Final Report

Renaming the predictors to appropriate names to improve the readability.

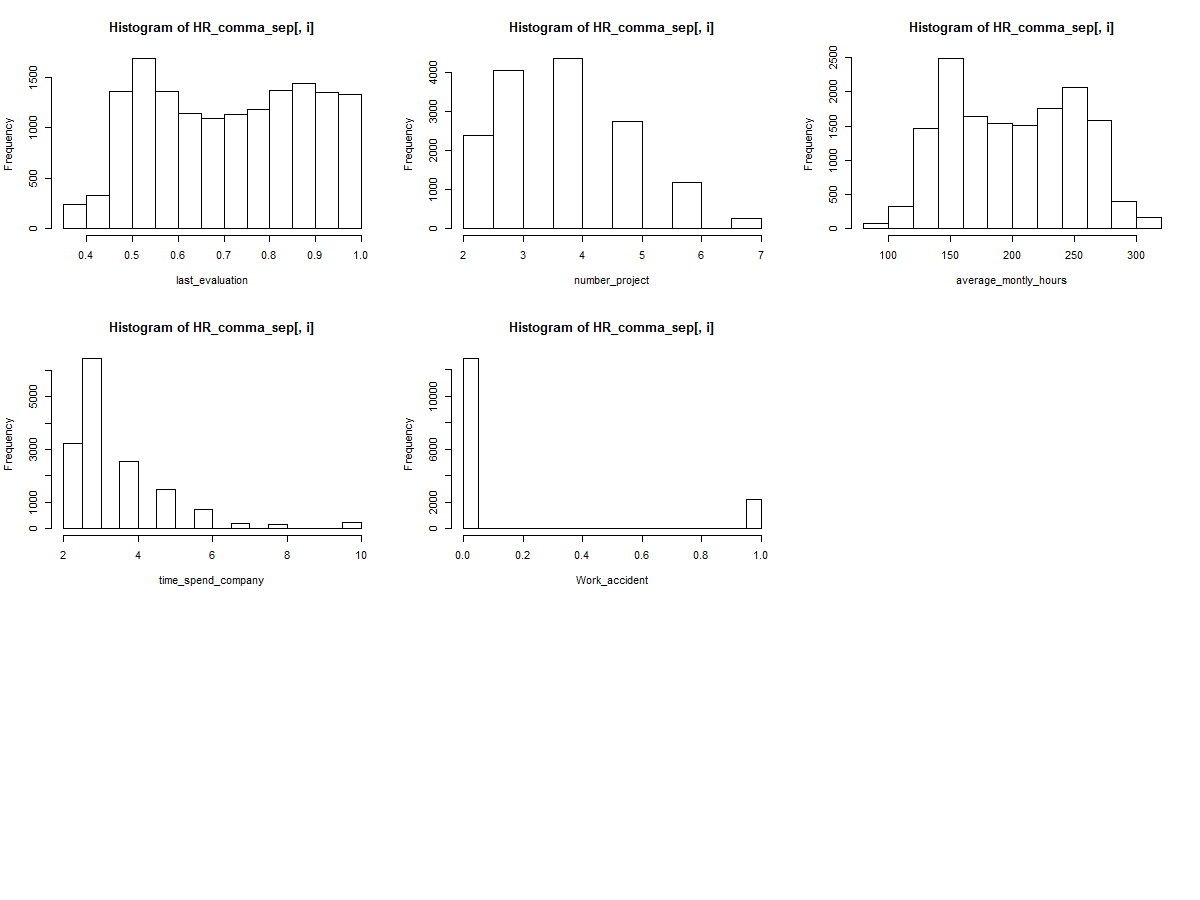
Adding unique identifier to each employee.

Finding the NA values in the table.

## [1] 0

Data Visualizations:

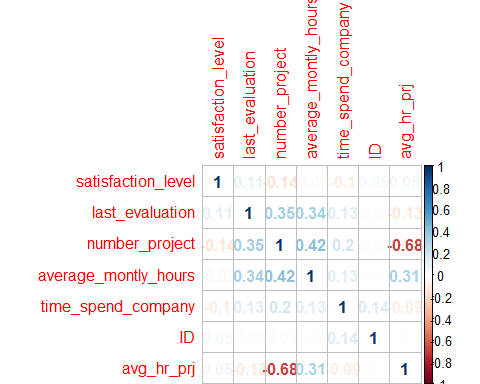
Finding the distribution of variables for the numeric data type

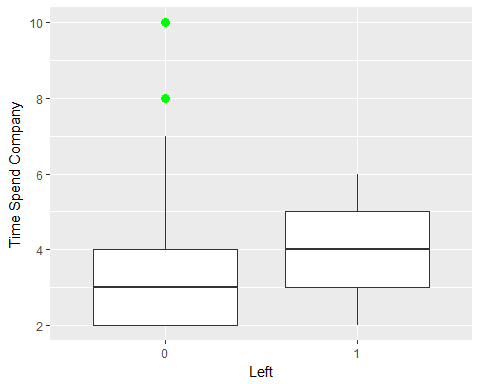
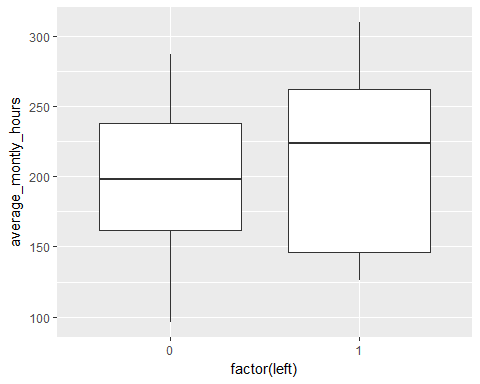
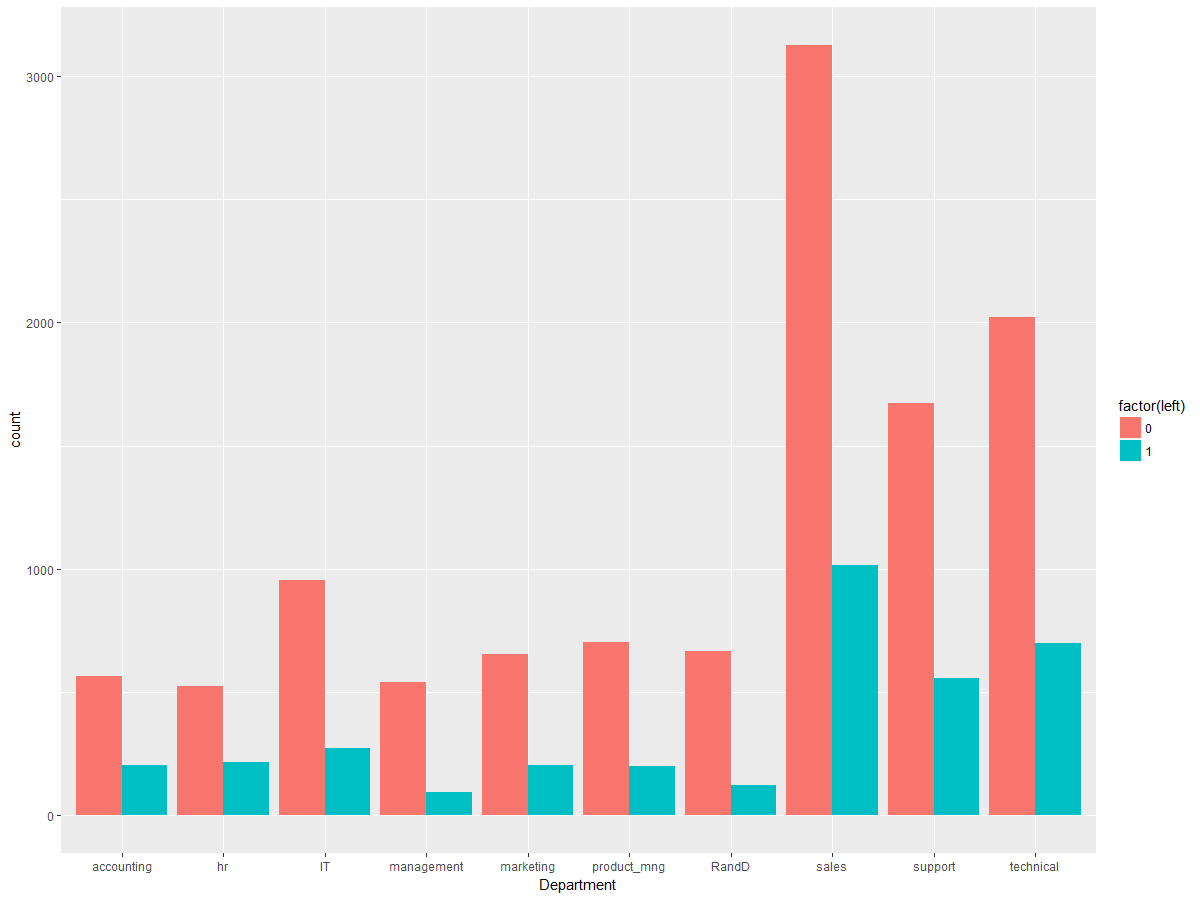
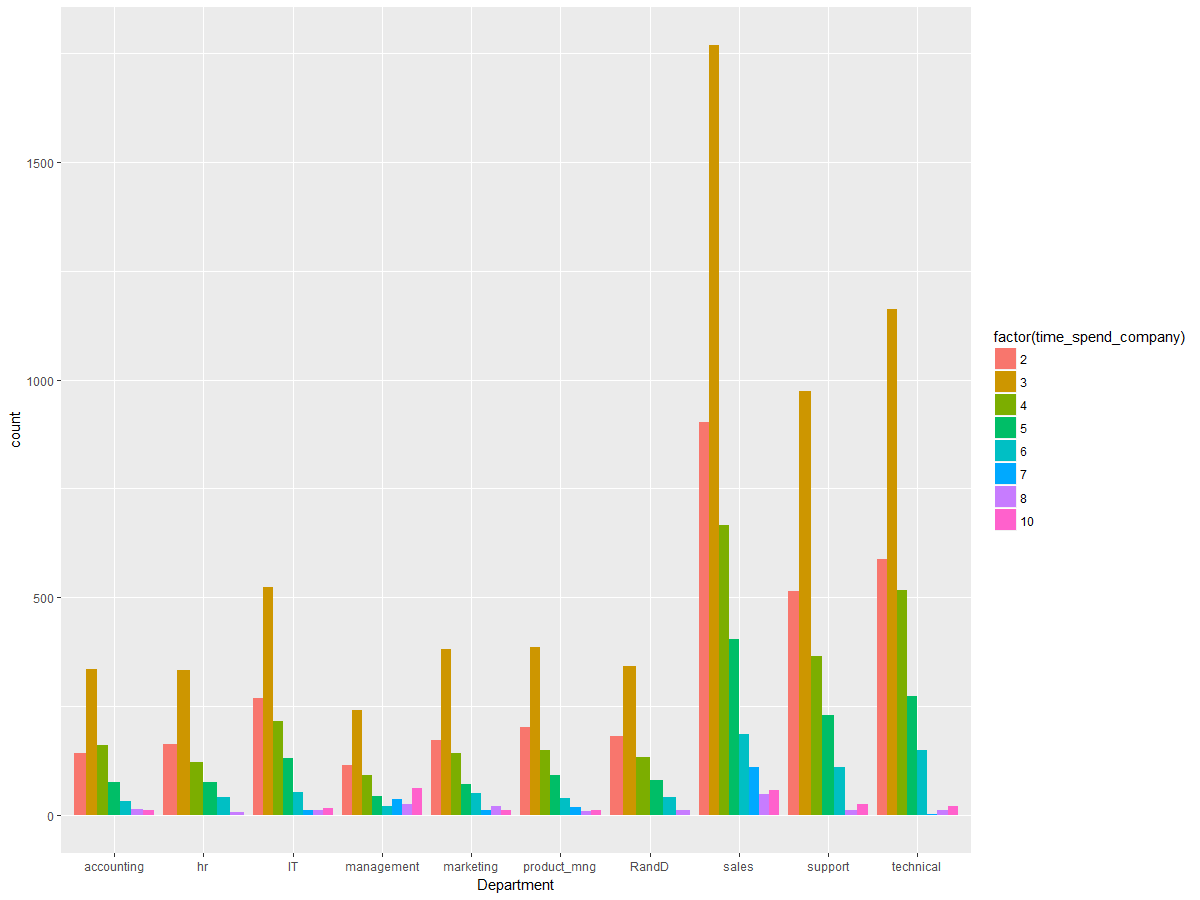


Finding the descriptive statistics.

## satisfaction\_level last\_evaluation number\_project average\_montly\_hours  
## Min. :0.0900 Min. :0.3600 Min. :2.000 Min. : 96.0   
## 1st Qu.:0.4400 1st Qu.:0.5600 1st Qu.:3.000 1st Qu.:156.0   
## Median :0.6400 Median :0.7200 Median :4.000 Median :200.0   
## Mean :0.6128 Mean :0.7161 Mean :3.803 Mean :201.1   
## 3rd Qu.:0.8200 3rd Qu.:0.8700 3rd Qu.:5.000 3rd Qu.:245.0   
## Max. :1.0000 Max. :1.0000 Max. :7.000 Max. :310.0   
##   
## time\_spend\_company Work\_accident left promotion\_last\_5years  
## Min. : 2.000 0:12830 0:11428 0:14680   
## 1st Qu.: 3.000 1: 2169 1: 3571 1: 319   
## Median : 3.000   
## Mean : 3.498   
## 3rd Qu.: 4.000   
## Max. :10.000   
##   
## Department salary ID avg\_hr\_prj   
## sales :4140 high :1237 Min. : 1 Min. : 194.0   
## technical :2720 low :7316 1st Qu.: 3750 1st Qu.: 525.6   
## support :2229 medium:6446 Median : 7500 Median : 648.0   
## IT :1227 Mean : 7500 Mean : 685.0   
## product\_mng: 902 3rd Qu.:11250 3rd Qu.: 816.0   
## marketing : 858 Max. :14999 Max. :1860.0   
## (Other) :2923   
## avg\_hr\_prj\_range HR\_Cat   
## (192,749] :9530 0:9530   
## (749,1.3e+03] :5186 1:5186   
## (1.3e+03,1.86e+03]: 283 2: 283   
##   
##   
##   
##

Finding the correlation between variables.



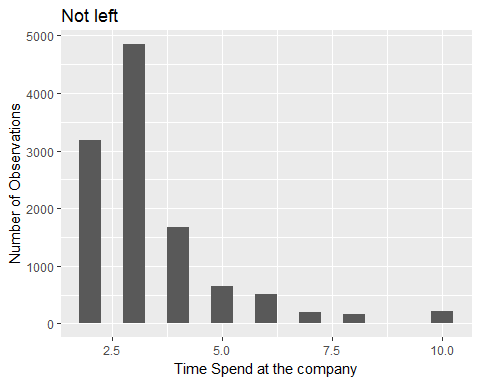
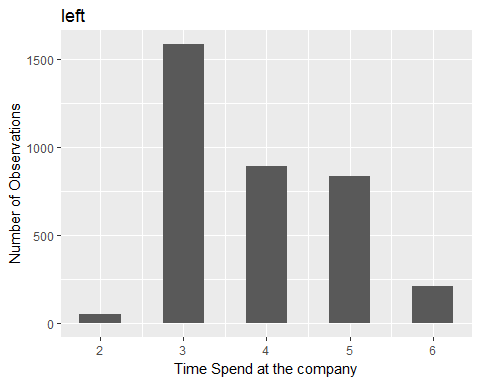
Exploratory Data Analysis (EDA)   

From the above plots we can conclude that there are few outliers in the data. So, finding the total number of observation who spend in company more than 8 years.

There are 376 employees who spend more than 8 years in the company. So, we cannot ignore these observations because majority employees are from sales and Management departments which are crutial departments in this company.

creating new data sets, for the employees who has 'left' and 'non-left' into two seperate tables.

Plotting the histograms for 'time spend in the company' using left and non-left data tables.



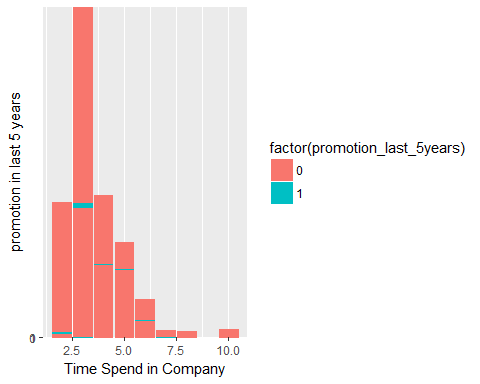
From above observations:

1. People who spend 6 or more years and who spend 2 years at the company are less likely to go;
2. People are more likely to leave when they have spent 3-5 years here;
3. A interesting group: 5-year-group. People who are in this group are more likely to leave than stay;
4. When the years people spent in the company lies in 3-5: the more they've been here, the more likely they leave.

Finding right answers to these questions only based on this dataset could be challenging, but we can guess a little:

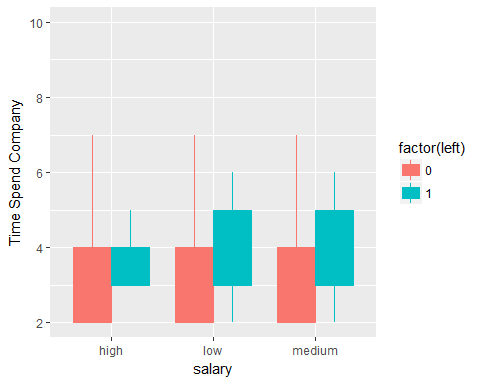
1. When a person spend 6 or more years at a certain company, he/she must be used to it and would not like to go unless he/she have to (so one more question can be raised here: why these people are leaving?)
2. When someone spend 3-5 years at a company, he/she is start to be treated as a 'professional' in his area which make him/her a job-hopper to pursue higher salary

Plot for 'Time Spend in the company' with 'Promotion from last 5 years'

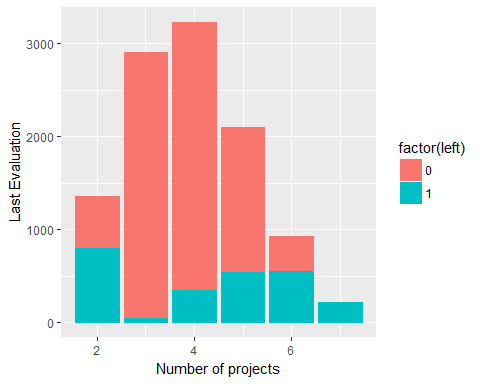
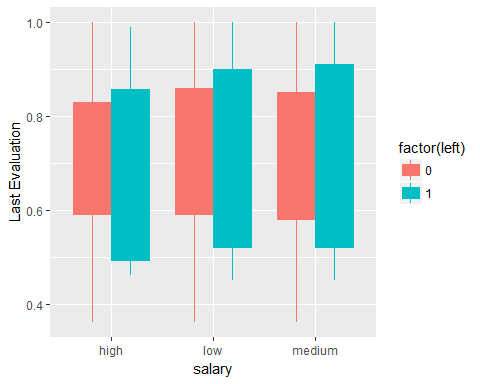
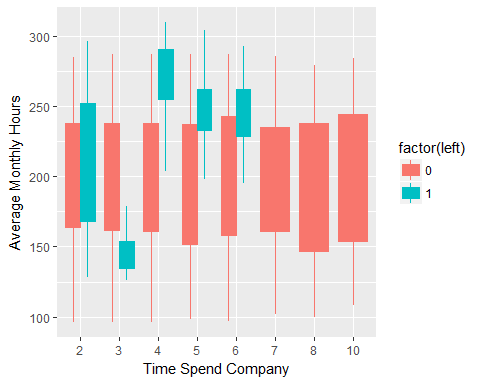
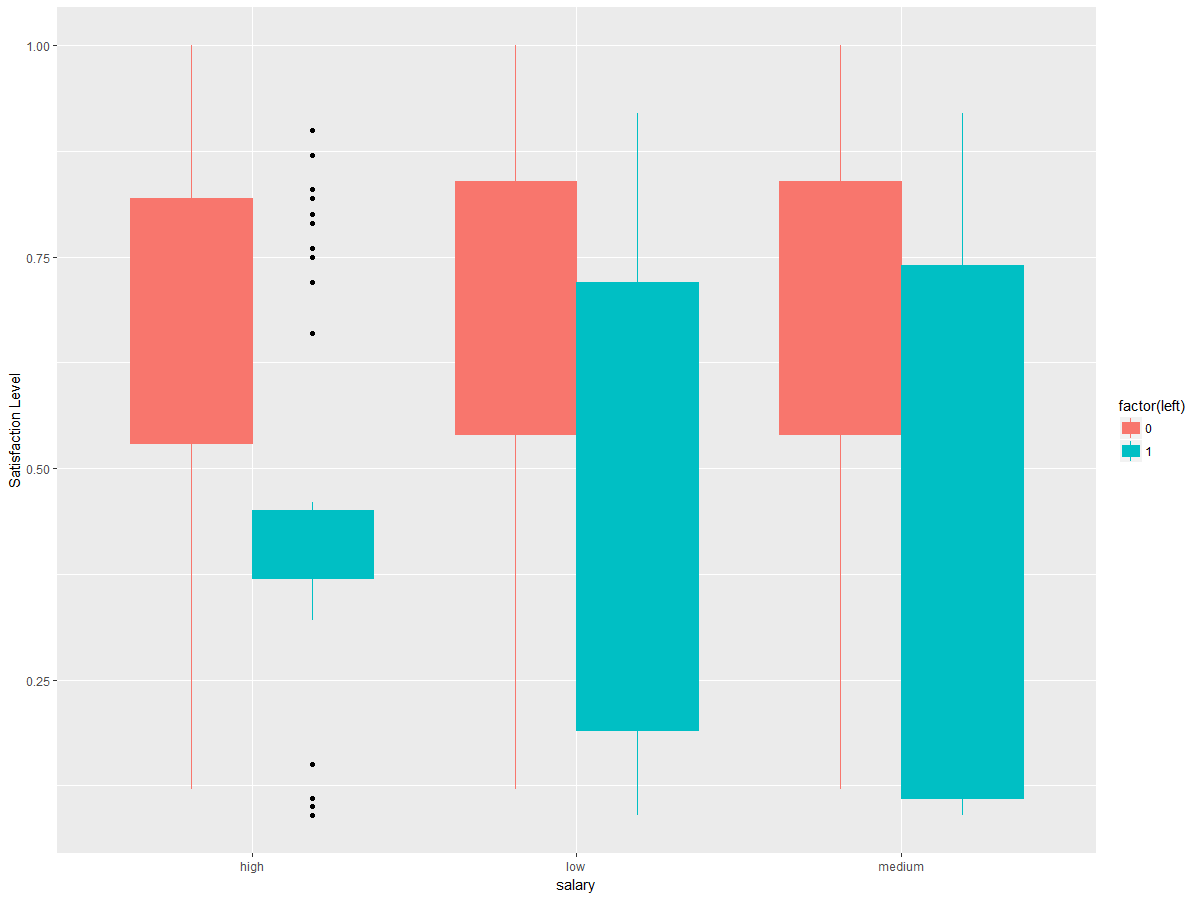


Very less people got promoted even though they are spending more time in the office.

plot salary VS time spend in the company.



From the above plot, we can conclude that low and medium income people are leaving the company.



who are valuable employess??

The evaluation criteria and Monthly hours spend in the company are considered as valuable. Here we are not considering the promotion because very less people got promoted in last 5 years.

For our analysis we are finding the average time an employee spent on each project. Then, we converted the variable into 3 levels.

In general an employee must work for 160 hours per month. We have splitted this variable into 3 levels and then according to the level we have given categories as [0,1,2]

Finding the total valuable employees.

## [1] 4386

There are total of 4386 valuable employees.

Decide who all are valuable employees:

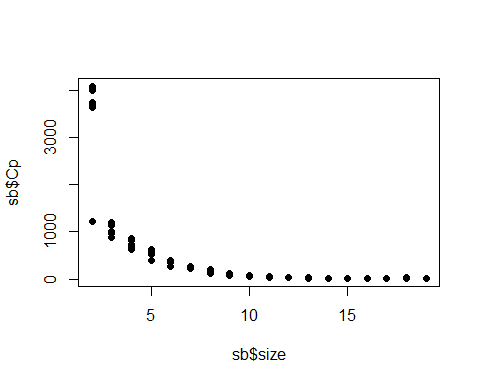
## Department number\_of\_employees number\_of\_employees\_left percent  
## 1 accounting 767 204 3.759804  
## 2 hr 739 215 3.437209  
## 3 IT 1227 273 4.494505  
## 4 management 630 91 6.923077  
## 5 marketing 858 203 4.226601  
## 6 product\_mng 902 198 4.555556  
## 7 RandD 787 121 6.504132  
## 8 sales 4140 1014 4.082840  
## 9 support 2229 555 4.016216  
## 10 technical 2720 697 3.902439

From the above table we can say that "management' and 'R and D' people are leaving more compared to other departments. The 'management' people are staying for long time and working for more hours in the company. The 'management' and 'R and D' people have more functional knowledge compared to other department. So, we are considering these people as valuable.

Feature Selection

Cp Model:

Plotting the cp method and finding the best subset model.



## 1 2 3 4 5 6 7 8 9 A B C   
## TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE FALSE TRUE   
## D E F G H I   
## TRUE FALSE FALSE FALSE TRUE TRUE

Forward selection:

Finding the summary for forward selection.

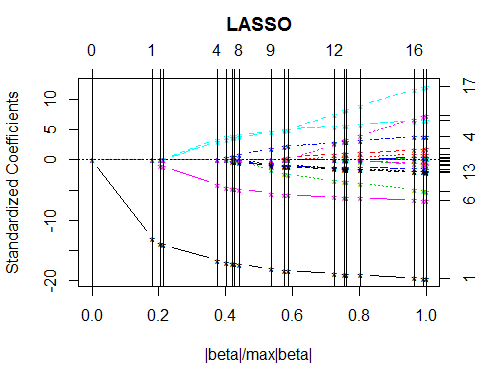
##   
## Call:  
## glm(formula = left ~ satisfaction\_level + salary + Work\_accident +   
## time\_spend\_company + number\_project + average\_montly\_hours +   
## promotion\_last\_5years + Department + last\_evaluation, family = binomial(link = "logit"),   
## data = HR\_comma\_sep)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.2248 -0.6645 -0.4026 -0.1177 3.0688   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.4762862 0.1938373 -7.616 2.61e-14 \*\*\*  
## satisfaction\_level -4.1356889 0.0980538 -42.178 < 2e-16 \*\*\*  
## salarylow 1.9440627 0.1286272 15.114 < 2e-16 \*\*\*  
## salarymedium 1.4132244 0.1293534 10.925 < 2e-16 \*\*\*  
## Work\_accident1 -1.5298283 0.0895473 -17.084 < 2e-16 \*\*\*  
## time\_spend\_company 0.2677537 0.0155736 17.193 < 2e-16 \*\*\*  
## number\_project -0.3150787 0.0213248 -14.775 < 2e-16 \*\*\*  
## average\_montly\_hours 0.0044603 0.0005161 8.643 < 2e-16 \*\*\*  
## promotion\_last\_5years1 -1.4301364 0.2574958 -5.554 2.79e-08 \*\*\*  
## Departmenthr 0.2323779 0.1313084 1.770 0.07678 .   
## DepartmentIT -0.1807179 0.1221276 -1.480 0.13894   
## Departmentmanagement -0.4484236 0.1598254 -2.806 0.00502 \*\*   
## Departmentmarketing -0.0120882 0.1319304 -0.092 0.92700   
## Departmentproduct\_mng -0.1532529 0.1301538 -1.177 0.23901   
## DepartmentRandD -0.5823659 0.1448848 -4.020 5.83e-05 \*\*\*  
## Departmentsales -0.0387859 0.1024006 -0.379 0.70486   
## Departmentsupport 0.0500251 0.1092834 0.458 0.64713   
## Departmenttechnical 0.0701464 0.1065379 0.658 0.51027   
## last\_evaluation 0.7309032 0.1491787 4.900 9.61e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 16465 on 14998 degrees of freedom  
## Residual deviance: 12850 on 14980 degrees of freedom  
## AIC: 12888  
##   
## Number of Fisher Scoring iterations: 5

Backward Selection:

##   
## Call:  
## glm(formula = left ~ satisfaction\_level + last\_evaluation + number\_project +   
## average\_montly\_hours + time\_spend\_company + Work\_accident +   
## promotion\_last\_5years + Department + salary, family = binomial(link = "logit"),   
## data = HR\_comma\_sep)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.2248 -0.6645 -0.4026 -0.1177 3.0688   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.4762862 0.1938373 -7.616 2.61e-14 \*\*\*  
## satisfaction\_level -4.1356889 0.0980538 -42.178 < 2e-16 \*\*\*  
## last\_evaluation 0.7309032 0.1491787 4.900 9.61e-07 \*\*\*  
## number\_project -0.3150787 0.0213248 -14.775 < 2e-16 \*\*\*  
## average\_montly\_hours 0.0044603 0.0005161 8.643 < 2e-16 \*\*\*  
## time\_spend\_company 0.2677537 0.0155736 17.193 < 2e-16 \*\*\*  
## Work\_accident1 -1.5298283 0.0895473 -17.084 < 2e-16 \*\*\*  
## promotion\_last\_5years1 -1.4301364 0.2574958 -5.554 2.79e-08 \*\*\*  
## Departmenthr 0.2323779 0.1313084 1.770 0.07678 .   
## DepartmentIT -0.1807179 0.1221276 -1.480 0.13894   
## Departmentmanagement -0.4484236 0.1598254 -2.806 0.00502 \*\*   
## Departmentmarketing -0.0120882 0.1319304 -0.092 0.92700   
## Departmentproduct\_mng -0.1532529 0.1301538 -1.177 0.23901   
## DepartmentRandD -0.5823659 0.1448848 -4.020 5.83e-05 \*\*\*  
## Departmentsales -0.0387859 0.1024006 -0.379 0.70486   
## Departmentsupport 0.0500251 0.1092834 0.458 0.64713   
## Departmenttechnical 0.0701464 0.1065379 0.658 0.51027   
## salarylow 1.9440627 0.1286272 15.114 < 2e-16 \*\*\*  
## salarymedium 1.4132244 0.1293534 10.925 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 16465 on 14998 degrees of freedom  
## Residual deviance: 12850 on 14980 degrees of freedom  
## AIC: 12888  
##   
## Number of Fisher Scoring iterations: 5

LASSO:

Plotting the LASSO,



From the above feature selection techniques we decided to consider the backward selection technique and fit the model.

Sampling the Data Set:

We are using the Stratified sampling for diving the data set into Train and Test Datasets. Stratified Sampling will use the strata for dividing the dataset. The biggest advantage of the Stratified Random Sampling is that it reduces bias.

Fitting Models

Genalized Linear Model:

The Generalized Linear Model is mainly used for classification. The GLM generalizes linear regression by allowing the linear model to be related to the response variable via a link function. The link function provides the relationship between the linear predictor and the mean of the distribution function. As our response variable is binary we tried the GLM with Binomial Distribution.

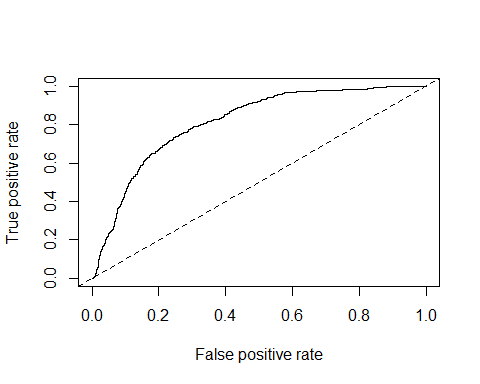
We can check the summary statistics after fitting the model:

##   
## Call:  
## glm(formula = left ~ ., family = binomial(link = "logit"), data = train\_HR)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.2337 -0.6625 -0.3982 -0.1131 2.9936   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.5094415 0.2152372 -7.013 2.33e-12 \*\*\*  
## satisfaction\_level -4.2152964 0.1105460 -38.132 < 2e-16 \*\*\*  
## last\_evaluation 0.7908794 0.1671411 4.732 2.23e-06 \*\*\*  
## number\_project -0.3277439 0.0238623 -13.735 < 2e-16 \*\*\*  
## average\_montly\_hours 0.0047127 0.0005782 8.151 3.60e-16 \*\*\*  
## time\_spend\_company 0.2668274 0.0173935 15.341 < 2e-16 \*\*\*  
## Work\_accident -1.4256792 0.0975332 -14.617 < 2e-16 \*\*\*  
## promotion\_last\_5years -1.2930664 0.2749530 -4.703 2.57e-06 \*\*\*  
## saleshr 0.2682942 0.1456479 1.842 0.065464 .   
## salesIT -0.2065620 0.1363738 -1.515 0.129855   
## salesmanagement -0.5569226 0.1797127 -3.099 0.001942 \*\*   
## salesmarketing 0.0256231 0.1469513 0.174 0.861579   
## salesproduct\_mng -0.1456615 0.1452384 -1.003 0.315903   
## salesRandD -0.6011740 0.1624358 -3.701 0.000215 \*\*\*  
## salessales -0.0133273 0.1130714 -0.118 0.906174   
## salessupport 0.0722732 0.1207990 0.598 0.549645   
## salestechnical 0.1572406 0.1178677 1.334 0.182190   
## salarylow 1.9672462 0.1421543 13.839 < 2e-16 \*\*\*  
## salarymedium 1.4093813 0.1429419 9.860 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 13205 on 11999 degrees of freedom  
## Residual deviance: 10271 on 11981 degrees of freedom  
## AIC: 10309  
##   
## Number of Fisher Scoring iterations: 5

Predicting the values for test data:

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 2201 557  
## 1 98 143  
##   
## Accuracy : 0.7816   
## 95% CI : (0.7664, 0.7963)  
## No Information Rate : 0.7666   
## P-Value [Acc > NIR] : 0.0267   
##   
## Kappa : 0.2094   
## Mcnemar's Test P-Value : <2e-16   
##   
## Sensitivity : 0.9574   
## Specificity : 0.2043   
## Pos Pred Value : 0.7980   
## Neg Pred Value : 0.5934   
## Prevalence : 0.7666   
## Detection Rate : 0.7339   
## Detection Prevalence : 0.9196   
## Balanced Accuracy : 0.5808   
##   
## 'Positive' Class : 0   
##

ROC Curve



Classification and Regression Trees(CART) Algorithm:

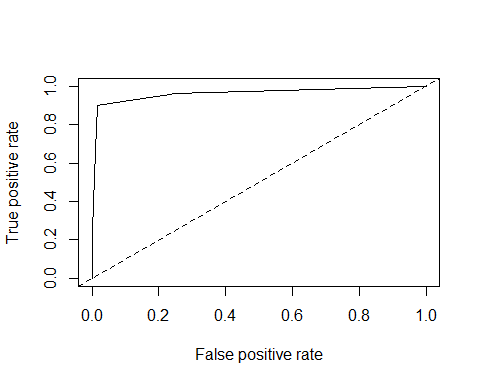
A CART tree is a binary decision tree that is constructed by splitting a node into two child nodes repeatedly, beginning with the root node that contains the whole learning sample.

## Call:  
## rpart(formula = left ~ ., data = train\_HR, method = "class")  
## n= 12000   
##   
## CP nsplit rel error xerror xstd  
## 1 0.26018809 0 1.0000000 1.0000000 0.016278125  
## 2 0.18025078 1 0.7398119 0.7398119 0.014562770  
## 3 0.07645420 3 0.3793103 0.3793103 0.010960299  
## 4 0.05154998 5 0.2264020 0.2264020 0.008636361  
## 5 0.03239289 6 0.1748520 0.1755486 0.007653584  
## 6 0.01532567 7 0.1424591 0.1431557 0.006939371  
## 7 0.01000000 8 0.1271334 0.1288750 0.006595791  
##   
## Variable importance  
## satisfaction\_level number\_project average\_montly\_hours   
## 35 18 17   
## last\_evaluation time\_spend\_company   
## 16 13   
##   
## Node number 1: 12000 observations, complexity param=0.2601881  
## predicted class=0 expected loss=0.23925 P(node) =1  
## class counts: 9129 2871  
## probabilities: 0.761 0.239   
## left son=2 (8677 obs) right son=3 (3323 obs)  
## Primary splits:  
## satisfaction\_level < 0.465 to the right, improve=1279.7800, (0 missing)  
## number\_project < 2.5 to the right, improve= 794.4772, (0 missing)  
## time\_spend\_company < 2.5 to the left, improve= 329.2523, (0 missing)  
## average\_montly\_hours < 286.5 to the left, improve= 320.6998, (0 missing)  
## last\_evaluation < 0.575 to the right, improve= 181.7632, (0 missing)  
## Surrogate splits:  
## number\_project < 2.5 to the right, agree=0.795, adj=0.260, (0 split)  
## average\_montly\_hours < 275.5 to the left, agree=0.755, adj=0.115, (0 split)  
## last\_evaluation < 0.485 to the right, agree=0.742, adj=0.067, (0 split)  
##   
## Node number 2: 8677 observations, complexity param=0.0764542  
## predicted class=0 expected loss=0.09634666 P(node) =0.7230833  
## class counts: 7841 836  
## probabilities: 0.904 0.096   
## left son=4 (7085 obs) right son=5 (1592 obs)  
## Primary splits:  
## time\_spend\_company < 4.5 to the left, improve=520.46260, (0 missing)  
## last\_evaluation < 0.815 to the left, improve=178.62090, (0 missing)  
## average\_montly\_hours < 216.5 to the left, improve=141.12140, (0 missing)  
## number\_project < 4.5 to the left, improve= 93.47856, (0 missing)  
## satisfaction\_level < 0.715 to the left, improve= 69.00932, (0 missing)  
## Surrogate splits:  
## last\_evaluation < 0.995 to the left, agree=0.823, adj=0.034, (0 split)  
## average\_montly\_hours < 298 to the left, agree=0.817, adj=0.003, (0 split)  
##   
## Node number 3: 3323 observations, complexity param=0.1802508  
## predicted class=1 expected loss=0.3876016 P(node) =0.2769167  
## class counts: 1288 2035  
## probabilities: 0.388 0.612   
## left son=6 (1929 obs) right son=7 (1394 obs)  
## Primary splits:  
## number\_project < 2.5 to the right, improve=355.6059, (0 missing)  
## time\_spend\_company < 4.5 to the right, improve=300.6491, (0 missing)  
## satisfaction\_level < 0.115 to the right, improve=272.2730, (0 missing)  
## last\_evaluation < 0.575 to the right, improve=131.4695, (0 missing)  
## average\_montly\_hours < 161.5 to the right, improve=128.7959, (0 missing)  
## Surrogate splits:  
## satisfaction\_level < 0.355 to the left, agree=0.887, adj=0.730, (0 split)  
## average\_montly\_hours < 161.5 to the right, agree=0.860, adj=0.667, (0 split)  
## last\_evaluation < 0.575 to the right, agree=0.856, adj=0.656, (0 split)  
## time\_spend\_company < 3.5 to the right, agree=0.847, adj=0.634, (0 split)  
## sales splits as LRLLLLLLLL, agree=0.584, adj=0.008, (0 split)  
##   
## Node number 4: 7085 observations  
## predicted class=0 expected loss=0.01425547 P(node) =0.5904167  
## class counts: 6984 101  
## probabilities: 0.986 0.014   
##   
## Node number 5: 1592 observations, complexity param=0.0764542  
## predicted class=0 expected loss=0.4616834 P(node) =0.1326667  
## class counts: 857 735  
## probabilities: 0.538 0.462   
## left son=10 (617 obs) right son=11 (975 obs)  
## Primary splits:  
## last\_evaluation < 0.815 to the left, improve=349.1980, (0 missing)  
## average\_montly\_hours < 216.5 to the left, improve=299.4059, (0 missing)  
## time\_spend\_company < 6.5 to the right, improve=214.2593, (0 missing)  
## satisfaction\_level < 0.715 to the left, improve=187.7324, (0 missing)  
## number\_project < 3.5 to the left, improve=164.6126, (0 missing)  
## Surrogate splits:  
## average\_montly\_hours < 215.5 to the left, agree=0.744, adj=0.339, (0 split)  
## number\_project < 3.5 to the left, agree=0.710, adj=0.251, (0 split)  
## satisfaction\_level < 0.705 to the left, agree=0.705, adj=0.240, (0 split)  
## time\_spend\_company < 6.5 to the right, agree=0.683, adj=0.182, (0 split)  
## Work\_accident < 0.5 to the right, agree=0.643, adj=0.079, (0 split)  
##   
## Node number 6: 1929 observations, complexity param=0.1802508  
## predicted class=0 expected loss=0.4157595 P(node) =0.16075  
## class counts: 1127 802  
## probabilities: 0.584 0.416   
## left son=12 (1219 obs) right son=13 (710 obs)  
## Primary splits:  
## satisfaction\_level < 0.115 to the right, improve=767.0086, (0 missing)  
## average\_montly\_hours < 242.5 to the left, improve=453.1309, (0 missing)  
## number\_project < 5.5 to the left, improve=408.5446, (0 missing)  
## last\_evaluation < 0.765 to the left, improve=326.8159, (0 missing)  
## time\_spend\_company < 3.5 to the left, improve=117.5403, (0 missing)  
## Surrogate splits:  
## average\_montly\_hours < 242.5 to the left, agree=0.867, adj=0.639, (0 split)  
## number\_project < 5.5 to the left, agree=0.842, adj=0.570, (0 split)  
## last\_evaluation < 0.765 to the left, agree=0.788, adj=0.425, (0 split)  
##   
## Node number 7: 1394 observations, complexity param=0.03239289  
## predicted class=1 expected loss=0.115495 P(node) =0.1161667  
## class counts: 161 1233  
## probabilities: 0.115 0.885   
## left son=14 (101 obs) right son=15 (1293 obs)  
## Primary splits:  
## last\_evaluation < 0.575 to the right, improve=155.463100, (0 missing)  
## average\_montly\_hours < 162 to the right, improve=135.775000, (0 missing)  
## satisfaction\_level < 0.355 to the left, improve=107.822500, (0 missing)  
## time\_spend\_company < 3.5 to the right, improve= 66.014810, (0 missing)  
## salary splits as LRR, improve= 7.910967, (0 missing)  
## Surrogate splits:  
## average\_montly\_hours < 162 to the right, agree=0.945, adj=0.238, (0 split)  
## satisfaction\_level < 0.355 to the left, agree=0.942, adj=0.198, (0 split)  
## time\_spend\_company < 3.5 to the right, agree=0.940, adj=0.168, (0 split)  
##   
## Node number 10: 617 observations  
## predicted class=0 expected loss=0.04538088 P(node) =0.05141667  
## class counts: 589 28  
## probabilities: 0.955 0.045   
##   
## Node number 11: 975 observations, complexity param=0.05154998  
## predicted class=1 expected loss=0.2748718 P(node) =0.08125  
## class counts: 268 707  
## probabilities: 0.275 0.725   
## left son=22 (182 obs) right son=23 (793 obs)  
## Primary splits:  
## average\_montly\_hours < 216.5 to the left, improve=178.60120, (0 missing)  
## time\_spend\_company < 6.5 to the right, improve=164.78540, (0 missing)  
## satisfaction\_level < 0.715 to the left, improve=129.82750, (0 missing)  
## number\_project < 3.5 to the left, improve= 91.67479, (0 missing)  
## salary splits as LRL, improve= 24.73369, (0 missing)  
## Surrogate splits:  
## time\_spend\_company < 6.5 to the right, agree=0.862, adj=0.258, (0 split)  
## number\_project < 3.5 to the left, agree=0.847, adj=0.181, (0 split)  
## satisfaction\_level < 0.715 to the left, agree=0.846, adj=0.176, (0 split)  
##   
## Node number 12: 1219 observations  
## predicted class=0 expected loss=0.0754717 P(node) =0.1015833  
## class counts: 1127 92  
## probabilities: 0.925 0.075   
##   
## Node number 13: 710 observations  
## predicted class=1 expected loss=0 P(node) =0.05916667  
## class counts: 0 710  
## probabilities: 0.000 1.000   
##   
## Node number 14: 101 observations  
## predicted class=0 expected loss=0.03960396 P(node) =0.008416667  
## class counts: 97 4  
## probabilities: 0.960 0.040   
##   
## Node number 15: 1293 observations  
## predicted class=1 expected loss=0.04949729 P(node) =0.10775  
## class counts: 64 1229  
## probabilities: 0.049 0.951   
##   
## Node number 22: 182 observations  
## predicted class=0 expected loss=0.09340659 P(node) =0.01516667  
## class counts: 165 17  
## probabilities: 0.907 0.093   
##   
## Node number 23: 793 observations, complexity param=0.01532567  
## predicted class=1 expected loss=0.1298865 P(node) =0.06608333  
## class counts: 103 690  
## probabilities: 0.130 0.870   
## left son=46 (44 obs) right son=47 (749 obs)  
## Primary splits:  
## time\_spend\_company < 6.5 to the right, improve=70.53844, (0 missing)  
## satisfaction\_level < 0.715 to the left, improve=62.57871, (0 missing)  
## number\_project < 3.5 to the left, improve=51.79525, (0 missing)  
## promotion\_last\_5years < 0.5 to the right, improve=12.01298, (0 missing)  
## salary splits as LRR, improve=11.96874, (0 missing)  
## Surrogate splits:  
## satisfaction\_level < 0.575 to the left, agree=0.952, adj=0.136, (0 split)  
## promotion\_last\_5years < 0.5 to the right, agree=0.952, adj=0.136, (0 split)  
##   
## Node number 46: 44 observations  
## predicted class=0 expected loss=0 P(node) =0.003666667  
## class counts: 44 0  
## probabilities: 1.000 0.000   
##   
## Node number 47: 749 observations  
## predicted class=1 expected loss=0.0787717 P(node) =0.06241667  
## class counts: 59 690  
## probabilities: 0.079 0.921

Predicting for Test data using CART model:

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 2260 69  
## 1 39 631  
##   
## Accuracy : 0.964   
## 95% CI : (0.9567, 0.9704)  
## No Information Rate : 0.7666   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.8978   
## Mcnemar's Test P-Value : 0.005262   
##   
## Sensitivity : 0.9830   
## Specificity : 0.9014   
## Pos Pred Value : 0.9704   
## Neg Pred Value : 0.9418   
## Prevalence : 0.7666   
## Detection Rate : 0.7536   
## Detection Prevalence : 0.7766   
## Balanced Accuracy : 0.9422   
##   
## 'Positive' Class : 0   
##

ROC Curve:



Random Forest Algorithm:

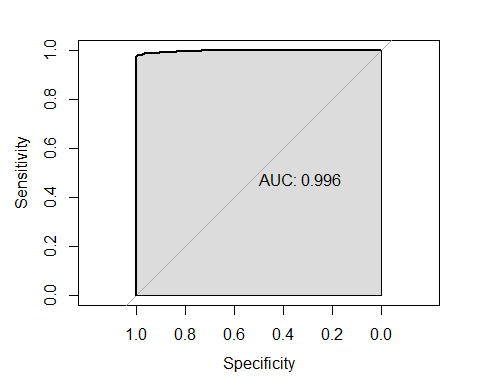
Random forests are a way of averaging multiple deep decision trees, trained on different parts of the same training set, with the goal of reducing the variance. Random Forests helps to reduce the overfitting of data which is pro in Decision Trees.

## 0 1 class.error  
## 0 9113 16 0.001752656  
## 1 86 2785 0.029954720

predicting for Test data using Random Forest model:

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 2297 20  
## 1 2 680  
##   
## Accuracy : 0.9927   
## 95% CI : (0.9889, 0.9954)  
## No Information Rate : 0.7666   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.9793   
## Mcnemar's Test P-Value : 0.0002896   
##   
## Sensitivity : 0.9991   
## Specificity : 0.9714   
## Pos Pred Value : 0.9914   
## Neg Pred Value : 0.9971   
## Prevalence : 0.7666   
## Detection Rate : 0.7659   
## Detection Prevalence : 0.7726   
## Balanced Accuracy : 0.9853   
##   
## 'Positive' Class : 0   
##

ROC Curve:



##   
## Call:  
## roc.default(response = test\_HR$left, predictor = fitted.values.rf1[, 2])  
##   
## Data: fitted.values.rf1[, 2] in 2299 controls (test\_HR$left 0) < 700 cases (test\_HR$left 1).  
## Area under the curve: 0.996

Support Vector Machines:

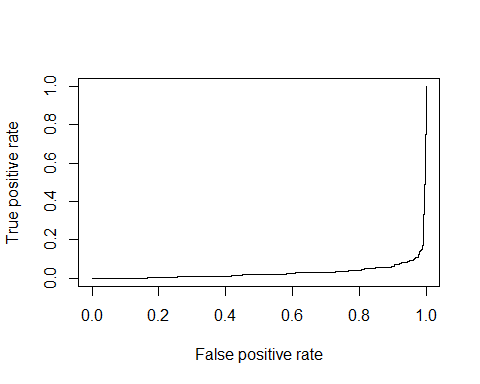
"Support Vector Machine" (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. Support Vector Machine is a frontier which best segregates the two classes (hyper-plane/ line).

##   
## Call:  
## svm(formula = left ~ ., data = train\_HR, type = "C-classification")  
##   
##   
## Parameters:  
## SVM-Type: C-classification   
## SVM-Kernel: radial   
## cost: 1   
## gamma: 0.05263158   
##   
## Number of Support Vectors: 1954  
##   
## ( 970 984 )  
##   
##   
## Number of Classes: 2   
##   
## Levels:   
## 0 1

Predicting for Test data using SVM:

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 2219 71  
## 1 80 629  
##   
## Accuracy : 0.9496   
## 95% CI : (0.9412, 0.9572)  
## No Information Rate : 0.7666   
## P-Value [Acc > NIR] : <2e-16   
##   
## Kappa : 0.8599   
## Mcnemar's Test P-Value : 0.515   
##   
## Sensitivity : 0.9652   
## Specificity : 0.8986   
## Pos Pred Value : 0.9690   
## Neg Pred Value : 0.8872   
## Prevalence : 0.7666   
## Detection Rate : 0.7399   
## Detection Prevalence : 0.7636   
## Balanced Accuracy : 0.9319   
##   
## 'Positive' Class : 0   
##

ROC Curve:



Gradient Boosting:

Gradient Boosting is basically about "boosting" many weak predictive models into a strong one, in the form of ensemble of weak models. It is a machine learning technique for regression and classification problems.