POLYTECHNIQUE
MONTRÉAL

LE GÉNIE
EN PREMIÈRE CLASSE

Devoir N3

MTH6312 - METHODES STATISTIQUES D'APPRENTISSAGE

2 novembre 2018

Lyes Heythem Bettache -1923715

Devoir N°3

Lyes Heythem BETTACHE

2 novembre 2018

R Markdown

QUESTION N°1

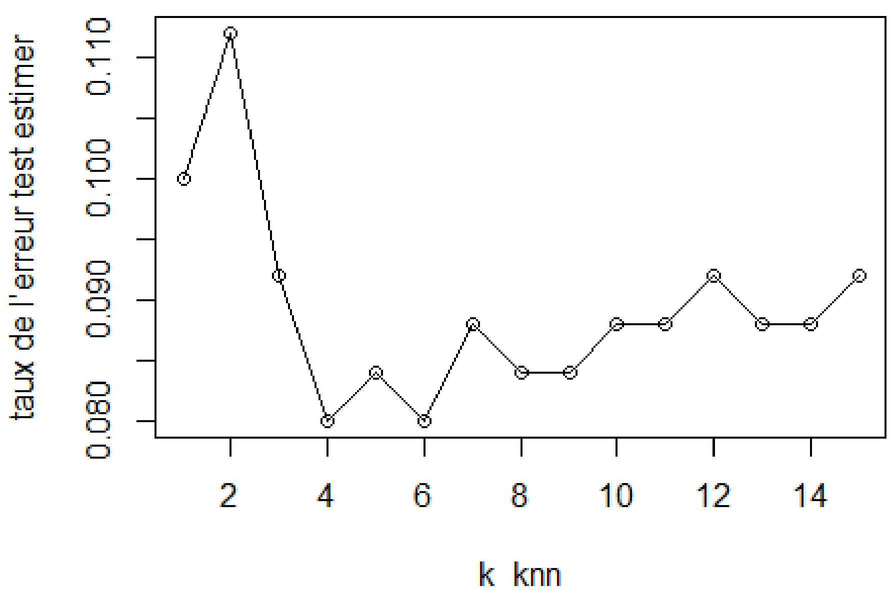
```
remove(list=ls())

# On importe les données
data=read.csv("C:/Users/defaultuser0.DESKTOPI1A8N1U.000/Desktop/Auto18/MTH631
2/d3/Equipement.csv")

# Convertir "default" à numeric
data$Y=as.numeric(data$Y == 'D')


data_xi = data[,c(1,2)]
data_y= data[,3]
```

a) KNN.

a-1) En utilisant la technique de validation croisée «LOOCV» sur les 250 observations, estimer le taux de l'erreur test pour différentes valeurs (une dizaine) du nombre de voisins, k.

```
# On utilise la librarie "class" pour KNN et "caret" pour CV

library(class)
library(caret)

## Loading required package: lattice

## Loading required package: ggplot2

set.seed(1923715)

folds = createFolds(data_y, k=250)
k_knn = 1:15
error_knn <- sapply(k_knn, function(k) {
          error_k <- sapply(seq_along(folds), function(i) {
             model_knn = knn(train=data_xi[ -folds[[i]], ], test=data_xi[ fol
ds[[i]], ], cl=data_y[ -folds[[i]] ], k = k)
```

```r
            mean(data_y[ folds[[i]] ] != model_knn)
})
mean(error_k)
})
knnRMS = data.frame(k_knn, error_knn)



results = matrix(c(k_knn , error_knn), nrow=2, byrow=TRUE)
rownames(results) = c("k_knn","error_knn")
cat("\n         \n")

results

##             [,1]  [,2]  [,3] [,4]  [,5] [,6]  [,7]  [,8]  [,9]  [,10]  [,11]
## k_knn        1.0 2.000 3.000 4.00 5.000 6.00 7.000 8.000 9.000 10.000 11.000
## error_knn    0.1 0.112 0.092 0.08 0.084 0.08 0.088 0.084 0.084  0.088  0.088
##

             [,12]  [,13]  [,14]  [,15]
## k_knn     12.000 13.000 14.000 15.000
## error_knn  0.092  0.088  0.088  0.092

plot(k_knn, error_knn, type="o",ylab="taux de l'erreur test estimer")
```



D'après la courbe du taux d'erreur test estimer en fonction k on a trouvé que k optimal égal 4

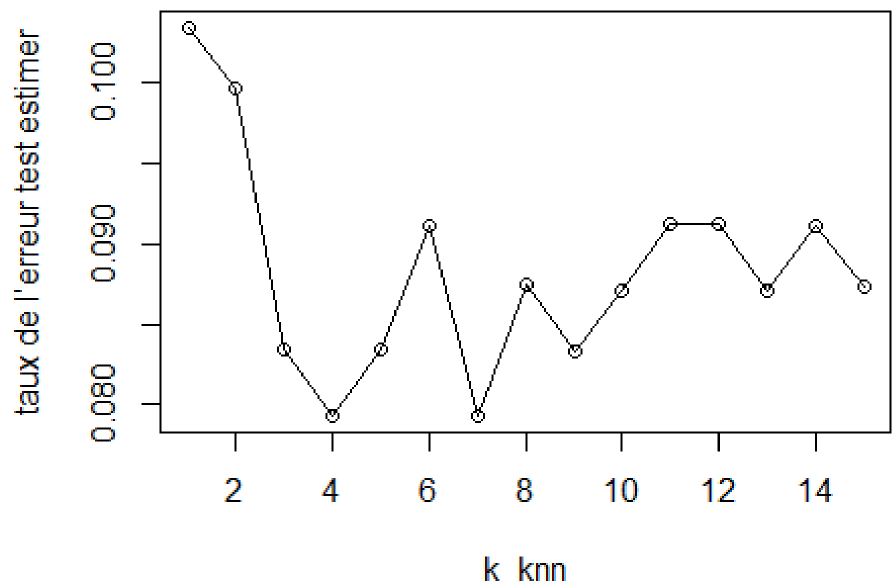
#k optimal égal 4 le taux d'erreur pour KNN : 0.08

a-2)-Reprendre la question ci-dessus en utilisant la technique de validation croisée «10-Fold CV» ou «5-Fold CV»

C'est la même chose si on choisir la technique de validation croisée «10-Fold CV» ou «5-Fold CV», dans notre cas on a choisi «10-Fold CV» pour augmenter la taille des données entrainements

«10-Fold CV»

```
set.seed(1923715)
folds = createFolds(data_y, k=10)
k_knn = 1:15
error_knn <- sapply(k_knn, function(k) {
          error_k <- sapply(seq_along(folds), function(i) {
             model_knn = knn(train=data_xi[ -folds[[i]], ], test=data_xi[ fol
ds[[i]], ], cl=data_y[ -folds[[i]] ], k = k)
            mean(data_y[ folds[[i]] ] != model_knn)
})
mean(error_k)
})
knnRMS = data.frame(k_knn, error_knn)



results = matrix(c(k_knn , error_knn), nrow=2, byrow=TRUE)
rownames(results) = c("k_knn","error_knn")
cat("\n         \n")

results

##                [,1]       [,2]       [,3]       [,4]       [,5]       [,6]
## k_knn     1.0000000 2.00000000 3.00000000 4.00000000 5.00000000 6.00000000
## error_knn 0.1034744 0.09962821 0.08344872 0.07928205 0.08344872 0.09114103
##

              [,7]       [,8]       [,9]       [,10]       [,11]
## k_knn     7.00000000 8.00000000 9.00000000 10.00000000 11.00000000
## error_knn 0.07929487 0.08746154 0.08329487  0.08712821  0.09129487
##

              [,12]       [,13]       [,14]       [,15]
## k_knn     12.00000000 13.00000000 14.00000000 15.00000000
## error_knn  0.09129487  0.08714103  0.09114103  0.08729487

plot(k_knn, error_knn, type="o",ylab="taux de l'erreur test estimer")
```



D’après la courbe du taux d’erreur test estimer en fonction k on a trouvé que k optimal égal 4

#k optimal égal 4 le taux d’erreur pour KNN : 0.07928205

a-3)-. Compte tenu des résultats ci-dessus, quelle valeur du nombre de voisins k devrait-on utiliser pour la classification des données du contexte par le KNN? Justifier brièvement.

D’après les 2 méthodes précédents on a trouvé que k optimal égal 4 c’est le résultat qui a donné le taux d’erreur le moins élevé

b) Régression logistique.

b-1). En utilisant la technique de validation croisée «LOOCV» sur les 250 observations, estimer le taux d'erreur test pour chacun des 2 modèles.

Modèle 1

```
library (boot)

glm_model1 = glm(Y ~ X1 + X2, data=data, family=binomial)
#coef(glm_model1)
cv.err_model1 = cv.glm(data,glm_model1)
# Représentation de l'erreur
cat(sprintf("le taux d'erreur test pour model1: %s", cv.err_model1$delta[1]))

## le taux d'erreur test pour model1: 0.0528246135989653
```

Modèle 2

```
library (boot)

glm_model2 = glm(Y ~ X1 + X2 + I(X1^2) + I(X2^2), data=data, family=binomial)
#coef(glm_model2)
cv.err_model2 = cv.glm(data,glm_model2)
# Représentation de l'erreur
cat(sprintf("le taux d'erreur test pour model2: %s", cv.err_model2$delta[1]))

## le taux d'erreur test pour model2: 0.0556619688814797
```

b-2. Reprendre la question ci-dessus en utilisant une des techniques de validation croisée «5-Fold CV» ou «10-Fold CV» de votre choix. Justifier brièvement votre choix.

C'est la même chose si on choisir la technique de validation croisée «10-Fold CV» ou «5-Fold CV», dans notre cas on a choisi «10-Fold CV» pour augmenter la taille des données entrainements

«10-Fold CV»

Modele1

```
library (boot)
set.seed(1923715)

glm.10f_model1 = glm(Y ~ X1 + X2, data=data, family=binomial)
#coef(glm.10f_model1)
cv10f.err_model1 = cv.glm(data,glm.10f_model1, K=10)
# Représentation de l'erreur
```

```r
cat(sprintf("le taux d'erreur test pour modele1: %s",cv10f.err_model1$delta[1
]))
```

```
## le taux d'erreur test pour modele1: 0.0529639985972981
```

Modele2

```r
library (boot)
set.seed(1923715)

glm.10f_model2 = glm(Y ~ X1 + X2 + I(X1^2) + I(X2^2), data=data, family=binom
ial)
#coef(glm.10f_model2)
cv10f.err_model2 = cv.glm(data,glm.10f_model2, K=10)
# Représentation de l'erreur
cat(sprintf("le taux d'erreur test pour modele2: %s",cv10f.err_model2$delta[1
]))
```

```
## le taux d'erreur test pour modele2: 0.0556930958137126
```

b-3). Compte tenu des résultats ci-dessus, lequel des 2 modèles de régression logistique devraiton utiliser pour la classification des données du contexte ? Justifier brièvement.

On remarque que le taux d'erreur test estimer pour le modelé 1 est meilleur par rapport le modèle 2 le taux d'erreur test pour model1: 0.052 le taux d'erreur test pour model2: 0.055 et aussi on remarque que le modèle 1 est plus simple donc prendre le premier modèle (modèle 1)

c) Analyse discriminante.

c-1). En utilisant la technique de validation croisée «LOOCV» sur les 250 observations, estimer le taux d'erreur test de l'analyse discriminante linéaire (LDA) et celui de l'analyse discriminante quadratique (QDA).

LDA

```r
library (MASS)
library (boot)
set.seed(1923715)
# LDA à déjà un parametre pour faire la LOOCV (CV=TRUE)

model_lda = lda(Y ~ X1 + X2, data=data, CV=TRUE)

# On utilise $class parce que c'est la prédiction qu'on va comparer avec les
données
```

```
error_lda = mean(model_lda$class != data$Y )
# Représentation de l'erreur
cat(sprintf("le taux d’erreur: %s\n",error_lda))

## le taux d’erreur: 0.08
```

QDA

```
library (MASS)
library (boot)
set.seed(1923715)

# QDA à déjà un parametre pour faire la LOOCV (CV=TRUE)
model_qda = qda(Y ~ X1 + X2, data=data, CV=TRUE)

# On utilise $class parce que c'est la prédiction qu'on va comparer avec les
données
error_qda = mean(model_qda$class != data$Y )
# Représentation de l'erreur
cat(sprintf("Taux d'erreur: %s\n",error_qda))

## Taux d'erreur: 0.08
```

c-2). Reprendre la question ci-dessus en utilisant une des techniques de validation croisée «5-Fold CV» ou «10-Fold CV» de votre choix. Justifier brièvement votre choix.

Dans cette partie lorsque j’ai utilisé la technique de validation croisée «10-Fold CV» j’ai trouvé le même Taux d’erreur estimer pour les deux méthode d’analyse discriminante LDA et QDA, donc j’ai choisi la technique de validation croisée «5-Fold CV» pour simplifier la comparaison

5-fold

LDA

```
library (MASS)
library(caret)
library (boot)
set.seed(1923715)


folds = createFolds(data_y, k=5)
error_lda <- sapply(seq_along(folds), function(i) {
model_lda = lda(Y ~ X1 + X2, data=data[-folds[[i]], ])
pred_lda = predict(model_lda, data[folds[[i]], ])
mean(pred_lda$class != data$Y[folds[[i]]] )
})
```

```
#error_lda
cat(sprintf("Taux d'erreur: %s\n",mean(error_lda)))

## Taux d'erreur: 0.076
```

QDA

```
library (MASS)
library(caret)
library (boot)
set.seed(1923715)


folds = createFolds(data_y, k=5)
error_Qda <- sapply(seq_along(folds), function(i) {
model_Qda = qda(Y ~ X1 + X2, data=data[-folds[[i]], ])
pred_Qda = predict(model_Qda, data[folds[[i]], ])
mean(pred_Qda$class != data$Y[folds[[i]]] )
})
#error_Qda
cat(sprintf("Taux d'erreur: %s\n",mean(error_Qda)))

## Taux d'erreur: 0.084
```

C-3). Compte tenu des résultats ci-dessus, laquelle des deux analyses discriminantes devrait-on utiliser pour la classification des données du contexte ? Justifier brièvement.

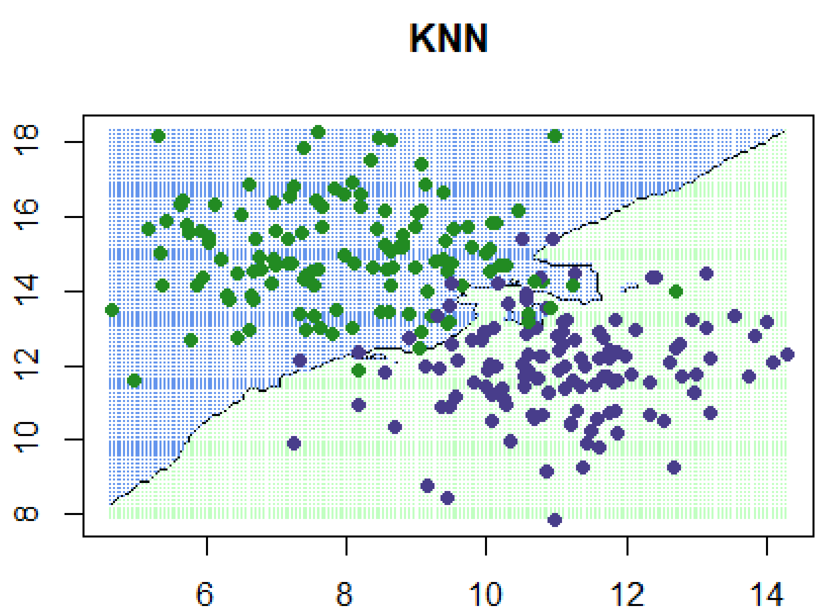
On remarque que le taux d'erreur test estimer pour le modèle LDA est meilleur par rapport le modèle QDA lorsque on a utilisé la technique de validation croisée «5-Fold CV» le taux d'erreur test pour modèle LDA: 0.076 le taux d'erreur test pour modèle QDA: 0.084 et aussi on remarque que le modèle LDA est plus simple donc on prendre le premier modèle (modèle LDA)

d) Résumé graphique et comparaison des méthodes.

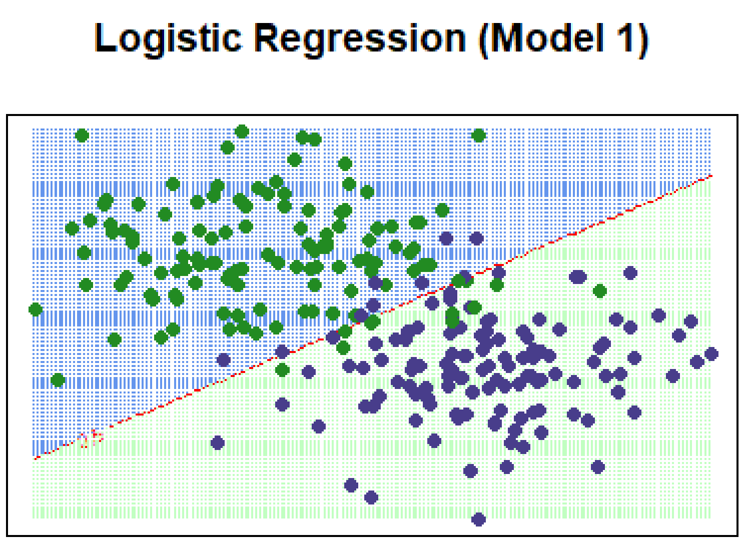
Tracer le nuage des 250 points (2 couleurs de votre choix) et ajouter au graphique les courbes (trois en tout, similaires à celles de la figure 5.7 page 185 dans ISL) séparant les deux classes dans chacun des cas suivants :

```
library(class)
#on trace la plateforme de knn
minx1= min(data[,1])
maxx1= max(data[,1])
x1 <- seq(from = minx1, to = maxx1,length.out=150)

minx2=min(data[,2])
maxx2=max(data[,2])
```

```
x2 <- seq(from = minx2, to = maxx2,length.out=100)

gd <- expand.grid(x1 = x1, x2 = x2)

#On calcule le modéle knn avec K optimal 4.
model.knn_opt=knn(train=data_xi,test=gd,cl=data_y,k=4,prob=TRUE)
probabilite <- attr(model.knn_opt, "prob")
probabilite <- ifelse(model.knn_opt=="1", probabilite, 1-probabilite)
probabiliteopt <- matrix(probabilite, length(x1), length(x2))


##### plot knn & LDA & Logic ######
# plot knn
contour(x1, x2, probabiliteopt, levels=0.5, labels="", xlab="", ylab="", main
= "KNN")



points(gd, pch=".", cex=1.2, col=ifelse(probabiliteopt >=0.5, "cornflowerblue
", "darkseagreen1"))
points(data_xi,pch=19, col=ifelse(data_y==1, "forestgreen", "darkslateblue"))

box()
```

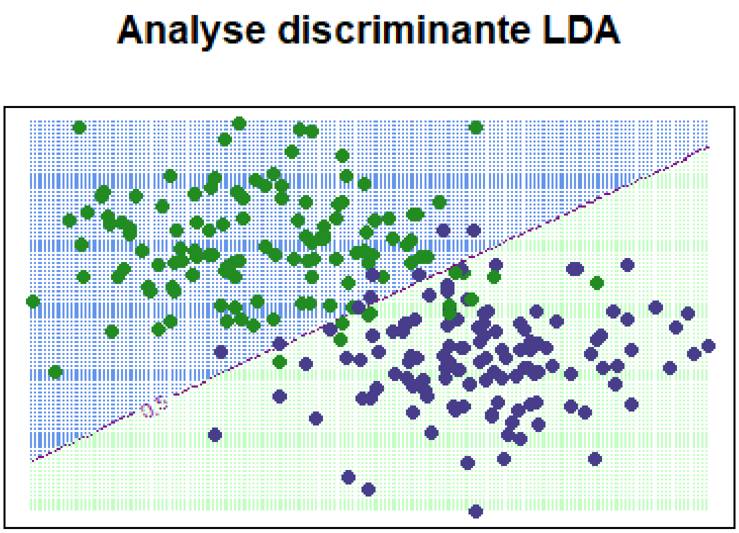


```r
#Logistic Regression

colnames(gd)=c("X1","X2")

model_glm = glm(Y ~ X1 + X2, data=data, family=binomial)
probabiliteglm = predict(model_glm, gd, type="response")
probabiliteoptglm= matrix(probabiliteglm, length(x1), length(x2))


contour(x1, x2, probabiliteoptglm, levels=0.5, main="Logistic Regression (Mod
el 1)", col="red",axes=FALSE)

points(gd, pch=".", cex=1.2, col=ifelse(probabiliteoptglm >=0.5, "cornflowerb
lue", "darkseagreen1"))
points(data_xi,pch=19, col=ifelse(data_y==1, "forestgreen", "darkslateblue"))
box()
```



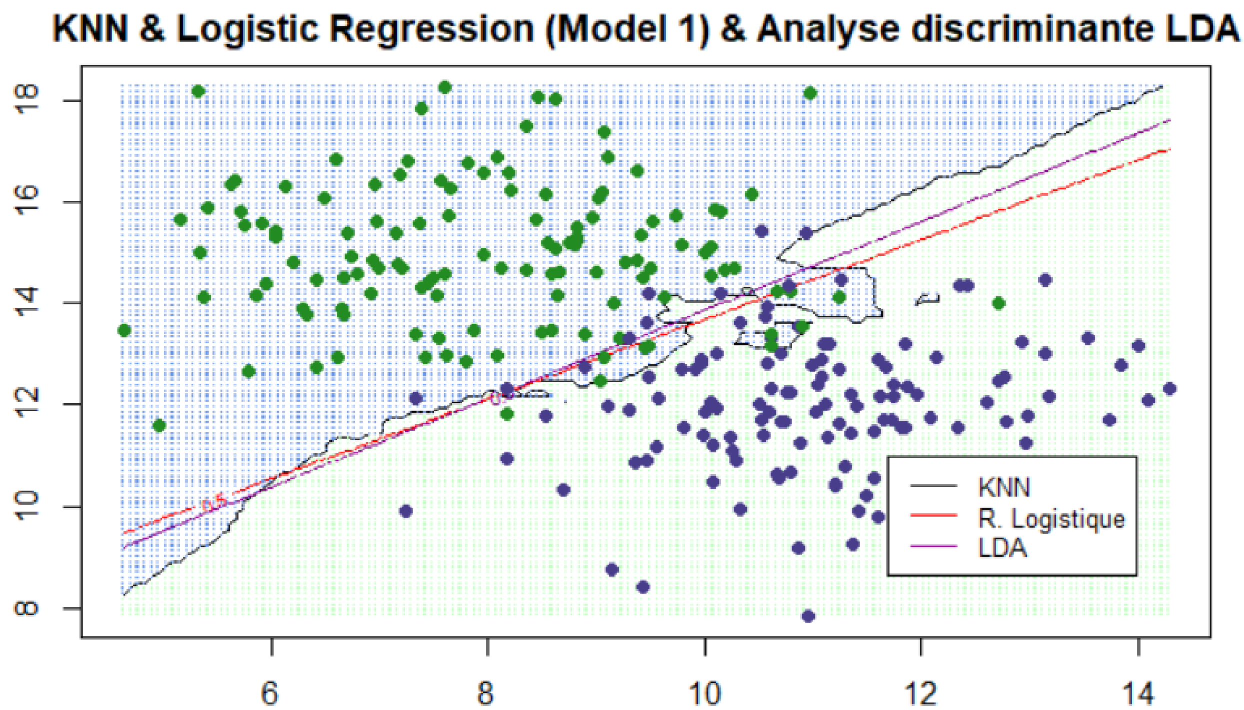

```r
#Analyse discriminante LDA

colnames(gd)=c("X1","X2")

model_lda = lda(Y ~ X1 + X2, data=data)
probabilitelda = predict(model_lda, gd, type="response")
prob=probabilitelda$posterior
probabiliteoptlda= matrix(prob, length(x1), length(x2))
```

```
contour(x1, x2, probabiliteoptlda, levels=0.5, main="Analyse discriminante LD
A", col="darkmagenta",axes=FALSE)

points(gd, pch=".", cex=1.2, col=ifelse(probabiliteoptlda <=0.5, "cornflowerb
lue","darkseagreen1"))
points(data_xi,pch=19, col=ifelse(data_y==1, "forestgreen", "darkslateblue"))
box()
```




```
##### plot knn & Analyse discriminante LDA & Logistic Regression (Model 1)

library(class)
#on trace la plateforme
minx1= min(data[,1])
maxx1= max(data[,1])
x1 <- seq(from = minx1, to = maxx1,length.out=150)

minx2=min(data[,2])
maxx2=max(data[,2])
x2 <- seq(from = minx2, to = maxx2,length.out=100)

gd <- expand.grid(x1 = x1, x2 = x2)

#On calcule le modéle knn avec K optimal 4.
model.knn_opt=knn(train=data_xi,test=gd,cl=data_y,k=4,prob=TRUE)
probabilite <- attr(model.knn_opt, "prob")
```

```r
probabilite <- ifelse(model.knn_opt=="1", probabilite, 1-probabilite)
probabiliteopt <- matrix(probabilite, length(x1), length(x2))

# plot knn

contour(x1, x2, probabiliteopt, levels=0.5, labels="", xlab="", ylab="", main
= "KNN & Logistic Regression (Model 1) & Analyse discriminante LDA")


# plot glm
colnames(gd)=c("X1","X2")

model_glm = glm(Y ~ X1 + X2, data=data, family=binomial)
probabiliteglm = predict(model_glm, gd, type="response")
probabiliteoptglm= matrix(probabiliteglm, length(x1), length(x2))

par(new=TRUE)
contour(x1, x2, probabiliteoptglm, levels=0.5, col="red",axes=FALSE)


# plot lda

model_lda = lda(Y ~ X1 + X2, data=data)
probabilitelda = predict(model_lda, gd, type="response")
prob=probabilitelda$posterior
probabiliteoptlda= matrix(prob, length(x1), length(x2))

par(new=TRUE)
contour(x1, x2, probabiliteoptlda, levels=0.5, col="darkmagenta",axes=FALSE)



points(gd, pch=".", cex=1.2, col=ifelse(probabiliteopt >=0.5, "cornflowerblue
", "darkseagreen1"))
points(data_xi,pch=19, col=ifelse(data_y==1, "forestgreen", "darkslateblue"))

legend(11.7, 11, legend=c("KNN", "R. Logistique", "LDA"), col=c("black","red"
, "darkmagenta"), lty=1, cex=0.8)
box()
```



QUESTION N°2

a) Donner une estimation ponctuelle thêta estimer

```
library(boot)
set.seed(1923715)

library(boot)
thetafct <- function(dd, index){

  X1<-dd$area[index]
  X2<-dd$peri[index]
  X3<-dd$shape[index]
  X4<-dd$perm[index]

  return(cor(log(abs(X1-3*X2)),pmax(250*X3,X4)))}

thetafct(rock, 1:48)

## [1] -0.1986994

thetaboot=boot(rock,thetafct,R=2000)
thetaboot

##
## ORDINARY NONPARAMETRIC BOOTSTRAP
##
##
```

```
## Call:
## boot(data = rock, statistic = thetafct, R = 2000)
##
##
## Bootstrap Statistics :
##       original       bias    std. error
## t1* -0.1986994 -0.007334332   0.1336759
```

b) Déduire des résultats qui précèdent un intervalle de confiance pour theta au niveau de confiance 95%. Commenter brièvement.

```
boot.ci(thetaboot, type="norm")

## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 2000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = thetaboot, type = "norm")
##
## Intervals :
## Level      Normal
## 95%   (-0.4534,  0.0706 )
## Calculations and Intervals on Original Scale
```

On remarque que notre thêta estimer se trouve dans notre intervalle de confiance

`[-0.4534,  0.0706]`

Et on remarque que la corrélation de notre modèle peut avoir une valeur négative

Question 3

a -

- on a $y_i = \beta_0 + \sum_{j=1}^{p} x_{ij}\beta_j + \varepsilon_i$ / $\varepsilon_1, \dots, \varepsilon_n$ sont independent

$$\varepsilon_i \sim N(0, \sigma^2) \Rightarrow P(Y_i|\theta) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left[-\frac{(\varepsilon_i)^2}{2\sigma^2}\right]$$

$$P(Y_i|\theta) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(Y_i - (\beta_0 + \sum_{j=1}^{p} x_{ij}\beta_j))^2}{2\sigma^2}\right]$$

$L(\theta)$?

$$L(\theta) = L(\theta/Y) = \prod_{i=1}^{n} P(Y_i|\theta)$$

$$= \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(Y_i - (\beta_0 + \sum_{j=1}^{p} x_{ij}\beta_j))^2}{2\sigma^2}\right]$$

$$L(\theta) = \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^n \exp\left[-\frac{1}{2\sigma^2} \cdot \sum_{i=1}^{n} (Y_i - (\beta_0 + \sum_{j=1}^{p} x_{ij}\beta_j))^2\right]$$

b-

on a

$$P(\beta) = \frac{1}{2b} \exp\left(-\frac{|\beta|}{b}\right), \quad |\beta| = \sum_{j=1}^{p} |\beta_j|$$

- le postérieur à double exponentielle avec moyenne 0 et paramètre b :

ISL P226

$$\# \cdots f(\beta / X, Y) \propto f(Y / X, \beta) P(\beta / X) = f(Y/X, \beta) P(\beta)$$

d'aprés la formule de la question (a) et #:

$$f(Y/\theta) P(\beta) = f(Y/X,\beta) P(\beta)$$

$$= \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^n \exp\left[-\frac{1}{2\sigma^2} \sum_{i=1}^{n} \left(y_i - (\beta_0 + \sum_{j=1}^{n} \beta_j x_{ij})\right)^2\right] \cdot \left[\frac{1}{2b} \exp\left(-\frac{|\beta|}{b}\right)\right]$$

$$\boxed{f(Y/X,\beta) P(\beta) = \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^n \left(\frac{1}{2b}\right) \cdot \exp\left[-\frac{1}{2\sigma^2} \sum_{i=1}^{n} \left(Y_i - (\beta_0 + \sum_{j=1}^{n} \beta_j X_{ij})\right)^2 - \frac{|\beta|}{b}\right]}$$

C -

Montrant que l'estimation de lasso pour β est le mode sous cette distribution postérieure, c'est la même que de montrer que la valeur la plus probable de β est donnée par la solution de lasso.

- on prendre l'equation de distribution postérieure Trouver dans la question (b), et on met le ln( )

$$\ln\left(f(Y/X,\beta)\,P(\beta)\right) = \ln\left[\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^n\left(\frac{1}{2b}\right)\cdot\exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}\left(Y_i-\left(\beta_0+\sum_{j=1}^{p}\beta_j X_{ij}\right)\right)^2-\frac{|\beta|}{b}\right\}\right]$$

$$= \ln\left[\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^n\left(\frac{1}{2b}\right)\right]-\left[\frac{1}{2\sigma^2}\sum_{i=1}^{n}\left(Y_i-\left(\beta_0+\sum_{j=1}^{p}\beta_j X_{ij}\right)\right)^2+\frac{|\beta|}{b}\right]$$

- on veut maximiser la postériore :

$$\hat{\beta} = \underset{\beta}{\arg\max}\; f(\beta/X,Y)$$

$$= \underset{\beta}{\arg\max}\; \ln\left[\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^n\left(\frac{1}{2b}\right)\right]-\left(\frac{1}{2\sigma^2}\sum_{i=1}^{n}\left(Y_i-\left(\beta_0+\sum_{j=1}^{p}\beta_j X_{ij}\right)\right)^2+\frac{|\beta|}{b}\right)$$

- on remarque que la maximisation de $f(\beta/X,Y)$ à équivalant de minimiser le 2em Terme (on a une sustraction)

donc on a :

$$= \arg\min_{\beta} \left\{ \frac{1}{2\sigma^2} \sum_{i=1}^{n} \left[Y_i - (\beta_0 + \sum_{j=1}^{p} \beta_j X_{ij}) \right]^2 + \frac{|\beta|}{b} \right\}$$

$$= \arg\min_{\beta} \left\{ \frac{1}{2\sigma^2} \sum_{i=1}^{n} \left[Y_i - (\beta_0 + \sum_{j=1}^{p} \beta_j X_{ij}) \right]^2 + \frac{1}{b} \sum_{j=1}^{p} |\beta_j| \right\}$$

$$= \arg\min_{\beta} \left\{ \frac{1}{2\sigma^2} \left[\sum_{i=1}^{n} \left[Y_i - (\beta_0 + \sum_{j=1}^{p} \beta_j X_{ij}) \right]^2 + \frac{2\sigma^2}{b} \sum_{j=1}^{p} |\beta_j| \right] \right\}$$

on pose $\lambda = \frac{2\sigma^2}{b}$:

$$= \arg\min_{\beta} \left\{ \sum_{i=1}^{n} \left[Y_i - (\beta_0 + \sum_{j=1}^{p} \beta_j X_{ij}) \right]^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}$$

$$= \arg\min_{\beta} RSS + \lambda \sum_{j=1}^{p} |\beta_j|$$

- d'apres l'équation de Lasso, la posteriori provient d'une distribution de laplace avec moyen 0 et une échelle commune b, le mode pour β est donné par la solution de Lasso lorsque $\boxed{\lambda = \frac{2\sigma^2}{b}}$

d-

on a $\beta_i \sim N(0,c)$ $P(\beta_i) = \frac{1}{\sqrt{2\pi c}} \cdot \exp\left(-\frac{\beta_i^2}{2c}\right)$

$f(Y/X,\beta)P(\beta)$?

$P(\beta)$? $P(\beta) = \prod_{i=1}^{p} P(\beta_i) = \prod_{i=1}^{p} \frac{1}{\sqrt{2\pi c}} \cdot \exp\left\{-\frac{\beta_i^2}{2c}\right\}$

$$\boxed{P(\beta) = \left(\frac{1}{\sqrt{2c\pi}}\right)^p \exp\left\{-\frac{1}{2c}\sum_{i=1}^{p}\beta_i^2\right\}}$$

d'apres la question (a) :

$f(Y/X,\beta)P(\beta) =$

$$= \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^n \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}\left[Y_i - \left(\beta_0 + \sum_{j=1}^{p}\beta_j X_{ij}\right)\right]^2\right\} \left(\frac{1}{\sqrt{2\pi c}}\right)^p \exp\left\{-\frac{1}{2c}\sum_{i=1}^{p}\beta_i^2\right\}$$

$$= \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^n \left(\frac{1}{\sqrt{2c\pi}}\right)^p \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}\left[Y_i - \left(\beta_0 + \sum_{j=1}^{p}\beta_j X_{ij}\right)\right]^2 - \frac{1}{2c}\sum_{i=1}^{p}\beta_i^2\right\}$$

e -

- c'est comme la question (C). , on montre que la valeur la plus probable de β. donnée par la solution de Ridge

- on prendr l'equation Trouver dans (d).

$$\ln\left(f(Y/X,\beta)P(\beta)\right) = \ln\left(\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^n\left(\frac{1}{\sqrt{2\pi c}}\right)^p \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}\left[Y_i-(\beta_0+\sum_{j=1}^{p}\beta_j x_{ij})\right]^2+\frac{1}{2c}\sum_{j=1}^{p}\beta_i^2\right\}\right)$$

$$= \ln\left[\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^n\left(\frac{1}{\sqrt{2\pi c}}\right)^p\right] - \left[\frac{1}{2\sigma^2}\sum_{i=1}^{n}\left[Y_i-(\beta_0+\sum_{j=1}^{p}\beta_j x_{ij})\right]^2+\frac{1}{2c}\sum_{i=1}^{p}\beta_i^2\right]$$

- on veut maximiser la posteriori:

$$\arg\max_{\beta} f(\beta/X,Y)$$

$$= \arg\max_{\beta} \ln\left[\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^n\left(\frac{1}{\sqrt{2\pi c}}\right)^p\right] - \left[\frac{1}{2\sigma^2}\sum_{i=1}^{n}\left[Y_i-(\beta_0+\sum_{j=1}^{p}\beta_j x_{ij})\right]^2+\frac{1}{2c}\sum_{i=1}^{p}\beta_i^2\right]$$

- on remarque que le maximisation de $f(\beta/X,Y)$ à équivalent de minimiser le 2em Terme (on a une sustraction (-))

donc on a:

$$= \arg\min_{\beta}\left\{\frac{1}{2\sigma^2}\sum_{i=1}^{n}\left[Y_i-(\beta_0+\sum_{j=1}^{p}\beta_j x_{ij})\right]^2+\frac{1}{2c}\sum_{i=1}^{p}\beta_i^2\right\}$$

$$= \arg\min_{\beta}\left\{\left(\frac{1}{2\sigma^2}\right)\left[\sum_{i=1}^{n}\left[Y_i-(\beta_0+\sum_{j=1}^{p}\beta_j x_{ij})\right]^2+\boxed{\frac{\sigma^2}{c}}\sum_{i=1}^{p}\beta_j^2\right]\right\}$$

On pose $\lambda = \frac{\sigma^2}{c}$:

$$= \arg\min_{\beta} \left(\frac{1}{2\sigma^2}\right) \left\{ \sum_{i=1}^{n} \left[Y_i - \left(\beta_0 + \sum_{j=1}^{p} \beta_j X_{ij}\right) \right]^2 + \lambda \sum_{i=1}^{p} \beta_i^2 \right\}$$

$$\boxed{= \arg\min_{\beta} RSS + \lambda . \sum_{i=1}^{p} \beta_i^2}$$

- d'aprés l'équation de Ridge, le postérieur provient d'une distribution normale avec moyen 0 et variance c, le mode pour β est donné par la solution de Ridge. lorsque $\boxed{\lambda = \frac{\sigma^2}{c}}$

- puisque le postérieur est gaussien, nous avons aussi que c'est le moyen postérieur.