

Rapport d'avancement

Première séance :

Durant cette première séance de TP, nous avons travaillé principalement sur deux parties : l'étude des k-means et l'implémentation de l'algorithme en python

1-Étude des k-means et des différents points clés de l'algorithme

Dans cette partie, nous avons pris connaissance de l'algorithme k-means en essayant de comprendre l'exemple donné dans le sujet de tp. Nous avons ensuite lu une partie du document « learning the k in k-means » donné sur moodle. Pour plus comprendre ce que l'algorithme fait réellement, nous avons essayé par la suite de l'implémenter.

2-Implémentation de l'algorithme et des points étudiés en python

Nous avons commencé à coder en python en testant d'abord le fichier « io.py ». Nous avons ensuite utilisé ce fichier pour continuer à coder les fonctions en respectant les consignes donnés dans le sujet de tp.

Nous avons utilisé la fonction « generate_random_data » donnée pour faire la génération aléatoire de données en fonction du nombre. Pour générer aléatoirement des centres, nous avons créé la fonction « generate_random_centres » qui produit une liste de k centres avec k donné en paramètre.

Nous avons ensuite chargé les données dans des fichiers comme dans l'exemple du fichier « io.py ».

Pour calculer la distance, nous avons implémenté une fonction « calcul_distance » qui prend en paramètres les données d'entrée et les centres. Comme le montre le code, nous obtenons une liste de valeurs avec les numéros d'observations et les distances calculées.

Nous avons terminé cette séance en réalisant la fonction affectation et mettre les données correspondantes dans un fichier. Cette fonction prend en paramètres les données d'entrée, les distances ainsi que les centres afin de faire correspondre chaque observation son centre correspondant.

Deuxième séance :

Durant cette deuxième partie, nous avons travaillé sur la partie implémentation de l'algorithme et des points étudiés en python.

Nous avons commencé à classer les observations dans chaque groupe créant ainsi des clusters. Après cela, nous avons calculé les nouveaux centres de chaque groupe. Pour continuer le même processus, nous sommes partis sur la boucle « while » et le critère qui est telle que la boucle s'arrête dès qu'on retrouve les mêmes centres.

Nous avons réussi à implémenter ces différents points de l'algorithme. IL nous reste à utiliser des données massives pour plus vérifier notre résultat.