Université de Montpellier Année 2019-2020
HMMA303

**Examen de contrôle continu**
**Durée 1h30 - Sans document - 16 octobre 2019**

**Exercice 1 (2 pts)** On se place dans le contexte de la classification supervisée binaire. Décrire l'ensemble des éléments contenus dans une matrice de confusion.

**Exercice 2 (8 pts)** Soit $Y$ un facteur binaire à expliquer, $Y \in \{0,1\}$ et $X \in [0,1]$ une variable explicative réelle. Nous supposons que $\mathbb{P}(Y=1) = 1/2$ et

$$X|Y=0 \sim f_0(x) = \mathbb{I}_{[0,1]}(x)$$

$$X|Y=1 \sim f_1(x) = 2x\mathbb{I}_{[0,1]}(x)\,.$$

**1) (2 pts)** Donner la loi de $Y|X=x$.

**2) (2 pts)** Donner le classifieur optimal $g^*$ lorsque

$$h(y,d) = \begin{cases} 0 & \text{si} \quad y=d \\ 10 & \text{si} \quad y=1 \quad \text{et} \quad d=0 \\ 1 & \text{si} \quad y=0 \quad \text{et} \quad d=1 \end{cases}.$$

**3) (2 pts)** Calculer le taux d'erreur moyen associé à $g^*$.

**4) (2 pts)** Calculer le taux d'erreur moyen du classifieur affectant au hasard un dixième des individus à la 0.

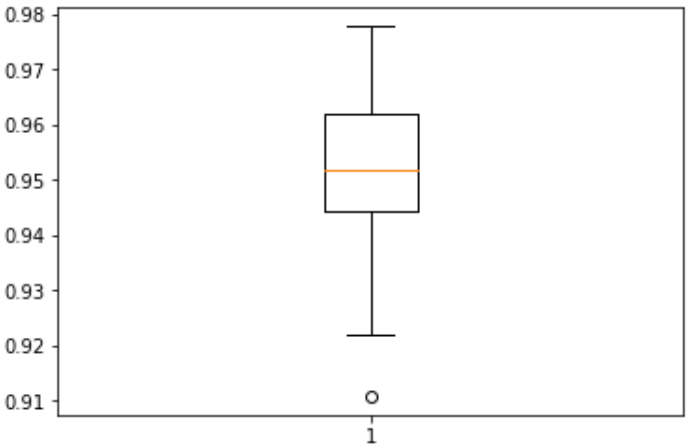
**Exercice 3 (4 pts)** Expliquer en détails le résultat produit par le code `R` donné ci-dessous.

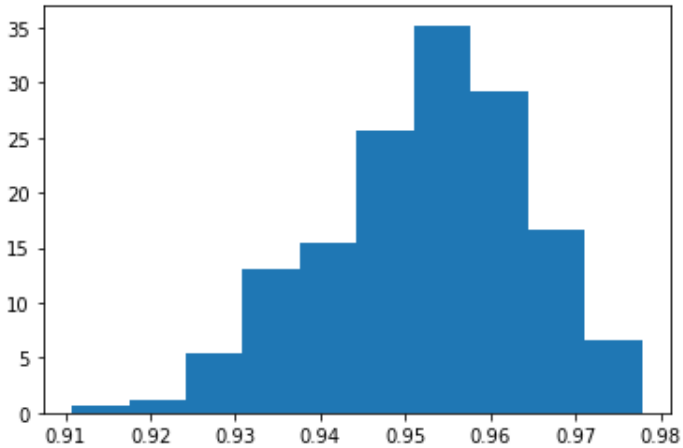
```
data(iris)
n <- 100
x <- scale(iris[1:n,1:2])
y <- as.numeric(iris[1:n,5])-1
fisher <- data.frame(y=as.factor(y),x1=x[,1],x2=x[,2])
len <- 10
x1 <- seq(min(x[,1]),max(x[,1]),length=len)
x2 <- seq(min(x[,2]),max(x[,2]),length=len)
grille <- expand.grid(x1=x1,x2=x2)
library(MASS)
model1 <- qda(y~x1+x2,data=fisher)
pred1 <- predict(model1,grille)$class
library(caret)
model2 <- knn3(y ~ .,data=fisher, k=5)
pred2 <- predict(model2,grille,type="class")
table(pred1,pred2)
# pred2
# pred1  0  1
#     0 43  2
#     1  4 51
```

**Exercice 4 (6 pts)** Expliquer en détails le résultat produit par le code `Python` donné ci-dessous.

```
import matplotlib.pyplot as plt
import pandas as pd

from sklearn import datasets

digits = datasets.load_digits()
X=digits.data
y=digits.target

from sklearn.model_selection import train_test_split
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import RepeatedKFold

X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.25,random_state=14)
FL = RepeatedKFold(n_splits=5, n_repeats=50)
scores = cross_val_score(LinearDiscriminantAnalysis(), X_train, y_train, cv = FL)
scores.shape
# (250,)
scores.mean()
# 0.9524574418284456
plt.boxplot(scores)
plt.hist(scores,normed=True)

model1 = LinearDiscriminantAnalysis()
model1.fit(X_train, y_train)
model1.score(X_test,y_test)
# 0.9488888888888889

print(pd.crosstab(y_test,model1.predict(X_test)))
# col_0   0   1   2   3   4   5   6   7   8   9
# row_0
# 0      43   0   0   0   0   0   0   0   0   0
# 1       0  32   0   0   0   0   0   0   0   1
# 2       0   2  35   0   0   0   0   0   0   0
# 3       0   0   1  39   0   1   0   0   1   0
# 4       0   0   0   0  43   0   0   0   1   1
# 5       0   0   0   0   0  39   0   0   0   2
# 6       0   0   0   0   0   0  47   0   0   0
# 7       0   0   0   0   0   0   0  48   1   2
# 8       0   4   0   0   0   0   0   0  42   2
# 9       0   1   0   1   0   0   0   0   2  59

from sklearn.model_selection import GridSearchCV
from sklearn.neighbors import KNeighborsClassifier

list(range(1,21))
# [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20]
```

```
param = [{"n_neighbors":list(range(1,21))}]
model2 = GridSearchCV(KNeighborsClassifier(),param,cv=5,n_jobs=-1)
model2.fit(X_train, y_train)
model2.best_params_
# {'n_neighbors': 1}
model2.best_score_
# 0.9851521900519673
model2.score(X_test,y_test)
# 0.9866666666666667

print(pd.crosstab(y_test,model2.predict(X_test)))
# col_0   0   1   2   3   4   5   6   7   8   9
# row_0
# 0      43   0   0   0   0   0   0   0   0   0
# 1       0  33   0   0   0   0   0   0   0   0
# 2       0   0  37   0   0   0   0   0   0   0
# 3       0   0   0  42   0   0   0   0   0   0
# 4       0   0   0   0  45   0   0   0   0   0
# 5       0   0   0   0   0  41   0   0   0   0
# 6       0   0   0   0   0   0  47   0   0   0
# 7       0   0   0   0   0   0   0  51   0   0
# 8       0   2   0   0   0   0   0   0  46   0
# 9       0   1   0   2   0   1   0   0   0  59
```



# Correction examen ML_controle_continu 16 octobre 2019 HMMA 303

## Exercice 1

Voir cours

## Exercice 2

1) $\mathbb{P}(Y=1|X=x)$

$$= \frac{2x}{1+2x}$$ pour tout $x \in [0,1]$

$$\mathbb{P}(Y=0|X=x) = \frac{1}{1+2x}$$

pour tout $x \in [0,1]$

2]

$$g^*(x) = \begin{cases} 0 & \text{si } \mathbb{P}(Y=0|X=x) \geq \frac{10}{11} \\ 1 & \text{si } \mathbb{P}(Y=0|X=x) < \frac{10}{11} \end{cases}$$

3]

$$\mathbb{P}(Y \neq g^*(X))$$

$$= \mathbb{P}(Y=1 \wedge g^*(x)=0) + \mathbb{P}(Y=0 \wedge g^*(x)=1)$$

$$\mathbb{P}(Y=0|X=x) \geq \frac{10}{11}$$

$$\Leftrightarrow \frac{1}{1+2x} \geq \frac{10}{11}$$

$$\Leftrightarrow 20x \leq 1$$

$$\Leftrightarrow x \leq 1/20$$

donc $g^*(x) = 0$ si $x \leq 1/20$

$$\mathbb{P}(Y \neq g^*(X)) = \mathbb{P}(Y=1 \wedge X \leq \tfrac{1}{20}) + \mathbb{P}(Y=0 \wedge X > \tfrac{1}{20})$$

(3)

$$\mathbb{P}(Y \neq g^*(X)) = \mathbb{P}(Y=1)\,\mathbb{P}(X \leq \tfrac{1}{20} \mid Y=1) + \mathbb{P}(Y=0)\,\mathbb{P}(X > \tfrac{1}{20} \mid Y=0)$$

$$\Leftrightarrow \mathbb{P}(Y \neq g^*(X)) = \frac{1}{2}\int_0^{1/20} 2x\,dx + \frac{1}{2}\int_{1/20}^{1} dx$$

$$\Leftrightarrow \mathbb{P}(Y \neq g^*(X)) = \left(\frac{1}{2}\right)\left[x^2\right]_0^{1/20} + \left(\frac{1}{2}\right)\left[x\right]_{1/20}^{1}$$

$$\Leftrightarrow \mathbb{P}(Y \neq g^*(X)) = \frac{1}{800} + \frac{1}{2}\left(1 - \frac{1}{20}\right)$$

$$\Leftrightarrow \mathbb{P}(Y \neq g^*(X)) = \frac{1}{800} + \frac{19}{40} = \frac{381}{800} \# 0{,}4$$

4) $g^{\#}(x) = \begin{cases} 0 & \text{si } U \leq 1/10 \\ 1 & \text{sinon} \end{cases}$

où $U \sim \mathcal{U}_{[0,1]}$ et $U \perp\!\!\!\perp X$, $U \perp\!\!\!\perp Y$

$$\mathbb{P}(Y \neq g^{\#}(X)) = \left(\frac{1}{2}\right)\mathbb{P}(U \leq \tfrac{1}{10}) + \left(\frac{1}{2}\right)\mathbb{P}(U > \tfrac{1}{10}) = \frac{1}{2}$$

Remarque : on bien, si on compare les tous n'avons il serait plus pertinent de comparer les coût moyen :

$$C(g^*) = \mathbb{E}\left[h(Y, g^*(X))\right]$$

$$C(g^*) = 10\,\mathbb{P}(Y=1 \cap g^*(X)=0) + \mathbb{P}(Y=0 \cap g^*(X)=1)$$

$$C(g^*) = 10 \left(\frac{1}{2}\right) \int_0^{1/20} 2x\,dx + \left(\frac{1}{2}\right) \int_{1/20}^{1} dx$$

$$C(g^*) = 5\,\frac{1}{400} + \left(\frac{1}{2}\right)\left(\frac{19}{20}\right)$$

$$C(g^*) = \frac{1}{80} + \frac{19}{40} = \frac{39}{80} \# 0{,}49$$

$$C(g^\#) = 10 \left(\frac{1}{2}\right)\frac{1}{10} + \left(\frac{1}{2}\right)\frac{9}{10}$$

$$C(g^\#) = \left(\frac{1}{2}\right)\left[1 + \frac{9}{10}\right] = \frac{19}{20} = 0.95$$

La différence est beaucoup plus significative.

## Exercice 3

(5)

On compare les classifications produites par 2 classifieurs :

- une méthode des k plus proches voisins avec un nombre de voisins correspondant à la valeur par défaut de la fonction knn3 de la bibliothèque caret (ici prendra $k = 5$)
- une stratégie d'analyse discriminante quadratique

On effectue cette comparaison sur les données iris de Fisher avec deux variables explicatives quantitatives et une variable à expliquer binaire. La taille de l'ensemble d'apprentissage est de 100 données.

# Exercici 4