**INF 413 – TP5-6**

Alan Gardin – Ronan Garet

13/03/2017

# Fonctionnement de l’algorithme

L’algorithme prend en entrée une liste de n points [p0, …,pn-1] à d coordonnées (x0, …,xd-1) et k, le nombre de barycentres.

L’algorithme renvoie en sortie C = [C1, …, Ck-1] ou Ci est une liste contenant l’ensemble des points du cluster associé au centre i.

Début de l’algorithme :

Choisir k centres [c0,…,ck-1]distincts de parmi [p0, …,pn-1]

Calculer (Li,j)(i,j) ∈ [0,n-1]² l’ « éloignement » entre chaque couple de points, où Lij=Lji correspond à l’éloignement entre les points i et j

Tant que condition d’arrêt non-respectée :

Pour i allant de 0 à n-1 :

On cherche j tel que Lij minimal et pj ∈ [c0,…,ck-1]

Ajouter pi à C

Fin Pour

Pour i allant de 0 à k-1 :

Calculer Bi, le barycentre de Ci

Remplacer ci par Bi

Fin Pour

Fin Tant que

Retourner C

# Calcul de l’ « éloignement » :

Dans un 1er temps, nous avons choisi le calcul de distance qui nous a paru le plus naturel : la distance euclidienne. Soit Dij la distance entre les points pi et pj, on a donc

Pour simplifier les calculs, on ne calculera pas la raçine carrée et on appellera cette valeur « éloignement ». On a donc Li,j  = Dij²

# Quantifier la qualité d’une solution

Pour un nombre de centres donnés,, une solution est d’autant plus optimale que la distance moyenne entre chaque point et son centre est faible. C’est cette valeur que nous utiliserons pour quantifier la qualité d’une solution par rapport à une autre.

# Choix du nombre de sommets

Dans un premier temps, le nombre de sommets sera choisi de manière arbitraire.

# Calcul du barycentre

On prend comme barycentre d’un ensemble de n points la moyenne de leurs coordonnées.

# Condition d’arrêt

L’algorithme des k-means converge vers une solution localement optimale. L’idéal serait donc d’arrêter le programme quand les solutions trouvées ne varient plus.

Cependant, il est possible que cela n’arrive jamais (présence de cycles).

Dans un premier temps, on choisira comme condition d’arrêt un nombre d’itérations choisi de façon arbitraire.

Une manière plus efficace que nous implanterions dans une version ultérieure serait de garder en mémoire la meilleure solution trouvée (distance moyenne entre les points et leur centre minimale) ; si cette solution n’a pas changé après un certain nombre d’itérations (choisi de façon arbitraire ?) on arrête le programme.

# Choix initial des centres

Dans un premier temps, le choix initial des centres se fera de manière aléatoire. Aucun point n’étant plus important qu’un autre on choisira une probabilité uniforme.

Cependant, l’algorithme des k-means ne fournissant qu’une solution localement optimale, les choix initiaux des centres s’avère crucial et devra être amélioré.

# Complexité de l’algorithme

* Le chois des centres initiaux est en O(1)
* Le calcul de la distance est en O(n.k.d) : pour chaque point, on calcule sa distance avec chaque barycentre, et le calcul de la distance entre points est en O(d)
* Le calcul des barycentres est en O(n.d+k) : Pour chaque cluster, on fait une addition des coordonnées de ses points qu’on divise par le nombre de points du cluster. Au final, on a un nombre de calculs en O(n.d+k).

Si on choisit comme condition d’arrêt un nombre donné d’itérations, on a une complexité de l’algorithme en O(n.k.d).

Si on se donne comme condition d’arrêt la convergence de la solution

# Utilisations possibles de ce genre d’algorithmes

* Identification de groupes d’individus dans une foule (une dimension par caractéristique étudiée : âge, revenus, …).
* Identification de regroupements d’étoiles afin de déterminer des galaxies