projet_ando

December 14, 2023

1 Projet ANDO

1.1 Projection Orthogonale

Définition :

Soit $\mathcal{E}$ un espace vecteuriel de dimension p.

Si $\mathcal{D}$ est une droite vecteurielle engendrée par le vecteur $\vec{a}$ qui passe par un point Q de $\mathbb{R}^p$, l'ensemble des vecteurs orthogonaux à $\mathcal{D}$ est un hyperplan appelé hyperplan normal à $\mathcal{D}$ et défini par :

$$\mathcal{D}^{\perp} = \left\{ \vec{h} \in \mathbb{R}^p \mid (\vec{h} \cdot \vec{a}) = 0 \right\}$$

Si x est un point arbitraire de $\mathbb{R}^p$ et si on note $\vec{x}$ le vecteur associé qui va de Q à ce point, on peut toujours le décomposer de la façon suivante :

$$\vec{x} = \vec{x}_{\mathcal{D}} + \vec{x}_{\perp} \text{ avec } \vec{x}_{\mathcal{D}} = \frac{(\vec{x} \cdot \vec{a})}{\|a\|^2} \vec{a}$$

Si on note $x_{\mathcal{D}}$ la projection du point sur la droite $\mathcal{D}$ et si on note x_i la ième composante du point x, on obtient alors les coordonnées du point $x_{\mathcal{D}}$:

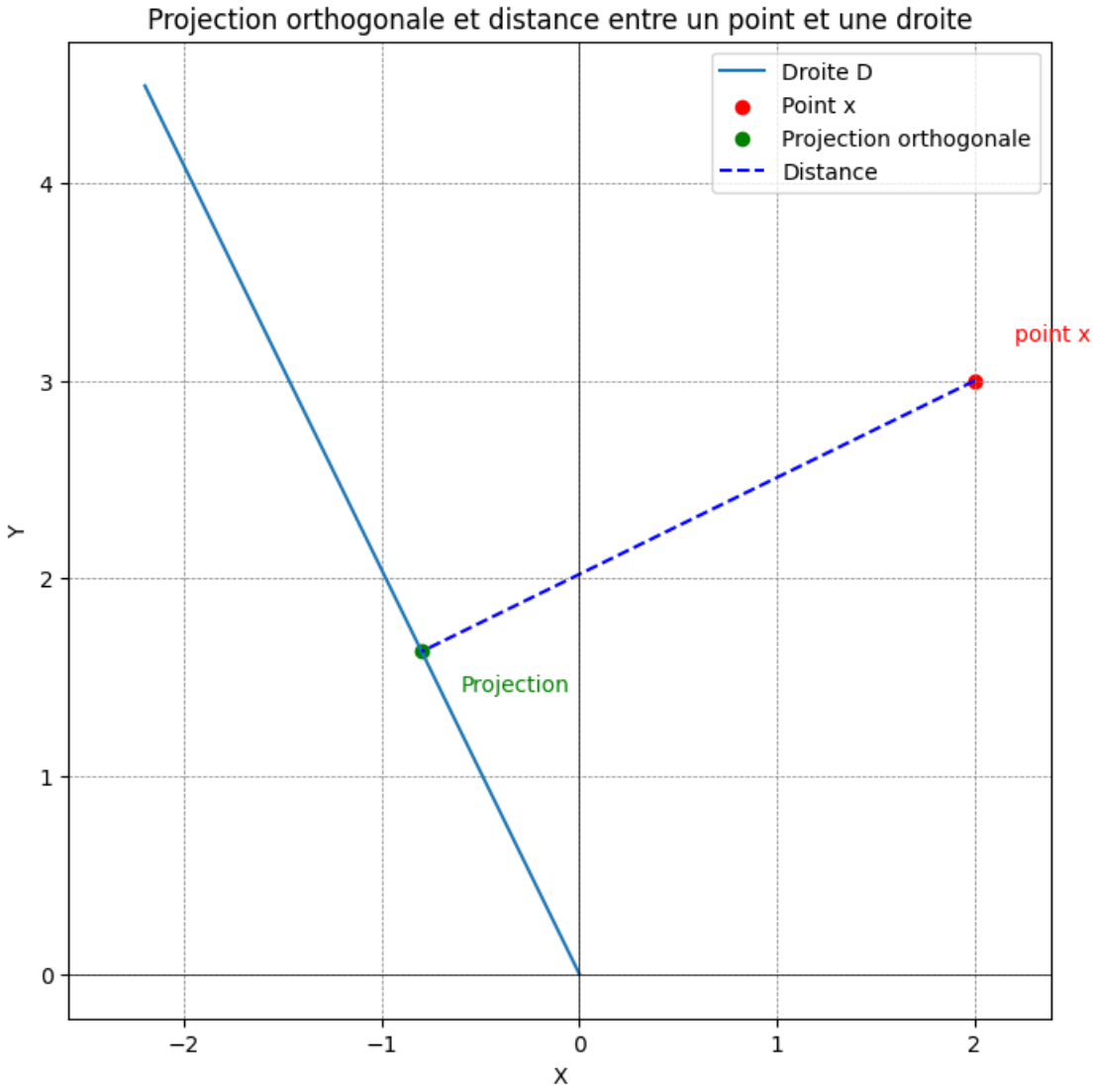
$$\forall i \in [1; p], x_{\mathcal{D}_i} = Q_i + \frac{\sum_{k=1}^{p} (x_k - Q_k) * a_k}{\|a\|^2} * a_i$$

Pour avoir la distance entre le point x et la droite $\mathcal{D}$, on a besoin de:

$$\|\vec{x}_{\perp}\| = \|(x_1 - x_{\mathcal{D}_1},\ x_2 - x_{\mathcal{D}_2},\ ...,\ x_p - x_{\mathcal{D}_p})\|$$

Dans la suite de ce document, on choisira de représenter une droite dans $\mathbb{R}^p$ par un de ses vecteurs directeurs unitaires, noté $\vec{u}$.

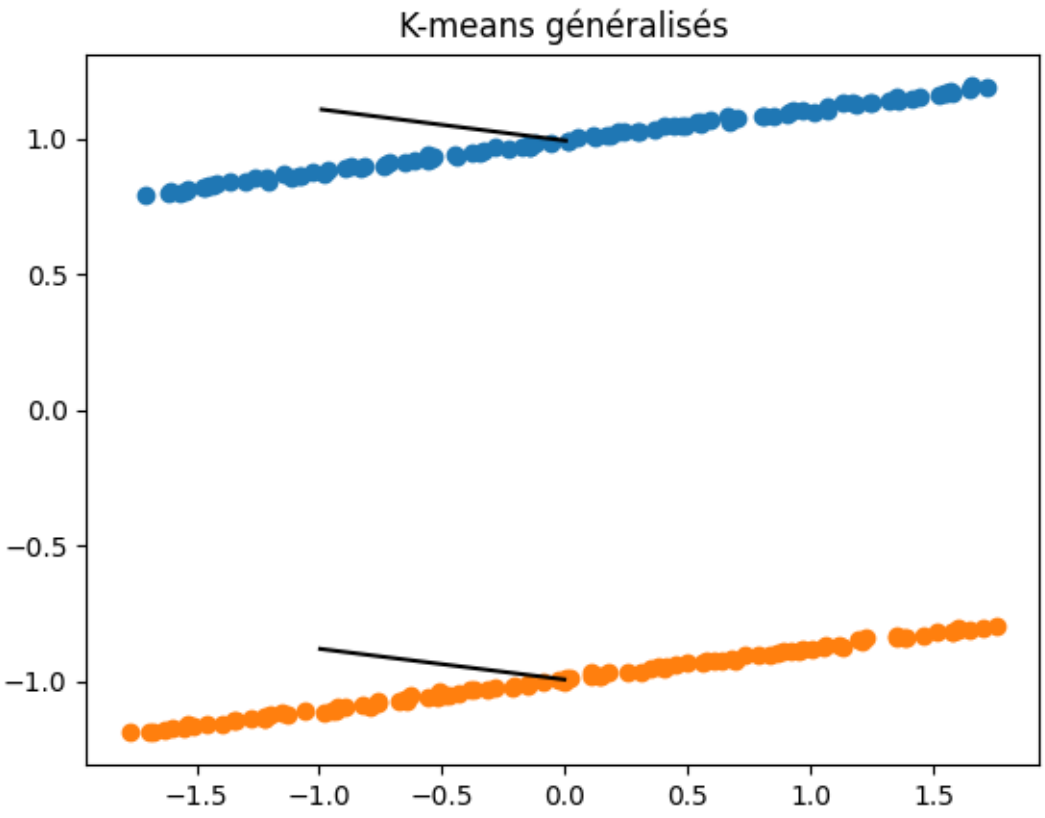
```
Coordonnées de la projection orthogonale: [-0.79802776  1.63075237]
Distance entre le point et la droite: 3.1150920360380336
```



2 Les nuées dynamiques ou les kmeans généralisés

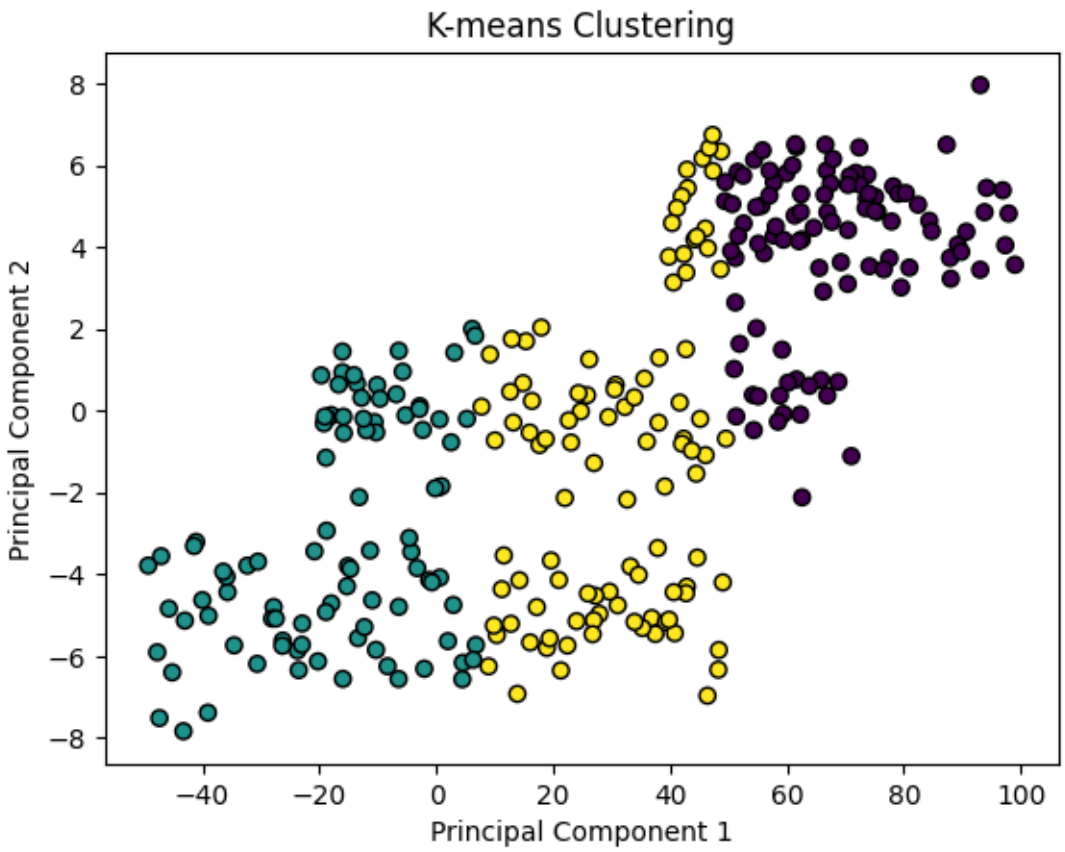
3 Exemples test

```
Convergence atteinte à l'itération 2
Classe 1 - Représentant : Droite(Passe par le point Point([0.00281384
0.99339162]), vecteur directeur [-0.99337191  0.11494457])
Classe 2 - Représentant : Droite(Passe par le point Point([-0.00281384
-0.99339162]), vecteur directeur [-0.99360973  0.11287029])
```




```
c:\Users\noahk\AppData\Local\Programs\Python\Python312\Lib\site-
packages\sklearn\cluster\_kmeans.py:1416: FutureWarning: The default value of
`n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init`
explicitly to suppress the warning
  super()._check_params_vs_input(X, default_n_init=10)
```

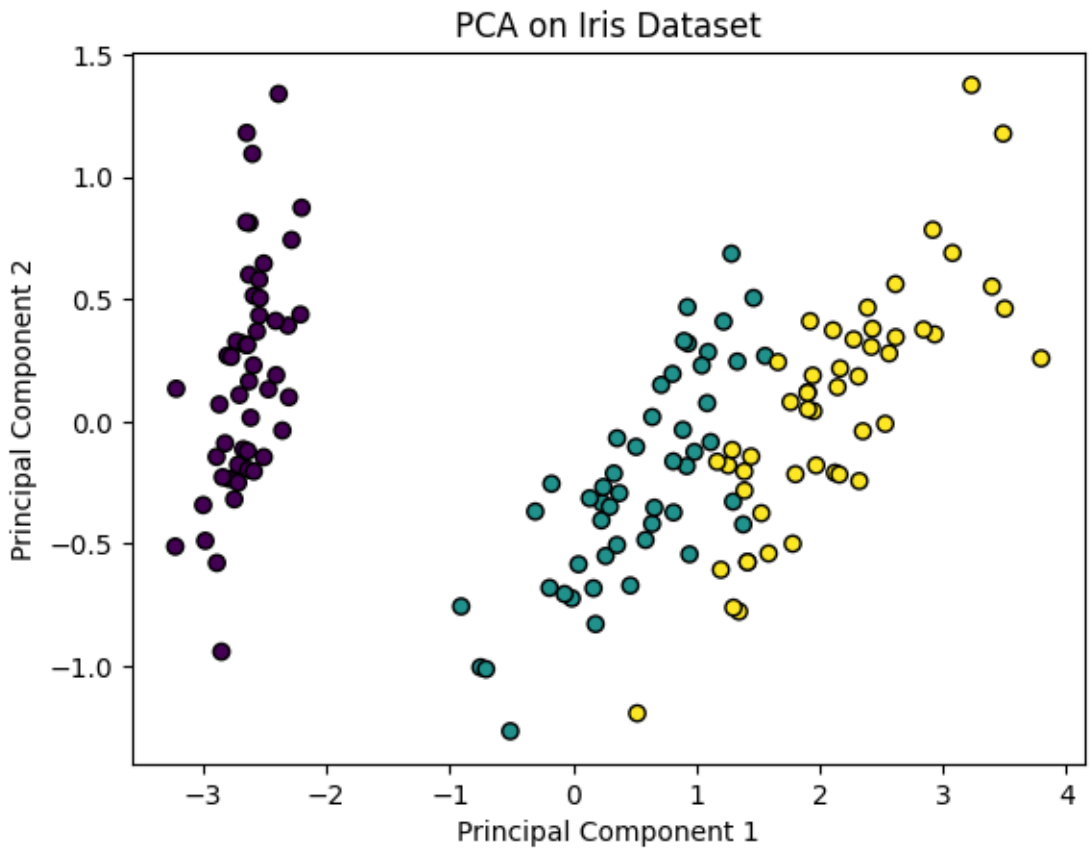
```
Text(0, 0.5, 'Principal Component 2')
```



4 Application au dataset iris

Nous allons dans cette partie tester notre algorithme des K-means généralisés sur le dataset *iris* et le comparer à deux autres modèles de clustering classiques, à savoir les K-means classiques et un modèle de mélange gaussien.

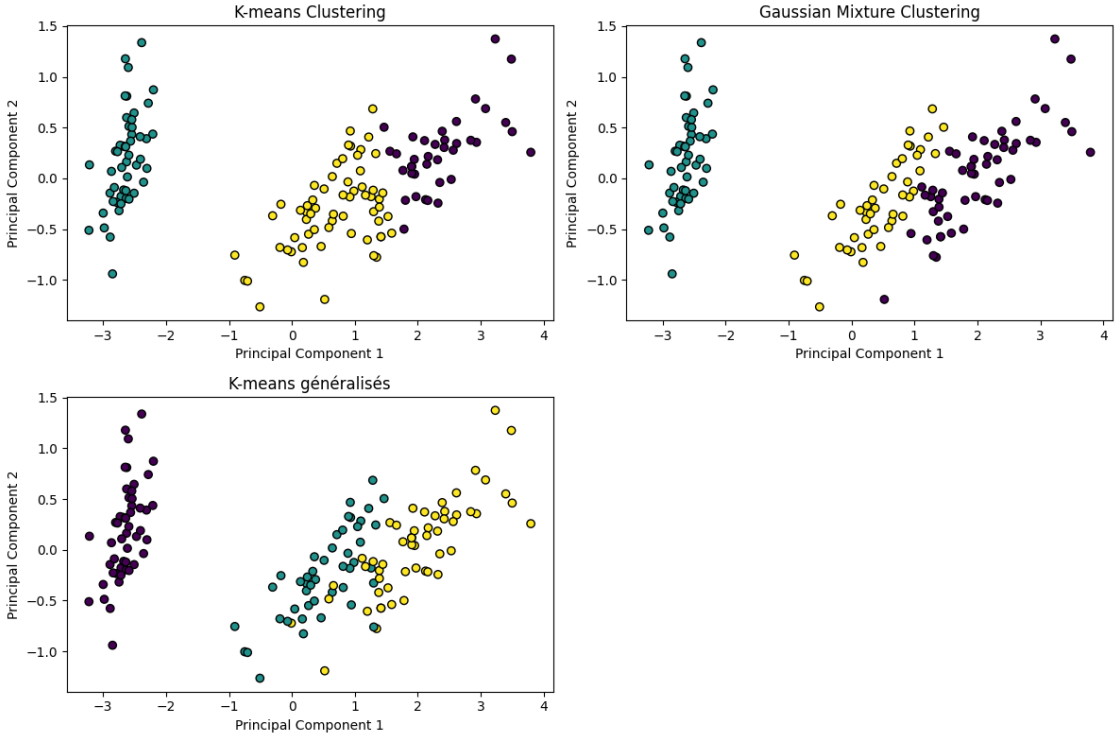
Dans un premier lieu, voici une représentation du dataset en 2 dimensions à l'aide d'une Analyse en Composantes Principales, avec chaque cluster représenté d'une couleur différente :



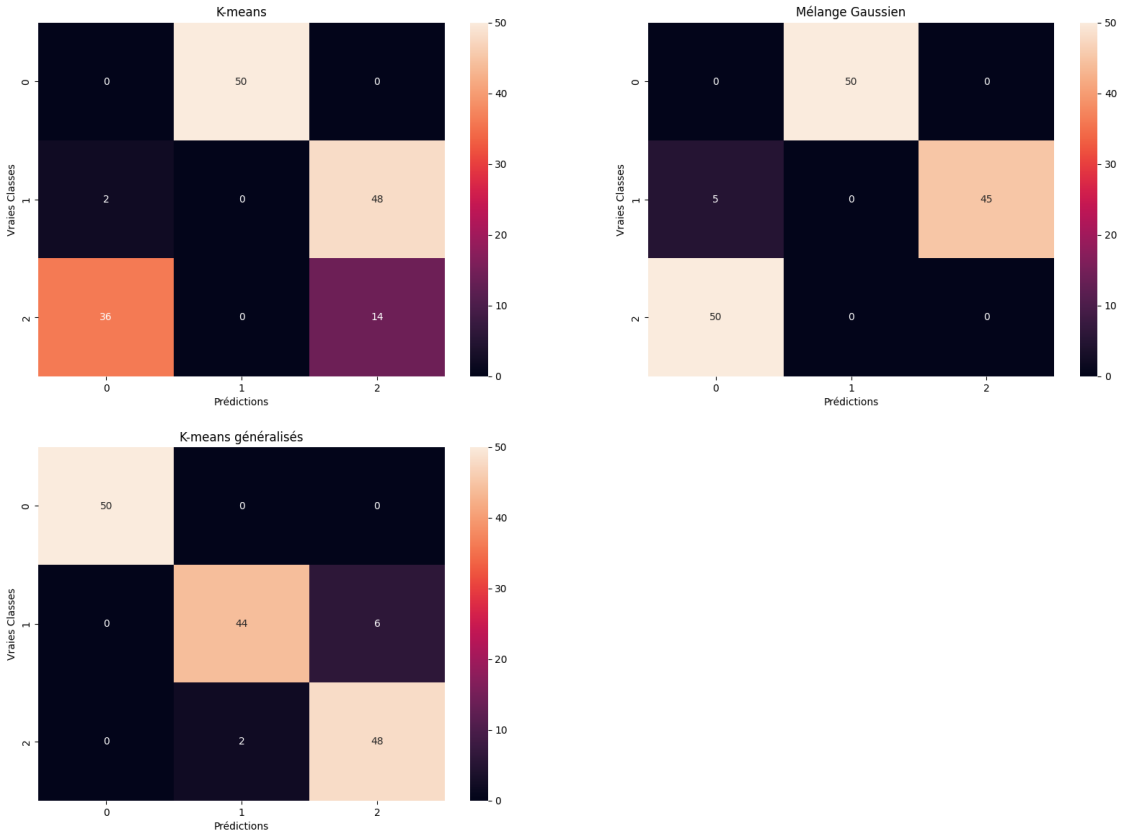
Nous observons qu'il y a exactement 3 clusters, ayant chacun une tendance de dispersion assez linéaire, et dont 2 sont assez proches l'un de l'autre. Voyons donc maintenant ce que les différentes méthodes de clustering donnent comme résultats.

Voici une visualisation des différents clustering :

```
Convergence atteinte à l'itération 14
```



Et voici leur puissance prédictive respective :



On observe donc que les deux versions des K-means sont assez précises, environ 90% pour le classique, et 94% pour le généralisé.

Cependant, le modèle de mélange gaussien est beaucoup plus précis, puisque seulement 5 fleurs sont mal classées, soit une précision de 96,6% !

Ceci pourrait s'expliquer par le fait que les algorithmes de K-means sont très sensibles à l'initialisation des clusters, qui sont déterminants.