Université de Montpellier
HAX912X - Modèles linéaires généralisés

Année 2023-2024

Examen final

Durée 2h - Sans document
Vendredi 27 octobre 2023

Exercice 1 (4 pts)

Décrire et commenter les résultats obtenus à l'aide du code Python ci-dessous.
Les données considérées sont les résultats d'une analyse chimique de vins cultivés dans la même région d'Italie par trois cultivateurs différents. Treize mesures ont été effectuées pour différents constituants présents dans les trois types de vin.

```
import numpy as np
import sklearn.linear_model as lm
from sklearn import preprocessing
from sklearn import datasets
dataset = datasets.load_wine()
X = dataset.data
X.shape
(178, 13)

y = dataset.target
np.unique(y)
array([0, 1, 2])

X = X[y != 0]
y = y[y != 0]
X.shape
(119, 13)

np.unique(y)
array([1, 2])

scaler = preprocessing.StandardScaler().fit(X)
X = scaler.transform(X)
X.mean(axis=0)
array([ 1.41810000e-15,  4.73943947e-16,  6.67253367e-15,  3.69452368e-16,
        1.20293598e-16, -2.21811365e-16, -4.09569670e-16, -6.69865657e-16,
       -2.00120033e-16, -4.31027763e-16,  7.57563946e-16, -9.19899078e-16,
        2.78721957e-17])

X.var(axis=0)
array([1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.])
```

```
model1 = lm.LogisticRegression()
model1.fit(X,y)
model1.penalty
'l2'
model1.C
1.0
model1.intercept_
array([-1.93238543])
model1.coef_
array([[ 0.67061806,  0.54717719,  0.63550791,  0.24868629,  0.16614675,
        -0.20010196, -1.34821171, -0.09664867, -0.56443121,  1.26159206,
        -1.16294376, -0.99888229,  0.33789776]])

from sklearn.model_selection import cross_val_score
from sklearn.model_selection import RepeatedKFold
Sel = RepeatedKFold(n_splits=5, n_repeats=100)
scores = cross_val_score(lm.LogisticRegression(), X, y, cv=Sel)
scores.shape
(500,)
scores.mean()
0.9775869565217391

from sklearn.model_selection import GridSearchCV
parameters = {'C':np.logspace(-4, 4, 50),'penalty':['l1','l2']}
model2 = GridSearchCV(lm.LogisticRegression(), parameters,cv=Sel)

model2.fit(X,y)
model2.best_params_
{'C': 0.02811768697974228, 'penalty': 'l2'}

model2.best_score_
0.9889710144927536
```

Exercice 2 (6 pts)

Décrire et commenter les résultats obtenus à l'aide du code Python ci-dessous. Les données considérées sont décrites en annexe.

```
import os
import numpy as np

import pandas as pd
from sklearn.linear_model import LogisticRegression
from sklearn import preprocessing
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from math import sqrt
from sklearn.metrics import classification_report
```

```
path = os.getcwd()
os.chdir('/Users/marin/TEACHING/2324/M2-GLM-HAX912X/EXAMENS')

data = pd.read_csv('Breast-Cancer-Wisconsin.csv')
y = data['diagnosis']
X = data.drop(['id','diagnosis','Unnamed: 32'], axis = 1)
X.head()
   radius_mean  texture_mean  ...  symmetry_worst  fractal_dimension_worst
0        17.99         10.38  ...          0.4601                  0.11890
1        20.57         17.77  ...          0.2750                  0.08902
2        19.69         21.25  ...          0.3613                  0.08758
3        11.42         20.38  ...          0.6638                  0.17300
4        20.29         14.34  ...          0.2364                  0.07678
[5 rows x 30 columns]

pd.value_counts(y, sort = True).sort_index()

B    357
M    212
Name: diagnosis, dtype: int64

scaler = preprocessing.StandardScaler().fit(X)
X = scaler.transform(X)

X_train, X_test, y_train, y_test = train_test_split(X,y,test_size = 0.2,
random_state = 2023)

pd.value_counts(y_train, sort = True).sort_index()
B    286
M    169
Name: diagnosis, dtype: int64

pd.value_counts(y_test, sort = True).sort_index()
B    71
M    43
Name: diagnosis, dtype: int64

forest = RandomForestClassifier(n_estimators=500, max_features=int(sqrt(30)),
oob_score=True,n_jobs=-1)
forest = forest.fit(X_train,y_train)
y_chap = forest.predict(X_test)
print(classification_report(y_test,y_chap))
              precision    recall  f1-score   support

           B       0.95      0.99      0.97        71
           M       0.97      0.91      0.94        43

    accuracy                           0.96       114
   macro avg       0.96      0.95      0.95       114
```

```
weighted avg       0.96      0.96      0.96       114

forest.oob_score_
0.9604395604395605

forest.oob_decision_function_
array([[0.        , 1.        ],
       [1.        , 0.        ],
       [0.05747126, 0.94252874],
       [0.        , 1.        ],
       [1.        , 0.        ],
       [0.99456522, 0.00543478],
...
       [1.        , 0.        ],
       [0.00534759, 0.99465241],
       [0.96907216, 0.03092784],

...
       [1.        , 0.        ],
       [1.        , 0.        ],
       [0.98809524, 0.01190476],
       [0.99450549, 0.00549451],
       [0.9893617 , 0.0106383 ],
       [0.12429379, 0.87570621],
       [0.56666667, 0.43333333],
       [0.        , 1.        ],
       [1.        , 0.        ],
       [0.87434555, 0.12565445],
       [1.        , 0.        ],
       [0.34104046, 0.65895954],
       [1.        , 0.        ],
       [1.        , 0.        ],
       [0.        , 1.        ],
       [0.9902439 , 0.0097561 ],
       [1.        , 0.        ],
       [1.        , 0.        ],
       [0.98969072, 0.01030928],
       [0.97354497, 0.02645503],
       [0.95854922, 0.04145078]])
```

Exercice 3 (8 pts)

On considère un n-échantillons $(y_1, x_1), \ldots, (y_n, x_n)$ du couple (y, x) où y est une variable aléatoire à valeurs dans $\{0, 1\}$ et x est une variable fixée prenant également les valeurs $\{0, 1\}$. On suppose qu'il existe une variable latente y^* telle que

$$f(y^*|x) = \frac{\exp(-(y^* - \beta_0 - \beta_1 x))}{(1 + \exp(-(y^* - \beta_0 - \beta_1 x)))^2}$$

et

$$y = \begin{cases} 1 & \text{si} \quad y^* \leq 0 \\ 0 & \text{si} \quad y^* > 0 \end{cases}.$$

1 (3 pts) Montrer qu'il s'agit d'un modèle linéaire généralisée et l'expliciter.
Indication : la primitive de $\frac{\exp(-(y-\mu))}{(1+\exp(-(y-\mu)))^2}$ est $\frac{1}{1+\exp(-(y-\mu))}$.

2 (2 pts) Est-ce le lien canonique qui a été choisi ?

3 (3 pts) Nous disposons de 100 observations (y_i, x_i) décrites dans la table de contingence suivante

	$y=0$	$y=1$
$x=0$	30	20
$x=1$	20	30

.

Donner la log-vraisemblance en fonction de β_0 et β_1.

Exercice 4 (2 pts)

Décrire la méthode de construction d'arbre de régression.

Annexe

Breast cancer data (Données sur le cancer du sein)

```
About Dataset

Features are computed from a digitized image of a fine needle aspirate (FNA)
of a breast mass. They describe characteristics of the cell nuclei present
in the image. n the 3-dimensional space is that described in:
[K. P. Bennett and O. L. Mangasarian: "Robust Linear Programming Discrimination
of Two Linearly Inseparable Sets", Optimization Methods and Software 1, 1992, 23-
34].

Attribute Information:

1) ID number
2) Diagnosis (M = malignant, B = benign)

3-32)

Ten real-valued features are computed for each cell nucleus:

a) radius (mean of distances from center to points on the perimeter)
b) texture (standard deviation of gray-scale values)
c) perimeter
d) area
e) smoothness (local variation in radius lengths)
f) compactness (perimeter^2 / area - 1.0)
```

g) concavity (severity of concave portions of the contour)
h) concave points (number of concave portions of the contour)
i) symmetry
j) fractal dimension ("coastline approximation" - 1)

The mean, standard error and "worst" or largest (mean of the three largest values) of these features were computed for each image, resulting in 30 features. For instance, field 3 is Mean Radius, field 13 is Radius SE, field 23 is Worst Radius.

All feature values are recoded with four significant digits.

Missing attribute values: none

Class distribution: 357 benign, 212 malignant

Correction Examen Final
27 octobre 2023
HAX912X

(1)

Exercice 1

4 points répartis uniformément

1) Importation des données avec un final (après suppression de certains individus)

119 individus
13 variables contenant une variable à expliquer y
1 variable binaire
les 12 variables explicatives sont continues réelles

(1 pt)

2] Modèle de régression logistique avec paramètres par défaut et mis en jeux résultats 12 et contraints de paralléliser sur C = 1 tous les coefficients estimés sont non nuls — (2) (1 pt)

3] Estimation de l'erreur de bonne classification par validation croisée à 5 ensembles répétée 100 fois résultat ≈ 0,98 % (moyenne des 500 estimations) (1 pt)

4] Choix de la meilleure méthode de régularisation entre Lasso (L1) et Ridge (L2) pour 50 valeurs de C entre e^{-4} et e^{4} uniformément réparties en échelle log —

Meilleur modèle
mélange avec C # 0,03
Tous ils bien classé environ 0,99%
Lorsque un échantillon fourni
non encore utilisé pour entraîner
ce Tous correctement -

Exercice 2

3 points devaient être mentionnés

1) les données des cellules d'une
image d'un tissu issu d'un
patient cancéreux sont analysées -
L'analyse statistique de la
cellule, il y en 569 dont
212 cancéreuses - Objectif construire
une fonction capable de prédire
si une cellule est cancéreuse ou non
à partir de 30 variables descriptives

Les 30 variables explicatives sont centrées réduites. Les classes sont séparées en deux échantillons : un échantillon d'apprentissage contient 455 individus et un échantillon test contient 114 individus. (4) (2 pts)

2) On classifie par forêt aléatoire et entraîné sur l'échantillon d'apprentissage. Il contient 500 arbres et à chaque nœud le meilleur variable est choisi parmi un sous-ensemble tiré aléatoirement de $\sqrt{30}$ variables. (2 pts)

Tous les bonnes classifications sur l'échantillon test est 96% -

3] Avec les forêts aléatoires, on peut déterminer une échantillon test en réalisant un croisé la qualité d'un classifieur. Dans ce cas, il suffit d'utiliser l'estimateur out-of-bag sur l'échantillon d'apprentissage - Dans ce cas, la taux de bonnes classifications est estimé à 96% - Idem à l'estimation obtenue sur l'échantillon de test. (5) (2 pts)

Exercice 3

1] $y \sim B(p(x))$

La loi de Bernoulli appartient à la famille exponentielle canon.

$$f(y|p) = p^{y}(1-p)^{1-y}\,\mathbb{1}_{\{0,1\}}(y)\,\nu(dy)$$

$$f(y|p) = \exp\left(y \log\left(\frac{p}{1-p}\right) + \log(1-p)\right)\nu(dy)$$

$$\theta = \log\left(\frac{p}{1-p}\right) \Rightarrow p = \frac{e^{\theta}}{1+e^{\theta}}$$

$$b(\theta) = -\log\left(\frac{1}{1+e^{\theta}}\right) = \log\left(1+e^{\theta}\right)$$

Sur ailleurs, $p(x) = \mathbb{P}(y=1)$

$$p(x) = \mathbb{P}(y^* \leq 0)$$

$$p(x) = \int_{-\infty}^{0} \frac{\exp\left(-(y^* - \beta_0 - \beta_1 x)\right)}{\left(1+\exp\left(-(y^* - \beta_0 - \beta_1 x)\right)\right)^2}\,\nu(dy^*)$$

$$\Leftrightarrow p(x) = \left[\frac{1}{1+\exp\left(-(y^* - \beta_0 - \beta_1 x)\right)}\right]_{-\infty}^{0}$$

$$\Leftrightarrow p(x) = \frac{1}{1+e^{\beta_0+\beta_1 x}} = \left[\frac{e^{-\beta_0-\beta_1 x}}{1+e^{-\beta_0-\beta_1 x}}\right]$$

$$\mathbb{E}(y) = f'(\theta) = \frac{e^{\theta}}{1+e^{\theta}} = p$$

$$\mathbb{E}(y) = \frac{1}{1+e^{\beta_0+\beta_1 x}} = f^{-1}(\beta_0+\beta_1 x)$$

Il s'agit bien d'un modèle linéaire généralisé.

2) Nous avons

$$\theta = \log\left(\frac{1}{1+e^{\beta_0+\beta_1 x}} \times \frac{1+e^{\beta_0+\beta_1 x}}{e^{\beta_0+\beta_1 x}}\right)$$

$$\theta = -\beta_0-\beta_1 x + \beta_0+\beta_1 x$$

Les modèles si l'on considère les paramètres $\tilde{\beta}_0 = -\beta_0$ et $\tilde{\beta}_1 = -\beta_1$

alors $\theta = \tilde{\beta}_0 + \tilde{\beta}_1 x$

Donc formellement ça n'est pas le lien canonique qui a été utilisé mais bien $(\hat{\beta}_0, \hat{\beta}_1)$ c'est bien lui.

En termes mathématiques, c'est noté.
En termes de maximisation, c'est noté.

3] $$\mathcal{L}(\beta_0,\beta_1) = -\sum_{i=1}^{100} y_i \log\left(1+e^{\beta_0+\beta_0 x_i}\right)$$

$$+\sum_{i=1}^{100}(1-y_i)\log\left(\frac{e^{\beta_0+\beta_1 x_i}}{1+e^{\beta_0+\beta_1 x_i}}\right)$$

$$\Leftrightarrow \mathcal{L}(\beta_0,\beta_1) = -\sum_{i=1}^{100} y_i(\beta_0+\beta_1 x_i)$$

$$+\sum_{i=1}^{100}\log\left(\frac{e^{\beta_0+\beta_1 x_i}}{1+e^{\beta_0+\beta_1 x_i}}\right)$$

$$\Leftrightarrow \mathcal{L}(\beta_0,\beta_1) = -50\beta_0 - 30\beta_1$$

$$+100\beta_0 + 50\beta_1 - \sum_{i=1}^{100}\log\left(1+e^{\beta_0+\beta_1 x_i}\right)$$

$$\Leftrightarrow \mathcal{L}(\beta_0,\beta_1) = 50\beta_0 + 20\beta_1 - 50\log\left(1+e^{\beta_0}\right)$$

$$-50\log\left(1+e^{\beta_0+\beta_1}\right)$$

Exercice 4

Pour construire un arbre de classification nous déterminons à chaque nœud l'arbre le meilleur couple (variable, seuil) en utilisant le critère des mauvais classés. Le critère est pondéré par les nombres d'individus dans les familles de droite et familles de gauche en sortie du nœud.