

Option Informatique

K-means

Antoine Lanco
lanco@lri.fr

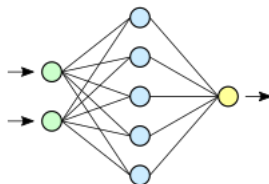
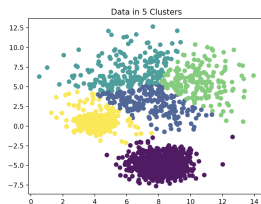
04/01/2023

Organisation

- Antoine LANCO
- Loïc LE-MOGNE
- 3 Cours - TD
- 3 TP
- Note : Mini projet

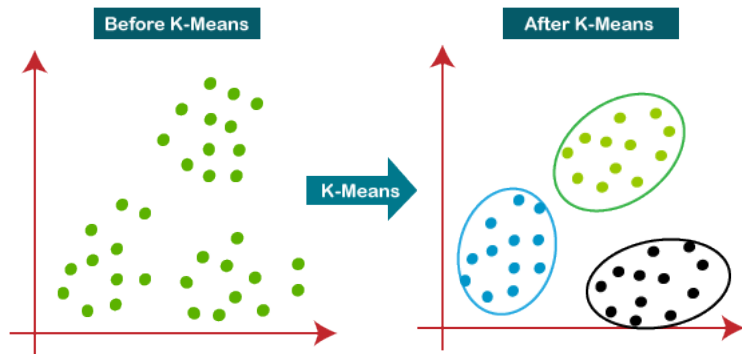
Intro

- Machine learning
- Algo supervisé
- Algo non-supervisé



- K-Means sur des images

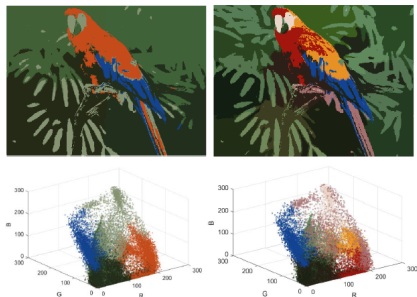
K-Means



- Minimiser la somme des distances
- Distance euclidienne (notion de similarité)
- Nombre de cluster
- Dimension (ici 2d)

Utilisation

- Publicité ciblée (dimension n)
- Dessiner une image avec n couleur (dimension 3)



Algo

- Comment calculer le nouveau centre du cluster ?
- Proposer une condition d'arrêt
- Proposer un algo k-means

Algo

- Algo de point fixe

$K \leftarrow$ le nombre de cluster

$T \leftarrow$ le training set

$TC \leftarrow K$ points dans T

while les centres changes **do**

for tous les éléments i dans T **do**

for tous les éléments j dans TC **do**

 calculer la distance euclidienne entre i et j

 affecter au point i le cluster le plus prêt

end for

end for

for tous les éléments j dans TC **do**

 re-calculer les centres de chaque cluster

end for

end while

Langage et structures de données

- Python
- Liste / Pair / Tableau associatif

Langage et structures de données

- Python
- Liste / Pair / Tableau associatif

```
point = (1,0,2) #x=1, y=2, cluster=3  
t_s = [(1,0,2),(5,6,1),(7,8,0)]  
cluster = [(4,5),...]
```

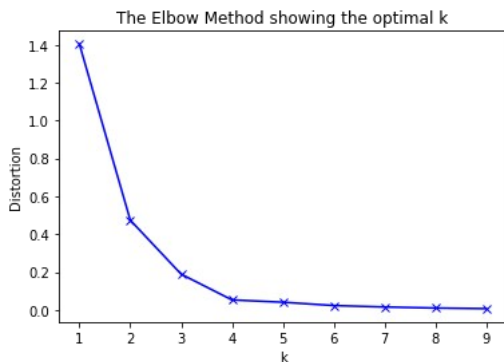
Complexité

- Nombre de point N
- Nombre de cluster K
- Nombre de dimension D

Pour chaque point on calcul la distance euclidienne avec tous les centres des cluster. Le calcul de la distance euclidienne est linéaire en la dimension. La complexité de cet algo est donc $\mathcal{O}(N * K * D)$. C'est donc un algo linéaire.

Convergence et nombre de cluster

- Comme pour tous les algo de point fixe il faut se poser la question de la convergence.
- En fonction de l'objectif il y a un nombre de cluster optimal.



Exemple & Démo

- Mathplotlib

