# script_glm_thr.R

*sergio*

*2020-05-27*

```
library(glmnet)
```

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

```
## Loaded glmnet 3.0-2
```

```
source('../utils/utils_oblig.R')
```

```
## Loading required package: caret
```

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

```
## Loading required package: ggplot2
```

```
## Registered S3 methods overwritten by 'ggplot2':
##   method         from
##   [.quosures     rlang
##   c.quosures     rlang
##   print.quosures rlang
```

```
set.seed(117)

script.name <- 'glm_thr'

script.date <- date()

script.start <- Sys.time()

print('Start')
```

```
## [1] "Start"
```

```
# leer el archivo dataset.csv de la carpeta

dataset <- read.csv('../data/dataset.csv')

# ver la estructura del dataset

# str(dataset)

# asignar el nombre del jugador como nombre de la fila

rownames(dataset) <- dataset$CustomerID

df <- na.omit(dataset[,-1])

df$ServiceArea <- NULL

print('** Distribucion a-priori de la variable a predecir')
```

```
## [1] "** Distribucion a-priori de la variable a predecir"
```

```r
print(prop.table(table(df$Churn)))
```

```
##
##        No       Yes
## 0.7131871 0.2868129
```

```r
df.part <- train_dev_partition(df, p = 0.9)

df.thr_vec <- seq(0.1, 0.9, 0.05)

df.fn_summary <- function(data, lev = NULL, model = NULL) {
  fn_summaryUtilityThr(data, df.thr_vec)
}

df.metric <- 'utility'

df.form <- Churn ~ .

print('** GLM')
```

```
## [1] "** GLM"
```

```r
df.glm.ctrl <- trainControl(method = 'cv',
                            number = 5,
                            verboseIter = TRUE,
                            classProbs = TRUE,
                            search = 'random',
                            summaryFunction = df.fn_summary)

df.glm <- train(form = df.form,
                data = df.part$train,
                method = 'glmnet',
                family = 'binomial',
                trControl = df.glm.ctrl,
                tuneLength = 5,
                metric = df.metric)
```

```
## + Fold1: alpha=0.26189, lambda=0.10504
## - Fold1: alpha=0.26189, lambda=0.10504
## + Fold1: alpha=0.17301, lambda=3.96064
## - Fold1: alpha=0.17301, lambda=3.96064
## + Fold1: alpha=0.89094, lambda=0.07584
## - Fold1: alpha=0.89094, lambda=0.07584
## + Fold1: alpha=0.03789, lambda=0.16332
## - Fold1: alpha=0.03789, lambda=0.16332
## + Fold1: alpha=0.26030, lambda=0.06467
## - Fold1: alpha=0.26030, lambda=0.06467
## + Fold2: alpha=0.26189, lambda=0.10504
## - Fold2: alpha=0.26189, lambda=0.10504
## + Fold2: alpha=0.17301, lambda=3.96064
## - Fold2: alpha=0.17301, lambda=3.96064
## + Fold2: alpha=0.89094, lambda=0.07584
## - Fold2: alpha=0.89094, lambda=0.07584
## + Fold2: alpha=0.03789, lambda=0.16332
## - Fold2: alpha=0.03789, lambda=0.16332
```

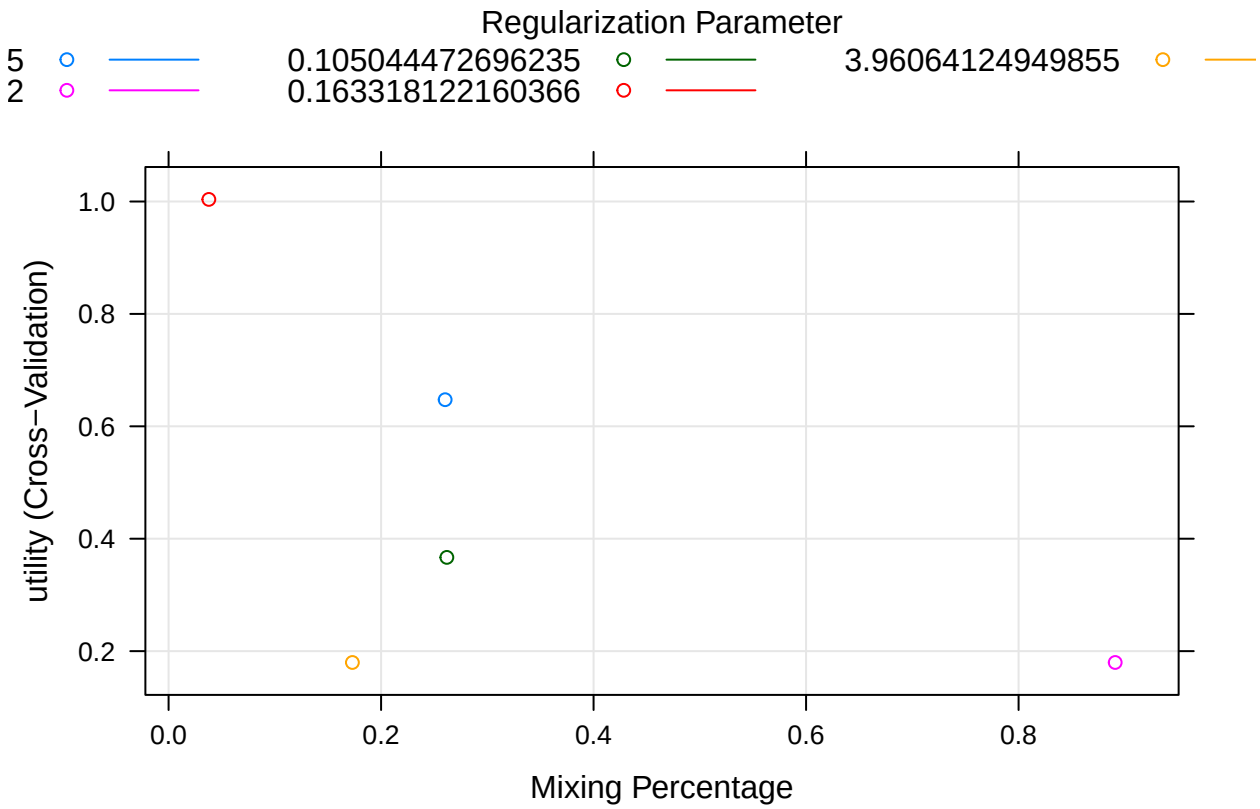
```
## + Fold2: alpha=0.26030, lambda=0.06467
## - Fold2: alpha=0.26030, lambda=0.06467
## + Fold3: alpha=0.26189, lambda=0.10504
## - Fold3: alpha=0.26189, lambda=0.10504
## + Fold3: alpha=0.17301, lambda=3.96064
## - Fold3: alpha=0.17301, lambda=3.96064
## + Fold3: alpha=0.89094, lambda=0.07584
## - Fold3: alpha=0.89094, lambda=0.07584
## + Fold3: alpha=0.03789, lambda=0.16332
## - Fold3: alpha=0.03789, lambda=0.16332
## + Fold3: alpha=0.26030, lambda=0.06467
## - Fold3: alpha=0.26030, lambda=0.06467
## + Fold4: alpha=0.26189, lambda=0.10504
## - Fold4: alpha=0.26189, lambda=0.10504
## + Fold4: alpha=0.17301, lambda=3.96064
## - Fold4: alpha=0.17301, lambda=3.96064
## + Fold4: alpha=0.89094, lambda=0.07584
## - Fold4: alpha=0.89094, lambda=0.07584
## + Fold4: alpha=0.03789, lambda=0.16332
## - Fold4: alpha=0.03789, lambda=0.16332
## + Fold4: alpha=0.26030, lambda=0.06467
## - Fold4: alpha=0.26030, lambda=0.06467
## + Fold5: alpha=0.26189, lambda=0.10504
## - Fold5: alpha=0.26189, lambda=0.10504
## + Fold5: alpha=0.17301, lambda=3.96064
## - Fold5: alpha=0.17301, lambda=3.96064
## + Fold5: alpha=0.89094, lambda=0.07584
## - Fold5: alpha=0.89094, lambda=0.07584
## + Fold5: alpha=0.03789, lambda=0.16332
## - Fold5: alpha=0.03789, lambda=0.16332
## + Fold5: alpha=0.26030, lambda=0.06467
## - Fold5: alpha=0.26030, lambda=0.06467
## Aggregating results
## Selecting tuning parameters
## Fitting alpha = 0.0379, lambda = 0.163 on full training set
```

```
print(df.glm)
```

```
## glmnet
##
## 42130 samples
##    55 predictor
##     2 classes: 'No', 'Yes'
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 33704, 33705, 33703, 33704, 33704
## Resampling results across tuning parameters:
##
##   alpha       lambda      utility    prob_thr
##   0.03788688  0.16331812  1.0037390  0.3
##   0.17300505  3.96064125  0.1798124  0.1
##   0.26029747  0.06467323  0.6473321  0.3
##   0.26189424  0.10504447  0.3667833  0.3
##   0.89094150  0.07583706  0.1798124  0.1
```

```
##
## utility was used to select the optimal model using the largest value.
## The final values used for the model were alpha = 0.03788688 and lambda
##  = 0.1633181.
```

```
plot(df.glm)
```




```
df.glm.model <- df.glm$finalModel

df.glm.model.coef <- predict(df.glm.model,
                             s = df.glm.model$lambdaOpt,
                             type = 'coefficients')

print(df.glm.model.coef)
```

```
## 84 x 1 sparse Matrix of class "dgCMatrix"
##                                       1
## (Intercept)                -8.527755e-01
## MonthlyRevenue              .
## MonthlyMinutes             -3.268115e-05
## TotalRecurringCharge       -1.335488e-03
## DirectorAssistedCalls       .
## OverageMinutes              2.390131e-04
## RoamingCalls                3.325798e-04
## PercChangeMinutes          -1.120819e-04
## PercChangeRevenues          2.058578e-04
## DroppedCalls                .
## BlockedCalls                .
## UnansweredCalls             .
```

```
## CustomerCareCalls          -2.203946e-03
## ThreewayCalls              -2.459976e-03
## ReceivedCalls               .
## OutboundCalls               .
## InboundCalls               -2.648910e-04
## PeakCallsInOut             -5.336858e-05
## OffPeakCallsInOut          -3.054776e-05
## DroppedBlockedCalls         .
## CallForwardingCalls         .
## CallWaitingCalls            .
## MonthsInService             .
## UniqueSubs                  1.559815e-02
## ActiveSubs                  .
## Handsets                    .
## HandsetModels              -1.528450e-02
## CurrentEquipmentDays        3.776994e-04
## AgeHH1                     -1.287522e-03
## AgeHH2                     -1.138788e-04
## ChildrenInHHYes             8.992750e-03
## HandsetRefurbishedYes       1.011746e-01
## HandsetWebCapableYes       -1.106534e-01
## TruckOwnerYes               .
## RVOwnerYes                  .
## HomeownershipUnknown        .
## BuysViaMailOrderYes        -1.241509e-02
## RespondsToMailOffersYes    -2.458097e-02
## OptOutMailingsYes           .
## NonUSTravelYes              .
## OwnsComputerYes             .
## HasCreditCardYes            .
## RetentionCalls              1.660845e-01
## RetentionOffersAccepted     .
## NewCellphoneUserYes         .
## NotNewCellphoneUserYes      .
## ReferralsMadeBySubscriber   .
## IncomeGroup                 .
## OwnsMotorcycleYes           .
## AdjustmentsToCreditRating  -8.965122e-03
## HandsetPrice100             .
## HandsetPrice130             .
## HandsetPrice150             .
## HandsetPrice180             .
## HandsetPrice200             .
## HandsetPrice240             .
## HandsetPrice250             .
## HandsetPrice30             -5.287034e-04
## HandsetPrice300             .
## HandsetPrice40              .
## HandsetPrice400             .
## HandsetPrice500             .
## HandsetPrice60              .
## HandsetPrice80              .
## HandsetPriceUnknown         .
## MadeCallToRetentionTeamYes  2.168516e-01
```

```
## CreditRating2-High          2.650604e-02
## CreditRating3-Good          1.303344e-02
## CreditRating4-Medium       -1.928061e-02
## CreditRating5-Low          -1.388897e-01
## CreditRating6-VeryLow       .
## CreditRating7-Lowest        .
## PrizmCodeRural              1.850308e-03
## PrizmCodeSuburban           .
## PrizmCodeTown               .
## OccupationCrafts            .
## OccupationHomemaker         .
## OccupationOther             .
## OccupationProfessional      .
## OccupationRetired           .
## OccupationSelf              .
## OccupationStudent           .
## MaritalStatusUnknown        2.476788e-02
## MaritalStatusYes            .
```

```
df.glm.results <- fn_results(df.glm)

print('Umbral')
```

```
## [1] "Umbral"
```

```
print(df.glm.results$prob_thr)
```

```
## [1] 0.3
```

```
print('Utilidad en train')
```

```
## [1] "Utilidad en train"
```

```
print(df.glm.results$utility)
```

```
## [1] 1.003739
```

```
print('Utilidad en dev')
```

```
## [1] "Utilidad en dev"
```

```
df.glm.dev.prob <- predict(df.glm, newdata = df.part$dev, type = 'prob')
df.glm.dev.pred <- fn_pred(df.glm.dev.prob, thr = df.glm.results$prob_thr)

df.glm.dev.utility <- fn_utility(df.glm.dev.pred, df.part$dev$Churn)

print(df.glm.dev.utility)
```

```
## [1] 0.8547319
```

```
print('Matriz de confusion en dev')
```

```
## [1] "Matriz de confusion en dev"
```

```
df.glm.dev.cm <- conf_matrix(df.glm.dev.pred, df.part$dev$Churn)

print(df.glm.dev.cm)
```

```
##           Reference
## Prediction   No  Yes
```

```
##        No  2456  843
##        Yes  880  502
```

```
print('** Generacion de la prediccion sobre test sample')
```

```
## [1] "** Generacion de la prediccion sobre test sample"
```

```
test_sample <- read.csv('../data/test_sample.csv')
rownames(test_sample) <- test_sample$CustomerID
test_sample$CustomerID <- NULL
test_sample$ServiceArea <- NULL

file_id <- paste0(c(script.name, script.date), collapse = ' ')

gen_prediction(df.glm, test_sample, prob_thr = df.glm.results$prob_thr, id = file_id)

print('Done')
```

```
## [1] "Done"
```

```
script.done <- Sys.time()

print(script.done - script.start)
```

```
## Time difference of 1.779007 mins
```

```
# [1] "Utilidad real de la prediccion"
# [1] 0.9698145
```