Apprentissage de données biologiques 2020-2021
Contrôle des connaissances (durée : 1h)

22 octobre 2020

Nom Prénom :

Tous les documents sont autorisés. La calculatrice est autorisée. L'usage du téléphone portable est interdit, même pour sa fonction calculatrice.

Le problème qui suit est inspiré du challenge proposé aux participants d'un congrès (Chimiométrie 2010, 2-3 décembre 2010, Paris, http://www.chimiometrie.fr/challenge2010.html).
La bière Trappiste est une bière de haute fermentation brassée sous le contrôle des moines Trappistes selon des critères bien définis par l'association privée *Association Internationale Trappiste*. L'objectif de cet exercice est d'authentifier une bière Trappiste à partir de données spectrométriques dans la gamme du moyen infra-rouge (pour chaque spectre, on dispose de mesures en 1762 nombres d'onde).

On importe les données dans la session de travail à l'aide des commandes suivantes :

```
trappmir = read.table("trappmir.txt",stringsAsFactors=TRUE)[,-1763]
str(trappmir[,1754:1763])
```

```
'data.frame':   200 obs. of  10 variables:
 $ dp1754   : num  -0.000114 0.000981 0.001448 0.002495 0.000941 ...
 $ dp1755   : num  0.000366 0.001853 0.001115 0.001053 0.001063 ...
 $ dp1756   : num  0.000369 0.00244 -0.000319 -0.00115 0.000339 ...
 $ dp1757   : num  -0.000301 0.001748 -0.000986 -0.001611 -0.000202 ...
 $ dp1758   : num  -0.000525 0.000698 -0.00029 -0.000676 0.000386 ...
 $ dp1759   : num  0.000253 0.001001 0.000804 -0.000272 0.001777 ...
 $ dp1760   : num  0.000324 0.001597 0.001627 -0.000491 0.002854 ...
 $ dp1761   : num  -0.00135 0.000935 0.001838 -0.000432 0.003087 ...
 $ dp1762   : num  -0.003261 -0.000544 0.0011 -0.000322 0.002757 ...
 $ Trappiste: Factor w/ 2 levels "Non Trappiste",..: 2 2 2 2 2 2 1 1 1 1 ...
```

A partir de ces données, l'objectif est de construire un outil prédisant au mieux le statut *Trappiste* ou non d'une bière.

Question 1

Quelles sont les variables réponse et explicatives dans cette problématique ? Donnez la nature (quantitative ou catégorielle) de ces variables et, si catégorielle, leurs modalités ?

Réponse

Question 2

Ecrire le modèle introduisant un score linéaire pour la probabilité qu'une bière soit de statut Trappiste à partir de son spectre moyen infra-rouge. Combien de paramètres à ce modèle ?

Réponse

On propose d'estimer ce score linéaire par la méthode de l'analyse discriminante linéaire. C'est l'objet des commandes suivantes qui utilise la fonction `lda` du package `MASS` et affiche un extrait des coefficients estimés :

```
require(MASS)
trappmir.lda = lda(Trappiste~.,data=trappmir)
head(coef(trappmir.lda)[,1])
```

```
       dp1         dp2         dp3         dp4         dp5         dp6
-720.37230 -1136.88788   162.28628   718.88440   198.95207    -2.89738
```

Question 3

En quoi peut-on affirmer que les coefficients dont un extrait est donné ci-dessus définissent un score linéaire optimal pour discriminer les bières Trappistes des autres bières ?

Réponse

On peut maintenant mettre en oeuvre une règle de prédiction du statut Trappiste ou non d'une bière à partir du score linéaire discriminant. Les commandes suivantes affichent les valeurs de ce score pour les six premières bières de l'échantillon :

```
trappmir.ld = predict(trappmir.lda)$x[,1]
head(trappmir.ld)
```

```
      V2       V3       V4       V5       V6       V7
0.122803 4.455358 2.212047 1.942473 1.726439 2.113459
```

Question 4

Quel est le statut, Trappiste ou non, prédit par la règle de Bayes pour chacune des six premières bières de l'échantillon ?

Réponse

Les commandes suivantes mettent en œuvre cette règle de Bayes pour prédire le statut Trappiste ou non d'une bière. Elles calculent aussi deux critères pour évaluer les performances de prédiction de cette règle :

```
trappmir.predictions = predict(trappmir.lda)$class
print(paste("Critere 1 = ",mean(trappmir$Trappiste==trappmir.predictions),sep=""))
```

```
[1] "Critere 1 = 0.97"
```

```
trappmir.cvlda = lda(Trappiste~.,data=trappmir,CV=TRUE)
trappmir.cvpredictions = trappmir.cvlda$class
print(paste("Critere 2 = ",mean(trappmir$Trappiste==trappmir.cvpredictions),sep=""))
```

```
[1] "Critere 2 = 0.52"
```

Question 5

Donnez une interprétation de la valeur de chacun des critères ci-dessus et expliquez pourquoi ces deux valeurs sont si différentes.

Réponse

Lorsque le nombre de variables explicatives est plus petit que la taille de l'échantillon, on peut montrer que le score linéaire discriminant calculé ci-dessus par la fonction `lda` s'obtient de manière équivalente en estimant par la méthode des moindres carrés le modèle de régression linéaire multiple de Y sur toutes les variables explicative, où $Y = \pm 1$ est un recodage numérique de la variable réponse (ici, $Y = +1$ pour les bières de statut Trappiste et $Y = -1$ pour les autres).

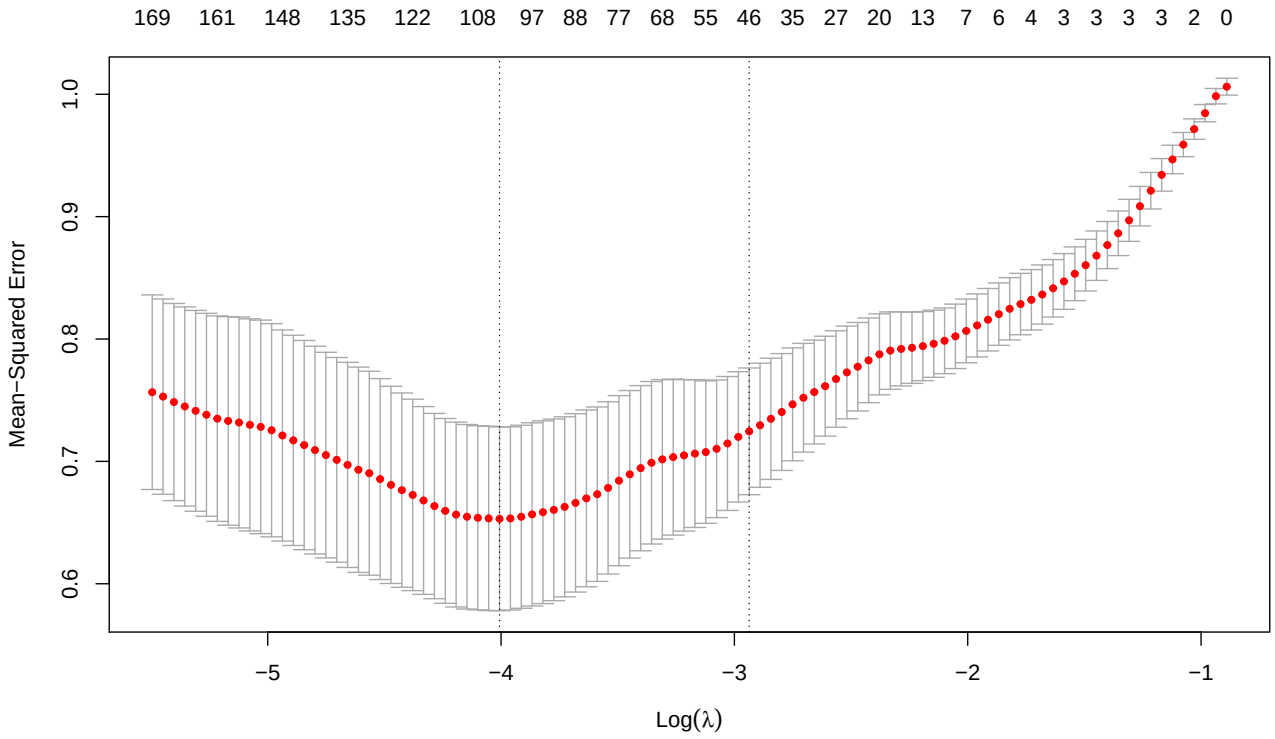
Question 6

En déduire l'expression d'un critère des moindres carrés pénalisé dont la minimisation conduit à une estimation parcimonieuse du score linéaire discriminant (parcimonieuse, certains coefficients pouvant être estimés à zéro si le paramètre λ de pénalité est suffisamment grand).

Réponse

Les commandes ci-après utilisent le package `glmnet` pour minimiser le critère des moindres carrés pénalisé de la question 6 et construire un graphique aidant au choix de la valeur optimale de λ :

```
require(glmnet)
x = as.matrix(trappmir[,-1763])
y = ifelse(trappmir$Trappiste=="Trappiste",1,-1)
trappmir.cvlasso = cv.glmnet(x=x,y=y,type.measure="mse")
plot(trappmir.cvlasso)
```



Question 7

D'après le graphique ci-dessus, si on choisit la valeur de λ minimisant la variance de l'erreur de prédiction de Y, combien de coefficients estimés du score linéaire sont non-nuls ?

Réponse

Les deux séries de commandes suivantes implémentent chacune une règle de prédiction fondée uniquement sur les variables sélectionnées précédemment :

```
# 1ère méthode
which.optimal = which.min(trappmir.cvlasso$cvm)
trappmir.lasso = glmnet(x=x,y=y,lambda=trappmir.cvlasso$lambda)
beta = coefficients(trappmir.lasso)[-1,which.optimal]
select = which(abs(beta)>1e-08)
trappmir.lda = lda(Trappiste~.,data=trappmir[,c(select,1763)])
trappmir.predictions1 = predict(trappmir.lda)$class
print(paste("Critere 1 = ",mean(trappmir$Trappiste==trappmir.predictions1),sep=""))
```

```
[1] "Critere 1 = 0.995"
```

```
trappmir.cvlda = lda(Trappiste~.,data=trappmir[,c(select,1763)],CV=TRUE)
trappmir.cvpredictions1 = trappmir.cvlda$class
print(paste("Critere 2 = ",mean(trappmir$Trappiste==trappmir.cvpredictions1),sep=""))
```

```
[1] "Critere 2 = 0.93"
```

```
# 2ème méthode
require(groupdata2)
predictions = predict(trappmir.lasso,newx=x)[,which.optimal]
trappmir.predictions2 = ifelse(sign(predictions)>0,"Trappiste","Non Trappiste")
print(paste("Critere 1 = ",mean(trappmir$Trappiste==trappmir.predictions2),sep=""))
```

```
[1] "Critere 1 = 0.985"
```

```
segs = fold(trappmir,k=10,cat_col="Trappiste")$".folds"
cvpredictions = rep(0,nrow=nrow(trappmir))
for (k in 1:10) {
   dta.lasso = glmnet(x[segs!=k,],y[segs!=k],lambda=trappmir.cvlasso$lambda)
   cvpredictions[segs==k] = predict(dta.lasso,newx=x[segs==k,])[,which.optimal]
}
trappmir.cvpredictions2 = ifelse(sign(cvpredictions)>0,"Trappiste","Non Trappiste")
print(paste("Critere 2 = ",mean(trappmir$Trappiste==trappmir.cvpredictions2),sep=""))
```

```
[1] "Critere 2 = 0.755"
```

Question 8

Décrivez les deux méthodes de prédiction ci-dessus en indiquant pour chacune le nombre de variables explicatives retenues dans le score linéaire, le critère optimisé pour estimer le score linéaire discriminant et la règle de prédiction mise en œuvre pour déduire le statut Trappiste ou non d'une bière à partir de la valeur de son score.

Réponse

Suite réponse

Question 9

Finalement, la sélection d'un sous-ensemble de variables explicatives a-t'elle amélioré les résultats commentés dans la question 5 ? (justifiez brièvement votre réponse)