project

Ning Hu

2023-04-15

```
library(tidyr)
library(ggplot2)
library(randomForest)
```

```
## randomForest 4.7-1.1
```

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

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

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

```
library(caret)
```

```
## Loading required package: lattice
```

```
library(glmnet)
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

```
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
## Loaded glmnet 4.1-6
```

```
## Data exploratory
bank = read.csv('data/bank-additional/bank-additional-full.csv', sep = ';')
summary(bank)
```

```
##       age            job              marital           education        
##  Min.   :17.00   Length:41188       Length:41188       Length:41188      
##  1st Qu.:32.00   Class :character   Class :character   Class :character  
##  Median :38.00   Mode  :character   Mode  :character   Mode  :character  
##  Mean   :40.02                                                           
##  3rd Qu.:47.00                                                           
##  Max.   :98.00                                                           
##    default            housing              loan             contact         
##  Length:41188       Length:41188       Length:41188       Length:41188      
##  Class :character   Class :character   Class :character   Class :character  
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character  
##                                                                             
##                                                                             
##                                                                             
##     month           day_of_week           duration         campaign     
##  Length:41188       Length:41188       Min.   :   0.0   Min.   : 1.000  
##  Class :character   Class :character   1st Qu.: 102.0   1st Qu.: 1.000  
##  Mode  :character   Mode  :character   Median : 180.0   Median : 2.000  
##                                        Mean   : 258.3   Mean   : 2.568  
##                                        3rd Qu.: 319.0   3rd Qu.: 3.000  
##                                        Max.   :4918.0   Max.   :56.000  
##      pdays          previous       poutcome          emp.var.rate      
##  Min.   :  0.0   Min.   :0.000   Length:41188       Min.   :-3.40000  
##  1st Qu.:999.0   1st Qu.:0.000   Class :character   1st Qu.:-1.80000  
##  Median :999.0   Median :0.000   Mode  :character   Median : 1.10000  
##  Mean   :962.5   Mean   :0.173                      Mean   : 0.08189  
##  3rd Qu.:999.0   3rd Qu.:0.000                      3rd Qu.: 1.40000  
##  Max.   :999.0   Max.   :7.000                      Max.   : 1.40000  
##  cons.price.idx  cons.conf.idx     euribor3m      nr.employed  
##  Min.   :92.20   Min.   :-50.8   Min.   :0.634   Min.   :4964  
##  1st Qu.:93.08   1st Qu.:-42.7   1st Qu.:1.344   1st Qu.:5099  
##  Median :93.75   Median :-41.8   Median :4.857   Median :5191  
##  Mean   :93.58   Mean   :-40.5   Mean   :3.621   Mean   :5167  
##  3rd Qu.:93.99   3rd Qu.:-36.4   3rd Qu.:4.961   3rd Qu.:5228  
##  Max.   :94.77   Max.   :-26.9   Max.   :5.045   Max.   :5228  
##       y            
##  Length:41188      
##  Class :character  
##  Mode  :character  
##                    
##                    
##                    
```

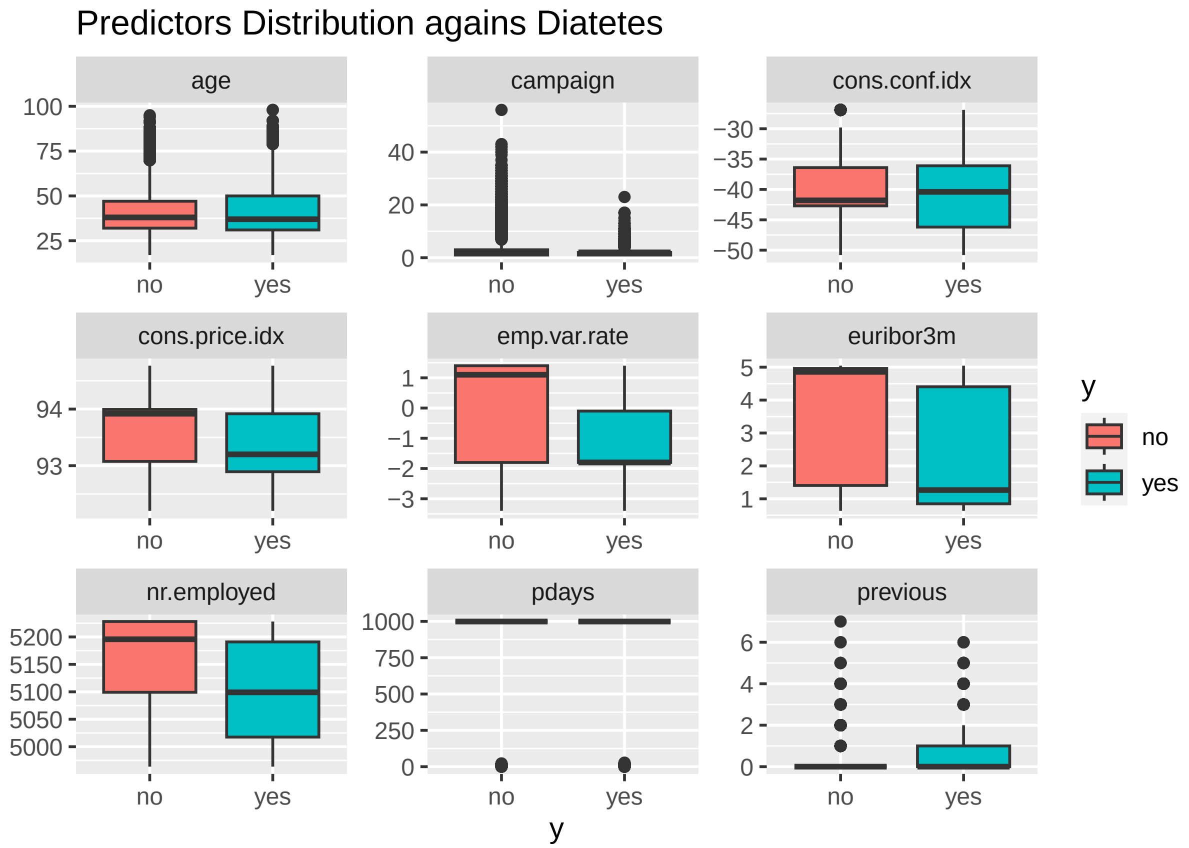
```
names(bank)
```

```
##  [1] "age"            "job"            "marital"        "education"     
##  [5] "default"        "housing"        "loan"           "contact"       
##  [9] "month"          "day_of_week"    "duration"       "campaign"      
## [13] "pdays"          "previous"       "poutcome"       "emp.var.rate"  
## [17] "cons.price.idx" "cons.conf.idx"  "euribor3m"      "nr.employed"   
## [21] "y"
```

```r
# remove predictors
# default a extremely unbalanced
# duration is unknown while predicting
remove.variable = c('default', 'duration')
bank = bank[,!names(bank) %in% remove.variable]


dummy.variables = c('job', 'marital', 'education',
                    'housing', 'loan', 'contact',
                    'month', 'day_of_week','poutcome',
                    'y')

for(name in dummy.variables) {
  bank[name] = as.factor(bank[[name]])
}
summary(bank)
```

```
##       age                   job            marital
##  Min.   :17.00   admin.     :10422   divorced: 4612
##  1st Qu.:32.00   blue-collar: 9254   married :24928
##  Median :38.00   technician : 6743   single  :11568
##  Mean   :40.02   services   : 3969   unknown :   80
##  3rd Qu.:47.00   management : 2924
##  Max.   :98.00   retired    : 1720
##                  (Other)    : 6156
##                 education        housing           loan           contact
##  university.degree  :12168   no     :18622   no     :33950   cellular :26144
##  high.school        : 9515   unknown:  990   unknown:  990   telephone:15044
##  basic.9y           : 6045   yes    :21576   yes    : 6248
##  professional.course: 5243
##  basic.4y           : 4176
##  basic.6y           : 2292
##  (Other)            : 1749
##      month       day_of_week    campaign          pdays          previous
##  may    :13769   fri:7827    Min.   : 1.000   Min.   :  0.0   Min.   :0.000
##  jul    : 7174   mon:8514    1st Qu.: 1.000   1st Qu.:999.0   1st Qu.:0.000
##  aug    : 6178   thu:8623    Median : 2.000   Median :999.0   Median :0.000
##  jun    : 5318   tue:8090    Mean   : 2.568   Mean   :962.5   Mean   :0.173
##  nov    : 4101   wed:8134    3rd Qu.: 3.000   3rd Qu.:999.0   3rd Qu.:0.000
##  apr    : 2632               Max.   :56.000   Max.   :999.0   Max.   :7.000
##  (Other): 2016
##         poutcome      emp.var.rate      cons.price.idx  cons.conf.idx
##  failure    : 4252   Min.   :-3.40000   Min.   :92.20   Min.   :-50.8
##  nonexistent:35563   1st Qu.:-1.80000   1st Qu.:93.08   1st Qu.:-42.7
##  success    : 1373   Median : 1.10000   Median :93.75   Median :-41.8
##                      Mean   : 0.08189   Mean   :93.58   Mean   :-40.5
##                      3rd Qu.: 1.40000   3rd Qu.:93.99   3rd Qu.:-36.4
##                      Max.   : 1.40000   Max.   :94.77   Max.   :-26.9
##
##    euribor3m      nr.employed     y
##  Min.   :0.634   Min.   :4964   no :36548
##  1st Qu.:1.344   1st Qu.:5099   yes: 4640
##  Median :4.857   Median :5191
##  Mean   :3.621   Mean   :5167
##  3rd Qu.:4.961   3rd Qu.:5228
##  Max.   :5.045   Max.   :5228
##
```

```
#bank.scaled = bank
# scale all continuous predictors
#for (name in setdiff(names(bank), dummy.variables)) {
#  bank.scaled[name] = scale(bank.scaled[name])
#}

bank[,!(names(bank) %in% dummy.variables[1:9])] %>%
  gather(-y, key = 'var', value = 'value') %>%
  ggplot(aes(x = y, y = value)) +
  geom_boxplot(aes(fill = y)) +
  facet_wrap(~var, scales = 'free') +
  ylab('')+
  ggtitle('Predictors Distribution agains Diatetes')
```



```
# add column: if the client was last contacted from a previous campaign
#bank$is.pcontacted = as.factor((bank$pdays == 999))

# missing value only account for 8% of the whole data
# no additaional process on missing value, treat them as a category
miss.name = 'unknown'
print('missing values')
```

```
## [1] "missing values"
```

```
for (name in dummy.variables) {
  print(paste(name, sum(bank[name] == miss.name), sep=':'))
}
```

```
## [1] "job:330"
## [1] "marital:80"
## [1] "education:1731"
## [1] "housing:990"
## [1] "loan:990"
## [1] "contact:0"
## [1] "month:0"
## [1] "day_of_week:0"
## [1] "poutcome:0"
## [1] "y:0"
```

```
# total records with missing values
sum(rowSums(bank==miss.name)>0)
```

```
## [1] 2943
```

```
pr.curve = function(pred, y) {
  recalls = c()
  precisions = c()

  for (threshold in seq(0, 1, by=0.01)) {
    yhat = ifelse(pred>threshold, 'yes', 'no')
    yhat = factor(yhat, levels = c('no', 'yes'), labels = c('no', 'yes'))
    recalls = c(recalls, recall(yhat, y, relevant = 'yes'))
    precisions = c(precisions, precision(yhat, y, relevant = 'yes'))
  }
  data.frame(recall = recalls, precision = precisions, threshold=seq(0, 1, by=0.01))
}

set.seed(5420)

# separate to train and test
train.size = 2*nrow(bank)/3
train.rows = sample(nrow(bank), train.size)


rf = randomForest(y~., data = bank, subset = train.rows, importance = TRUE)
rf
```

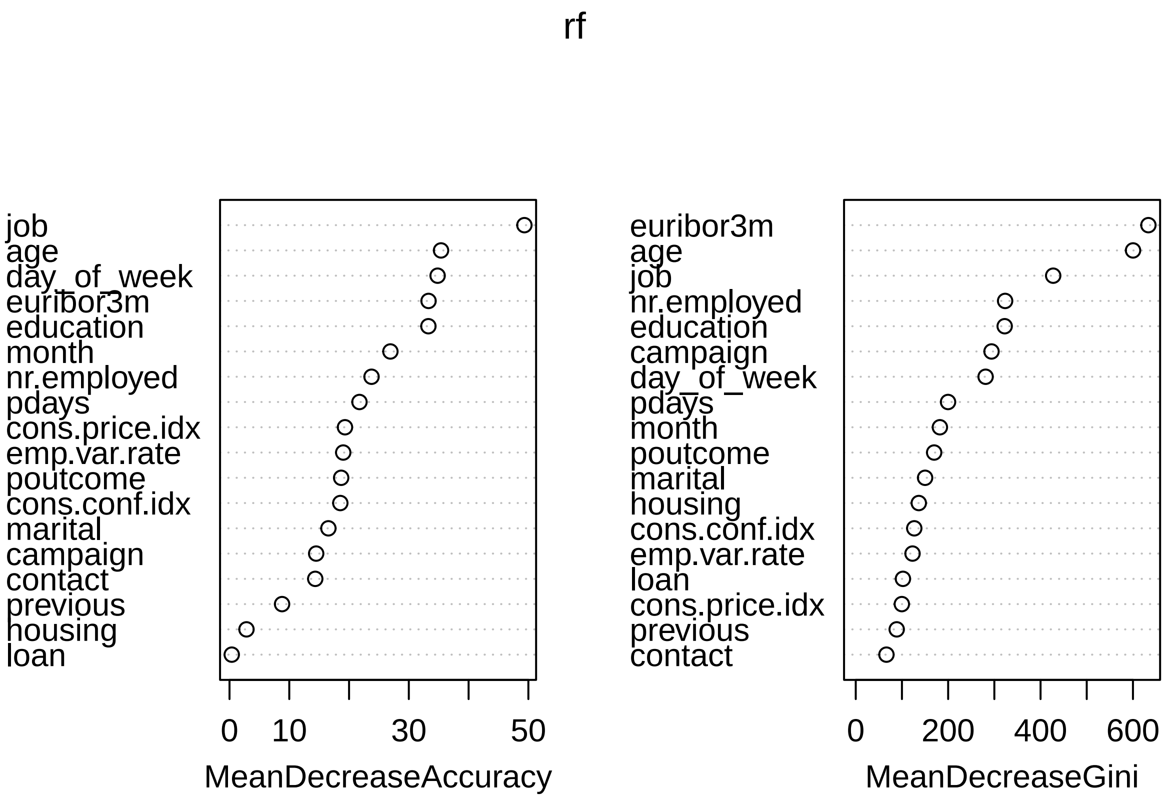
```
##
## Call:
##  randomForest(formula = y ~ ., data = bank, importance = TRUE,      subset = train.rows)
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 4
##
```

```
##         OOB estimate of  error rate: 10.43%
## Confusion matrix:
##        no yes class.error
## no  23669 642   0.0264078
## yes  2223 924   0.7063870
```

```
yhat.rf = predict(rf, newdata = bank[-train.rows,], type='prob')
#confusionMatrix(yhat.norm, bank[-train.rows, 'y'])
pr.rf = pr.curve(yhat.rf[,'yes'], bank[-train.rows, 'y'])

pr.rf$type = 'RF'


importance(rf)
```

```
##                        no         yes MeanDecreaseAccuracy MeanDecreaseGini
## age            36.2958035  -3.79177710            35.3859048        600.14474
## job            56.7230287 -14.01171493            49.3340220        427.35221
## marital        17.9019050  -2.58654343            16.5430319        150.10480
## education      34.1963001   0.50296773            33.2733214        322.35301
## housing         3.3929888  -0.34948738             2.8375034        136.48883
## loan            0.3730673   0.06219239             0.3740652        102.14547
## contact        11.1040111  25.52766861            14.3441883         66.43997
## month          26.2315182 -10.94829139            26.9175663        182.11714
## day_of_week    36.3727999  -1.07743292            34.8188780        281.08233
## campaign       10.2859199  10.42018414            14.4988762        293.88781
## pdays           8.1561100  26.50234707            21.7410649        199.78561
## previous        7.8032107   3.24009428             8.7924984         88.68205
## poutcome       13.6088217  14.04864784            18.6786793        169.68921
## emp.var.rate   18.3148838   2.22817635            19.0233140        122.89921
## cons.price.idx 19.2723364 -13.49077101            19.3202683         99.75588
## cons.conf.idx  18.2262713  -9.06997109            18.5368083        126.67364
## euribor3m      31.0114341   6.41700313            33.2980190        633.41846
## nr.employed    21.0422642  15.78838362            23.7608870        323.32416
```

```
varImpPlot(rf)
```

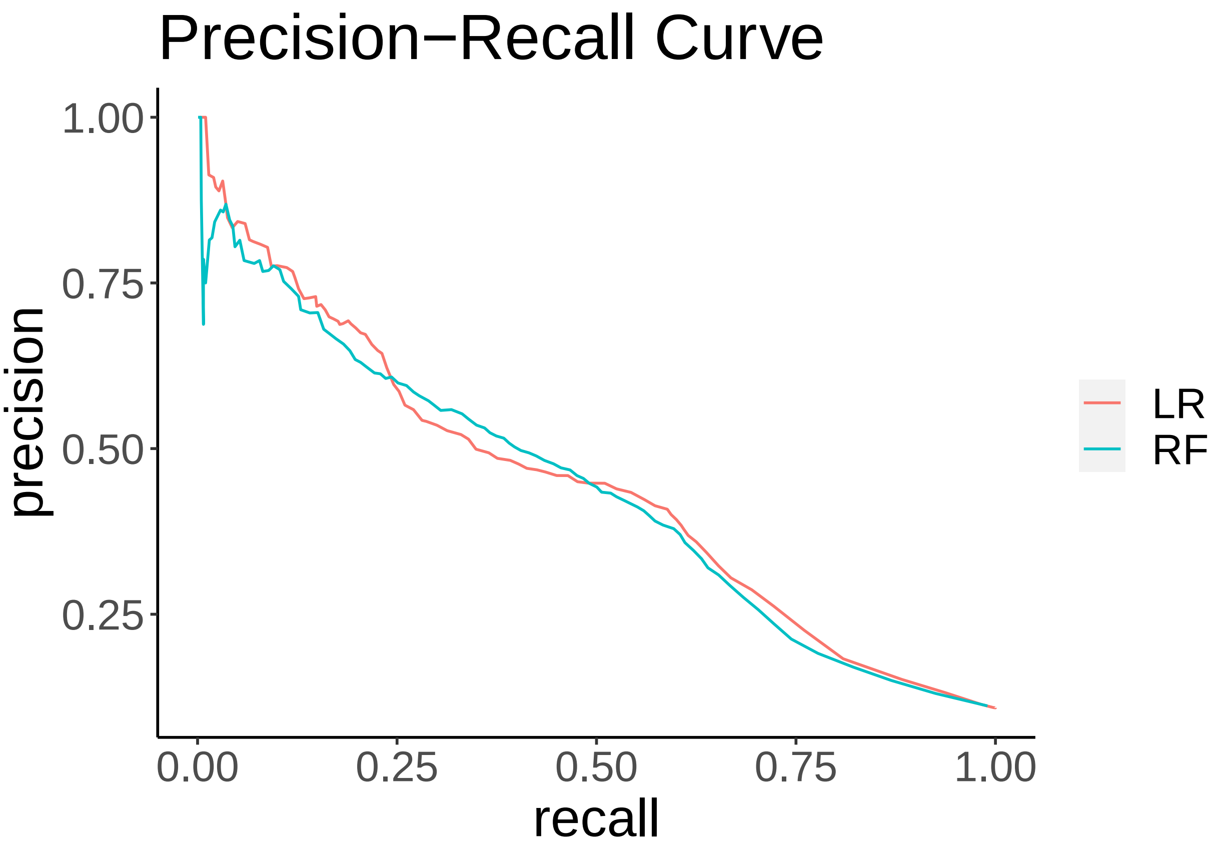


```
## LR stepwise
trControl = trainControl(method = 'cv', number = 10)
lr.step = train(y~.,
      data = bank[train.rows,],
      method = 'glmStepAIC',
      family = 'binomial',
      direction = 'backward',
      trControl = trControl,
      trace = F
    )
summary(lr.step$finalModel)
```

```
##
## Call:
## NULL
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1173  -0.3901  -0.3257  -0.2674   3.0167
##
## Coefficients:
##                              Estimate Std. Error z value Pr(>|z|)
## (Intercept)                -3.110e+02  3.234e+01  -9.617  < 2e-16 ***
## `jobblue-collar`           -1.387e-01  6.624e-02  -2.094 0.036227 *
## jobmanagement              -1.512e-01  8.884e-02  -1.702 0.088703 .
## jobretired                  2.263e-01  8.937e-02   2.533 0.011325 *
## jobstudent                  2.387e-01  1.127e-01   2.118 0.034183 *
## maritalsingle               8.790e-02  4.936e-02   1.781 0.074969 .
## educationbasic.9y          -1.600e-01  7.457e-02  -2.145 0.031920 *
## educationuniversity.degree  1.155e-01  5.072e-02   2.276 0.022825 *
## contacttelephone           -6.945e-01  7.767e-02  -8.942  < 2e-16 ***
## monthaug                    6.793e-01  1.136e-01   5.981 2.21e-09 ***
## monthdec                    7.301e-01  2.227e-01   3.279 0.001042 **
## monthjun                   -8.244e-01  1.171e-01  -7.040 1.92e-12 ***
## monthmar                    1.809e+00  1.435e-01  12.602  < 2e-16 ***
## monthmay                   -3.212e-01  7.515e-02  -4.274 1.92e-05 ***
## monthnov                   -2.450e-01  8.560e-02  -2.862 0.004205 **
## monthoct                    3.360e-01  1.263e-01   2.661 0.007780 **
## monthsep                    5.824e-01  1.652e-01   3.525 0.000423 ***
## day_of_weekmon             -2.069e-01  5.646e-02  -3.665 0.000247 ***
## day_of_weekwed              1.071e-01  5.524e-02   1.939 0.052492 .
## campaign                   -4.640e-02  1.120e-02  -4.142 3.45e-05 ***
## pdays                      -1.029e-03  2.292e-04  -4.491 7.10e-06 ***
## poutcomenonexistent         4.990e-01  6.941e-02   7.188 6.56e-13 ***
## poutcomesuccess             8.003e-01  2.302e-01   3.477 0.000508 ***
## emp.var.rate               -1.688e+00  1.466e-01 -11.509  < 2e-16 ***
## cons.price.idx              2.571e+00  2.326e-01  11.053  < 2e-16 ***
## cons.conf.idx               3.574e-02  5.480e-03   6.521 7.00e-11 ***
## nr.employed                 1.365e-02  2.116e-03   6.451 1.11e-10 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 19553  on 27457  degrees of freedom
## Residual deviance: 15330  on 27431  degrees of freedom
## AIC: 15384
##
## Number of Fisher Scoring iterations: 6
```

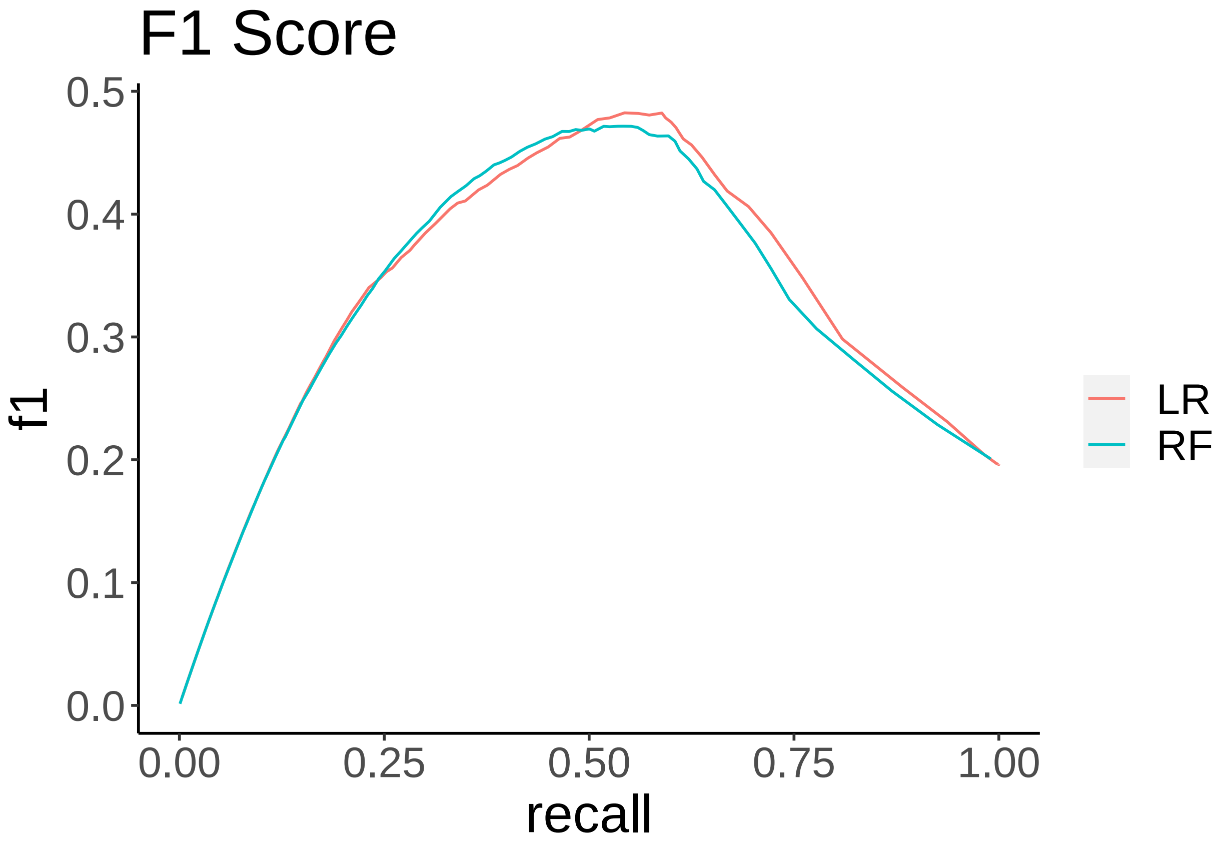
```
yhat.lr.step = predict.train(lr.step,
                       newdata = bank[-train.rows,],
                       type="prob")
pr.lr.step = pr.curve(yhat.lr.step[,'yes'], bank[-train.rows, 'y'])
pr.lr.step$type = 'LR'

# plot pr curve
data = rbind(pr.rf, pr.lr.step)
ggplot(data = data)+
  geom_line(aes(x = recall, y=precision, col = type))+
  ggtitle('Precision-Recall Curve')+
  labs(color = NULL)+
  theme(text = element_text(size = 20),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black")
      )
```

```
## Warning: Removed 14 rows containing missing values (‘geom_line()‘).
```



```
# F1 score
data$f1 = 2*data$recall*data$precision/(data$recall+data$precision)
ggplot(data = data)+
  geom_line(aes(x = recall, y=f1, col=type))+
  ggtitle('F1 Score')+
  labs(color = NULL)+
  theme(text = element_text(size = 20),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black")
      )
```

```
## Warning: Removed 14 rows containing missing values (`geom_line()`).
```


F1 Score
f1
0.5
0.4
0.3
0.2
0.1
0.0
0.00
0.25
0.50
0.75
1.00
recall
LR
RF