Caret

jingwen

12/9/2020

Introduction

Dans cet article, nous utiliserons le package **caret** pour expliquer et pratiquer *le prétraitement et la segmentation des données*.

Afin de mieux comprendre la fonction, cet article prendra le données de carte de crédit (https://www.kaggle.com/sakshigoyal7/credit-card-customers) comme exemple pour la démonstration.

Importez le package caret

```
library(caret)
```

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

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

Si nous voulons effacer le répertoire de travail avant cela, utilisez:

```
rm(list = ls())
```

Prétraitement des données

Traitement des valeurs manquantes

1. supprimer:

```
##Importer des données
dat1=read.table(file ="/Users/jingwensu/Desktop/BankChurners1.csv",encoding = "uft-8",sep=",",heade
r = T,row.names = 1)
print(ncol(dat1))
```

```
## [1] 20
```

```
##Afficher la taille de l'ensemble de données
nrow(dat1)
```

```
## [1] 10127
```

```
##Supprimer les valeurs manquantes
dat=na.omit(dat1)
nrow(dat)
```

```
## [1] 10127
```

2.La fonction **preProcess ()** Il est intégrée à l'ensemble d'apprentissage. Il inclut:

1. Méthode médiane: method = "medianImpute"

```
#imputation_k = preProcess (datTrain, method = "medianImpute")
#datTrain1 = prédire (imputation_k, datTrain)
#datTest1 = prédire (imputation_k, datTest)
```

2. kNearest Neighbor Method: method = “knnImpute”; Cette méthode peut montrer que le résultat est composé de plusieurs décimales. Veuillez normaliser si nécessaire.
3. Traitement de la variable de variance 0 Lorsqu’un certain groupe de valeurs de données sont identiques ou sont toutes égales à 0, cela n’a aucun sens pour notre analyse, nous le supprimons donc.

```
#dim (datTrain)
#(nzv = nearZeroVar (datTrain))
#datTrain = datTrain [, - nzv]
```

Convertir le type de données

```
dat$Gender =factor(dat$"Gender", levels = c("F","M"),labels =  c("Femme","homme"))
head(dat)
```

```
##              Attrition_Flag Customer_Age Gender Dependent_count Education_Level
## 768805383 Existing Customer           45  homme               3     High School
## 818770008 Existing Customer           49  Femme               5        Graduate
## 713982108 Existing Customer           51  homme               3        Graduate
## 769911858 Existing Customer           40  Femme               4     High School
## 709106358 Existing Customer           40  homme               3      Uneducated
## 713061558 Existing Customer           44  homme               2        Graduate
##           Marital_Status Income_Category Card_Category Months_on_book
## 768805383        Married     $60K - $80K          Blue             39
## 818770008         Single  Less than $40K          Blue             44
## 713982108        Married    $80K - $120K          Blue             36
## 769911858        Unknown  Less than $40K          Blue             34
## 709106358        Married     $60K - $80K          Blue             21
## 713061558        Married     $40K - $60K          Blue             36
##           Total_Relationship_Count Months_Inactive_12_mon Contacts_Count_12_mon
## 768805383                        5                      1                     3
## 818770008                        6                      1                     2
## 713982108                        4                      1                     0
## 769911858                        3                      4                     1
## 709106358                        5                      1                     0
## 713061558                        3                      1                     2
##           Credit_Limit Total_Revolving_Bal Avg_Open_To_Buy Total_Amt_Chng_Q4_Q1
## 768805383        12691                 777           11914                1.335
## 818770008         8256                 864            7392                1.541
## 713982108         3418                   0            3418                2.594
## 769911858         3313                2517             796                1.405
## 709106358         4716                   0            4716                2.175
## 713061558         4010                1247            2763                1.376
##           Total_Trans_Amt Total_Trans_Ct Total_Ct_Chng_Q4_Q1
## 768805383            1144             42                1.625
## 818770008            1291             33                3.714
## 713982108            1887             20                2.333
## 769911858            1171             20                2.333
## 709106358             816             28                2.500
## 713061558            1088             24                0.846
##           Avg_Utilization_Ratio
## 768805383                 0.061
## 818770008                 0.105
## 713982108                 0.000
## 769911858                 0.760
## 709106358                 0.000
## 713061558                 0.311
```

Créer une partition de données

1. Divisez 80% des données dans l'ensemble d'apprentissage et 20% comme ensemble de test.

```
trainIndex= createDataPartition(dat$Attrition_Flag, p= .8, list = FALSE, times = 1)
##Ensemble d'entraînement
datTrain= dat[trainIndex, ]
##Ensemble d'essai
datTest= dat[-trainIndex, ]
## La proportion de la variable dépendante à chaque niveau sur l'ensemble complet
table(dat$Attrition_Flag)/nrow(dat)
```

```
##
## Attrited Customer Existing Customer
##         0.1606596         0.8393404
```

```
## La proportion de chaque niveau de la variable dépendante sur l'ensemble d'apprentissage
table(datTrain$Attrition_Flag)/nrow(datTrain)
```

```
##
## Attrited Customer Existing Customer
##         0.1607011         0.8392989
```

```
## La proportion de la variable dépendante à chaque niveau de l'ensemble de test
table(datTest$Attrition_Flag)/nrow(datTest)
```

```
##
## Attrited Customer Existing Customer
##         0.1604938         0.8395062
```

2. Validation croisée

```
set.seed(1234)
index= createFolds(dat$Gender,k=3,list = FALSE,returnTrain = TRUE)

testIndex=which(index==1)
datTraincv=dat[-testIndex,]##ensemble d'entraînement
datTestcv=dat[testIndex,]##ensemble d'entraînement
```

3. À propos de la segmentation des séries chronologiques

```
# data3= createTimeSlices(1:nrow(growdata),initialWindow=5,horizon=2,fixedWindow=TRUE)
```

Où 5 correspond à la fenêtre initiale, 2 signifie que l'ensemble de test correspond aux deux derniers chiffres de l'ensemble d'apprentissage, et la fenêtre fixe signifie que la largeur de l'ensemble d'apprentissage est la même. Si vous souhaitez partir du premier échantillon à chaque fois, définissez-le sur FALSE et par défaut sur TRUE.

Traitement standardisé

Standardisez l'ensemble de test avec la moyenne et la variance de l'ensemble d'apprentissage.

```
preProcValues= preProcess(datTrain,method = c("center","scale"))
trainTransformed= predict(preProcValues, datTrain)
testTransformed= predict(preProcValues,datTest)
```

Sélection de variables

Méthode d'emballage rfe

```
#### Pour choisir le nombre de variables
subsets= c(2,5,10,15,20,25,30,35,40)
#### Définir les paramètres de contrôle, les fonctions consistent à déterminer quel type de modèle
 est utilisé pour trier les variables indépendantes, voici une forêt aléatoire: selon la fonction o
bjectif ou le score d'effet de prédiction, sélectionnez plusieurs fonctionnalités à chaque fois; la
méthode consiste à déterminer la méthode d'échantillonnage à utiliser, Le cv utilisé ici est la val
idation croisée
ctrl = rfeControl(functions = rfFuncs,method = "cv")
x=trainTransformed[,-which(colnames(trainTransformed) %in%"Gender")]
y=trainTransformed[,"Gender"]
Profile=rfe(x,y, sizes=c(1:5),rfeControl = ctrl)
Profile$optVariables
```

```
##  [1] "Income_Category"          "Credit_Limit"
##  [3] "Avg_Open_To_Buy"          "Total_Trans_Amt"
##  [5] "Customer_Age"             "Avg_Utilization_Ratio"
##  [7] "Total_Trans_Ct"           "Total_Revolving_Bal"
##  [9] "Months_on_book"           "Total_Ct_Chng_Q4_Q1"
## [11] "Card_Category"            "Attrition_Flag"
## [13] "Total_Amt_Chng_Q4_Q1"     "Contacts_Count_12_mon"
## [15] "Dependent_count"          "Total_Relationship_Count"
## [17] "Marital_Status"           "Education_Level"
## [19] "Months_Inactive_12_mon"
```

```
## Formation et réglage du modèle
dat.train=trainTransformed[,c(Profile$optVariables,"Gender")]
dat.test=testTransformed[,c(Profile$optVariables,"Gender")]
## Forêt aléatoire
set.seed(1234)
gbmFitl=train(Gender ~., data = dat.train,method="rf")
# Utilisé pour former le modèle
importance=varImp(gbmFitl, scale=FALSE)
# Obtenez l'importance de chaque variable
plot(importance,xlab = "importance")
```