Université de Carthage
Ecole Supérieure de la Statistique et de l'Analyse de l'Information

Examen de Data Mining

2 ème année du cycle de formation d'ingénieurs

Durée de l'épreuve : 1 heure 30 - Documents non autorisés
Nombre de pages : 4 - Date de l'épreuve : 16 mai 2023

Exercice 1 : On considère le data frame contenant 3 variables $X1$, $X2$ et $X3$, et d'une variable Y possédant les deux modalités 0 et 1, sur un échantillon I de 20 individus. Par la suite, on cherche à expliquer Y en fonction de $X1$, $X2$ et $X3$ par un modèle de régression logistique.
Pour cela, on utilise les commandes du logiciel R. Le dataframe ainsi que les résultats de cette régression logistique sont donnés à **l'Annexe I.**

1- Commenter les résultats des différents tests.

2- Donner la classe d'affectation de l'individu ayant les caractéristiques suivantes : $X1 = 50; X2 = 126; X3 = 1$. Calculer $P(Y=1/x)$

3- Déterminer la classe d'affectation de chacun des individus et en déduire le taux de bien classés.

4- Calculer les odds-ratio de chacune des variables puis interpréter les.

5- Compléter le code R suivant afin d'effectuer une sélection pas à pas forward.

```
modele_simple <- glm(..., "binomial")
> pr.f.step<-step(..., scope = ~ ..., dir=...)
```

Exercice 2 : On considère le jeu de données des Iris de Fisher dont les statistiques descriptives sont données ci-dessous :

```
Sepal.Length    Sepal.Width     Petal.Length    Petal.Width
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300
 Median :5.800   Median :3.000   Median :4.350   Median :1.300
 Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
       Species
 setosa    :50
 versicolor:50
 virginica :50
```

On voudrait expliquer, à l'aide d'un arbre de décision sous Python, la variable `Species`, que l'on note `Y`. par les quatres autres variables du jeu de données et qui constituent le tableau des variables explicatives que l'on notera dans la suite `X`.

On considère l'arbre donné à l'Annexe II.

1- A partir de cet arbre, donner les règles qui mènent à une classification dans la classe des 'virginica'.

On voudrait optimiser cet arbre à l'aide de la fonction `GridSearchCV` de Python.

2- Compléter le code suivant en remplaçant chacune des 14 lettres par l'expression, la ou les valeurs adéquates.

```
### Début du code Python  ###
X_train, X_test, Y_train, Y_test = train_test_split(.(a)., .(b)., test_size=.(c).,
random_state=.(d).)

clf = DecisionTreeClassifier()
params = {
    'ccp_alpha' : [.(e).]
    'max_depth': [.(f).],
    'min_samples_split': [.(g).]
    'criterion': [.(h).]
}

grid_search = GridSearchCV(.(i)., param_grid=.(j)., cv=.(k).)
.(l)..fit(.(m).)
Y_pred = grid_search.predict(X_test)
accuracy_clf = accuracy_score(.(n).)

### Fin du code Python  ###
```

Annexe I : Résultats de la régression logistique - Exercice 1

```
>  df
    X1  X2 X3 Y
P1  50 126  1 1
P2  49 126  0 1
P3  46 144  0 1
P4  49 139  0 1
P5  62 154  1 1
P6  35 156  1 1
P7  67 160  0 0
P8  65 140  0 0
P9  47 143  0 0
P10 58 165  0 0
P11 57 115  1 0
P12 59 145  0 0
P13 44 175  0 0
P14 41 153  0 0
P15 54 152  0 0
P16 52 169  0 0
P17 57 168  1 0
P18 50 158  0 0
P19 44 170  0 0
P20 49 171  0 0


> modelel <- glm(Y~ X1+X2+X3, family=binomial,df)

> summary(modelel)

Call:
glm(formula = Y ~ X1 + X2 + X3, family = binomial, data = df)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.9773  -0.5437  -0.3876   0.5093   1.7577

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) 14.49379    7.95464   1.822   0.0684 .
X1          -0.12563    0.09380  -1.339   0.1805
X2          -0.06356    0.04045  -1.572   0.1161
X31          1.77901    1.50449   1.182   0.2370
---

> anova(modelel,test='Chisq')

     Df Deviance Resid. Df Resid. Dev Pr(>Chi)
NULL                    19     24.435
X1    1   1.4896        18     22.945  0.22228
X2    1   4.7943        17     18.151  0.02855
```

```
X3     1   1.5330         16      16.618  0.21566
---

> predict(modele1, don, type="response")
        P1         P2         P3         P4         P5         P6
0.87894733 0.58154537 0.39220275 0.37820752 0.21335852 0.87655486
        P7         P8         P9        P10        P11        P12
0.01640958 0.07103688 0.37750865 0.03624840 0.85841939 0.10575388
       P13        P14        P15        P16        P17        P18
0.10366373 0.40566043 0.12437705 0.05836647 0.17271990 0.13818549
       P19        P20
0.13712678 0.07370700
```

Annexe II : Arbre de décision - Exercice 2