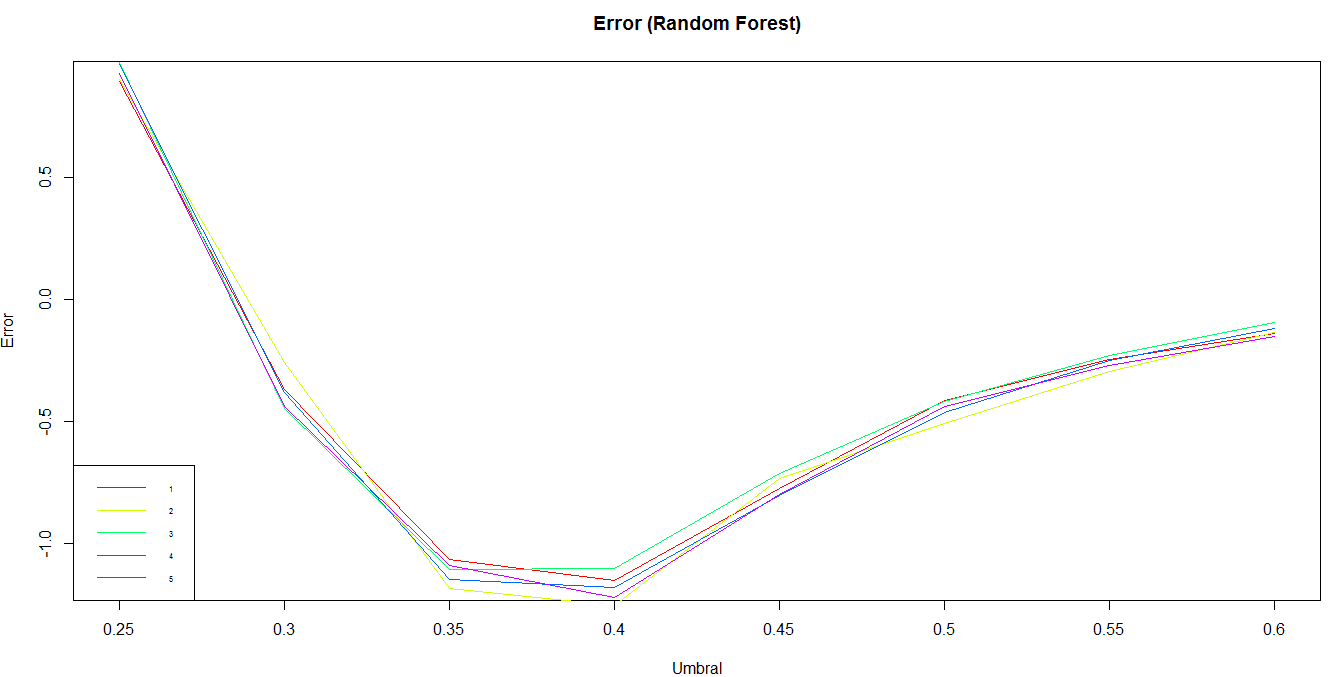
1ra corridad



h.rf\_ctrl <- list(ctrl1 = list(ntree = 550, mtry = 6)

,ctrl2 = list(ntree = 550, mtry = 7)

,ctrl3 = list(ntree = 600, mtry = 5)

,ctrl4 = list(ntree = 600, mtry = 6)

,ctrl5 = list(ntree = 600, mtry = 7)

)

[[1]]

[1] 0.8941918 -0.3672859 -1.0682255 -1.1509716 -0.7756780 -0.4158659 -0.2439675 -0.1387999

[[2]]

[1] 0.9027333 -0.2589152 -1.1867393 -1.2550715 -0.7329703 -0.5071535 -0.2936152 -0.1329276

[[3]]

[1] 0.97213325 -0.44843049 -1.10666240 -1.10239163 -0.71535341 -0.41960282 -0.22955370 -0.09449071

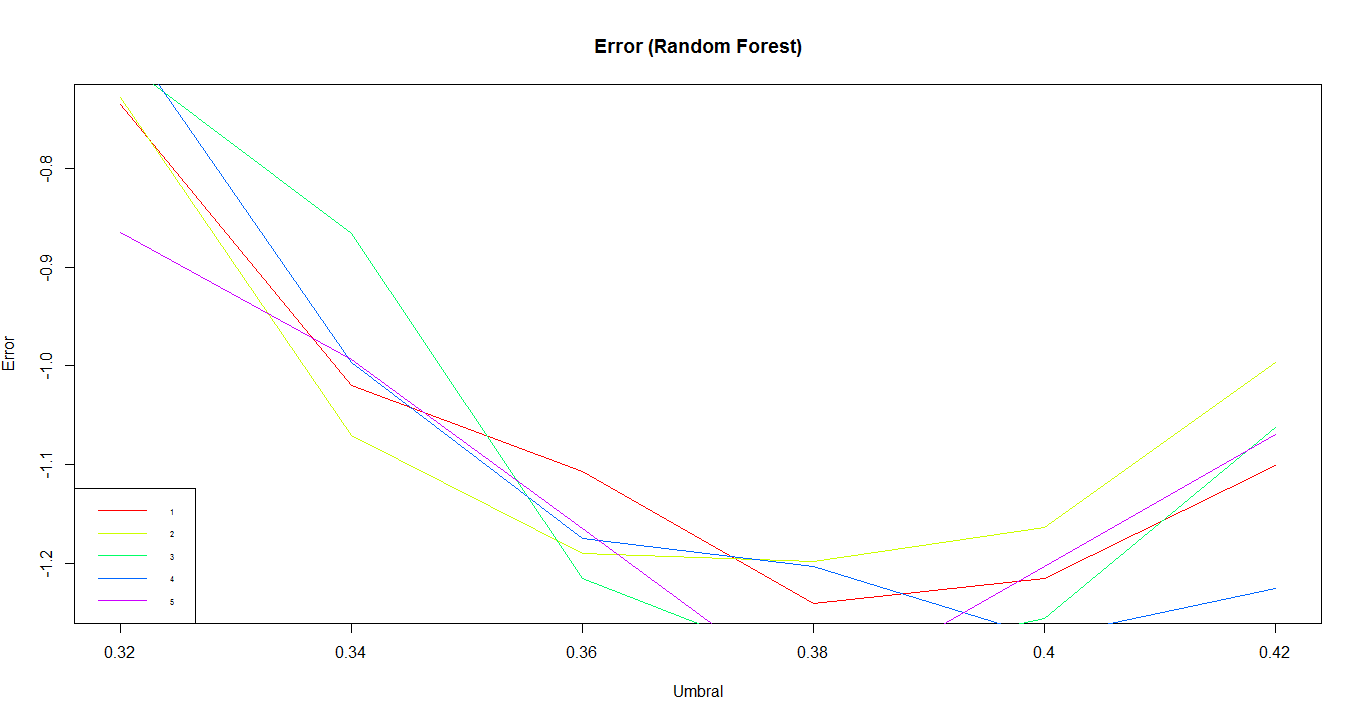
[[4]]

[1] 0.9662609 -0.3795644 -1.1467008 -1.1808670 -0.8055734 -0.4617766 -0.2503737 -0.1179799

[[5]]

[1] 0.9272902 -0.4382874 -1.0901132 -1.2230408 -0.8002349 -0.4398890 -0.2701260 -0.1521461

2da corrida



h.rf\_ctrl <- list(ctrl2 = list(ntree = 550, mtry = 7)

,ctrl4 = list(ntree = 600, mtry = 6)

,ctrl5 = list(ntree = 600, mtry = 7)

,ctrl6 = list(ntree = 600, mtry = 8)

,ctrl7 = list(ntree = 700, mtry = 7)

)

R version 3.5.2 (2018-12-20) -- "Eggshell Igloo"

Copyright (C) 2018 The R Foundation for Statistical Computing

Platform: x86\_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.

You are welcome to redistribute it under certain conditions.

Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.

Type 'contributors()' for more information and

'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or

'help.start()' for an HTML browser interface to help.

Type 'q()' to quit R.

[Workspace loaded from C:/Users/4051/Datos/ProyectoBgD/machineLearning/.RData]

> source("Utility/utils.R")

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

randomForest 4.6-14

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

Attaching package: ‘randomForest’

The following object is masked from ‘package:dplyr’:

combine

Loaded gbm 2.1.5

Warning messages:

1: package ‘e1071’ was built under R version 3.5.3

2: package ‘rpart.plot’ was built under R version 3.5.3

3: package ‘dplyr’ was built under R version 3.5.3

4: package ‘randomForest’ was built under R version 3.5.3

5: package ‘gbm’ was built under R version 3.5.3

>

> #Vector de umbrales para la predición

> h.umbral <- seq(0.25, to = 0.6, by = 0.05)

>

> #Se carga los datos con las ETL generales

> h.data <- loadData()

>

> #Generar la particion de test & train

> h.ntrain <- nTrain

> h.part <- partition\_train\_test(h.data, ntrain = h.ntrain)

> h.train <- h.part$train

> h.test <- h.part$test

>

>

> # Particion para cross validation

> h.cv\_part <- partition\_cv(df = h.train)

>

> # Formulas

> h.formula <- 'as.factor(Churn) ~ .'

> fn\_err <- fn\_err\_cost #fn\_err\_cla

>

> #Definir el valor de mtry y ntree

> h.rf\_ctrl <- list(ctrl1 = list(ntree = 550, mtry = 6)

+ ,ctrl2 = list(ntree = 550, mtry = 7)

+ ,ctrl3 = list(ntree = 600, mtry = 5)

+ ,ctrl4 = list(ntree = 600, mtry = 6)

+ ,ctrl5 = list(ntree = 600, mtry = 7)

+ )

>

>

> # Entrenamiento de randomforest train, formula, ctrl

> h.randomForest\_fit <- rf\_fit\_ctrl(h.train,

+ h.formula,

+ h.rf\_ctrl)

>

>

> # Importancia de las variables para el primer control

> # importance(h.randomForest\_fit[[1]])

> # varImpPlot(h.randomForest\_fit[[1]])

>

> #Probabilidad en test

> h.randomForest\_prob <- rf\_prob(h.randomForest\_fit,

+ h.test)

>

> #Predicciones utilizando el umbral

> h.randomForest\_pred <- rf\_pred(h.randomForest\_prob,

+ h.umbral)

>

> #Error en test

> h.randomForest\_predErr <- rf\_pred\_err(h.randomForest\_pred,

+ h.test$Churn)

>

> #Plot del error en RamdomForest

> plot\_umbral\_err(h.randomForest\_predErr, main = 'Error (Random Forest)', umbral = h.umbral)

> View()

Error in View : invalid first argument

> View(h.randomForest\_predErr)

> h.randomForest\_predErr[[1]]

[1] 0.8941918 -0.3672859 -1.0682255 -1.1509716 -0.7756780 -0.4158659 -0.2439675 -0.1387999

> pit

Error: object 'pit' not found

> print(h.randomForest\_predErr)

[[1]]

[1] 0.8941918 -0.3672859 -1.0682255 -1.1509716 -0.7756780 -0.4158659 -0.2439675 -0.1387999

[[2]]

[1] 0.9027333 -0.2589152 -1.1867393 -1.2550715 -0.7329703 -0.5071535 -0.2936152 -0.1329276

[[3]]

[1] 0.97213325 -0.44843049 -1.10666240 -1.10239163 -0.71535341 -0.41960282 -0.22955370 -0.09449071

[[4]]

[1] 0.9662609 -0.3795644 -1.1467008 -1.1808670 -0.8055734 -0.4617766 -0.2503737 -0.1179799

[[5]]

[1] 0.9272902 -0.4382874 -1.0901132 -1.2230408 -0.8002349 -0.4398890 -0.2701260 -0.1521461

> h.randomForest\_fit

Call:

randomForest(formula = as.formula(formula), data = train, ntree = ctrl[[i]]$ntree, mtry = ctrl[[i]]$mtry)

Type of random forest: classification

Number of trees: 550

No. of variables tried at each split: 7

OOB estimate of error rate: 28.29%

Confusion matrix:

0 1 class.error

0 26042 573 0.02152921

1 10020 810 0.92520776

[[2]]

Call:

randomForest(formula = as.formula(formula), data = train, ntree = ctrl[[i]]$ntree, mtry = ctrl[[i]]$mtry)

Type of random forest: classification

Number of trees: 600

No. of variables tried at each split: 6

OOB estimate of error rate: 28.35%

Confusion matrix:

0 1 class.error

0 26101 514 0.01931242

1 10101 729 0.93268698

[[3]]

Call:

randomForest(formula = as.formula(formula), data = train, ntree = ctrl[[i]]$ntree, mtry = ctrl[[i]]$mtry)

Type of random forest: classification

Number of trees: 600

No. of variables tried at each split: 7

OOB estimate of error rate: 28.24%

Confusion matrix:

0 1 class.error

0 26069 546 0.02051475

1 10028 802 0.92594645

[[4]]

Call:

randomForest(formula = as.formula(formula), data = train, ntree = ctrl[[i]]$ntree, mtry = ctrl[[i]]$mtry)

Type of random forest: classification

Number of trees: 600

No. of variables tried at each split: 8

OOB estimate of error rate: 28.31%

Confusion matrix:

0 1 class.error

0 26001 614 0.0230697

1 9985 845 0.9219760

[[5]]

Call:

randomForest(formula = as.formula(formula), data = train, ntree = ctrl[[i]]$ntree, mtry = ctrl[[i]]$mtry)

Type of random forest: classification

Number of trees: 700

No. of variables tried at each split: 7

OOB estimate of error rate: 28.37%

Confusion matrix:

0 1 class.error

0 26073 542 0.02036446

1 10080 750 0.93074792

> print(h.randomForest\_predErr)

[[1]]

[1] -0.7356395 -1.0196455 -1.1071962 -1.2401239 -1.2150331 -1.1007901

[[2]]

[1] -0.7281657 -1.0708947 -1.1899423 -1.1984839 -1.1637839 -0.9966902

[[3]]

[1] -0.6897288 -0.8664318 -1.2155669 -1.3063207 -1.2561392 -1.0623532

[[4]]

[1] -0.6592996 -0.9961563 -1.1749947 -1.2027546 -1.2748238 -1.2251762

[[5]]

[1] -0.8653641 -0.9934871 -1.1643177 -1.3362161 -1.2032885 -1.0692932