



République Algérienne Démocratique et Populaire



Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université des Sciences et de la Technologie Houari Boumediene

Faculté d'Informatique
Département Informatique

Spécialité : Bio-Informatique

Rapport Module Fouille de Données

Thème

Exploration, Prétraitement et Clustering de Données

Réalisé par :

LAIB Ayoub

Table des matières

1.	Introduction	1
2.	TP N°1 : Exploration des Données et Prétraitement	2
2.1.	Installation de l'environnement Anaconda	2
2.1.1.	Anaconda Navigator :	3
2.2.	Présentation de l'Application de clustering :	4
2.3.	Manipulation et exploration du fichier d'apprentissage :	5
2.3.1.	Ouverture de benchmarks :	5
2.3.2.	Ouverture de Fichier :	5
2.3.3.	Affichage des Données Distinctes :	7
2.3.4.	Le nombre d'instances :	7
2.3.5.	Caractéristiques des attributs :	8
2.3.6.	Résumé statistique des attributs : Min, Max, Médiane, Q1 et Q3 :	8
2.3.6.1.	Interprétation :	9
2.3.7.	Boxplots des attributs :	10
2.3.7.1.	Interprétation et visualisation :	11
2.3.8.	Scatter plot :	12
2.3.8.1.	Visualisation :	12
2.3.8.2.	Interprétation :	13
2.3.8.3.	Interprétation 2 :	14
2.3.9.	Mode, moyenne et médiane :	14
2.3.9.1.	Interprétation :	15
2.3.10.	Détection et remplacement des valeurs manquantes :	16
2.3.11.	La normalisation MIN/MAX :	17
2.3.12.	La normalisation Z-score :	18
3.	TP N°2 : Clustering avec K-Means et K-Medoids	19
3.1.	Identification du nombre optimal de clusters avec la courbe d'Elbow :	19
3.1.1.	Interprétation :	20
3.2.	Application de l'algorithme K-Means :	21
3.3.	Application de l'algorithme K-Medoids :	22
3.4.	Évaluation des performances des clusters :	23
4.	TP N°3 : Clustering avec Agnes, Diana et DBSCAN :	24
4.1.	Application l'algorithme Agnes :	24
4.2.	Application l'algorithme Diana:	25
4.3.	Application l'algorithme DBSCAN:	27
4.3.1.	Évaluation des performances des clusters :	29
4.3.1.1.	Interprétation :	29
5.	Comparaison des méthodes à travers un histogramme des inerties :	30
6.	Conclusion :	32
6.1.	Perspectives pour de futurs travaux en fouille de données :	33

Table des Figures

Figure 1: Interface d'Anaconda Navigator	3
Figure 2: Présentation de l'Application de Clustering	4
Figure 3: Fonctionnalité d'Ouverture de Fichier	5
Figure 4: Ensemble de données dans un dataframe.....	6
Figure 5: Affichage des Valeurs Distinctes pour Chaque Attribut.....	7
Figure 6: Nombre d'Instances dans le Jeu de Données.....	8
Figure 7: Caractéristiques des Attributs	8
Figure 8: Résumé Statistique des Attributs: Min, Max, Médiane, Q1 et Q3	9
Figure 9: Comparaison des Distributions des Attributs à l'Aide de Boxplots	10
Figure 10: Scatter Plot du Benchmark : Âge vs Sexe.....	12
Figure 11: Scatter Plot du Benchmark : Âge vs chol.....	13
Figure 12: Mode, Moyenne et Médiane pour Chaque Attribut	15
Figure 13: Message d'information : Remplacement des valeurs manquantes : Terminé.....	16
Figure 14: Fichier de données contenant des valeurs manquantes	16
Figure 15: Fichier après repérition des valeurs manquantes	17
Figure 16: Normalisation MIN/MAX des données	17
Figure 17: Normalisation Z-score	18
Figure 18: Courbe d'Elbow pour l'identification du nombre optimal de clusters	20
Figure 19: Résultats du clustering K-means.....	21
Figure 20: Résultats du clustering K-mediods	22
Figure 21: Performances de KMeans	23
Figure 22: Performances de KMediods	23
Figure 23: Dendrogramme résultant de l'algorithme AGNES.....	25
Figure 24: Performances de Agnes.....	25
Figure 25: Dendrogramme résultant de l'algorithme DIANA	26
Figure 26 : Performances de DIANA	26
Figure 27: Résultats du clustering DBScan.....	28
Figure 28: Performances DBScan	29
Figure 29: Histogramme de l'inertie intra-classe.....	30
Figure 30: Histogramme de l'inertie intre-classe.....	31

1.Introduction

La fouille de données, ou "data mining", est un processus essentiel dans l'exploration et l'analyse de grandes quantités de données afin de découvrir des connaissances précieuses et souvent cachées. Dans un contexte où la quantité de données disponibles ne cesse d'augmenter, la capacité à extraire des informations significatives devient important pour de nombreuses applications, allant de la recherche scientifique à la prise de décision en entreprise.

Au cours du semestre S2 M1BIOINFO, nous avons exploré divers aspects de la fouille de données à travers trois travaux pratiques. Ces travaux ont été conçus pour nous permettre de comprendre les principes fondamentaux de la manipulation, de l'exploration et du traitement des données, ainsi que l'application de différentes techniques de clustering pour découvrir des structures intrinsèques dans les ensembles de données.

Dans ce rapport, nous présenterons en détail notre expérience et nos résultats obtenus lors de ces trois TPs. Nous commencerons par une exploration approfondie des données, en mettant l'accent sur la préparation des données et la visualisation des distributions des attributs. Ensuite, nous aborderons les techniques de clustering, en appliquant plusieurs algorithmes tels que K-Means, K-Medoids, Agnes, Diana et DBSCAN, pour découvrir des regroupements naturels dans nos ensembles de données.

Chaque TP sera examiné individuellement, en décrivant les objectifs spécifiques, les méthodes utilisées, les résultats obtenus et les analyses effectuées. Enfin, nous conclurons en discutant des avantages et des limites des différentes approches de clustering, ainsi que des perspectives pour de futures recherches dans le domaine de la fouille de données.

2. TP N°1 : Exploration des Données et Prétraitement

Le TP1 vise à initier les étudiants aux fondamentaux de la fouille de données en mettant l'accent sur la manipulation et l'exploration des données. Dans ce contexte, les participants utilisent l'environnement Anaconda et le package SkLearn pour ouvrir, lire et visualiser des fichiers de données. L'objectif est d'acquérir des compétences telles que la gestion des valeurs manquantes, la normalisation des données et la visualisation des distributions d'attributs. Ce TP constitue ainsi une première étape essentielle avant d'aborder des techniques plus avancées de fouille de données, telles que le clustering et la classification.

2.1. Installation de l'environnement Anaconda

Anaconda est un outil de distribution Python open source utilisé pour la gestion de packages et d'environnement de développement. Il sert à la réalisation de projets Python et de langages de programmation connexes, tel que R et Julia. Anaconda simplifie le processus de configuration et de gestion de différents packages et bibliothèques Python, ce qui en fait un outil indispensable pour les développeurs et les data scientist.

Son importance en data science est marquée par les packages qu'il propose comme : Numpy, Panda, Jupyter et Python. On peut aussi directement gérer ses environnements de développement et ses packages depuis son outil "conda". Il est également multi-plateforme, ce qui permet de l'installer sur Linux, Windows ou MacOS.



2.1.1. Anaconda Navigator :

Anaconda Navigator est l'interface de navigation d'Anaconda, elle permet de lancer les différentes API disponibles et de gérer les différents packages et environnements du logiciel. Cette interface permet de naviguer simplement dans le logiciel, sans devoir connaître toutes les lignes de code de Conda. Dans Anaconda Navigator, vous pouvez retrouver ces différentes API:

- JupyterLab
- JupyterNotebook
- Spyder
- Pycharm
- VSCode
- Orane 3 APP
- RStudio
- Anaconda powerShell

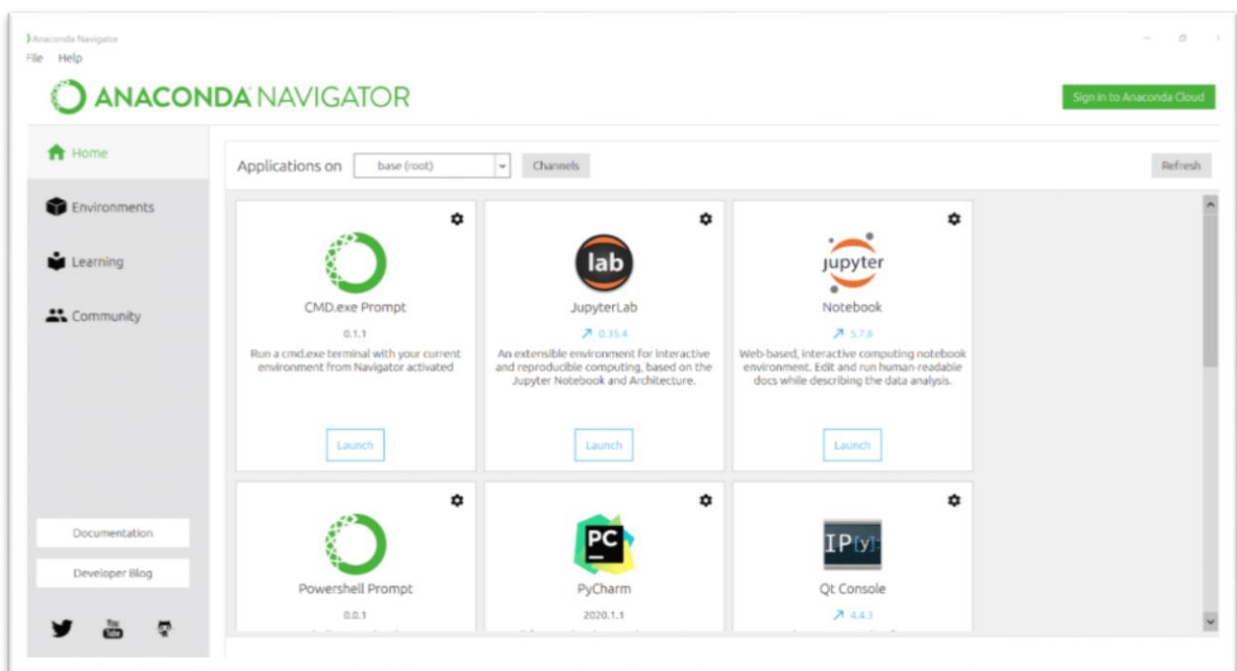


Figure 1: Interface d'Anaconda Navigator

2.2. Présentation de l'Application de clustering :

Notre application de fouille de données offre un ensemble d'outils puissants pour l'exploration et l'analyse de données, visant à extraire des informations utiles à partir de grands ensembles de données. Conçue pour offrir une interface conviviale et intuitive, cette application permet aux utilisateurs d'appliquer diverses techniques de fouille de données, y compris le prétraitement des données, le clustering, et l'évaluation des performances des algorithmes de clustering. À travers cette application, nous visons à fournir aux utilisateurs, qu'ils soient novices ou experts en fouille de données, un environnement flexible et interactif pour explorer et découvrir des structures cachées dans leurs données.

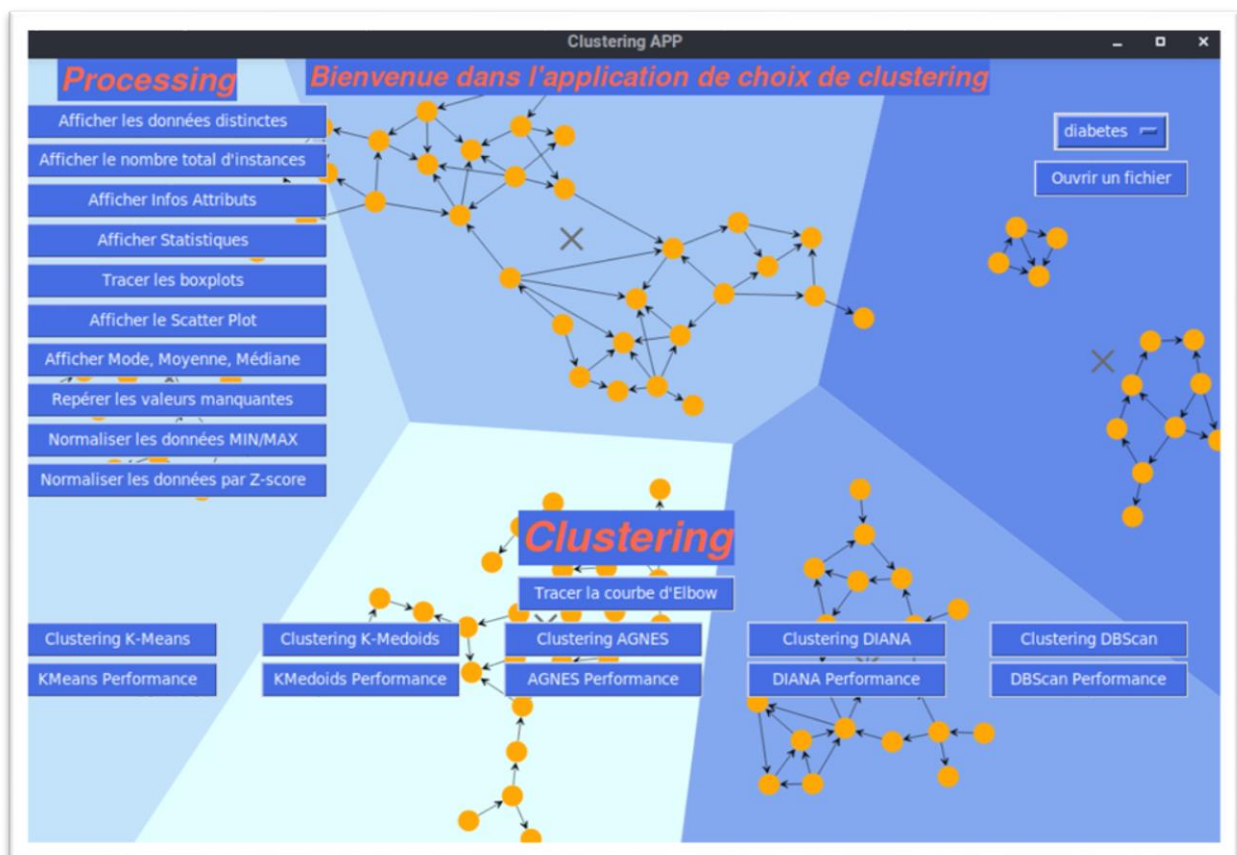


Figure 2: Présentation de l'Application de Clustering

2.3. Manipulation et exploration du fichier d'apprentissage :

2.3.1. Ouverture de benchmarks :

Dans ce TP, nous avons utilisé le fichier heart.csv, contenant des données médicales sur des patients. Chaque ligne représente un patient et chaque colonne correspond à une caractéristique médicale telle que l'âge, le sexe, la pression artérielle, le taux de cholestérol, etc. L'objectif de cette étape était de comprendre la structure des données et d'identifier les caractéristiques pertinentes pour la fouille de données. En explorant ce fichier, nous avons pu vérifier la qualité des données, repérer d'éventuelles valeurs manquantes et avoir une première impression de la distribution des variables. Cette étape d'ouverture des benchmarks est importante car elle nous permet de préparer les données pour les étapes suivantes de prétraitement et d'analyse, telles que la normalisation et la visualisation des données.

2.3.2. Ouverture de Fichier :

Notre application offre une fonctionnalité d'ouverture de fichier qui permet aux utilisateurs de charger des ensembles de données au format CSV (Comma-Separated Values) ou ARFF (Attribute-Relation File Format).

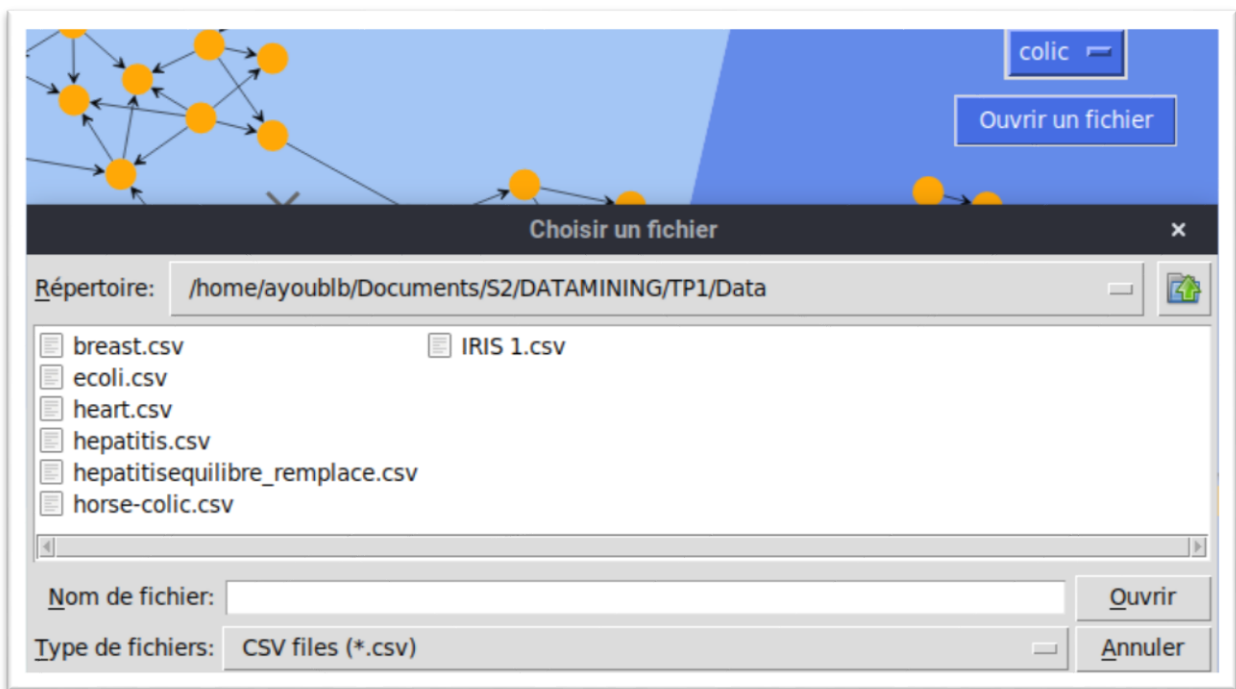


Figure 3: Fonctionnalité d'Ouverture de Fichier

Voici l'ensemble de nos données :

Données													-	□	×
age	sex	cp	trtbps	chol	fbs	restecg	thalachh	exng	oldpeak	slp	caa	thall	output		
60	1	3	145	233	1	0	150	0	2.3	0	0	1	1		
35	1	2	130	250	0	1	187	0	3.5	0	0	2	1		
41	0	1	130	204	0	0	172	0	1.4	2	0	2	1		
55	1	1	120	236	0	1	178	0	0.8	2	0	2	1		
56	0	0	120	354	0	1	163	1	0.6	2	0	2	1		
55	1	0	140	192	0	1	148	0	0.4	1	0	1	1		
56	0	1	140	294	0	0	153	0	1.3	1	0	2	1		
44	1	1	120	263	0	1	173	0	0.0	2	0	3	1		
52	1	2	172	199	1	1	162	0	0.5	2	0	3	1		
57	1	2	150	168	0	1	174	0	1.6	2	0	2	1		
54	1	0	140	239	0	1	160	0	1.2	2	0	2	1		
48	0	2	130	275	0	1	139	0	0.2	2	0	2	1		
49	1	1	130	266	0	1	171	0	0.6	2	0	2	1		
64	1	3	110	211	0	0	144	1	1.8	1	0	2	1		
55	0	3	150	283	1	0	162	0	1.0	2	0	2	1		
50	0	2	120	219	0	1	158	0	1.6	1	0	2	1		
58	0	2	120	340	0	1	172	0	0.0	2	0	2	1		
66	0	3	150	226	0	1	114	0	2.6	0	0	2	1		
40	1	0	150	247	0	1	171	0	1.5	2	0	2	1		
69	0	3	140	239	0	1	151	0	1.8	2	2	2	1		
59	1	0	135	234	0	1	161	0	0.5	1	0	3	1		
44	1	2	130	233	0	1	179	1	0.4	2	0	2	1		
42	1	0	140	226	0	1	178	0	0.0	2	0	2	1		

Figure 4: Ensemble de données dans un dataframe

2.3.3. Affichage des Données Distinctes :

Cette fonctionnalité permet aux utilisateurs d'afficher les valeurs uniques de chaque attribut dans leur ensemble de données chargé. Plutôt que d'afficher les données de manière détaillée, cette fonctionnalité présente une liste des valeurs distinctes pour chaque attribut

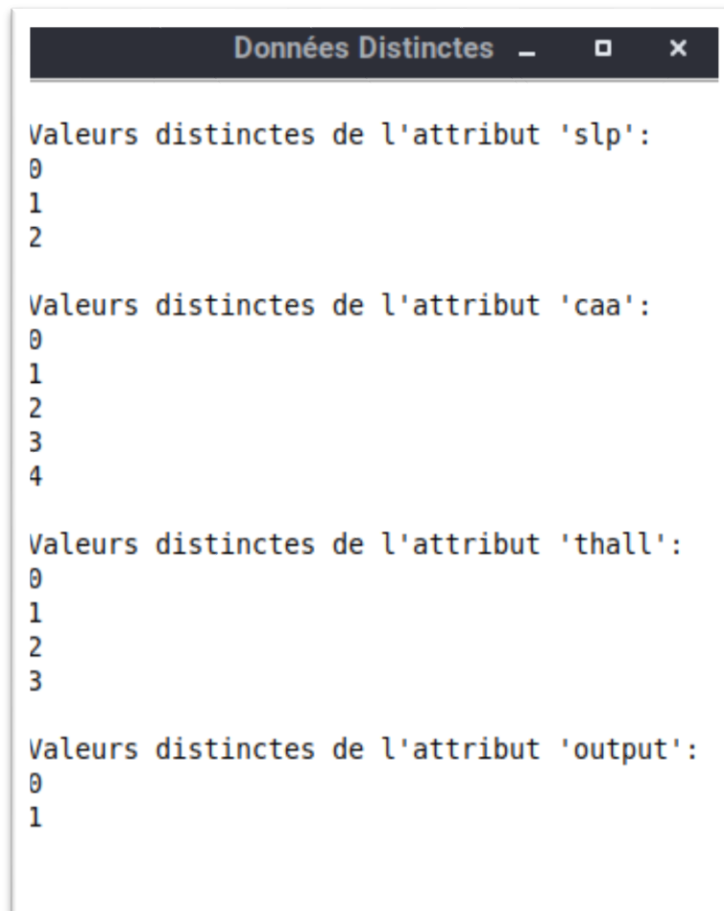


Figure 5: Affichage des Valeurs Distinctes pour Chaque Attribut

2.3.4. Le nombre d'instances :

Le nombre d'instances représente le nombre total de lignes dans le jeu de données, ce qui équivaut au nombre de patients inclus dans l'étude médicale. Il est essentiel d'avoir une compréhension claire de ce nombre, car il définit la taille de l'échantillon sur lequel nous effectuons notre analyse.

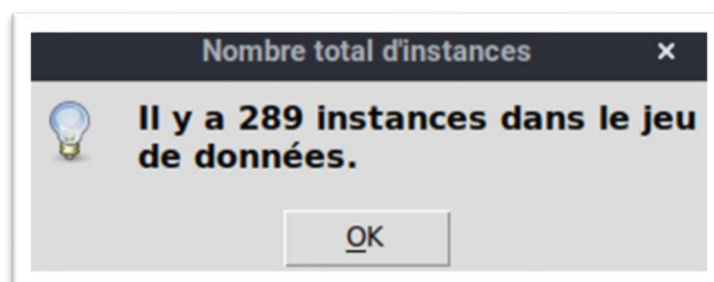


Figure 6: Nombre d'Instances dans le Jeu de Données

2.3.5. Caractéristiques des attributs :

Les attributs sont les différentes caractéristiques ou variables mesurées pour chaque individu. Par exemple, parmi les attributs de ce fichier, nous avons l'âge, le sexe, la pression artérielle, le taux de cholestérol, etc. Il est essentiel de connaître le nombre d'attributs et leur type (numérique ou catégorique), car cela définit les dimensions de notre ensemble de données et les variables sur lesquelles nous baserons notre analyse. Cette identification des attributs est importante pour choisir les bonnes techniques d'analyse et de prétraitement de données adaptées à notre étude de fouille de données.

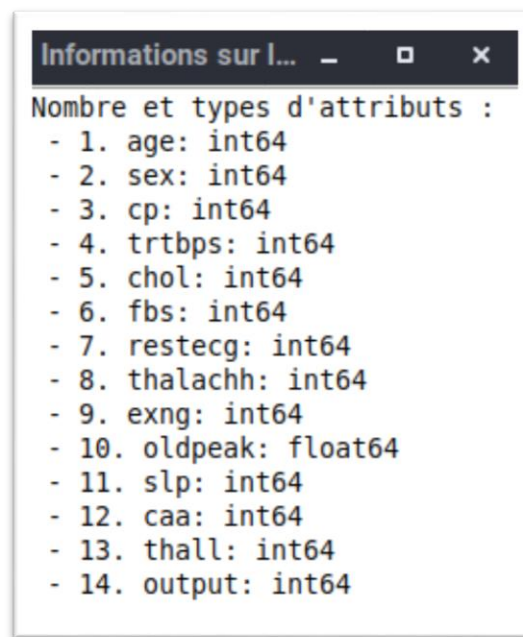


Figure 7: Caractéristiques des Attributs

2.3.6. Résumé statistique des attributs : Min, Max, Médiane, Q1 et Q3 :

Ces mesures fournissent des informations ont un rôle sur la distribution des données et permettent de mieux comprendre la variabilité des valeurs pour chaque attribut.

- Le minimum (Min) représente la valeur la plus basse observée pour un attribut donné.
- Le maximum (Max) représente la valeur la plus élevée observée pour cet attribut.
- La médiane (Median) est la valeur qui divise l'échantillon en deux parties égales, ce qui signifie que la moitié des valeurs sont inférieures à la médiane et l'autre moitié sont supérieures.
- Le premier quartile (Q1) est la valeur qui sépare le premier quart des données, indiquant que 25% des données sont inférieures à cette valeur.
- Le troisième quartile (Q3) est la valeur qui sépare le dernier quart des données, ce qui signifie que 75% des données sont inférieures à cette valeur.

Ces cinq nombres fournissent une vision détaillée de la distribution des données pour chaque attribut, ce qui est essentiel pour évaluer la variabilité et la dispersion des valeurs dans notre ensemble de données.

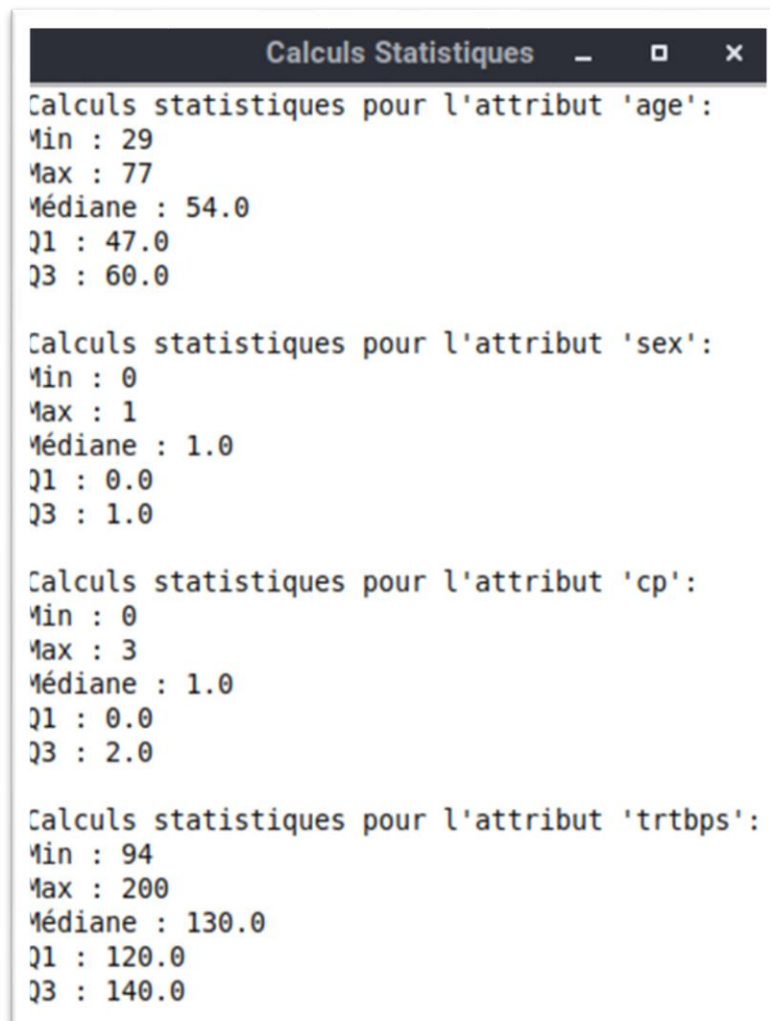


Figure 8: Résumé Statistique des Attributs: Min, Max, Médiane, Q1 et Q3

2.3.6.1. Interprétation :

Pour l'attribut 'age', la valeur minimale observée est de 29 ans et la valeur maximale est de 77 ans. La médiane de 54 ans indique que la moitié des individus ont moins de 54 ans et l'autre moitié ont plus de 54 ans. Le premier quartile (Q1) de 47 ans signifie que 25% des individus ont moins de 47 ans, tandis que le troisième quartile (Q3) de 60 ans indique que 75% des individus ont moins de 60 ans.

Pour l'attribut 'trtbps' (tension artérielle au repos), les valeurs minimale et maximale sont respectivement de 94 et 200. La médiane de 130.0 indique que la tension artérielle moyenne est de 130 mmHg. Les quartiles Q1 et Q3 de 120.0 et 140.0 indiquent la distribution de la tension artérielle avec 25% des individus ayant une tension artérielle inférieure à 120 mmHg et 75% des individus ayant une tension artérielle inférieure à 140 mmHg.

2.3.7. Boxplots des attributs :

Les boxplots sont des diagrammes qui représentent graphiquement la distribution des données en utilisant cinq statistiques principales : le minimum, le premier quartile (Q1), la médiane, le troisième quartile (Q3) et le maximum. Ils permettent de détecter les valeurs aberrantes, de comparer la dispersion des données entre différents attributs et d'identifier les tendances centrales.

En affichant les boxplots de chaque attribut sur un même graphe, nous pouvons comparer visuellement les distributions des différentes variables et détecter rapidement les éventuelles différences ou similarités dans leurs distributions. Cela nous aide à mieux comprendre la variabilité des données et à identifier les caractéristiques importantes pour notre analyse ultérieure. Cette visualisation globale des boxplots facilite également l'interprétation des résultats et la prise de décision dans le processus de fouille de données.

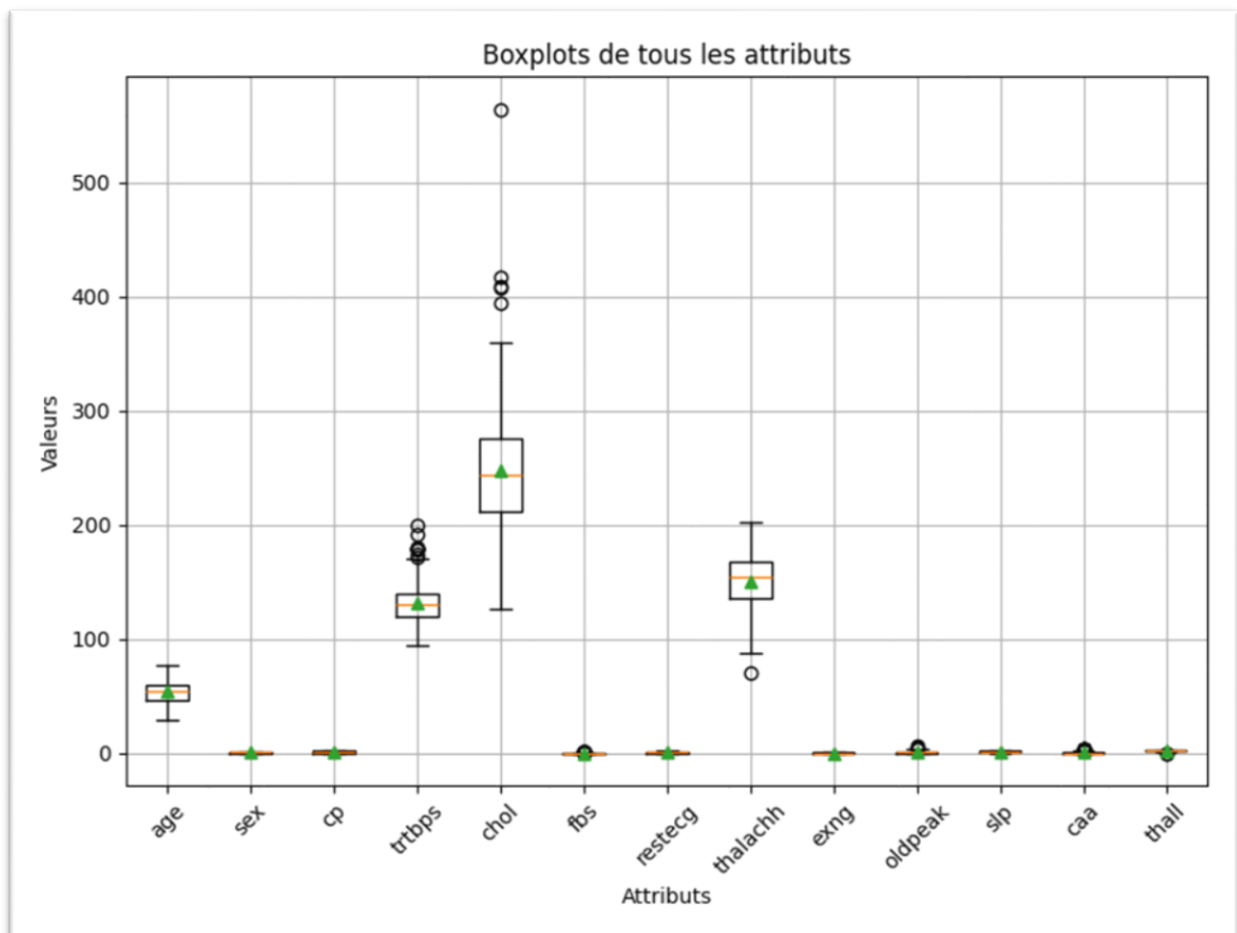


Figure 9: Comparaison des Distributions des Attributs à l'Aide de Boxplots

2.3.7.1. Interprétation et visualisation :

Pour l'attribut 'age', la distribution semble être centrée autour de la médiane de 54 ans, avec une dispersion allant de 29 à 77 ans. La majorité des individus semblent se concentrer entre 47 et 60 ans, avec quelques valeurs aberrantes plus jeunes ou plus âgées.

L'attribut 'sex' semble être une variable binaire avec deux catégories (0 et 1). La médiane de 1.0 indique que la majorité des individus sont probablement de sexe masculin, tandis que la majorité de ceux de sexe féminin sont représentés par les valeurs 0.

Pour l'attribut 'cp' (type de douleur thoracique), la distribution semble être concentrée autour des valeurs 1 et 2, avec quelques valeurs pour les types 0 et 3.

L'attribut 'trtbps' (tension artérielle au repos) montre une médiane de 130.0 mmHg, avec une plage allant de 94 à 200 mmHg. La majorité des individus semblent avoir une tension artérielle comprise entre 120 et 140 mmHg, avec quelques valeurs aberrantes à des niveaux plus élevés ou plus bas.

Pour l'attribut 'chol' (cholestérol sérique), la distribution semble être plus étalée, avec une médiane de 243.0 mg/dl et une plage allant de 126 à 564 mg/dl. Cela indique une variabilité plus importante dans les niveaux de cholestérol par rapport à d'autres attributs.

L'attribut 'fbs' (taux de sucre dans le sang à jeun) montre une distribution fortement biaisée vers la valeur 0, avec la majorité des individus ayant un taux de sucre dans le sang inférieur à 1.

Les attributs 'restecg', 'exng', 'slp', 'caa', 'thall' et 'output' semblent être des variables catégorielles avec plusieurs catégories représentées dans les données.

2.3.8. Scatter plot :

Le scatter plot est un diagramme qui représente les valeurs de deux variables sur un graphique en deux dimensions, où chaque point de données correspond à une paire de valeurs pour ces deux variables.

En affichant le scatter plot du benchmark, nous pouvons visualiser la répartition des données dans l'espace bidimensionnel et examiner la corrélation ou la relation entre les variables. Cette visualisation nous permet de détecter des tendances, des schémas ou des clusters potentiels dans les données, ce qui est essentiel pour comprendre la structure sous-jacente du jeu de données et guider nos analyses ultérieures.

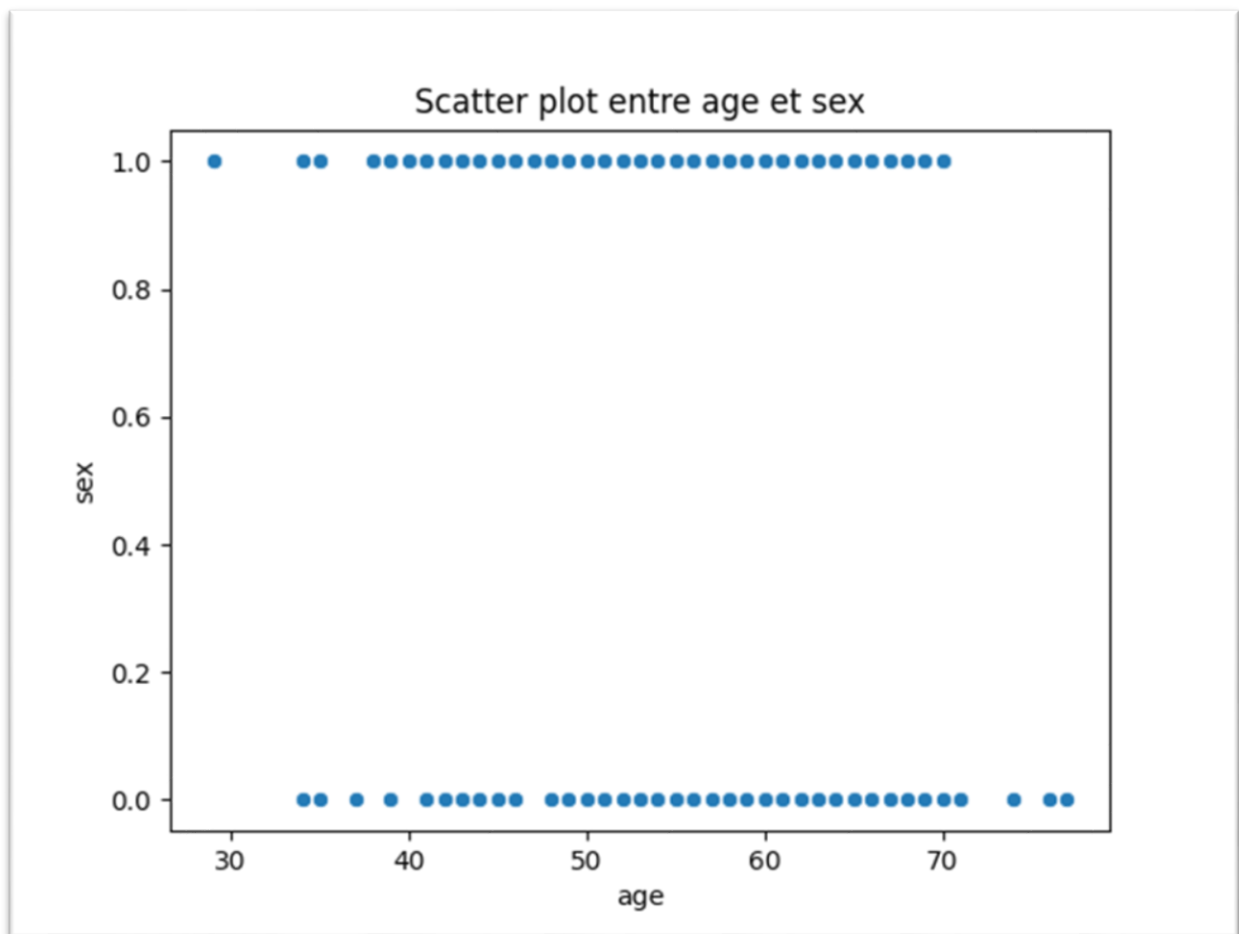


Figure 10: Scatter Plot du Benchmark : Âge vs Sexe

2.3.8.1. Visualisation :

Sur l'axe des x, nous plaçons les valeurs d'âge, tandis que sur l'axe des y, nous plaçons les valeurs de sexe (par exemple, 0 pour les femmes et 1 pour les hommes). Chaque point sur le scatter plot représente un individu, où la position horizontale correspond à son âge et la position verticale correspond à son sexe.

2.3.8.2. Interprétation :

En observant le scatter plot, nous remarquons que les points se répartissent sur deux lignes distinctes, représentant les hommes (sexe 1) et les femmes (sexe 0).

- Pour les hommes (sexe 1), nous observons une dispersion des points sur une plage d'âge allant d'environ 15 ans à 72 ans, avec une concentration plus dense autour des âges moyens.
- Pour les femmes (sexe 0), les points sont également répartis sur une plage d'âge allant d'environ 35 ans à 79 ans, avec une densité plus forte autour des âges moyens.

Cette répartition distincte des points montre une séparation claire entre les deux sexes en termes d'âge.

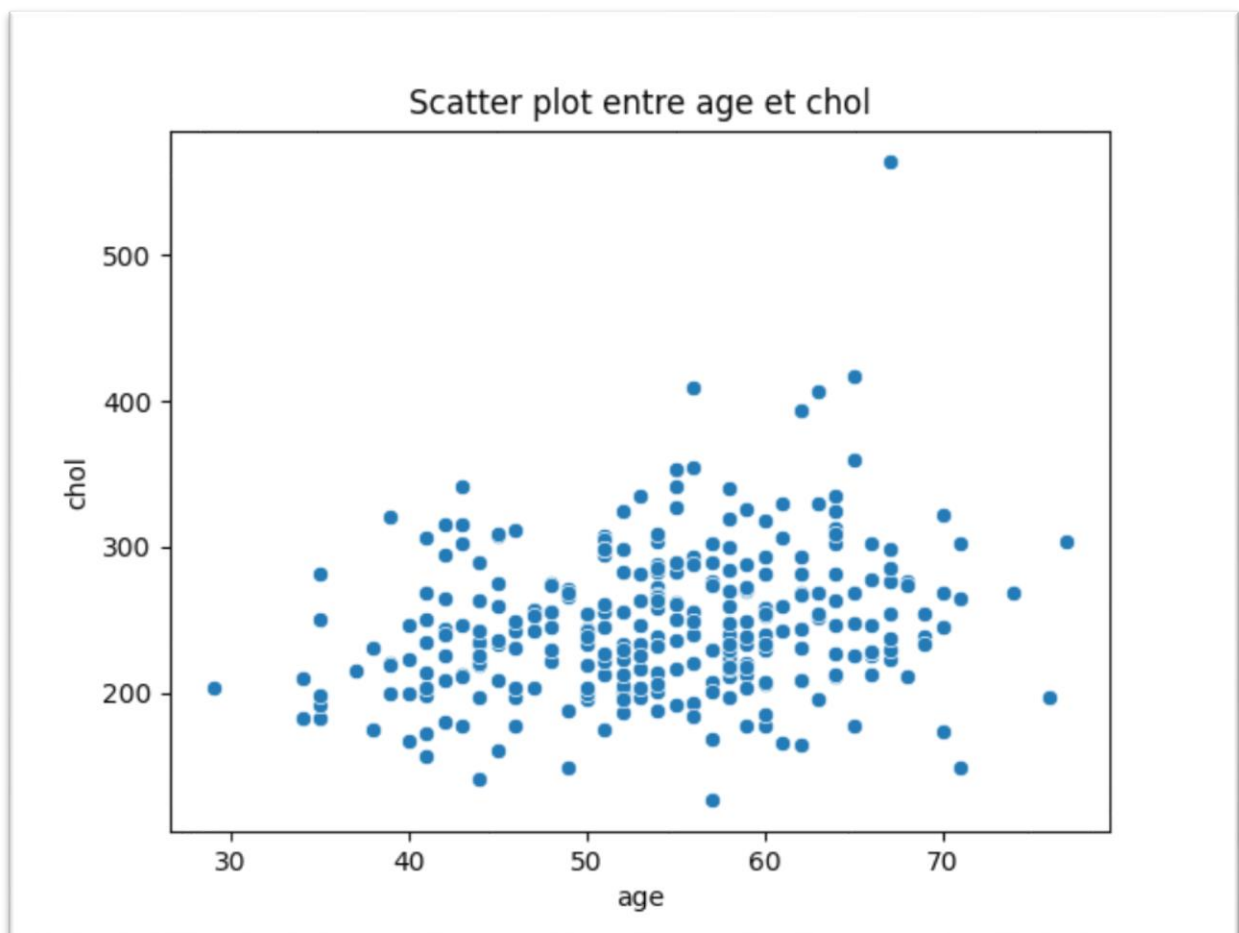


Figure 11: Scatter Plot du Benchmark : Âge vs chol

2.3.8.3. Interprétation 2 :

En observant le scatter plot entre l'âge et le taux de cholestérol (chol), nous pouvons tirer les conclusions suivantes :

- La dispersion des points semble assez uniforme sur l'ensemble de la plage d'âge. Cependant, nous pouvons identifier une tendance générale à une augmentation du taux de cholestérol avec l'âge, bien que cette relation ne soit pas parfaitement linéaire.
- Nous observons des points dispersés sur toute la plage de valeurs du taux de cholestérol, allant d'environ 126 mg/dL à 564 mg/dL. Cela indique une grande variabilité dans les niveaux de cholestérol au sein de notre ensemble de données.
- Il est également intéressant de noter que même parmi les individus plus jeunes, il existe une certaine variabilité dans les niveaux de cholestérol, avec des valeurs allant jusqu'à plus de 400 mg/dL.
- Cependant, nous remarquons également une concentration plus dense de points dans la plage de valeurs inférieures du taux de cholestérol, ce qui indique que la majorité des individus de notre ensemble de données ont des niveaux de cholestérol relativement bas.

2.3.9. Mode, moyenne et médiane :

Ces mesures statistiques nous permettent d'obtenir une compréhension approfondie de la distribution des données pour chaque attribut, en mettant en évidence les valeurs les plus fréquentes, la tendance centrale et la dispersion des données.

- Le mode représente la valeur la plus fréquente dans un ensemble de données. Il indique la valeur qui se produit le plus souvent pour un attribut donné.
- La moyenne (mean) est la somme de toutes les valeurs d'un attribut divisée par le nombre total d'observations. Elle fournit une mesure de la tendance centrale des données.
- La médiane (median) est la valeur qui divise l'échantillon en deux parties égales. Elle est moins sensible aux valeurs extrêmes que la moyenne et offre une meilleure représentation de la tendance centrale lorsque les données sont asymétriques ou contiennent des valeurs aberrantes.

En affichant le mode, la moyenne et la médiane pour chaque attribut, nous obtenons une vue d'ensemble des caractéristiques clés de la distribution des données. Cela nous aide à mieux comprendre les caractéristiques centrales des données et à identifier les schémas ou les tendances importantes pour notre analyse de fouille de données.

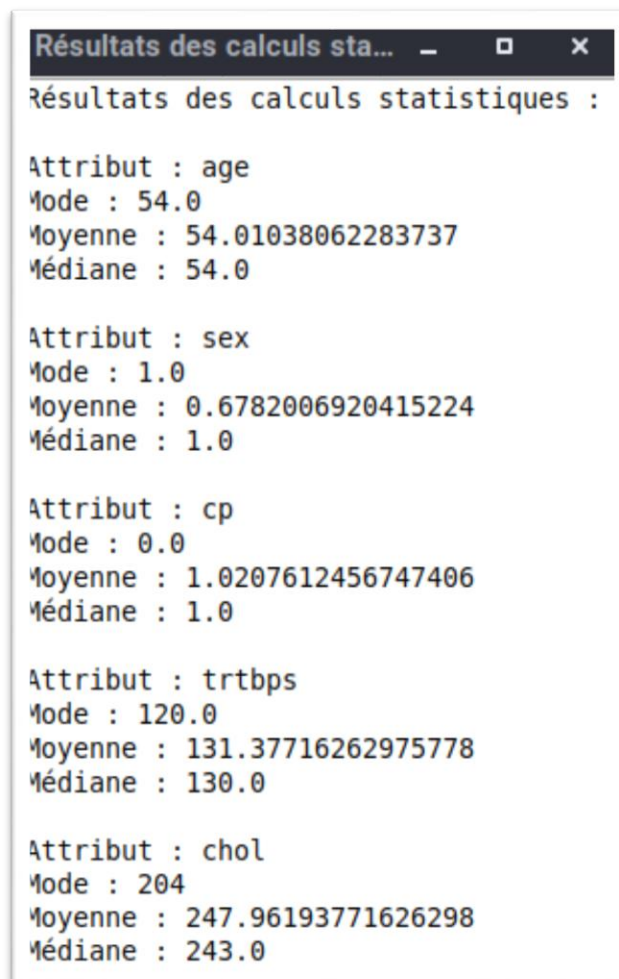


Figure 12: Mode, Moyenne et Médiane pour Chaque Attribut

2.3.9.1. Interprétation :

L'attribut 'trtbps' (pression artérielle au repos) présente un mode de 120.0, ce qui indique une concentration importante de valeurs autour de cette valeur. La moyenne est légèrement plus élevée que le mode, ce qui montre une légère asymétrie vers les valeurs plus élevées. La médiane étant égale à 130.0, cela indique une distribution relativement symétrique des valeurs de pression artérielle.

En ce qui concerne l'attribut 'chol' (taux de cholestérol sérique en mg/dL), le mode est 204, ce qui indique une concentration importante de valeurs autour de cette valeur. La moyenne et la médiane sont également proches de cette valeur, ce qui montre une distribution relativement symétrique des niveaux de cholestérol.

2.3.10. Détection et remplacement des valeurs manquantes :

Les valeurs manquantes peuvent avoir un impact significatif sur nos analyses et nos modèles, il est donc préférant de les repérer et de les remplacer de manière appropriée.

Pour repérer les valeurs manquantes, nous avons parcouru le jeu de données pour identifier les cases vides ou les valeurs codées spécifiques qui indiquent une absence de données. Une fois identifiées, nous avons évalué l'impact de ces valeurs manquantes sur nos analyses et avons choisi une stratégie de remplacement appropriée.

Pour remplacer les valeurs manquantes, nous avons utilisé différentes techniques en fonction de la nature des données et du contexte de l'étude. Par exemple, pour les données numériques, nous pourrions remplacer les valeurs manquantes par la moyenne ou la médiane des valeurs existantes. Pour les données catégoriques, nous pourrions utiliser la valeur la plus fréquente (mode) pour remplacer les valeurs manquantes.

En gérant les valeurs manquantes de manière appropriée, nous assurons l'intégrité et la fiabilité de nos analyses, ce qui nous permet d'obtenir des résultats plus précis et significatifs dans le cadre de notre étude de fouille de données.

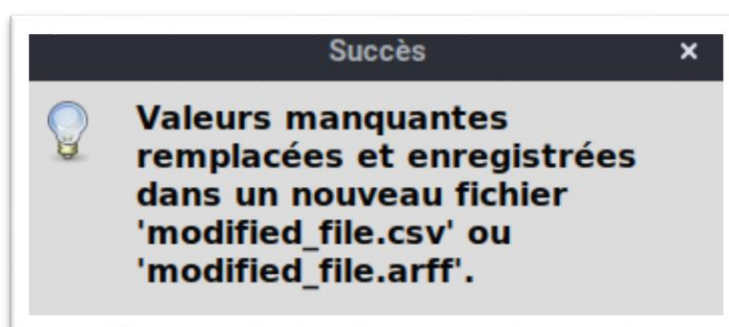


Figure 13: Message d'information : Remplacement des valeurs manquantes : Terminé

	A	B	C	D	E	F	G	H	I	J
2	2	1	530101	38.5	66	28	3	3?		2
3	1	1	534817	39.2	88	20?	?		4	1
4	2	1	530334	38.3	40	24	1	1	3	1
5	1	9	5290409	39.1	164	84	4	1	6	2
6	2	1	530255	37.3	104	35?	?		6	2
7	2	1	528355	?	?		2	1	3	1
8	1	1	526802	37.9	48	16	1	1	1	1
9	1	1	529607	?	60?		3?	?		1
10	2	1	530051	?	80	36	3	4	3	1
11	2	9	5299629	38.3	90?		1?		1	1
12	1	1	528548	38.1	66	12	3	3	5	1
13	2	1	527927	39.1	72	52	2?		2	1
14	1	1	528031	37.2	42	12	2	1	1	1
15	2	9	5291329		38	92	1	1	2	1
16	1	1	534917	38.2	76	28	3	1	1	1
17	1	1	530233	37.6	96	48	3	1	4	1
18	1	9	5301219	?	128	36	3	3	4	2
19	2	1	526639	37.5	48	24?	?	?	?	
20	1	1	5290481	37.6	64	21	1	1	2	1
21	2	1	532110	39.4	110	35	4	3	6?	
22	1	1	530157	39.9	72	60	1	1	5	2
23	2	1	529340	38.4	48	16	1?		1	1
24	1	1	521681	38.6	42	34	2	1	4?	
25	1	9	534998	38.3	130	60?		3?		1

Figure 14: Fichier de données contenant des valeurs manquantes

	A	B	C	D	E	F	G	H	I	J
1	surgery	Age	Hospital Number	rectal temperature	pulse	respiratory rate	temperature of extremities	peripheral pulse	mucous membranes	capillary refill time
2	2.0	1	530101	38.5	66.0	28.0	3.0	3.0	2.8537549407114624	2.0
3	1.0	1	534817	39.2	88.0	20.0	2.348360655737705	2.017316017316017	4.0	1.0
4	2.0	1	530334	38.3	40.0	24.0	1.0	1.0	3.0	1.0
5	1.0	9	5290409	39.1	164.0	84.0	4.0	1.0	6.0	2.0
6	2.0	1	530255	37.3	104.0	35.0	2.348360655737705	2.017316017316017	6.0	2.0
7	2.0	1	528355	38.16791666666666	71.91304347826087	30.417355371900825	2.0	1.0	3.0	1.0
8	1.0	1	526802	37.9	48.0	16.0	1.0	1.0	1.0	1.0
9	1.0	1	529607	38.16791666666666	60.0	30.417355371900825	3.0	2.017316017316017	2.8537549407114624	1.0
10	2.0	1	530051	38.16791666666666	80.0	36.0	3.0	4.0	3.0	1.0
11	2.0	9	5299629	38.3	90.0	30.417355371900825	1.0	2.017316017316017	1.0	1.0
12	1.0	1	528548	38.1	66.0	12.0	3.0	3.0	5.0	1.0
13	2.0	1	527927	39.1	72.0	52.0	2.0	2.017316017316017	2.0	1.0
14	1.0	1	528031	37.2	42.0	12.0	2.0	1.0	1.0	1.0
15	2.0	9	5291329	38.0	92.0	28.0	1.0	1.0	2.0	1.0
16	1.0	1	534917	38.2	76.0	28.0	3.0	1.0	1.0	1.0
17	1.0	1	530233	37.6	96.0	48.0	3.0	1.0	4.0	1.0
18	1.0	9	5301219	38.16791666666666	128.0	36.0	3.0	3.0	4.0	2.0
19	2.0	1	526639	37.5	48.0	24.0	2.348360655737705	2.017316017316017	2.8537549407114624	1.30597014925373
20	1.0	1	5290481	37.6	64.0	21.0	1.0	1.0	2.0	1.0
21	2.0	1	532110	39.4	110.0	35.0	4.0	3.0	6.0	1.30597014925373
22	1.0	1	530157	39.9	72.0	60.0	1.0	1.0	5.0	2.0
23	2.0	1	529340	38.4	48.0	16.0	1.0	2.017316017316017	1.0	1.0
24	1.0	1	521681	38.6	42.0	34.0	2.0	1.0	4.0	1.30597014925373
25	1.0	9	534998	38.3	130.0	60.0	2.348360655737705	3.0	2.8537549407114624	1.0

Figure 15: Fichier après répétition des valeurs manquantes

2.3.11. La normalisation MIN/MAX :

La normalisation des données est une étape importante dans le processus de prétraitement, car elle permet de mettre toutes les variables sur une échelle commune, ce qui facilite la comparaison et l'interprétation des données.

La normalisation par MIN/MAX consiste à transformer les valeurs de chaque attribut de sorte qu'elles soient toutes comprises entre 0 et 1. Pour cela, nous avons utilisé la formule suivante :

$$X_{\text{norm}} = (X - X_{\text{min}}) / (X_{\text{max}} - X_{\text{min}})$$

Où :

- X_{norm} est la valeur normalisée,
- X est la valeur originale de l'attribut,
- X_{min} est la valeur minimale de l'attribut,
- X_{max} est la valeur maximale de l'attribut.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	age	sex	cp	tubps	chol	tbs	resteca	thalachh	exng	oldpeak	slp	caa	thal	output
2	0.6458333333333334	1.0	1.0	0.48113207547169823	0.24429223744292233	1.0	0.0	0.6030534351145037	0.0	0.3709677419354838	0.0	0.0	0.3333333333333333	1.0
3	0.125	1.0	0.6666666666666666	0.339622641509434	0.28310502283105016	0.0	0.5	0.8854961832061069	0.0	0.564516129032258	0.0	0.0	0.6666666666666666	1.0
4	0.25	0.0	0.3333333333333333	0.339622641509434	0.1780821917808219	0.0	0.0	0.7709923664122137	0.0	0.2258064516129032	1.0	0.0	0.6666666666666666	1.0
5	0.5416666666666666	1.0	0.3333333333333333	0.24528301886792458	0.2511415525114155	0.0	0.5	0.816793893129771	0.0	0.12903225806451613	1.0	0.0	0.6666666666666666	1.0
6	0.5624999999999999	0.0	0.0	0.24528301886792458	0.5205479452054793	0.0	0.5	0.7022900763358778	1.0	0.0967741935483871	1.0	0.0	0.6666666666666666	1.0
7	0.5416666666666666	1.0	0.0	0.4339622641509434	0.1506849315068493	0.0	0.5	0.5877862595419846	0.0	0.06451612903225806	0.5	0.0	0.3333333333333333	1.0
9	0.3125	0.0	0.3333333333333333	0.4339622641509434	0.3835616438356165	0.0	0.0	0.6259541984732825	0.0	0.20967741935483872	0.5	0.0	0.6666666666666666	1.0
10	0.4791666666666666	1.0	0.3333333333333333	0.24528301886792458	0.3127853881278539	0.0	0.5	0.7786259541984731	0.0	0.0	1.0	0.0	1.0	1.0
11	0.5833333333333334	1.0	0.6666666666666666	0.7358490566037735	0.16666666666666669	1.0	0.5	0.6946564885496184	0.0	0.08064516129032258	1.0	0.0	1.0	1.0
12	0.5208333333333334	1.0	0.6666666666666666	0.5283018867924528	0.0958904109589041	0.0	0.5	0.7862595419847328	0.0	0.25806451612903225	1.0	0.0	0.6666666666666666	1.0
13	0.5208333333333334	1.0	0.0	0.4339622641509434	0.2579908675799087	0.0	0.5	0.6793893129770993	0.0	0.1935483870967742	1.0	0.0	0.6666666666666666	1.0
14	0.3958333333333337	0.0	0.6666666666666666	0.339622641509434	0.3401826484018265	0.0	0.5	0.5190839694656487	0.0	0.03225806451612903	1.0	0.0	0.6666666666666666	1.0
15	0.4166666666666663	1.0	0.3333333333333333	0.339622641509434	0.31963470319634696	0.0	0.5	0.763358778625954	0.0	0.0967741935483871	1.0	0.0	0.6666666666666666	1.0
16	0.7291666666666666	1.0	1.0	0.15094339622641517	0.1940639269406393	0.0	0.0	0.5572519083969466	1.0	0.2903225806451613	0.5	0.0	0.6666666666666666	1.0
17	0.5416666666666666	0.0	1.0	0.5283018867924528	0.3584474885844749	1.0	0.0	0.6946564885496184	0.0	0.16129032258064516	1.0	0.0	0.6666666666666666	1.0
18	0.4374999999999999	0.0	0.6666666666666666	0.24528301886792458	0.2123287671232877	0.0	0.5	0.6641221374045801	0.0	0.25806451612903225	0.5	0.0	0.6666666666666666	1.0
19	0.6041666666666666	0.0	0.6666666666666666	0.24528301886792458	0.4885844748858447	0.0	0.5	0.7709923664122137	0.0	0.0	1.0	0.0	0.6666666666666666	1.0
20	0.7708333333333334	0.0	1.0	0.5283018867924528	0.22831050228310495	0.0	0.5	0.3282442748091603	0.0	0.41935483870967744	0.0	0.0	0.6666666666666666	1.0
21	0.2291666666666663	1.0	0.0	0.5283018867924528	0.2762557077625571	0.0	0.5	0.763358778625954	0.0	0.24193548387096775	1.0	0.0	0.6666666666666666	1.0
22	0.8333333333333334	0.0	1.0	0.4339622641509434	0.2579908675799087	0.0	0.5	0.6106870229007634	0.0	0.2903225806451613	1.0	0.5	0.6666666666666666	1.0
23	0.6249999999999999	0.0	0.0	0.3867924528301888	0.24657534246575336	0.0	0.5	0.6870229007633587	0.0	0.08064516129032258	1.0	0.0	1.0	1.0
24	0.3125	1.0	0.6666666666666666	0.339622641509434	0.24429223744292233	0.0	0.5	0.8244274809160305	1.0	0.06451612903225806	1.0	0.0	0.6666666666666666	1.0
25	0.2708333333333337	1.0	0.0	0.4339622641509434	0.22831050228310495	0.0	0.5	0.816793893129771	0.0	0.0	1.0	0.0	0.6666666666666666	1.0
26	0.6666666666666666	1.0	0.6666666666666666	0.5283018867924528	0.2671232876712329	1.0	0.5	0.5038167938931296	1.0	0.16129032258064516	0.5	0.0	0.6666666666666666	1.0

Figure 16: Normalisation MIN/MAX des données

2.3.12. La normalisation Z-score :

En normalisant les données par ZScore, nous avons garanti que toutes les variables sont mises à la même échelle, ce qui réduit les biais potentiels dans nos analyses et nos modèles. Cela nous permet d'obtenir des résultats plus cohérents et plus significatifs lors de l'exploration et de l'interprétation des données.

	A	B	C	D	E
1	age	sex	cp	trtbps	chol
2	0.6570084677968723	0.6888321972137825	1.9301866716889002	0.7789777005552799	-0.2904843468130696
3	-2.0852712432559297	0.6888321972137825	0.954969979174753	-0.07874857120167622	0.03956875121944903
4	-1.427124112603257	-1.4517323726225952	-0.020246713339394155	-0.07874857120167622	-0.8535161022803074
5	0.10855252558631195	0.6888321972137825	-0.020246713339394155	-0.6505660857063136	-0.23223968245438986
6	0.21824371402842402	-1.4517323726225952	-0.9954634058535413	-0.6505660857063136	2.058717115653681
7	0.10855252558631195	0.6888321972137825	-0.9954634058535413	0.4930689433029612	-1.0864947597150263
8	0.21824371402842402	-1.4517323726225952	-0.020246713339394155	0.4930689433029612	0.8938238284800856
9	-1.0980505472769209	0.6888321972137825	-0.020246713339394155	-0.6505660857063136	0.291962296773728
10	-0.22052103974002427	0.6888321972137825	0.954969979174753	2.3228849897178008	-0.950590542878107
11	0.3279349024705361	0.6888321972137825	0.954969979174753	1.0648864578075987	-1.5524520745844645
12	-0.0011386628558001234	0.6888321972137825	-0.9954634058535413	0.4930689433029612	-0.1739950180957101
13	-0.6592857935084726	-1.4517323726225952	0.954969979174753	-0.07874857120167622	0.5249409542084471
14	-0.5495946050663605	0.6888321972137825	-0.020246713339394155	-0.07874857120167622	0.3502069611324078
15	1.0957732215653206	0.6888321972137825	1.9301866716889002	-1.2223836002109512	-0.7176118854433879
16	0.10855252558631195	-1.4517323726225952	1.9301866716889002	1.0648864578075987	0.6802600591649264
17	-0.4399034166242484	-1.4517323726225952	0.954969979174753	-0.6505660857063136	-0.5622927804869086
18	0.43762609091264815	-1.4517323726225952	0.954969979174753	-0.6505660857063136	1.7869086819798419
19	1.3151555984495447	-1.4517323726225952	1.9301866716889002	1.0648864578075987	-0.42638856364998906
20	-1.5368153010453691	0.6888321972137825	-0.9954634058535413	1.0648864578075987	-0.01867591313923074
21	1.644229163775881	-1.4517323726225952	1.9301866716889002	0.4930689433029612	-0.1739950180957101
22	0.5473172793547603	0.6888321972137825	-0.9954634058535413	0.2071601860506425	-0.27106945869350973
23	-1.0980505472769209	0.6888321972137825	0.954969979174753	-0.07874857120167622	-0.2904843468130696
24	-1.317432924161145	0.6888321972137825	-0.9954634058535413	0.4930689433029612	-0.42638856364998906
25	0.7666996562389844	0.6888321972137825	0.954969979174753	1.0648864578075987	-0.09633546561747042

Figure 17: Normalisation Z-score

3. TP N°2 : Clustering avec K-Means et K-Medoids

Le TP2 se concentre sur l'application des techniques de clustering, une méthode essentielle de la fouille de données, à un ensemble de données spécifique. L'objectif principal est d'explorer et de découvrir des structures intrinsèques dans nos données en regroupant les observations similaires en clusters ou en groupes. Nous nous concentrons sur l'utilisation des algorithmes de clustering tels que K-Means et K-Medoids pour effectuer cette tâche.

Une partie importante de ce TP consiste à déterminer le nombre optimal de clusters, ce qui est important pour obtenir des résultats significatifs. Nous utilisons des techniques telles que la courbe d'Elbow pour aider à sélectionner ce nombre de manière empirique, facilitant ainsi une segmentation appropriée des données.

3.1. Identification du nombre optimal de clusters avec la courbe d'Elbow :

La courbe d'Elbow est une méthode graphique qui nous permet d'évaluer la qualité des clusters en fonction du nombre de clusters choisi. Elle trace la variance expliquée par le modèle en fonction du nombre de clusters, et nous observons généralement un coude (elbow) dans le graphique où l'ajout de clusters supplémentaires ne produit plus de gain significatif en termes de variance expliquée.

L'objectif est de sélectionner le nombre de clusters qui se trouve au niveau du coude, car cela indique un compromis optimal entre la complexité du modèle et sa capacité à expliquer la variance des données. Une fois le nombre optimal de clusters identifié, nous pouvons procéder à l'application des algorithmes de clustering avec ce nombre de clusters pour segmenter nos données de manière appropriée.

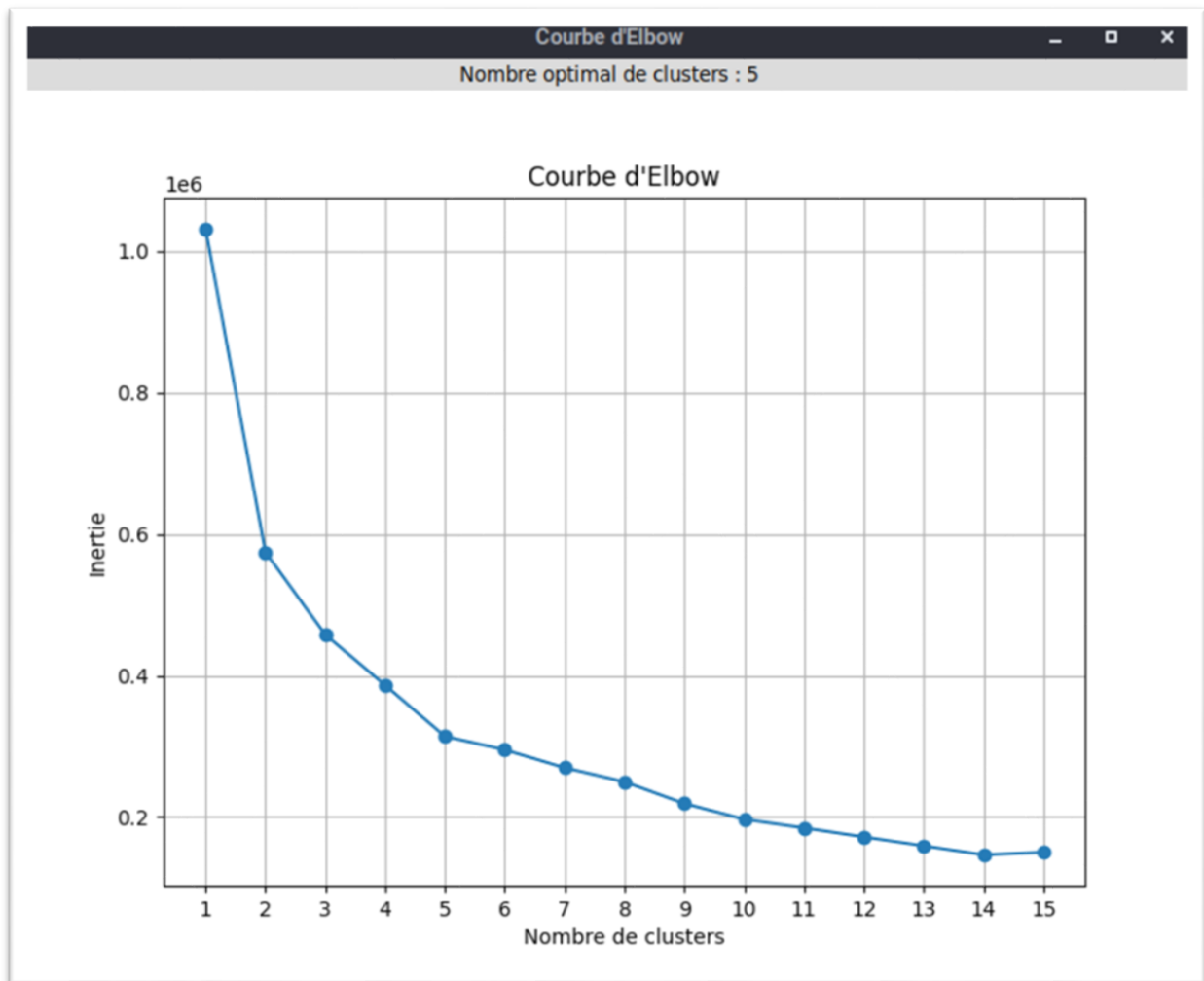


Figure 18: Courbe d'Elbow pour l'identification du nombre optimal de clusters

3.1.1. Interprétation :

La courbe d'Elbow montre une décroissance de l'inertie intra-cluster à mesure que le nombre de clusters augmente. Cependant, à partir de cinq clusters, cette décroissance devient moins prononcée, formant un coude en montrant cinq clusters représentent un compromis optimal entre la réduction de l'inertie intra-cluster et la complexité du modèle. Ainsi, cinq clusters peuvent être considérés comme le nombre optimal pour notre ensemble de données.

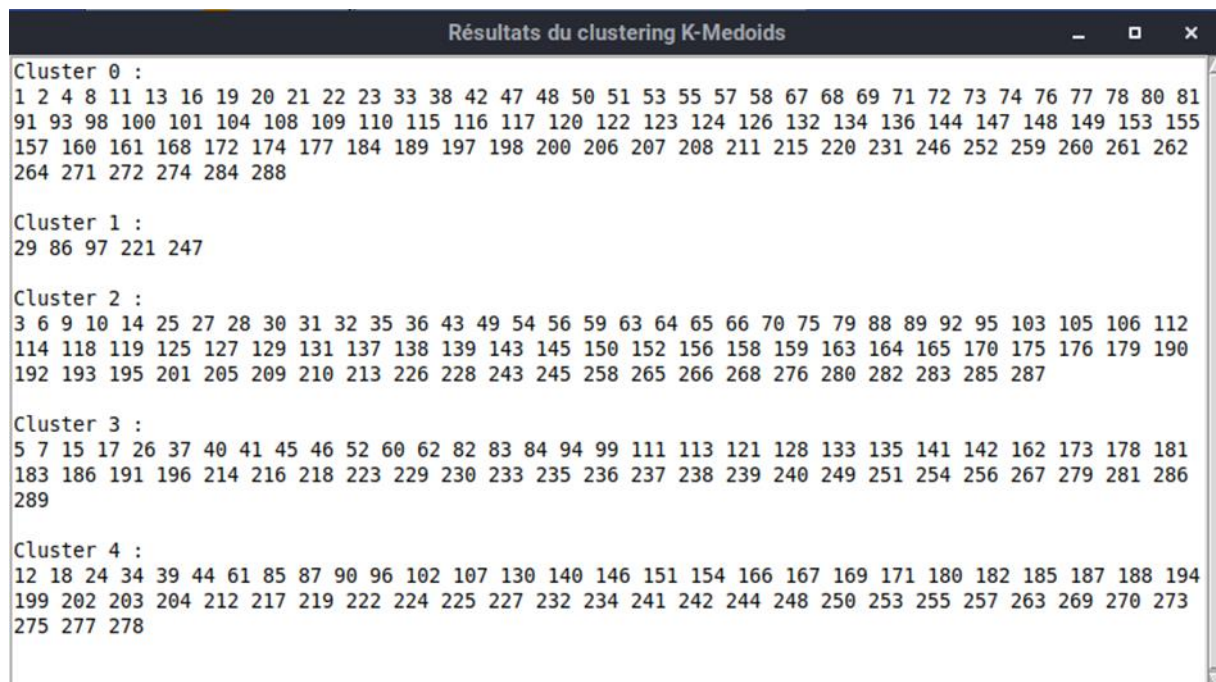
3.2. Application de l'algorithme K-Means :

L'algorithme K-Means est l'un des algorithmes de clustering les plus couramment utilisés, il divise un ensemble de données en k clusters de manière à minimiser la variance intra-cluster tout en maximisant la variance entre les clusters.

L'application de l'algorithme K-Means se déroule en plusieurs étapes :

1. Initialisation : Nous commençons par choisir aléatoirement k centres de cluster dans l'espace des données.
2. Attribution : Chaque point de données est attribué au cluster dont le centre est le plus proche.
3. Réaffectation : Les centres des clusters sont recalculés en prenant la moyenne des points attribués à chaque cluster.
4. Convergence : Les étapes 2 et 3 sont répétées jusqu'à ce qu'il n'y ait plus de changements significatifs dans l'attribution des points aux clusters.

Une fois que l'algorithme converge, nous obtenons les clusters finaux, où chaque cluster est représenté par son centre. Ces clusters nous fournissent une segmentation de nos données en groupes homogènes, ce qui facilite l'analyse et l'interprétation des tendances et des schémas présents dans les données.



```
Résultats du clustering K-Medoids
Cluster 0 :
1 2 4 8 11 13 16 19 20 21 22 23 33 38 42 47 48 50 51 53 55 57 58 67 68 69 71 72 73 74 76 77 78 80 81
91 93 98 100 101 104 108 109 110 115 116 117 120 122 123 124 126 132 134 136 144 147 148 149 153 155
157 160 161 168 172 174 177 184 189 197 198 200 206 207 208 211 215 220 231 246 252 259 260 261 262
264 271 272 274 284 288

Cluster 1 :
29 86 97 221 247

Cluster 2 :
3 6 9 10 14 25 27 28 30 31 32 35 36 43 49 54 56 59 63 64 65 66 70 75 79 88 89 92 95 103 105 106 112
114 118 119 125 127 129 131 137 138 139 143 145 150 152 156 158 159 163 164 165 170 175 176 179 190
192 193 195 201 205 209 210 213 226 228 243 245 258 265 266 268 276 280 282 283 285 287

Cluster 3 :
5 7 15 17 26 37 40 41 45 46 52 60 62 82 83 84 94 99 111 113 121 128 133 135 141 142 162 173 178 181
183 186 191 196 214 216 218 223 229 230 233 235 236 237 238 239 240 249 251 254 256 267 279 281 286
289

Cluster 4 :
12 18 24 34 39 44 61 85 87 90 96 102 107 130 140 146 151 154 166 167 169 171 180 182 185 187 188 194
199 202 203 204 212 217 219 222 224 225 227 232 234 241 242 244 248 250 253 255 257 263 269 270 273
275 277 278
```

Figure 19: Résultats du clustering K-means

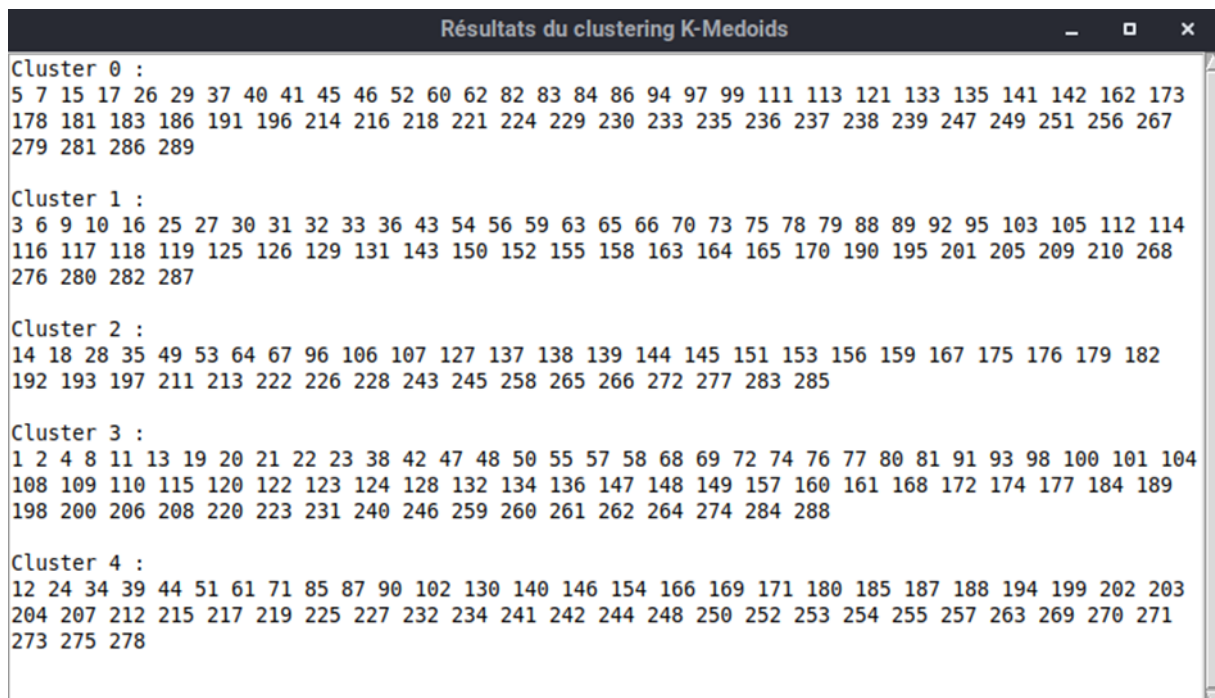
3.3. Application de l'algorithme K-Medoids :

L'algorithme de clustering K-Medoids à nos données. Contrairement à l'algorithme K-Means qui utilise les moyennes comme centres de clusters, K-Medoids utilise des points de données réels comme médoids, ce qui le rend plus robuste aux valeurs aberrantes.

Le processus d'application de l'algorithme K-Medoids est similaire à celui de K-Means :

1. Initialisation : Nous sélectionnons initialement k points de données comme médoids.
2. Attribution : Chaque point de données est attribué au médoid le plus proche, formant ainsi des clusters.
3. Réaffectation : Nous cherchons à minimiser la somme des distances entre chaque point de données et son médoid dans chaque cluster.
4. Convergence : Les étapes 2 et 3 sont répétées jusqu'à ce qu'il n'y ait plus de changements significatifs dans l'attribution des points aux clusters.

Une fois que l'algorithme converge, nous obtenons les clusters finaux, où chaque cluster est représenté par son médoid. Ces clusters nous fournissent une segmentation de nos données en groupes homogènes, ce qui facilite l'analyse et l'interprétation des tendances et des schémas présents dans les données. Comparé à K-Means, K-Medoids est souvent préféré lorsque les données sont plus sensibles aux valeurs aberrantes ou lorsque les distances entre les points ne sont pas facilement définies en termes de moyennes.



```
Résultats du clustering K-Medoids

Cluster 0 :
5 7 15 17 26 29 37 40 41 45 46 52 60 62 82 83 84 86 94 97 99 111 113 121 133 135 141 142 162 173
178 181 183 186 191 196 214 216 218 221 224 229 230 233 235 236 237 238 239 247 249 251 256 267
279 281 286 289

Cluster 1 :
3 6 9 10 16 25 27 30 31 32 33 36 43 54 56 59 63 65 66 70 73 75 78 79 88 89 92 95 103 105 112 114
116 117 118 119 125 126 129 131 143 150 152 155 158 163 164 165 170 190 195 201 205 209 210 268
276 280 282 287

Cluster 2 :
14 18 28 35 49 53 64 67 96 106 107 127 137 138 139 144 145 151 153 156 159 167 175 176 179 182
192 193 197 211 213 222 226 228 243 245 258 265 266 272 277 283 285

Cluster 3 :
1 2 4 8 11 13 19 20 21 22 23 38 42 47 48 50 55 57 58 68 69 72 74 76 77 80 81 91 93 98 100 101 104
108 109 110 115 120 122 123 124 128 132 134 136 147 148 149 157 160 161 168 172 174 177 184 189
198 200 206 208 220 223 231 240 246 259 260 261 262 264 274 284 288

Cluster 4 :
12 24 34 39 44 51 61 71 85 87 90 102 130 140 146 154 166 169 171 180 185 187 188 194 199 202 203
204 207 212 215 217 219 225 227 232 234 241 242 244 248 250 252 253 254 255 257 263 269 270 271
273 275 278
```

Figure 20: Résultats du clustering K-medoids

3.4. Évaluation des performances des clusters :

L'évaluation des performances des clusters est essentielle pour mesurer l'efficacité des algorithmes de clustering dans la segmentation des données.

Pour évaluer les performances des clusters, nous utilisons généralement des mesures telles que l'inertie intra-cluster et l'inertie inter-cluster.

1. L'inertie intra-cluster mesure la cohésion des clusters en calculant la somme des distances euclidiennes entre chaque point de données et le centre de son cluster. Une faible inertie intra-cluster indique que les points au sein de chaque cluster sont proches les uns des autres, ce qui indique une bonne cohésion intra-cluster.
2. L'inertie inter-cluster mesure la séparation entre les clusters en calculant la somme des distances entre les centres de tous les clusters. Une faible inertie inter-cluster indique que les centres des clusters sont bien séparés les uns des autres, ce qui signifie une bonne séparation inter-cluster.

En évaluant ces deux mesures, nous sommes en mesure de déterminer la qualité des clusters obtenus avec chaque algorithme de clustering. Des valeurs d'inertie intra-cluster faibles et d'inertie inter-cluster élevées indiquent une segmentation précise des données avec des clusters compacts et bien séparés. Ces mesures nous aident à comparer les performances des algorithmes de clustering et à sélectionner celui qui convient le mieux à notre ensemble de données spécifique.



Figure 21: Performances de KMeans

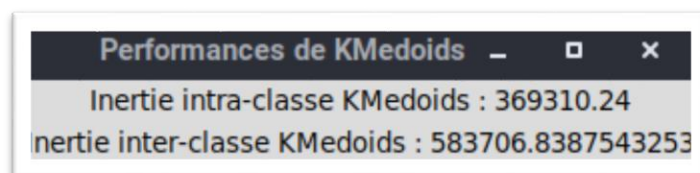


Figure 22: Performances de KMedoids

4. TP N°3 : Clustering avec Agnes, Diana et DBSCAN :

Le TP3 se concentre sur l'application de différentes techniques de clustering, telles qu'Agglomerative Hierarchical Clustering (AGNES), Divisive Analysis Clustering (DIANA), et Density-Based Spatial Clustering of Applications with Noise (DBSCAN), à un ensemble de données donné. L'objectif principal est d'explorer ces algorithmes de clustering et d'évaluer leurs performances sur des données réelles, en mettant l'accent sur la découverte de structures intrinsèques et la détection de groupes similaires au sein des données.

4.1. Application l'algorithme Agnes :

AGNES est une méthode de clustering hiérarchique qui construit une hiérarchie de clusters en fusionnant progressivement les clusters les plus similaires.

Le processus d'application de l'algorithme AGNES se déroule comme suit :

- **Initialisation** : Chaque point de données est initialement considéré comme un cluster individuel.
- **Calcul de la similarité** : La similarité entre chaque paire de clusters est calculée en utilisant une mesure de distance appropriée, telle que la distance euclidienne.
- **Fusion des clusters** : Les deux clusters les plus similaires sont fusionnés pour former un nouveau cluster.
- **Répétition** : Les étapes 2 et 3 sont répétées jusqu'à ce qu'un seul cluster contenant tous les points de données soit obtenu.

Une fois que l'algorithme AGNES est terminé, nous obtenons une hiérarchie de clusters qui peut être représentée sous forme de dendrogramme. Cette représentation hiérarchique permet d'explorer différentes granularités de clusters, allant des clusters individuels aux clusters globaux, ce qui offre une flexibilité dans l'interprétation des résultats du clustering.

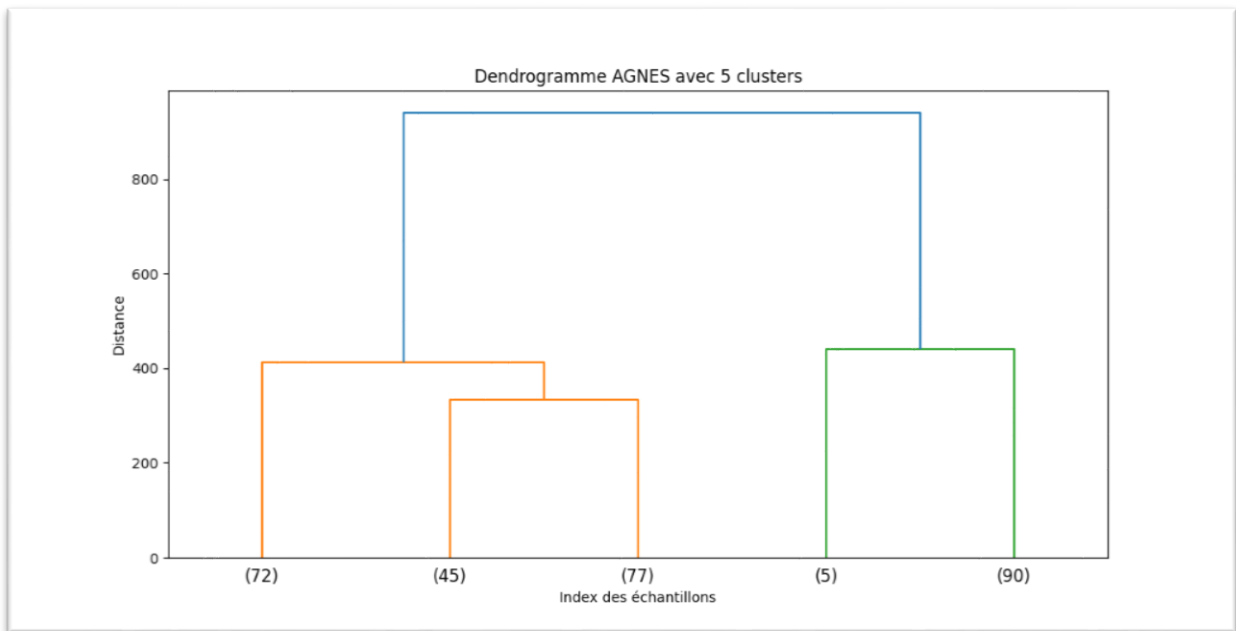


Figure 23: Dendrogramme résultant de l'algorithme AGNES

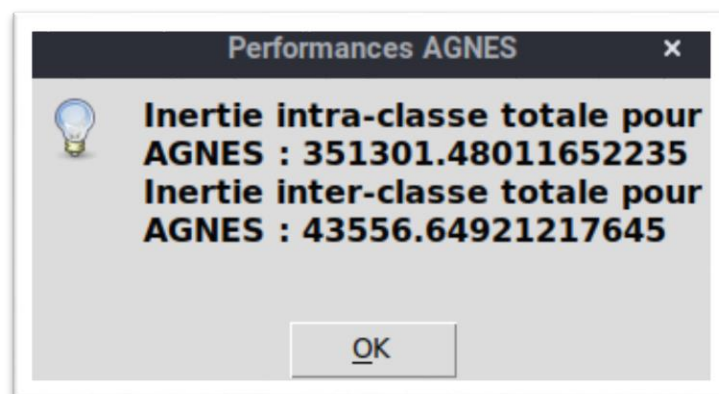


Figure 24: Performances de Agnes

4.2. Application l'algorithme Diana:

DIANA est une méthode de clustering hiérarchique qui fonctionne de manière inverse à AGNES. Contrairement à AGNES, qui commence avec tous les points dans un cluster et les divise progressivement, DIANA commence avec un seul cluster contenant tous les points de données et le divise progressivement en sous-clusters plus petits.

Le processus d'application de l'algorithme DIANA se déroule comme suit :

- Initialisation : Tous les points de données sont regroupés dans un seul cluster.
- Séparation des clusters : Les points de données au sein du cluster sont analysés pour déterminer les sous-clusters les plus disparates. Ces sous-clusters sont ensuite séparés du cluster d'origine.

- Répétition : Les étapes 2 sont répétées de manière itérative jusqu'à ce que chaque cluster ne puisse plus être divisé.

Une fois que l'algorithme DIANA est terminé, nous obtenons une hiérarchie de clusters, souvent représentée sous forme de dendrogramme. Cette représentation hiérarchique nous permet d'explorer différentes granularités de clusters, ce qui offre une flexibilité dans l'interprétation des résultats du clustering.

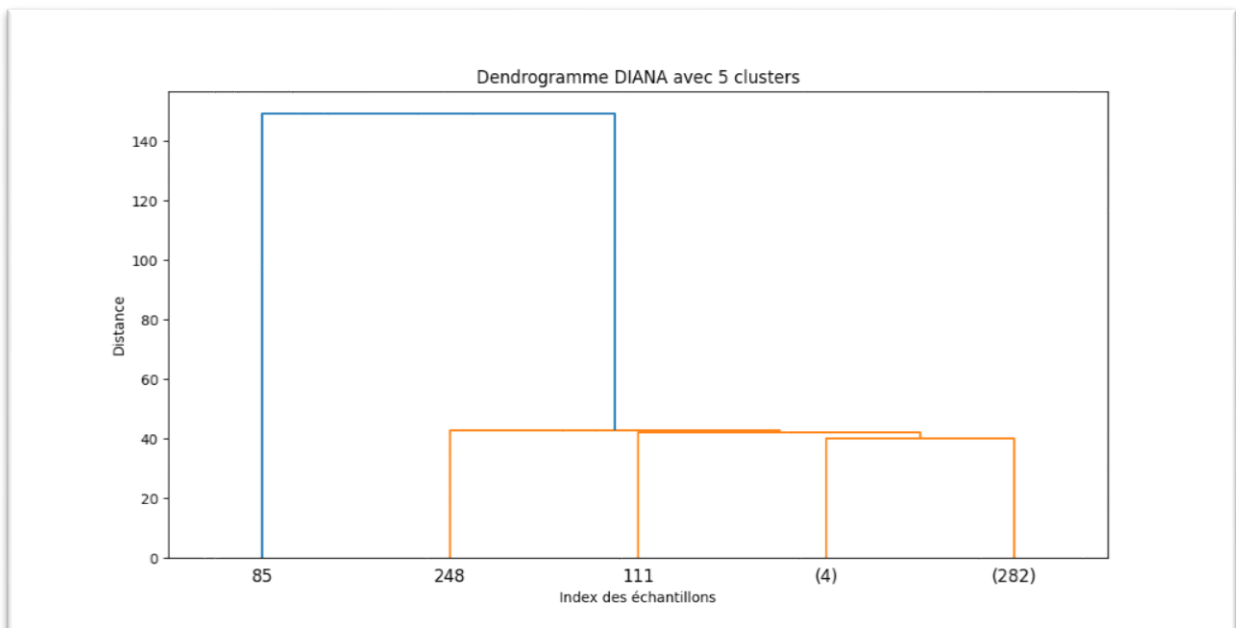


Figure 25: Dendrogramme résultant de l'algorithme DIANA

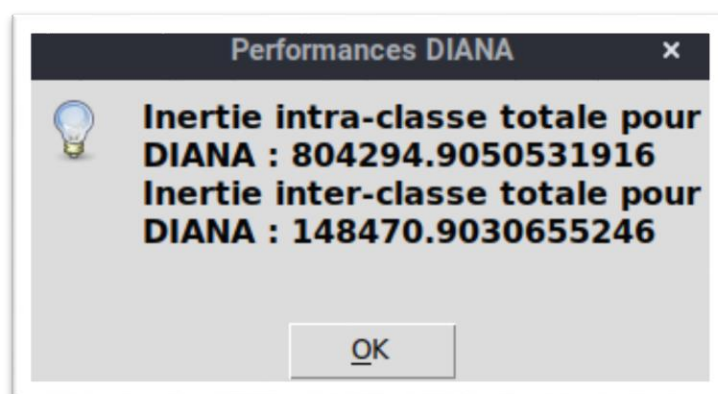


Figure 26 : Performances de DIANA

4.3. Application l'algorithme DBSCAN:

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) est un algorithme de clustering largement utilisé dans le domaine de l'apprentissage automatique non supervisé. Contrairement à K-means, qui nécessite de spécifier le nombre de clusters à l'avance, DBSCAN peut identifier le nombre de clusters dans les données lui-même. Il est particulièrement efficace pour trouver des clusters de forme arbitraire dans des ensembles de données à forte densité et peut également gérer efficacement du bruit.

Voici une explication détaillée de l'algorithme DBSCAN :

1. Densité et points de base :

- DBSCAN se base sur deux paramètres principaux : la distance minimale (eps) et le nombre minimum de points (MinPts).
- Un point est considéré comme un point de base s'il a au moins MinPts voisins dans une sphère de rayon eps autour de lui, y compris lui-même.
- Les points de base sont les points à partir desquels DBSCAN commence à développer des clusters.

2. Expansion des clusters :

- À partir de chaque point de base ou point déjà visité, DBSCAN explore tous les points atteignables dans la sphère de rayon eps.
- Si un point atteignable est également un point de base, tous les points atteignables depuis ce point sont également inclus.
- Ce processus se poursuit récursivement jusqu'à ce que tous les points atteignables aient été visités.

3. Types de points :

- Un point peut être classé en trois catégories dans DBSCAN :
 - **Point central** : Un point de base avec au moins MinPts voisins dans sa sphère eps.
 - **Point bord** : Un point qui n'est pas un point central mais se trouve dans la sphère eps d'un point central.
 - **Point de bruit** : Un point qui n'est pas un point central et qui n'a pas de points dans sa sphère eps.

4. Formation de clusters :

- DBSCAN forme un cluster en regroupant tous les points (centraux et bord) atteignables depuis un point central.
- Les points bord peuvent être inclus dans plusieurs clusters s'ils sont à la frontière de plusieurs clusters.

5. Traitement du bruit :

- Les points de bruit ne sont pas inclus dans des clusters mais sont souvent considérés comme du bruit ou des outliers.
- Cela rend DBSCAN robuste au bruit dans les données.

6. Avantages de DBSCAN :

- Peut identifier des clusters de forme arbitraire.
- Peut gérer efficacement du bruit et des outliers.
- Nécessite moins de paramètres à ajuster par rapport à K-means.

7. Inconvénients de DBSCAN :

- Sensible aux paramètres eps et MinPts, qui peuvent nécessiter un réglage minutieux.
- Peut avoir du mal avec des clusters de densités différentes.

```
Clusters DBScan  _  □  ×
Cluster 0:
10, 20, 37, 49, 75, 92, 107, 119, 188
-----
Cluster 1:
21, 22, 41, 46, 67, 259
-----
Cluster 2:
5, 29, 55, 64, 69, 91, 104, 117, 128, 169, 281, 28
6
-----
Cluster 3:
11, 33, 43, 47, 50, 70, 76, 114, 135, 168, 170, 18
6, 201, 205, 211, 214, 219, 249, 256, 270, 277
-----
Cluster 4:
2, 15, 68, 74, 115, 116, 142, 148, 159, 275
-----
```

Figure 27: Résultats du clustering DBScan

4.3.1. Évaluation des performances des clusters :

Nous évaluons les performances des clusters obtenus avec les algorithmes AGNES et DIANA en utilisant les mesures d'inertie intra-cluster et d'inertie inter-cluster

Pour l'algorithme DBSCAN, nous évaluons les performances des clusters en utilisant trois mesures différentes :

La silhouette : Cette mesure évalue à quel point chaque point est similaire à son propre cluster par rapport aux autres clusters. La silhouette varie de -1 à 1, où une valeur élevée indique que le point est bien assorti à son propre cluster et mal assorti aux clusters voisins.

L'indice de Calinski-Harabasz : Cette mesure calcule le rapport entre la dispersion inter-cluster et la dispersion intra-cluster. Un indice de Calinski-Harabasz plus élevé indique une meilleure séparation entre les clusters.

L'indice de Davies-Bouldin : Cette mesure évalue la compacité des clusters ainsi que leur séparation les uns par rapport aux autres. Un indice de Davies-Bouldin plus faible indique une meilleure partition des données.

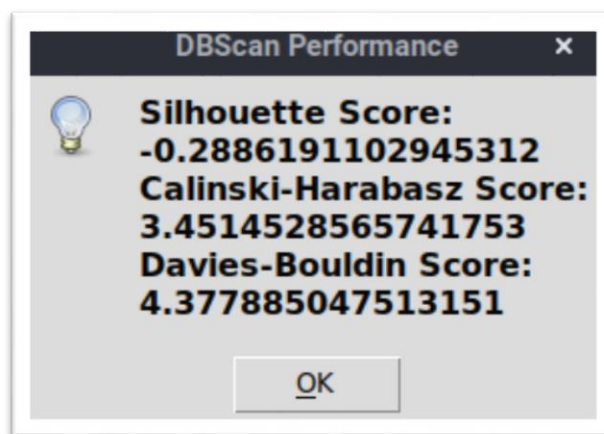


Figure 28: Performances DBScan

4.3.1.1. Interprétation :

Silhouette Score : Le score de silhouette est négatif, ce qui indique que les clusters obtenus ne sont pas bien séparés. Un score de silhouette proche de 0 indique que les clusters se chevauchent ou sont mal définis, ce qui peut être le cas lorsque les données ont une densité variable ou que les clusters sont mal délimités.

Calinski-Harabasz Index : Avec un score de 3.45, le critère de Calinski-Harabasz est relativement faible. Cela indique que les clusters ne sont pas très compacts et que la séparation entre les clusters n'est pas très claire. Un score plus élevé aurait indiqué une meilleure séparation entre les clusters.

Davies-Bouldin Index : Avec un score de 4.38, le critère de Davies-Bouldin est également élevé. Cela signifie qu'il y a une certaine similarité moyenne entre les clusters, ce qui peut être dû à une mauvaise séparation entre les clusters ou à une densité variable des données.

5. Comparaison des méthodes à travers un histogramme des inerties :

La comparaison des performances des méthodes de clustering est essentielle pour déterminer laquelle est la plus adaptée à un ensemble de données particulier. Une façon courante d'évaluer ces performances est de comparer les inerties intra-cluster obtenues à partir de différentes méthodes.

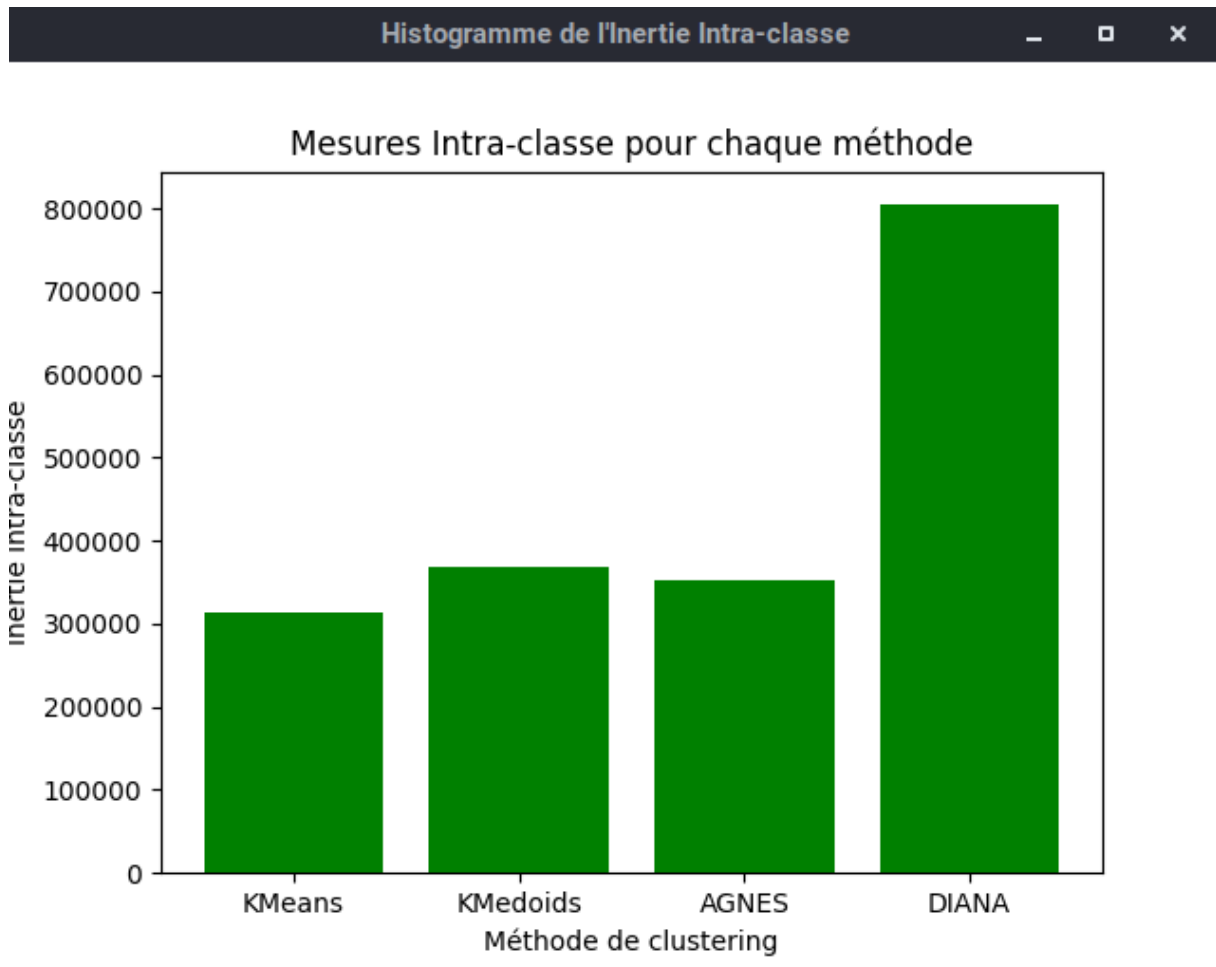


Figure 29: Histogramme de l'inertie intra-classe

Plus l'inertie intra-cluster est faible, meilleure est la cohésion des clusters.

Dans cet histogramme, on constate que l'algorithme K-means a la plus faible inertie intra-cluster, suivie de près par Agnes, puis K-medoids, et enfin Diana. Cela indique que K-means a formé des clusters plus compacts et cohérents que les autres méthodes.

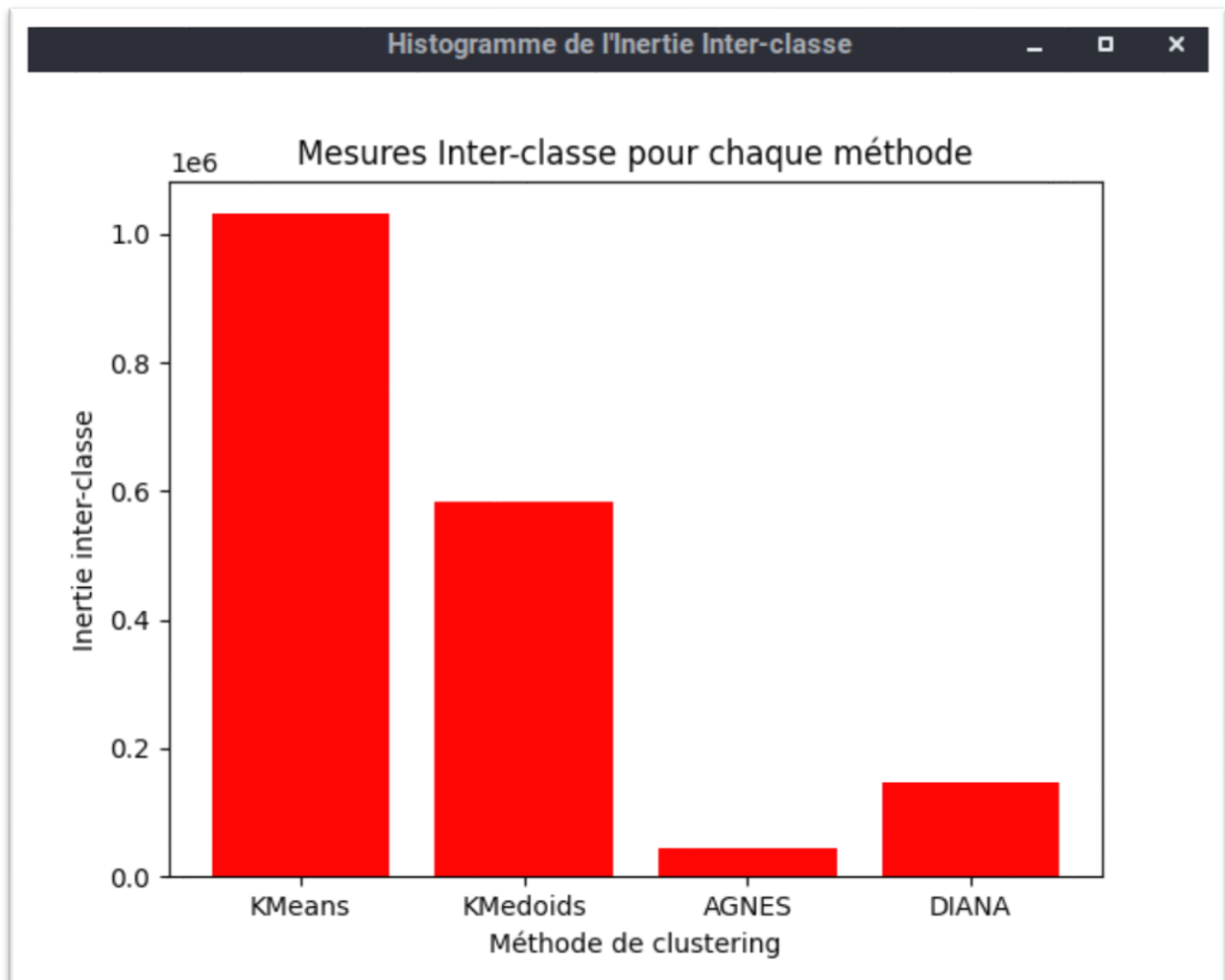


Figure 30: Histogramme de l'inertie inter-classe

Plus l'inertie inter-cluster est élevée, meilleure est la séparation entre les clusters.

Dans cet histogramme, on remarque que l'algorithme K-means a la plus élevée inertie inter-cluster, suivie de Diana, K-medoids, et enfin Agnes. Cela indique que K-means a réussi à séparer efficacement les clusters, tandis que Agnes a obtenu la séparation la moins efficace.

Décision finale :

En considérant à la fois l'inertie intra-cluster et inter-cluster, ainsi que les performances relatives des méthodes, il semble que K-means soit la meilleure méthode pour ces données. Elle a produit des clusters compacts et bien séparés par rapport aux autres méthodes.

6. Conclusion :

Notre parcours dans ce projet de fouille de données a été une véritable aventure à travers les dédales de nos informations. À chaque étape, nous avons plongé dans un monde de nombres et de schémas, cherchant à extraire des connaissances utiles. Parmi les différentes méthodes explorées, les algorithmes de regroupement ont joué un rôle central en nous aidant à diviser nos données en groupes significatifs.

Une des principales conclusions que nous avons tirées est que l'algorithme K-means est très polyvalent. Il a la capacité de créer des groupes bien définis et compacts, ce qui en fait une option solide pour nos données. Cependant, son principal inconvénient est qu'il faut souvent spécifier à l'avance le nombre de groupes, ce qui peut être difficile lorsque la structure des données n'est pas claire.

D'autre part, les méthodes basées sur la densité, comme DBSCAN, sont une alternative intéressante. Elles peuvent détecter des groupes de forme arbitraire et gérer efficacement les valeurs aberrantes, ce qui en fait des outils puissants dans certaines situations. Cependant, leur performance dépend beaucoup des paramètres choisis, comme la distance de voisinage.

Les méthodes hiérarchiques, comme Agnes et Diana, offrent une approche ascendante ou descendante pour diviser nos données en groupes. Leur capacité à révéler une structure hiérarchique peut être utile pour comprendre les relations entre les groupes, mais elles peuvent être sensibles à la taille et à la complexité des données.

En résumé, chaque méthode de regroupement a ses avantages et ses limites, et le choix dépendra des caractéristiques de nos données et de nos objectifs d'analyse. Ce projet nous a permis de mieux comprendre ces nuances et de prendre des décisions éclairées. Au-delà des résultats spécifiques, il nous a ouvert les portes d'un monde fascinant d'analyse des données, nous incitant à continuer à explorer les trésors cachés dans les vastes océans de données qui nous entourent.

6.1. Perspectives pour de futurs travaux en fouille de données :

Dans le cadre de la fouille de données, plusieurs perspectives peuvent être envisagées pour des travaux futurs, notamment :

- a. **Exploration de nouvelles techniques de clustering** : La fouille de données est un domaine en évolution constante, avec l'émergence de nouvelles techniques et algorithmes de clustering. Il serait intéressant d'explorer des approches plus récentes telles que le clustering spectral, le clustering basé sur la densité adaptative, ou encore les méthodes de clustering en ligne.
- b. **Intégration de données multi-sources** : De plus en plus de données sont disponibles à partir de différentes sources et de différentes modalités (texte, image, vidéo, etc.). Les futures recherches pourraient se concentrer sur le développement de techniques de clustering capables de traiter efficacement des ensembles de données hétérogènes et multimodaux.
- c. **Exploration de l'apprentissage non supervisé profond** : Les récentes avancées dans le domaine de l'apprentissage profond offrent de nouvelles opportunités pour la fouille de données. Les travaux futurs pourraient explorer l'utilisation de réseaux de neurones profonds pour le clustering, en tirant parti de leur capacité à extraire des représentations de données complexes et à apprendre des structures sous-jacentes.
- d. **Intégration de l'interprétabilité** : L'interprétabilité des résultats de clustering est un aspect important, en particulier dans les domaines où la prise de décision humaine est nécessaire. Les travaux futurs pourraient se concentrer sur le développement de techniques de clustering interprétables et sur la compréhension des facteurs qui influent sur la qualité et la robustesse des clusters obtenus.
- e. **Applications spécifiques** : Enfin, les futurs travaux en fouille de données pourraient se concentrer sur des domaines d'application spécifiques, tels que la santé, le commerce électronique, la biologie, etc. En adaptant les techniques de clustering aux besoins et aux caractéristiques spécifiques de chaque domaine, il est possible d'obtenir des résultats plus pertinents et plus significatifs.