Classification supervisée : fiche TP1

Contents


```
knitr::opts_chunk$set(echo = TRUE, eval = FALSE, error = TRUE)
```

1 Reprise des exemples du cours

Lancer les commandes suivantes pour retrouver les résultats des exercices de la première séance de cours.

1.1 Test du Chi-deux

```
V3V1<-matrix(c(30,20,30,20,10,15,10,15),4,2,byrow=TRUE)
V3V1
chi2 = chisq.test(V3V1)
str(chi2)
chi2
1-pchisq(5.3571,3)
sum(chi2$residuals^2)
```

1. Commenter code et résultats :

1.2 Test de Fisher

```r
x <- c(4,5,7,8,9,2,3,4,6,7,8)
y <- c(rep(0,5),rep(1,6))
cbind(x,y)
lm(x~factor(y))
factor(y)
anova(lm(x~y))
SCF <- (mean(x[1:5])-mean(x))^2*5+(mean(x[6:11])-mean(x))^2*6
SCR <- sum(c((x[1:5]-mean(x[1:5]))^2,(x[6:11]-mean(x[6:11]))^2))
Fstat <- (SCF/1)/(SCR/9)
pval <- pf(Fstat,1,9,lower.tail=FALSE)
pval
Rsq <- SCF/(SCF+SCR)
Rsq
c(SCF,SCR,Fstat,pval,Rsq)
summary(lm(x~y))$r.squared
pchisq(6.585^2,df = 1, lower.tail = FALSE)
```

2. Commenter code et résultats :

2 Analyse préliminaire du jeu de données `iris`, ANOVA et MANOVA

2.1 Analyse préliminaire du jeu de données `iris`

Dans cette partie on utilisera les données `iris`.

3. Faire `data("iris")` dans R.

```r
data("iris")
```

4. Renommer les variables “Sepal.Length”, “Sepal.Width”, “Petal.Length”, “Petal.Width”,“Species” en “X1”, “X2”, “X3”, “X4”, “Y”.

```r
names(iris) <- c("X1","X2","X3","X4","Y")
```

5. Représenter graphiquement le lien entre X1 et Y :

```r
library(dplyr)
library(ggplot2)
iris %>%
  ggplot(aes(x = Y, y = X1)) +
  geom_boxplot()
```

Puis faire de même pour les autres variables :

```r
library("tidyr")
iris %>%
  gather("variable","mesure",-Y) %>%
  ggplot(aes(x = Y, y = mesure)) +
  geom_boxplot() +
  facet_wrap(~ variable, scales = "free_y")
```

Commenter :

2.2 ANOVA

Réaliser l'ANOVA de X1 en fonction de Y et obtenir le R^2 associé, faire de même pour les autres variables. A partir des p-values, indiquer si la variable Y a une influence sur l'ensemble des variables. Quelle est la variable la mieux expliquée par Y ?

Ajustement du modèle linéaire pour X1

```r
lm(X1 ~ Y, data = iris)
summary(lm(X1 ~ Y, data = iris)) # Résumé
summary(lm(X1 ~ Y, data = iris))$r.squared
```

Extension à chacune des variables

```r
sapply(names(iris)[-5],
       function(x) summary(lm(as.formula(paste(x,"~ Y")),
                              data = iris))$r.squared)
```

6. Commenter ces résultats

Calcul de l'ANOVA (calcul de la p-value du test)

```r
anova(lm(X1~Y,data=iris))
anova(lm(X1~Y,data=iris))$`Pr(>F)`
anova(lm(X1~Y,data=iris))$`Pr(>F)`[1]
```

Extension à chacune des variables

```r
sapply(names(iris)[-5],
       function(x) anova(lm(as.formula(paste(x,"~ Y")),
                            data = iris))$`Pr(>F)`[1])
```

7. Commentez ces résultats : la sous-espèce a-t'elle un effet significatif sur l'espérance de X1 ? de X2 ? de X3 ? de X4 ? Sur l'espérance de $X = \begin{pmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{pmatrix}$?

Ici vous avez testé :

$$H_{0j} = \{\mu_{1j} = \mu_{2j} = \mu_{3j}\} \text{ contre } H_{1j} = \{\exists i \neq i' | \mu_{ij} \neq \mu_{i'j}\},$$

pour $j \in \{1,2,3,4\}$, c'est-à-dire pour chacune des variables séparément.

Mais, ce que nous souhaitons tester ici est : y-a-t'il une différence entre groupes pour au moins une des variables ? :

$$H_0 = \{\mu_1 = \mu_2 = \mu_3\} \text{ contre } H_1 = \{\exists j \in \{1,2,3,4\}, \exists i \neq i' | \mu_{ij} \neq \mu_{i'j}\}$$

avec $\mu_i = \begin{pmatrix} \mu_{i1} \\ \mu_{i2} \\ \mu_{i3} \\ \mu_{i4} \end{pmatrix}$.

Comment recoller les morceaux ???

Remarquons d'abord que :

$$H_0 = \cap_{j=1}^{4} H_{0j} = H_{01} \cap H_{02} \cap H_{03} \cap H_{04}$$

Ainsi H_0 est fausse du moment qu'au moins une des H_{0j} est fausse. La question est alors quel est le risque de première espèce α_{global} de rejeter H_0 à tord quand on se donne un risque de première espèce α de rejeter H_{0j} à tord pour $j \in \{1,2,3,4\}$? Et comment choisir α de manière à maintenir un risque global α_{global} ?

On note p_j les probabilités critiques associées à chacune de H_{0j}. Sous H_{0j} on sait que p_j suit une loi uniforme sur $[0;1]$ ($p_j \sim U([0;1])$). En notant A_j l'événement H_{0j} est rejeté, $A_j = \{p_j \leq \alpha\}$.

$$P_{H_0}(\text{rejet de } H_0 \text{ à tord}) = P_{H_0}(\cup_{j=1}^{d} A_j) = P_{H_0}(\cup_{j=1}^{d} \{p_j \leq \alpha\}) \leq \sum_{j=1}^{d} P_{H_{0j}}(p_j \leq \alpha) = d \times \alpha$$

Ainsi, si on veut s'assurer que $P_{H_0}(\text{rejet de } H_0 \text{ à tord}) \leq \alpha_{global}$, on peut choisir $\alpha = \frac{\alpha_{global}}{d}$. Il s'agit de la correction de **Bonferroni** (cette correction est plutôt frustre et on peu parfois lui préférer d'autres corrections comme l'utilisation du False Discovery Rate (FDR) qui vise à controler le pourcentage de faux positifs).

8. On se donne un risque de première espèce $\alpha_{global} = 0,05$, réaliser l'ajustement de Bonferroni. Rejettez-vous H_0 ?

```
alpha_glo = 0.05
d = 4
alpha = alpha_glo/d
alpha

pvalue = sapply(names(iris)[-5],
       function(x) anova(lm(as.formula(paste(x,"~ Y")),
                            data = iris))$`Pr(>F)`[1])
pvalue
any(pvalue < alpha) # TRUE : moins une des
# p-valeurs est inférieure à 0,0125 donc on rejette H_0 au risque global alpha = 0,05 ! La distribution
```

2.3 MANOVA

Contrairement à la situation précédente on souhaite tester directement H_0 contre H_1, ce qui impose un modèle sur la distribution du vecteur X sachant la classe Y :

- $X|Y = k \sim \mathcal{N}_d(\mu_i; \Sigma_i)$: Hypothèse de normalité sachant la classe
- $\Sigma_1 = \Sigma_2 = \cdots = \Sigma_K = \Sigma$: Hypothèse d'homogénéité des variances

En utilisant la fonction `ggpairs` du package `GGally` on représente les corrélations deux à deux entre les différentes variables en fonction de la variable `Y` comme suit :

```
library(GGally)
ggpairs(iris, columns = 1:4, aes(color = Y, alpha = 0.8))
```

9. Commenter le graphique obtenus, que dire des hypothèses de normalité et d'homogénéité des variances ?

A l'aide de la fonction `mshapiro.test` de la librairie `mvnormtest` réaliser un test de normalité pour chacune des classes :

```
# install.packages("mvnormtest")
library(mvnormtest)
mshapiro.test(as.matrix(t(iris[iris$Y=="versicolor",1:4])))
mshapiro.test(as.matrix(t(iris[iris$Y=="setosa",1:4])))
mshapiro.test(as.matrix(t(iris[iris$Y=="virginica",1:4])))
```

10. Commenter :

A l'aide de la fonction contenue dans le fichier `BoxMTest.R` on réalise le test d'égalité des matrices de variances-covariances.

```
source("BoxMTest.R") # Fichier à récupérer sur moodle
BoxMTest(iris[,1:4],iris$Y)
```

11. Commenter :

A l'aide de la fonction `manova` de R tester l'égalité des espérances des groupes : `manova(cbind(X1,X2,X3,X4) ~ Y, data = iris)`

```
iris_manova = manova(cbind(X1,X2,X3,X4)~Y,data=iris)
```

Obtenir les résumés à partir de la fonction `summary` appliquée à l'objet précédent :

```
summary(_ _ _) # compléter
```

12. Commenter
13. Aller voir dans l'aide de la fonction `summary.manova` pour modifier la statistique de test utilisée

```
help("summary.manova")
```

14. Commenter les résultats obtenus :

Par la suite on va calculer les matrices W et B qui pourraient être utilisées pour récalculer les statistiques de test ci-dessus.

3 Analyse factorielle discriminante (iris de Fisher)

3.1 Calcul des matrices

15. Calculer V la matrice de variance-covariance globale, à partir de la fonction `cov.wt` en utilisant l'option `method = "ML"`. Expliquer à quoi sert cette option.

Attention on prendra garde de récupérer le bon élément de sortie de la fonction `cov.wt`, fonction qui ressort une liste contenant entre autres `cov`, `center`, ...

On calcule les vecteurs des moyennes pour chaque groupe $\bar{X}_i$ en s'aidant de la fonction `by`, et en restructurant le résultat sous forme d'un tableau.

Constituez la matrice G de centres des classes composée d'une colonne par variable et d'une ligne par sous-espèce (on rappelle que la fonction `t` permet de transposer un tableau)

```
by(iris[,1:4],iris$Y, colMeans)
simplify2array(by(iris[,1:4],iris$Y, colMeans))
G = _ _ _
```

16. En déduire :

$$B = \sum_{i=1}^{K} \frac{n_i}{n} (\bar{X}_i - \bar{X})(\bar{X}_i - \bar{X})^T$$

où $\bar{X}_i$, $\bar{X}$ sont respectivement les vecteurs colonnes des moyennes intra-classes et de la moyenne globale. Pour cela on pourra remarquer que B est la matrice de covariance des centres des classes pondérés par leurs effectifs (penser à `cov.wt` et à son argument `wt`).

```
B = cov.wt(_ _ _, _ _ _ , _ _ _)
```

17. Vérifier qu'on retrouve bien : $V = W + B$

```
# Proposer un indicateur synthétique du fait que V = W + B
```

3.2 Réalisation de l'AFD

On rappelle que dans R l'ACP peut se réaliser à la main comme suit :

```
eigen(V) # Decomposition en valeurs propres
eigen(V)$values
ACP=eigen(V)$vectors
c=as.matrix(iris[,1:4])%*%ACP[,1:2]
plot(c,col=iris$Y)

c = as.data.frame(c)
names(c) <- c("C1","C2")
c %>% mutate(Y = iris$Y) %>%
  ggplot(aes(x = C1, y = C2, color = Y, shape = Y)) +
  geom_point()
```

18. Commenter, quel est le pourcentage d'inertie expliqué par chacun des axes ? Par les deux premiers axes

19. Calculer les coordonnées d_1 et d_2 des points projetés sur les deux premières composantes discriminantes, sachant qu'en AFD on diagonalise la matrice $V^{-1}B$. Adapter le code pour réaliser l'AFD, et commenter les résultats (on rappelle que l'inverse s'obtient avec la fonction `solve` et le produit matriciel avec l'opérateur `%*%`) :

Quels est la part de variance de d_1 expliquée par la classe ? De d_2

20. Reprendre le code précédent en remplaçant $V^{-1}B$ par $W^{-1}B$.

Que dire ? Quel est le lien entre les différents vecteurs propres et valeurs propres ?

21. Comparer les résultats obtenus à ceux obtenus en ACP.

3.3 Calcul des scores discriminants

On souhaite calculer les fonctions de score pour chacun des groupes, ces fonctions nous serviront ensuite à affecter chaque individu au groupe de plus grand score (équivalent à la minimisation de la distance de Mahalanobis).

On rappelle que le calcul des fonctions de score pour chaque groupe s'effectue comme suit :

$$s_i(x) = \alpha_{i0} + \alpha_{i1}x_1 + \alpha_{i2}x_2 + \alpha_{i3}x_3 + \alpha_{i4}x_4$$

avec $\alpha_{i0} = -\bar{X}_i^T W^{-1} \bar{X}_i$ et

$$\begin{pmatrix} \alpha_{i1} \\ \vdots \\ \alpha_{ip} \end{pmatrix} = 2W^{-1}\bar{X}_i$$

22. Construire le tableau des coefficients :

	Setosa	Versicolor	Virginica
Constante	α_{10}	α_{20}	α_{30}
X_1	α_{11}	α_{21}	α_{31}
X_2	α_{12}	α_{22}	α_{32}
X_3	α_{13}	α_{23}	α_{33}
X_4	α_{14}	α_{24}	α_{34}

Aide : Dans l'AFD, la notion de score est liée au calcul de la règle de décision. Une observation $x = (x_1, x_2, x_3, x_p)$ sera affectée au groupe avec le score $s_i(x)$ maximal.

Rappel :

$$\hat{y} = \arg\min_i (x - \bar{X}_i)^T W^{-1} (x - \bar{X}_i)$$

Ce calcul revient à maximiser $2x^T W^{-1}\bar{X}_i - \bar{X}_i^T W^{-1}\bar{X}_i$.

23. Calculer les scores des individus à partir de cette règle (simple calcul matriciel, on pourra rajouter une colonne de 1 à la matrice des données à l'aide de la fonction `cbind`)

24. En déduire le classement de chacun des individus à partir de ces scores (en utilisant de façon appropriée les fonctions `apply` et `which.max`) :