K-mean: Initialisation

Loïc Le Mogne

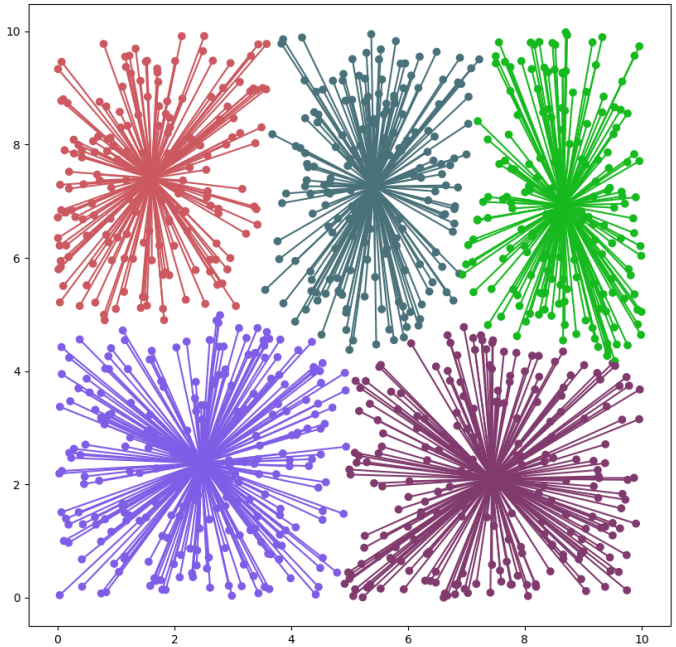
06/01/2023

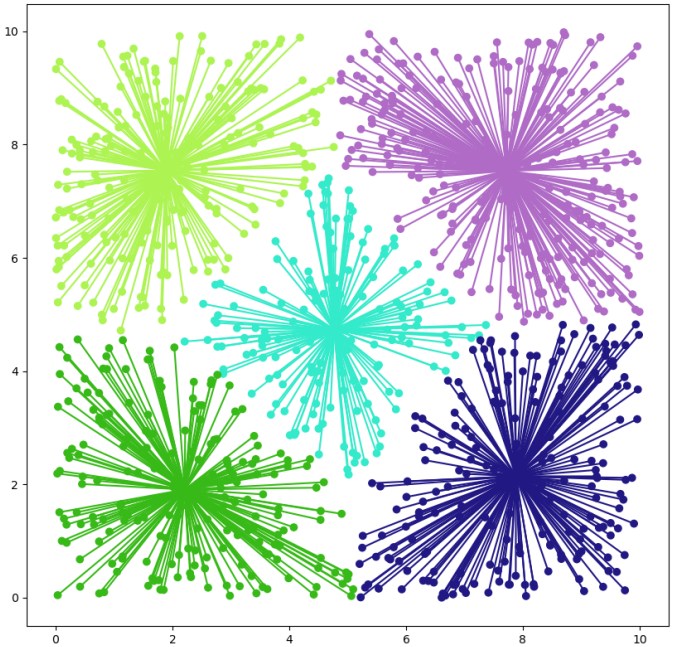
Rappel

- L'algo k-mean est un algo d'approximation
- Il chercher à minimiser une valeur (distance)

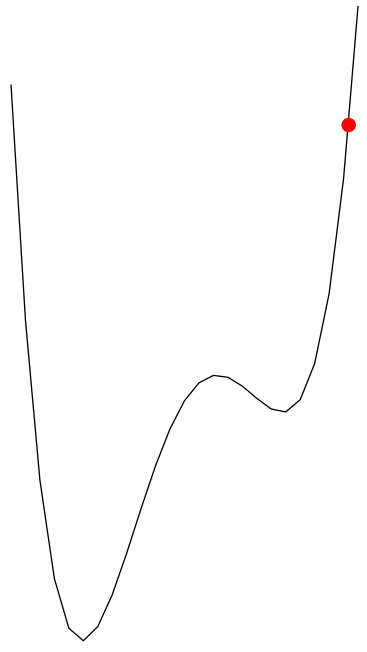
L'effet de l'initialisation

Sur les mêmes données de départ: plusieurs résultats selon l'initialisation

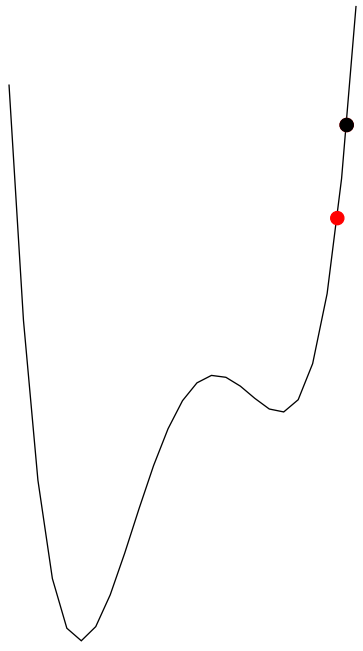




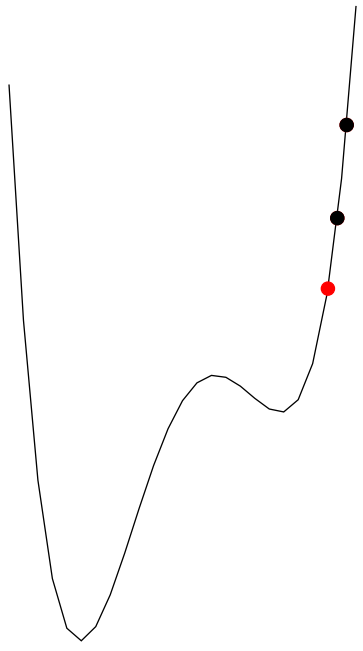
Un dessin: la descente de gradient



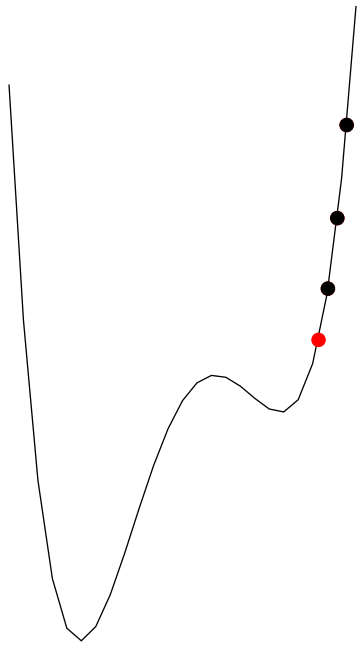
Un dessin: la descente de gradient



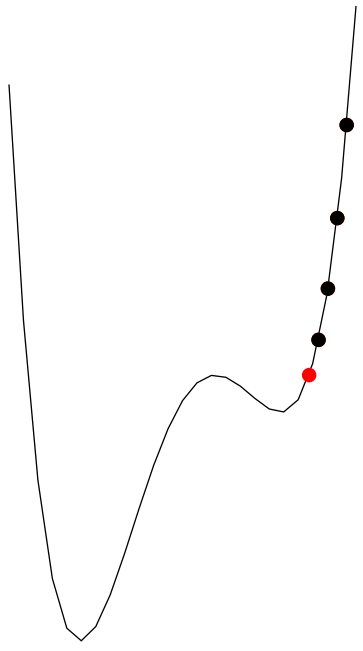
Un dessin: la descente de gradient



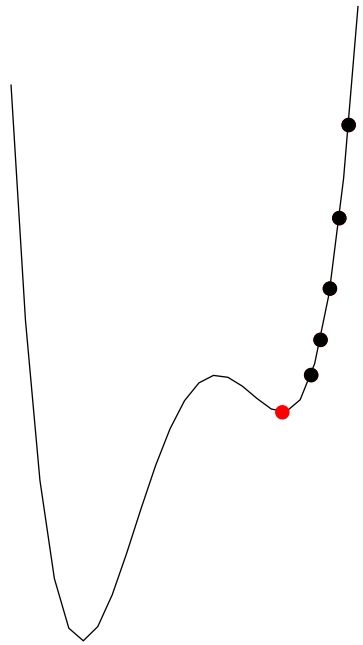
Un dessin: la descente de gradient



Un dessin: la descente de gradient



Un dessin: la descente de gradient



Optimiser l'initialisation?

La grande question autour de l'initialisation:
Comment bien initialiser nos k points/clusters pour augmenter nos chances de tomber sur le minimum global (i.e. obtenir le meilleur résultat)?

Contexte général

La question n'est pas propre au problème des k-means:

- En apprentisage supervisé: échantillon initial
- En apprentissage non-supervisé: modèle* initial

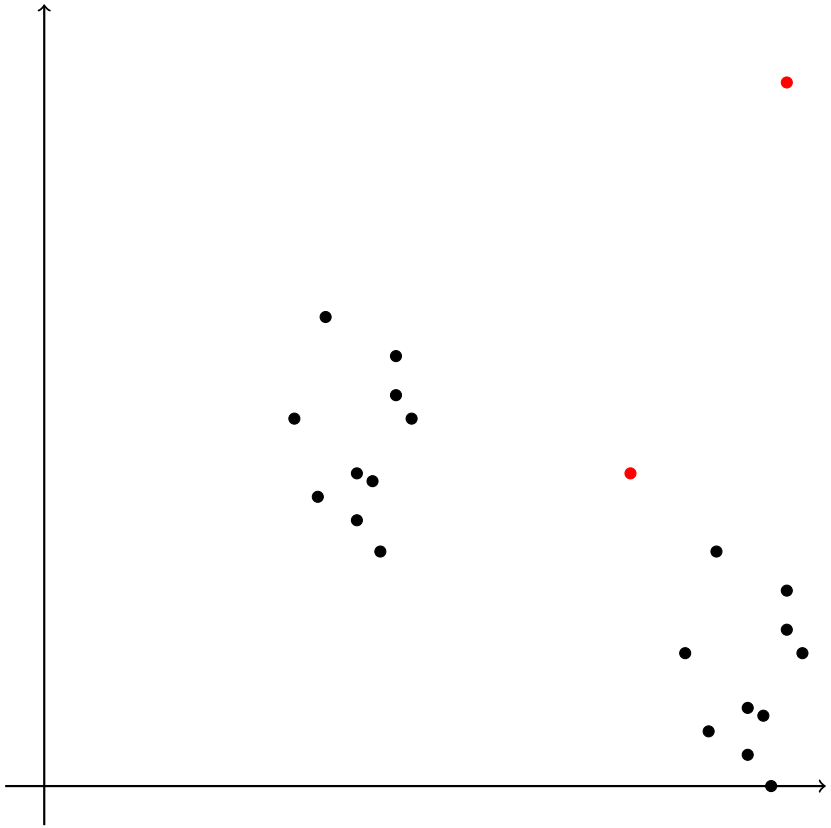
*Pour k-means, le modèle correspond aux clusters considérés

Approche la plus naïve (Jancey)

Prendre des k points aléatoires de l'espace (pas nécessairement de notre ensemble de points).

Approche la plus naïve (Jancey)

Prendre des k points aléatoires de l'espace (pas nécessairement de notre ensemble de points).



L'approche de Forgy

On assigne à chaque point un cluster aléatoire, puis on commence les itérations en commençant par calculer les centres.
Plus de problème de cluster vide.

L'approche de Forgy-MacQueen

On choisit k points parmis notre ensemble de points de manière aléatoire avec une proba uniforme:

- pas de cluster vide
- plus de chance de prendre des points dans des régions denses (i.e. proche de centre de clusters)

L'approche de Forgy-MacQueen

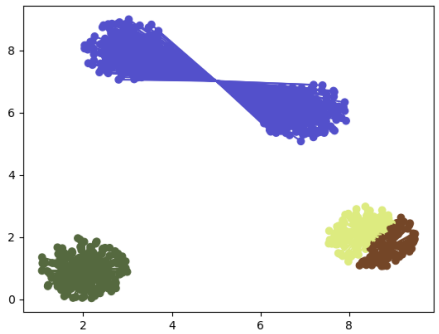


Figure 1: Résultat après Forgy-MacQueen

L'approche de Forgy-MacQueen

Un défaut: forte probabilité de choisir plusieurs points proche d'un même centre.

L'approche de Forgy-MacQueen

Un défaut: forte probabilité de choisir plusieurs points proche d'un même centre.
Une manière de régler ce problème consiste à faire I initialisations et continuer avec celle qui minimise la distance. Cette méthode est couteuse : $\mathcal{O}(K * N * I)$.

L'approche Maximin

Comment choisir (assez naivement) k points éloignés?

L'approche Maximin

Comment choisir (assez naivement) k points éloignés?

- Prendre un point aléatoire
- Choisir le i ème point comme étant le point le plus éloignés des $i-1$ précédents

L'approche Maximin

Problème: Les points de notre data set très éloignés des autres ont de grandes chances d'être choisis

Une dernière approche: k-means ++

On va combiner les méthodes de Forgy-MacQueen et de Maximin.

Une dernière approche: k-means ++

On va combiner les méthodes de Forgy-MacQueen et de Maximin.
Comment le faire?

Une dernière approche: k-means ++

- On choisit le premier point aléatoirement

Une dernière approche: k-means ++

- On choisit le premier point aléatoirement
- On choisit aléatoirement le i ème point en pondérant par la distance minimal au $i-1$ points déjà choisis

Une dernière approche: k-means ++

- On choisit le premier point aléatoirement
- On choisit aléatoirement le i ème point en pondérant par la distance minimal au $i-1$ points déjà choisis
- Plusieurs pondérations possible, la plus fréquente: $\frac{md(x)^2}{\sum_{j=1}^{N} md(x_j)^2}$

Peut-on calculer intelligement md (la distance minimal entre x et les i centres déjà choisis)?

Une dernière approche: k-means ++

- On choisit le premier point aléatoirement
- On choisit aléatoirement le i ème point en pondérant par la distance minimal au $i-1$ points déjà choisis
- Plusieurs pondérations possible, la plus fréquente: $\frac{md(x)^2}{\sum_{j=1}^{N} md(x_j)^2}$

Peut-on calculer intelligement md (la distance minimal entre x et les i centres déjà choisis)?
Quelle est la compléxité de cette méthode?