Projet - PAM

Algorithmes et concepts pour la science des données

Baptiste Deldicque, Nicolas Fond-Massany, Ines Lebib, Marc Wang

December 8, 2021

1 Compléxité de l'algorithme

Pour rappel, PAM (*Partitionning Around Medoids*) consiste à faire l'algorithme suivant :

1. Choisir aléatoirement k objets du *dataset* D comme configuration de départ.
2. Tant que le coût de la configuration actuel descend :
   (a) Associé chaque objet non représentatif au cluster le plus proche (avec une distance de *Manhattan* dans notre cas)
   (b) Pour chaque medoid m et pour chaque non-medoid o faire :
      i. Calculer le coût E de la configuration ou o est un medoid à la place de m.
      ii. Si ce coût E est meilleur que le coût précédent S, on garde cette configuration et S devient E.
   (c) Si le meilleur échange (au sens de la diminution du coût) permet de dimnuer le coût, effectuer le meilleur échange (m, o).

L'algorithme est composé de 2 parties. Une première, souvent appeller la **phase *build*** car on initialise les valeurs de départs. Et une seconde qu'on appelle souvent la **phase de *swap*** car on échange les medoids afin de trouver un coût minimum.

La phase d'initialisation *(1)* a une complexité négligeable. En effet, on cherche ici à simplement prendre K objets au hasard dans le *dataset*. C'est une complexité en $O(n)$.

La phase d'échange *(2)* :

- *(b)* L'idée est ici de chercher la meilleur configuration en calculant le coût minimal à chaque échange entre m et o. On va répeter cette opération $(n-k)^2$ fois avec n le nombre de clusters et k le nombre d'objet du *dataset*. On enlève k à n car on a déjà k medoids et qu'on ne peut pas avoir deux medoids identiques.

- Une fois le minimum trouvé on va chercher une configuration encore meilleur. Pour cela, on répète l'opération précèdente tant que le coût diminue. Cette opération est dans le pire des cas fait k fois.

Finalement, on se retrouve avec une complexité en $O(k(n-k)^2)$.

2 Comparaison K-MEANS / PAM