Méthodes de Partionnement et d'apprentissage non supervisé

Classification Hiérarchique et Kmeans

Anne Badel et Frédéric Guyon

2019-02-20

Partitionnement et apprentissage

Partionnement = Clustering

Partionnement = Clustering

Apprentissage

Apprentissage: Séparation linéaire

Méthodes

Géométrie et distances

Partitionnement et apprentissage

Partionnement = Clustering

Partionnement = Clustering

Apprentissage

Apprentissage: Séparation linéaire

Méthodes

Géométrie et distances

Les données

Les variables

Visualisation des données

Cas d'étude : TCGA Breast Invasive Cancer (BIC)

TP : analyse de données d'expression

Géométrie et distances

Distances

Distances utilisées dans R

Distances utilisées dans R

Autres distances non géométriques (pour information)

Distances plus classiques en génomique

Distances entre groupes

Distances entre groupes

Les données

```
str(mes.iris)
```

```
'data.frame':   150 obs. of  4 variables:
 $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 
 $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1
 $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1
 $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0
```

```
summary(mes.iris)
```

```
  Sepal.Length    Sepal.Width     Petal.Length    Petal.Wid
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.
 Median :5.800   Median :3.000   Median :4.350   Median :1.
 Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.
```

Visualisation des données

Visualisation des données - coloration par espèces

Visualisation des données

Nettoyage des données (1): données manquantes

Nettoyage des données (2) : variables constantes

Normalisation

On peut visuellement regarder l'effet de la normalisation :

par un plot des données

. . . par une boîte à moustaches (boxplot)

. . . par une image

. . . par une projection sur une ACP

La matrice de distances

La classification hiérarchique

Principe

Notion importante, cf distances

L'algorithme

étape 1 :

- départ : n individus = n clusters distincts
- calcul des distances entre tous les individus
  - choix de la métrique à utiliser en fonction du type de données
- regroupement des 2 individus les plus proches => (n-1) clusters

au départ

identification des individus les plus proches

construction du dendrogramme

étape j :

calcul des nouveaux représentants BE et CD

calcul des distances de l'individu restant A aux points moyens

A est plus proche de ...

dendrogramme

pour finir

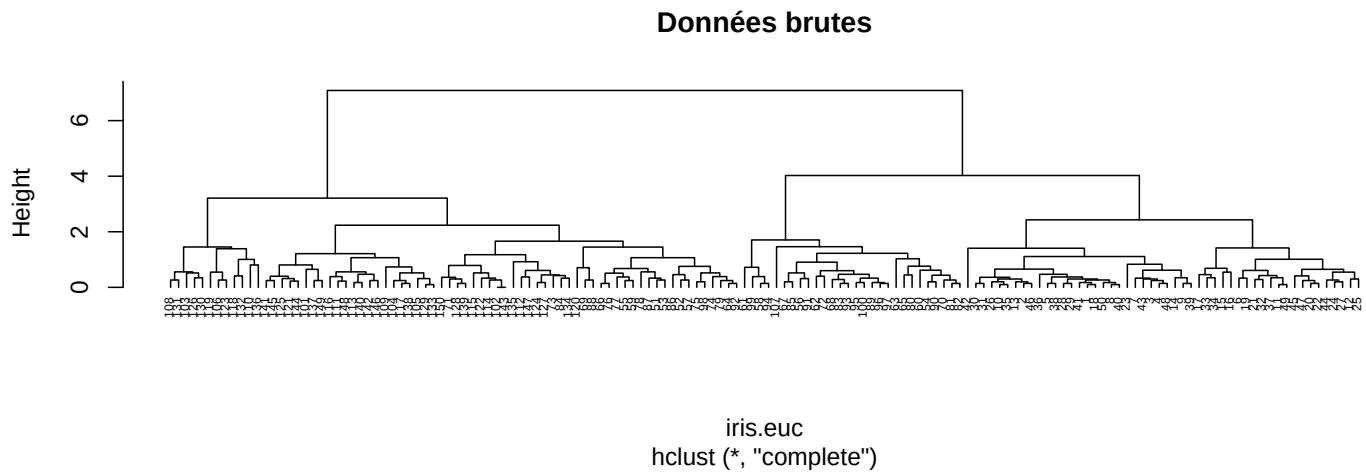
- à l'étape $(n-1)$, tous les individus sont regroupés dans un même cluster

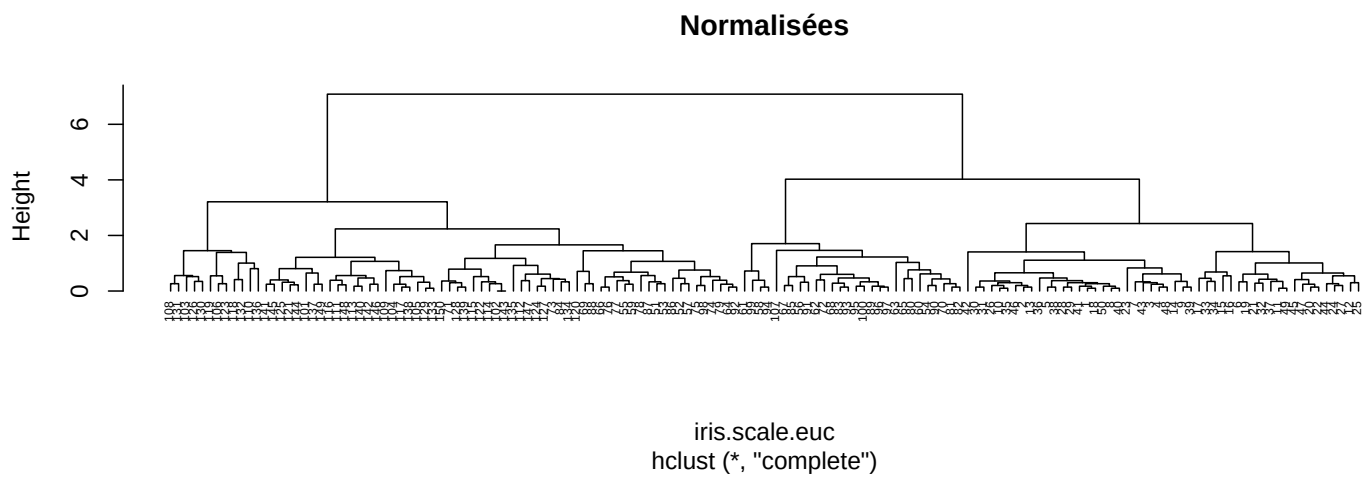
dendrogramme final

Je ne fais pas attention à ce que je fais ...

Sur données normalisées

```
par(mfrow = c(2, 1))
plot(iris.hclust, hang = -1, cex = 0.5, main = "Données bru
plot(iris.scale.hclust, hang = -1, cex = 0.5, main = "Norma
```





En utilisant une autre métrique

En utilisant un autre critère d'aggrégation

Les k-means

L'algorithme

étape 1 :

Choix des centres provisoires

Calcul des distances aux centres provisoires

Affectation à un cluster

Calcul des nouveaux centres de classes

Etape j :

Fin :

Arrêt :

Un premier k-means en 5 groupes

```r
iris.scale.kmeans5$cluster
```

```
  [1] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
[148] 4 4 3
```

```r
table(iris.scale.kmeans5$cluster)
```

```
 1  2  3  4  5
12 50 37 24 27
```

Visualisation des clusters

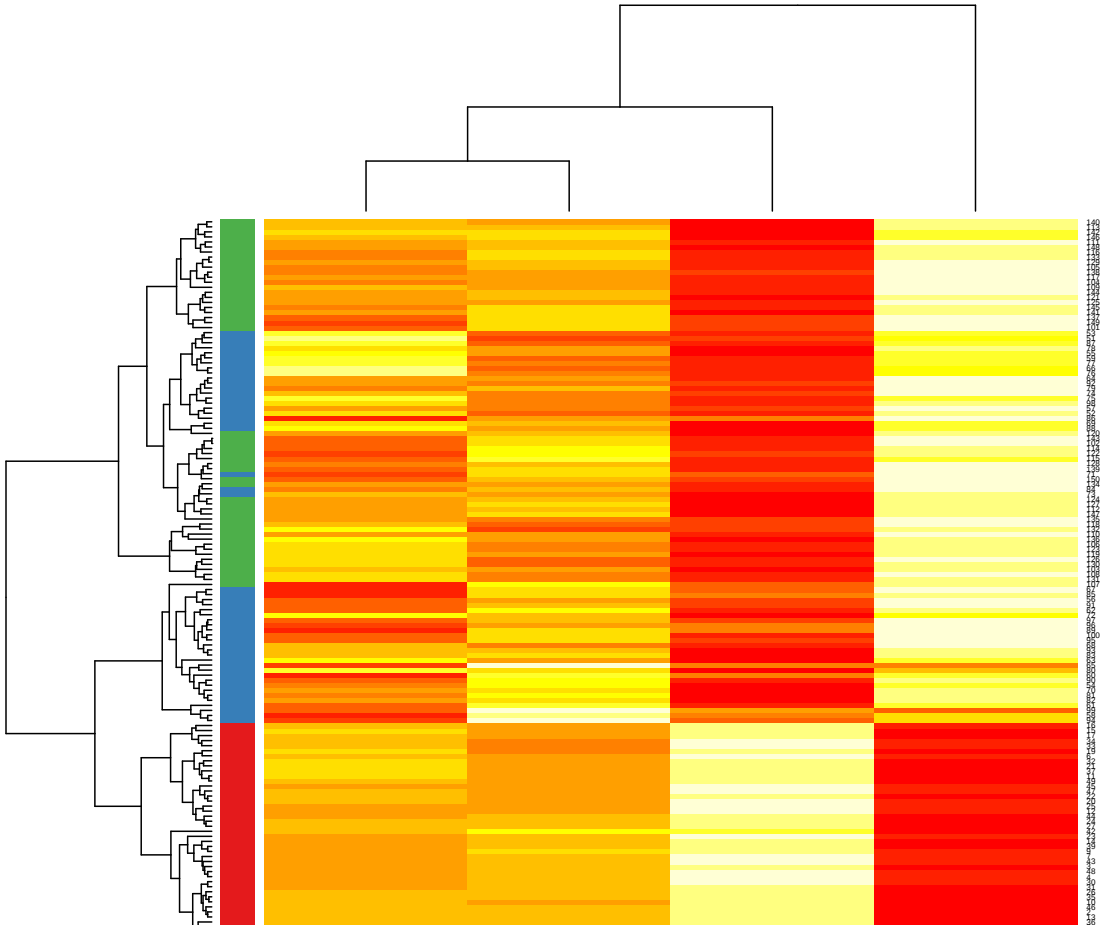
Combien de clusters ?

Classification hiérarchique

K-means

Heatmap

```
my_group <- as.numeric(as.factor(substr(variete, 1 , 2)))
my_col <- brewer.pal(3, "Set1")[my_group]
heatmap(mes.iris.scaled, RowSideColors = my_col,
        margins = c(7,4), cexCol = 1.4, cexRow = 0.5)
```



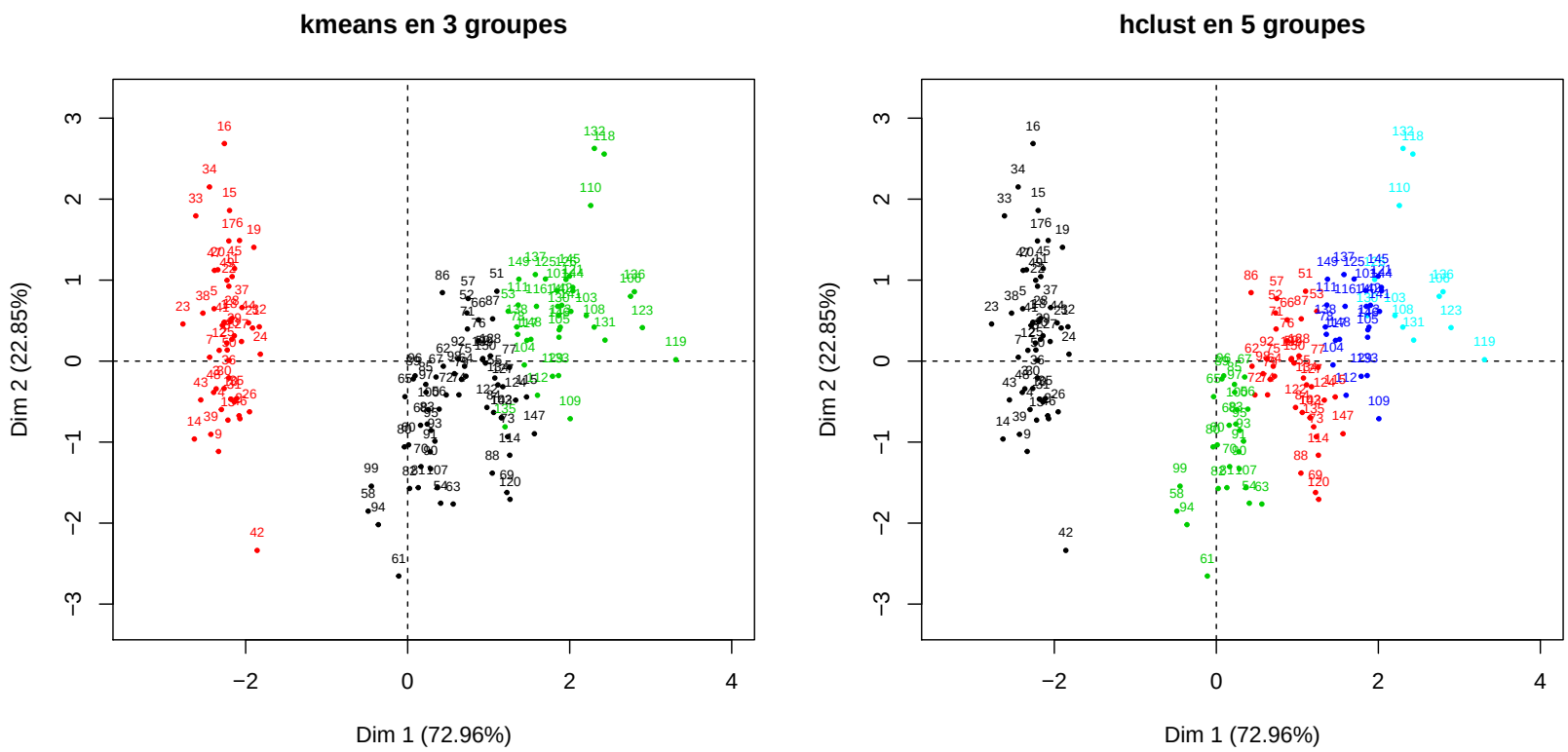
Comparaison de clustering: Rand Index

Comparaison de clustering: Adjusted Rand Index

Comparaison des résultats des deux classifications

- par une visualisation

```
par(mfrow=c(1,2))
plot(iris.scaled.acp, col.ind=cluster.kmeans3, choix="ind"
plot(iris.scaled.acp, col.ind=cluster.hclust5, choix="ind"
```




```
par(mfrow=c(1,1))
```

Comparaison avec la réalité

La réalité

Comparer k-means avec la réalité

Setosa vs others

Visualisation

```
variete2 <- rep("notSetosa", 150)
variete2[variete=="setosa"] <- "setosa"
variete2 = factor(variete2)
table(variete2)
```

```
variete2
notSetosa    setosa
      100        50
```

```
par(mfrow=c(1,2))
plot(iris.scaled.acp, col.ind=variete2, title="variétés obs
cluster.kmeans2 <- kmeans(mes.iris.scaled, center=2)$cluste
plot(iris.scaled.acp, col.ind=cluster.kmeans2, title="kmean
```

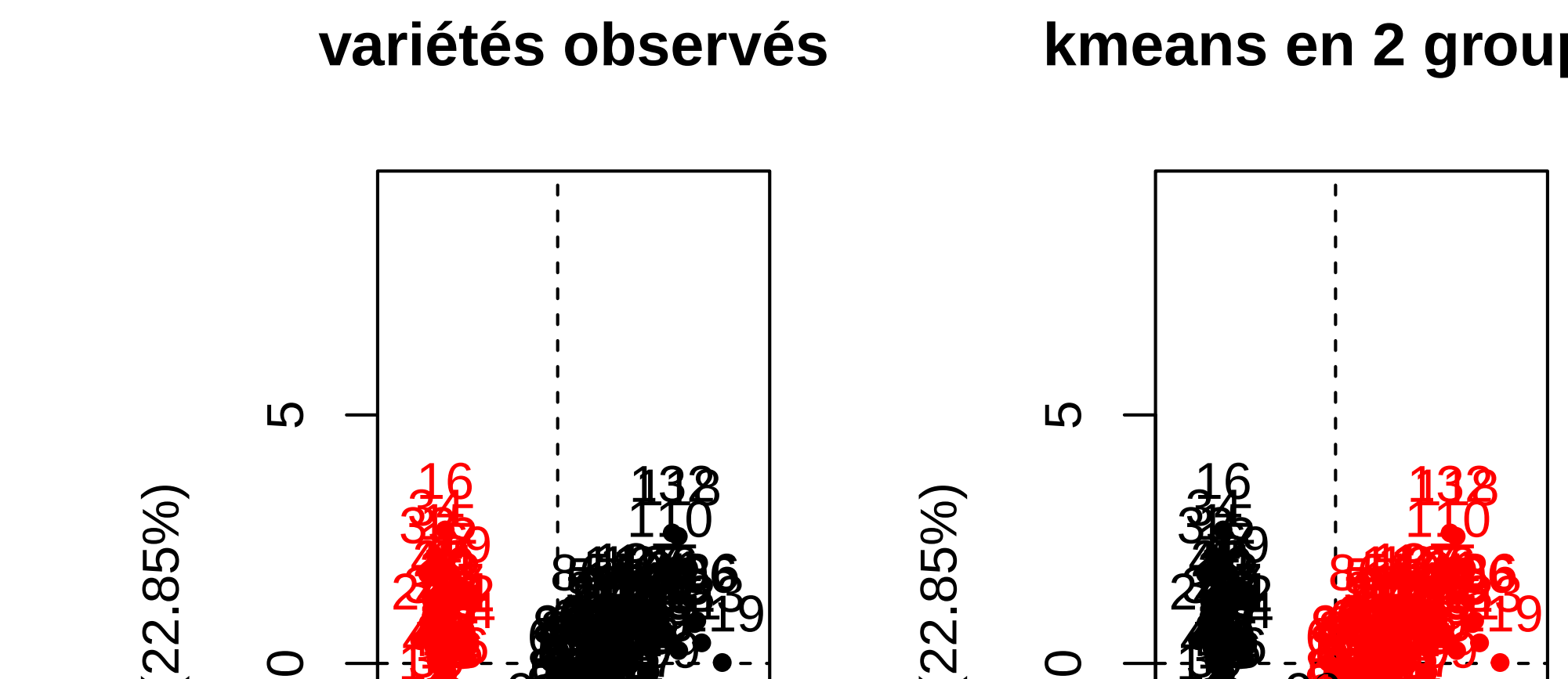


Table de confusion et calcul de performances

```
conf.kmeans <- table(variete2, cluster.kmeans2)
kable(conf.kmeans)
```

	1	2
notSetosa	3	97
setosa	50	0

- table de confusion, taux de bien prédits, spécificité, sensibilité, ...

```
TP <- conf.kmeans[1,1]
FP <- conf.kmeans[1,2]
FN <- conf.kmeans[2,1]
TN <- conf.kmeans[2,2]
P <- TP + FN              # nb positif dans la réalité
N <- TN + FP              # nb négatif dans la réalité
FPrate <- FP / N          # = false alarm rate
Spe <- TN / N         # = spécificité
Sens <- recall <- TPrate <- TP / P        # = hit rate ou re
PPV <- precision <- TP / (TP + FP)
accuracy <- (TP + TN) / (P + N)
F.measure <- 2 / (1/precision + 1/recall)
performance <- c(FPrate, TPrate, precision, recall, accurac
names(performance) <- c("FPrate", "TPrate", "precision", "r
```

```
kable(performance, digits=3)
```

	x
FPrate	1.000
TPrate	0.057
precision	0.030
recall	0.057
accuracy	0.020
F.measure	0.039
Spe	0.000
PPV	0.030

- rand index et adjusted rand index

```
clues::adjustedRand(as.numeric(variete2), cluster.kmeans2)
```

```
     Rand        HA        MA        FM   Jaccard
0.9605369 0.9204051 0.9208432 0.9639434 0.9302767
```

Versicolor vs !Versicolor

Visualisation

```
variete2 <- rep("notVersicolor", 150)
variete2[variete=="versicolor"] <- "versicolor"
variete2 = factor(variete2)
table(variete2)
```

```
variete2
notVersicolor    versicolor
          100            50
```

```
par(mfrow=c(1,2))
plot(iris.scaled.acp, col.ind=variete2)
cluster.kmeans2 <- kmeans(mes.iris.scaled, center=2)$cluste
plot(iris.scaled.acp, col.ind=cluster.kmeans2)
```

Table de confusion et calcul de performances

```
kable(performance, digits=3)
```

	x
FPrate	0.943
TPrate	0.515
precision	0.500
recall	0.515
accuracy	0.353
F.measure	0.508
Spe	0.057
PPV	0.500

```
clues::adjustedRand(as.numeric(variete2), cluster.kmeans2)
```

```
      Rand          HA          MA          FM     Jaccard
0.53995526 0.07211421 0.07722223 0.57895580 0.40737752
```

Contact: anne.badel@univ-paris-diderot.fr