

# Apprentissage non supervisé



Montaigne 2023-2024

– mpi23@arrtes.net –

# Algorithme des k moyennes

On dispose d'un jeu de données pour lesquelles **aucune classification** n'est préalablement **connue**.

Une classification ne peut être obtenue que par un algorithme d'**apprentissage non supervisé**.

L'**algorithme des  $k$  moyennes** en est un, l'entier  $k$  faisant référence au nombre de classes et non au nombre de plus proches voisins comme dans l'algorithme du chapitre précédent !

# Algorithme des $k$ moyennes

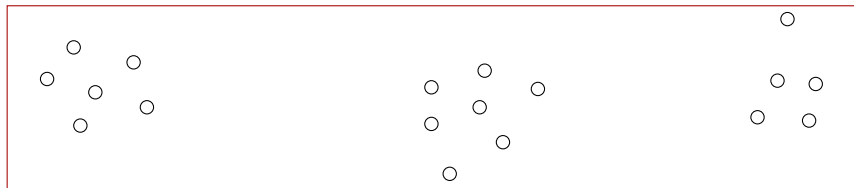
Étant un ensemble de données représentées par des points d'un espace, on cherche à déterminer un nombre de classes  $k$  fixé à l'avance pour classer ces points. Le principe de l'algorithme des  $k$  moyennes est la suivante.

- ◆ Choisir  $k$  points **candidats**, notés  $\mu_0, \mu_1, \dots, \mu_{k-1}$  qui vont représenter la moyenne des points dans chaque classe.
- ◆ Pour chaque point  $p$  des données, choisir la classe  $i$  qui minimise la distance de  $p$  à chaque  $\mu_i$ .
- ◆ Une fois tous les points classifiés, chaque  $\mu_i$  est **mis à jour** en faisant la **moyenne** des points de la classe  $i$ .
- ◆ La procédure précédente est itérée jusqu'à ce que les moyennes n'évoluent plus. Les classifications sont alors établies.

**Comment choisir initialement les  $\mu_i$  ?**

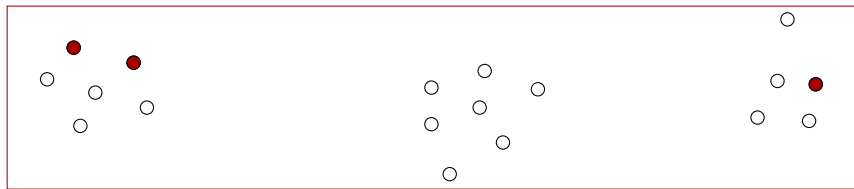
# Algorithme des $k$ moyennes

Illustrons l'algorithme des  $k$  moyennes sur un exemple. Supposons que l'on cherche à classifier les 18 points suivants en  $k = 3$  classes.



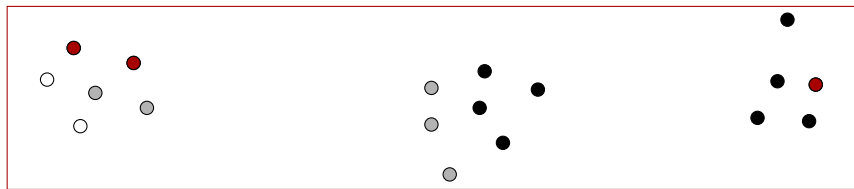
# Algorithme des $k$ moyennes

On choisit trois points au hasard parmi les points des données.



# Algorithme des $k$ moyennes

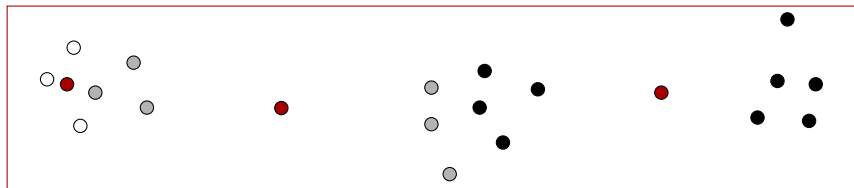
Par un calcul de distances, une première classification affecte trois points à la première classe (en blanc), six points à la deuxième classe (en gris) et neuf points à la troisième classe (en noir).



Sur le schéma, chaque point rouge appartient à l'une des classes.

# Algorithme des $k$ moyennes

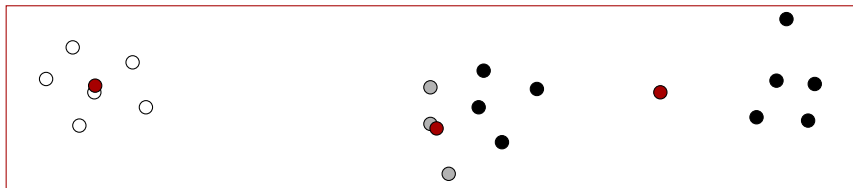
Puis les moyennes sont recalculées et se déplacent (en rouge).





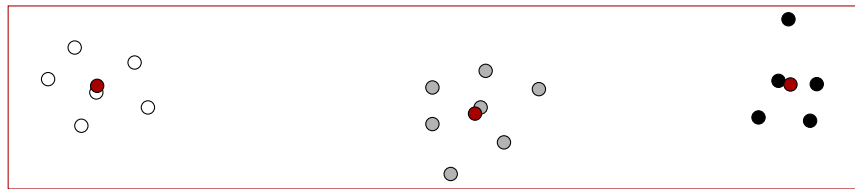
# Algorithme des $k$ moyennes

Une nouvelle étape fait évoluer la classification et les moyennes sont recalculées.



# Algorithme des $k$ moyennes

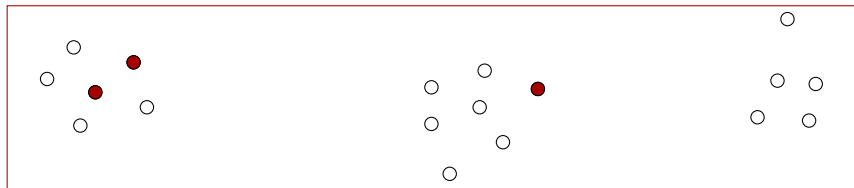
Une dernière étape établit que la classification n'évolue pas.



Ici, le résultat obtenu est conforme à nos attentes.

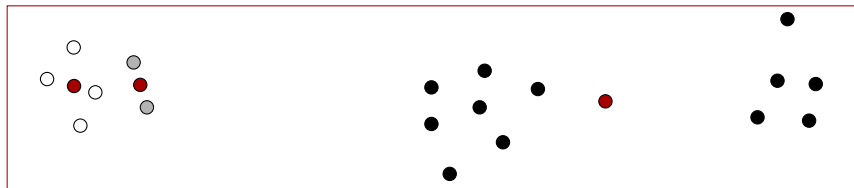
# Algorithme des $k$ moyennes

Mais ça peut ne pas être toujours le cas suivant le choix des valeurs initiales des  $\mu_i$ , comme l'illustre la situation représentée ci-dessous.



# Algorithme des $k$ moyennes

Cette fois, l'algorithme converge en une seule étape vers une classification différente. Deux moyennes se retrouvent associées aux six points du groupe de gauche et la troisième moyenne aux douze points des deux groupes de droite. Le résultat semble clairement moins pertinent que le précédent.



# Mise en œuvre

# Mise en œuvre

Code sur machine

# Convergence

La terminaison de l'algorithme des  $k$  moyennes n'est pas évidente. Bien qu'il n'existe qu'un nombre fini de classifications possibles, à savoir  $k^N$  où  $N$  est le nombre de données, l'algorithme pourrait **a priori** boucler circulairement entre plusieurs classifications différentes.

On peut cependant montrer que ce n'est pas possible : **l'algorithme des  $k$  moyennes converge toujours**. La preuve est hors programme.

En pratique, on n'attend pas systématiquement la convergence de l'algorithme. On se donne plutôt un nombre maximal d'itérations et par ailleurs une proportion minimale d'éléments qui doivent changer de classe à chaque étape.

# Convergence

Utilisons l'algorithme des  $k$  moyennes pour **classifier 10 000 images de caractères manuscrits**, en choisissant  $k = 10$ .



Images représentant de  $28 \times 28$  pixels, en 256 niveaux de gris  
(<http://yann.lecun.com/exdb/mnist/>)

L'algorithme converge au bout de 33 étapes, en quelques secondes. En outre, puisqu'on connaît la classification des images, on peut calculer une sorte de matrice de confusion.



# Convergence

Dans cette matrice, les colonnes correspondent aux classes attribuées par l'algorithme aux différentes images et non au chiffre.

	$c_0$	$c_1$	$c_2$	$c_3$	$c_4$	$c_5$	$c_6$	$c_7$	$c_8$	$c_9$
0	5	423	471	41	3	21	3	3	29	2
1	4	0	0	1	2	1	3	1112	2	2
2	21	1	8	19	662	32	20	173	50	5
3	33	2	4	465	20	9	43	61	379	16
4	543	1	0	0	11	16	3	57	1	348
5	60	9	9	196	1	22	81	211	207	67
6	16	12	8	8	13	852	2	98	5	0
7	346	4	2	0	4	1	5	81	0	627
8	18	5	4	40	8	16	518	104	205	26
9	465	7	2	6	2	3	2	33	13	445

**Ne pas lire** : 423 chiffres 0 ont été identifiés comme des 1

**mais** : 423 chiffres 0 ont été identifiés dans la classe  $c_1$ .

Les images du 1 et du 6 sont plutôt bien identifiés : classe  $c_7$  et  $c_5$ .

Les images du 0 sont séparées en deux classes  $c_1$  et  $c_2$ .