decision_tree_using_caret_package

June 9, 2018

0.1 In this exercise, we will use the HR dataset and understand the following using caret package:

1. Building the decision tree model
2. Creating the Confusion Matrix and ROC plot on train data
3. Creating the Confusion Matrix and ROC plot on test data

There are bugs/missing code in the entire exercise. The participants are expected to work upon them.

0.2 Here are some useful links:

1. **Read** about interaction variable coding
2. Refer **link** to know about adding lables to factors
3. Refer **link** to relvel factor variables
4. **Read** about the issues in stepwise regression
5. **Read** about the modelling activity via caret package
6. The **complete** list of tuning parameter for different models in caret package

---

1 Code starts here

We are going to use below mentioned libraries for demonstrating logistic regression:

```
In [1]: library(caret)    #for data partition. Model building
        #library(Deducer) #for ROC plot
        library(ROCR)     #for ROC plot (other way)
        #library(rattle)   #for plotting tree
        library(rpart)
```

```
Loading required package: lattice
Loading required package: ggplot2
Loading required package: gplots

Attaching package: gplots

The following object is masked from package:stats:
```

```
lowess
```

1.1 Data Import and Manipulation

1.1.1 1. Importing a data set

Give the correct path to the data

```
In [2]: raw_df <- read.csv("/Users/Rahul/Documents/Datasets/IMB533_HR_Data_No_Missing_Value.csv
```

Note that `echo = FALSE` parameter prevents printing the R code that generated the plot.

1.1.2 2. Structure and Summary of the dataset

```
In [3]: str(raw_df)
        summary(raw_df)
```

```
'data.frame':        8995 obs. of  18 variables:
 $ SLNO                       : int  1 2 3 4 5 6 7 9 11 12 ...
 $ Candidate.Ref              : int  2110407 2112635 2112838 2115021 2115125 2117167 2119124 2
 $ DOJ.Extended               : Factor w/ 2 levels "No","Yes": 2 1 1 1 2 2 2 2 1 1 ...
 $ Duration.to.accept.offer   : int  14 18 3 26 1 17 37 16 1 6 ...
 $ Notice.period              : int  30 30 45 30 120 30 30 0 30 30 ...
 $ Offered.band               : Factor w/ 4 levels "E0","E1","E2",..: 3 3 3 3 3 2 3 2 2 2 ...
 $ Pecent.hike.expected.in.CTC: num  -20.8 50 42.8 42.8 42.6 ...
 $ Percent.hike.offered.in.CTC: num  13.2 320 42.8 42.8 42.6 ...
 $ Percent.difference.CTC     : num  42.9 180 0 0 0 ...
 $ Joining.Bonus              : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 1 1 1 ...
 $ Candidate.relocate.actual  : Factor w/ 2 levels "No","Yes": 1 1 1 1 2 1 1 1 1 1 ...
 $ Gender                     : Factor w/ 2 levels "Female","Male": 1 2 2 2 2 2 2 1 1 2 ...
 $ Candidate.Source           : Factor w/ 3 levels "Agency","Direct",..: 1 3 1 3 3 3 3 2 3 3 .
 $ Rex.in.Yrs                 : int  7 8 4 4 6 2 7 8 3 3 ...
 $ LOB                        : Factor w/ 9 levels "AXON","BFSI",..: 5 8 8 8 8 8 8 7 2 3 ...
 $ Location                   : Factor w/ 11 levels "Ahmedabad","Bangalore",..: 9 3 9 9 9 9 9
 $ Age                        : int  34 34 27 34 34 34 32 34 26 34 ...
 $ Status                     : Factor w/ 2 levels "Joined","Not Joined": 1 1 1 1 1 1 1 1 1 1
```

```
      SLNO        Candidate.Ref     DOJ.Extended Duration.to.accept.offer
 Min.   :    1   Min.   :2109586   No :4788     Min.   :  0.00
 1st Qu.: 3208   1st Qu.:2386476   Yes:4207     1st Qu.:  3.00
 Median : 5976   Median :2807482                Median : 10.00
 Mean   : 5971   Mean   :2843647                Mean   : 21.43
 3rd Qu.: 8739   3rd Qu.:3300060                3rd Qu.: 33.00
 Max.   :12333   Max.   :3836076                Max.   :224.00
```

```
Notice.period    Offered.band Pecent.hike.expected.in.CTC
Min.   :  0.00   E0: 211      Min.   :-68.83
1st Qu.: 30.00   E1:5568      1st Qu.: 27.27
Median : 30.00   E2:2711      Median : 40.00
Mean   : 39.29   E3: 505      Mean   : 43.86
3rd Qu.: 60.00                3rd Qu.: 53.85
Max.   :120.00                Max.   :359.77

Percent.hike.offered.in.CTC Percent.difference.CTC Joining.Bonus
Min.   :-60.53              Min.   :-67.270        No :8578
1st Qu.: 22.09              1st Qu.: -8.330        Yes: 417
Median : 36.00              Median :  0.000
Mean   : 40.66              Mean   : -1.574
3rd Qu.: 50.00              3rd Qu.:  0.000
Max.   :471.43              Max.   :300.000

Candidate.relocate.actual    Gender               Candidate.Source
No :7705                  Female:1551   Agency           :2585
Yes:1290                  Male  :7444   Direct           :4801
                                        Employee Referral:1609



  Rex.in.Yrs          LOB              Location          Age
Min.   : 0.000   INFRA  :2850   Chennai  :3150   Min.   :20.00
1st Qu.: 3.000   ERS    :2426   Noida    :2727   1st Qu.:27.00
Median : 4.000   BFSI   :1396   Bangalore:2230   Median :29.00
Mean   : 4.239   ETS    : 691   Hyderabad: 341   Mean   :29.91
3rd Qu.: 6.000   CSMP   : 579   Mumbai   : 197   3rd Qu.:34.00
Max.   :24.000   AXON   : 568   Gurgaon  : 146   Max.   :60.00
                 (Other): 485   (Other)  : 204
       Status
Joined    :7313
Not Joined:1682
```

Create a new data frame and store the raw data copy. This is being done to have a copy of the raw data intact for further manipulation if needed.

```
In [4]: filter_df <- na.omit(raw_df) # listwise deletion of missing
```

1.1.3 3. Create train and test dataset

Reserve 80% for *training* and 20% of *test* *Correct the error in the below code chunk*

```
In [5]: set.seed(2341)
        trainIndex <- createDataPartition(filter_df$Status, p = 0.80, list = FALSE)
        train_df <- filter_df[trainIndex,]
        test_df <- filter_df[-trainIndex,]
```

We can pull the specific attribute needed to build the model is another data frame. This agian is more of a hygine practice to not touch the **train** and **test** data set directly.

Correct the error in the below code chunk

```
In [6]: dt_train_df <- as.data.frame(train_df[,c("DOJ.Extended",
                                                     "Duration.to.accept.offer",
                                                     "Notice.period",
                                                     "Offered.band",
                                                     "Percent.difference.CTC",
                                                     "Joining.Bonus",
                                                     "Gender",
                                                     "Candidate.Source",
                                                     "Rex.in.Yrs",
                                                     "LOB",
                                                     "Location",
                                                     "Age",
                                                     "Status"
        )])
```

Correct the error in the below code chunk

```
In [7]: dt_test_data <- as.data.frame(test_df[,c("DOJ.Extended",
                                                   "Duration.to.accept.offer",
                                                   "Notice.period",
                                                   "Offered.band",
                                                   "Percent.difference.CTC",
                                                   "Joining.Bonus",
                                                   "Gender",
                                                   "Candidate.Source",
                                                   "Rex.in.Yrs",
                                                   "LOB",
                                                   "Location",
                                                   "Age",
                                                   "Status"
        )])
```

1.2 Model Building: Using the caret() package

There are a number of models which can be built using caret package. To get the names of all the models possible.

```
In [8]: names(getModelInfo())
```

1. ’ada’ 2. ’AdaBag’ 3. ’AdaBoost.M1’ 4. ’adaboost’ 5. ’amdai’ 6. ’ANFIS’ 7. ’avNNet’ 8. ’awnb’ 9. ’awtan’ 10. ’bag’ 11. ’bagEarth’ 12. ’bagEarthGCV’ 13. ’bagFDA’ 14. ’bagFDAGCV’ 15. ’bam’ 16. ’bartMachine’ 17. ’bayesglm’ 18. ’binda’ 19. ’blackboost’ 20. ’blasso’ 21. ’blassoAveraged’ 22. ’bridge’ 23. ’brnn’ 24. ’BstLm’ 25. ’bstSm’ 26. ’bstTree’ 27. ’C5.0’ 28. ’C5.0Cost’ 29. ’C5.0Rules’ 30. ’C5.0Tree’ 31. ’cforest’ 32. ’chaid’ 33. ’CSimca’ 34. ’ctree’ 35. ’ctree2’ 36. ’cubist’ 37. ’dda’ 38. ’deepboost’ 39. ’DENFIS’ 40. ’dnn’ 41. ’dwdLinear’ 42. ’dwdPoly’ 43. ’dwdRadial’ 44. ’earth’ 45. ’elm’ 46. ’enet’ 47. ’evtree’ 48. ’extraTrees’ 49. ’fda’ 50. ’FH.GBML’ 51. ’FIR.DM’ 52. ’foba’ 53. ’FRBCS.CHI’ 54. ’FRBCS.W’ 55. ’FS.HGD’ 56. ’gam’ 57. ’gamboost’ 58. ’gamLoess’ 59. ’gamSpline’ 60. ’gaussprLinear’ 61. ’gaussprPoly’ 62. ’gaussprRadial’ 63. ’gbm_h2o’ 64. ’gbm’ 65. ’gcvEarth’ 66. ’GFS.FR.MOGUL’ 67. ’GFS.LT.RS’ 68. ’GFS.THRIFT’ 69. ’glm.nb’ 70. ’glm’ 71. ’glmboost’ 72. ’glmnet_h2o’ 73. ’glmnet’ 74. ’glmStepAIC’ 75. ’gpls’ 76. ’hda’ 77. ’hdda’ 78. ’hdrda’ 79. ’HYFIS’ 80. ’icr’ 81. ’J48’ 82. ’JRip’ 83. ’kernelpls’ 84. ’kknn’ 85. ’knn’ 86. ’krlsPoly’ 87. ’krlsRadial’ 88. ’lars’ 89. ’lars2’ 90. ’lasso’ 91. ’lda’ 92. ’lda2’ 93. ’leapBackward’ 94. ’leapForward’ 95. ’leapSeq’ 96. ’Linda’ 97. ’lm’ 98. ’lmStepAIC’ 99. ’LMT’ 100. ’loclda’ 101. ’logicBag’ 102. ’LogitBoost’ 103. ’logreg’ 104. ’lssvmLinear’ 105. ’lssvmPoly’ 106. ’lssvmRadial’ 107. ’lvq’ 108. ’M5’ 109. ’M5Rules’ 110. ’manb’ 111. ’mda’ 112. ’Mlda’ 113. ’mlp’ 114. ’mlpKerasDecay’ 115. ’mlpKerasDecayCost’ 116. ’mlpKerasDropout’ 117. ’mlpKerasDropoutCost’ 118. ’mlpML’ 119. ’mlpSGD’ 120. ’mlpWeightDecay’ 121. ’mlpWeightDecayML’ 122. ’monmlp’ 123. ’msaenet’ 124. ’multinom’ 125. ’mxnet’ 126. ’mxnetAdam’ 127. ’naive_bayes’ 128. ’nb’ 129. ’nbDiscrete’ 130. ’nbSearch’ 131. ’neuralnet’ 132. ’nnet’ 133. ’nnls’ 134. ’nodeHarvest’ 135. ’null’ 136. ’OneR’ 137. ’ordinalNet’ 138. ’ORFlog’ 139. ’ORFpls’ 140. ’ORFridge’ 141. ’ORFsvm’ 142. ’ownn’ 143. ’pam’ 144. ’parRF’ 145. ’PART’ 146. ’partDSA’ 147. ’pcaNNet’ 148. ’pcr’ 149. ’pda’ 150. ’pda2’ 151. ’penalized’ 152. ’PenalizedLDA’ 153. ’plr’ 154. ’pls’ 155. ’plsRglm’ 156. ’polr’ 157. ’ppr’ 158. ’PRIM’ 159. ’protoclass’ 160. ’pythonKnnReg’ 161. ’qda’ 162. ’QdaCov’ 163. ’qrf’ 164. ’qrnn’ 165. ’randomGLM’ 166. ’ranger’ 167. ’rbf’ 168. ’rbfDDA’ 169. ’Rborist’ 170. ’rda’ 171. ’regLogistic’ 172. ’relaxo’ 173. ’rf’ 174. ’rFerns’ 175. ’RFlda’ 176. ’rfRules’ 177. ’ridge’ 178. ’rlda’ 179. ’rlm’ 180. ’rmda’ 181. ’rocc’ 182. ’rotationForest’ 183. ’rotationForestCp’ 184. ’rpart’ 185. ’rpart1SE’ 186. ’rpart2’ 187. ’rpartCost’ 188. ’rpartScore’ 189. ’rqlasso’ 190. ’rqnc’ 191. ’RRF’ 192. ’RRFglobal’ 193. ’rrlda’ 194. ’RSimca’ 195. ’rvmLinear’ 196. ’rvmPoly’ 197. ’rvmRadial’ 198. ’SBC’ 199. ’sda’ 200. ’sdwd’ 201. ’simpls’ 202. ’SLAVE’ 203. ’slda’ 204. ’smda’ 205. ’snn’ 206. ’sparseLDA’ 207. ’spikeslab’ 208. ’spls’ 209. ’stepLDA’ 210. ’stepQDA’ 211. ’superpc’ 212. ’svmBoundrangeString’ 213. ’svmExpoString’ 214. ’svmLinear’ 215. ’svmLinear2’ 216. ’svmLinear3’ 217. ’svmLinearWeights’ 218. ’svmLinearWeights2’ 219. ’svmPoly’ 220. ’svmRadial’ 221. ’svmRadialCost’ 222. ’svmRadialSigma’ 223. ’svmRadialWeights’ 224. ’svmSpectrumString’ 225. ’tan’ 226. ’tanSearch’ 227. ’treebag’ 228. ’vbmpRadial’ 229. ’vglmAdjCat’ 230. ’vglmContRatio’ 231. ’vglmCumulative’ 232. ’widekernelpls’ 233. ’WM’ 234. ’wsrf’ 235. ’xgbDART’ 236. ’xgbLinear’ 237. ’xgbTree’ 238. ’xyf’

To get the info on specific model:

```
In [9]: getModelInfo()$glmnet$type
```

1. ’Regression’ 2. ’Classification’

The below chunk of code is standarized way of building model using caret package. Setting in the control parameters for the model.

```
In [10]: objControl <- trainControl(method = "cv", number = 2,
                                    summaryFunction = twoClassSummary,
                                    classProbs = TRUE,
                                    savePredictions = TRUE)
```

Using search grid to fine tune the model

```
In [11]: search_grid <- expand.grid(cp=c(0.001,0.002, 0.003,0.004))
```

The model building starts here. > 1. **metric= "ROC"** uses ROC curve to select the best model.Accuracy, Kappa are other options. To use this change twoClassSummary to defaultSummary in **ObjControl** 2. **verbose = FALSE**: does not show the processing output on console

The factor names at times may not be consistent. R may expect **"Not.Joined"** but the actual level may be **"Not Joined"** This is corrected by using **make.names()** function to give syntactically valid names. Type ?rpart.control in console to get the list of parameters which control the tree growth.

```
In [12]: #dt_train_df$StatusFactor <- as.factor(ifelse(dt_train_df$Status == "Joined", 1,0))
         set.seed(766)
         levels(dt_train_df$Status) <- make.names(levels(factor(dt_train_df$Status)))
         formula <- as.formula(Status~.)

         dt_caret_model <- caret:::train.formula(formula,
                                     dt_train_df,
                                     method = 'rpart', #method missing
                                     metric = "ROC",
                                     maxdepth = 2,
                                     trControl = objControl,
                                    tuneGrid = search_grid)
```

1.3 Model Evaluation

1.3.1 1. One useful plot from caret package is the variable importance plot

In case you get an error "Invalid Graphic state", uncomment the line below

```
In [13]: dt_caret_model$bestTune
         (dt_caret_model$finalModel)
         #dev.off()
         #fancyRpartPlot(dt_caret_model$finalModel)
```

cp
0.001

```
n= 7197

node), split, n, loss, yval, (yprob)
      * denotes terminal node

    1) root 7197 1346 Joined (0.81297763 0.18702237)
      2) Notice.period< 37.5 4748  651 Joined (0.86288964 0.13711036)
        4) Duration.to.accept.offer< 99.5 4719  628 Joined (0.86692096 0.13307904)
    8) Duration.to.accept.offer< 55.5 4504  572 Joined (0.87300178 0.12699822) *
    9) Duration.to.accept.offer>=55.5 215   56 Joined (0.73953488 0.26046512)
     18) GenderMale< 0.5 36    3 Joined (0.91666667 0.08333333) *
     19) GenderMale>=0.5 179   53 Joined (0.70391061 0.29608939)
       38) Duration.to.accept.offer>=61.5 133   33 Joined (0.75187970 0.24812030)
         76) Duration.to.accept.offer< 70.5 65    7 Joined (0.89230769 0.10769231) *
         77) Duration.to.accept.offer>=70.5 68   26 Joined (0.61764706 0.38235294)
          154) DOJ.ExtendedYes>=0.5 54   16 Joined (0.70370370 0.29629630) *
          155) DOJ.ExtendedYes< 0.5 14    4 Not.Joined (0.28571429 0.71428571) *
       39) Duration.to.accept.offer< 61.5 46   20 Joined (0.56521739 0.43478261)
         78) Percent.difference.CTC>=-5.93 34   12 Joined (0.64705882 0.35294118) *
         79) Percent.difference.CTC< -5.93 12    4 Not.Joined (0.33333333 0.66666667) *
  5) Duration.to.accept.offer>=99.5 29    6 Not.Joined (0.20689655 0.79310345) *
3) Notice.period>=37.5 2449  695 Joined (0.71621070 0.28378930)
  6) LOBINFRA>=0.5 598   92 Joined (0.84615385 0.15384615) *
  7) LOBINFRA< 0.5 1851  603 Joined (0.67423015 0.32576985)
   14) Duration.to.accept.offer>=25.5 898  224 Joined (0.75055679 0.24944321)
     28) Duration.to.accept.offer< 109.5 874  207 Joined (0.76315789 0.23684211)
       56) Percent.difference.CTC>=-6.855 539  102 Joined (0.81076067 0.18923933) *
       57) Percent.difference.CTC< -6.855 335  105 Joined (0.68656716 0.31343284)
        114) Age>=28.5 176   41 Joined (0.76704545 0.23295455)
          228) LocationBangalore< 0.5 113   18 Joined (0.84070796 0.15929204) *
          229) LocationBangalore>=0.5 63   23 Joined (0.63492063 0.36507937)
            458) LOBERS>=0.5 25    4 Joined (0.84000000 0.16000000) *
            459) LOBERS< 0.5 38   19 Joined (0.50000000 0.50000000)
              918) Rex.in.Yrs< 5.5 11    3 Joined (0.72727273 0.27272727) *
              919) Rex.in.Yrs>=5.5 27   11 Not.Joined (0.40740741 0.59259259)
               1838) Age< 32.5 14    6 Joined (0.57142857 0.42857143) *
               1839) Age>=32.5 13    3 Not.Joined (0.23076923 0.76923077) *
        115) Age< 28.5 159   64 Joined (0.59748428 0.40251572)
          230) LOBETS>=0.5 8    0 Joined (1.00000000 0.00000000) *
          231) LOBETS< 0.5 151   64 Joined (0.57615894 0.42384106)
            462) Duration.to.accept.offer< 35.5 26    6 Joined (0.76923077 0.23076923) *
            463) Duration.to.accept.offer>=35.5 125   58 Joined (0.53600000 0.46400000)
              926) Duration.to.accept.offer>=39.5 117   51 Joined (0.56410256 0.43589744)
               1852) Rex.in.Yrs>=3.5 33   10 Joined (0.69696970 0.30303030)
                 3704) Duration.to.accept.offer>=89 7    0 Joined (1.00000000 0.00000000)
                 3705) Duration.to.accept.offer< 89 26   10 Joined (0.61538462 0.3846153
                   7410) Duration.to.accept.offer< 75.5 19    5 Joined (0.73684211 0.263
                   7411) Duration.to.accept.offer>=75.5 7    2 Not.Joined (0.28571429 0.
               1853) Rex.in.Yrs< 3.5 84   41 Joined (0.51190476 0.48809524)
                 3706) Rex.in.Yrs< 2.5 15    4 Joined (0.73333333 0.26666667) *
                 3707) Rex.in.Yrs>=2.5 69   32 Not.Joined (0.46376812 0.53623188)
                   7414) Duration.to.accept.offer>=54 50   24 Joined (0.52000000 0.48000
                    14828) Duration.to.accept.offer< 71.5 23    7 Joined (0.69565217 0.30
                    14829) Duration.to.accept.offer>=71.5 27   10 Not.Joined (0.37037037
                      29658) Duration.to.accept.offer>=91 12    5 Joined (0.58333333 0.4
                      29659) Duration.to.accept.offer< 91 15    3 Not.Joined (0.20000000
```

```
                7415) Duration.to.accept.offer< 54 19    6 Not.Joined (0.31578947 0.68
           927) Duration.to.accept.offer< 39.5 8    1 Not.Joined (0.12500000 0.8750000
  29) Duration.to.accept.offer>=109.5 24    7 Not.Joined (0.29166667 0.70833333) *
15) Duration.to.accept.offer< 25.5 953  379 Joined (0.60230850 0.39769150)
  30) Duration.to.accept.offer< 0.5 80   10 Joined (0.87500000 0.12500000) *
  31) Duration.to.accept.offer>=0.5 873  369 Joined (0.57731959 0.42268041)
    62) Age>=31.5 327  110 Joined (0.66360856 0.33639144)
     124) Joining.BonusYes>=0.5 30    1 Joined (0.96666667 0.03333333) *
     125) Joining.BonusYes< 0.5 297  109 Joined (0.63299663 0.36700337)
       250) Percent.difference.CTC>=-7.07 204   62 Joined (0.69607843 0.30392157)
         500) Age>=36.5 14    0 Joined (1.00000000 0.00000000) *
         501) Age< 36.5 190   62 Joined (0.67368421 0.32631579)
          1002) Candidate.SourceEmployee Referral>=0.5 30    5 Joined (0.83333333 0.16
          1003) Candidate.SourceEmployee Referral< 0.5 160   57 Joined (0.64375000 0.3
            2006) LOBEAS< 0.5 147   49 Joined (0.66666667 0.33333333)
              4012) Percent.difference.CTC< 1.41 110   31 Joined (0.71818182 0.2818181
                8024) Candidate.SourceDirect>=0.5 71   16 Joined (0.77464789 0.2253521
                8025) Candidate.SourceDirect< 0.5 39   15 Joined (0.61538462 0.3846153
                 16050) GenderMale< 0.5 7    0 Joined (1.00000000 0.00000000) *
                 16051) GenderMale>=0.5 32   15 Joined (0.53125000 0.46875000)
                   32102) LocationChennai< 0.5 16    4 Joined (0.75000000 0.25000000)
                   32103) LocationChennai>=0.5 16    5 Not.Joined (0.31250000 0.687500
              4013) Percent.difference.CTC>=1.41 37   18 Joined (0.51351351 0.48648649
                8026) Percent.difference.CTC>=4.2 30   11 Joined (0.63333333 0.366666
                8027) Percent.difference.CTC< 4.2 7    0 Not.Joined (0.00000000 1.0000
            2007) LOBEAS>=0.5 13    5 Not.Joined (0.38461538 0.61538462) *
       251) Percent.difference.CTC< -7.07 93   46 Not.Joined (0.49462366 0.50537634)
         502) Duration.to.accept.offer< 1.5 8    1 Joined (0.87500000 0.12500000) *
         503) Duration.to.accept.offer>=1.5 85   39 Not.Joined (0.45882353 0.54117647)
          1006) Percent.difference.CTC< -7.22 77   38 Not.Joined (0.49350649 0.5064935
            2012) Percent.difference.CTC>=-9.17 14    3 Joined (0.78571429 0.21428571)
            2013) Percent.difference.CTC< -9.17 63   27 Not.Joined (0.42857143 0.5714
              4026) Offered.bandE1>=0.5 37   17 Joined (0.54054054 0.45945946)
                8052) Candidate.SourceDirect>=0.5 12    2 Joined (0.83333333 0.166666
                8053) Candidate.SourceDirect< 0.5 25   10 Not.Joined (0.40000000 0.600
                 16106) Percent.difference.CTC>=-16.25 15    6 Joined (0.60000000 0.40
                 16107) Percent.difference.CTC< -16.25 10    1 Not.Joined (0.10000000
              4027) Offered.bandE1< 0.5 26    7 Not.Joined (0.26923077 0.73076923)
                8054) Percent.difference.CTC< -13.965 8    3 Joined (0.62500000 0.375
                8055) Percent.difference.CTC>=-13.965 18    2 Not.Joined (0.11111111
          1007) Percent.difference.CTC>=-7.22 8    1 Not.Joined (0.12500000 0.8750000
    63) Age< 31.5 546  259 Joined (0.52564103 0.47435897)
     126) DOJ.ExtendedYes>=0.5 242   96 Joined (0.60330579 0.39669421)
       252) Candidate.SourceEmployee Referral>=0.5 26    4 Joined (0.84615385 0.15384
       253) Candidate.SourceEmployee Referral< 0.5 216   92 Joined (0.57407407 0.4259
         506) Notice.period< 67.5 176   67 Joined (0.61931818 0.38068182)
          1012) Age>=30.5 12    1 Joined (0.91666667 0.08333333) *
          1013) Age< 30.5 164   66 Joined (0.59756098 0.40243902)
            2026) LOBBFSI< 0.5 152   58 Joined (0.61842105 0.38157895)
              4052) LOBERS< 0.5 55   16 Joined (0.70909091 0.29090909) *
              4053) LOBERS>=0.5 97   42 Joined (0.56701031 0.43298969)
                8106) Duration.to.accept.offer>=1.5 87   35 Joined (0.59770115 0.4022
                8107) Duration.to.accept.offer< 1.5 10    3 Not.Joined (0.30000000 0.
            2027) LOBBFSI>=0.5 12    4 Not.Joined (0.33333333 0.66666667) *
         507) Notice.period>=67.5 40   15 Not.Joined (0.37500000 0.62500000)
          1014) LOBETS>=0.5 10    3 Joined (0.70000000 0.30000000) *
          1015) LOBETS< 0.5 30    8 Not.Joined (0.26666667 0.73333333) *
     127) DOJ.ExtendedYes< 0.5 304  141 Not.Joined (0.46381579 0.53618421)
       254) Rex.in.Yrs< 2.5 35    7 Joined (0.80000000 0.20000000) *
       255) Rex.in.Yrs>=2.5 269  113 Not.Joined (0.42007435 0.57992565)
         510) Duration.to.accept.offer< 3.5 88   42 Joined (0.52272727 0.47727273)
          1020) Rex.in.Yrs>=4.5 30    9 Joined (0.70000000 0.30000000)
            2040) Offered.bandE1>=0.5 9    0 Joined (1.00000000 0.00000000) *
            2041) Offered.bandE1< 0.5 21    9 Joined (0.57142857 0.42857143)
              4082) Percent.difference.CTC>=-5.155 14    4 Joined (0.71428571 0.28571
              4083) Percent.difference.CTC< -5.155 7    2 Not.Joined (0.28571429 0.71
          1021) Rex.in.Yrs< 4.5 58   25 Not.Joined (0.43103448 0.56896552)
            2042) LocationChennai>=0.5 27   13 Joined (0.51851852 0.48148148)
              4084) Candidate.SourceDirect>=0.5 20    8 Joined (0.60000000 0.40000000
              4085) Candidate.SourceDirect< 0.5 7    2 Not.Joined (0.28571429 0.71428
            2043) LocationChennai< 0.5 31   11 Not.Joined (0.35483871 0.64516129) *
         511) Duration.to.accept.offer>=3.5 181  67 Not.Joined (0.37016575 0.6298342
          1022) LOBCSMP>=0.5 12    3 Joined (0.75000000 0.25000000) *
          1023) LOBCSMP< 0.5 169   58 Not.Joined (0.34319527 0.65680473)
            2046) Percent.difference.CTC>=-7.07 113   45 Not.Joined (0.39823009 0.601
              4092) Age>=28.5 49   23 Joined (0.53061224 0.46938776)
                8184) Duration.to.accept.offer< 4.5 11    3 Joined (0.72727273 0.2727
                8185) Duration.to.accept.offer>=4.5 38   18 Not.Joined (0.47368421 0.
                 16370) Duration.to.accept.offer>=7.5 23    9 Joined (0.60869565 0.39
                 16371) Duration.to.accept.offer< 7.5 15    4 Not.Joined (0.26666667
              4093) Age< 28.5 64   19 Not.Joined (0.29687500 0.70312500) *
            2047) Percent.difference.CTC< -7.07 56   13 Not.Joined (0.23214286 0.7678
```

1.3.2 2. The prediction and confusion Matrix on train data.

The syntax for prediction in caret is almost similar expect the the **type** attribute expects input as **'raw'** or **'prob'**. In case of prob, the predicted value holds the probability of both positive and negative class.

```
In [14]: #Missing code. May result in error
         levels(dt_train_df$Status) <- make.names(levels(factor(dt_train_df$Status)))
         caretPredictedClass <- predict(object = dt_caret_model, dt_train_df[,1:12], type = 'ra
         confusionMatrix(caretPredictedClass,dt_train_df$Status)
```

```
Confusion Matrix and Statistics
```

```
            Reference
Prediction   Joined Not.Joined
  Joined       5735       1017
  Not.Joined    116        329

               Accuracy : 0.8426
                 95% CI : (0.834, 0.8509)
    No Information Rate : 0.813
    P-Value [Acc > NIR] : 2.667e-11

                  Kappa : 0.3026
 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.9802
            Specificity : 0.2444
         Pos Pred Value : 0.8494
         Neg Pred Value : 0.7393
             Prevalence : 0.8130
         Detection Rate : 0.7969
   Detection Prevalence : 0.9382
      Balanced Accuracy : 0.6123

       'Positive' Class : Joined
```

1.3.3 3. Confusion Matrix on the test data

The **predict** function is used to get the predicted class on the new dataset.

```
In [15]: levels(dt_test_data$Status) <- make.names(levels(factor(dt_test_data$Status)))
         dtCaretTestPredictedClass = predict(dt_caret_model, dt_test_data, type = "raw")
         confusionMatrix(dtCaretTestPredictedClass,dt_test_data$Status)
```

```
Confusion Matrix and Statistics

            Reference
Prediction   Joined Not.Joined
  Joined       1415        276
  Not.Joined     47         60

               Accuracy : 0.8204
                 95% CI : (0.8018, 0.8378)
    No Information Rate : 0.8131
    P-Value [Acc > NIR] : 0.2256

                  Kappa : 0.1985
```

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

            Sensitivity : 0.9679
            Specificity : 0.1786
         Pos Pred Value : 0.8368
         Neg Pred Value : 0.5607
             Prevalence : 0.8131
         Detection Rate : 0.7870
   Detection Prevalence : 0.9405
      Balanced Accuracy : 0.5732

       'Positive' Class : Joined
```

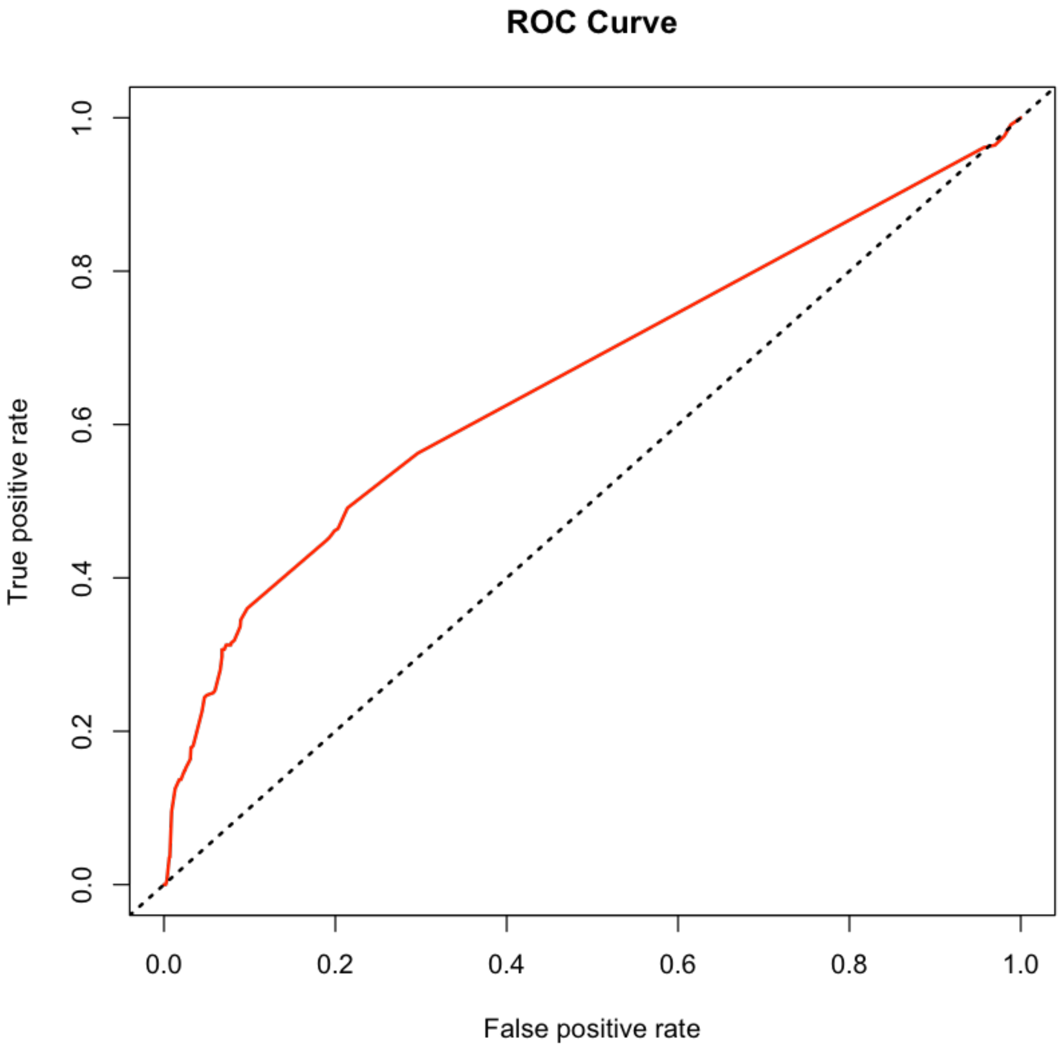
1.3.4 4. ROC Plot on the test data

ROCR package can be used to evaluate the model performace on the test data. The same package can also be used to get the model performace on the test data.

```
In [16]: #error in below line
         dtCaretTestPredictedProbability = predict(dt_caret_model, dt_test_data, type = "prob")
         dtPredObj <- prediction(dtCaretTestPredictedProbability[2],dt_test_data$Status)
         dtPerfObj <- performance(dtPredObj, "tpr","fpr")
         #dev.off()
         plot(dtPerfObj,main = "ROC Curve",col = 2,lwd = 2)
         abline(a = 0,b = 1,lwd = 2,lty = 3,col = "black")
         performance(dtPredObj, "auc")
```

```
An object of class "performance"
Slot "x.name":
[1] "None"

Slot "y.name":
[1] "Area under the ROC curve"

Slot "alpha.name":
[1] "none"

Slot "x.values":
list()

Slot "y.values":
[[1]]
[1] 0.6590654


Slot "alpha.values":
```

```
list()
```



End of Document