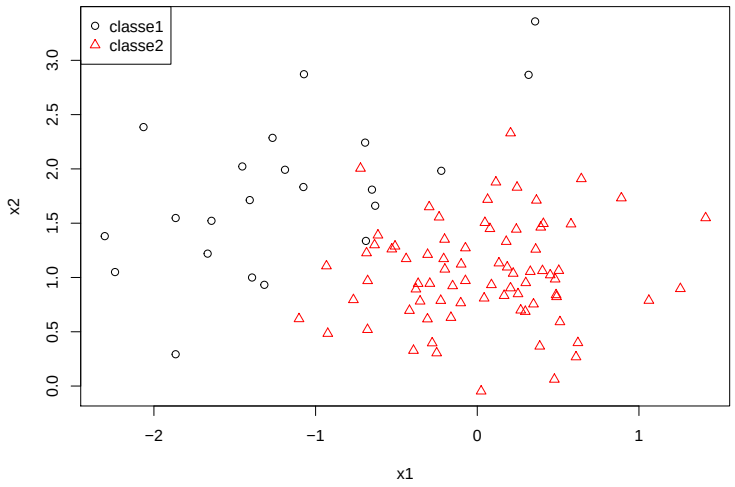
Apprentissage supervisé

TP2 : analyse discriminante linéaire et quadratique

Exercice 1. Les objectifs :
- découvrir l'analyse discriminante linéaire et quadratique.
- découvrir fonctions `lda` et `qda`.
- programmer l'analyse discriminante quadratique.

On utilise dans cet exercice les données synth_train.txt.

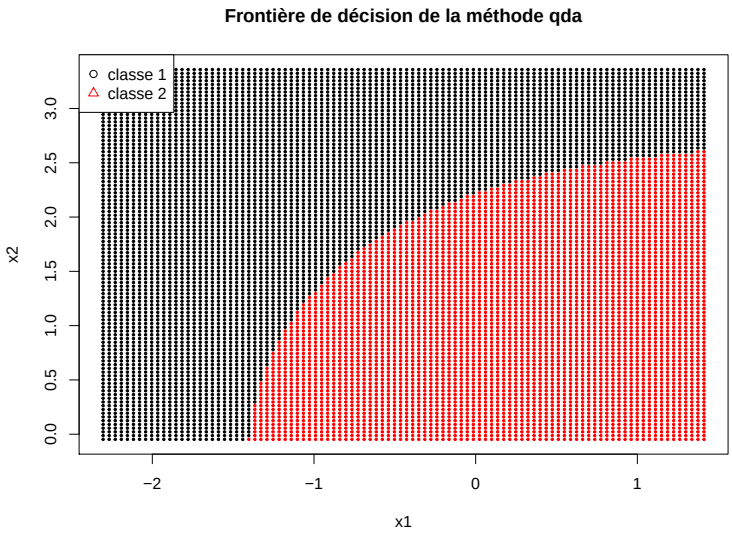
```
train <- read.table(file="../data/synth_train.txt", header=TRUE)
Xtrain <- train[,-1]
Ytrain <- train$y
plot(Xtrain, pch=Ytrain, col=Ytrain)
legend("topleft", legend=c("classe1", "classe2"), pch=1:2, col=1:2)
```



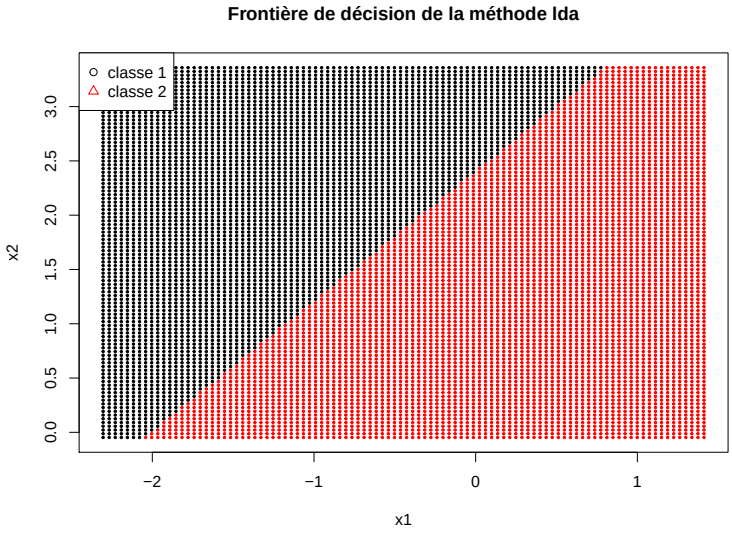
1. En analyse discriminante quadratique on fait l'hypothèse paramétrique gaussienne que $X|Y = k \sim \mathcal{N}(\mu_k, \Sigma_k)$:
$$f(x|Y = k) = \frac{1}{(2\pi)^{p/2}|\Sigma_k|^{1/2}} \exp(-\frac{1}{2}(x - \mu_k)^T \Sigma_k^{-1}(x - \mu_k))$$
Estimer sur les données d'apprentissage les paramètres inconnus μ_k, Σ_k et $\pi_k = \mathbb{P}(Y = k)$ pour $k = 1, 2$
2. Prédire la classe de la nouvelle observation $x = (0, 1)$ avec la méthode d'analyse discriminante quadratique. Pour cela, calculer $Q_1(x)$ et $Q_2(x)$ et vérifier que x est bien affecté à la classe 2.
3. Estimer la probabilité pour que le point $x = (0, 1)$ appartienne à la classe 1 et la probabilité pour qu'il appartienne à la classe 2.
4. Implémenter une fonction `adq_estim` qui prend en entrée une matrice de données X et un vecteur de classes Y et estime le vecteur de paramètres inconnus $\theta = (\pi_1, \ldots, \pi_K, \mu_1, \ldots, \mu_K, \Sigma_1, \ldots, \Sigma_K)$. La sortie de cette fonction doit être une liste de taille K et chaque élément de cette liste doit être la liste des estimations de π_k, μ_k et Σ_k. N'oubliez pas de nommer les éléments des listes afin de

rendre les résultats plus lisibles. Vérifiez enfin que vous retrouvez les résultats de la question 1) avec cette fonction.

5. Implémenter une fonction `adq_pred` qui prend en entrée les paramètres estimés avec la fonction `adq_estim` et une matrice de nouvelles données à prédire. Cette fonction doit fournir en sortie une liste avec le vecteur des prédictions et la matrice des probabilités à posteriori de ces nouvelles observations. Vérifiez enfin que vous retrouvez les résultats des questions 2) et 3) avec cette fonction.
6. Prédire maintenant les classes les points de coordonnées $(0, 1)$ et $(-2, 2)$ et estimer leurs probabilités à posteriori d'être dans chacune des deux classes.
7. Utiliser maintenant les fonctions `qda` et `predict.qda` du package `MASS` pour prédire la classe de ces deux points et estimer leurs probabilités à posteriori. Vérifier que vous retrouvez les résultats obtenus à la question 6).
8. Représenter la frontière de décision de la méthode `qda` à partir de la grille de points du TP1.



9. En analyse discriminante linéaire, les matrices de covariance sont supposées égales. Estimer sur les données d'apprentissage la matrice $\Sigma = \Sigma_1 = \Sigma_2$.
10. On veut maintenant prédire la classe de la nouvelle observation $x = (0, 1)$ avec la méthode d'analyse discriminante linéaire. Calculer $L_1(x)$ et $L_2(x)$ et vérifier que x est bien affecté à la classe 2.
11. Estimer la probabilité pour que le point $x = (0, 1)$ appartienne à la classe 1 et la probabilité pour qu'il appartienne à la classe 2.
12. Utiliser maintenant les fonctions `lda` et `predict.lda` du package `MASS` pour prédire la classe des deux points $(0, 1)$ et $(-2, 2)$ et estimer leurs probabilités à posteriori. Vérifier que vous retrouvez les résultats des questions précédentes pour le point $(0, 1)$.
13. Représenter la frontière de décision de la méthode lda à partir de la grille de points du TP1.



Exercice 2. Faire de la reconnaissance automatique de caractères manuscrits.

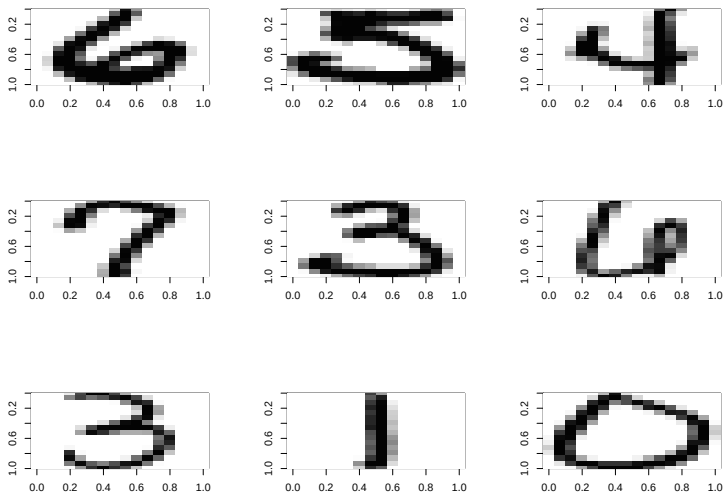
Récupérer les jeux de données numbers_train.txt et numbers_test.txt. Chaque fichier contient 500 images de dimension 16×16 et chaque image représente un caractère manuscript (un chiffre entre 0 et 9). On a donc $Y \in \{0, \ldots, 9\}$ et $X = (X^1, \ldots, X^{256}) \in \mathbb{R}^{256}$. Il s'agit d'image en niveaux de gris où chaque pixel prend une valeur entre 0 (noir) et 1 (blanc).

1. Importer les 500 images du fichier numbers_train.txt.

```
data <- read.table("../data/numbers_train.txt", header=TRUE)
Xtrain <- as.matrix(data[,-1])
Ytrain <- as.factor(data[,1])
```

2. Visualiser les neuf premières images.

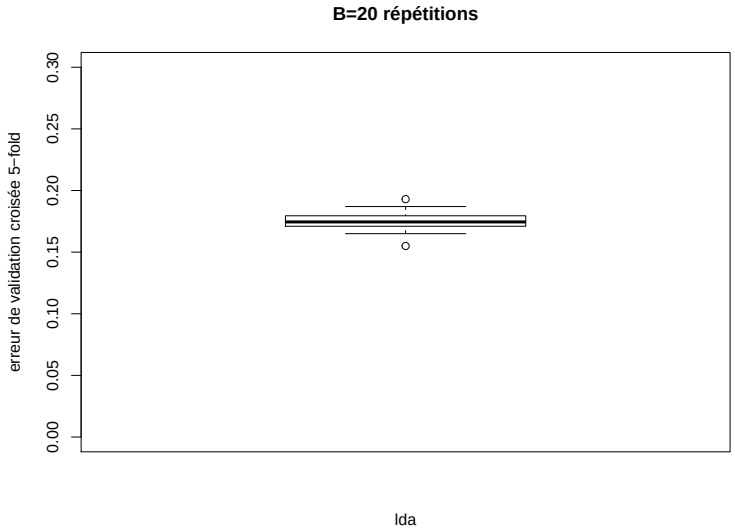
```
par(mfrow=c(3,3))
for (i in 1:9){
  image(matrix(Xtrain[i,],16,16), col=gray(1:100/100), ylim=c(1,0))
}
```



3. Prédire avec la méthode `lda` les classes des 500 images de l'ensemble d'apprentissage et calculer le taux d'erreur d'apprentissage.
4. Importer les 500 images du fichier numbers_train.txt.

```
data <- read.table("../data/numbers_test.txt", header=TRUE)
Xtest <- as.matrix(data[,-1])
Ytest <- as.factor(data[,1])
```

5. Prédire maintenant les classes des 500 images de l'ensemble test et calculer le taux d'erreur test.
6. Reprendre toutes les données (apprentissage et test) pour calculer l'erreur de validation croisée LOO de la méthode `lda`. Que constatez-vous ?
7. Recommencez en utilisant cette fois l'argument `CV=TRUE` de la fonction `lda` quel taux d'erreur de validation croisée LOO trouvez-vous ?
8. Reprendre toutes les données (apprentissage et test) pour calculer l'erreur de validation croisée 5-folds de la méthode `lda`, $B = 20$ fois. Conclure. N'hesitez pas à sauver les résultats pour ne pas avoir à relancer la procédure à chaque fois.



Exercice 3. Objectifs :

- Explorer des données par Analyse en Composantes Principales.
- Comparer les performances de plusieurs méthodes et choisir la "meilleure".

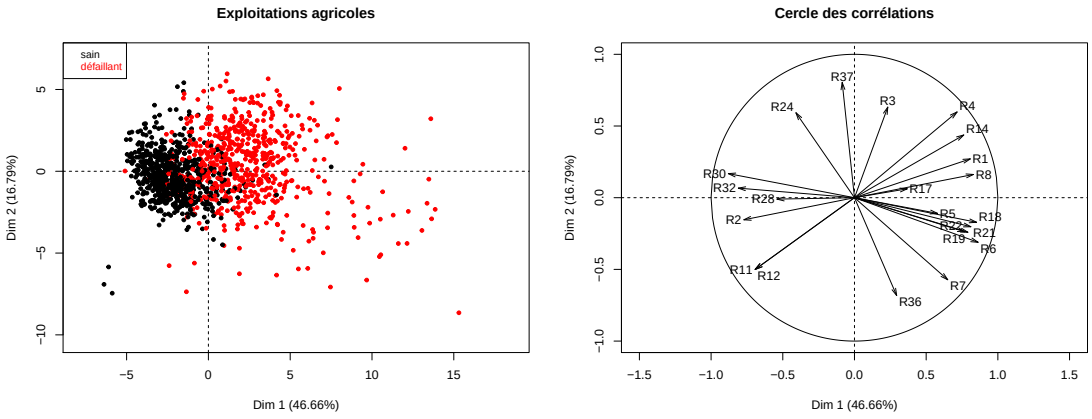
On reprend l'application à des données réelles du TP1 où $n = 1260$ exploitations agricoles étaient réparties en $K = 2$ classes : la classe des exploitations saines et la classe des exploitations défaillantes. Mais on utilise maintenant le jeu de données complet [1] où les exploitations agricoles sont décrites par $p = 22$ critères économiques et financiers.

```
load("../data/Desbois_complet.rda")
dim(data)

## [1] 1260   23

colnames(data)

##  [1] "DIFF" "R1"   "R2"   "R3"   "R4"   "R5"   "R6"   "R7"   "R8"   "R11"
## [11] "R12"  "R14"  "R17"  "R18"  "R19"  "R21"  "R22"  "R24"  "R28"  "R30"
## [21] "R32"  "R36"  "R37"

X <- data[,-1]
Y <- data$DIFF
Y <- as.factor(Y)
levels(Y) <- c("sain", "défaillant")
```

La variable qualitative à expliquer est toujours la variable difficulté de paiement (0=sain et 1=défaillant) avec ici

- prédire 1 (défaillant)=positif,
- prédire 0 (sain)=négatif.

1. On réalise d'abord une Analyse en Compostantes Principales des données d'entrées (la matrice X). Le vecteur des sorties Y est uniquement utilisé pour colorer les exploitations agricoles sur le premier plan factoriel. Retrouver les graphiques ci-dessous et les interpréter. A votre avis, seras-il difficile de construire une bonne règle de classification ?



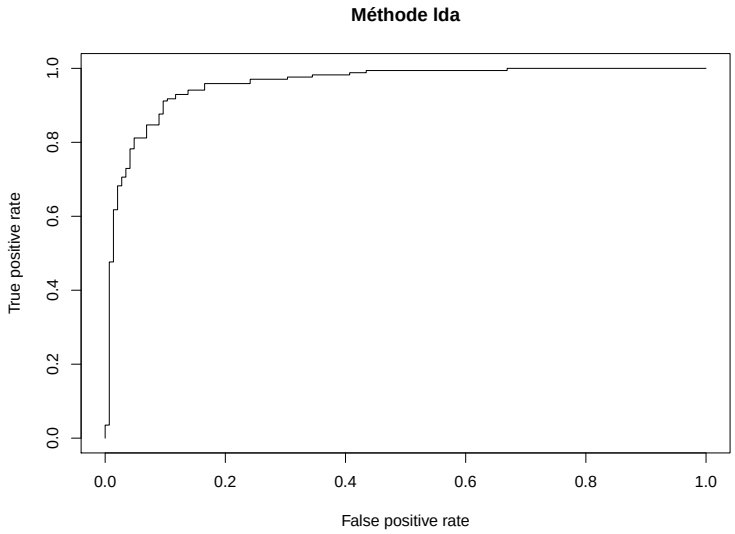
2. Découpez aléatoirement les $n = 1260$ exploitations agricoles en 945 exploitations pour les données d'apprentissage et 360 exploitations pour les données test.

1. http ://publications-sfds.fr/index.php/csbigs/article/view/351/331

```
tr <- sample(1:nrow(X),945)
Xtrain <- X[tr,]
Ytrain <- Y[tr]
Xtest <- X[-tr,]
Ytest <- Y[-tr]
```

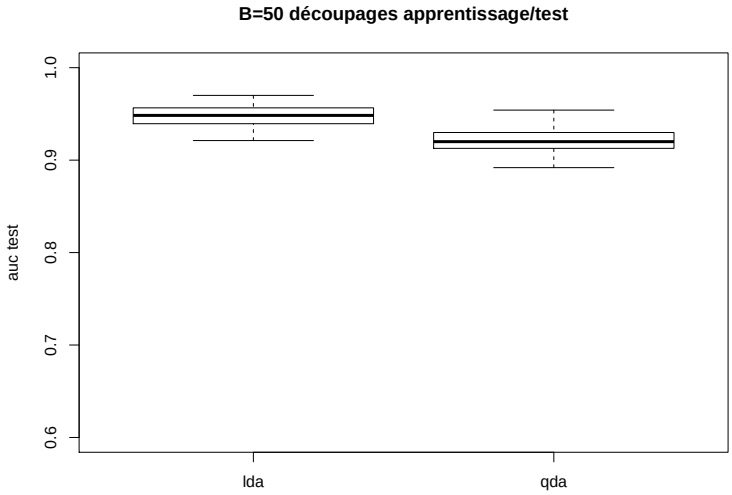
3. Constuire la courbe ROC de la méthode d'analyse disciminante linéaire (lda) en suivant la procédure suivante :
   (a) Estimer les paramètres de la méthode lda sur les données d'apprentissage.
   (b) En déduire le score (probabilité à posteriori d'être défaillante) des exploitations agricoles des données test.
   (c) Constuire à partir de ce score (probabilités à posteriori d'être défaillante) la courbe ROC.

   Quelle est la valeur de l'auc sur les données test de ce découpage ?



4. Même question avec la méthode `qda`. Quelle méthode obtient le meilleur `auc` sur les données test ?

5. On veut maintenant comparer les performances des méthodes d'analyse discriminantes linéaires (lda) et quadratique (qda) **sur plusieurs découpages**. Pour cela, on utilise la méthodologie suivante :
   (a) Découper aléatoirement B fois les données en 80% de données d'apprentissage (945 exploitations) et 20% de données test (360 exploitations).
   (b) Pour chaque découpage :
      i. Apprendre les paramètres des méthodes lda et qda.
      ii. En déduire les scores (probabilités à posteriori d'être défaillante) des données test avec les deux méthodes.
      iii. Calculer le critère AUC de chaque méthode.
   (c) Représenter les boxplots des AUC obtenus avec les méthodes lda et qda.



Quelle méthode semble construire le meilleur score ?

6. Même question avec cette fois le taux d'erreur.

Exercice 4. Objectifs :

— Sélectionner des variables en analyse discriminante linéaire.
— Evaluer si la sélection de variable détériore la qualité des prédictions.

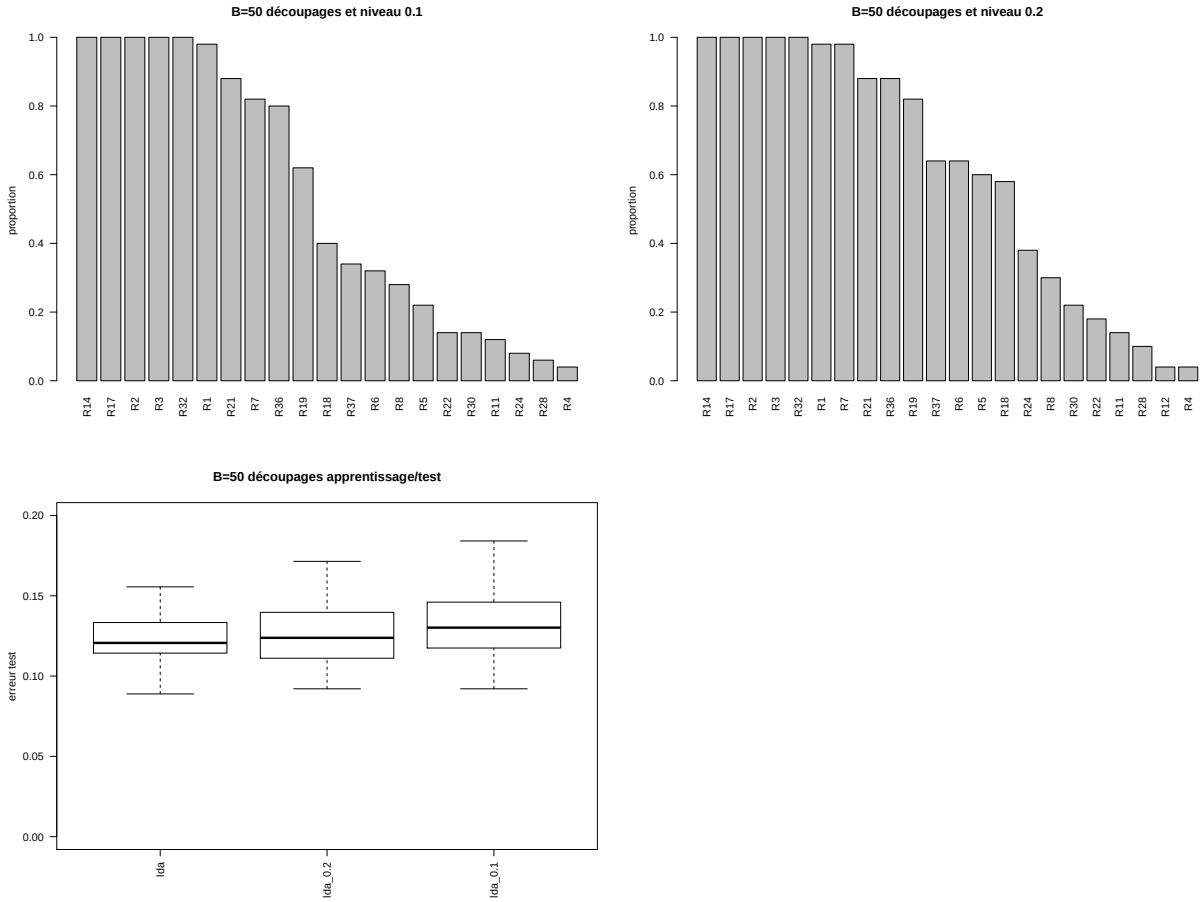
On veut connaître les variables importantes pour prédire si une exploitation agricole est saine ou défaillante. Il existe de nombreuses méthodes de sélection de variables. Ici on utilise une méthode simple de sélection pas à pas ascendante en **analyse discriminante linéaire**, implémentée dans la fonction `greedy.wilks` du package `klaR`.

1. On reprend les données de l'exercice 3 et on effectue un premier découpage apprentissage/test.

```
tr <- sample(1:nrow(X),945)
Xtrain <- X[tr,]
Ytrain <- Y[tr]
Xtest <- X[-tr,]
Ytest <- Y[-tr]
```

2. Appliquer aux données d'apprentissage la procédure de sélection de variable pas à pas ascendante implémentée dans la fonction `greedy.wilks`. Décrire rapidement cette procédure. Quel est le paramètre à fixer ? Quelles sont les variables sélectionnées avec la valeur par défaut de ce paramètre ?
3. Appliquer maintenant la même procédure avec la valeur 0.1 de ce paramètre. Le nombre de variables sélectionnées a-t-il augmenté ou diminué ?
4. Estimer les paramètres de la méthode lda avec les variables ainsi sélectionnées. Prédire ensuite les données test. Quel est le taux d'erreur test ?
5. Estimer les paramètres de la méthode lda avec toutes les variables. Prédire ensuite les données test. Quel est le taux d'erreur test ? Comparer au taux d'erreur test de la question précédente.
6. Refaire les trois questions précédentes avec un autre découpage apprentissage/test. Les variables sélectionnées sont-elles différentes ? Les taux d'erreur test sont-ils différents ?

7. Ecrire le code permettant de retrouver les graphiques ci-dessous.



Quelles variables sont le plus souvent sélectionnées ? Comparer les performances des trois méthodes ?

8. Finalement, c'est la méthode "lda_0.1" qui est choisie. Quelles sont les variables présentes dans le modèle final ? Prédire la classe de la 3ème exploitation agricole avec ce modèle.

Exercice 5. Objectif : Construire le score de Fisher.

En analyse discriminante linéaire, on constuit souvent le score de Fisher qui prend ses valeurs dans $\mathbb{R}$ et mesure un risque par exemple un risque financier (ici risque pour une exploitation agricole d'être défaillante). Ce score est définit par

$$\Delta(x) = L_1(x) - L_0(x) = x^T\Sigma^{-1}(\mu_1 - \mu_0) - \frac{1}{2}(\mu_1 + \mu_0)^T\Sigma^{-1}(\mu_1 - \mu_0) + \log(\frac{\pi_1}{\pi_0}).$$

1. On reprend les données de l'exercice 3 et les variables sélectionnées à la fin de l'exercice 4 et on effectue un découpage apprentissage/test.

```
set.seed(1234)
tr <- sample(1:nrow(X),945)
Xtrain <- X[tr,varsel]
Ytrain <- Y[tr]
Xtest <- X[-tr,varsel]
Ytest <- Y[-tr]
```

2. Estimer sur les données d'apprentissage les paramètres $(\pi_0, \pi_1, \mu_0, \mu_1, \Sigma)$ de la méthode lda en prenant "1=défaillant" et "0=sain".
3. En déduire les coefficients du score de Fisher.

```
##             [,1]
## intercept  -0.40
## R1          4.79
## R32       -13.92
## R14         1.23
## R17        37.69
## R2         -4.93
## R3          5.32
## R36         0.82
## R21        -1.18
## R7          4.65
## R18       -10.33
## R19        -3.57
## R6         -2.02
## R37        -3.20
## R24         7.09
## R5         -3.76
```

4. Calculer le score de Fisher des exploitations agricoles de l'échantillon test. Quelles sont les 5 exploitations les plus à risque ?
5. Prédire la classe des exploitations agricoles des données test à partir de ce score.
6. Enfin, retrouver à partir de ce score les probabilités à posteriori d'être défaillant des 5 exploitations agricoles les plus à risque. Retrouver ce résultat avec la fonction lda.
7. Pour finir, comment appliquer avec la fonction lda la règle géométrique de classement ?

Exercice 6. Règle géométrique de classement.

On ajoute maintenant l'hypothèse d'égalité des probabilités à priori. La méthode d'analyse discriminante linéaire s'interprète alors comme une règle géométrique de classement qui consiste à affecter une nouvelle observation à la classe la plus proche (celle dont le centre de gravité est le plus proche). La distance (au carré) entre une observation x et un centre de gravité g_k est alors :

$$D_k(x) = (x - \mu_k)^T \Sigma^{-1} (x - \mu_k)$$

Et la formule permettant de retrouver les probabilités à posteriori est :

$$\mathbb{P}(Y = k | X = x) = \frac{\exp -\frac{1}{2} (D_k(x))}{\sum_{\ell=1}^{K} \exp \frac{1}{2} (D_\ell(x))}$$

1. Quel est le nom de cette distance ?
2. En repartant de l'exercice 5, calculer $D_1(x)$ et $D_0(x)$ pour l'exploitation agricole "5". Quelle est alors la classe prédite pour cette exploitation. Est-elle correcte ?
3. Calculer la probabilité à posteriori pour que cette exploitation agricole soit défaillante. Retrouver ce résultat avec la fonction `lda`
4. Prédire toutes les données de l'ensemble test avec la règle géométrique.