ctrl F Agency

text-mining : Clusterisation – kmeans (k-moyennes)

Text-mining Basics : Tips and Tricks R(S)

Par Thibaut LOMBARD

Décembre 2016

ctrl F Agency

Kmean.r

```
packages <- function(paquets)
{
new.paquets <- paquets[!(paquets %in% installed.packages()[, "Package"])]
if (length(new.paquets))
install.packages(new.paquets, dependencies = TRUE,
repos='http://cran.rstudio.com/')
sapply(paquets, require, character.only = TRUE)
}
packages(c("NLP", "tm", "cluster", "factoextra", "NbClust"))
```

ctrl F Agency

L'algorithme k-means :

- Fait partie des algorithme d'apprentissage non-supervisé
- Sert à résoudre des problèmes de classification
- Aide à détailler des configurations de données
- Permet l'organisation des données en groupes distincts
- Peut servir à résoudre les problèmes de classification par l'identification (qualitatif et/ou quantitatif)

ctrl F Agency

Exemples d'utilisation de L'algorithme k-means

- Classer les e-mail de manière automatisée
- Regrouper les clients dans un segment (secteur de marché)
- Regrouper des serveurs ensemble pour optimiser l'espace
- Identifier les génomes par la visualisation
- En médecine, il sert à identifier certaines pathologies (cancers, virus etc..)
- Utilisé dans les technologies OCR (reconnaissance de formes)
- . . .

Les stratégies K-means

$$SS(k) = \sum_{i=1}^{n} \sum_{j=1}^{p} (x_{ij} - \bar{x}_{kj})^2$$

- **K-means clustering – (MacQueen , 1967)** La partition de n points en k ensembles S. Minimisation de la distance entre les points à l'intérieur de chaque partitions par le calcul de centroid.
- **K-medoids clustering ou PAM (Partitioning Around Medoids, Kaufman & Rousseew, 1990)** Chaque cluster est représenté par un des objets dans le cluster.

ctrl F Agency

Méthode

1. Sélection des k centroids (pour k lignes au hasard)
2. Fait correspondre chaque points à son centroid le plus proche
3. Re-calcul les centroids comme la moyenne de tous les points contenu dans le cluster(pour des centroid d'un vecteur p-length moyen, ou p représente le nombre de variables)
4. Fait correspondre chaque points à leur plus proche centroids
5. Continue les étapes 3 et 4 jusqu'à ce que les observations ne soient pas réitérées aux maximum.

Création de la matrice

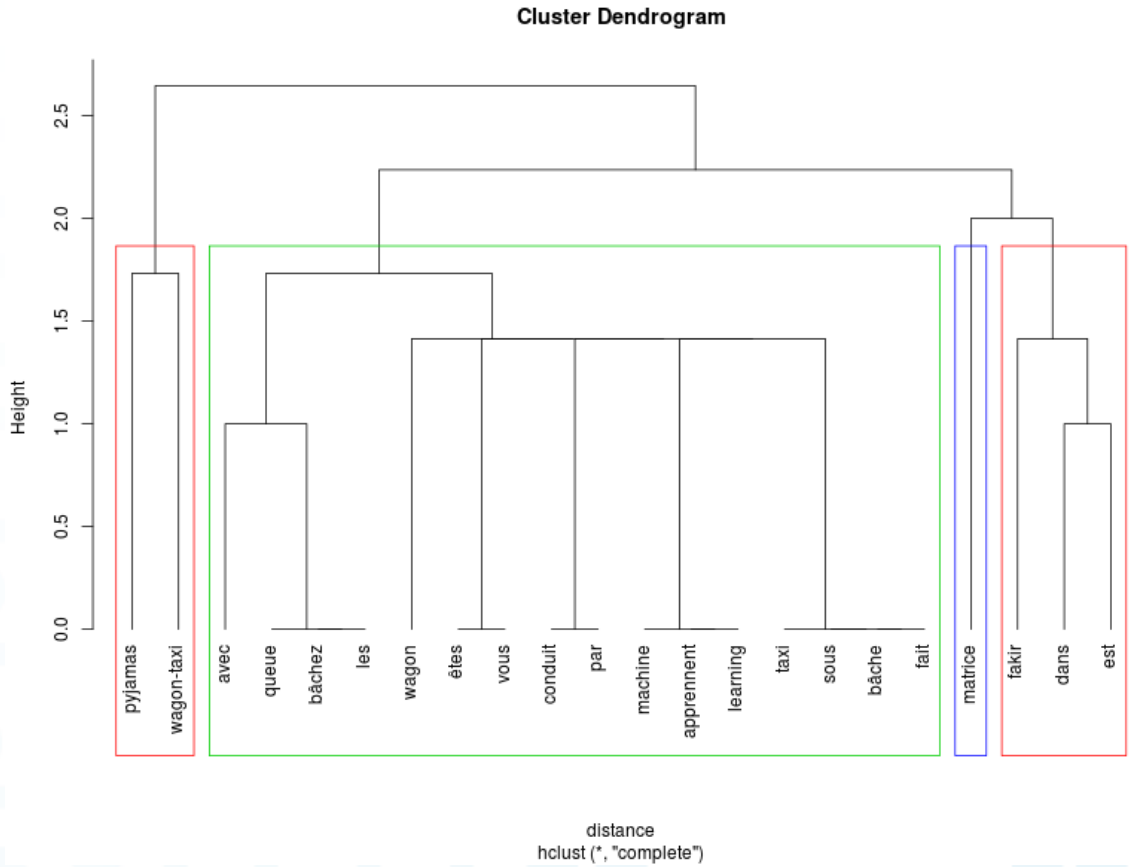
```
# Ajoutons un Pangramme
# info : https://fr.wikipedia.org/wiki/Pangramme
monTexte <- c("Bâchez la queue du wagon-taxi avec les pyjamas du fakir",
              "la matrice du wagon-taxi",
              "le fakir est dans la matrice",
              "le taxi fait le pyjamas sous la bâche",
              "le fakir est dans le wagon",
              "la matrice est conduit par le fakir",
              "vous êtes avec la matrice de pyjamas",
              "le wagon-taxi de pyjamas apprennent le machine learning")
x <- data.frame(monTexte)

# Mets le tableau x en document term matrix
docs <- Corpus(DataframeSource(x))
dtm <- DocumentTermMatrix(docs)
inspect(dtm)
dtmss <- removeSparseTerms(dtm, 0.9)
```

Première estimation du nombre de cluster (k=4)

```r
# Clustering hierarchique avec hclust
# Calcul de distance sur un espace Euclidien
distance <- dist(t(dtmss),method = "euclidean")
distance
# Utilise la method="complete" et non Ward.D
hc <- hclust(distance, method = "complete")
plot(hc, hang = -1)
# Ajoute des rectangles multicolores
rect.hclust(hc, k = 4, border = 2:4)
```

ctrl F Agency

Première estimation du nombre de cluster (k=4)



ctrl F Agency

Résultats de la distance matrix

apprennent	avec	bâche	bâchez	conduit	dans	est	
avec	1.732051						
bâche	1.414214	1.732051					
bâchez	1.414214	1.000000	1.414214				
conduit	1.414214	1.732051	1.414214	1.414214			
dans	1.732051	2.000000	1.732051	1.732051	1.732051		
est	2.000000	2.236068	2.000000	2.000000	1.414214	1.000000	
êtes	1.414214	1.000000	1.414214	1.414214	1.414214	1.732051	2.000000
fait	1.414214	1.732051	0.000000	1.414214	1.414214	1.732051	2.000000
fakir	2.236068	2.000000	2.236068	1.732051	1.732051	1.414214	1.000000
learning	0.000000	1.732051	1.414214	1.414214	1.414214	1.732051	2.000000
les	1.414214	1.000000	1.414214	0.000000	1.414214	1.732051	2.000000
machine	0.000000	1.732051	1.414214	1.414214	1.414214	1.732051	2.000000
matrice	2.236068	2.000000	2.236068	2.236068	1.732051	2.000000	1.732051
par	1.414214	1.732051	1.414214	1.414214	0.000000	1.732051	1.414214
pyjamas	1.732051	1.414214	1.732051	1.732051	2.236068	2.449490	2.645751
queue	1.414214	1.000000	1.414214	0.000000	1.414214	1.732051	2.000000
sous	1.414214	1.732051	0.000000	1.414214	1.414214	1.732051	2.000000
taxi	1.414214	1.732051	0.000000	1.414214	1.414214	1.732051	2.000000
vous	1.414214	1.000000	1.414214	1.414214	1.414214	1.732051	2.000000
wagon	1.414214	1.732051	1.414214	1.414214	1.414214	1.000000	1.414214
wagon-taxi	1.414214	1.732051	2.000000	1.414214	2.000000	2.236068	2.449490

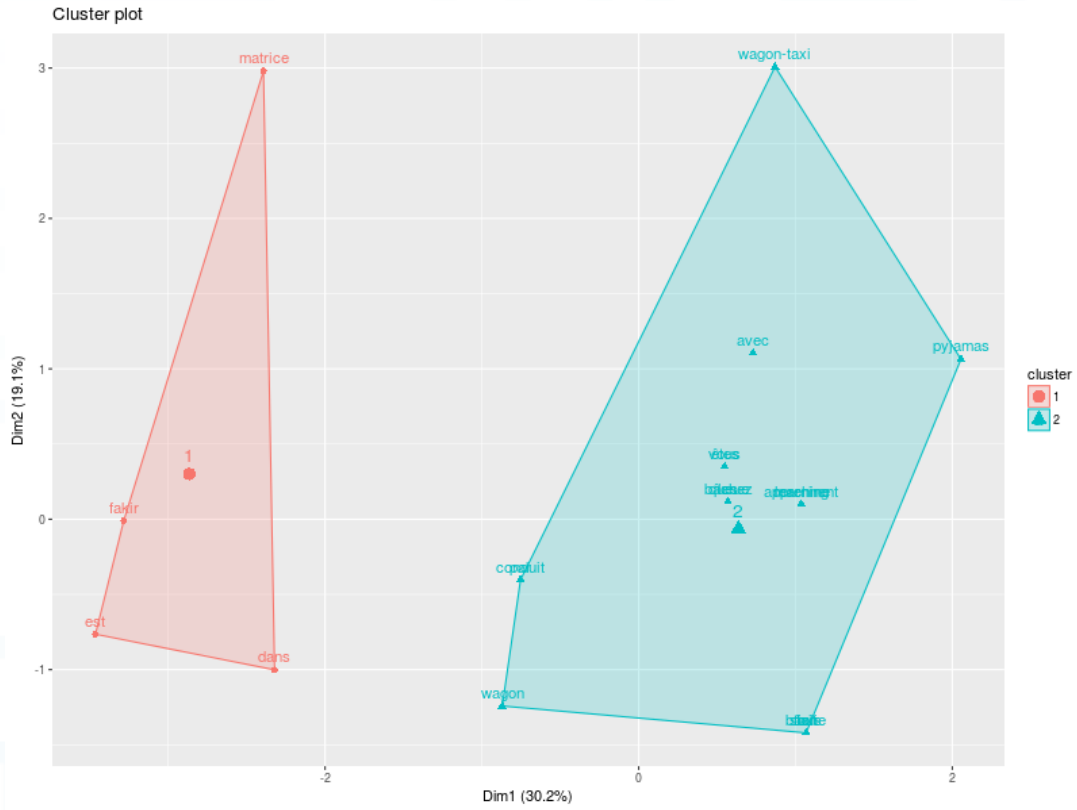
ctrl F Agency

Visualisation du nombre optimal de cluster pour k=2

```
# Méthode nbclust
datatwo <-  dist(t(dtmss),method = "euclidean")
res <- NbClust(datatwo, distance = "euclidean",
                  min.nc = 2, max.nc = 10,
                  method = "complete", index ="gap")
res
# Créé la visualisation cluster
km <- kmeans(datatwo, 2, nstart = 25)
fviz_cluster(km, data = scale(t(dtmss)))
```

ctrl F Agency

Visualisation du nombre optimal de cluster pour k=2



ctrl F Agency

Résultats k-mean

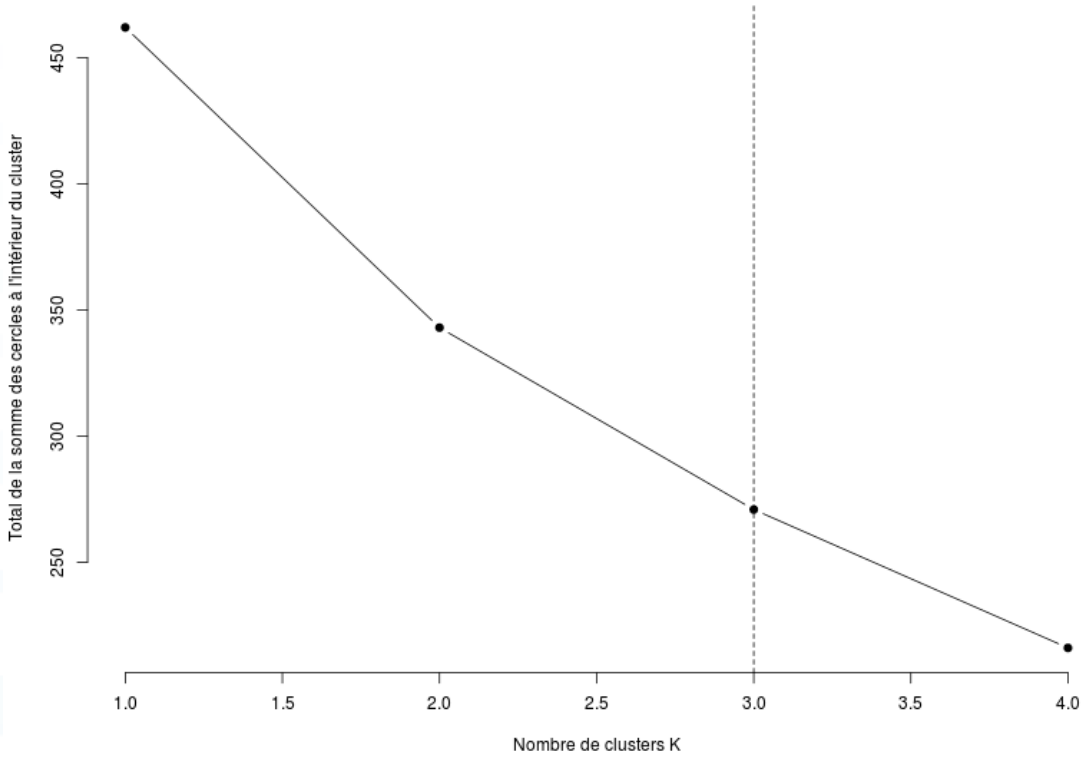
```
K-means clustering with 2 clusters of sizes 4, 18

Cluster means:
  apprennent     avec    bâche   bâchez  conduit     dans       est     êtes
1   2.051047 2.059017 2.051047 1.925042 1.652591 1.103553 0.9330127 1.925042
2   1.213827 1.414821 1.167803 1.173157 1.352939 1.774126 1.9763305 1.284268
      fait    fakir learning      les  machine  matrice      par  pyjamas
1 2.051047 1.103553 2.051047 1.925042 2.051047 1.433013 1.652591 2.498555
2 1.167803 2.066804 1.213827 1.173157 1.213827 2.122806 1.352939 1.702171
     queue     sous     taxi     vous    wagon wagon-taxi
1 1.925042 2.051047 2.051047 1.925042 1.595583   2.289423
2 1.173157 1.167803 1.167803 1.284268 1.431506   1.663855

Clustering vector:
apprennent       avec      bâche     bâchez    conduit       dans       est
         2          2          2          2          2          1          1
      êtes       fait      fakir   learning        les    machine    matrice
         2          2          1          2          2          2          1
       par    pyjamas      queue       sous       taxi      vous      wagon
         2          2          2          2          2          2          2
wagon-taxi
         2

Within cluster sum of squares by cluster:
[1]  11.72809 100.78828
 (between_SS / total_SS =  26.4 %)
```

Méthode elbow

```
# Méthode elbow
distance.scaled <- scale(distance)
k.max <- 4 # Maximal number of clusters
data <- distance.scaled
wss <- sapply(1:k.max,
        function(k){kmeans(data, k, nstart=10 )$tot.withinss})

plot(1:k.max, wss,
       type="b", pch = 19, frame = FALSE,
       xlab="Nombre de clusters K",
       ylab="Total de la somme des cercles à l'intérieur du cluster")
abline(v = 3, lty =2)
```

ctrl F Agency

Méthode elbow



ctrl F Agency

Liens connexes :

- Documentation hclust
- Déterminer le nombre optimal de clusters (3 méthodes)
- Partitioning cluster analysis
- Visual enhancement of clustering Analysis
- CH3-Classification
- Classification non supervisée
- Wikipedia – K-moyennes

ctrl F Agency

Q/A