Project VI: GAM, MARS, PPR

Appiah Prince[*] University of Texas at El Paso (UTEP)

November 22, 2022

Contents

[*]pappiah@miners.utep.edu

1 Data Preparation

1.1 Bring in the data

```
hr <- read.csv("HR_comma_sep.csv")
head(hr)
```

```
##   satisfaction_level last_evaluation number_project average_montly_hours
## 1               0.38            0.53              2                  157
## 2               0.80            0.86              5                  262
## 3               0.11            0.88              7                  272
## 4               0.72            0.87              5                  223
## 5               0.37            0.52              2                  159
## 6               0.41            0.50              2                  153
##   time_spend_company Work_accident left promotion_last_5years sales salary
## 1                  3             0    1                     0 sales    low
## 2                  6             0    1                     0 sales medium
## 3                  4             0    1                     0 sales medium
## 4                  5             0    1                     0 sales    low
## 5                  3             0    1                     0 sales    low
## 6                  3             0    1                     0 sales    low
```

```
dim(hr)
```

```
## [1] 14999    10
```

```
str(hr)
```

```
## 'data.frame':    14999 obs. of  10 variables:
##  $ satisfaction_level   : num  0.38 0.8 0.11 0.72 0.37 0.41 0.1 0.92 0.89 0.42 ...
##  $ last_evaluation      : num  0.53 0.86 0.88 0.87 0.52 0.5 0.77 0.85 1 0.53 ...
##  $ number_project       : int  2 5 7 5 2 2 6 5 5 2 ...
##  $ average_montly_hours : int  157 262 272 223 159 153 247 259 224 142 ...
##  $ time_spend_company   : int  3 6 4 5 3 3 4 5 5 3 ...
```

```
##  $ Work_accident        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ left                 : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ promotion_last_5years: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ sales                : chr  "sales" "sales" "sales" "sales" ...
##  $ salary               : chr  "low" "medium" "medium" "low" ...
```

The data set contains 14,999 observations and 10 variables. The binary target left indicates whether a employee left the company. There are 5 continuous variables and 5 categorical/ordinal variables.

1.2 Change the categorical variable Salary to ordinal

```
hr$salary <- factor(hr$salary, levels = c("low", "medium", "high"),
                    ordered = TRUE)
str(hr$salary)
```

```
##  Ord.factor w/ 3 levels "low"<"medium"<..: 1 2 2 1 1 1 1 1 1 1 ...
```

1.3 Change the column name for variable sales to department

```
colnames(hr)[9] <- "department"
names(hr)
```

```
##  [1] "satisfaction_level"    "last_evaluation"       "number_project"
##  [4] "average_montly_hours"  "time_spend_company"    "Work_accident"
##  [7] "left"                  "promotion_last_5years" "department"
## [10] "salary"
```

1.4 Make the target variable left categorical using the factor function

```
hr$left <- factor(hr$left)
str(hr$left)
```

```
##  Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
```

1.5 Checking for missing values

```
library(questionr)
freq.na(hr)
```

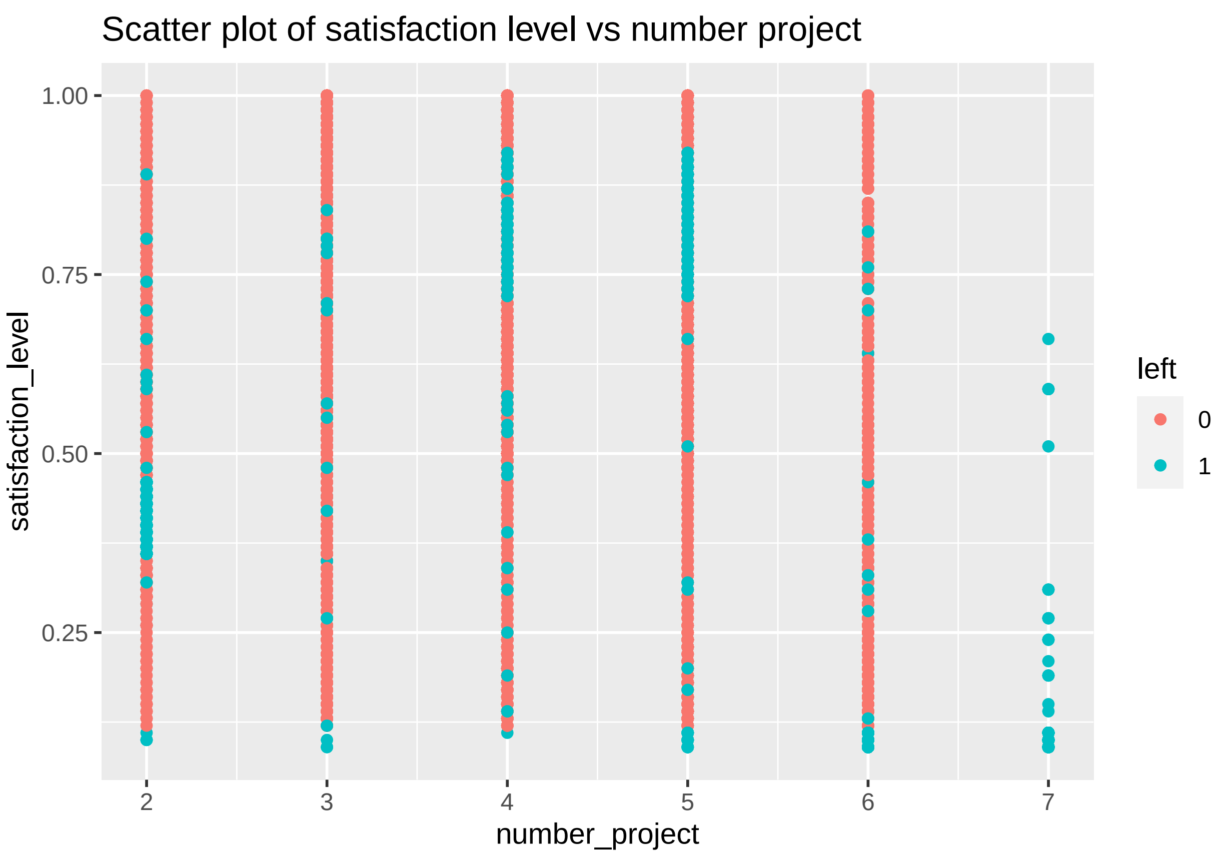
```
##                       missing %
## satisfaction_level          0 0
## last_evaluation             0 0
## number_project              0 0
## average_montly_hours        0 0
## time_spend_company          0 0
## Work_accident               0 0
## left                        0 0
## promotion_last_5years       0 0
## department                  0 0
## salary                      0 0
```

There are no missing values in the data.

2 Exploratory Data Analysis (EDA)

2.1 Scatter plot of satisfaction_level versus number_project

```
library(ggplot2)
ggplot(hr,  aes(x = number_project, y = satisfaction_level)) +
  geom_point(aes(colour = left)) +
  ggtitle("Scatter plot of satisfaction level vs number project")
```



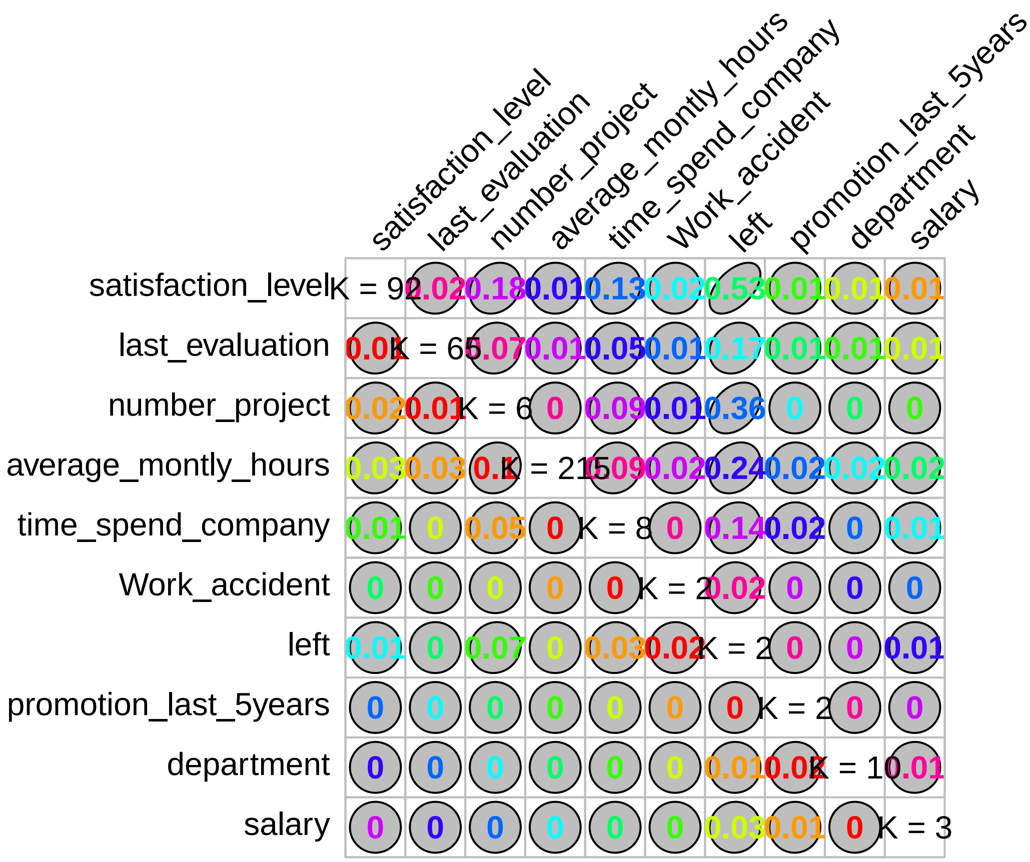
- From the scatterplot we see that employees who had 7 number of project were not satisfied so they left the company.
- Majority of employees with 2, 3 and 6 number of projects did not leave the company.
- There is almost equal number of proportion of employees who left and stayed with 4 and 5 number of projects.

2.2 Computing and Visualizing correltion matrix among the variables

```
# Correlation matrix
library(GoodmanKruskal)
data <- GKtauDataframe(hr)
data
```

```
##                       satisfaction_level last_evaluation number_project
## satisfaction_level                92.000           0.016          0.181
## last_evaluation                    0.012          65.000          0.066
## number_project                     0.019           0.006          6.000
## average_montly_hours               0.034           0.026          0.096
## time_spend_company                 0.008           0.004          0.049
## Work_accident                      0.000           0.000          0.002
## left                               0.008           0.003          0.075
## promotion_last_5years              0.000           0.000          0.000
## department                         0.001           0.001          0.001
## salary                             0.000           0.000          0.002
##                       average_montly_hours time_spend_company Work_accident
## satisfaction_level                   0.012              0.132         0.021
## last_evaluation                      0.008              0.053         0.011
## number_project                       0.003              0.094         0.009
## average_montly_hours               215.000              0.085         0.023
## time_spend_company                   0.002              8.000         0.005
## Work_accident                        0.000              0.001         2.000
## left                                 0.001              0.028         0.024
## promotion_last_5years                0.000              0.001         0.002
## department                           0.001              0.002         0.001
## salary                               0.000              0.001         0.000
##                        left promotion_last_5years department salary
## satisfaction_level    0.529                 0.011      0.009  0.013
## last_evaluation       0.169                 0.009      0.006  0.007
## number_project        0.358                 0.001      0.001  0.005
## average_montly_hours  0.242                 0.023      0.023  0.021
## time_spend_company    0.141                 0.021      0.005  0.007
## Work_accident         0.024                 0.002      0.000  0.000
## left                  2.000                 0.004      0.000  0.012
## promotion_last_5years 0.004                 2.000      0.002  0.005
## department            0.006                 0.023     10.000  0.011
## salary                0.025                 0.010      0.003  3.000
## attr(,"class")
## [1] "GKtauMatrix"
```

```r
# Visualization of the correlation matrix
plot(data, corColors = "magenta")
```



- Each of the continuous variables(satisfaction_level,last_evaluation,number_project, average_montly_hours,time_spend_company) has a larger association with the target variable left while the categorical variables have a very small association(approximately no association) with the target variable left.
- We also observed that there is approximately no association between the categorical variables and the continuous variables.

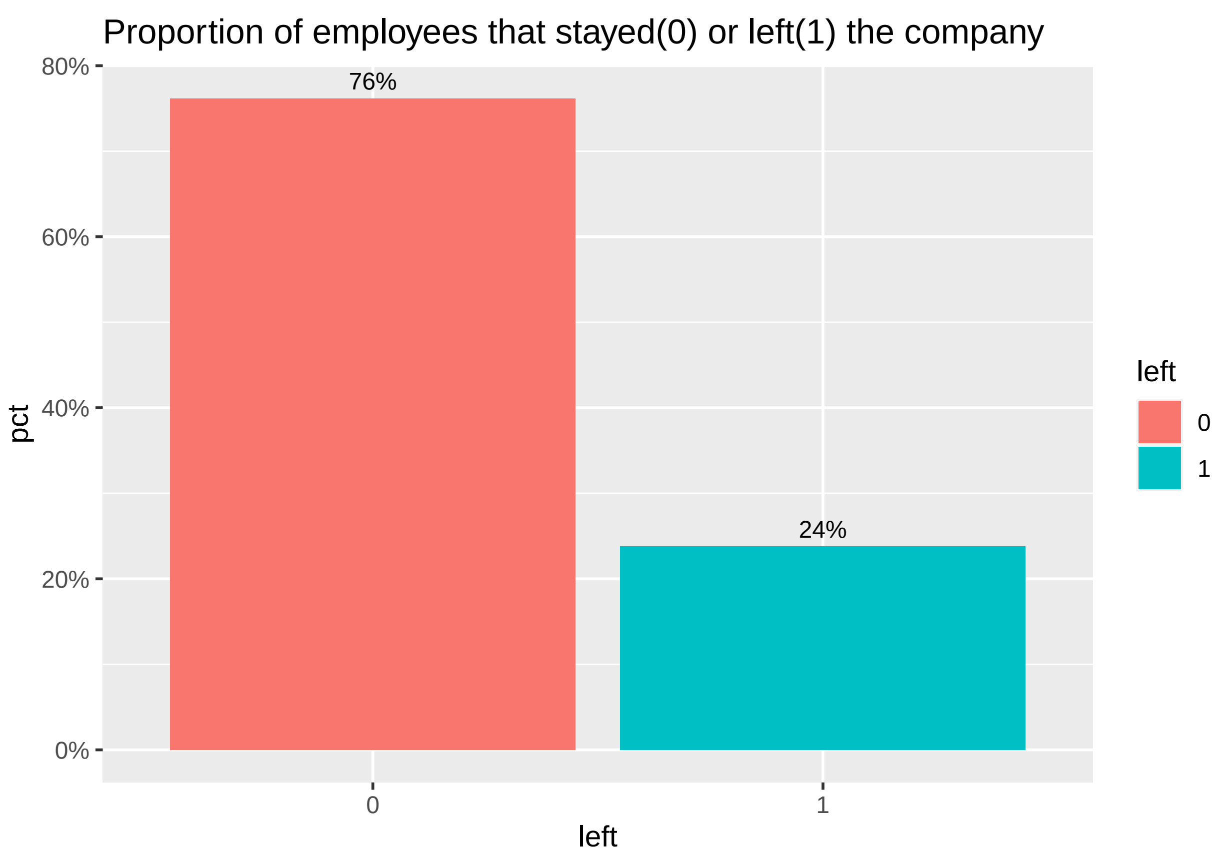
2.3 Bar Plot of the target variable left

```r
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
hr %>%
count(left) %>%
mutate(pct = prop.table(n)) %>%
ggplot(aes(x = left, y = pct, label = scales::percent(pct), fill=left)) +
geom_col(position = 'dodge') +
geom_text(position = position_dodge(width = .9),
vjust = -0.5,
size = 3) +
scale_y_continuous(labels = scales::percent) +
ggtitle("Proportion of employees that stayed(0) or left(1) the company")
```




```
theme(legend.position = "none")
```

```
## List of 1
##  $ legend.position: chr "none"
##  - attr(*, "class")= chr [1:2] "theme" "gg"
##  - attr(*, "complete")= logi FALSE
##  - attr(*, "validate")= logi TRUE
```

There is 76% of the employees that did not leave the company while 24% of the employees left the company.

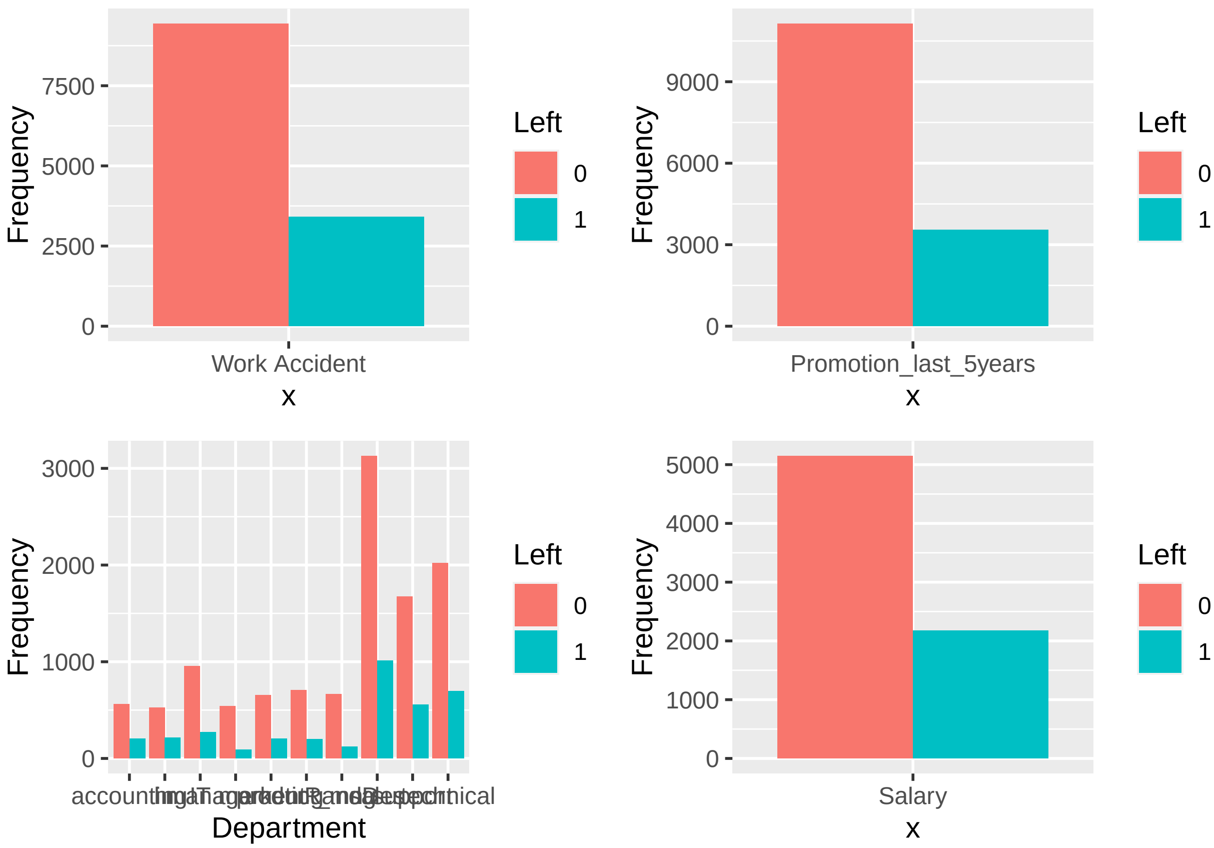
2.4 Proportion of left with respect to categorical variables

```
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine
```

```
tab <- table(hr$Work_accident, hr$left)
df <- data.frame(tab)
colnames(df) <- c("Work Accident", "Left", "Frequency")

pt1 <- ggplot(df, aes(x = 'Work Accident', y = Frequency, fill = Left)) +
  geom_bar(stat = "identity", position = "dodge")

tab1 <- table(hr$promotion_last_5years, hr$left)
df1 <- data.frame(tab1)
colnames(df1) <- c("promotion_last_5years", "Left", "Frequency")

pt2 <- ggplot(df1, aes(x = 'Promotion_last_5years', y = Frequency, fill = Left)) +
  geom_bar(stat = "identity", position = "dodge")

tab2 <- table(hr$department, hr$left)
df2 <- data.frame(tab2)
colnames(df2) <- c("Department", "Left", "Frequency")

pt3 <- ggplot(df2, aes(x = Department, y = Frequency, fill = Left)) +
  geom_bar(stat = "identity", position = "dodge")
```

```
tab3 <- table(hr$salary, hr$left)
df3 <- data.frame(tab3)
colnames(df3) <- c("Salary", "Left", "Frequency")

pt4 <- ggplot(df3, aes(x = 'Salary', y = Frequency, fill = Left)) +
  geom_bar(stat = "identity", position = "dodge")

grid.arrange(pt1, pt2, pt3, pt4, nrow = 2)
```

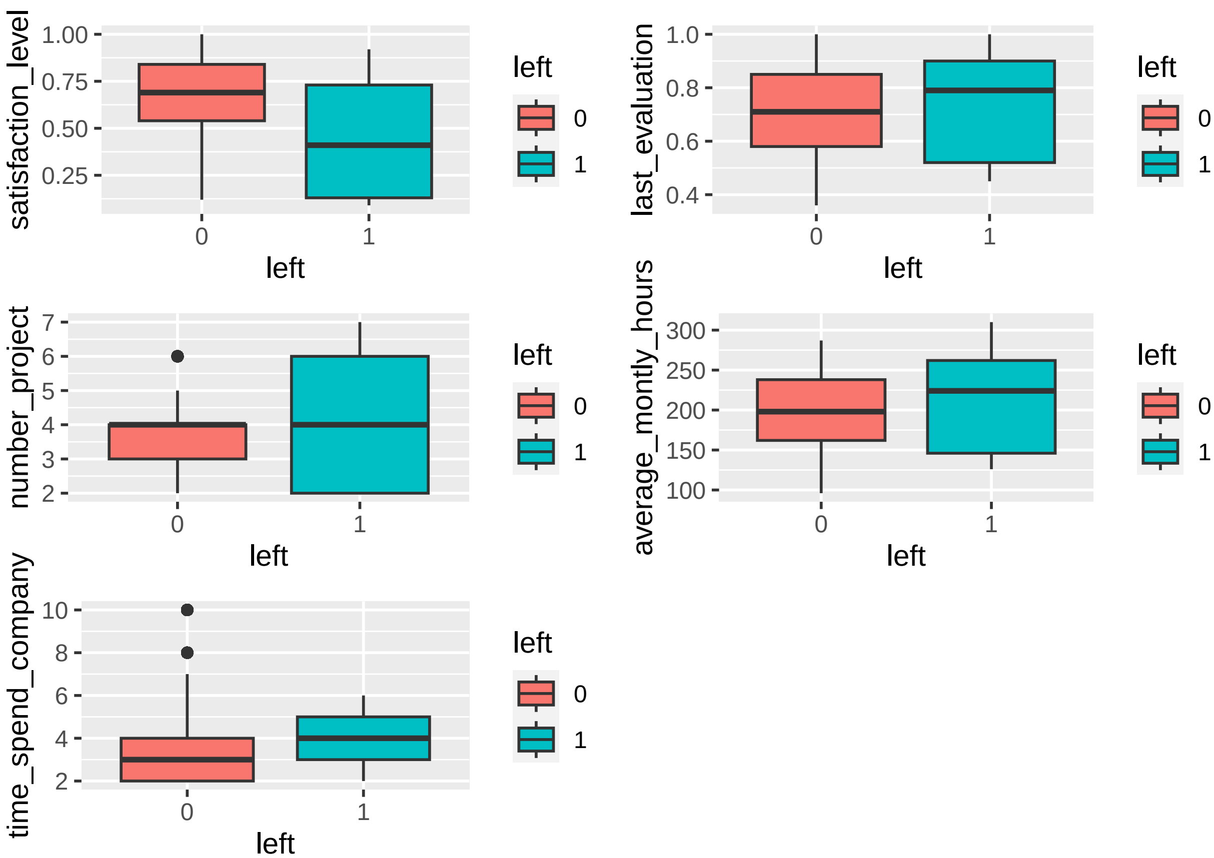


Considering the categorical variables, we see that the percentage of employees who did not leave the company is greater(more than 50%) of the percentage of employees who left the company.

2.5 Proportion of left with respect to continuous variables

```
ct1 <- ggplot(hr, aes(x =left, y = satisfaction_level, fill = left)) +
  geom_boxplot()

ct2 <- ggplot(hr, aes(x =left, y = last_evaluation, fill = left)) +
  geom_boxplot()
```

```
ct3 <- ggplot(hr, aes(x =left, y = number_project, fill = left)) +
  geom_boxplot()

ct4 <- ggplot(hr, aes(x =left, y = average_montly_hours, fill = left)) +
  geom_boxplot()

ct5 <- ggplot(hr, aes(x =left, y = time_spend_company, fill = left)) +
  geom_boxplot()

grid.arrange(ct1,ct2,ct3,ct4,ct5, nrow = 3)
```



- Considering the satisfaction_level, we see from the plot that the median of employees who did not leave the company is greater than those who left.
- Considering the last_evaluation, we see from the plot that the median of employees who did not leave the company is greater than those who left. The difference between these two medians is not much, which to some extent explains why some people left the company.
- Finally, considering number_project,average_montly_hours and time_spend_company, we see that the difference in median between those who left and those who stayed is not large. Thus, this also explain why some of the employees left the company.

3 Data Partitioning

```
set.seed(126)
sample_hr <- sample(nrow(hr), (2.0/3.0)*nrow(hr), replace = FALSE)
train_set <- hr[sample_hr, ] # training set
test_set <- hr[-sample_hr, ] #test set
dim(train_set)
```

```
## [1] 9999   10
```

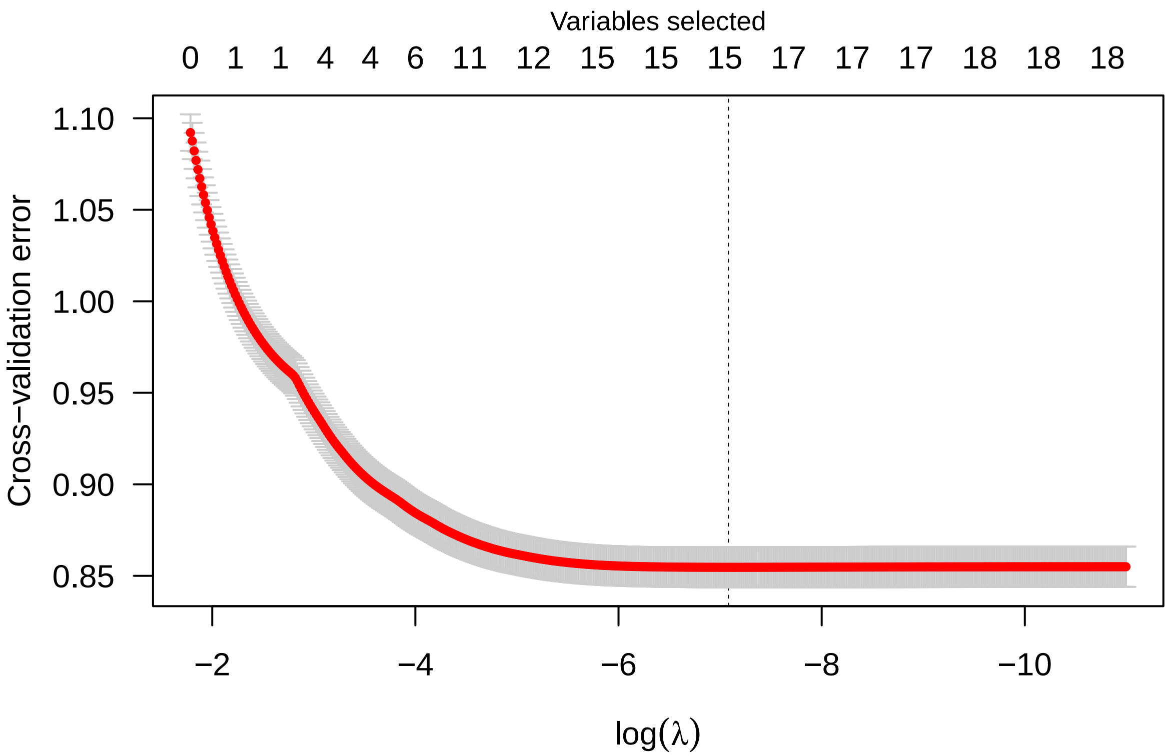
```
dim(test_set)
```

```
## [1] 5000   10
```

We have 9999 observations with 10 variables in the train set while we have 5000 observations with 10 variables in the test data.

4 Logistic Regression

```
set.seed(123)
library(ncvreg);
y <- train_set$left
formula0 <- left ~.
X <- model.matrix(as.formula(formula0), data=train_set)
cvfit.lasso <- cv.ncvreg(X=X,y=y, nfolds=5, family="binomial",
            penalty="lasso",lambda.min=.0001, nlambda=500,eps=.01,
            max.iter=1000)
plot(cvfit.lasso)
```



Selecting the best tuning parameter

```
cvfit.lasso$lambda.min
```

```
## [1] 0.000839499
```

Important Predictor Variables

```
result.lasso <- cvfit.lasso$fit
beta.hat <- as.vector(result.lasso$beta[-1, cvfit.lasso$min])
cutoff <- 0
terms <- colnames(X)[abs(beta.hat) > cutoff]
terms
```

```
##  [1] "satisfaction_level"    "last_evaluation"       "number_project"
##  [4] "average_montly_hours"  "time_spend_company"    "Work_accident"
##  [7] "promotion_last_5years" "departmenthr"          "departmentIT"
## [10] "departmentmanagement"  "departmentmarketing"   "departmentproduct_mng"
## [13] "departmentRandD"       "departmenttechnical"   "salary.L"
## [16] "salary.Q"
```

We see that all the variables are important.

Final Best Model Fit

```
formula01 <- left ~ satisfaction_level + last_evaluation + number_project +

formula.lasso <- as.formula(formula01)
fit.lasso <- glm(formula.lasso, data = train_set, family="binomial")
smy <- summary(fit.lasso)
smy$coefficients
```

```
##                           Estimate    Std. Error      z value      Pr(>|z|)
## (Intercept)            -0.211390049 0.1877744782  -1.12576560  2.602647e-01
## satisfaction_level     -4.183581780 0.1201568776 -34.81766392 1.314491e-265
## last_evaluation         0.721463217 0.1839471605   3.92212207  8.777250e-05
## number_project         -0.296531116 0.0262458854 -11.29819440  1.339436e-29
## average_montly_hours    0.003873890 0.0006398701   6.05418284  1.411322e-09
## time_spend_company      0.264703794 0.0191828952  13.79894907  2.585950e-43
## Work_accident          -1.588937203 0.1112945812 -14.27686044  3.050372e-46
## promotion_last_5years -2.094044328 0.3831425630  -5.46544428  4.617488e-08
## departmenthr            0.280181821 0.1607747185   1.74269825  8.138634e-02
## departmentIT           -0.176459867 0.1486681018  -1.18693832  2.352519e-01
## departmentmanagement   -0.439534110 0.1925193229  -2.28306491  2.242655e-02
## departmentmarketing     0.004128582 0.1606506928   0.02569912  9.794973e-01
## departmentproduct_mng -0.183297614 0.1617154445  -1.13345769  2.570221e-01
## departmentRandD        -0.582118681 0.1770027690  -3.28875466  1.006317e-03
## departmentsales        -0.029952971 0.1250395167  -0.23954804  8.106806e-01
## departmentsupport      -0.015900192 0.1339792635  -0.11867651  9.055316e-01
## departmenttechnical     0.029375443 0.1303583538   0.22534377  8.217119e-01
## salary.L               -1.271996367 0.1066888182 -11.92248999  9.036673e-33
## salary.Q               -0.294637823 0.0705255183  -4.17774772  2.944099e-05
```

```
smy$aic
```

```
## [1] 8530.854
```

- The AIC = 8530.854 is larger which indicates poor performance of our model.
- At significance level of 0.05 we see that all the p values are less than 0.05 indicating that all the predictors are statistically significant.

Obtaining the associated odds ratio and the 95% confidence intervals for the odds ratio

```
exp(cbind('Odd ratio' = coef(fit.lasso), confint(fit.lasso)))
```

```
## Waiting for profiling to be done...
```

```
##                          Odd ratio      2.5 %     97.5 %
## (Intercept)             0.80945828 0.55954819 1.16837520
## satisfaction_level      0.01524381 0.01202485 0.01926008
## last_evaluation         2.05744149 1.43526855 2.95207817
## number_project          0.74339250 0.70596311 0.78247179
## average_montly_hours    1.00388140 1.00262533 1.00514366
## time_spend_company      1.30304495 1.25496872 1.35300361
## Work_accident           0.20414246 0.16335075 0.25277425
## promotion_last_5years 0.12318791 0.05370628 0.24562516
## departmenthr            1.32337041 0.96580717 1.81430909
## departmentIT            0.83823242 0.62664360 1.12259232
## departmentmanagement    0.64433654 0.44021380 0.93692871
## departmentmarketing     1.00413712 0.73279579 1.37593971
## departmentproduct_mng 0.83252035 0.60604004 1.14274182
## departmentRandD         0.55871338 0.39390900 0.78877410
## departmentsales         0.97049117 0.76093710 1.24256838
## departmentsupport       0.98422555 0.75797965 1.28186703
## departmenttechnical     1.02981116 0.79889248 1.33201861
## salary.L                0.28027154 0.22600571 0.34354934
## salary.Q                0.74480129 0.64698792 0.85327292
```

- The estimated odds for satisfaction_level is exp(-4.183581780) = 0.01524381. For each increase in 1 unit of satisfaction_level,the estimated odds of an employee to leave the company decreases by a factor of 0.01524381 holding the other predictors constants.
- The estimated odds for last_evaluation is exp(0.721463217) = 2.057441. For each increase in 1 unit of last_evaluation,the estimated odds of an employee to leave the company decreases by a factor of 2.057441 holding the other predictors constants.
- The estimated odds for time_spend_company is exp(0.264703794) = 1.303045. For each increase in 1 unit of time_spend_company,the estimated odds of an employee to leave the company decreases by a factor of 1.303045. holding the other predictors constants.

Applying the final logistic model to the test data

```
yobs <- as.numeric(as.character(test_set$left))
phat <- predict(fit.lasso, newdata=test_set, type="response")
cutoff <- 0.5
yhat <- (phat <= cutoff) + 0
table(yobs, yhat)
```

```
##     yhat
## yobs    0    1
##    0  286 3504
##    1  418  792
```

ROC CURVE AND AUC

```
suppressPackageStartupMessages(library(verification))
a.ROC <- roc.area(obs=yobs, pred=phat)$A
print(a.ROC)
```

```
## [1] 0.8144848
```

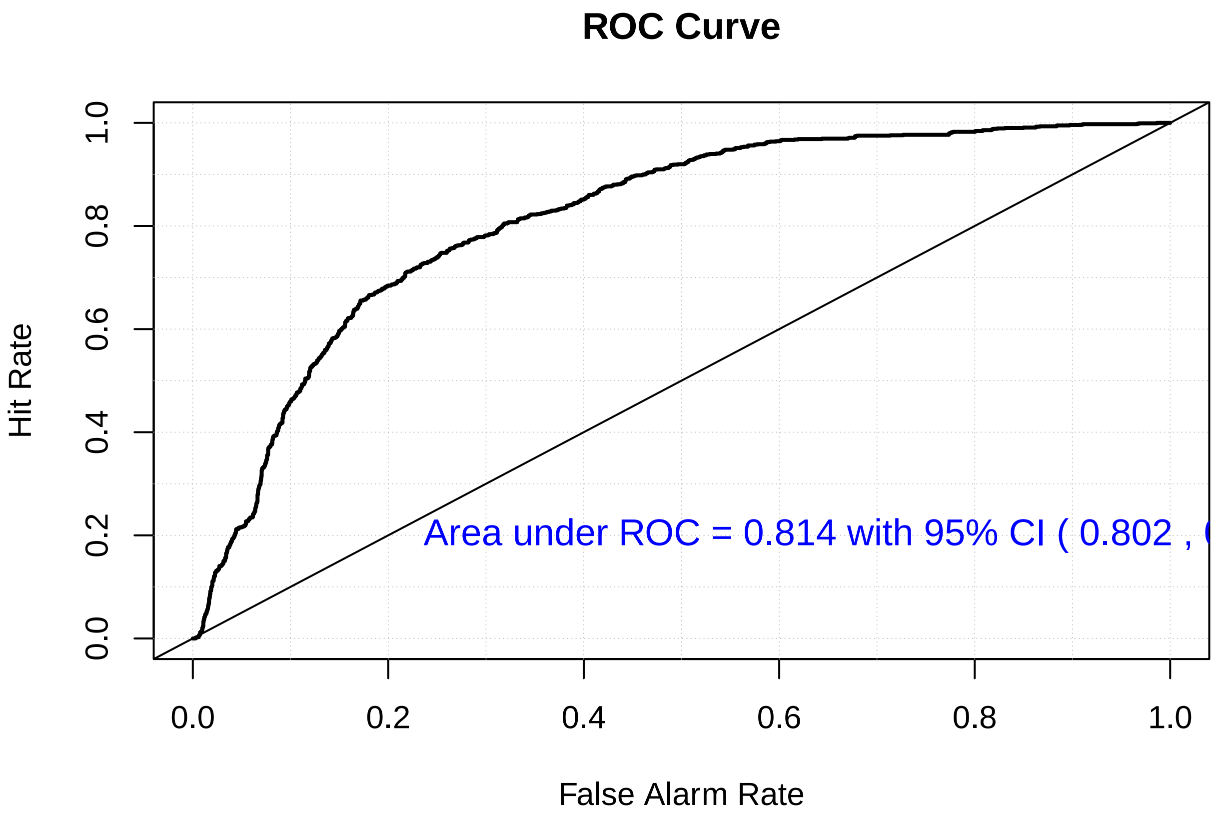
```
suppressPackageStartupMessages(library(cvAUC))
AUC <- ci.cvAUC(predictions=phat, labels=yobs, folds=1:NROW(test_set), confidence=0.95)
auc.ci <- round(AUC$ci, digits=3)

suppressPackageStartupMessages(library(verification))
mod.glm <- verify(obs=yobs, pred=phat)
```

```
## If baseline is not included, baseline values  will be calculated from the  sample obs
```

```
roc.plot(mod.glm, plot.thres = NULL)
```

```
## Warning in roc.plot.default(c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, : Large
## amount of unique predictions used as thresholds. Consider specifying thresholds.
```

```
text(x=0.7, y=0.2, paste("Area under ROC =", round(AUC$cvAUC, digits=3),
    "with 95% CI (", auc.ci[1], ",", auc.ci[2], ").",
    sep=" "), col="blue", cex=1.2)
```




```
log_reg_lasso <- round(AUC$cvAUC, digits=4)
```

The area under the curve according to regularized logistic regression using lasso as penalty function is 0.814.

5 Random Forest

```
library(randomForest)
```

```
## randomForest 4.6-14
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:gridExtra':
##
##     combine
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
fit.rf <- randomForest(left ~., data=train_set,importance=TRUE, proximity=TRUE, ntree=50
fit.rf;
```

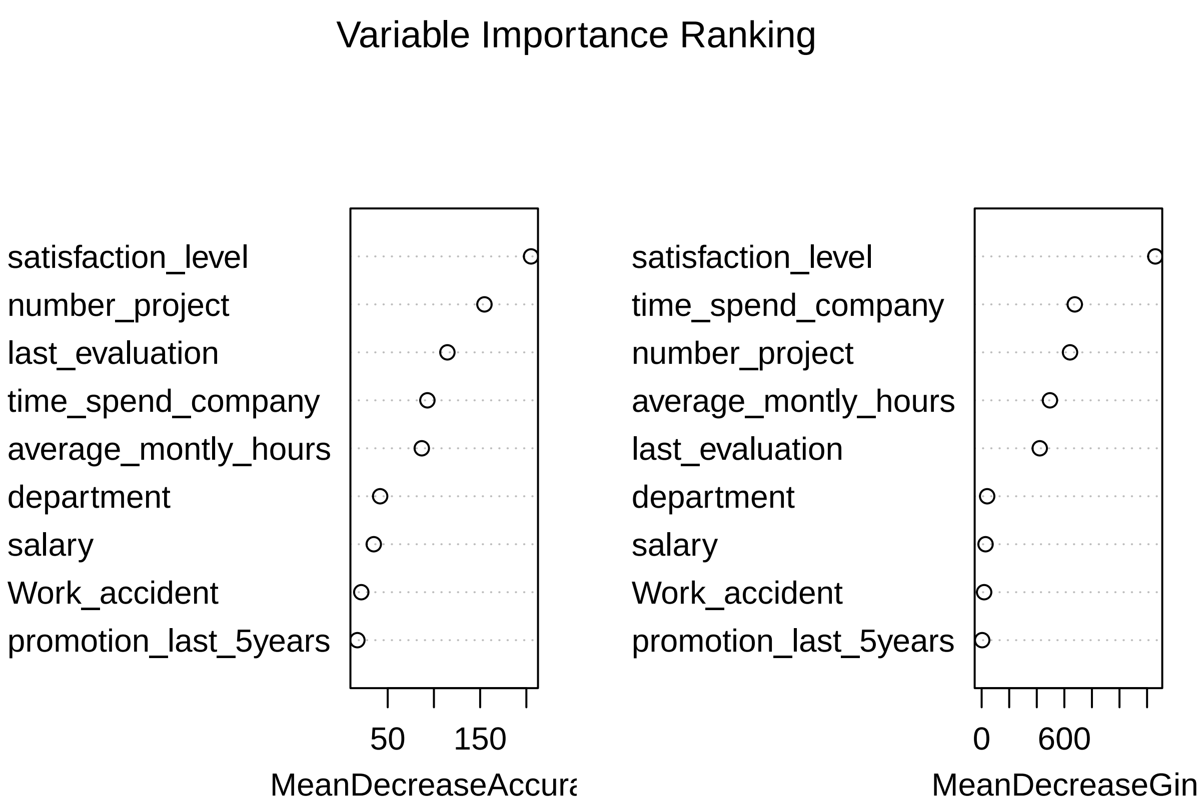
```
##
## Call:
##  randomForest(formula = left ~ ., data = train_set, importance = TRUE,      proximity
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 3
##
##         OOB estimate of  error rate: 0.87%
## Confusion matrix:
##      0    1 class.error
## 0 7629    9 0.001178319
## 1   78 2283 0.033036849
```

```r
rf_yhat <- predict(fit.rf, newdata=test_set, type="prob")[, 2]
```

```r
# VARIABLE IMPORTANCE RANKING
round(importance(fit.rf), 2)
```

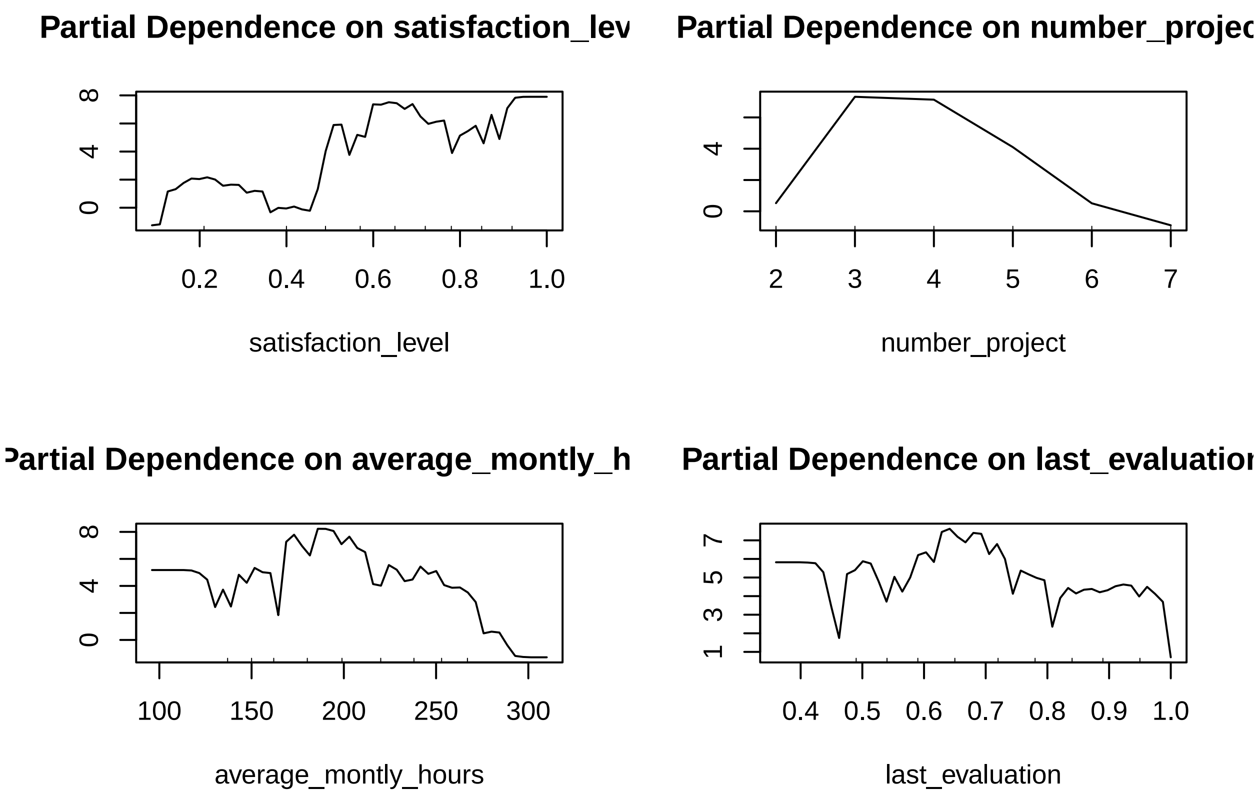
```
##                            0      1 MeanDecreaseAccuracy MeanDecreaseGini
## satisfaction_level     58.67 237.63               205.05          1259.76
## last_evaluation        23.17 113.04               114.55           421.47
## number_project         48.98 151.91               154.66           641.12
## average_montly_hours   64.32  77.72                86.84           495.49
## time_spend_company     58.31  86.03                92.93           675.67
## Work_accident           7.87  22.55                21.42            18.02
## promotion_last_5years   7.52  16.29                17.18             3.94
## department             12.75  55.20                41.82            40.14
## salary                 16.00  37.13                34.90            28.34
```

```r
varImpPlot(fit.rf, main="Variable Importance Ranking")
```



Using mean decrease accuracy, we see that the first three important variables are satisfaction_level,last_evaluation and number_project respectively.

```r
# PARTIAL DEPENDENCE PLOT
par(mfrow=c(2,2))
partialPlot(fit.rf, pred.data=train_set, x.var=satisfaction_level, rug=TRUE)
partialPlot(fit.rf, pred.data=train_set, x.var=number_project, rug=TRUE)
partialPlot(fit.rf, pred.data=train_set, x.var=average_montly_hours, rug=TRUE)
partialPlot(fit.rf, pred.data=train_set, x.var=last_evaluation, rug=TRUE)
```

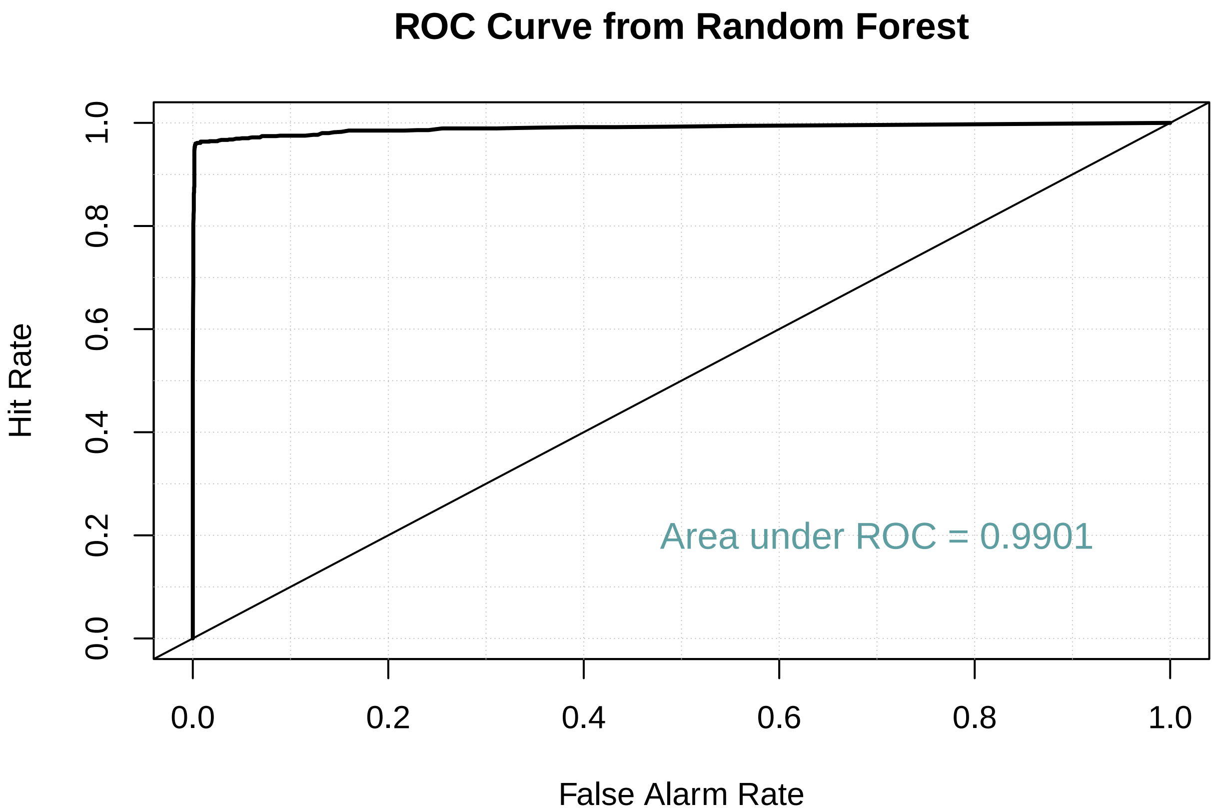


Clearly, we see that the plots show non-linearity. The strong non-linearity shown on these plots show the inadequacy of linear logistic regression model.

```r
AUC.RF <- roc.area(obs=yobs, pred=rf_yhat)$A
mod.rf <- verify(obs=yobs, pred=rf_yhat)
```

```
## If baseline is not included, baseline values  will be calculated from the  sample obs
```

```r
roc.plot(mod.rf, plot.thres = NULL, col="red", main="ROC Curve from Random Forest")
text(x=0.7, y=0.2, paste("Area under ROC =", round(AUC.RF, digits=4),
    sep=" "), col="cadetblue", cex=1.2)
```



The area under the curve according to random forest model is 0.9905.

6 Generalized Additive Model(GAM)

```
library(gam)
```

```
## Loading required package: splines

## Loading required package: foreach

## Loaded gam 1.20.2
```

```
fit.gam <- gam( left ~ s(satisfaction_level,6) + s(number_project,6) + s(time_spend_comp
+ salary , family = binomial,
    data=train_set, trace=TRUE,
    control = gam.control(epsilon=1e-04, bf.epsilon = 1e-04, maxit=50, bf.maxit = 50))

smy1 <- summary(fit.gam)
smy1$parametric.anova
```

```
## Anova for Parametric Effects
##                               Df  Sum Sq Mean Sq  F value    Pr(>F)    
## s(satisfaction_level, 6)       1    58.1   58.11  45.7862 1.393e-11 ***
## s(number_project, 6)           1     0.5    0.54   0.4225    0.5157    
## s(time_spend_company, 6)       1   483.1  483.07 380.6022 < 2.2e-16 ***
## s(last_evaluation, 6)          1    51.8   51.82  40.8282 1.737e-10 ***
## s(average_montly_hours, 6)     1    24.5   24.47  19.2773 1.142e-05 ***
## department                     9    18.5    2.05   1.6162    0.1044    
## Work_accident                  1    70.0   70.04  55.1817 1.189e-13 ***
## promotion_last_5years          1     4.1    4.14   3.2606    0.0710 .  
## salary                         2    66.5   33.23  26.1839 4.553e-12 ***
## Residuals                   9956 12636.5    1.27                       
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
smy1$anova
```

```
## Anova for Nonparametric Effects
##                             Npar Df Npar Chisq    P(Chi)    
## (Intercept)                                                 
## s(satisfaction_level, 6)          5     873.95 < 2.2e-16 ***
## s(number_project, 6)              4     449.44 < 2.2e-16 ***
## s(time_spend_company, 6)          5     175.24 < 2.2e-16 ***
```

```
## s(last_evaluation, 6)           5     277.10 < 2.2e-16 ***
## s(average_montly_hours, 6)       5     331.51 < 2.2e-16 ***
## department
## Work_accident
## promotion_last_5years
## salary
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# Prediction on the test set
yhat.gam <- predict(fit.gam, newdata=test_set, type="response", se.fit=FALSE)
```

- We see that the predictors satisfaction_level,number_project, time_spend_company,last_evaluation and average_montly_hours are statistically significant while department, work_accident,promotion_ and salary are not statistically significant under Anova for Nonparametric effects.
- Under Anova for Parametric Effects, number_project,department and promotion_last_5years are not statistically significant.

Variable/Model Selection

```
fit.step <- step.Gam(fit.gam, scope=list("satisfaction_level"=~1 +satisfaction_level + l
                "last_evaluation"=~1+ last_evaluation + lo(last_evaluation)+ s(last_eval
                "number_project"=~1 + number_project + s(number_project, 2) + s(number_p
                    "average_montly_hours"=~1 + average_montly_hours + s(average_montly_
    "time_spend_company"=~1 + time_spend_company + s(time_spend_company, 2) + s(time_spe
            scale =2, steps=1000, parallel=TRUE, direction="both")
```

```
## Start:  left ~ s(satisfaction_level, 6) + s(number_project, 6) + s(time_spend_company
```

```
## Warning: executing %dopar% sequentially: no parallel backend registered
```

```
summary(fit.step)
```

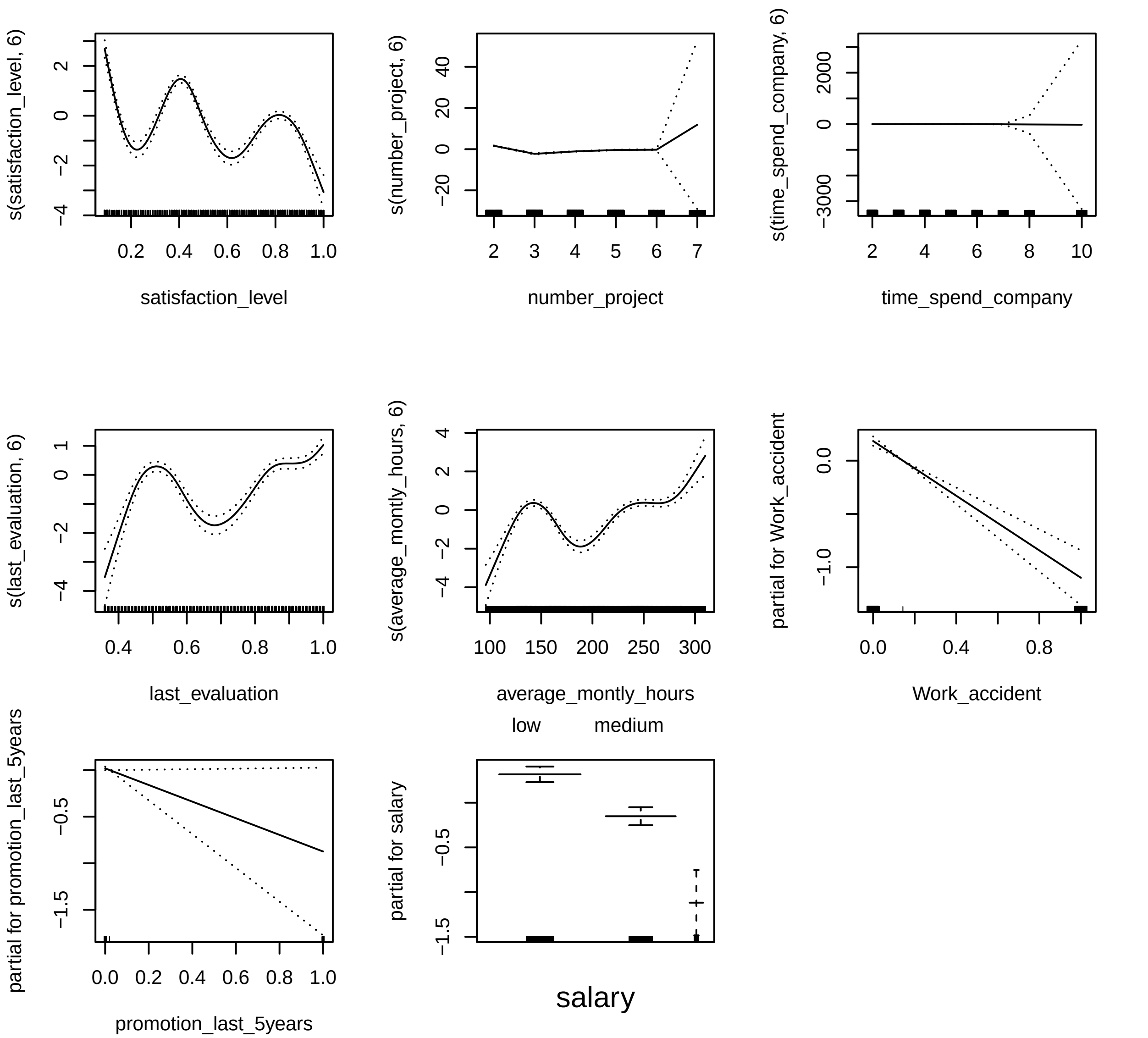
```
##
## Call: gam(formula = left ~ s(satisfaction_level, 6) + s(number_project,
##     6) + s(time_spend_company, 6) + s(last_evaluation, 6) + s(average_montly_hours,
##     6) + department + Work_accident + promotion_last_5years +
##     salary, family = binomial, data = train_set, control = gam.control(epsilon = 1e-0
##     bf.epsilon = 1e-04, maxit = 50, bf.maxit = 50), trace = TRUE)
## Deviance Residuals:
##        Min         1Q     Median         3Q        Max
## -3.114e+00 -2.281e-01 -8.284e-02 -4.710e-06  3.663e+00
##
## (Dispersion Parameter for binomial family taken to be 1)
##
##     Null Deviance: 10930.31 on 9998 degrees of freedom
## Residual Deviance: 3053.991 on 9955.999 degrees of freedom
## AIC: 3139.992
##
## Number of Local Scoring Iterations: NA
##
## Anova for Parametric Effects
##                                Df  Sum Sq Mean Sq  F value    Pr(>F)
## s(satisfaction_level, 6)        1    58.1   58.11  45.7862 1.393e-11 ***
## s(number_project, 6)            1     0.5    0.54   0.4225    0.5157
## s(time_spend_company, 6)        1   483.1  483.07 380.6022 < 2.2e-16 ***
## s(last_evaluation, 6)           1    51.8   51.82  40.8282 1.737e-10 ***
## s(average_montly_hours, 6)      1    24.5   24.47  19.2773 1.142e-05 ***
## department                      9    18.5    2.05   1.6162    0.1044
## Work_accident                   1    70.0   70.04  55.1817 1.189e-13 ***
## promotion_last_5years           1     4.1    4.14   3.2606    0.0710 .
```

```
## salary                         2    66.5   33.23   26.1839 4.553e-12 ***
## Residuals                   9956 12636.5    1.27
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Anova for Nonparametric Effects
##                             Npar Df Npar Chisq    P(Chi)
## (Intercept)
## s(satisfaction_level, 6)          5     873.95 < 2.2e-16 ***
## s(number_project, 6)              4     449.44 < 2.2e-16 ***
## s(time_spend_company, 6)          5     175.24 < 2.2e-16 ***
## s(last_evaluation, 6)             5     277.10 < 2.2e-16 ***
## s(average_montly_hours, 6)        5     331.51 < 2.2e-16 ***
## department
## Work_accident
## promotion_last_5years
## salary
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Plotting the (nonlinear) functional forms for continuous predictors

```
par(mfrow=c(2,3))
plot(fit.step, se =TRUE)
```

```
## Warning in gplot.default(x = c("product_mng", "technical", "sales",
## "technical", : The "x" component of "partial for department" has class
## "character"; no gplot() methods available
```



- Each smoothing parameter was determined adaptively in the backfitting algorithm. The smoothing splines were used and optimization of the tuning parameter is automatically done through minimum GCV.
- The Stepwise selection with AIC was used to do the variable selection.
- The strong non-linearity shown on these plots show the inadequacy of (linear) logistic regression model.

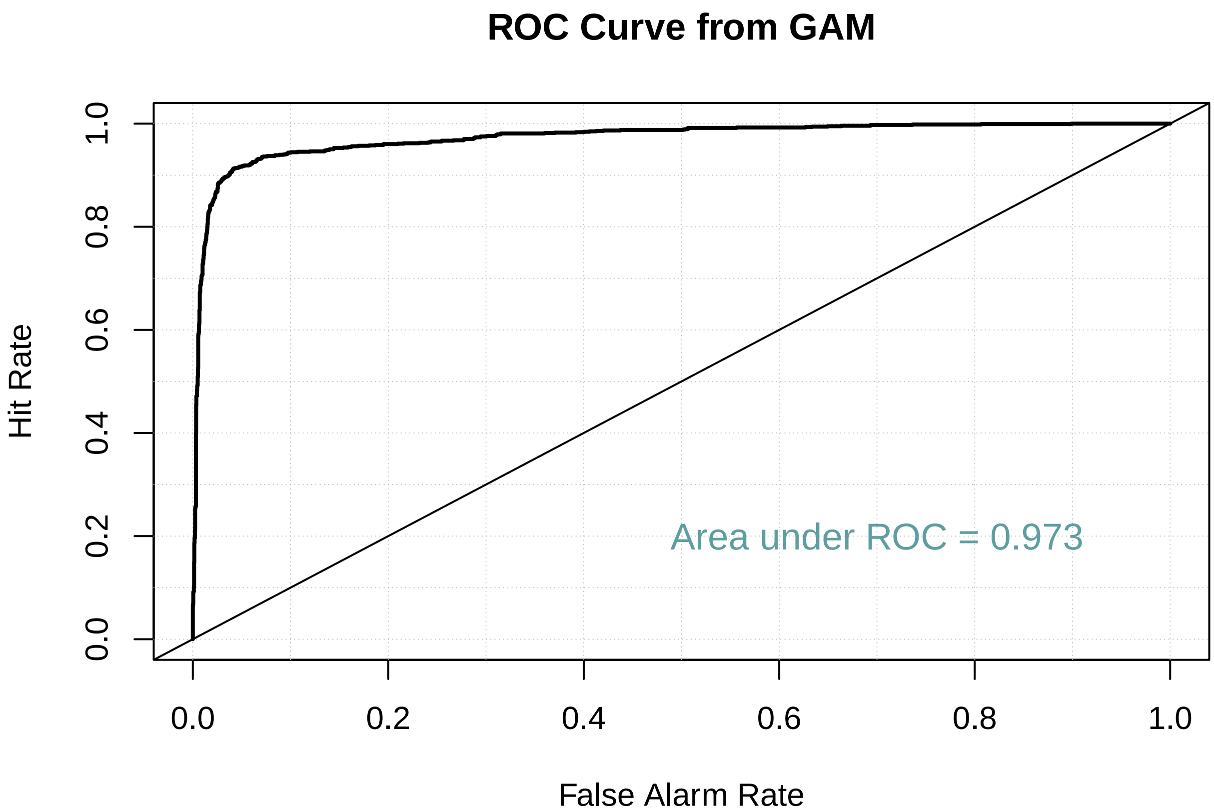
```
suppressPackageStartupMessages(library(verification))
yhat.gam <- predict(fit.step, newdata=test_set, type="response", se.fit=FALSE)
AUC.GAM <- roc.area(obs=yobs, pred=yhat.gam)$A
mod.gam <- verify(obs=yobs, pred=yhat.gam)
```

```
## If baseline is not included, baseline values  will be calculated from the  sample obs
```

```
roc.plot(mod.gam, plot.thres = NULL, col="red", main="ROC Curve from GAM")
```

```
## Warning in roc.plot.default(c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, : Large
## amount of unique predictions used as thresholds. Consider specifying thresholds.
```

```
text(x=0.7, y=0.2, paste("Area under ROC =", round(AUC.GAM, digits=4),
    sep=" "), col="cadetblue", cex=1.2)
```



The area under the curve according to GAM model is 0.973.

7 Multivariate Adaptive Regression Splines

```
library("earth")
```

```
## Loading required package: Formula

## Loading required package: plotmo

## Loading required package: plotrix

##
## Attaching package: 'plotrix'

## The following object is masked from 'package:fields':
##
##     color.scale

## Loading required package: TeachingDemos
```

```
library(ggplot2)   # plotting
library(caret)     # automating the tuning process
```

```
## Loading required package: lattice

##
## Attaching package: 'lattice'

## The following object is masked from 'package:boot':
##
##     melanoma

## Registered S3 method overwritten by 'pROC':
##   method    from
##   lines.roc verification
```

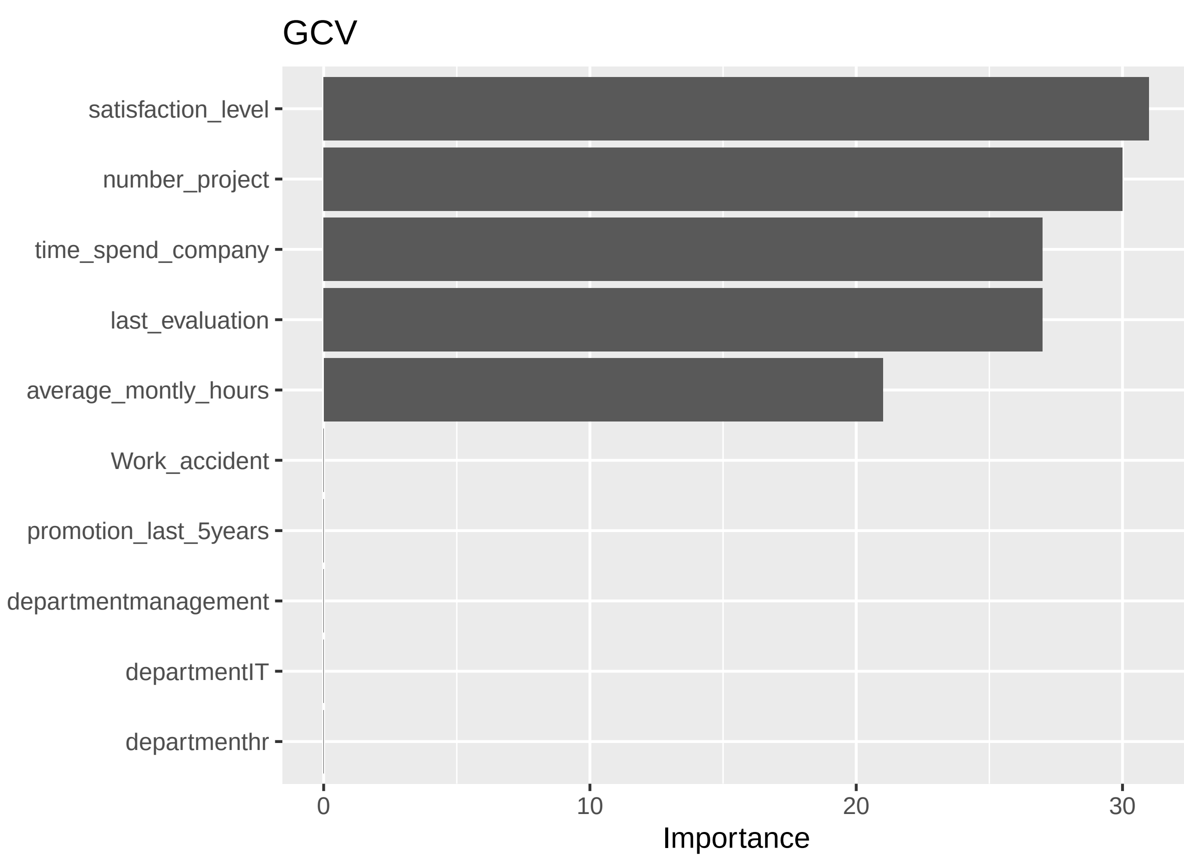
```
library(vip)       # variable importance
```

```
##
## Attaching package: 'vip'

## The following object is masked from 'package:utils':
##
##     vi
```

```r
library(pdp)       # variable relationships
fit.mars <- earth(left ~ .,  data = train_set, degree=3,
    glm=list(family=binomial(link = "logit")))
summary(fit.mars) %>% .$coefficients %>% head(10)
```

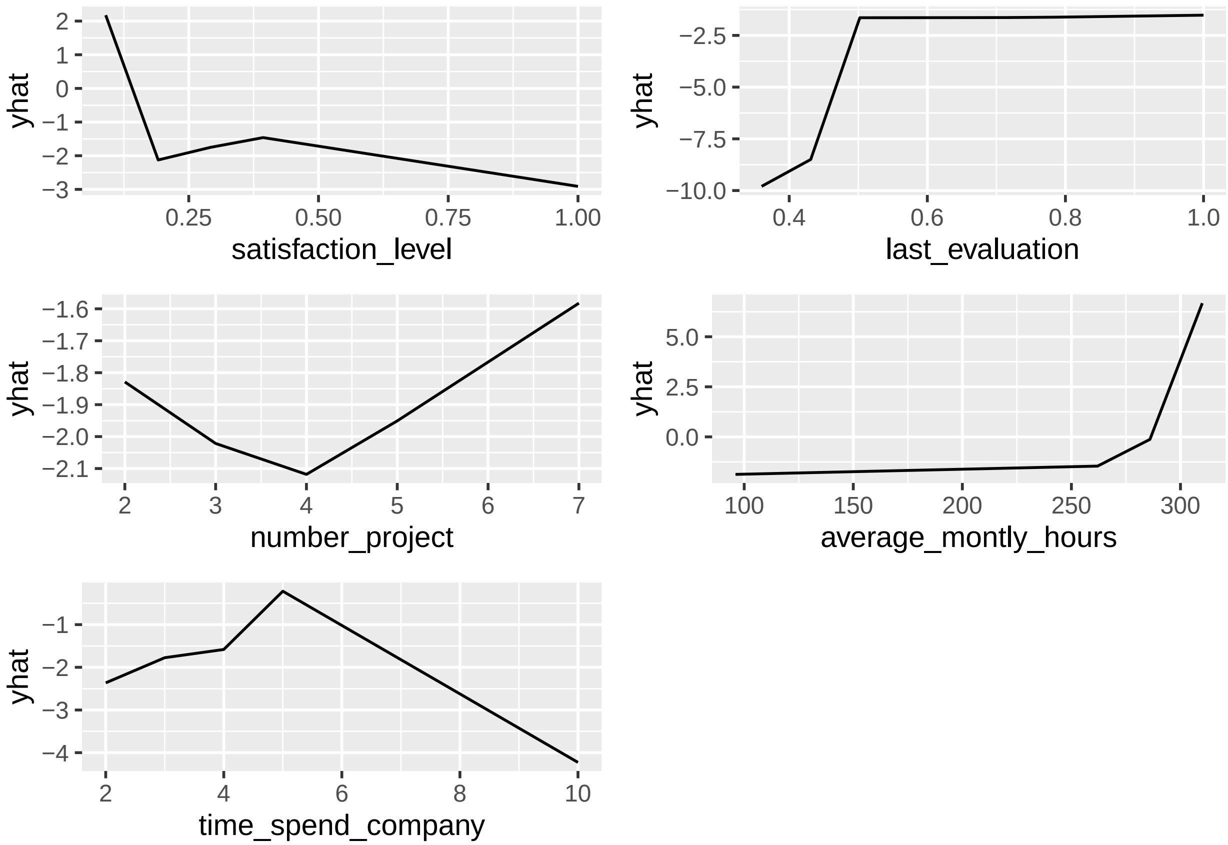
```
##                                                                               1
## (Intercept)                                                         -0.01900902
## h(number_project-3)                                                  0.03902892
## h(3-number_project)                                                  1.11377850
## h(number_project-3)*h(time_spend_company-5)                         -0.01875275
## h(number_project-3)*h(5-time_spend_company)                          0.02858534
## h(satisfaction_level-0.38)*h(3-number_project)                      -2.54617545
## h(0.38-satisfaction_level)*h(3-number_project)                      -4.64743314
## h(satisfaction_level-0.24)*h(number_project-3)                      -0.07765670
## h(0.24-satisfaction_level)*h(number_project-3)                       0.26419562
## h(satisfaction_level-0.24)*h(last_evaluation-0.75)*h(number_project-3)  0.83958808
```

```r
# VARIABLE IMPORTANCE PLOT
vip(fit.mars, num_features = 10) + ggtitle("GCV")
```



The first three variable of importance are Satisfaction_level, number_project and time_spend_company respectively.

```
# PARTIAL DEPENDENCE PLOT
p1 <- partial(fit.mars, pred.var = "satisfaction_level", grid.resolution = 10)%>%autoplo
p2 <- partial(fit.mars, pred.var = "last_evaluation", grid.resolution = 10)%>%autoplot()
p3 <- partial(fit.mars, pred.var = "number_project", grid.resolution = 10)%>%autoplot()
p4 <- partial(fit.mars, pred.var = "average_montly_hours", grid.resolution = 10)%>%autop
p5 <- partial(fit.mars, pred.var = "time_spend_company", grid.resolution = 10)%>%autoplo
grid.arrange(p1, p2, p3, p4, p5, nrow = 3)
```




```
# PREDICTION
library(cvAUC)
yhat.mars <- predict(fit.mars, newdata=test_set, type="response")
AUC.MARS <- ci.cvAUC(predictions=yhat.mars, labels=yobs, folds=1:length(yhat.mars), conf
```

```
## Warning in if (class(predictions) == "list" | class(labels) == "list") {: the
## condition has length > 1 and only the first element will be used
```

```
## $cvAUC
## [1] 0.9750833
##
## $se
```

```
## [1] 0.003047395
##
## $ci
## [1] 0.9691105 0.9810561
##
## $confidence
## [1] 0.95
```

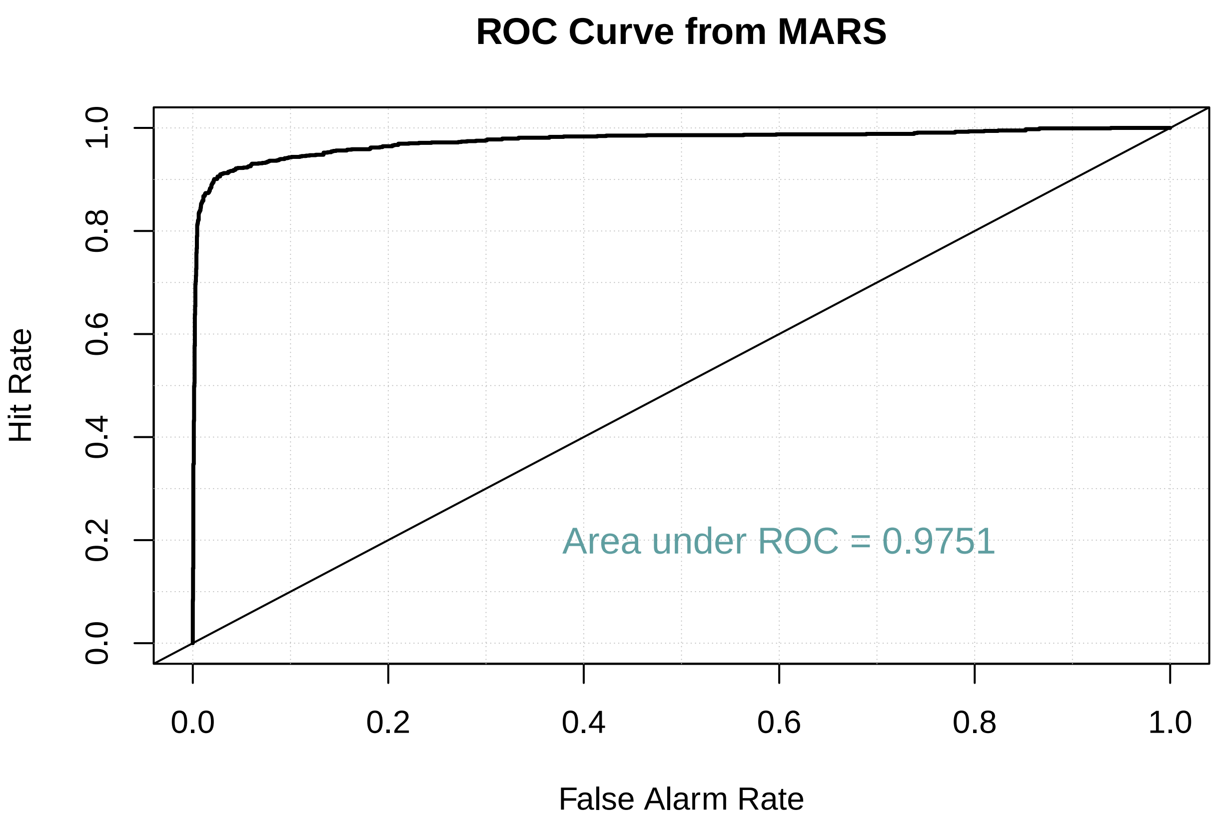
```
auc.ci <- round(AUC.MARS$ci, digits=4)
library(verification)
mod.mars <- verify(obs=yobs, pred=yhat.mars)
```

```
## If baseline is not included, baseline values  will be calculated from the  sample obs
```

```
roc.plot(mod.mars, plot.thres = NULL, main="ROC Curve from MARS")
```

```
## Warning in roc.plot.default(c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, : Large
## amount of unique predictions used as thresholds. Consider specifying thresholds.
```

```
text(x=0.6, y=0.2, paste("Area under ROC =", round(AUC.MARS$cvAUC, digits=4),
    sep=" "), col="cadetblue", cex=1.2)
```



The area under the curve according to MARS is 0.9751

8 Project Pursuit Regression

```
train_set$left <- as.numeric(as.character(train_set$left))
fit.ppr <- ppr(left ~ ., sm.method = "supsmu",
    data = train_set, nterms = 2, max.terms = 10, bass=3)
summary(fit.ppr)
```

```
## Call:
## ppr(formula = left ~ ., data = train_set, sm.method = "supsmu",
##     nterms = 2, max.terms = 10, bass = 3)
##
## Goodness of fit:
##  2 terms  3 terms  4 terms  5 terms  6 terms  7 terms  8 terms  9 terms
## 458.1165 450.3077 444.7791 425.0369 398.9121   0.0000   0.0000   0.0000
## 10 terms
##   0.0000
##
## Projection direction vectors ('alpha'):
##                       term 1        term 2
## satisfaction_level    -0.6151367730  0.0619886607
## last_evaluation       -0.3415662171  0.3008649519
## number_project        -0.0860166884  0.0542528923
## average_montly_hours  -0.0008288968  0.0009933959
## time_spend_company     0.3424034847  0.0165443235
## Work_accident         -0.0111389954 -0.0136409010
## promotion_last_5years  0.0348483473 -0.0556967020
## departmentaccounting  -0.2019829878 -0.2974768384
## departmenthr          -0.1700990232 -0.2939100305
## departmentIT          -0.1761064836 -0.3022499068
## departmentmanagement  -0.2134019845 -0.3074845890
## departmentmarketing   -0.1882018881 -0.2989378943
## departmentproduct_mng -0.1954305366 -0.2997171080
## departmentRandD       -0.2190203543 -0.3008401130
## departmentsales       -0.1901833335 -0.3008587742
## departmentsupport     -0.1935255961 -0.2984949925
## departmenttechnical   -0.1929146696 -0.2975618076
## salary.L              -0.0152988879 -0.0204003363
## salary.Q              -0.0046295102 -0.0089782745
##
## Coefficients of ridge terms ('beta'):
##    term 1    term 2
## 0.1891072 0.2283901
```

```r
fit1.ppr <- update(fit.ppr, bass=5, nterms=4)
summary(fit1.ppr)
```

```
## Call:
## ppr(formula = left ~ ., data = train_set, sm.method = "supsmu",
##     nterms = 4, max.terms = 10, bass = 5)
##
## Goodness of fit:
##  4 terms  5 terms  6 terms  7 terms  8 terms  9 terms 10 terms
## 439.0660 376.8213 407.6758 389.1513 378.7404 363.6315 375.1392
##
## Projection direction vectors ('alpha'):
##                       term 1        term 2        term 3        term 4
## satisfaction_level     0.5246101071  0.0821527004 -0.4626097031 -0.6216385685
## last_evaluation       -0.1111719681  0.2222217399  0.5227893260 -0.5131895452
## number_project        -0.0233810181  0.0469072274  0.0681764746 -0.0722726958
## average_montly_hours  -0.0002543281  0.0007008554  0.0014051646 -0.0011989502
## time_spend_company     0.0086370060  0.0159715946 -0.1166035597  0.3843426731
## Work_accident          0.0204419607 -0.0058231591 -0.0645300834 -0.0053282882
## promotion_last_5years -0.0129181044 -0.0289567617 -0.0417755094  0.0330939201
## departmentaccounting   0.2639941937 -0.3064307641  0.2289947272 -0.1444833003
## departmenthr           0.2585096361 -0.3025048341  0.2354563851 -0.1136633821
## departmentIT           0.2702227225 -0.3084075130  0.1996727483 -0.1207528794
## departmentmanagement   0.2882168947 -0.3135122016  0.2075844823 -0.1700410238
## departmentmarketing    0.2674376236 -0.3071703602  0.2174623484 -0.1406405285
## departmentproduct_mng  0.2670044950 -0.3075417340  0.2054318428 -0.1432513156
## departmentRandD        0.2445853665 -0.3030297679  0.2295527857 -0.1521800954
## departmentsales        0.2673503905 -0.3077440722  0.2137245482 -0.1298299732
## departmentsupport      0.2689174581 -0.3052600846  0.2206816817 -0.1406155373
## departmenttechnical    0.2622364921 -0.3048109220  0.2247776649 -0.1356622684
## salary.L               0.0512785435 -0.0115469477 -0.0957008549 -0.0164095097
## salary.Q               0.0279803000 -0.0055066764 -0.0379500756 -0.0090973935
##
## Coefficients of ridge terms ('beta'):
##    term 1    term 2    term 3    term 4
## 0.2129129 0.2280402 0.3348101 0.3047044
```

```r
# PREDICTION
yhat.ppr <- predict(fit1.ppr, newdata=test_set)
yhat.ppr <- scale(yhat.ppr,center = min(yhat.ppr),scale = max(yhat.ppr)-min(yhat.ppr))

# AUC AND ROC CURVE
AUC.PPR <- ci.cvAUC(predictions=yhat.ppr, labels=yobs, folds=1:length(yhat.ppr), confide
```

```
## Warning in if (class(predictions) == "list" | class(labels) == "list") {: the
## condition has length > 1 and only the first element will be used
```

```
## $cvAUC
## [1] 0.9648913
##
## $se
## [1] 0.003432747
##
## $ci
## [1] 0.9581632 0.9716194
##
## $confidence
## [1] 0.95
```

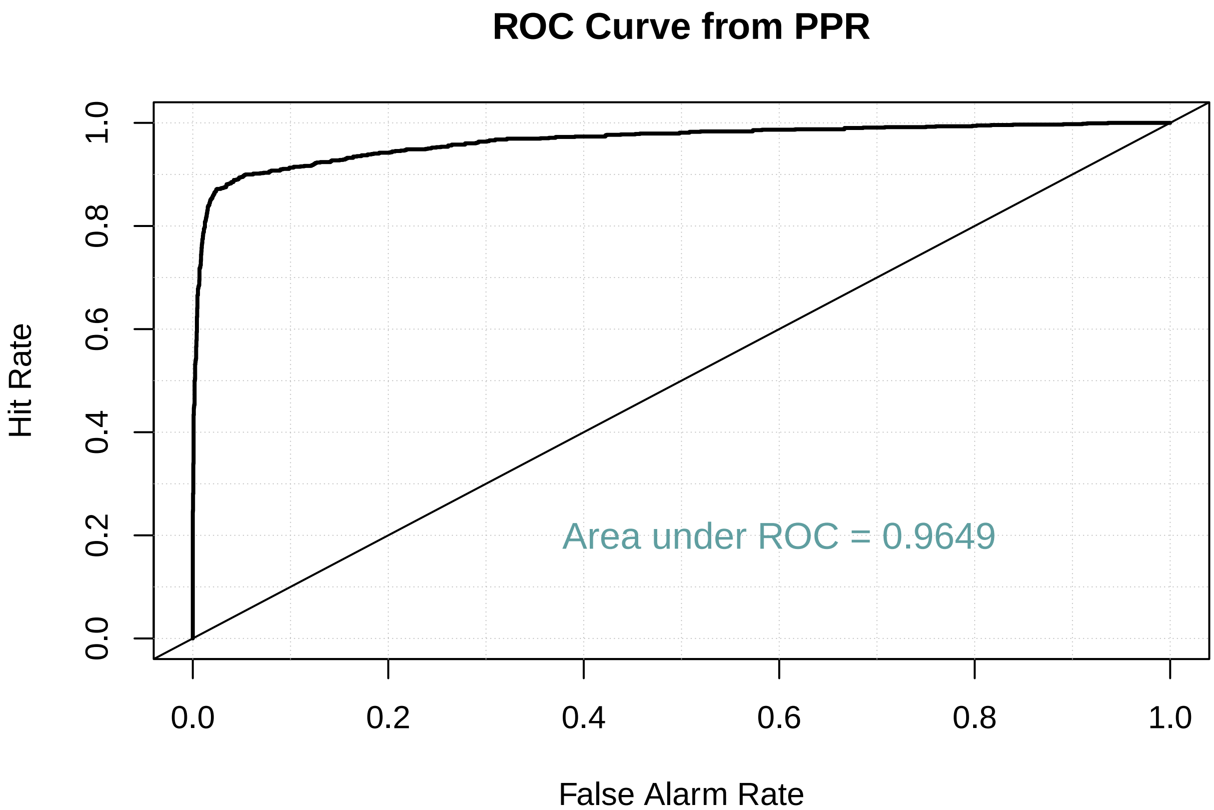
```
auc.ci <- round(AUC.PPR$ci, digits=4)
library(verification)
mod.ppr <- verify(obs=yobs, pred=yhat.ppr)
```

```
## If baseline is not included, baseline values  will be calculated from the  sample obs
```

```
roc.plot(mod.ppr, plot.thres = NULL,  main="ROC Curve from PPR")
```

```
## Warning in roc.plot.default(c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, : Large
## amount of unique predictions used as thresholds. Consider specifying thresholds.
```

```
text(x=0.6, y=0.2, paste("Area under ROC =", round(AUC.PPR$cvAUC, digits=4),
    sep=" "), col="cadetblue", cex=1.2)
```



The area under the curve according to PPR model is 0.9649

9 Results and Comparison

```
Measure <- c(log_reg_lasso,round(AUC.RF, digits=4),round(AUC.GAM, digits=4),round(AUC.MA
Measures <- data.frame("Method"= c("LASSO","Random Forest","GAM","MARS","PPR"), "AUC"= M
```

```
##          Method    AUC
## 1         LASSO 0.8145
## 2 Random Forest 0.9901
## 3           GAM 0.9730
## 4          MARS 0.9751
## 5           PPR 0.9649
```

```
knitr::kable(Measures, align = "lc")
```

Method	AUC
LASSO	0.8145
Random Forest	0.9901
GAM	0.9730
MARS	0.9751
PPR	0.9649

By using AUC as a criteria, we see that random forest model outperforms all the other models since it has the highest AUC while logistic regression model perform the least.

From the results of the five models, we see that the important predictor variables that help best predict the employee retention are satisfaction_level,last_evaluation,number_project and time_spend_company. Therefore, the company has to pay more attention to these variables and find ways to improve in these areas to help maximize the rate at which employees stay in the company.

We also observed from the five models that they seem not to be good for categorical variables.