travail CARET

vincent obertelli

12/6/2020

Nous allons désormais apprendre à utiliser le package **Caret**. Cette librairie permet de faire de l'analyse prédictive.

Tout comme les autres librairies abordées plus tôt, il faut d'abord installer le package pour pouvoir l'utiliser :

install.packages("caret")

Pour mieux comprendre cette librairie, nous allons utiliser une exemple :

```
library(caret)
```

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

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

```
# Import  du dataset
orange <- read.csv('https://raw.githubusercontent.com/selva86/datasets/master/orange_juice_withmissing.

# Structure de la dataframe
str(orange)
```

```
## 'data.frame':    1070 obs. of  18 variables:
##  $ Purchase      : chr  "CH" "CH" "CH" "MM" ...
##  $ WeekofPurchase: int  237 239 245 227 228 230 232 234 235 238 ...
##  $ StoreID       : int  1 1 1 1 7 7 7 7 7 7 ...
##  $ PriceCH       : num  1.75 1.75 1.86 1.69 1.69 1.69 1.69 1.75 1.75 1.75 ...
##  $ PriceMM       : num  1.99 1.99 2.09 1.69 1.69 1.99 1.99 1.99 1.99 1.99 ...
##  $ DiscCH        : num  0 0 0.17 0 0 0 0 0 0 0 ...
##  $ DiscMM        : num  0 0.3 0 0 0 0 0.4 0.4 0.4 0.4 ...
##  $ SpecialCH     : int  0 0 0 0 0 0 1 1 0 0 ...
##  $ SpecialMM     : int  0 1 0 0 0 1 1 0 0 0 ...
##  $ LoyalCH       : num  0.5 0.6 0.68 0.4 0.957 ...
##  $ SalePriceMM   : num  1.99 1.69 2.09 1.69 1.69 1.99 1.59 1.59 1.59 1.59 ...
##  $ SalePriceCH   : num  1.75 1.75 1.69 1.69 1.69 1.69 1.69 1.75 1.75 1.75 ...
##  $ PriceDiff     : num  0.24 -0.06 0.4 0 0 0.3 -0.1 -0.16 -0.16 -0.16 ...
##  $ Store7        : chr  "No" "No" "No" "No" ...
##  $ PctDiscMM     : num  0 0.151 0 0 0 ...
##  $ PctDiscCH     : num  0 0 0.0914 0 0 ...
##  $ ListPriceDiff : num  0.24 0.24 0.23 0 0 0.3 0.3 0.24 0.24 0.24 ...
##  $ STORE         : int  1 1 1 1 0 0 0 0 0 0 ...
```

```r
# afficher du début à la 6ème ligne et les 10 columnes
head(orange[, 1:10])
```

```
##   Purchase WeekofPurchase StoreID PriceCH PriceMM DiscCH DiscMM SpecialCH
## 1       CH            237       1    1.75    1.99   0.00    0.0         0
## 2       CH            239       1    1.75    1.99   0.00    0.3         0
## 3       CH            245       1    1.86    2.09   0.17    0.0         0
## 4       MM            227       1    1.69    1.69   0.00    0.0         0
## 5       CH            228       7    1.69    1.69   0.00    0.0         0
## 6       CH            230       7    1.69    1.99   0.00    0.0         0
##   SpecialMM  LoyalCH
## 1         0 0.500000
## 2         1 0.600000
## 3         0 0.680000
## 4         0 0.400000
## 5         0 0.956535
## 6         1 0.965228
```

Nous allons désormais commencer l'algorythme d'analyse prédictif. On va prendre un échantillon de données, puis le diviser en deux parties : entrainement (environ 80% des données) et test (20% des données).

```r
library(caret)
# Création du "seed", référence de l'algorythme aléatoire
set.seed(100)

# Etape 1: préparation des données d'entrainement
trainRowNumbers <- createDataPartition(orange$Purchase, p=0.8, list=FALSE)

# Etape 2: Création du dataset d'entrainement
trainData <- orange[trainRowNumbers,]

# Etape 3: Création du dataset test
testData <- orange[-trainRowNumbers,]

# Stocker X et Y pour les utiliser plus tard.
x = trainData[, 2:18]
y = trainData$Purchase
```

Maintenant que nous avons créer nos bases on va tester si l'algorythme est bon pour prévoir. Nous allons entrainer notre programme sur les 80% et tenter de prévoir sur les 20% restants. Pour cela nous allons créer une liste de prédiction qui va essayer de deviner correctement les 20% restant. Une fois que cela est fait on peut calculer l'accuracy (ou précision).Plus l'accuracy est élévée, plus le système de prédiction est élevé.

Pour cette partie, les codes que j'ai essayé non pas fonctionné, je vais donc chercher comment réussir cette partie pendant la semaine et pendant les cours de R.