PREDICTION OF ONLINE SHOPPERS PURCHASING INTENTION MODEL

Contents:

1. EXECUTIVE SUMMARY:

The E-Commerce industry is one of the world's major industries that must be constantly updated with cutting-edge technology to provide the best services to customers, with the goal of predicting online shoppers' purchasing decisions. Many people who visit ecommerce websites may not intend to buy anything. This could be because of several factors. However, we can determine whether a user is likely to purchase or not based on their activity on the ecommerce website. We used Google Analytics data from an ecommerce website to investigate the possibility of predicting customer purchase intent in this project. Machine Learning algorithms are used to create highly accurate prediction models. Ecommerce businesses can benefit greatly from the ability to predict customer purchase intent because it allows them to better understand the digital retail space.

1.1. PROJECT MOTIVATION:

Shopping dynamics are changing around the world as retail shopping continues to shift to E-commerce shopping. E-commerce is already a significant retail market. Customers frequently browse e-commerce site pages before placing orders or abandoning their browsing without making a purchase. This information can help businesses better cater to customer preferences and mutually benefit both the business and the customers by recommending products tailored to each customer and thus increasing sales for the businesses.

2. DATA SOURCE:

The dataset is open-source, and it is available to the public on the website UCI Machine Learning Repository.

2.1 SOURCE LINK:

https://archive.ics.uci.edu/ml/datasets/Online+Shoppers+Purchasing+Intention+Dataset

Number of Records: 12330

Number of Columns: 18

Size of the dataset: 1047 KB

2.2. DATA DESCRIPTION:

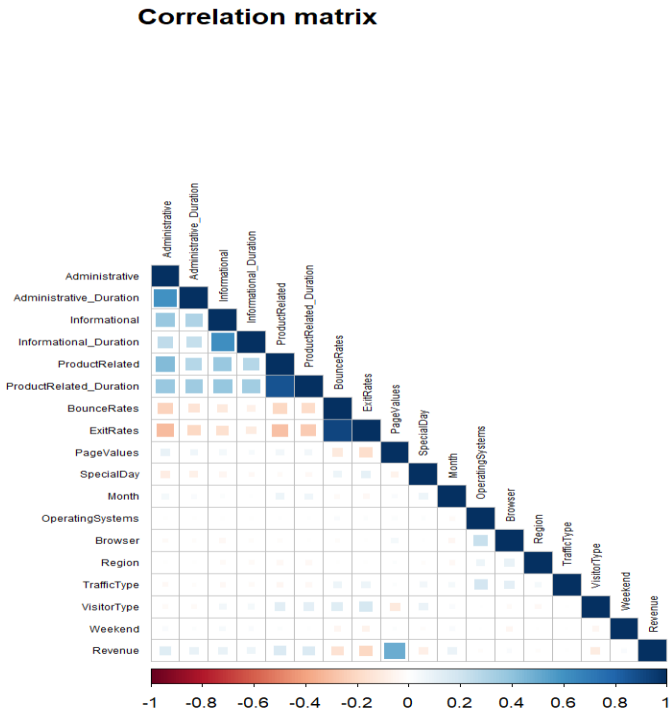
The description of each column is shown below.

S.NO	VARIABLES	DISCRIPTION
1	Administrative	Represent the number of different types of pages visited by the visitor in that session and total time spent in each of these page categories.
2	Administrative Duration	Represent the number of different types of pages visited by the visitor in that session and total time spent in each of these page categories.
3	Informational	Represents the detailed information' regarding products.
4	Informational Duration	Represent the number of different types of pages visited by the visitor in that session and total time spent in each of these page categories.
5	Product Related	Represent the different types of product details.
6	Product Related Duration	Represent the number of different types of pages visited by the visitor in that session and total time spent in each of these page categories.
7	Bounce Rate	Feature for a web page refers to the percentage of visitors who enter the site from that page and then leave without triggering any other requests to the analytics server during that session.
8	Exit Rate	Feature for a specific web page is calculated as for all pageviews to the page, the percentage that were the last in the session.
9	Page Value	Feature represents the average value for a web page that a user visited before completing an e-commerce transaction.
10	Special Day	Feature indicates the closeness of the site visiting time to a specific special day (e.g. Mother's Day, Valentine's Day) in which the sessions are more likely to be finalized with transaction.
11	Month	Value indicating whether the date of the visit is month of the year.
12	Operating Systems	A Boolean value indicating whether the date of the visit.
13	Browser	The Number of types of Customers visited.
14	Region	A Boolean value indicating whether the date of the visit is weekend, and month of the year.
15	Traffic Type	The Number of types of Customers visited.
16	Visitor Type	Feature indicates whether the visitor is a new visitor or returning.
17	Weekend	Indicates whether the day of the week that the session started falls on the weekend or not.

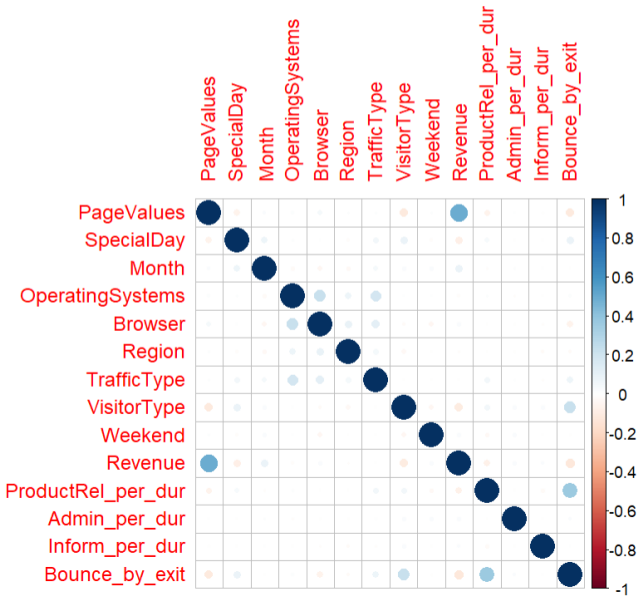
18	Revenue	The target “Revenue” demonstrates that the majority of customers failed to complete the purchasing

3. DATA PREPROCESSING & EXPLORATORY DATA ANALYSIS:

3.1. Correlation Plot (Before Modification)

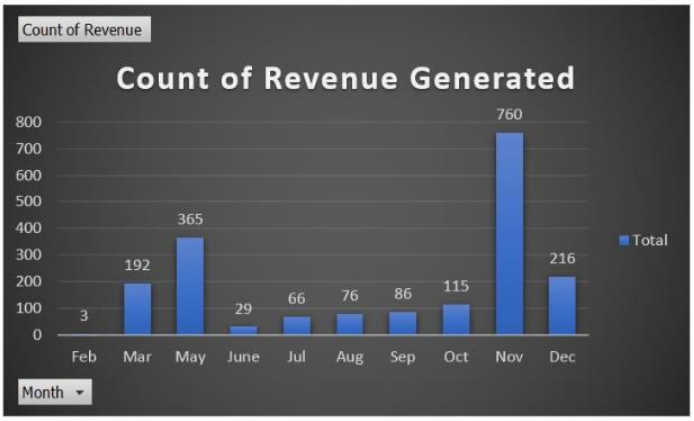


3.2. Correlation Plot (After Modification -cutoff 0.60)

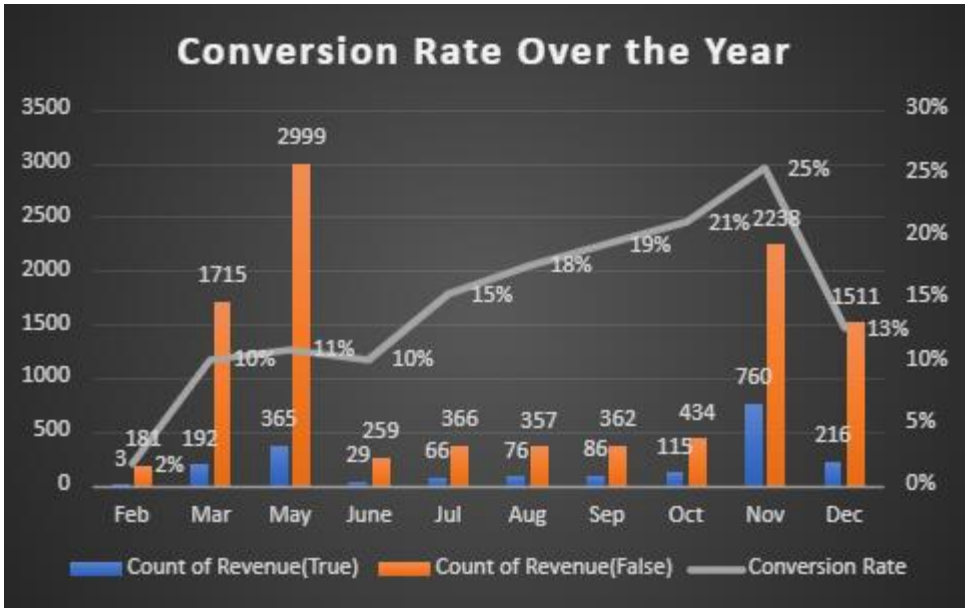


Our dataset had 0 null values and based on Correlation Matrix, we analyzed that there were a few pair of numerical columns that were highly correlated to each other. To avoid the impact of their correlation on our model's accuracy we decided to merge each of the pair of correlated columns into single column as a new attribute. We now have a total of 14 columns on which we are doing our model building. We also get an indication that the variable PageValues has potentially high correlation with our target output variable.

3.3. Count of Revenue Generated:

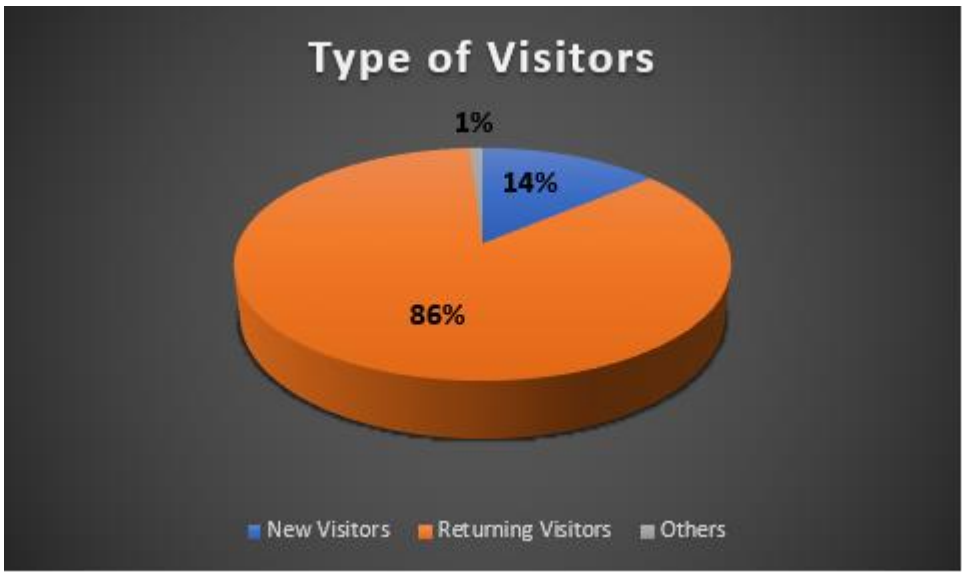


3.4. Conversion Rate Over the Year:



3.5. Page Value Over the Year:

3.6. Types of Visitors:

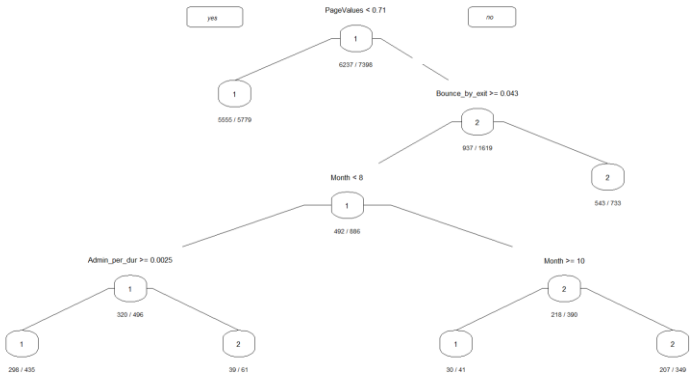


4. BI MODELLING:

4.1. Model 1 – Decision Tree:

The Decision Tree is a powerful and widely used tool for classification and prediction. A flowchart is similar to a tree structure, with each internal node representing a test on an attribute, each branch representing a test outcome, and each leaf node (terminal node) holding a class label.

DEFAULT TREE:



Model Accuracy of Training Data of the Default Tree – 90.19%:

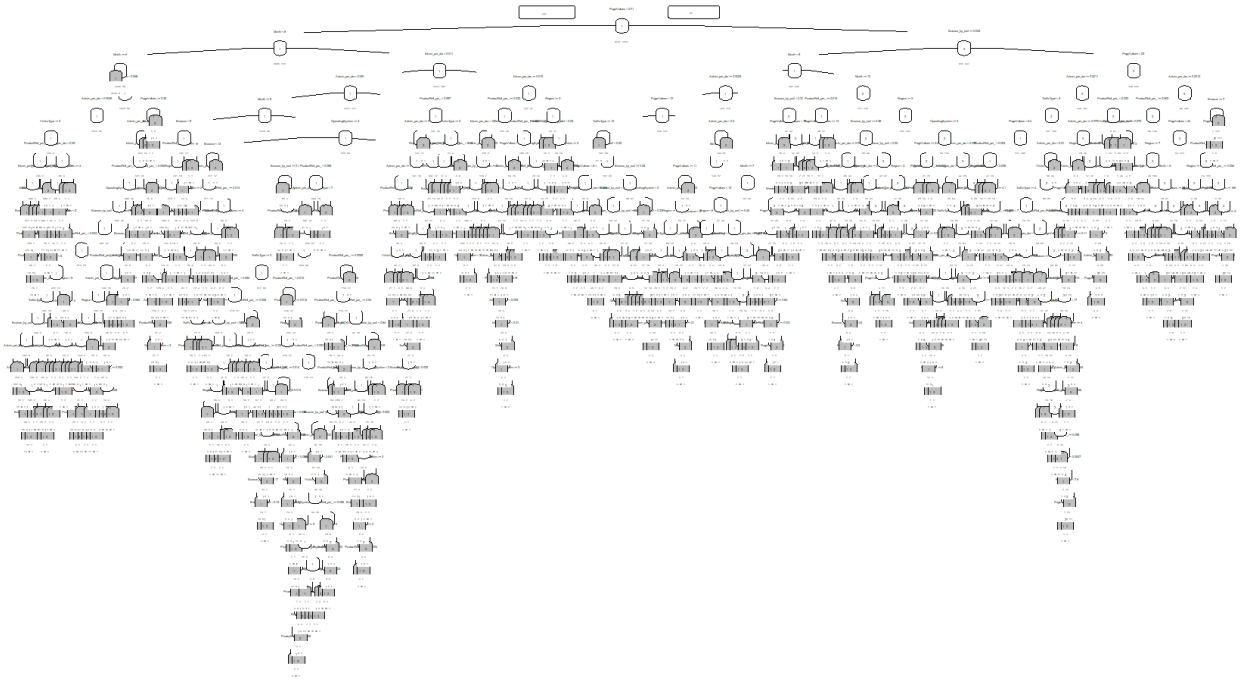
```
Confusion Matrix and Statistics

          Reference
Prediction    1    2
         1 5883  372
         2  354  789

               Accuracy : 0.9019
                 95% CI : (0.8949, 0.9086)
    No Information Rate : 0.8431
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.6268

 Mcnemar's Test P-Value : 0.5281

            Sensitivity : 0.9432
            Specificity : 0.6796
         Pos Pred Value : 0.9405
         Neg Pred Value : 0.6903
             Prevalence : 0.8431
         Detection Rate : 0.7952
   Detection Prevalence : 0.8455
      Balanced Accuracy : 0.8114

       'Positive' Class : 1
```

Model Accuracy of Validation Data of the Default Tree – 89.29%:

```
Confusion Matrix and Statistics

          Reference
Prediction    1    2
         1 3923  266
         2  262  481

               Accuracy : 0.8929
                 95% CI : (0.884, 0.9014)
    No Information Rate : 0.8485
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.5826

 Mcnemar's Test P-Value : 0.8961

            Sensitivity : 0.9374
            Specificity : 0.6439
         Pos Pred Value : 0.9365
         Neg Pred Value : 0.6474
             Prevalence : 0.8485
         Detection Rate : 0.7954
   Detection Prevalence : 0.8494
      Balanced Accuracy : 0.7907

       'Positive' Class : 1
```

Deepest Tree:

To the point where misclassification rate for training dataset is 0%



Model Accuracy of Training Data of the Deepest Tree – 100%:

```
Confusion Matrix and Statistics

          Reference
Prediction    1    2
         1 6237    0
         2    0 1161

               Accuracy : 1
                 95% CI : (0.9995, 1)
    No Information Rate : 0.8431
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 1

 Mcnemar's Test P-Value : NA

            Sensitivity : 1.0000
            Specificity : 1.0000
         Pos Pred Value : 1.0000
         Neg Pred Value : 1.0000
             Prevalence : 0.8431
         Detection Rate : 0.8431
   Detection Prevalence : 0.8431
      Balanced Accuracy : 1.0000

       'Positive' Class : 1

> |
```

Model Accuracy of Validation Data of the Deepest Tree- 86.64:

```
Confusion Matrix and Statistics

          Reference
Prediction    1    2
         1 3848  322
         2  337  425

               Accuracy : 0.8664
                 95% CI : (0.8566, 0.8758)
    No Information Rate : 0.8485
    P-Value [Acc > NIR] : 0.0002081

                  Kappa : 0.4844

 Mcnemar's Test P-Value : 0.5855042

            Sensitivity : 0.9195
            Specificity : 0.5689
         Pos Pred Value : 0.9228
         Neg Pred Value : 0.5577
             Prevalence : 0.8485
         Detection Rate : 0.7802
   Detection Prevalence : 0.8455
      Balanced Accuracy : 0.7442

       'Positive' Class : 1
```

Thus, we can see while the training data shows 100% accuracy the validation data for the deepest tree is only at 86.64%. This suggests overfitting problem and that we need to prune the tree to get desired level of accuracy in our model.

Post-Pruning:

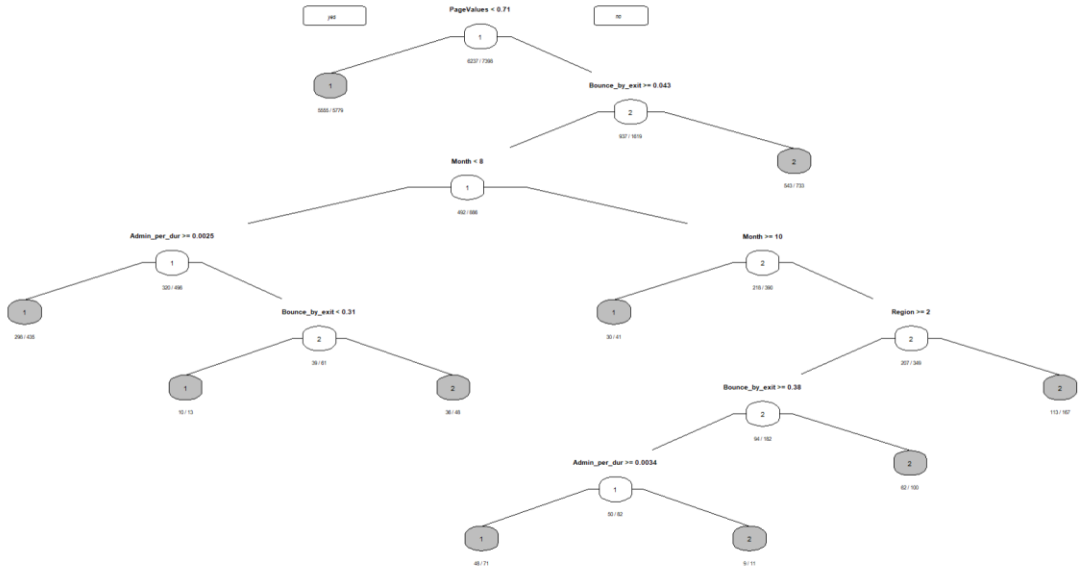
Finding the point where misclassification rate in the validation dataset is the minimum.

Conclusion at nsplit 9 our tree will show the lowest misclassification rate in the validation dataset as per cp table.

```
Classification tree:
rpart(formula = Revenue ~ ., data = train.df, method = "class",
    cp = 1e-05, minsplit = 1, xval = 5)

Variables actually used in tree construction:
 [1] Admin_per_dur      Bounce_by_exit     Browser            Inform_per_dur
 [5] Month              OperatingSystems   PageValues         ProductRel_per_dur
 [9] Region             SpecialDay         TrafficType        VisitorType
[13] Weekend

Root node error: 1161/7398 = 0.15693

n= 7398
```

	CP	nsplit	rel error	xerror	xstd
1	0.21963824	0	1.00000000	1.00000	0.026947
2	0.08440999	1	0.78036176	0.78811	0.024390
3	0.03962102	2	0.69595177	0.72351	0.023504
4	0.01636520	3	0.65633075	0.68906	0.023007
5	0.01464255	4	0.63996555	0.68906	0.023007
6	0.00775194	5	0.62532300	0.68475	0.022944
7	0.00602929	7	0.60981912	0.66581	0.022662
8	0.00344531	9	0.59776055	0.66408	0.022636
9	0.00301464	17	0.56933678	0.67356	0.022778
10	0.00258398	19	0.56330749	0.68562	0.022956
11	0.00229687	22	0.55555556	0.68389	0.022931
12	0.00215332	31	0.53143842	0.68562	0.022956
13	0.00200976	47	0.49009475	0.69423	0.023083
14	0.00172265	50	0.48406546	0.69681	0.023120
15	0.00147656	72	0.44616710	0.70284	0.023208
16	0.00143554	80	0.43324720	0.70284	0.023208
17	0.00129199	96	0.40654608	0.69509	0.023095
18	0.00114844	131	0.35831180	0.70457	0.023233
19	0.00110742	137	0.35142119	0.70457	0.023233
20	0.00107666	154	0.33074935	0.70457	0.023233
21	0.00103359	158	0.32644272	0.73730	0.023698
22	0.00086133	169	0.31438415	0.74763	0.023841
23	0.00064599	332	0.17312661	0.77606	0.024229
24	0.00057422	340	0.16795866	0.81309	0.024718
25	0.00051680	391	0.13781223	0.82171	0.024829
26	0.00043066	396	0.13522825	0.90009	0.025802
27	0.00034453	637	0.01464255	0.90525	0.025864
28	0.00032300	642	0.01291990	0.91990	0.026037
29	0.00028711	655	0.00861326	0.91990	0.026037
30	0.00021533	682	0.00086133	0.92334	0.026078
31	0.00001000	686	0.00000000	0.92420	0.026088

Pruned Tree:



Model Accuracy of Training Data of the Pruned Tree – 90.62%:

```
Confusion Matrix and Statistics

          Reference
Prediction    1    2
         1 5941  398
         2  296  763

               Accuracy : 0.9062
                 95% CI : (0.8993, 0.9127)
    No Information Rate : 0.8431
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.6323

 Mcnemar's Test P-Value : 0.0001261

            Sensitivity : 0.9525
            Specificity : 0.6572
         Pos Pred Value : 0.9372
         Neg Pred Value : 0.7205
             Prevalence : 0.8431
         Detection Rate : 0.8031
   Detection Prevalence : 0.8569
      Balanced Accuracy : 0.8049

       'Positive' Class : 1
```

Model Accuracy of Validation Data of the Pruned Tree – 89.31%:

```
Confusion Matrix and Statistics

          Reference
Prediction    1    2
         1 3955  297
         2  230  450

               Accuracy : 0.8931
                 95% CI : (0.8842, 0.9016)
    No Information Rate : 0.8485
    P-Value [Acc > NIR] : < 2e-16

                  Kappa : 0.5684

 Mcnemar's Test P-Value : 0.00404

            Sensitivity : 0.9450
            Specificity : 0.6024
         Pos Pred Value : 0.9302
         Neg Pred Value : 0.6618
             Prevalence : 0.8485
         Detection Rate : 0.8019
   Detection Prevalence : 0.8621
      Balanced Accuracy : 0.7737

       'Positive' Class : 1
```

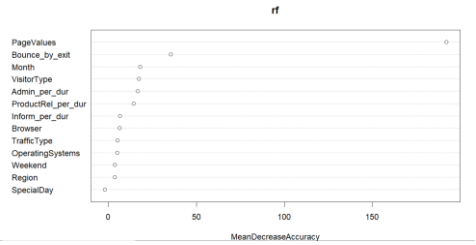
Accuracy Comparison:

We are getting highest accuracy level with the pruned tree at 89.31%.

	Training	Test	Difference
PT	90.62	89.31	1.31
DEPT	100	86.64	13.36
DT	90.19	89.29	0.9

Applying Random Forest:

Not Intuitive Anymore as we have 500 trees, we can explain decision making/ data-driven insights



Model Accuracy for Training Data-97.55%:

```
Confusion Matrix and Statistics

          Reference
Prediction    1    2
         1 6228  172
         2    9  989

               Accuracy : 0.9755
                 95% CI : (0.9718, 0.9789)
    No Information Rate : 0.8431
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.9019

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.9986
            Specificity : 0.8519
         Pos Pred Value : 0.9731
         Neg Pred Value : 0.9910
             Prevalence : 0.8431
         Detection Rate : 0.8418
   Detection Prevalence : 0.8651
      Balanced Accuracy : 0.9252

       'Positive' Class : 1
```

Model Accuracy for Validation Data-89.66%:

```
Confusion Matrix and Statistics

          Reference
Prediction    1    2
         1 4003  328
         2  182  419

               Accuracy : 0.8966
                 95% CI : (0.8878, 0.905)
    No Information Rate : 0.8485
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.5626

 Mcnemar's Test P-Value : 1.356e-10

            Sensitivity : 0.9565
            Specificity : 0.5609
         Pos Pred Value : 0.9243
         Neg Pred Value : 0.6972
             Prevalence : 0.8485
         Detection Rate : 0.8116
   Detection Prevalence : 0.8781
      Balanced Accuracy : 0.7587

       'Positive' Class : 1
```

Applying Boosted Trees:

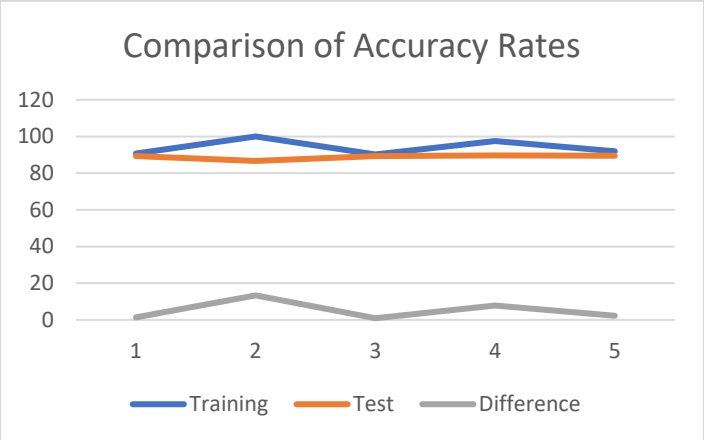
Models Accuracy for Training Data-91.89%:

```
Confusion Matrix and Statistics

          Reference
Prediction    1    2
         1 6021  384
         2  216  777

               Accuracy : 0.9189
                 95% CI : (0.9124, 0.925)
    No Information Rate : 0.8431
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.6743

 Mcnemar's Test P-Value : 9.248e-12

            Sensitivity : 0.9654
            Specificity : 0.6693
         Pos Pred Value : 0.9400
         Neg Pred Value : 0.7825
             Prevalence : 0.8431
         Detection Rate : 0.8139
   Detection Prevalence : 0.8658
      Balanced Accuracy : 0.8173

       'Positive' Class : 1
```

Model Accuracy for Validation Data -89.56%:

```
Confusion Matrix and Statistics

          Reference
Prediction    1    2
         1 3974  304
         2  211  443

               Accuracy : 0.8956
                 95% CI : (0.8867, 0.904)
    No Information Rate : 0.8485
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.5719

 Mcnemar's Test P-Value : 5.035e-05

            Sensitivity : 0.9496
            Specificity : 0.5930
         Pos Pred Value : 0.9289
         Neg Pred Value : 0.6774
             Prevalence : 0.8485
         Detection Rate : 0.8058
   Detection Prevalence : 0.8674
      Balanced Accuracy : 0.7713

       'Positive' Class : 1
```

Final Accuracy Comparison Table:



Number	Model Type	Training	Test	Difference
1	Pruned Tree	90.62	89.31	1.31
2	Deeper Tree	100	86.64	13.36
3	Default Tree	90.19	89.29	0.9
4	Random Forest(500 trees)	97.55	89.66	7.89
5	Boosted Tree	91.89	89.56	2.33

Inference:

- Although, the final accuracy for Boosted Trees is slightly higher than the Pruned Tree we are going to go ahead with the pruned tree as it has a lower difference between training and validation dataset accuracy.
- The variables mentioned above are the most important feature of the dataset for the pruned tree and default tree. Four input variables (PageValues, Bounce_by_exit, Month, Adim_per_dur) are coming out as more important for output variable revenue as also suggested by Logistic Regression Model.
- Comparing the Sensitivity and Specificity of the three models (default tree, deepest tree, pruned tree, Applying Random Forest, and Applying Boosted Tree).Sensitivity is the metric evaluates a model's ability to predict true positive of each available category. Specificity is the metric evaluates a model's ability to predict true negatives of each available category.

  Sensitivity = True Positives/ True Positives + False Negatives

  Specificity = True Negatives/ True Negatives + False Positives

	Default Tree	Deepest Tree	Pruned Tree	Random Forest	Boosted Tree
Sensitivity	0.9374	0.9195	0.9450	0.9565	0.9496
Specificity	0.6439	0.5689	0.6024	0.5609	0.5930

4.2. Model 2 – Logistic Regression:

The relationship between the dependent variable and one or more independent variables can be better understood using logistic regression. It is used to predict Customer Purchase accuracy rate for revenue when the dependent variable (target) is categorical.

General Logistic Regression:

```
Call:
glm(formula = Revenue ~ ., family = "binomial", data = train.df)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-5.3972  -0.4756  -0.4004  -0.2584   3.8207

Coefficients:
                         Estimate   Std. Error z value             Pr(>|z|)
(Intercept)        -2.801056550  0.250478539 -11.183 < 0.0000000000000002 ***
PageValues          0.080488050  0.002890005  27.850 < 0.0000000000000002 ***
SpecialDay         -1.503601212  0.304935494  -4.931      0.0000008185880 ***
Month               0.121346491  0.017775746   6.827      0.0000000000087 ***
OperatingSystems   -0.094207931  0.048049347  -1.961              0.04992 *
Browser             0.024755376  0.024163703   1.024              0.30561
Region             -0.010402230  0.016465755  -0.632              0.52755
TrafficType         0.006195465  0.010220064   0.606              0.54438
VisitorType        -0.109419158  0.054145151  -2.021              0.04330 *
Weekend             0.204429996  0.089199028   2.292              0.02191 *
ProductRel_per_dur -0.000015880  0.000004423  -3.590              0.00033 ***
Admin_per_dur       0.000005910  0.000002988   1.978              0.04791 *
Inform_per_dur      0.000004948  0.000002312   2.140              0.03232 *
Bounce_by_exit     -0.404983594  0.137122656  -2.953              0.00314 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 6290.0  on 7397  degrees of freedom
Residual deviance: 4562.2  on 7384  degrees of freedom
AIC: 4590.2

Number of Fisher Scoring iterations: 7

>
```

First 5 Actual and Predicted Records:

```
> data.frame(actual = valid.df$Revenue[1:5], predicted = logit.reg.pred[1:5])
  actual   predicted
2      0 0.063105576
3      0 0.006821198
4      0 0.050468527
6      0 0.049651125
7      0 0.005182763
>
```

Model Accuracy of validation data actual and predicted records-88.4%:

```
> confusionMatrix(as.factor(logit.reg.pred.classes), as.factor(valid.df$Revenue))
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 4078  506
         1   66  282

               Accuracy : 0.884
                 95% CI : (0.8748, 0.8928)
    No Information Rate : 0.8402
    P-Value [Acc > NIR] : < 0.00000000000000022

                  Kappa : 0.4418

 Mcnemar's Test P-Value : < 0.00000000000000022

            Sensitivity : 0.9841
            Specificity : 0.3579
         Pos Pred Value : 0.8896
         Neg Pred Value : 0.8103
             Detection Rate : 0.8268
       Detection Prevalence : 0.9294
          Balanced Accuracy : 0.6710

           'Positive' Class : 0

> |
```

Model Selection for Full Logistic Regression:

```
Call:
glm(formula = Revenue ~ 1, family = "binomial", data = train.df)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-0.573  -0.573  -0.573  -0.573   1.943

Coefficients:
            Estimate Std. Error z value            Pr(>|z|)
(Intercept) -1.72372    0.03244  -53.14 <0.0000000000000002 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 6290  on 7397  degrees of freedom
Residual deviance: 6290  on 7397  degrees of freedom
AIC: 6292

Number of Fisher Scoring iterations: 4

> |
```

```
> backwards = step(full.logit.reg)
Start:  AIC=4590.16
Revenue ~ PageValues + SpecialDay + Month + OperatingSystems +
    Browser + Region + TrafficType + VisitorType + Weekend +
    ProductRel_per_dur + Admin_per_dur + Inform_per_dur + Bounce_by_exit

                   Df Deviance    AIC
- TrafficType       1   4562.5 4588.5
- Region            1   4562.6 4588.6
- Browser           1   4563.2 4589.2
<none>                  4562.2 4590.2
- Admin_per_dur     1   4565.7 4591.7
- OperatingSystems  1   4566.1 4592.1
- VisitorType       1   4566.2 4592.2
- Inform_per_dur    1   4566.3 4592.3
- Weekend           1   4567.3 4593.3
- Bounce_by_exit    1   4571.3 4597.3
- ProductRel_per_dur 1  4588.7 4614.7
- SpecialDay        1   4594.7 4620.7
- Month             1   4611.9 4637.9
- PageValues        1   5954.6 5980.6

Step:  AIC=4586.9
Revenue ~ PageValues + SpecialDay + Month + OperatingSystems +
    Browser + VisitorType + Weekend + ProductRel_per_dur + Admin_per_dur +
    Inform_per_dur + Bounce_by_exit

                   Df Deviance    AIC
- Browser           1   4563.9 4585.9
<none>                  4562.9 4586.9
- Admin_per_dur     1   4566.5 4588.5
- OperatingSystems  1   4566.7 4588.7
- VisitorType       1   4566.9 4588.9
- Inform_per_dur    1   4567.1 4589.1
- Weekend           1   4568.1 4590.1
- Bounce_by_exit    1   4571.8 4593.8
- ProductRel_per_dur 1  4589.4 4611.4
- SpecialDay        1   4595.1 4617.1
- Month             1   4613.3 4635.3
- PageValues        1   5956.0 5978.0

Step:  AIC=4588.53
Revenue ~ PageValues + SpecialDay + Month + OperatingSystems +
    Browser + Region + VisitorType + Weekend + ProductRel_per_dur +
    Admin_per_dur + Inform_per_dur + Bounce_by_exit

                   Df Deviance    AIC
- Region            1   4562.9 4586.9
- Browser           1   4563.6 4587.6
<none>                  4562.5 4588.5
- Admin_per_dur     1   4566.1 4590.1
- OperatingSystems  1   4566.2 4590.2
- VisitorType       1   4566.6 4590.6
- Inform_per_dur    1   4566.7 4590.7
- Weekend           1   4567.7 4591.7
- Bounce_by_exit    1   4571.5 4595.5
- ProductRel_per_dur 1  4589.0 4613.0
- SpecialDay        1   4594.8 4618.8
- Month             1   4612.7 4636.7
- PageValues        1   5955.4 5979.4

Step:  AIC=4585.93
Revenue ~ PageValues + SpecialDay + Month + OperatingSystems +
    VisitorType + Weekend + ProductRel_per_dur + Admin_per_dur +
    Inform_per_dur + Bounce_by_exit

                   Df Deviance    AIC
<none>                  4563.9 4585.9
- OperatingSystems  1   4567.2 4587.2
- Admin_per_dur     1   4567.5 4587.5
- VisitorType       1   4567.9 4587.9
- Inform_per_dur    1   4568.1 4588.1
- Weekend           1   4569.0 4589.0
- Bounce_by_exit    1   4573.3 4593.3
- ProductRel_per_dur 1  4590.2 4610.2
- SpecialDay        1   4595.8 4615.8
- Month             1   4613.9 4633.9
- PageValues        1   5959.7 5979.7
```

```
> summary(backwards)

Call:
glm(formula = Revenue ~ PageValues + SpecialDay + Month + OperatingSystems +
    VisitorType + Weekend + ProductRel_per_dur + Admin_per_dur +
    Inform_per_dur + Bounce_by_exit, family = "binomial", data = train.df)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-5.4043  -0.4747  -0.4006  -0.2572   3.8283

Coefficients:
                     Estimate   Std. Error z value             Pr(>|z|)
(Intercept)       -2.771507371  0.239455705 -11.574 < 0.0000000000000002 ***
PageValues         0.080517497  0.002887802  27.882 < 0.0000000000000002 ***
SpecialDay        -1.487063480  0.304389153  -4.885     0.00000103217863 ***
Month              0.121351480  0.017743253   6.839     0.00000000000796 ***
OperatingSystems  -0.083711811  0.046988685  -1.782             0.074826 .
VisitorType       -0.109083532  0.054087257  -2.017             0.043716 *
Weekend            0.202255046  0.089100886   2.270             0.023210 *
ProductRel_per_dur -0.000015811  0.000004421  -3.576             0.000349 ***
Admin_per_dur      0.000005945  0.000002983   1.993             0.046269 *
```

```
Inform_per_dur      0.000004924  0.000002314   2.128             0.033327 *
Bounce_by_exit     -0.407094366  0.136529066  -2.982             0.002866 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 6290.0  on 7397  degrees of freedom
Residual deviance: 4563.9  on 7387  degrees of freedom
AIC: 4585.9

Number of Fisher Scoring iterations: 7

> 
```

Model Accuracy of Validation data backward full logistic regression-88.36%:

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 4077  507
         1   67  281

               Accuracy : 0.8836
                 95% CI : (0.8743, 0.8924)
    No Information Rate : 0.8402
    P-Value [Acc > NIR] : < 0.00000000000000022

                  Kappa : 0.4399

 Mcnemar's Test P-Value : < 0.00000000000000022

            Sensitivity : 0.9838
            Specificity : 0.3566
         Pos Pred Value : 0.8894
         Neg Pred Value : 0.8075
             Prevalence : 0.8402
         Detection Rate : 0.8266
   Detection Prevalence : 0.9294
      Balanced Accuracy : 0.6702

       'Positive' Class : 0

> 
```

Final General Logistic Regression:

```
Call:
glm(formula = Revenue ~ PageValues + ProductRel_per_dur + Month +
    SpecialDay + Bounce_by_exit + Weekend + Inform_per_dur +
    VisitorType + Admin_per_dur + OperatingSystems, family = "binomial",
    data = train.df)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-5.4043  -0.4747  -0.4006  -0.2572   3.8283

Coefficients:
                       Estimate   Std. Error z value             Pr(>|z|)
(Intercept)        -2.771507371  0.239455705 -11.574 < 0.0000000000000002 ***
PageValues          0.080517497  0.002887802  27.882 < 0.0000000000000002 ***
ProductRel_per_dur -0.000015811  0.000004421  -3.576             0.000349 ***
Month               0.121351480  0.017743253   6.839     0.00000000000796 ***
SpecialDay         -1.487063480  0.304389153  -4.885     0.00000103217863 ***
Bounce_by_exit     -0.407094366  0.136529066  -2.982             0.002866 **
Weekend             0.202255046  0.089100886   2.270             0.023210 *
Inform_per_dur      0.000004924  0.000002314   2.128             0.033327 *
VisitorType        -0.109083532  0.054087257  -2.017             0.043716 *
Admin_per_dur       0.000005945  0.000002983   1.993             0.046269 *
OperatingSystems   -0.083711811  0.046988685  -1.782             0.074826 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 6290.0  on 7397  degrees of freedom
Residual deviance: 4563.9  on 7387  degrees of freedom
AIC: 4585.9

Number of Fisher Scoring iterations: 7

>
```

Model Accuracy of Validation data full logistic regression-88.36%:

```
> confusionMatrix(as.factor(final.logit.reg.pred.classes), as.factor(valid.df$Revenue))
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 4077  507
         1   67  281

               Accuracy : 0.8836
                 95% CI : (0.8743, 0.8924)
    No Information Rate : 0.8402
    P-Value [Acc > NIR] : < 0.00000000000000022

                  Kappa : 0.4399

 Mcnemar's Test P-Value : < 0.00000000000000022

            Sensitivity : 0.9838
            Specificity : 0.3566
         Pos Pred Value : 0.8894
         Neg Pred Value : 0.8075
             Prevalence : 0.8402
```

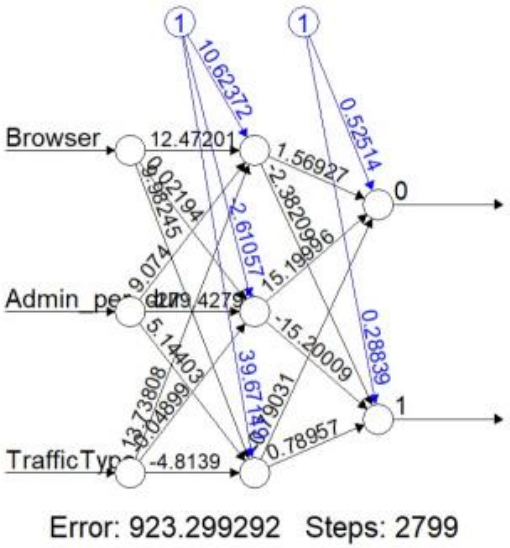
```
          Detection Rate : 0.8266
    Detection Prevalence : 0.9294
       Balanced Accuracy : 0.6702

        'Positive' Class : 0

>
```

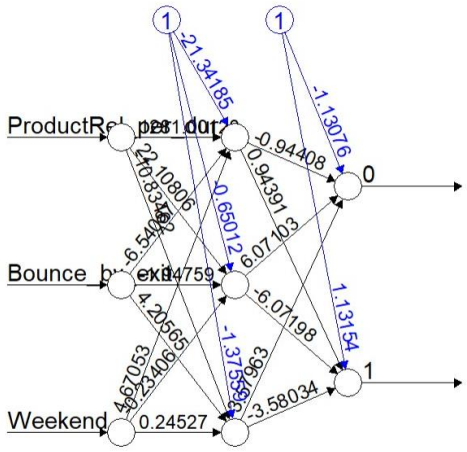
Inference:

- We can infer that the odds of increasing the satisfaction level are more inclined towards the variables PageValues, Bounce_by_exit, Month, Adim_per_dur) which again agrees with the output suggested by Decision Tree analysis done above.
- Comparing the testing accuracy of the two models (backward and full logistic regression), the validation accuracy of both models is same 88.36%.
- Comparing Logistic Regression and Decision Tree, Decision Tree is better fit for this Model

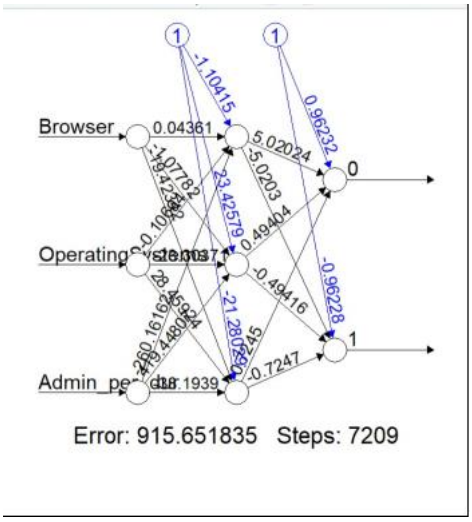
4.3 Model 3 – Neural Network's:

The neural network uses parallel information processing to extract meaningful information and detect hidden patterns in complex data sets.

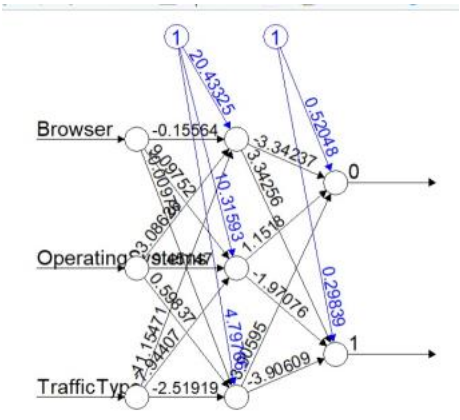




Error: 907.176453 Steps: 58196



Error: 915.651835 Steps: 7209



Error: 933.137099 Steps: 16215

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 4077  507
         1   67  281

               Accuracy : 0.8836
                 95% CI : (0.8743, 0.8924)
    No Information Rate : 0.8402
    P-Value [Acc > NIR] : < 0.00000000000000022

                  Kappa : 0.4399

 Mcnemar's Test P-Value : < 0.00000000000000022

            Sensitivity : 0.9838
            Specificity : 0.3566
         Pos Pred Value : 0.8894
         Neg Pred Value : 0.8075
             Prevalence : 0.8402
         Detection Rate : 0.8266

    Detection Prevalence : 0.9294
       Balanced Accuracy : 0.6702

        'Positive' Class : 0

>
```

Inferences:

- The Accuracy of validation dataset for neural networks is 88.36%

5. CONCLUSION:

- **We recommend Decision Tree for this dataset after comparing all the results for three models.** Though the scores are not the best, they are influenced by the dataset's extreme outliers and skewness. As a result, resampling the data or adding more data will affect the model's accuracy and may improve predictions.
- **As highlighted by the models the dependent variables** Page Values, Bounce_by_exit, Month, Adim_per_dur) are likely to have more impact on the output variable revenue so we could use them to segment our audience for marketing campaigns and get more desirable marketing outcomes.

	Decision Tree	Logistic Regression	Neural Network's
Validation Dataset	89.31%	88.36%	81.39%

Insights from EDA:

- Holiday season October and November witnessed comparatively higher conversion rate than other months with highest being in November.
- However, we find the month of May to be particularly interesting as it has the highest number of visits but below average conversion rate of 11%. Average conversion rate is 14.4%. Need to get further data to bifurcate the visitor base and investigate the reasons for this.
- Almost 86% of the visitors were loyal/returning visitors we need to work on targeting them in the right month and the right time and tapping into their full potential to contribute towards revenue. **We can use this model to run a targeted Loyalty Program to retain these returning customers.**

Potential Economic Impact:

- Let's assume the company would plan to run a promotional (Price drop) mailing campaign targeted towards those customers that are predicted to generate revenue while visiting.
- Average Cost of mailing $1 and average revenue from respondent $50.
- If we follow our decision tree model and the final confusion matrix to find the target audience we can see we would be able to generate net profit of $176,490 by running the campaign.
- This is a very simple model to explain the impact of targeting using appropriate techniques and algorithms.

	Actual	
Predicted	**1**	2
1	3955	297
2	230	450

Cost	$21,260.00
Revenue	$197,750.00
Net Profit	$176,490.00
Projected Profit%	830%
by tageting using the model	

6. REFERENCES:

- https://www.pluralsight.com/guides/explore-r-libraries:-rpart
- https://thatascience.com/learn-machine-learning/gini-entropy/#:~:text=Gini%20index%20and%20entropy%20is,only%20one%20class%20is%20pure.
- https://discuss.analyticsvidhya.com/t/how-does-complexity-parameter-cp-work-in-decision-tree/6589
- https://stats.stackexchange.com/questions/524510/logistic-regression-what-is-the-link-between-the-binomial-family-and-the-binomi
- https://www.geeksforgeeks.org/get-or-set-dimensions-of-a-matrix-in-r-programming-dim-function/#:~:text=dim()%20function%20in%20R,matrix%2C%20array%20or%20data%20frame.&text=Parameters%3A,array%2C%20matrix%20or%20data%20frame.
- https://www.rdocumentation.org/packages/base/versions/3.6.2/topics/sample
- https://stackoverflow.com/questions/34617883/how-to-remove-multiple-columns-in-r-dataframe
- https://machinelearningmastery.com/confusion-matrix-machine-learning/