Université Paris Descartes
UFR de Mathématiques et Informatique

PARIS DESCARTES

Classification
Examen du 6/1/2016

Préalables : Ouvrir Rstudio, puis entrer les commandes suivantes :

```
library(rpart)
library(MASS)
```

On écrira toutes les commandes demandées dans l'éditeur en prenant soin d'enregistrer le fichier à la fin de la séance (peu importe le nom et l'emplacement du fichier).

Exercice 1

On considère le tableau de données suivant (9 individus, 2 variables quantitatives et un groupe (0 ou 1) pour chaque individu) :

i	1	2	3	4	5	6	7	8	9
X_i^1	2.0	3.0	4.0	4.0	5.0	5.0	7.0	6.0	5.0
X_i^2	2.0	5.0	6.0	3.0	2.0	1.0	3.0	4.0	5.0
Y_i	0	0	0	1	1	1	1	1	0

1. Représenter les données sur un graphique à deux axes (X^1 en abscisse, X^2 en ordonnée).
2. Soit à présent un nouvel individu $X_{10} = (X_{10}^1, X_{10}^2) = (4.0, 4.0)$, que l'on va classer dans l'un des deux groupes 0 ou 1 suivant la méthode des k plus proches voisins pour la distance euclidienne. Placer le nouvel individu sur le graphique puis déterminer à la main son classement dans les cas $k = 1$, $k = 3$, $k = 5$, $k = 7$, $k = 9$.

Exercice 2

Le jeu de données fgl contient les résultats de diverses mesures chimiques et physiques effectuées sur des fragments de verre récupérés au sol. Chaque fragment est classé suivant son type correspondant à l'usage d'origine du verre : verre de fenêtre, de bouteille, de phare de véhicule, etc.

1. Charger les données avec la commande data(fgl), puis observer les données pour comprendre comment elles sont organisées.
2. Séparer les données en deux parties, un jeu de données train pour la phase d'apprentissage contenant $m = 200$ observations tirées au sort parmi les 214, et un jeu de données test contenant les autres observations.
3. A l'aide de la fonction rpart, appliquer la méthode CART sur les données train. On utilisera ici les paramètres par défaut de rpart.
4. Afficher l'arbre de classification obtenu (fonctions plot et text). Dans la fenêtre graphique de Rstudio, cliquer sur le bouton "Export", puis "Save as pdf", et enregistrer le graphique dans un fichier (peu importe son nom et son emplacement).

5. Effectuer à la main la classification du premier individu des données fgl à partir de l'arbre de classification obtenu précédemment.
6. A l'aide de la fonction predict, classer tous les individus des données test avec l'arbre de classification précédent. Calculer le taux d'erreurs obtenu en comparant avec les vrais groupes des données test.
7. Copier le code correspondant aux questions 2, 3 et 6 et le placer dans une boucle afin d'effectuer 100 essais, et calculer la moyenne des taux d'erreurs obtenus.
8. Par défaut rpart ne développe pas jusqu'au bout les arbres de classification. Appliquer à nouveau la méthode CART en forçant le développement jusqu'au bout de l'arbre, et comparer le taux moyen d'erreurs sur 100 essais avec le taux obtenu à la question précédente. La méthode est-elle plus performante en développant les arbres entièrement ? Pourquoi ?

Le *bagging* est une méthode destinée à améliorer les performances des méthodes de classification supervisées (ici CART). Il consiste à appliquer l'algorithme suivant :

- Données d'entrée : train, contenant m observations, et test contenant p observations.
- Pour k allant de 1 à N on répète :
  - on tire au sort **avec remise** m individus parmi ceux des données train. Ceci forme un nouveau jeu de données traink
  - on construit l'arbre de classification avec les données traink
  - on classe les données test avec l'arbre, et on enregistre le classement obtenu.
- Enfin on affecte à chaque individu de test la classe la plus représentée parmi les N groupes.

9. Ecrire une fonction bagging = function(train,test,N) correspondant à l'algorithme du bagging. Pour l'étape d'enregistrement des classements, on utilisera une matrice C de taille $N \times p$ contenant en ligne k le classement des individus test obtenu avec le k^e arbre. Enfin pour la dernière étape, on utilisera la commande suivante :

   ```
   apply(C,2,function(x) names(which.max(table(x))))
   ```

10. Déterminer le taux d'erreurs moyen obtenu avec cette nouvelle méthode (avec $N = 25$) en tirant au sort 100 fois les données train et test comme auparavant, et comparer avec les taux d'erreurs précédents. Commenter.