Travaux pratiques de
Machine learning
Cycle pluridisciplinaire d'études supérieures
Université Paris sciences et lettres

Joon Kwon

vendredi 14 avril 2023

On considère un jeu de données médicales. Chaque exemple correspond à une tumeur du sein. Les variables explicatives portent sur des caractéristiques observées de la tumeur. La variable à prédire indique s'il s'agit d'une tumeur maligne (0) ou bénigne (1). Il s'agit donc d'un problème de classification binaire.

```
import numpy as np
import matplotlib.pyplot as plt

from sklearn.datasets import load_breast_cancer
data = load_breast_cancer()
X = data.data
y = data.target
```

On pourra consulter une description plus détaillée du jeu de données à l'aide de la commande suivante.

```
print(data.DESCR)
```

Question 1. — Combien y a-t-il d'exemples et de variables explicatives ?

Question 2. — Séparer le jeu de données en un échantillon d'apprentissage (qui sera également utilisé pour de la validation croisée), et un échantillon de test.

Question 3. — Entraîner un Perceptron sur l'échantillon d'apprentissage et observer son score sur l'échantillon de test.

Question 4. — Trouver, pour l'algorithme kNN, la meilleure valeur de l'hyperparamètre k par validation croisée. On tracera les courbes de validation correspondantes. Pour la meilleure valeur de k trouvée, entraîner l'algorithme sur l'échantillon d'apprentissage et observer son score sur l'échantillon de test.

On souhaite à présent entraîner des régressions logistiques avec pénalisation LASSO. Dans `scikit-learn`, le coefficient de régularisation de la régression logistique s'appelle `C` et correspond à l'inverse du paramètre λ. Autrement dit, plus `C` est grand, plus la régularisation est faible. Le type de régularisation, en l'occurrence LASSO, se spécifie par l'argument `penalty='l1'`. Enfin, plusieurs méthodes de calcul sont disponibles, nous allons choisir `'liblinear'`. Voici un exemple.

```
from sklearn.linear_model import LogisticRegression
logreg =
 ↪  LogisticRegression(C=.1,penalty='l1',solver='liblinear')
logreg.fit(X_train,y_train)
print(logreg.score(X_test,y_test))
```

Question 5. — Comprendre le fonctionnement de la fonction `np.logspace` en consultant sa documentation. L'utiliser pour créer un `array` contenant un ensemble de valeurs à essayer pour le paramètre de régularisation `C`.

Question 6. — Parmi les valeurs envisagées pour `C`, choisir celle qui semble la meilleure par validation croisée. On tracera les courbes de validation correspondantes. Avec la meilleure valeur de `C` trouvée, entraîner l'algorithme sur l'échantillon d'apprentissage et observer son score sur l'échantillon de test.