



RECONNAISSANCES DES FORMES – TP N°06

Classification supervisée par analyse discriminante

Sami BARCHID – Anthony SLIMANI

Sommaire

<i>Introduction</i>	<i>2</i>
<i>Affichage des observations d'apprentissage</i>	<i>3</i>
<i>Estimation des moyennes et covariance des observations d'apprentissage</i>	<i>4</i>
<i>Analyse discriminante linéaire.....</i>	<i>6</i>
<i>Analyse discriminante quadratique</i>	<i>8</i>
<i>Annexe 1 – Macro du TP 6.....</i>	<i>9</i>

Introduction

L'objectif de ce TP est de classer des données multivariées (= données possédant plusieurs attributs, 2 attributs pour ce TP) par analyse discriminante linéaire ou quadratique.

Ce rapport est divisé en trois parties :

- **Analyse des ensembles de données** : présentation et analyse des données multivariées utilisées pour ce TP.
- **Analyse discriminante linéaire** : expérimentation et interprétations des résultats de classification obtenus par analyse discriminante linéaire sur les données multivariées utilisées.
- **Analyse discriminante quadratique** : expérimentation et interprétations des résultats de classification obtenus par analyse discriminante quadratique sur les données multivariées utilisées.

Affichage des observations d'apprentissage

Durant ce TP, nous avons eu à disposition deux ensembles :

- Un ensemble d'apprentissage présenté en figure 1.
- Un ensemble de test présent en figure 2.

Notre but est de déterminer des lignes de décisions afin de classer les points sur l'ensemble d'apprentissage et voir si cette classification est correcte en le testant sur l'ensemble de test.

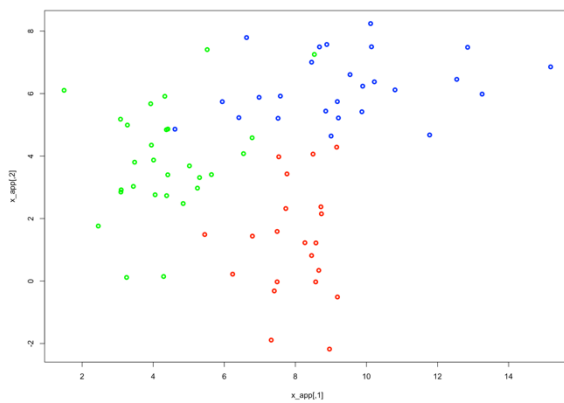


FIGURE 1 - ENSEMBLE D'APPRENTISSAGE

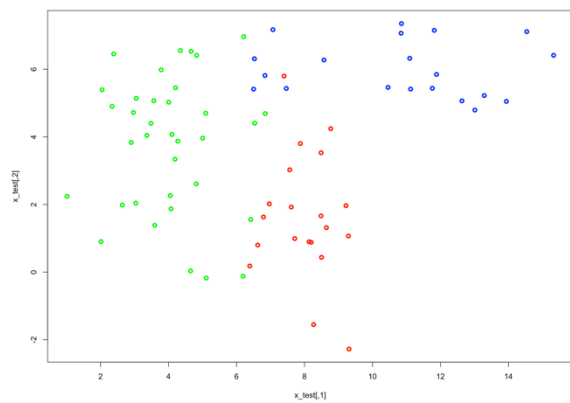


FIGURE 2 - ENSEMBLE DE TEST

Tailles des populations :

- Ensemble d'apprentissage : **75**
 - Classe 1 (rouge) : **21**
 - Classe 2 (bleu) : **26**
 - Classe 3 (vert) : **28**
- Ensemble de test : **75**
 - Classe 1 (rouge) : **20**
 - Classe 2 (bleu) : **20**
 - Classe 3 (vert) : **35**

Sur l'ensemble d'apprentissage, on constate que les probabilités à priori p_1 , p_2 et p_3 des classes 1, 2 et 3 (respectivement) sont à peu près égales : p_1 et p_2 sont chacune à 30% tandis que la probabilité à priori p_3 représente 40% de la population. On peut dire les populations des différentes classes sont équiprobables.

Estimation des moyennes et covariance des observations d'apprentissage

Pour cette deuxième partie, nous avons dû estimer les vecteurs moyennes des trois classes selon les deux attributs. Nous obtenons les vecteurs moyennes suivants :

- $\overrightarrow{v_{M1}} : (7,95; 1,24) - \overrightarrow{v_{m1}} : (8 ; 1)$
- $\overrightarrow{v_{M2}} : (9,39; 6,22) - \overrightarrow{v_{m2}} : (10 ; 6)$
- $\overrightarrow{v_{M3}} : (4,37; 3,87) - \overrightarrow{v_{m3}} : (4 ; 4)$

D'après ces résultats, on peut conclure que les points :

- De la classe 1 seront regroupés autour des coordonnées (8 ; 1)
- De la classe 2 seront regroupés autour des coordonnées (10 ; 6)
- De la classe 3 seront regroupés autour des coordonnées (4 ; 4)

On remarque bien cela quand on fait apparaître ces points sur l'ensemble comme on peut le voir sur la figure 4 où les vecteurs moyens calculés sont représentés par des cercles solides.

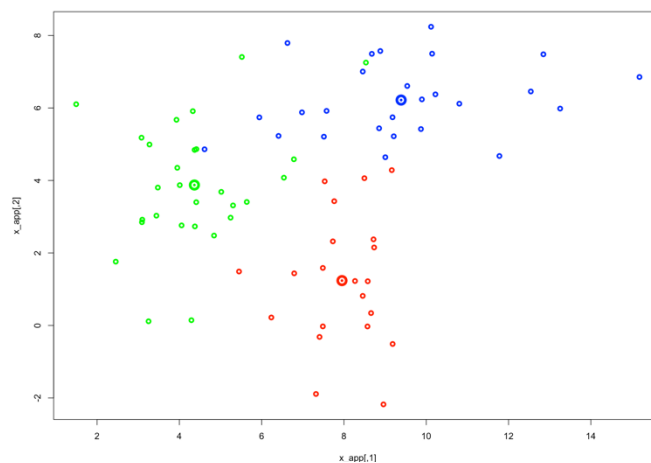


FIGURE 3 - REPRESENTATION DES VECTEURS MOYENS SUR L'ENSEMBLE D'APPRENTISSAGE

Puis, nous avons calculé les matrices de covariance des trois classes :

$$\begin{aligned} \bullet \quad \Sigma_1 &= \begin{pmatrix} 0,91 & 0 \\ 0 & 10,64 \end{pmatrix} & \bullet \quad \Sigma_2 &= \begin{pmatrix} 34,25 & 0,37 \\ 0,37 & 1,10 \end{pmatrix} & \bullet \quad \Sigma_3 &= \begin{pmatrix} 4,10 & 0,52 \\ 0,52 & 9,49 \end{pmatrix} \end{aligned}$$

On constate que les éléments non-diagonaux de Σ_1 , Σ_2 et Σ_3 sont proches de zéro, ce qui signifie que les deux attributs, représentés par l'abscisse et l'ordonnée, sont indépendants. Nous pouvons, de plus, affirmer que pour un échantillon de taille plus importante, nous aurions obtenus des éléments non-diagonaux se rapprochant encore plus de 0. On constate également que :

- Σ_1 et Σ_3 dépendent plus du deuxième attribut ce qui explique la répartition verticale de ces deux classes
- Σ_2 dépend plus du premier attribut ce qui explique la répartition horizontal de cette classe.

Nous avons, ensuite, comparé les éléments diagonaux des matrices de covariances avec les variances théoriques pour chacune des classes selon les attributs 1 et 2 présentés en tableau 1.

Classe		Selon attribut 1	Selon attribut 2
1	Matrices de covariances	0,95	3,26
	Carré de l'écart-type théorique	1	4
2	Matrices de covariances	5,85	1,04
	Carré de l'écart-type théorique	9	1
3	Matrices de covariances	2,03	3,08
	Carré de l'écart-type théorique	2,25	4

TABLEAU 1 - VARIANCES DES ATTRIBUTS DE L'ENSEMBLE D'APPRENTISSAGE

Nous remarquons que, en général, les éléments diagonaux des matrices de covariances sont égaux par rapport aux carrés des écarts-types selon les attributs 1 et 2. Les légères différences observées s'expliquent par le fait que la taille de la population est réduite, ce qui laisse une plus grande marge d'erreur par rapport à la valeur théorique.

Analyse discriminante linéaire

Nous avons dû appliquer l'analyse linéaire discriminante linéaire (**LDA**) sur les données d'apprentissage afin de déterminer les lignes de décisions pour séparer la classe 1 des autres et également pour la classe 2. On obtient alors le résultat présenté en figure 4 avec la ligne vert représentant la séparation de la classe 3 par rapport aux classes 2 et 3, et en ligne bleu la séparation de la classe 2 par rapport aux classes 1 et 3. On constate que 8 points sont mal classés, ce qui correspond à un pourcentage de réussite de $67/75 = 89,33\%$.

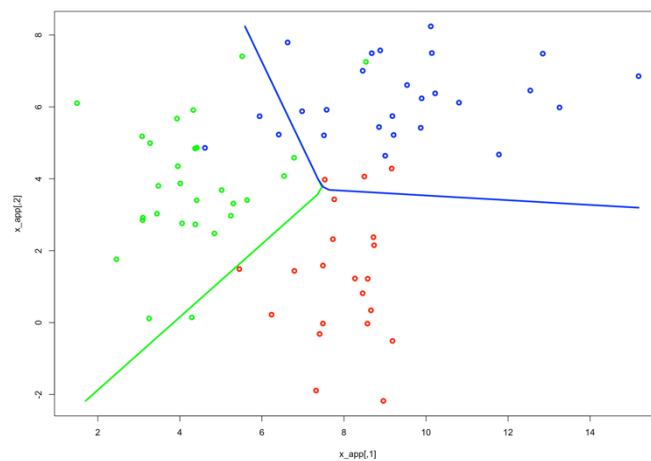


FIGURE 4 - LIGNES DE DECISIONS PAR LDA

La figure 5 représente les données de test avec une couleur correspondant aux classes originales et une forme spécifique correspondant aux classes d'assignation par LDA selon la légende suivante :

- Classe 1 :
 - Forme : cercle
 - Couleur : rouge
- Classe 2 :
 - Forme : triangle
 - Couleur : bleu
- Classe 3 :
 - Forme : croix
 - Couleur : vert
 -

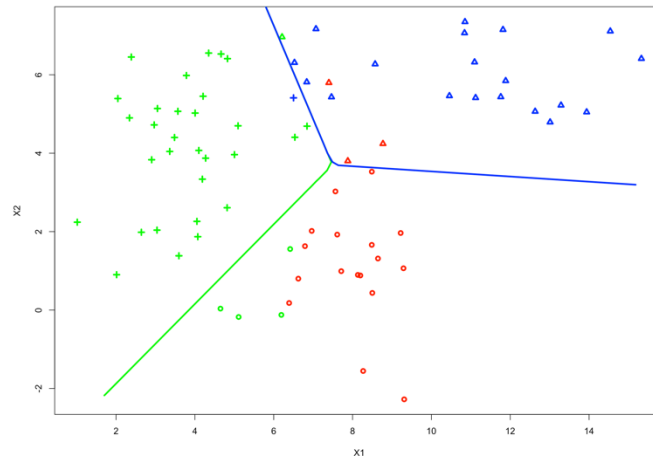


FIGURE 5 - REGLE DE DECISION PAR LDA SUR LES DONNEES DE TEST

Nous avons, ensuite, compté le nombre d'observations faussées (la classe d'assignation par LDA ne correspond pas à la classe originale, c'est-à-dire que les formes trouvées ne correspondent pas à la couleur de la classe). Nous obtenons 9 observations faussées, ce qui implique que 66 points ont été correctement classés. Nous obtenons alors 66/75 points correctes, c'est-à-dire un taux de classification correcte de **88%**, qui est le taux de classification correcte calculé précédemment.

Nous avons, ensuite, comparé les taux de bonnes classifications par classe sur le tableau suivant :

- Classe 1 : 85%
- Classe 2 : 95%
- Classe 3 : 85,7%

Nous remarquons que la classe dont les observations de tests ont été les mieux classées était la **classe 2**.

Analyse discriminante quadratique

Pour cette dernière partie de ce TP, nous avons dû appliquer l'analyse quadratique discriminante linéaire (**QDA**) de la même manière que l'on a faite précédemment avec la LDA. On obtient en figures 6 et 7 les résultats de l'application de la QDA sur les données d'apprentissage et de test. Nous remarquons que les lignes de décisions produisent un nombre de points mal classifiés plus petit que celui obtenu avec les lignes de décisions par analyse linéaire discriminante. En effet, on observe 4 points mal classifiés par analyse quadratique par rapport aux 8 points mal classés obtenus par analyse linéaire pour les données d'apprentissage. Pour l'application de la règle de classification par QDA sur les données de test, on obtient un taux de bonne classification de **93,33%** avec comme taux de bonnes classifications par classe :

- Classe 1 : 95%
- Classe 2 : 100%
- Classe 3 : 88,57%

Nous remarquons que la classe dont les observations de tests ont été les mieux classées était la **classe 2**.

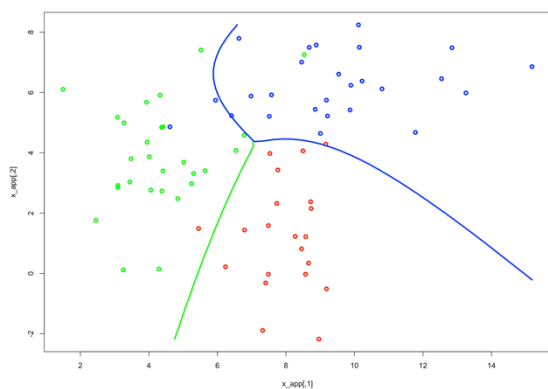


FIGURE 6 – REGLE DE DECISION PAR QDA SUR LES DONNEES D'APPRENTISSAGE

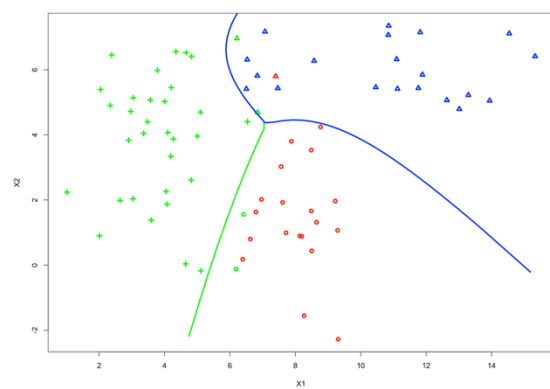


FIGURE 7 - REGLE DE DECISION PAR QDA SUR LES DONNEES DE TEST

En conclusion de nos expérimentations, nous observons donc que, dans notre cas, l'analyse discriminante quadratique est **plus efficace** que l'analyse discriminante linéaire dû au fait que l'analyse quadratique classe mieux les observations ayant des valeurs proches selon les attributs 1 et 2.

Annexe 1 – Macro du TP 6

```
library("MASS")
library("lattice")
load("./simul-2017.Rdata")

#Valeur pour mettre plus en valeur les screens pour le rapport
lwd_default <- 1

#####
# Q1.1
#####
#Affichage de l'ensemble d'apprentissage
couleur_app<-rep('red',n_app)
couleur_app[classe_app==1]='red'
couleur_app[classe_app==2]='blue'
couleur_app[classe_app==3]='green'

plot(x_app, col = couleur_app, lwd = lwd_default)

#Affichage de l'ensemble de test
couleur_test<-rep('red',n_test)
couleur_test[classe_test==1]='red'
couleur_test[classe_test==2]='blue'
couleur_test[classe_test==3]='green'
plot(x_test, col = couleur_test, lwd = lwd_default)

#####
# Q1.2
#####
#Affichage de l'ensemble d'apprentissage
plot(x_app, col = couleur_app, lwd = lwd_default)

# Déclaration des vecteurs moyennes
M1 = seq(1,2)
M2 = seq(1,2)
M3 = seq(1,2)

# Vecteur moyenne de la classe 1
M1[1] = mean(x_app[classe_app==1,1])
M1[2] = mean(x_app[classe_app==1,2])

# Vecteur moyenne de la classe 2
M2[1] = mean(x_app[classe_app==2,1])
M2[2] = mean(x_app[classe_app==2,2])

# Vecteur moyenne de la classe 3
M3[1] = mean(x_app[classe_app==3,1])
```

```

M3[2] = mean(x_app[classe_app==3,2])

points(M1[1], y = M1[2], pch = 20, col = "red", lwd=15)
points(M2[1], y = M2[2], pch = 20, col = "blue", lwd=15)
points(M3[1], y = M3[2], pch = 20, col = "green", lwd=15)

#Affichage de l'ensemble d'apprentissage
plot(x_app, col = couleur_app, lwd = lwd_default)

# Matrices de co variance
Sigma1 <- matrix(1,2,2)
Sigma2 <- matrix(1,2,2)
Sigma3 <- matrix(1,2,2)

for (i in 1:2) {
  for (j in 1:2) {
    Sigma1[i,j]=cov(as.vector(x_app[classe_app==1,i]),
as.vector(x_app[classe_app==1,j]))
    Sigma2[i,j]=cov(as.vector(x_app[classe_app==2,i]),
as.vector(x_app[classe_app==2,j]))
    Sigma3[i,j]=cov(as.vector(x_app[classe_app==3,i]),
as.vector(x_app[classe_app==3,j]))
  }
}

s1 = s1^2
s2 = s2^2
s3 = s3^2

#####
# Q1.3
#####

# Grille d'estimation de la densité de probabilité en 50 intervalles selon 1er
attribut
xp1 <- seq(min(x_app[,1]),max(x_app[,1]),length=50)
# Grille d'estimation de la densité de probabilité en 50 intervalles selon 2eme
attribut
xp2 <- seq(min(x_app[,2]),max(x_app[,2]),length=50)

grille <- expand.grid(x1=xp1,x2=xp2)
x_app_lda<-lda(x_app,classe_app)

# Estimation des densités de probabilités a posteriori dans Zp
grille = cbind(grille[,1],grille[,2])
Zp <- predict(x_app_lda,grille)

#Affichage des lignes de décision
zp_3<-Zp[["posterior"]][,3]-pmax(Zp[["posterior"]][,1],Zp[["posterior"]][,2])

```

```

zp_2<-Zp[["posterior"]][,2]-pmax(Zp[["posterior"]][,1],Zp[["posterior"]][,3])

contour(xp1,xp2,matrix(zp_3,50),add=TRUE,levels=0,drawlabels=FALSE, col="green",
lwd = lwd_default)
contour(xp1,xp2,matrix(zp_2,50),add=TRUE,levels=0,drawlabels=FALSE, col="blue", lwd
= lwd_default)

#####
# Q1.4
#####

assigne_test <- predict(x_app_lda, newdata=x_test)
# Estimation des taux de bonnes classifications
table_classification_test_lda <-table(classe_test, assigne_test[["class"]])
# table of correct class vs. classification
diag(prop.table(table_classification_test_lda, 1))
# total percent correct
taux_bonne_classif_test_lda <-sum(diag(prop.table(table_classification_test_lda)))

# Création du vecteur contenant le code de la forme des données test assignées aux
shape<-rep(1,n_test) ;
# Forme des données assignées
shape[assigne_test[["class"]]==1]=1 ;
shape[assigne_test[["class"]]==2]=2 ;
shape[assigne_test[["class"]]==3]=3 ;
plot(x_test,col=couleur_test,pch=shape,xlab = "X1", ylab = "X2", lwd = lwd_default)

#Affichage des lignes de décision
contour(xp1,xp2,matrix(zp_3,50),add=TRUE,levels=0,drawlabels=FALSE, col="green",
lwd = lwd_default)
contour(xp1,xp2,matrix(zp_2,50),add=TRUE,levels=0,drawlabels=FALSE, col="blue", lwd
= lwd_default)

#####
# Q1.5
#####
#Affichage de l'ensemble d'apprentissage
plot(x_app, col = couleur_app, lwd = lwd_default)

x_app_qda<-qda(x_app,classe_app)

# Estimation des densités de probabilités a posteriori dans Zp
Zp <- predict(x_app_qda,grille)

#Affichage des lignes de décision
zp_3<-Zp[["posterior"]][,3]-pmax(Zp[["posterior"]][,1],Zp[["posterior"]][,2])
zp_2<-Zp[["posterior"]][,2]-pmax(Zp[["posterior"]][,1],Zp[["posterior"]][,3])
contour(xp1,xp2,matrix(zp_3,50),add=TRUE,levels=0,drawlabels=FALSE, col="green",
lwd = lwd_default)

```

```

contour(xp1,xp2,matrix(zp_2,50),add=TRUE,levels=0,drawlabels=FALSE, col="blue", lwd
= lwd_default)

assigne_test <- predict(x_app_qda, newdata=x_test)
# Estimation des taux de bonnes classifications
table_classification_test_qda <-table(classe_test, assigne_test[["class"]])
# table of correct class vs. classification
diag(prop.table(table_classification_test_qda, 1))
# total percent correct
taux_bonne_classif_test_qda <-sum(diag(prop.table(table_classification_test_qda)))

# Création du vecteur contenant le code de la forme des données test assignées aux
shape<-rep(1,n_test) ;
# Forme des données assignées
shape[assigne_test[["class"]]==1]=1 ;
shape[assigne_test[["class"]]==2]=2 ;
shape[assigne_test[["class"]]==3]=3 ;
# Affichage avec code couleur et forme adaptées
plot(x_test,col=couleur_test,pch=shape,xlab = "X1", ylab = "X2", lwd = lwd_default)

#Affichage des lignes de décision
contour(xp1,xp2,matrix(zp_3,50),add=TRUE,levels=0,drawlabels=FALSE, col="green",
lwd = lwd_default)
contour(xp1,xp2,matrix(zp_2,50),add=TRUE,levels=0,drawlabels=FALSE, col="blue", lwd
= lwd_default)

```