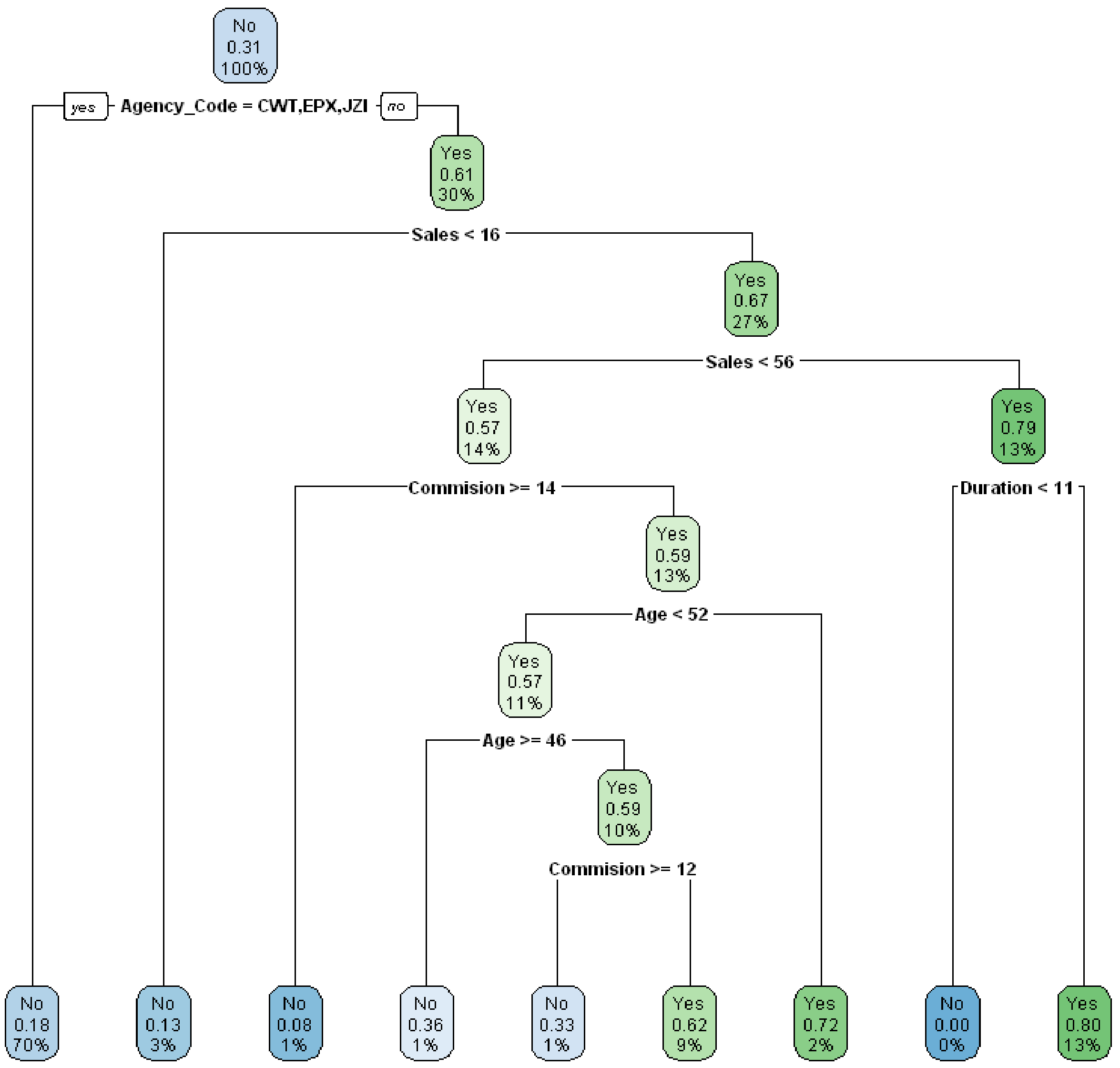
CART Model

```
# CART Model ------------------------------------------------------------
CART.ctrl <- rpart.control(
   minsplit = 9,
   minbucket = 3,
   cp = 0,
   xval = 10
)
CART <- rpart(formula = Claimed~.,
          data = train_data,
          method = 'class',
          control = CART.ctrl)
print(CART)
rpart.plot(CART)
printcp(CART)
```

The CP value from the table comes out to be 0.00412159

	CP	nsplit	rel error	xerror	xstd
1	0.22102009	0	1.00000	1.00000	0.032698
2	0.08191654	1	0.77898	0.77898	0.030247
3	0.00850077	2	0.69706	0.69706	0.029084
4	0.00772798	4	0.68006	0.68624	0.028919
5	0.00412159	8	0.64915	0.70015	0.029131

Prune Tree



Prediction for Train data

Confusion Matrix

```
Predicted
Actual    No  Yes
No  1279  173
Yes  267  380
```

Confusion Matrix and Statistics

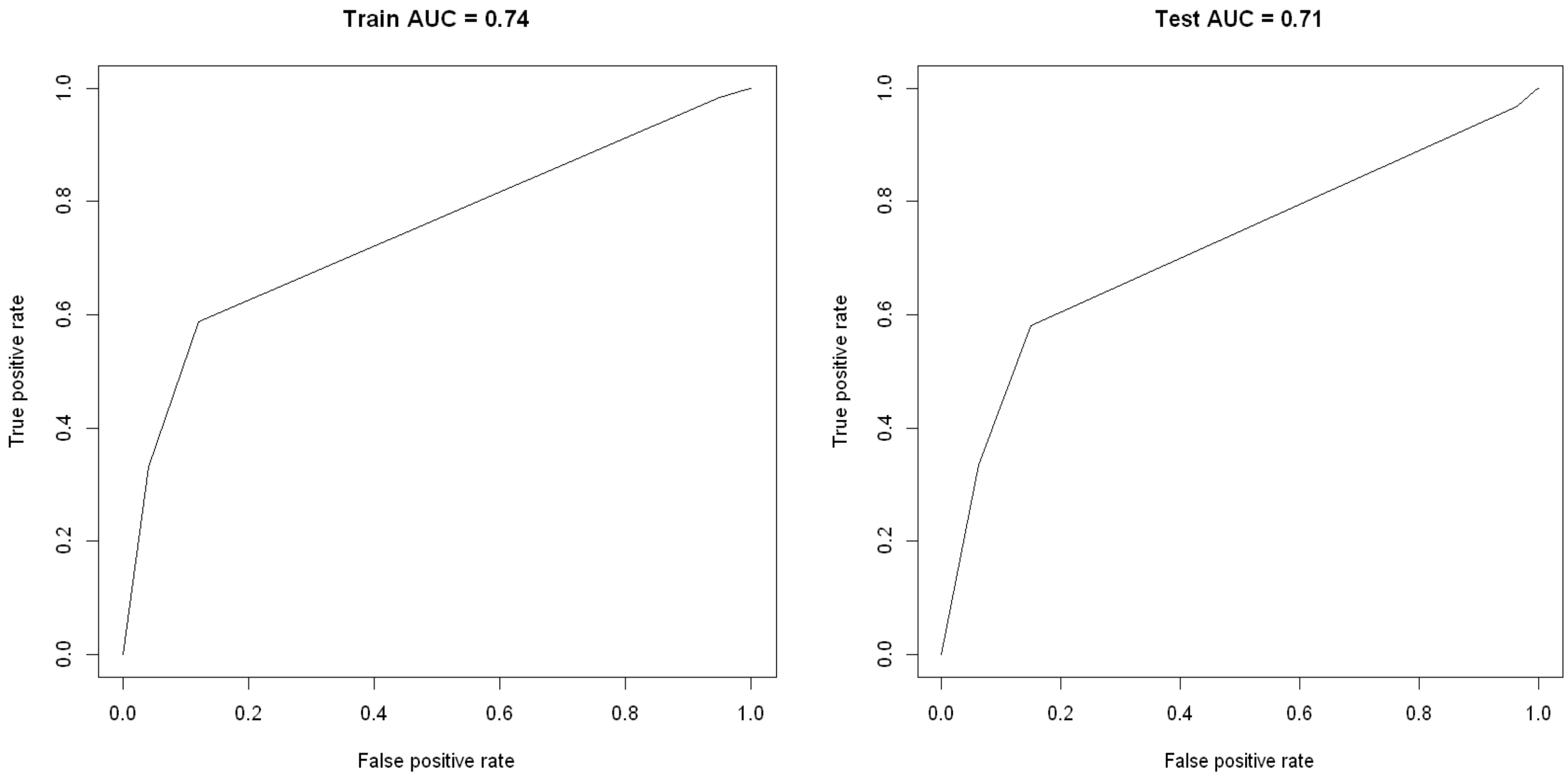
```
Accuracy : 0.7904
95% CI : (0.7723, 0.8076)
No Information Rate : 0.6918
P-Value [Acc > NIR] : < 2.2e-16

Kappa : 0.4878

Mcnemar's Test P-Value : 9.267e-06

Sensitivity : 0.5873
Specificity : 0.8809
Pos Pred Value : 0.6872
Neg Pred Value : 0.8273
Precision : 0.6872
Recall : 0.5873
F1 : 0.6333
Prevalence : 0.3082
Detection Rate : 0.1810
Detection Prevalence : 0.2635
Balanced Accuracy : 0.7341

'Positive' Class : Yes
```

Prediction for test data

```
Confusion Matrix and Statistics

          Reference
Prediction  No Yes
       No  529 116
       Yes  93 161

               Accuracy : 0.7675
                 95% CI : (0.7385, 0.7948)
    No Information Rate : 0.6919
    P-Value [Acc > NIR] : 2.871e-07

                  Kappa : 0.4419

 Mcnemar's Test P-Value : 0.1281

            Sensitivity : 0.5812
            Specificity : 0.8505
         Pos Pred Value : 0.6339
         Neg Pred Value : 0.8202
              Precision : 0.6339
                 Recall : 0.5812
                     F1 : 0.6064
             Prevalence : 0.3081
         Detection Rate : 0.1791
   Detection Prevalence : 0.2825
      Balanced Accuracy : 0.7159

       'Positive' Class : Yes
```

Sensitivity decreased because of imbalance of data set.

AUC Graph



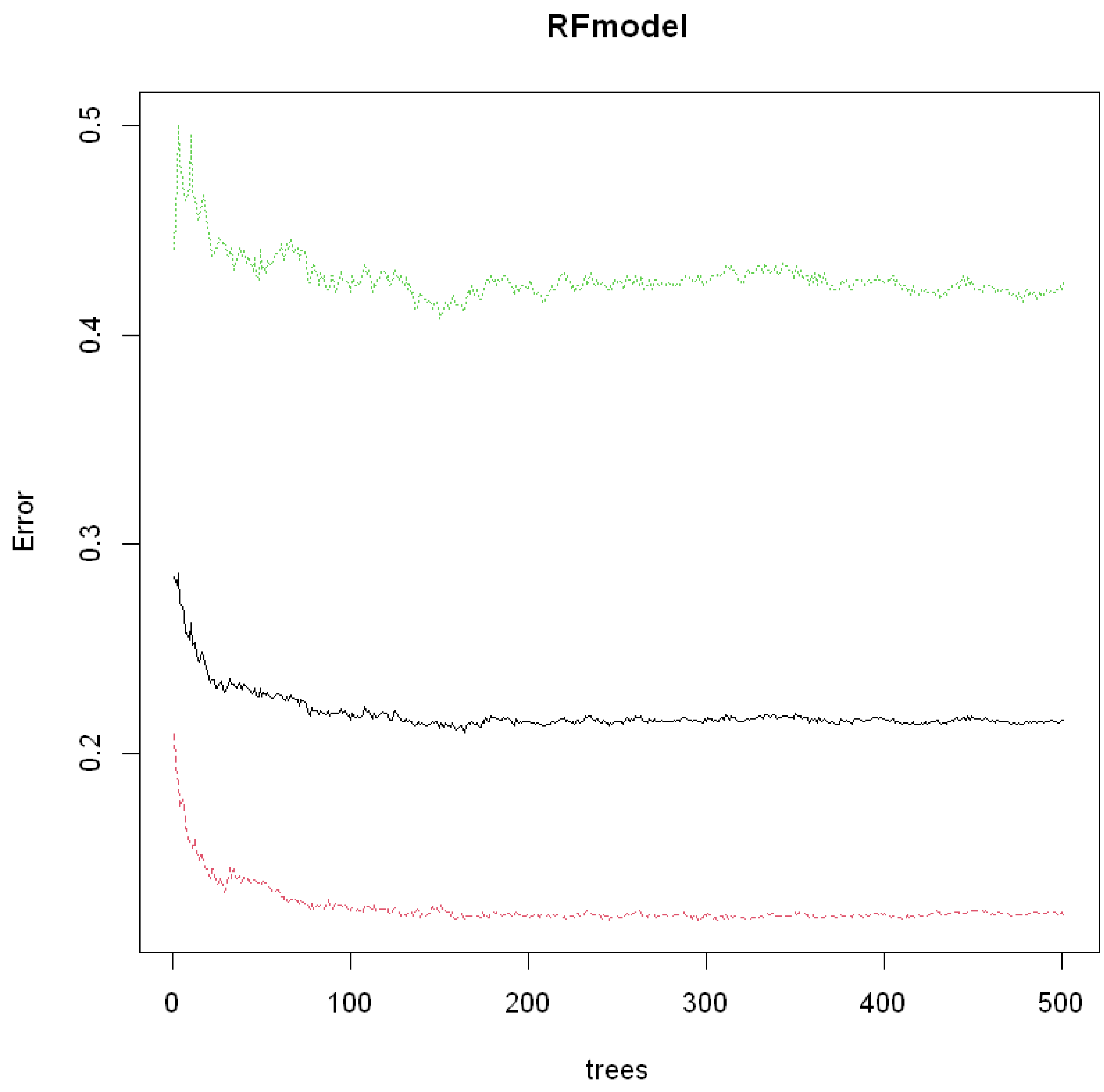
Variable Importance

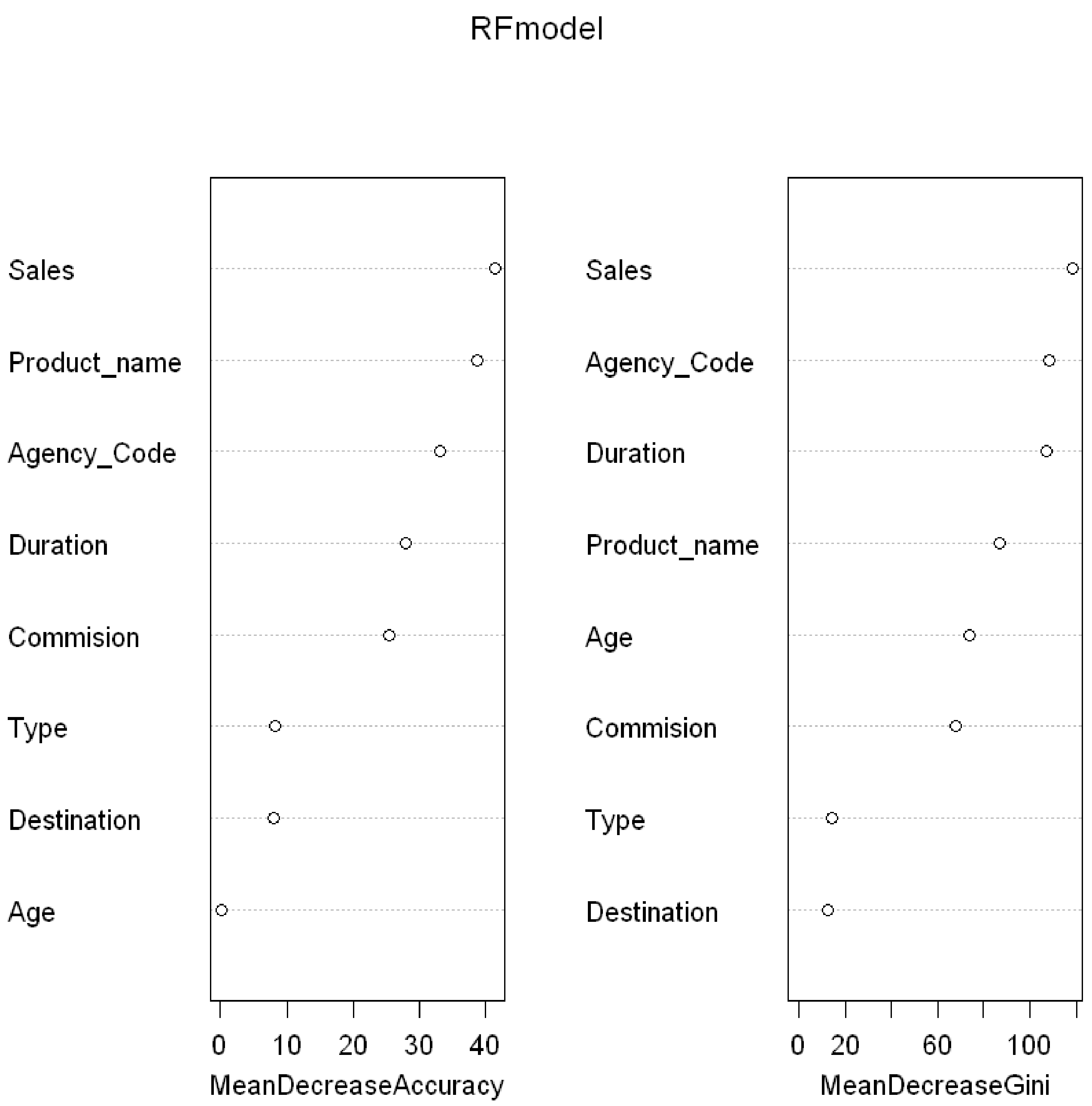
Variable	Importance
Agency_Code	100.000000
Type	75.510204
Product_name	69.817142
Sales	51.150188
Commision	50.272592
Duration	29.140684
Age	3.145081

Random Forest

A matrix: 8 × 4 of type dbl

	No	Yes	MeanDecreaseAccuracy	MeanDecreaseGini
Age	9.6494370	-11.041922	0.1004471	73.81515
Agency_Code	4.1884027	39.447334	33.0059375	108.05316
Type	-0.3009871	8.729256	8.1905528	14.36812
Commision	1.6641325	23.718661	25.4202877	68.01690
Duration	-1.1963366	30.748921	27.8810443	107.36589
Sales	-7.4735207	44.544592	41.2551195	118.19953
Product_name	16.0250782	28.089218	38.6818849	86.65685
Destination	2.8310476	7.110413	7.8970378	12.34429



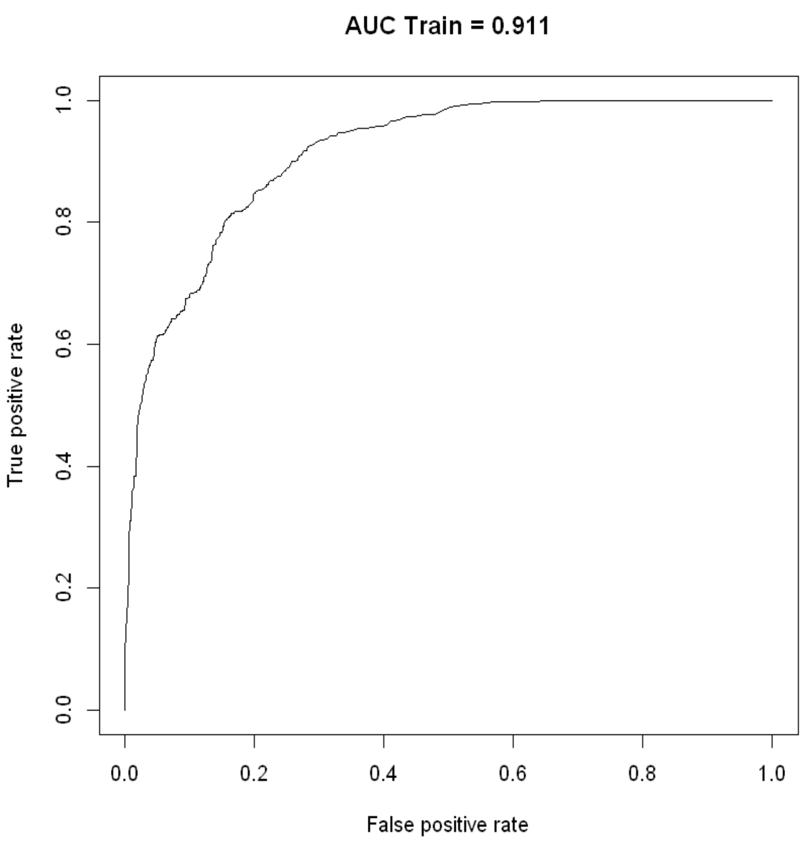


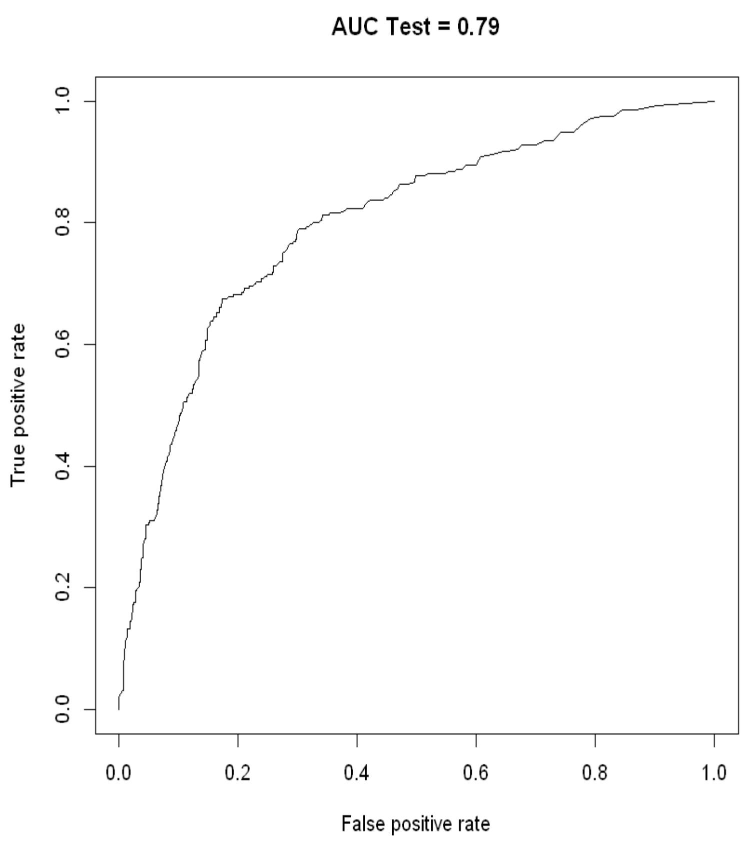
After tuning of Model

```
Confusion Matrix and Statistics


        No  Yes
  No  1307  206
  Yes  145  441

               Accuracy : 0.8328
                 95% CI : (0.8161, 0.8485)
    No Information Rate : 0.6918
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.5974

 Mcnemar's Test P-Value : 0.001362

            Sensitivity : 0.6816
            Specificity : 0.9001
         Pos Pred Value : 0.7526
         Neg Pred Value : 0.8638
              Precision : 0.7526
                 Recall : 0.6816
                     F1 : 0.7153
             Prevalence : 0.3082
         Detection Rate : 0.2101
   Detection Prevalence : 0.2792
      Balanced Accuracy : 0.7909

       'Positive' Class : Yes
```

```
Confusion Matrix and Statistics


      No Yes
 No  529 103
 Yes  93 174

               Accuracy : 0.782
                 95% CI : (0.7535, 0.8086)
    No Information Rate : 0.6919
    P-Value [Acc > NIR] : 9.901e-10

                  Kappa : 0.4835

 Mcnemar's Test P-Value : 0.5203

            Sensitivity : 0.6282
            Specificity : 0.8505
         Pos Pred Value : 0.6517
         Neg Pred Value : 0.8370
              Precision : 0.6517
                 Recall : 0.6282
                     F1 : 0.6397
             Prevalence : 0.3081
         Detection Rate : 0.1935
   Detection Prevalence : 0.2970
      Balanced Accuracy : 0.7393

       'Positive' Class : Yes
```

AUC Graph(For train data)



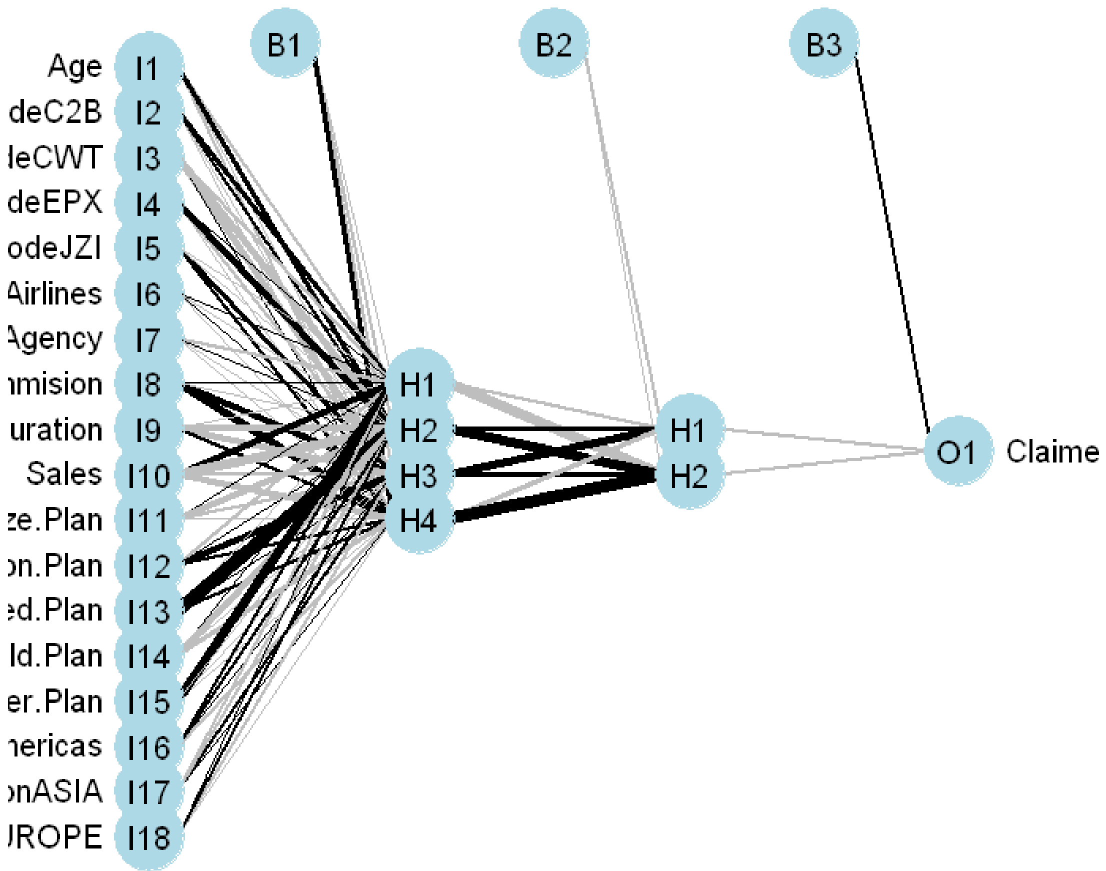


ANN Model

Dummy Data

```
data.frame':       3000 obs. of  21 variables:
 $ Age                          : num  0.947 -0.1998 0.0869 -0.1998 -0.4865 ...
 $ Agency.CodeC2B               : num  1.499 -0.667 -0.667 -0.667 -0.667 ...
 $ Agency.CodeCWT               : num  -0.432 -0.432 2.314 -0.432 -0.432 ...
 $ Agency.CodeEPX               : num  -0.914 1.094 -0.914 1.094 -0.914 ...
 $ Agency.CodeJZI               : num  -0.294 -0.294 -0.294 -0.294 3.398 ...
 $ TypeAirlines                 : num  1.257 -0.796 -0.796 -0.796 1.257 ...
 $ TypeTravel Agency            : num  -1.257 0.796 0.796 0.796 -1.257 ...
 $ Commision                    : num  -0.543 -0.57 -0.337 -0.57 -0.323 ...
 $ ChannelOffline               : num  -0.125 -0.125 -0.125 -0.125 -0.125 ...
 $ ChannelOnline                : num  0.125 0.125 0.125 0.125 0.125 ...
 $ Duration                     : num  -0.47 -0.269 -0.5 -0.492 -0.127 ...
 $ Sales                        : num  -0.816 -0.569 -0.712 -0.484 -0.597 ...
 $ Product.NameBronze Plan      : num  -0.526 -0.526 -0.526 -0.526 1.901 ...
 $ Product.NameCancellation Plan: num  -0.54 -0.54 -0.54 1.85 -0.54 ...
 $ Product.NameCustomised Plan  : num  1.281 1.281 1.281 -0.781 -0.781 ...
 $ Product.NameGold Plan        : num  -0.194 -0.194 -0.194 -0.194 -0.194 ...
 $ Product.NameSilver Plan      : num  -0.407 -0.407 -0.407 -0.407 -0.407 ...
 $ DestinationAmericas          : num  -0.345 -0.345 2.893 -0.345 -0.345 ...
 $ DestinationASIA              : num  0.466 0.466 -2.146 0.466 0.466 ...
 $ DestinationEUROPE            : num  -0.278 -0.278 -0.278 -0.278 -0.278 ...
 $ Claimed                      : num  0 0 0 0 0 1 0 0 0 0 ...
```

ANN Model



ANN

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 1221  168
         1  231  479

               Accuracy : 0.8099
                 95% CI : (0.7925, 0.8265)
    No Information Rate : 0.6918
    P-Value [Acc > NIR] : < 2e-16

                  Kappa : 0.566

 Mcnemar's Test P-Value : 0.00191

            Sensitivity : 0.7403
            Specificity : 0.8409
         Pos Pred Value : 0.6746
         Neg Pred Value : 0.8790
              Precision : 0.6746
                 Recall : 0.7403
                     F1 : 0.7060
             Prevalence : 0.3082
         Detection Rate : 0.2282
   Detection Prevalence : 0.3383
      Balanced Accuracy : 0.7906

       'Positive' Class : 1
```

For Test

```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 622 277
         1   0   0

               Accuracy : 0.6919
                 95% CI : (0.6605, 0.7219)
    No Information Rate : 0.6919
    P-Value [Acc > NIR] : 0.5162

                  Kappa : 0

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.0000
            Specificity : 1.0000
         Neg Pred Value : 0.6919
                 Recall : 0.0000
                     F1 :     NA
             Prevalence : 0.3081
         Detection Rate : 0.0000
   Detection Prevalence : 0.0000
      Balanced Accuracy : 0.5000

       'Positive' Class : 1
```

Conclusion:

Random Forest have a better prediction value for the data having a AUC of around 90% . As for ANN because of lack of data the prediction model is not suitable for the specific data The data completely depended on the product name and the customer preferred them in difference the commission or sales value.

Q5 : Inference: Basis on these predictions, what are the business insights and recommendations

As we could see from the following model the customer preference run around the customization plan have the highest claim of all suggesting the travel agency bending towards customer decided planning have higher claim.