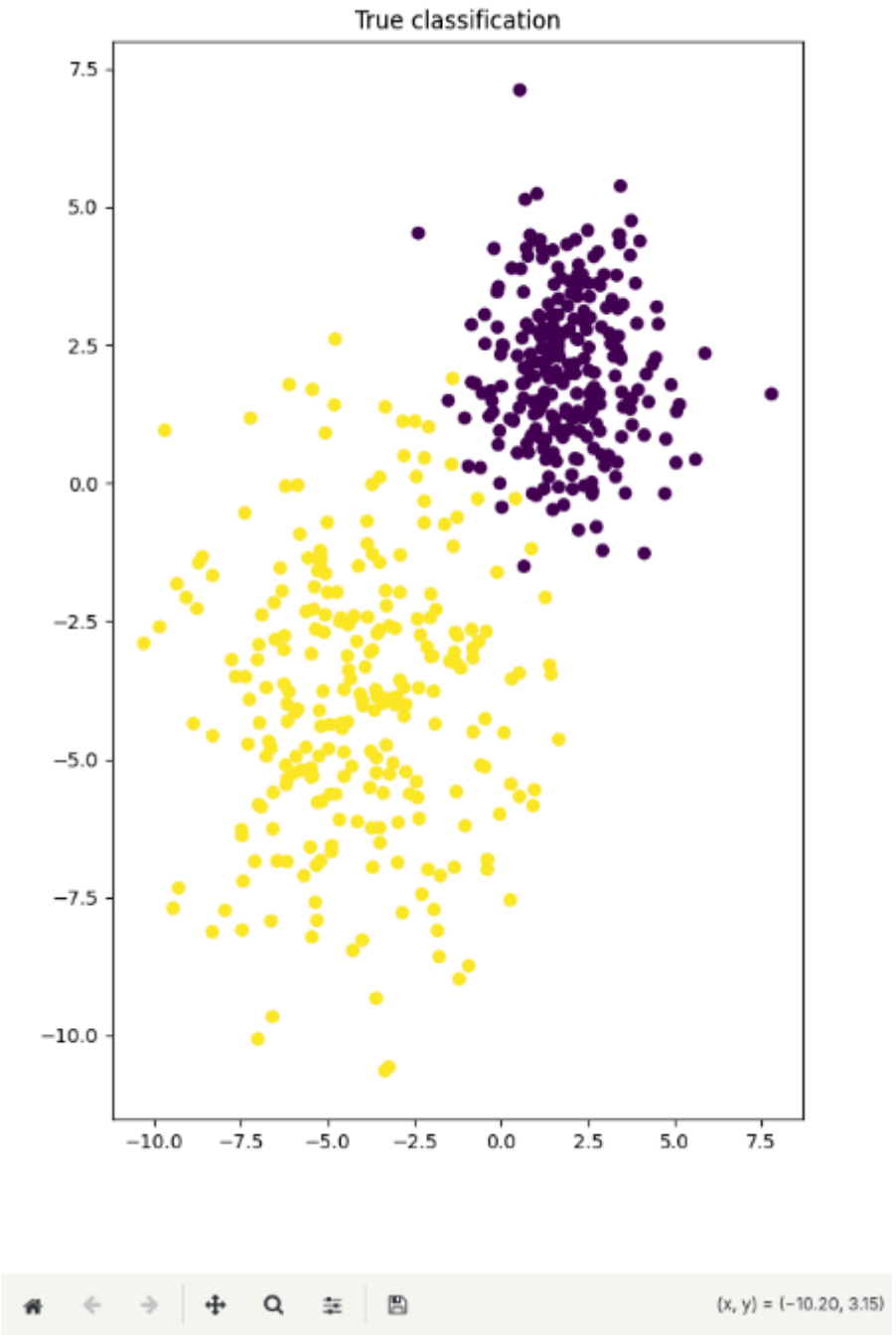
Rapport TP1 Analyse de donnée

Ex1

1.1

Q1.



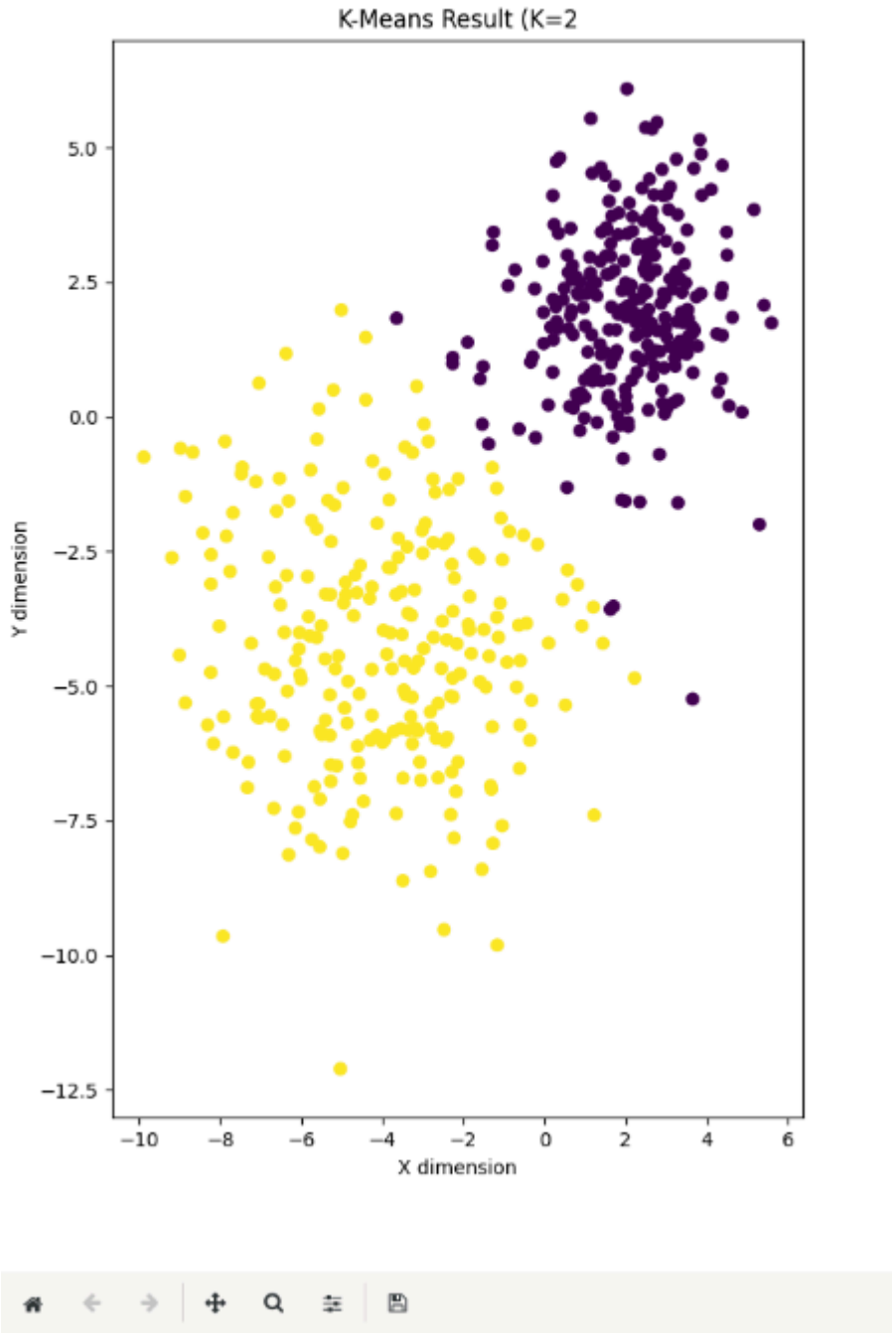
Donnée originale

1.2

Q2.

La valeur `kmeans.labels_` est une liste d'entier identifiant a quel cluster chaque valeur appartient.
Ici, avec 512 entités dans notre modèle de test, la taille de labels_ vaut 512, et chacun des valeurs correspond à l'indice du cluster auquel le point ciblé appartient.

Q3.



Donnée calculée avec la méthode KMean

On voit que les deux clusters sont très similaires, bien qu'il y a quelques imprécisions dans le cluster KMeans

Q4.

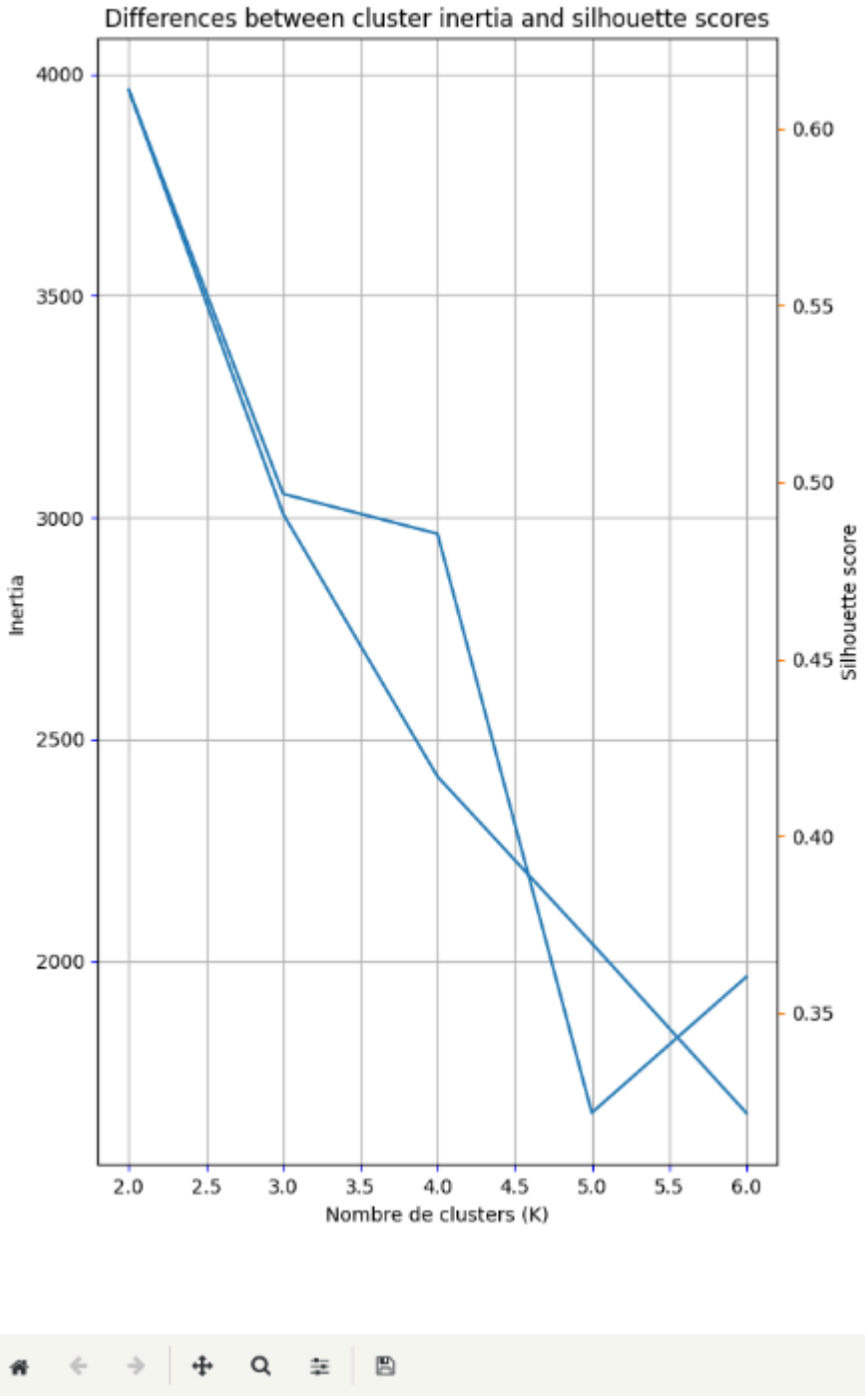
En moyenne, le résultat de `adjusted_rand_score(true_labels, kmeans.labels_)` vaut 0.85, ce qui correspond à une forte similarité entre les deux clusters.

Q5.

`n_init` sert à définir le nombre d'itérations que sklearn doit effectuer pour calculer la meilleur version du clustering
Plus n_init est grande, plus on réduit les chances d'avoir une simulation imprécise à cause d'un mauvais départ.

1.2.2

Q3.



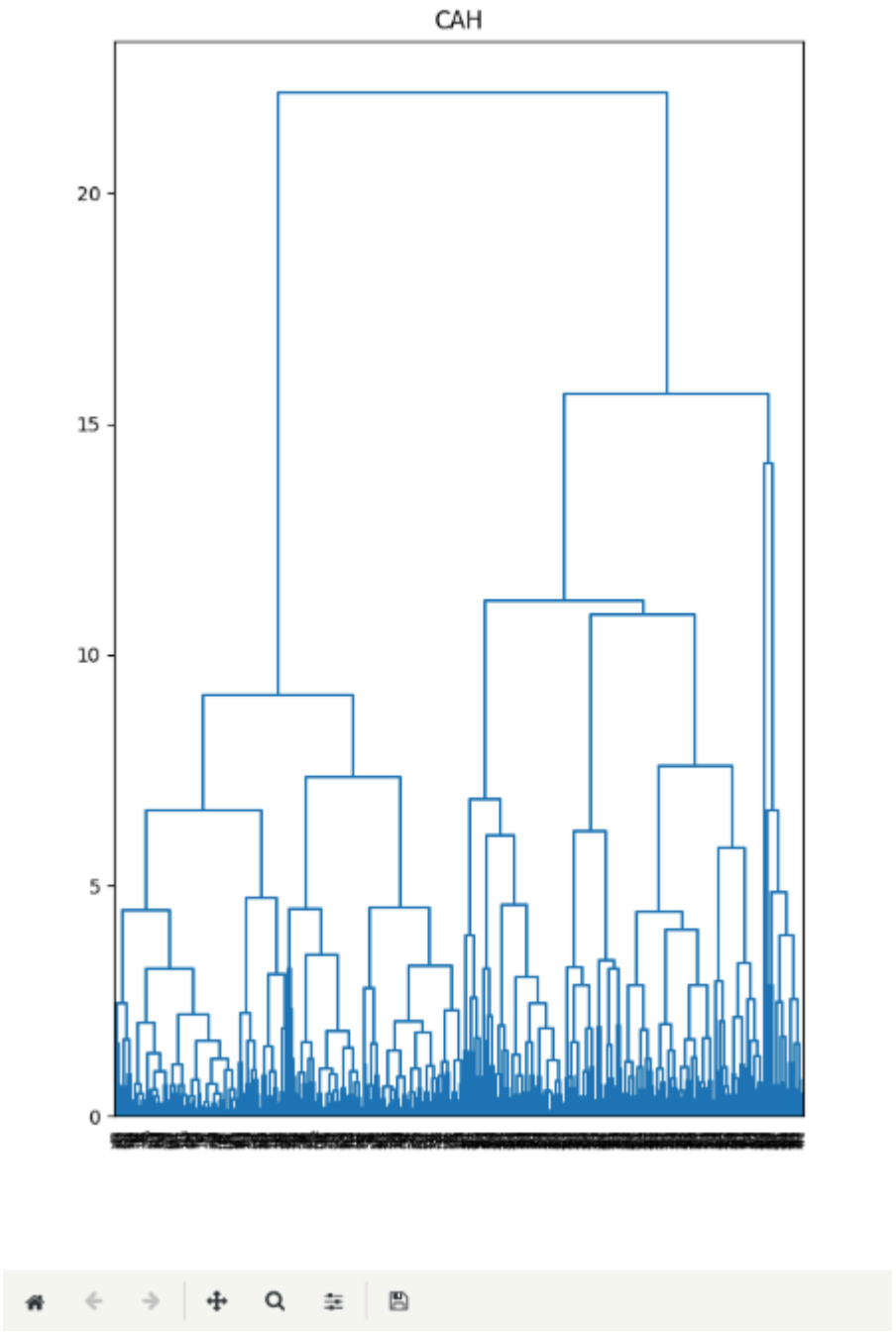
Graphe d'inertie et de silhouette

Q4.

D'après le graphe, on voit que K=2 est la bonne valeur
Ce n'est pas évident avec l'inertie, mais la silhouette est au plus haut a K=2, et chute ensuite

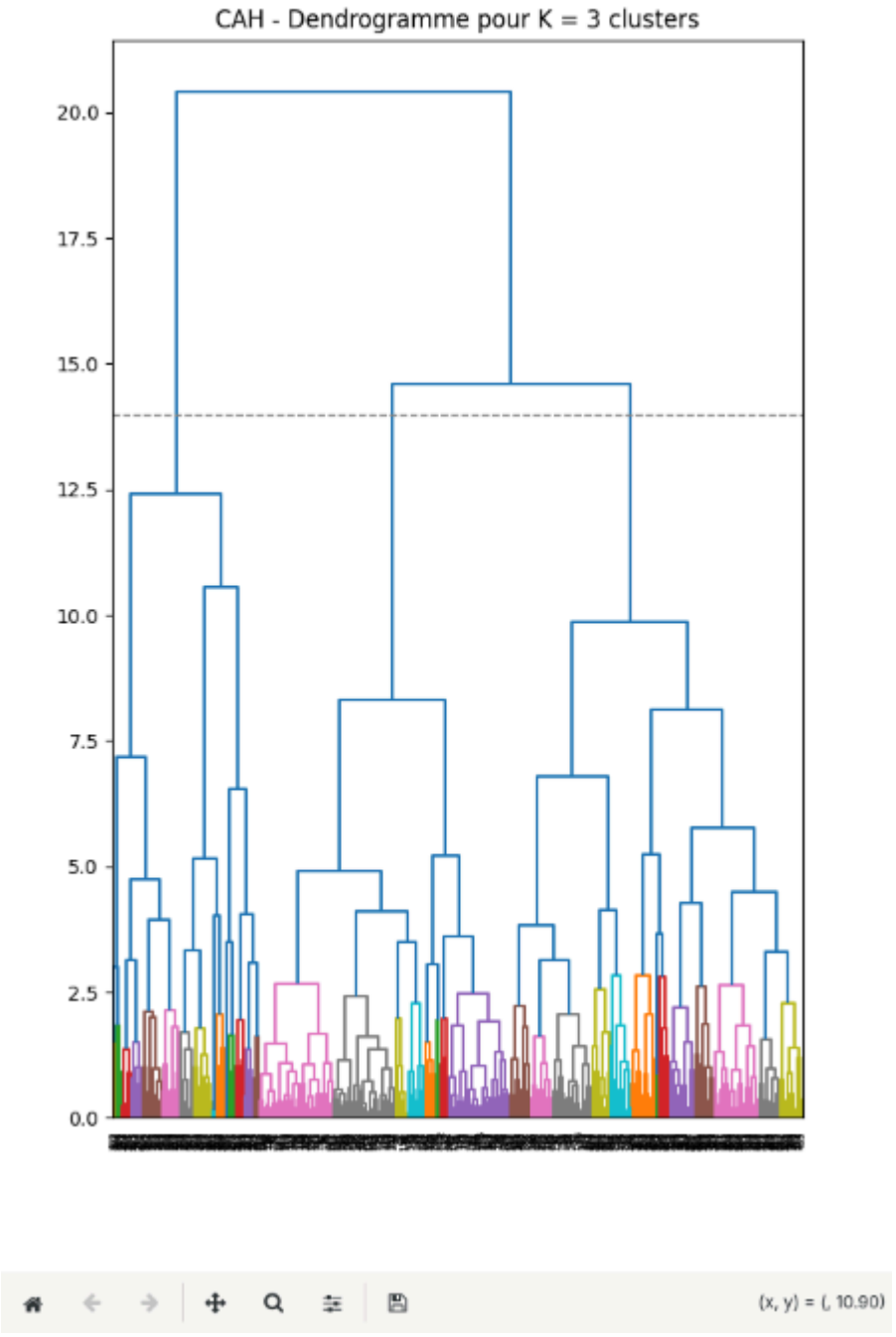
1.3

Q2.



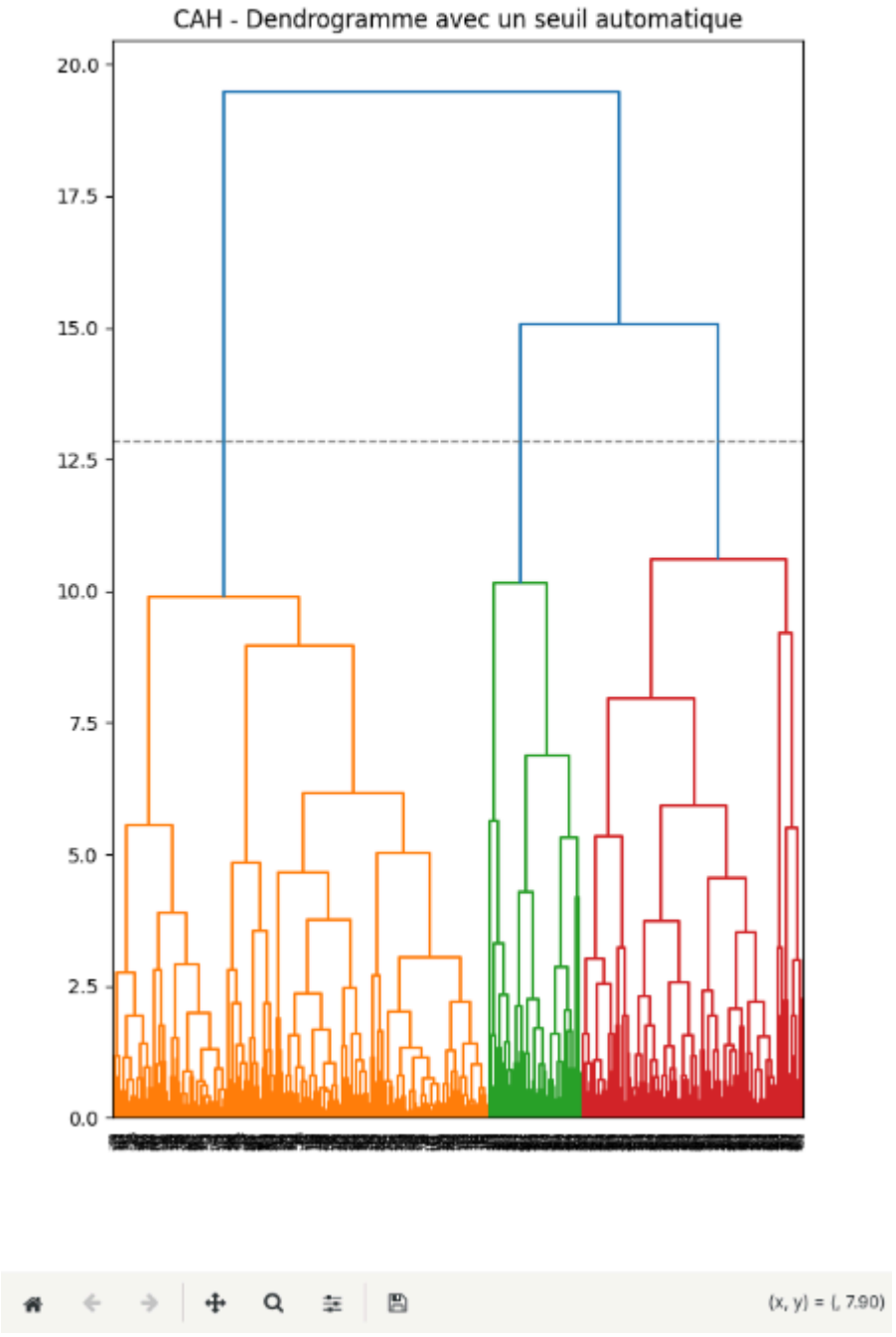
Dendrogramme sans seuil

Q3.



Dendrogramme avec seuil = 3

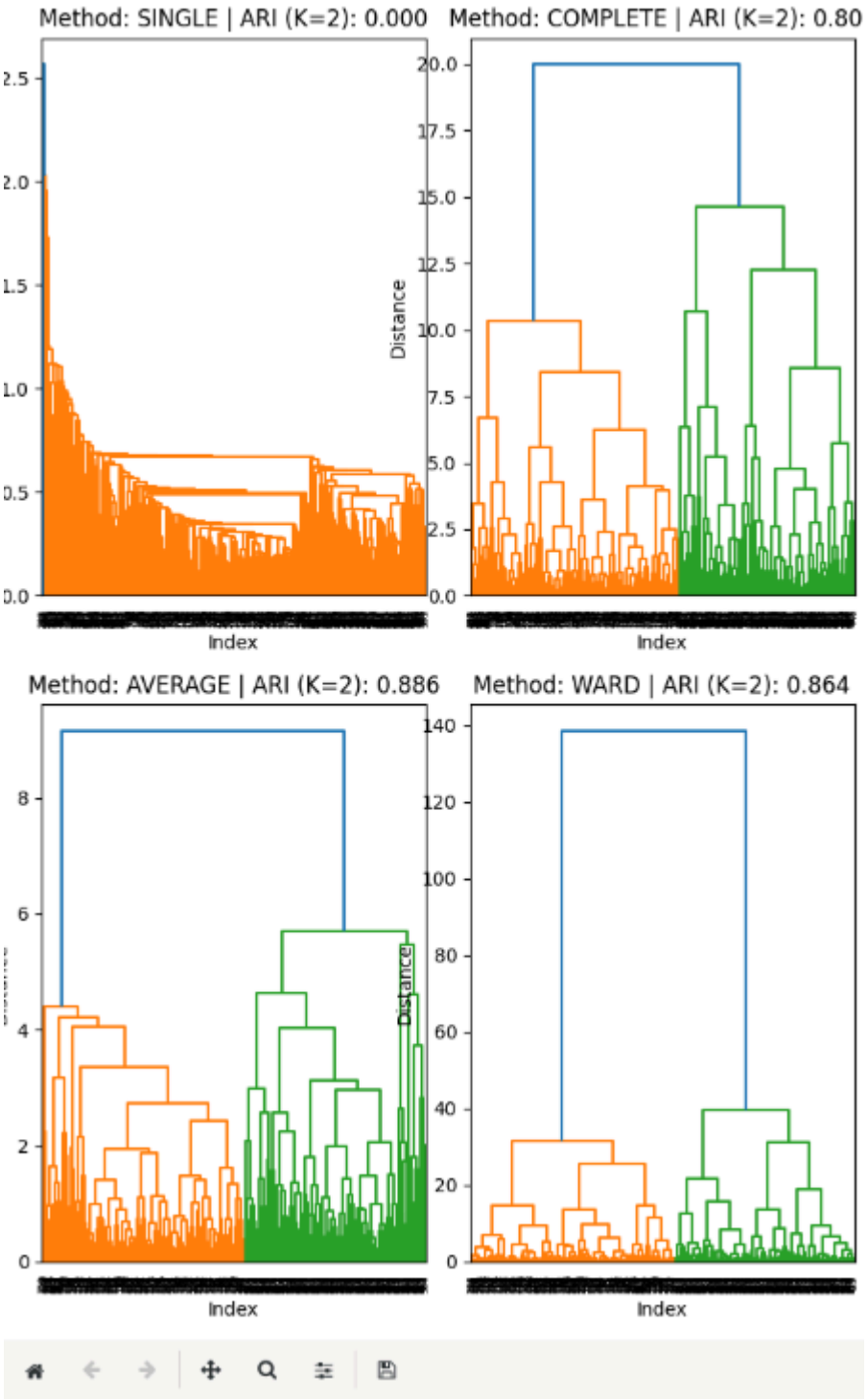
Q4.



Dendrogramme avec seuil automatique

Pour calculer le seuil de façon automatique, on utilise le résultat du linkage, en récupérateur les valeurs à l'index -2 et -3 et en calculant leur moyenne. Cette moyenne est la valeur intermédiaire permettant de couper le dendrogramme en 3 clusters.

Q6.



Dendrogramme de comparaison entre les 4 méthodes de classification

1. Method: SINGLE

- **Score ARI :** `0.000` (C'est le pire score possible, équivalent à un classement aléatoire).

- **Analyse :** On voit un "effet d'escalier" très prononcé. Au lieu de créer deux beaux groupes distincts, l'algorithme a isolé de tout petits points un par un sur la gauche, et a gardé un immense groupe informe (le gros bloc orange).

2. Method: COMPLETE

- **Score ARI :** `0.801`.
- **Analyse :** L'arbre est bien séparé en deux branches principales (Bleu/Orange et Vert).

3. Method: AVERAGE

- **Score ARI :** `0.886` (Le meilleur score).
- **Analyse :** L'arbre est très équilibré.

4. Method: WARD

- **Score ARI :** `0.864` (Excellent score).
- **Visuel :** L'arbre a la forme la plus équilibré. Les deux grands groupes sont séparés très haut (la barre horizontale bleue), et les sous-groupes sont très bas. Cela indique des clusters très cohérents.

Exercice 2

2.1

Q2.

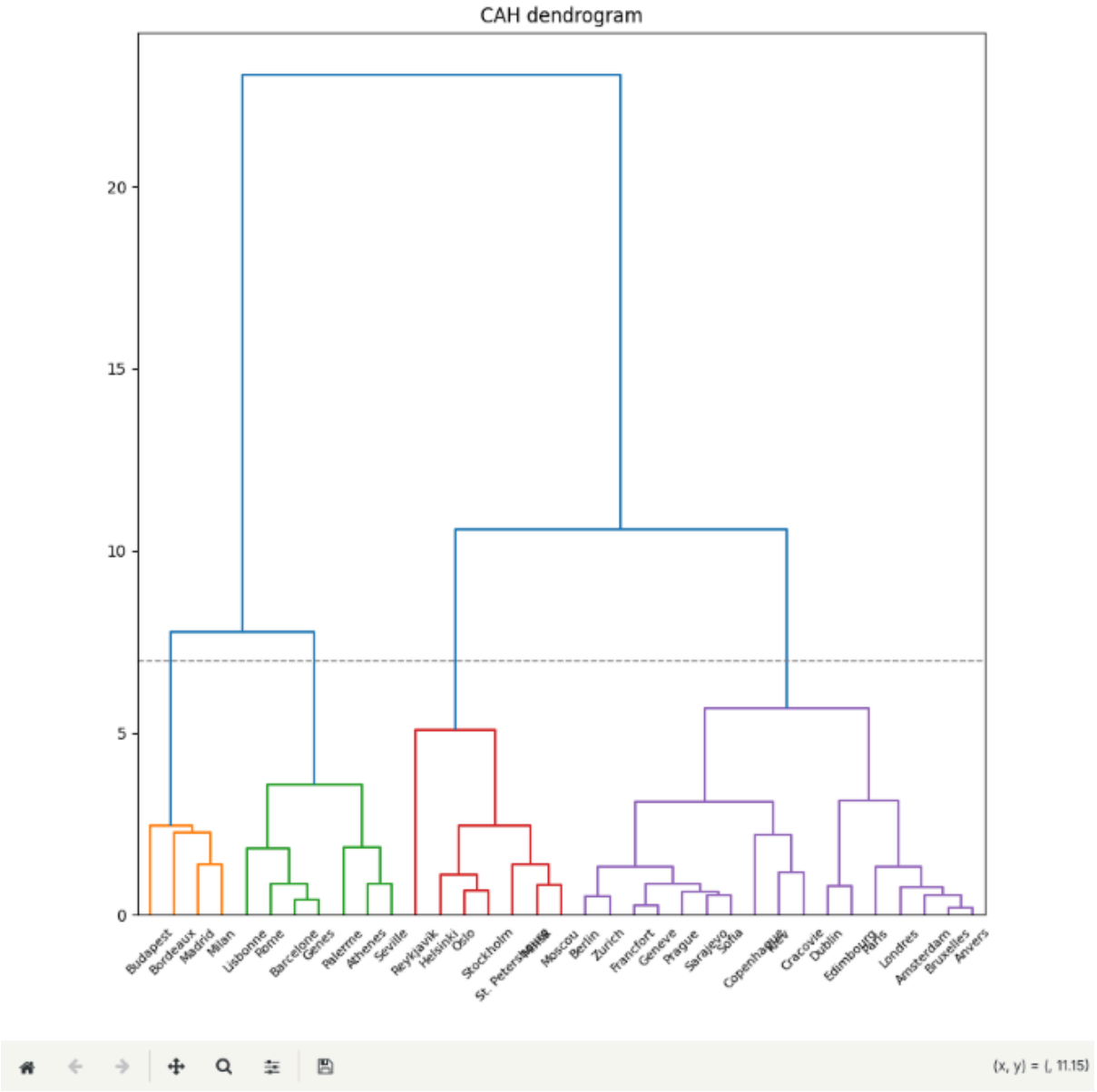
On voit qu'avec une valeur de K faible, il y a peu de différences dans les couleurs, et donc une grosse perte de qualité et de précision
Avec une valeur de K élevé (>64), les différences deviennent indiscernables

Exemple avec K=2

Exemple avec K=64

2.2

Q3.

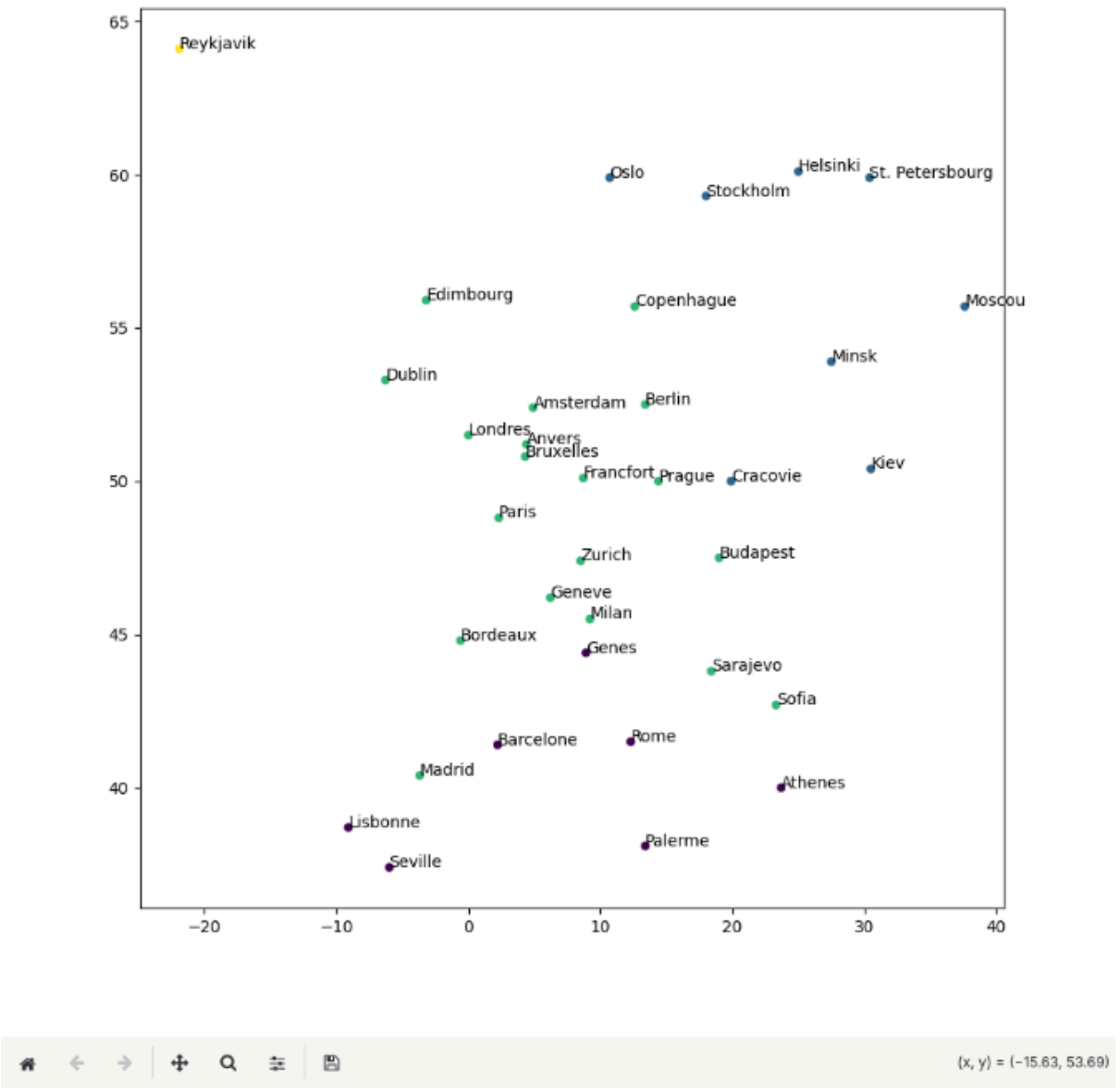


Avec seuil = 4, on voit que les clusters sont découpés convenablement

Q3.

La première classe contient 4 éléments,
La deuxième 7,
La troisième 7,
La quatrième 17

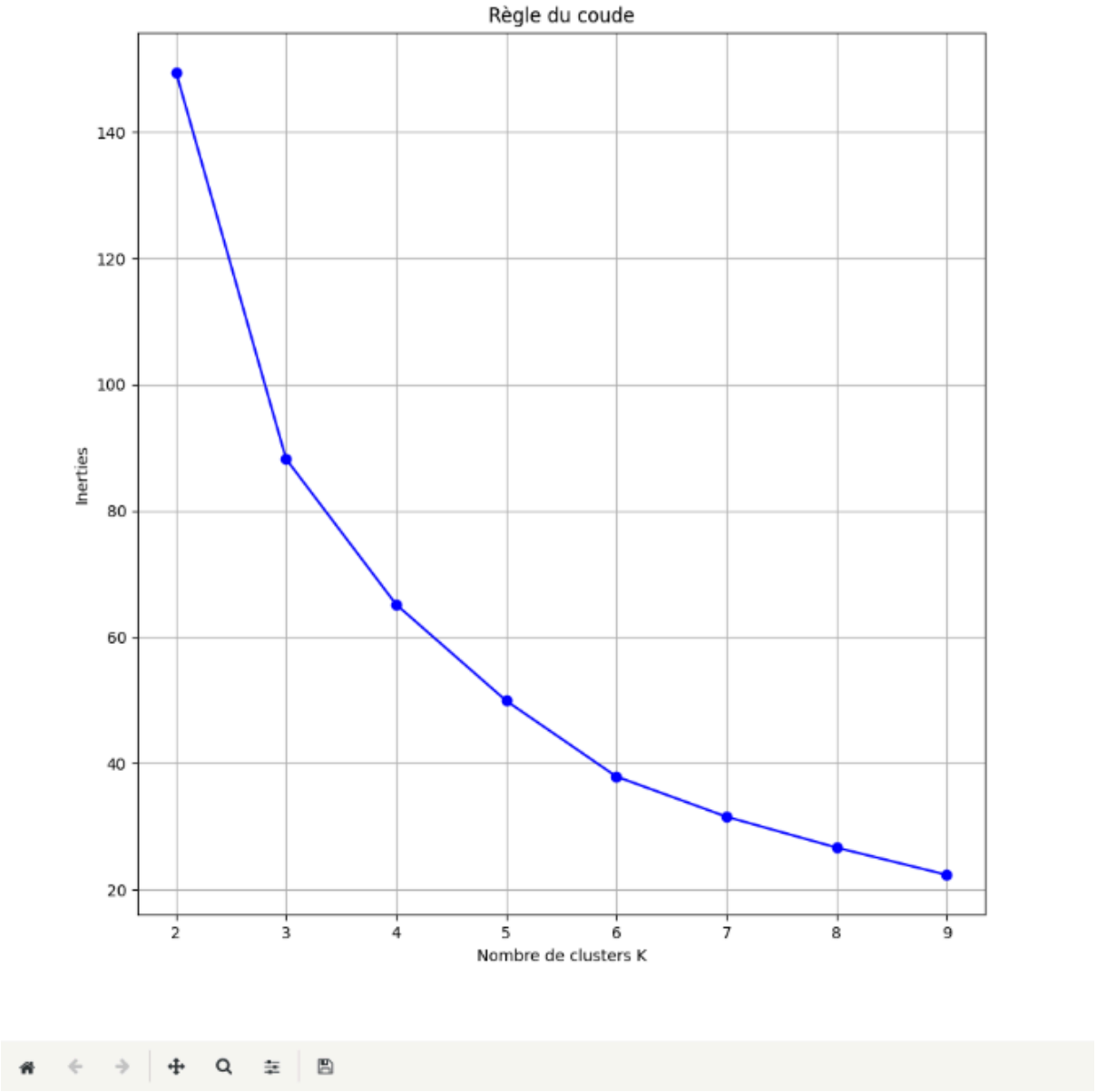
Q4.



Représentation avec une autre mesure de dissimilarité (average)

2.3

Q2.

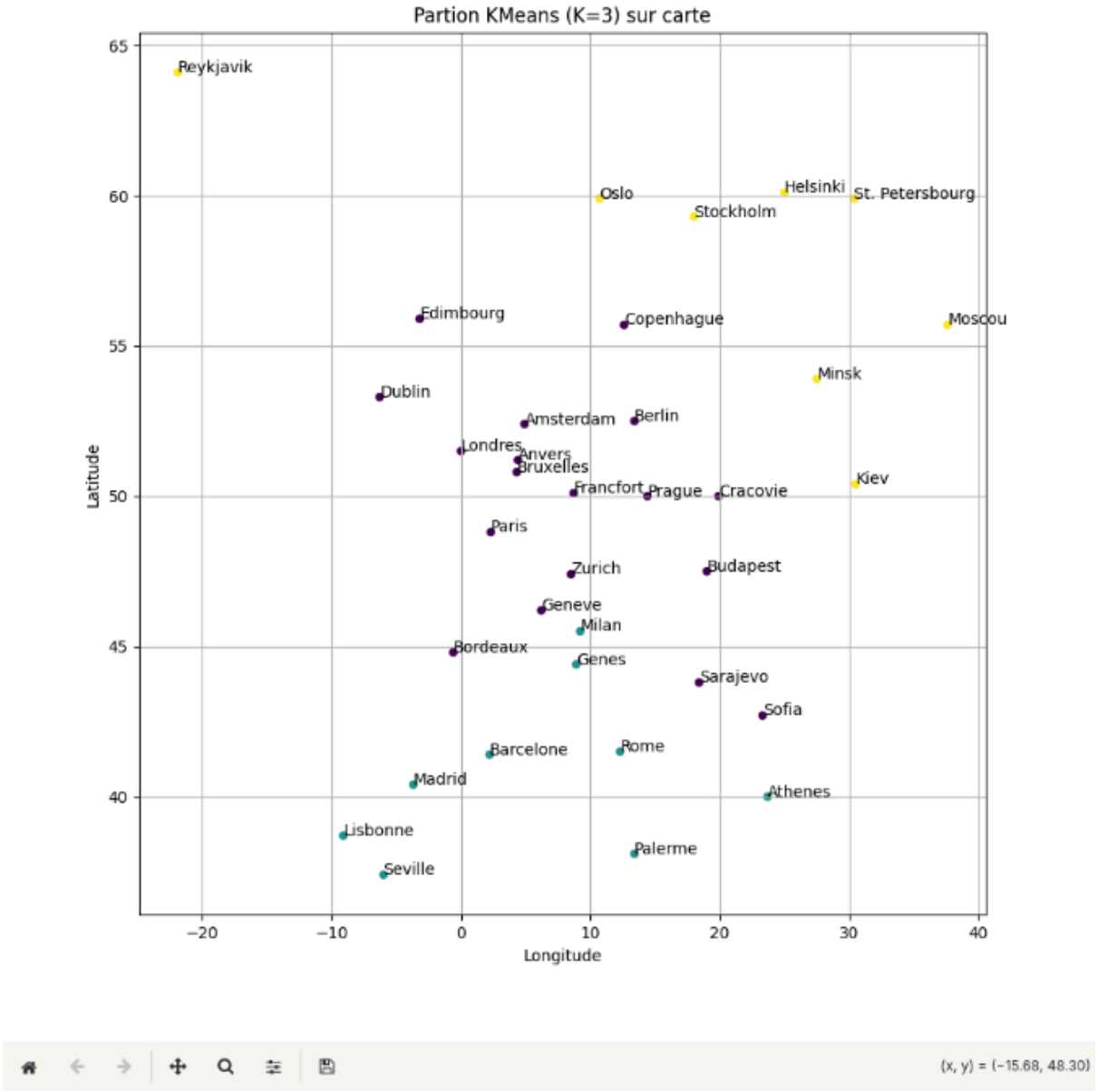


Graphe montrant la régle du coude appliquée au cluster des températures

On voit que la chute la plus brutale se fait à K = 3

On peut donc en conclure que la valeur de K optimale est 3

Q3.



Graphe des partitions en fonction des données géographique

2.4

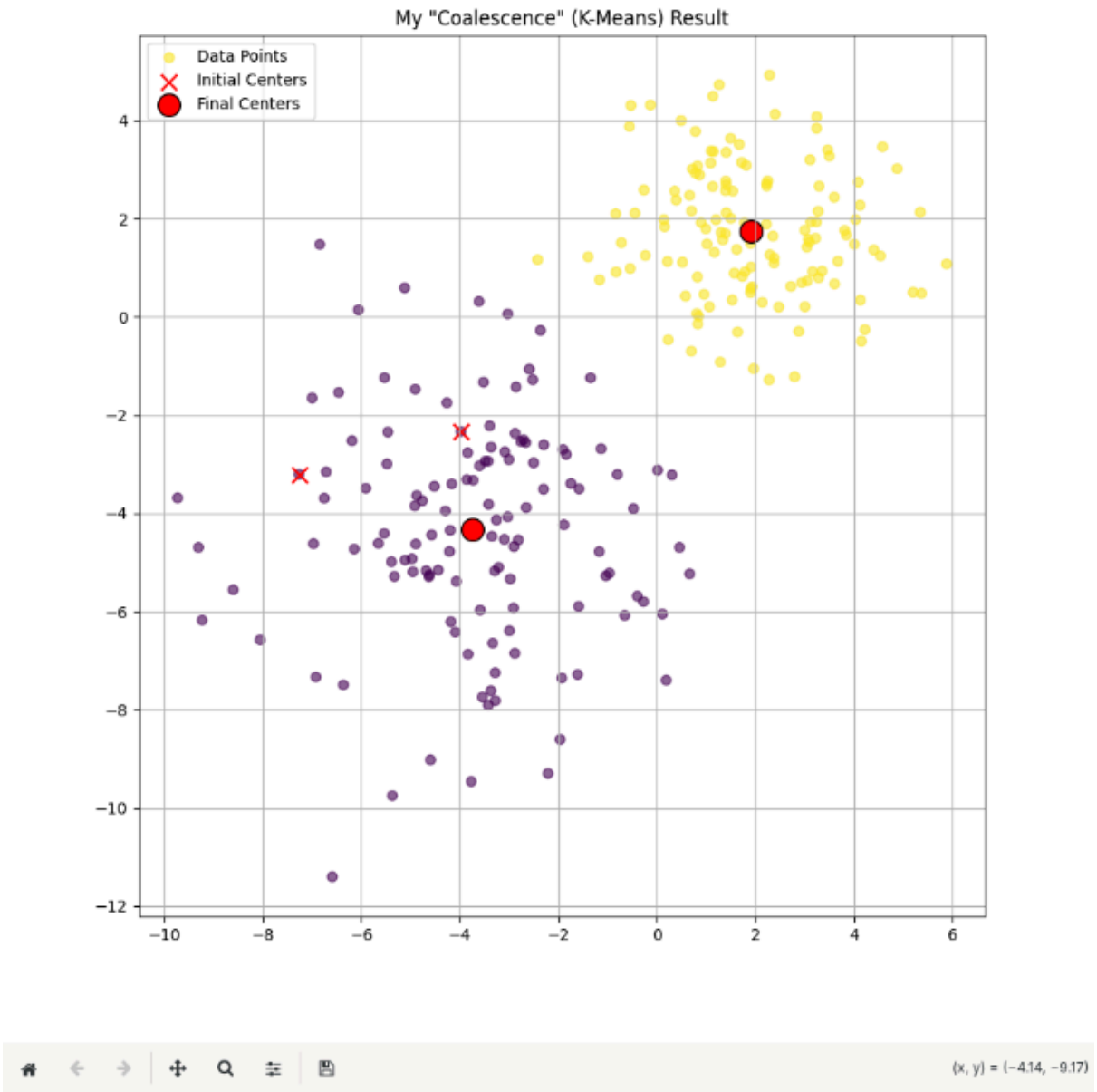
Q1.

	CAH			
KMeans	\	0	1	2
	0	2	2	0
	1	0	7	0
	2	0	0	7
	3	16	0	1

Tableau de comparaison entre CAH et KMeans

Ex 3

Résultat de l'algorithme de coalescence :



Après plusieurs tests, on voit que l'algorithme produit des résultats précis et reproductibles, avec peu d'erreurs.