TP5 : Algorithme des k moyennes

Exercice 1 : introduction

La fonction kmeans()

Le langage R fournit par défaut une fonction *kmeans*().

```
kmeans(x, centers, iter.max = 10, nstart = 1,
       algorithm = c("Hartigan-Wong", "Lloyd"
       ,"Forgy","MacQueen"), trace=FALSE)
```

Pour utiliser cette fonction, il est nécessaire de spécifier au minimum deux paramètres :

- x : un data frame ou une matrice des données. Toutes les valeurs doivent être numériques. Si vos données contiennent une colonne d'identifiants ID, il faudrait la retirer pour qu'elle n'impacte pas le calcul des centres.
- *centers* : correspond au nombre de clusters K. Si on met $centers = 5$, les données seront regroupées en 5 clusters. Il faudrait déterminer la bonne valeur de *centers*.

Les autres paramètres sont optionnels :

- *iter.max* : le nombre d'itérations maximum de l'algorithme (affectation des clusters et mise à jour des centres).
- *nstart* : le nombre d'exécutions de l'algorithme. Chaque exécution démarre par K centroïdes choisis aléatoirement .
- *algorithm* : le langage R offre trois implémentations de l'algorithme *kmeans* (l'implémentation par défaut est celle de "Hartigan-Wong" qui est généralement le plus rapide). Pour comprendre la différence entre les différents algorithmes, vous pouvez consulter les références fournis en fin de TP.
- *trace* : contrôler l'affichage de la trace des exécutions.

Utilisation de la fonction *kmeans*()

Pour se familiariser avec la fonction *kmeans*(), nous allons l'appliquer sur le jeu de données "iris.csv" puis nous comparons les résultats de la segmentation avec les espèces de l'attribut *Species*.

Tout d'abord, chargez la base de données, retirez la colonne des identifiants et normalisez les 4 premières colonnes.

```
iris <- read.csv("iris.csv", header = T)
iris <- iris[,-1]
irisN <- as.data.frame(lapply(iris[,c(1:4)],normalize))
irisN <- cbind(irisN, iris$Species)
colnames(irisN) <- c("SepalLength" , "SepalWidth", "PetalLength", "PetalWidth", "Spec
head(irisN)
```

La commande suivante exécute l'algorithme des k-moyennes sur le jeu de données IRIS en utilisant les quatre premiers attributs et 3 centres.

```
kmeans (irisN [,1:4], 3)
```

La fonction *kmeans*() a associé chaque donnée à un groupe. Les groupes sont numérotés de 1 à k. La fonction *kmeans*() fournit en sortie une liste d'objets. Accédez aux éléments de la liste et expliquer leur contenu :

```
iris.3means <-kmeans (irisN [,1:4], 3)
iris.3means$cluster
iris.3means$centers
iris.3means$withinss
iris.3means$tot.withinss
iris.3means$ betweenss
iris.3means$size
```

Visualisation

Visualisez le jeu de données IRIS projeté sur les attributs longueur et largeur des pétales.

```
plot (irisN$PetalLength, irisN$PetalWidth)
```

On souhaite colorer les points pour visualiser les groupes.

- Faites en sorte que chaque donnée soit colorée en fonction de la valeur du cinquième attribut *Species*. Pour cela, vous pouvez utiliser les foncions *subset*(), *plot*() et *points*().
- Faites en sorte que chaque donnée soit colorée en fonction du groupe dans lequel *kmeans* l'a placé.
- Affichez la table de confusion entre les espèces des iris dans le data frame et les classification trouvée par *kmeans*().

Trouver le meilleur K

On voudrait trouver le nombre de groupes optimal. Pour cela, on va essayer plusieurs valeurs de k entre 2 et 10. Pour chaque valeur de k, on va executer 30 fois la fonction $kmeans()$ puis on retourne la moyenne des inerties

— inertie interclasse : $iris.kmeans\$tot.withinss$

— inertie intraclasse : $iris.kmeans\$betweenss$

Tracez un graphique qui montre l'évolution de inertie interclasse en fonction des valeurs de K testées. Quelle est la meilleure classification ?

Exercice 2 : autres jeux de données

On considère deux nouveaux jeux de données "serpentins.txt" et "concentriques.txt". Faîtes sur chacun une segmentation en 3 groupes. Visualisez le jeu de données en colorant chaque point par une couleur différente en fonction du groupe auquel $kmeans()$ l'a associé. Qu'en pensez-vous ?

Références

Forgey, E. (1965). "Cluster Analysis of Multivariate Data : Efficiency vs. Interpretability of Classification". In : Biometrics.

Lloyd, S. (1982). "Least Squares Quantization in PCM". In : IEEE Trans. Information Theory.

Hartigan, J. A. and M. A. Wong (1979). "Algorithm AS 136 : A k-means clustering algorithm". In : Applied Statistics 28.1, pp. 100–108.

MacQueen, J. B. (1967). "Some Methods for classification and Analysis of Multivariate Observations". In : Berkeley Symposium on Mathematical Statistics and Probability