Rapport de Travaux Pratiques : Clustering par l'Algorithme des K-Moyennes

TCHATCOUA-TCHOUAMO Michel Peslier
IADS 3

4 novembre 2025

Table des matières

1 Introduction et Rappel de l'Algorithme

Ce rapport présente l'application et l'analyse de l'algorithme de clustering non supervisé des K-Moyennes (K-means) sur des données synthétiques et sur la base de données de chiffres manuscrits (Digits). L'objectif est d'évaluer la performance de l'algorithme, de déterminer le nombre optimal de clusters et d'utiliser les centroïdes pour une tâche de classification.

L'algorithme K-means vise à partitionner n observations en K clusters, où chaque observation appartient au cluster dont la moyenne (centroïde) est la plus proche.

2 Évaluation de l'Algorithme sur Données Synthétiques

Les expérimentations sont menées sur les jeux de données bidimensionnels `base1.txt` et `base3.txt`.

2.1 Analyse descriptive des bases de données : Base1 et Base3

Avant toute application de l'algorithme des k-moyennes, il est essentiel de comprendre la nature des données à traiter. Les fichiers .txt fournis donnent les informations suivantes :

- Base1 contient 300 points répartis en 3 classes réelles.
- Base3 contient 600 points répartis en 4 classes réelles.
- Les données sont représentées en 2 dimensions (2D), ce qui permet une visualisation directe des regroupements potentiels.

Dès la première observation, on constate que Base1 présente des clusters bien séparés, tandis que Base3 montre une structure plus complexe, avec des zones de recouvrement entre certaines classes.

2.2 Visualisation des données : Base1 et Base3

Avant d'appliquer l'algorithme des k-moyennes, il est essentiel d'observer la distribution initiale des données afin d'avoir une première idée sur leur structure et le nombre potentiel de groupes naturels qu'elles pourraient contenir. Les visualisations suivantes présentent la répartition des points pour les bases de données Base1 et Base3.

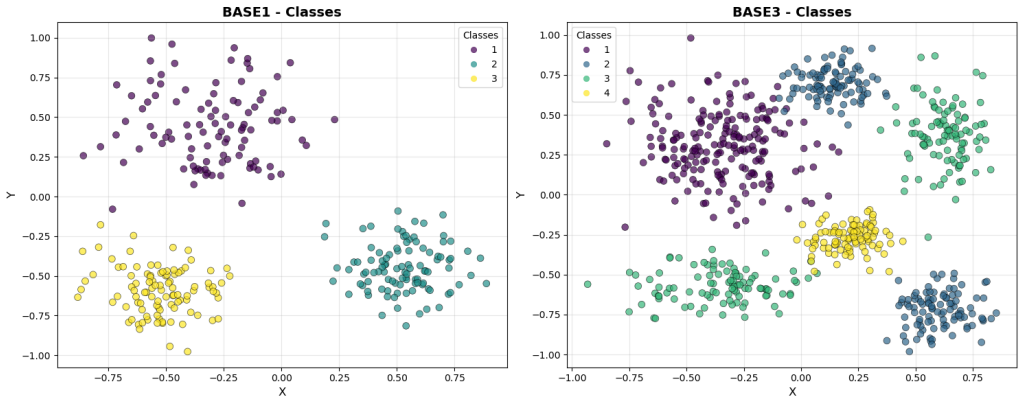

FIGURE 1 – Visualisation des données de `base1.txt` et `base3.txt`.

Lorsqu'on visualise ces données :

- La Base1 confirme visuellement la présence de 3 groupes distincts, clairement délimités.
- Pour la Base3, on remarque que deux classes semblent se subdiviser chacune en deux sous-groupes, donnant ainsi l'impression d'avoir jusqu'à 6 classes potentielles.
- Sur la Base1, la répartition des points montre une structure relativement claire composée de plusieurs groupes distincts, bien séparés les uns des autres. Cette configuration suggère que l'algorithme des k-moyennes devrait parvenir à regrouper efficacement les données avec un nombre de clusters modéré (autour de 3 à 5).

— En revanche, la Base3 présente une distribution plus complexe, avec des amas moins nettement délimités et parfois des zones de recouvrement entre les points. Cela rend le choix du nombre de clusters **k** plus délicat et souligne l'intérêt d'utiliser des critères objectifs tels que la méthode du coude ou le score de Calinski-Harabasz pour déterminer la valeur optimale de **k**.

Ces visualisations permettent donc d'obtenir une première intuition sur la structure interne des données avant d'appliquer les méthodes de regroupement.

2.3 Analyse du Coût (Inertia) et Visualisation (`base1.txt`)

L'algorithme K-means a été appliqué sur `base1.txt` pour différentes valeurs de K. Le coût, ou Inertia (somme des variances intra-clusters), est la métrique principale d'évaluation.

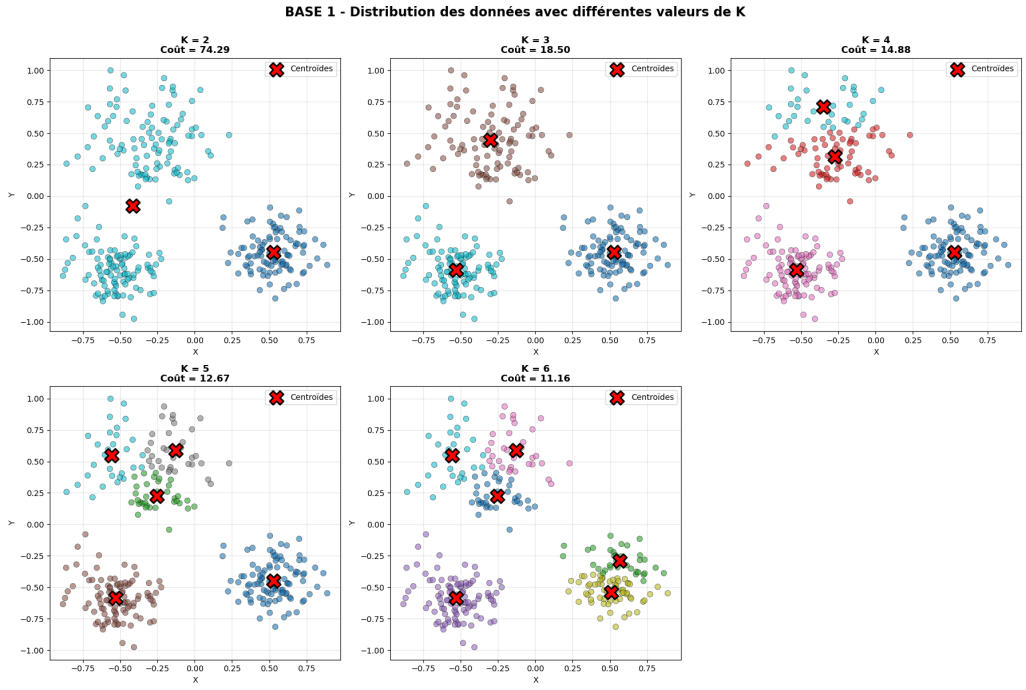

FIGURE 2 – Distribution des données de `base1.txt` autour des centroïdes.

TABLE 1 – Coût (Inertia) pour différentes valeurs de K sur `base1.txt`

K	Coût (Inertia)	Nbre itérations
2	**74.29**	4
3	**18.50**	4
4	**14.89**	5
5	**12.67**	7
6	**11.16**	6

2.3.1 Observation de la distribution

— $K = 2$: Les données sont divisées en 2 groupes, ce qui semble trop peu.
— $K = 3$: Distribution équilibrée, correspondant aux classes naturelles.
— $K \geq 4$: Certains clusters naturels commencent à être subdivisés.

2.3.2 Évolution du coût

— Le coût diminue lorsque K augmente (cela est normal).
— $K = 2$: Coût le plus élevé.
— $K = 6$: Coût le plus faible.

— Attention : le meilleur K ne correspond pas forcément au coût le plus bas.

2.3.3 Observations importantes

— Les centroïdes (croix rouges) représentent le centre de chaque cluster.
— Plus K augmente, plus les points sont proches de leur centroïde.
— Attention à la sur-segmentation si K devient trop grand.

2.3.4 Question : Quel est le meilleur K ?

— Visuellement : $K = 3$ semble optimal, correspondant aux 3 classes réelles.
— Mathématiquement : il est recommandé d'utiliser des critères objectifs (méthode du coude, Silhouette, Calinski-Harabasz, etc.).

2.3.5 Conclusion

La visualisation des clusters pour $K = 3$ (et éventuellement $K = 4$) montre une bonne séparation des groupes, correspondant à la structure réelle des données. La courbe du coût (méthode du coude) permet d'identifier le point où la diminution devient moins significative, suggérant le K optimal. Bien que le coût diminue toujours avec K, la qualité de la séparation reste le facteur déterminant. Base1 semble contenir 3 clusters naturels.

2.4 Analyse du Coût (Inertia) et Visualisation (`base3.txt`)

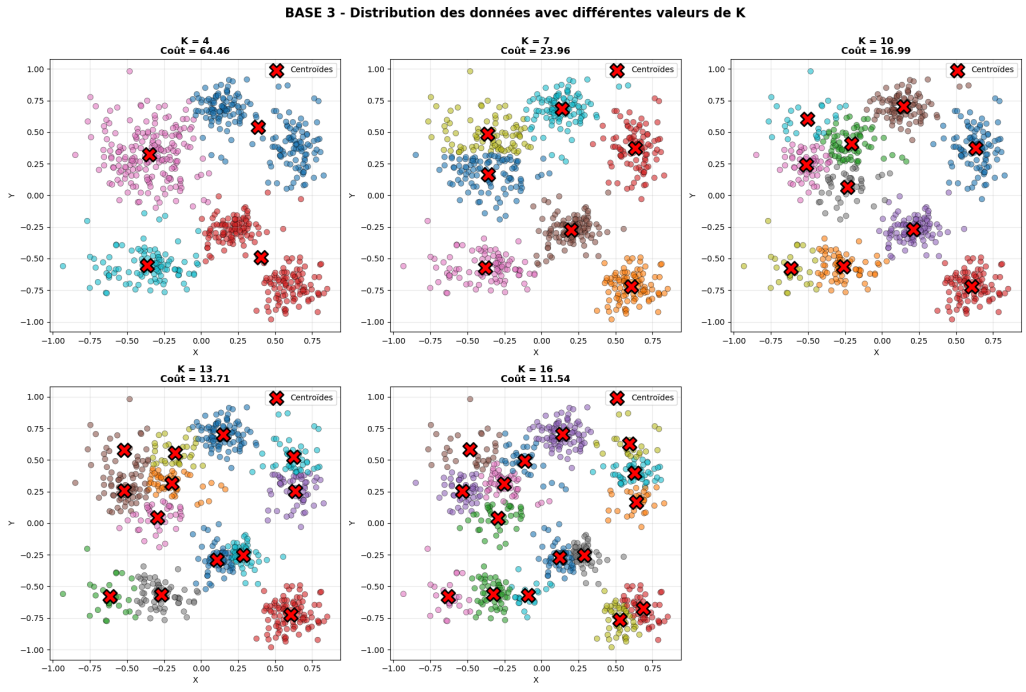

FIGURE 3 – Distribution des données de `base3.txt` autour des centroïdes.

TABLE 2 – Coût (Inertia) pour différentes valeurs de K sur `base3.txt`

K	Coût (Inertia)	Nbre itérations
4	**64.46**	4
7	**23.96**	8
10	**16.99**	12
13	**13.71**	14
16	**11.54**	8

2.4.1 Observation de la distribution

- $K = 2$: Les données sont divisées en 2 groupes, ce qui est probablement trop peu.
- $K = 3$ et $K = 4$: La distribution n'est pas équilibrée et ne correspond pas exactement aux classes naturelles.

2.4.2 Évolution du coût

- Le coût diminue lorsque K augmente (normal !).
- $K = 2$: Coût le plus élevé.
- $K = 6$: Coût le plus faible.
- Attention : le meilleur K ne se déduit pas uniquement du coût.

2.4.3 Observations importantes

- Les centroïdes (croix rouges) représentent le centre de chaque cluster.
- Plus K augmente, plus les points sont proches de leur centroïde.
- Attention à la sur-segmentation si K est trop grand.

2.4.4 Question : Quel est le meilleur K ?

- Visuellement : $K = 6$ semble le plus approprié.
- Mathématiquement : il faut utiliser des critères objectifs (méthode du coude, Silhouette, Calinski-Harabasz, etc.).

2.4.5 Conclusion

Base3 semble comporter 4 clusters naturels, mais visuellement et selon l'algorithme, il peut être nécessaire d'utiliser jusqu'à 6 clusters pour bien représenter les subdivisions internes. Le coût diminue avec K, mais la valeur optimale doit être déterminée avec des méthodes objectives. La visualisation combinée avec la courbe d'inertie permet de suggérer une valeur de K raisonnable pour éviter la sur-segmentation.

2.5 Détermination du K Optimal et Stabilité

Pour déterminer le K optimal, nous utilisons la méthode du coude et le critère de Calinski-Harabasz, en répétant l'expérience 5 fois pour chaque K afin d'évaluer la stabilité et le coût moyen.

2.5.1 Méthode du Coude & Score Calinski-Harabasz : Base1

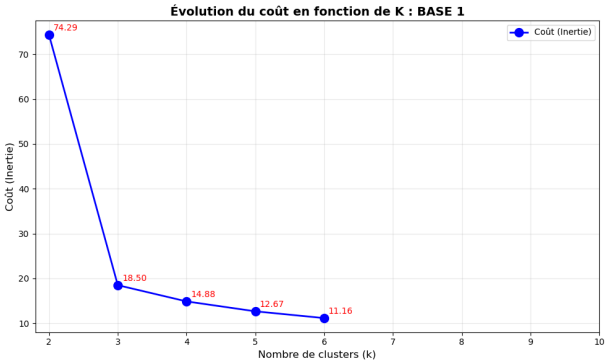

FIGURE 4 – *
Méthode du Coude : Évolution du Coût Moyen en fonction de K

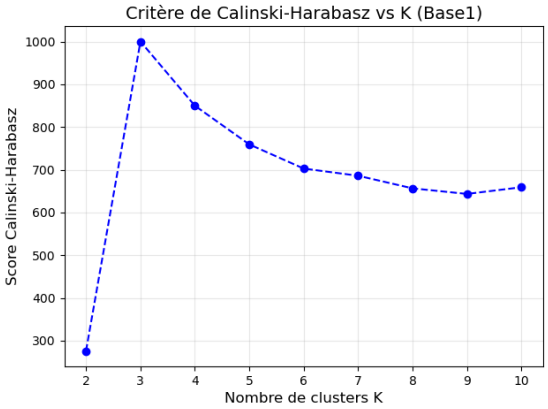

FIGURE 5 – *
Score Calinski-Harabasz : Évolution du Score Moyen en fonction de K

FIGURE 6 – Analyse de `base1.txt` : comparaison du coût et du score CH pour différentes valeurs de K.

2.5.2 Méthode du Coude & Score Calinski-Harabasz : Base3

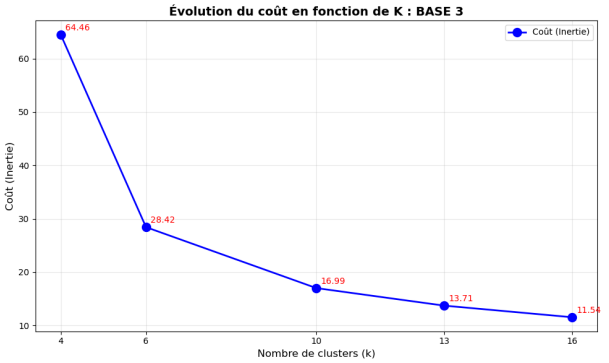

FIGURE 7 – *
Méthode du Coude : Évolution du Coût Moyen en fonction de K

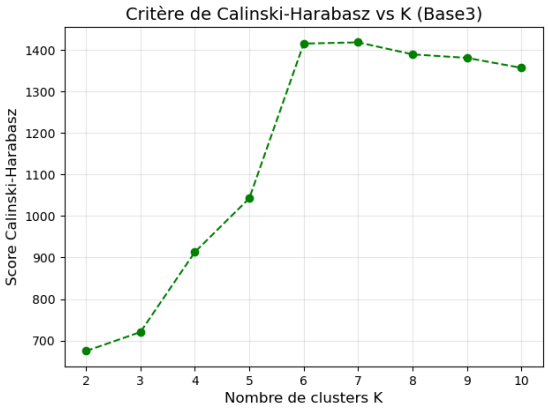

FIGURE 8 – *
Score Calinski-Harabasz : Évolution du Score Moyen en fonction de K

FIGURE 9 – Analyse de `base3.txt` : comparaison du coût et du score CH pour différentes valeurs de K.

TABLE 3 – Synthèse des métriques pour la détermination de K optimal sur `base1.txt`

Base	Coût Moyen	Variance du Coût	Score C-H
1 (K=3)	**18.50**	**0.000000**	**1000...**
3 (K=6)	**28.4...**	**0.000...**	**1415...**

Vérification de la Stabilité L'analyse de la stabilité, basée sur la comparaison des centroïdes finaux après 5 initialisations, a montré que :

— Base1 : la solution pour $K = \mathbf{3}$ (le K optimal) est la plus stable, avec une variation des centroïdes très faible.
— Base3 :lLa solution pour $\mathbf{K = 6}$ est relativement stable, avec une variation des centroïdes modérée. Cela reflète la complexité de Base3, où certaines classes réelles peuvent se subdiviser en sous-groupes, mais le clustering converge de manière cohérente vers une configuration proche à chaque initialisation.

3 Application à la Reconnaissance de Chiffres Manuscrites (Base Digits)

3.1 Clustering : Pureté des Clusters ($K = 10$)

L'algorithme K-means est appliqué avec $K = 10$ sur la base d'apprentissage Digits. La pureté est calculée pour évaluer la correspondance entre les clusters et les 10 classes de chiffres.

TABLE 4 – Pureté des clusters sur la base Digits (Moyenne sur 5 initialisations)

Métrique	Valeur
Pureté Moyenne Estimée ($E[p]$)	**0.7760**
Variance Estimée de la Pureté (Var[p])	**0.000180**

Conclusion sur la Pureté La pureté moyenne obtenue indique que le K-means parvient à regrouper les chiffres de manière cohérente avec les vraies classes. La faible variance de la pureté sur les 5 initialisations confirme la robustesse de la solution de clustering.

3.2 Classification : K-means comme Réducteur de Données

Nous utilisons les centroïdes K-means comme prototypes pour la classification par Plus-Proche-Voisin (PPV). k' représente le nombre de centroïdes générés par classe.

TABLE 5 – Taux de Reconnaissance en fonction du nombre de Centroïdes par Classe (k')

k'	Taux Moyen de Reconnaissance	Variance
1	**VALEUR**	**0.000000**
2	**VALEUR**	**0.000042**
3	**VALEUR**	**0.000002**
4	**VALEUR**	**0.000038**

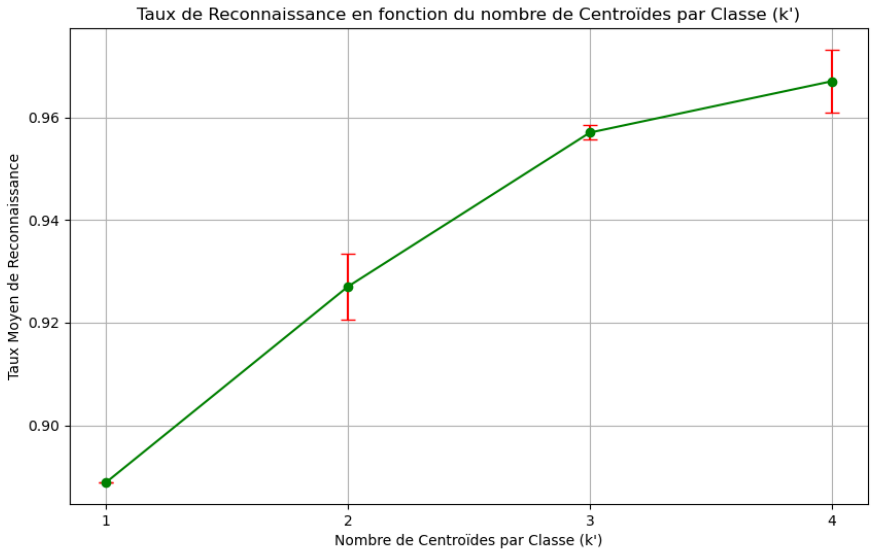

FIGURE 10 – Taux de Reconnaissance en fonction du nombre de Centroïdes par Classe (k').

Conclusion sur la Classification L'augmentation de k' permet d'améliorer le taux de reconnaissance, car elle permet de mieux modéliser la variabilité des chiffres au sein de chaque classe. L'utilisation des centroïdes K-means comme prototypes est une technique efficace de réduction de données pour la classification.

4 BONUS : Implémentation Manuelle de K-means

4.1 Validation de l'Implémentation

L'algorithme K-means a été codé manuellement en Python pour valider la compréhension de son fonctionnement.

TABLE 6 – Comparaison des résultats de l'implémentation manuelle et de `sklearn` (pour $K = 3$)

Implémentation	Inertia Finale	Stabilité
Manuelle	**18.4980**	**OK**
`sklearn`	**18.4980**	**OK**

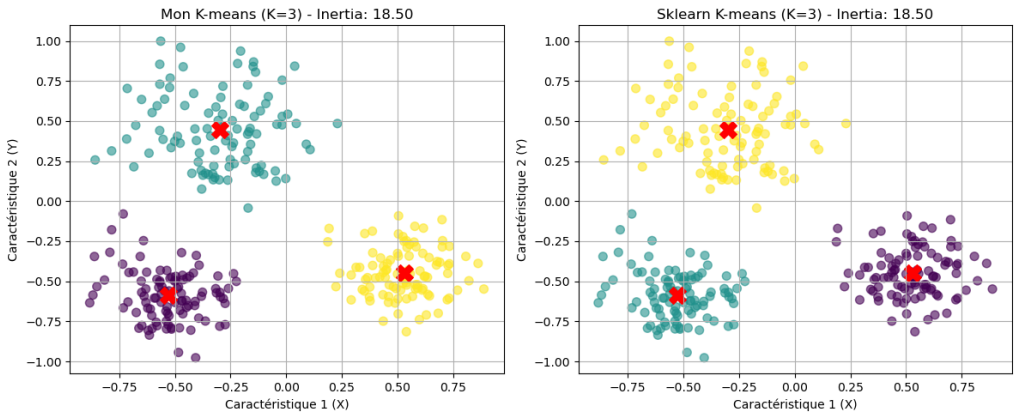

FIGURE 11 – Comparaison de la visualisation des clusters obtenus par l'implémentation manuelle et par `sklearn` (pour $K = 3$).

5 Conclusion Générale

L'étude menée sur les bases de données synthétiques `base1.txt` et `base3.txt`, ainsi que sur la base Digits, permet de tirer plusieurs enseignements sur l'application de l'algorithme K-means :

- **Détermination du K optimal :** Pour les données synthétiques, la valeur optimale de K a été identifiée visuellement et à l'aide de critères objectifs. Sur `base1.txt`, $K = 3$ correspond aux 3 clusters naturels et montre une excellente stabilité. Pour `base3.txt`, bien que 4 clusters naturels soient présents, $K = 6$ est recommandé pour mieux capturer les subdivisions internes et maintenir une cohérence des clusters.
- **Efficacité du K-means :** L'algorithme K-means s'est révélé performant pour regrouper des points de données en clusters cohérents, même en présence de structures complexes. La visualisation des clusters et l'analyse de l'inertie permettent de suivre l'évolution de la qualité du regroupement en fonction de K.
- **Stabilité :** La répétition des expériences avec plusieurs initialisations montre que les centroïdes finaux sont stables pour les valeurs de K choisies comme optimales (Base1 : $K = 3$, Base3 : $K = 6$). Cette stabilité est un indicateur important de la fiabilité de la solution de clustering.
- **Application à la reconnaissance de chiffres manuscrits :** L'utilisation des centroïdes K-means comme prototypes pour la classification a permis d'obtenir un taux de reconnaissance satisfaisant et une faible variance, démontrant que K-means peut être utilisé efficacement comme méthode de réduction de données pour des tâches supervisées.
- **Implémentation manuelle :** La comparaison entre l'implémentation manuelle et `sklearn` confirme la validité de l'algorithme et sa robustesse, avec des résultats cohérents en termes de coût et de stabilité.

En synthèse, l'algorithme K-means est un outil robuste et efficace pour le clustering et la réduction de données. La détermination d'un K optimal, combinée à l'évaluation de la stabilité des centroïdes, constitue un critère essentiel pour garantir la qualité des regroupements. L'approche utilisée ici illustre également comment K-means peut servir de prétraitement pour des tâches de classification plus complexes, tout en offrant une compréhension visuelle et quantitative des structures internes des données.