Université Abou Bekr Belkaid
Tlemcen Algérie
جامعة أبي بكر بلقايد
تلمسان الجزائر

République Algérienne Démocratique et Populaire
Université Abou Bakr Belkaid– Tlemcen
Faculté de Technologie
Département d'Informatique

Mémoire de fin d'études

pour l'obtention du diplôme de Licence en Informatique

Thème

La classification hiérarchique ascendante

Réalisé par :

- Abdellaoui Nezha Nesrine

Présenté le 10Juin 2014 devant la commission d'examination composée de MM.

- Chaouche ramadene Lamia *(Encadreur)*
- Lahassaini .M. *(Examinateur)*
- Benziane.Y. *(Examinateur)*

Année universitaire: 2013-2014

Remerciement

Je commence d'abord par remercier le bon dieu de m'avoir donné le courage et la volonté pour réaliser ce travail.

Je tiens remercier Madame Chaouche Ramadhan : mon encadreur que je ne saurais jamais remercier assez pour ses précieux conseils, sa compréhension et sa confiance en moi et que j'ai bien profité de son savoir et son expérience.

J'adresse mes respects à Monsieur Lahassaini.M. Qui m'a fait l'honneur de l'avoir jugé mon mémoire.

Tous mes sincères remerciements à Monsieur Benzian.Y d'avoir aimablement accepté de juger ce travail.

En fin, à tout les personne ayant contribué de prés au de loin à ce travail. Un grand merci

Dédicaces

Je dédie ce travail à :

A mes grands parents. Aucune dédicace ne saurait exprimer ce que je ressens pour vous. Je vous remercie pour tout le soutien exemplaire et l'amour exceptionnel que vous me portez et j'espère que votre bénédiction m'accompagnera toujours

A mes parents je vous remercie pour tout vous efforts, votre générosité et votre amour surtout je vous procurer santé et prospérité.

A mes tantes. Chérifa, Fadéla, Fatéma, Amina veillez percevoir à travers ce travail, l'expression de ma profonde affection. J'espère que vous retrouvez dans la dédicace de ce travail, le témoignage de mes sentiments sincères et mes vœux de santé et de bonheur.

A ma chère cousine Hidayet veuillez accepter l'expression de ma profonde gratitude pour votre soutien, encouragement, et affection je te souhaite beaucoup de réussite, de bonheur et prospérité.

A ma chère Meriem merci pour ton amour, amitié. Tu étais toujours là pour me soutenir, et m'écouter. Merci .

A tout mes enseignants tout au long de mes études.

Et enfin à tous ceux ou celles qui me sont chères et que j'ai omis involontairement de citer.

Table des matières

Liste des figures

Liste des tableaux

Liste des abréviations

ACP Analyse en Composante Principales

AFC Analyse Factorielle des Correspondances

AFCM Analyse Factorielle des Correspondances Multiples

AFD Analyse Factorielle Discriminante

API Application Programming Interface

CHA Classification Hiérarchique Ascendante

CHD Classification Hiérarchique Descendante

IDE Integreted Development Environnement

JDK Java Development Kit

JVM Machine Virtuelle Java

JRE Java Runtime Environment

K-PPV k-Plus Proche Voisin

SVM Machines à Support de Vecteurs

Rappel de définitions

CART dont l'acronyme signifie « Classification And Regression Trees », s'attelle à construire un arbre de décision en classifiant un ensemble d'enregistrements. Cet arbre fournit un modèle pour classer de nouveaux échantillons. Il a été publié par Leo Breiman en 1984.

CHAID (**CH**i-squared **A**utomatic **I**nteraction **D**etector) est une technique de type *arbre de décision*. Elle a été publiée en 1980 par Gordon V. Kass[1]. Elle peut être utilisée pour la prédiction (comme la régression linéaire) ou pour la détection d'interaction entre variables.

Parcimonie la parcimonie est un principe consistant à n'utiliser que le minimum de causes élémentaires pour expliquer un phénomène.

Discrimination : la discrimination consiste à déterminer une fonction qui sépare au mieux les données selon un critère prédéfini.
Échantillon : sous-ensemble de la population.
Individus ou unités statistiques : éléments de la population.

Inertie : valeur caractérisant la concentration ou la dispersion de points sur un axe, un plan ou tout espace. L'inertie peut être représentée par une variance.
Le tableau de Burt est une façon de disposer des informations d'ordre qualitatif afin de les traiter par le calcul.

Les *points atypiques* *(aberrants)* sont par définition des observations peu fréquentes, c'est-à-dire des points qui ne suivent pas la distribution caractéristique du reste des données.

Modalité : les modalités d'un caractère sont les valeurs (mesurable ou non) prises par cette variable.
Population : ensemble des données étudiées.

Ressemblance : deux individus se ressemblent, ou sont proches, s'ils possèdent des valeurs proches pour l'ensemble des variables.
Taxonomie : littéralement la science des lois de l'ordre, c'est la science de la classification, parfois limitée à la botanique.

Introduction générale

La classification est un des nombreux domaines de la Fouille de données qui vise à extraire l'information à partir de grands volumes de données en utilisant différentes techniques computationnelles de l'apprentissage, des statistiques et des reconnaissances des formes. On cite les deux méthodes principale supervisée et non supervisée. Une des deux approches fondamentales de la classification non supervisée est la classification hiérarchique qui inclut deux méthodes principale la classification hiérarchique ascendante et descendante. Ces méthodes hiérarchiques diffèrent entre elles par le choix du critère de ressemblance et par la façon de mesurer les ressemblances entre un nouveau groupe fusionné et les autres inchangés.

Dans le cadre de ce mémoire, nous décrivons les principales étapes de CAH qui définies par un certain nombre de variables, en les regroupant de façon hiérarchique. Elle commence par agréger celles qui sont les plus semblables entre elles, puis les observations ou groupes d'observations un peu moins semblables et ainsi de suite jusqu'au regroupement trivial de l'ensemble de l'échantillon. Ces agrégations se font deux à deux. Les liens hiérarchiques apparaissent sur un dendrogramme, qui nous montre les liaisons entre les classes et la hauteur des branches nous indique leur niveau de proximité.

Ce mémoire contient trois chapitres qui se répartissent comme suit :

Dans le premier chapitre un petit aperçu sur l'analyse des données qui est une analyse statistique et descriptive avec ses principes classes analyse factorielle et analyse par classification. Nous introduisons le concept de chaque méthode en présentant les objectifs et les domaines d'application.

Enfin une brève description sera donnée sur la qualité de classification.

Le deuxième chapitre présent une description précise de classification hiérarchique ascendante présentant ses principes les plus importants comme le calcul de distance, le partitionnement des classes, l'algorithme de CAH. Ensuite nous avons présentées le dendrogramme (arbre hiérarchique des données).

Le troisième chapitre ce chapitre est organisé sur deux parties :

La première : une représentation générale de l'interface graphique Netbeans et ces composants.

La deuxième : décrit l'application que nous avons effectuée sur un ensemble de points dans un plan avec les résultats obtenus

On termine par une conclusion générale de notre travail qui décrit l'objectif principal de ce travail.

Chapitre1 :

Introduction à l’analyse des données

Chapitre1 Analyse des données

1. Analyse des données

1.1. Définition

L'analyse des données c'est aujourd'hui l'expression consacré pour désigner les analyses statistiques descriptives multidimensionnelles. [1] Elle autorise des études globales incluant toutes les caractéristiques de ces données, elle permet de traiter un nombre très important de données et faire sortir les relations pouvant existées entre eux la regroupe de façon à faire apparaitre clairement et les rend homogènes.

1.2. Les méthodes d'analyse des données

L'analyse des données comporte deux principales orientations : l'analyse factorielle et analyse par classification.

1.2.1. Analyse factorielle

L'analyse factorielle est une méthode qui cherche à représenter de grands ensembles de données par peu de variables. Les variables ainsi déterminées permettent une représentation synthétique. [2]

Autre mot, en présence d'un tableau de données réelles sous forme de matrices X (n,p) de n lignes et p colonnes.

Les n lignes de X représentent un nuage de n points dans un Espace Vectoriel R^p de dimensionp. (Respectivement, les p colonnes représentent p points dans l'E.V.R^n).

Une représentation graphique de ces points dans cet espace est bien entendu impossible lorsque p>2.

L'analyse factorielle inclut plusieurs méthodes dépendent de types des tableaux [13].

1.2.1.1.Analyse en composante principales ACP

L'ACP est l'une des méthodes les plus employées. Elle est particulièrement adaptée aux variables quantitatives, continues, a priori corrélées entre elles.[4] Elle est utilisée pour réduire p variables corrélées en un nombre q de variables non corrélées de telles manières que les q variables soient des combinaisons linéaires des p variables initiales, que leur variance soit maximale et que les nouvelles variables soient orthogonales entre elles suivant une distance particulière.

Les composantes, les nouvelles variables, définissent un sous-espace à q dimension sur lequel sont projetés les individus avec un minimum de perte

d'information. Dans cet espace le nuage de points est plus facile à représenté et l'analyse est plus aisée [2].

L'ACP se fait :

- Soit pour l'étude d'une population donnée en cherchant à déterminer la typologie des individus et des variables. Par exemple la biométrie.
- Soit pour réduire les dimensions des données sans perte importante d'information. Par exemple en traitement du signal et des images.

1.2.1.2. Analyse factorielle des correspondances AFC

L'AFC a été conçue pour l'étude des tableaux de contingence obtenus par croisement de variables qualitatives. Cette analyse peut être présentée sous de nombreux points de vues, notamment comme un cas particuliers de l'analyse canoniques ou encore de l'analyse factorielle discriminante. Elle peut aussi être étudiée comme un ACP avec une métrique spéciale (celle de khi-deux).

L'analyse factorielle des correspondances vises à ressembler en un nombre réduit de dimensions la plus grande partie de l'information initial en s'attachant non pas eux valeurs absolus mais aux correspondances entre les variables, c'est-à-dire aux valeurs relatives l'AFC offre la particularité de fournir un espace de représentation commun aux variables et aux individus. Pour cela l'AFC raisonne à partir de tableau réduit ou de fréquences. L'AFC peut être appliqué aux tableaux de mesures homogènes (même système d'unités), aux tableaux de notes, de rangs, de préférences [2].

1.2.1.3. Analyse factorielle des correspondances multiples AFCM

L'AFCM est considérée comme l'application la plus féconde de l'analyse des correspondances. Permet l'étude de plusieurs variables qualitatives, et aussi quantitatives après construction de classes.

Formellement, une AFCM est une AFC appliquée sur le tableau disjonctif complet, ou bien une AFC appliquée sur le tableau de Burt, ces deux tableaux étant issus du tableau initial. Un tableau disjonctif complet est un tableau où les variables sont remplacées par leurs modalités et les éléments par 1 si la modalité est remplie par 0 sinon pour chaque individu. Un tableau de Burt est le tableau de contingence des p variables prises deux à deux [2].

L'AFCM est donc très bien adaptée au traitement d'enquêtes socio-économiques, et les tableaux logiques.

1.2.1.4. Analyse factorielle discriminante AFD

L'AFD est une méthode descriptive et prédictive fondée sur un modèle paramétrique. Elle est généralement appelée analyse linéaire discriminante (linéaire Analysis Discrinant (LDA) en anglais)

Cette technique projette des classes prédéfinis sur des plans factoriels discriminant le plus possible. Le tableau de données décrit n individus sur lesquels p variables quantitatives et une variable qualitative à q modalités ont été mesurées. La variable qualitative permet de définir les q classes et le regroupement des individus dans ces classes. L'AFD se propose de trouver q-1 variables, appelées variables discriminantes, dont les axes séparent le plus projections des q classes qui découpent le nuage de points.

L'AFD est une approche très utilisée dans les nombreux domaines tels qu'en médecine pour prédiction de maladies, en météorologie pour prédire un risque d'avalanche ou en finance pour prédire un comportement boursier, pour la reconnaissance des formes, ou encore pour le contrôle de qualité [4].

1.2.1.5. Analyse canonique

L'analyse canonique permet de comparer deux groupes de variables quantitatives appliqués tous deux sur les mêmes individus. Le but de l'analyse canonique est de comparer ces deux groupes de variables pour savoir s'ils décrivent un phénomènes, auquel cas l'analyste pourra se passer d'un des deux groupes de variables.

Plus formellement, si X_1 et X_2 sont deux groupes de variables, l'analyse canonique cherche des couples de vecteurs (ξ_{1i},η_{2i})

Combinaisons possibles. Ces variables sont dénommées variables canoniques.

L'analyse canonique est effectuée surtout dans le domaine médical (analyser les mêmes échantillons par deux laboratoires différents) [4].

1.2.2. Analyse par classification

La classification est une partition où chaque élément est affecté à une classe donnée selon la méthode de classification qui peut être directe en un nombre fixé de classes ou sous la forme d'une hiérarchie emboîtée à plusieurs niveaux d'agrégation. Le modèle général s'appuie sur la distance entre un individu et un autre individu ou groupe. Plus cette distance est réduite, plus les deux entités

sont proches et la classification se fait sur cette base quelque soit la méthode utilisée dans la détermination des classes, le critère de regroupement ou la nature de la distance utilisée.

Très souvent, la méthode consiste à calculer une matrice de distances ou de similarités entre les individus qui sont souvent des espaces en géographie en fonction des données correspondantes aux différentes variables ou aux divers scores factoriels considérés. On obtient ainsi une matrice des Distances (D) ou des Similarités (S) qui nous permet d'agréger les individus en classes selon un schéma hiérarchique ou non [20].

1.3.Les tableaux analysables

Les données se représentent généralement sous la forme d'un tableau rectangulaire, dont les lignes correspondent à des individus ou unités statiques et les colonnes à des variables appelées caractères ou caractéristiques.

1.1.1. Les types de variables

Une variable est une caractéristique pouvant prendre plusieurs des valeurs d'un ensemble d'observations possibles, auquel une mesure ou une qualité peut être appliqué. Par exemple, l'âge, le poids, la couleur, etc.

La distinction entre les types de variables décide des analyses graphiques utilisables ou encore des tests statistiques.

Il existe deux types de variables :

1.1.1.1. Les variables qualitatives

Comme le nom indique, les variables qualitatives contiennent des valeurs qui expriment une qualité. Toutes les variables qualitatives sont discontinues. C'est parce qu'elles décrivent des attributs par leurs natures, sont discontinus. Comme le sexe, la couleur, l'état civil, etc. il y a deux types de variables différentes [9]:

- **Les variables qualitatives nominales**

  Une variable est qualitatives nominales quand ses valeurs sont des éléments d'une catégorie non hiérarchique (les modalités ne peuvent pas être ordonnées) c'est-à-dire ses éléments ne peuvent pas se ranger dans une gradation logique. Exemple :

  Le sexe -femme profession -médecin

  -homme -ingénieur

- **Les variables qualitatives ordinales**

  Une variable est qualitative ordinale quand ses valeurs sont des éléments d'une catégorie hiérarchique (les modalités peuvent être ordonnées) c'est-à-dire que ses éléments peuvent être rangés dans une gradation logique. Exemple :

  - Niveau de scolarité : primaire, secondaire, universitaire ;
  - Jugement et appréciation : jamais, rarement, à l'occasion, fréquemment, toujours ;

1.1.1.2. Les variables quantitatives

Une variable est dite quantitatives si toute ses valeurs possible sont numériques (représentées par des quantités). Au même pour les variables quantitatives il existe deux types sont [8] :

- **Variables quantitatives discrètes**

  Une variable est dite discrète, si l'ensemble des valeurs possible est dénombrable. Il est inutile d'utiliser des classes pour exprimer. Exemple :

  - Le nombre de présence au centre commercial par mois
  - Le nombre d'enfant moyens dans chaque famille

- **Variables quantitatives continus**

  Une variable est dite continue, si l'ensemble des valeurs possibles est continu. Il est donc préférable de les exprimer en classe de largeur égale. Exemple :

  - Si on mesure la superficie
  - Si on mesure la température

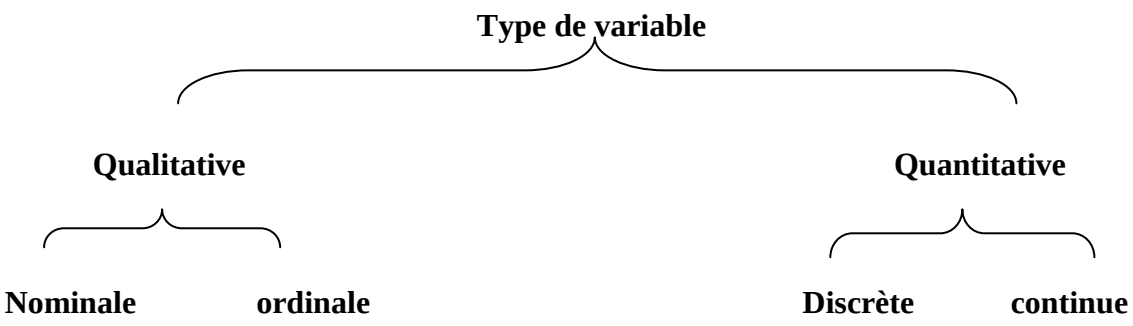

Figure 1.1 : Les types de données

1.2.L'objectif d'analyse des données

L'objectif de l'analyse des données est répondre aux problèmes posés par des tableaux de grandes dimensions. Les objectifs sont souvent présentés en fonction du type de méthodes, ainsi deux objectifs ressortent : la visualisation des données dans meilleur espace réduit et le regroupement dans tout l'espace.

Les méthodes ne se limitent pas à une représentation des données, ou du moins pour la rendre plus aisée, elles recherchent les ressemblances entre les individus et les liaisons entre les variables. Ces proximités entre individus et variables vont permettre à l'opérateur de déterminer une typologie des individus et des variables, et ainsi il pourra interpréter ses données et fournir une synthèse des résultats des analyses. Nous voyons donc que les deux objectifs précédemment cités sont très liés voir indissociables, ce qui entraine souvent l'utilisation conjointe de plusieurs méthodes d'analyse de données [7] .

1.3.Les domaines d'application

Aujourd'hui l'analyse des données est employée dans un grand nombre de domaines qu'il est impossible d'énumérer. Actuellement ces méthodes sont beaucoup utilisées en science humaine, cette technique est utilisée pour cerner les résultats des enquêtes d'opinion (l'interprétation des sondages).

La sociologie compte beaucoup sur l'analyse de données pour comprendre la vie et de son amélioration.

En économie pour décrire la structure et la taille de ces organismes, en marketing (la gestion de clientèle) et les domaines industriels (assurance, banque, etc.).

Aussi, en traitement du signal et des images, ou elles sont souvent employées comme prétraitements (qui peuvent être vus comme des filtres) [7].

2. La classification

2.1.Un peu d'histoire

En 1813 Augustin Pyramus de Candolle a utilisé pour désigner la science des lois de la classification des formes vivantes selon les critères de regroupement : taille, forme des feuilles, racines, etc. sous le nom Taxinomie (grec : ordre, arrangement et loi).

Linné (en science naturelles), et Koppen (classification des climats) en 1911.

Et pour la première fois en 1939 l'utilisation de terme classification avec ces différents algorithmes par Tryon.

Robert R. Sokal et Peter H.A. Sneath présentent en 1963 des méthodes quantitatives appliquées à la taxinomie [12] .

2.2.Définition

Classifier c'est regrouper entre eux des objets similaires selon certain critères par les diverses techniques de classification visent toutes à répartir n individus caractérisés par p variablesX_1,X_2, …, X_p en un certain nombre m de sous groupes aussi homogènes que possible. Selon la méthode de classification qui peut être directe en un nombre fixé de classes ou sous la forme d'une hiérarchie à plusieurs niveaux d'agrégation. Le modèle général s'appuie sur la distance entre un individu et un autre. Plus cette distance est réduite, plus les deux entités sont proches et la classification se fait sur cette base quelque soit la méthode utilisée, ce critère de regroupement ou la nature de distance utilisée.

2.3.Formalisation mathématique de problème de classification

En terme mathématique, un problème de classification comporte les ingrédients suivants :

- Une population de N individu I^i (i variant de 1 à N)
- P variables descriptives X_d^i qui permettent de décrire les individus ; elles sont aussi appelées plus simplement descripteurs (d variant de 1 à P)
- C classes C_k dans lesquelles on cherche à ranger les individus (k variant de1 à C)

Résoudre un problème de classification, c'est trouver une application de l'ensemble des objets à classer, décrits par les variables descriptives choisies, dans l'ensemble des classes [3].

L'algorithme ou la procédure qui réalise cette application est appelé classifieur.

2.4. Préparation des données en vue d'une classification

Des variables sans rapport avec le problème posé peuvent entrainer une classification futile car elles ne peuvent qu'affecter négativement les mesures de proximité et éliminer la tendance à la structuration en classes

Un analyse exploratoire préliminaire est donc essentielle pour éliminer ces variables inappropriées, réduire la cardinalité des variables catégorielle, mettre en évidence la présence d'oublier et homogénéiser, si nécessaire, les variables

hétérogènes à prendre en compte simultanément dans une méthode de classification.

D'autre part, une standardisation de variables numériques retenues pour la classification permet de donner le même poids à toutes ces variables dans l'analyse [3].

2.5. L'objectif de la classification

La classification a pris aujourd'hui une place importante en analyse des données exploratoire et décisionnelle, l'objectif exploratoire vise à découvrir une partition hypothétique dans un ensemble d'objets. Dans l'analyse décisionnelle, on cherche généralement à affecter tout nouvel objet à des groupes préalablement définis.

La classification a pour but plus simple est répartir l'échantillon en groupes d'observation homogènes, chaque groupe étant bien différencié des autres.

On veut en général obtenir des sections à l'intérieur des groupes principaux, puis des subdivisions plus petites de ces sections, et ainsi de suite. En bref, on désire avoir une hiérarchie de plus en plus fine, sur l'ensemble d'observations initial[7].

2.6. Les domaines d'application

La classification a un rôle à jouer dans toutes les sciences et les techniques qui font appel à la statistique multidimensionnelle.

- Le domaine médical : les classifications des maladies, de traitements curatifs de ces maladies ou de symptômes pour ces maladies peuvent mener à des typologies très utiles.
- Le domaine du marketing : pour le lancement d'un nouveau produit, le responsable du service de marketing d'une entreprise pourra chercher à constituer des groupes de villes semblables vis-à-vis d'un certain nombre critères, et choisir dans chaque groupe une ville type utilisée comme marché-test.
- Le domaine politique : un candidat aux élections municipales fixera sa stratégie électorale en fonction de différents types d'électeurs construits à partir de différentes caractéristiques.
- En psychiatrie : un diagnostic correct de groupes de symptômes comme paranoïa, la schizophrénie, etc. est indispensable à la réussite de la thérapie.
- En archéologie les chercheurs ont tenté d'établir des typologies d'outils en pierre, objets funéraire, etc. En utilisant des techniques de classification.

- Les sciences biologiques : botanique, zoologie, écologie, etc.
- Les sciences de terre et des eaux : géologie, pédologie, géographie, études des pollutions.
- Les techniques dérivées : les enquêtes d'opinions, etc.
- Les sciences économiques [3].

2.7. Les termes désignant la classification

Plusieurs termes sont utilisés dans la littérature pour désigner une technique de classification parmi lesquels : classification automatique, apprentissage non supervisé (dans le domaine de la reconnaissance des formes), analyse typologique, taxinomie ou taxonomie numérique (en biologie et zoologie), nosologie(en médecine), partition dans la théorie des graphes.

Le terme anglais pour désigner une technique de classification est clustering (non supervised) classification [3].

2.8. Les méthodes de classification

Il existe un grand nombre de méthodes et surtout beaucoup de variantes. Il est d'objectif de les différentier grossièrement soit par leur structure de classification, soit par le type de représentation des classes.

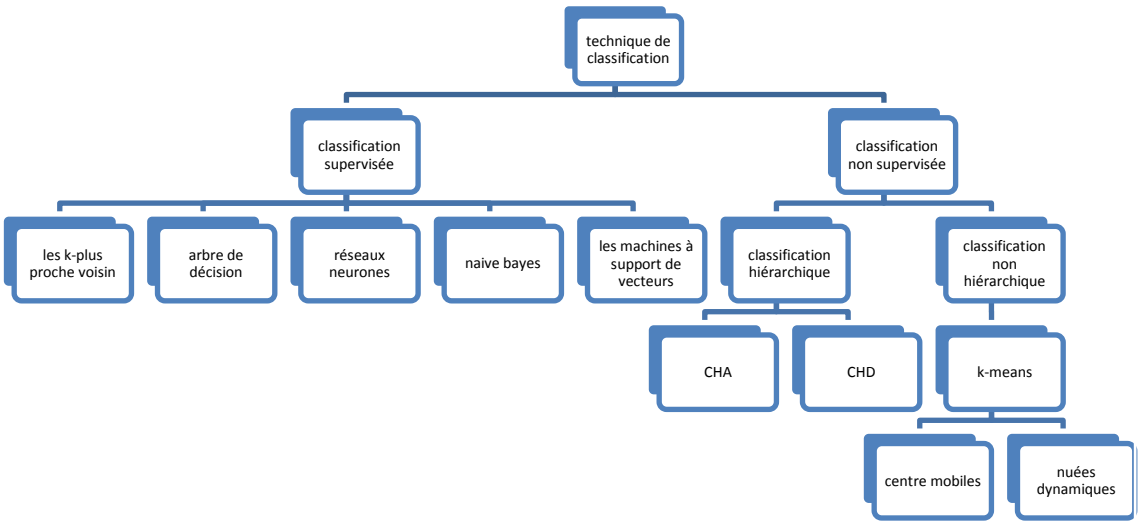

Figure1.22 : les méthodes de classification

Ainsi, nous distinguer selon les critères deux types :

2.8.1. La classification supervisée

Dans le contexte supervisé on dispose déjà d'exemples dont la classe est connue et étiquetée. Les données sont donc associées à des labels des classes notés Q = { q_1 , q_2 ,….., q_n }. L'objectif est alors d'apprendre à l'aide d'un modèle d'apprentissage des règles qui permettent de prédire la classe des nouvelles observations ce qui revient à déterminer une fonction Cl qui à partir des descripteurs (D) de l'objet associe une classe qi et de pouvoir aussi affecter toute nouvelle observation à une classe parmi les classes disponibles. Ceci revient à la fin à trouver une fonction qu'on note Ys qui associe chaque élément de X un élément de Q. On construit alors un modèle en vue de classer les nouvelles données. Parmi les méthodes supervisées on cite : les k-plus proches voisins, les arbres de décision, les réseaux de neurones, les machines à support de vecteurs(SVM) et les classificateurs de Bayes.

Quelque soit le type de la classification, on est confronté à problèmes. Dans le cas supervisé, un problème important peut être le manque de données pour réaliser l'apprentissage ou la disponibilité de données inadéquates par exemple incertaines et imprécises ce qui empêche la construction d'un modèle correct [11].

2.8.1.1. Les k-plus proches voisins

La méthode des k-plus proches voisins (noté K-PPV ou K-NN pour K-Nearst-Neighbors en anglais) consiste à déterminer pour chaque nouvel individu que l'on veut classer, la liste des k plus proches voisins parmi les individus déjà classé. L'individu est affecté à la classe qui contient le plus d'individus parmi ces k plus proches voisins. Cette méthode nécessite de choisir une distances (la plus classique est la distance Euclidienne), et donc le nombre k de voisins à prendre en compte.

Cette méthode supervisée et non-paramétrique est souvent performante. De plus son apprentissage est assez simple [26].

2.8.1.2. Arbre de décision

Les arbres de décision sont les plus populaires des méthodes d'apprentissage. L'apprentissage se fait par partitionnement récursif selon des règles sur les variables explicatives suivant les critères de partitionnement et les données, on

dispose de différentes méthodes, dont CART, CHAID….. Ces méthodes peuvent s'appliquer à une variable. Deux types d'arbres de décisions sont ainsi définis [25] :

- ✓ **Arbre de classification** la variable expliquée est de type nominal. A chaque étape du partitionnement, on cherche à réduire l'impureté totale des deux nœuds fils par rapport au nœud père.
- ✓ **Arbre de régression** la variable expliquée est de type numérique et il s'agit de prédire une valeur la plus proche possible de la vraie valeur.

Construire un tel arbre consiste à définir un nœud, chaque nœud permettant de faire une partition des objets en 2 groupes sur la base d'une des variables explicatives. Il convient donc :

- → définir un critère permettant de sélectionner le meilleur nœud possible à une étape donnée ;
- → De définir quand s'arrête le découpage, en définissant un nœud terminal (feuille) ;
- → D'attribuer au nœud terminal la classe ou la valeur la plus probable
- → D'élaguer l'arbre quand le nombre de nœuds devient trop important en sélectionnant un sous arbre optimal à partir de l'arbre maximal ;
- → Valider l'arbre à partir d'une validation croisée ou d'autres techniques.

2.8.1.3. Réseaux neurones

Les réseaux de neurones sont des approximateurs universels parcimonieux ; ils peuvent donc être utilisés pour modéliser ou commander tout processus, statique ou dynamique, non linéaire : en raison de leur parcimonie, ils sont avantageux par rapport aux autres approximateurs et notamment au flou - dès que le processus à modéliser ou à commander possède plus de deux ou trois entrées. Néanmoins, comme toute autre technique, les réseaux de neurones sont soumis à des contraintes : étant des outils statistiques, ils traitent uniquement de données *numériques*, dont le nombre et la représentativité doivent être convenables même si, leur parcimonie leur permet d'utiliser moins de données que d'autres méthodes statistiques. S'il est possible de tirer profit, pour la conception du réseau, des connaissances, même imprécises,

que l'on peut avoir sur le processus, il faut qu'elles soient sous forme *mathématique* : les réseaux de neurones ne permettent pas de traiter aisément des données linguistiques [15].

2.8.1.4. Naïve bayes

Nommées d'après le théorème de Bayes, ces méthodes sont qualifiées de "naïve" ou "simple" car elles supposent l'indépendance des variables. L'idée est d'utiliser des conditions de probabilité observées dans les données.

On calcule la probabilité de chaque classe [24].

2.8.1.5. Les machines à support de vecteurs

Cette technique - initiée par Vapnik - tente de séparer linéairement les exemples positifs des exemples négatifs dans l'ensemble des exemples. Chaque exemple doit être représenté par un vecteur de dimension n. La méthode cherche alors l'hyperplan qui sépare les exemples positifs des exemples négatifs, en garantissant que la marge entre le plus proche des positifs et des négatifs soit maximale. Intuitivement, cela garantit un bon niveau de généralisation car de nouveaux exemples pourront ne pas être trop similaires à ceux utilisés pour trouver l'hyperplan mais être tout de même situés franchement d'un côté ou l'autre de la frontière. L'efficacité des SVM est supérieure à celle de toutes les autres méthodes sur la classification de textes. Son efficacité est aussi très bonne pour la reconnaissance de formes. Un autre intérêt est la sélection de Vecteurs Supports qui représentent les vecteurs discriminant grâce auxquels est déterminé l'hyperplan. Les exemples utilisés lors de la recherche de l'hyperplan ne sont alors plus utiles et seuls ces vecteurs supports sont utilisés pour classer un nouveau cas. Cela en fait une méthode très rapide [26].

2.8.2. La classification non supervisée

Cette classification est aussi appelée "classification automatique", "clustering" ou encore "regroupement". Dans ce type de classification on est amené à identifier les populations d'un ensemble de données. On suppose qu'on dispose d'un ensemble d'objets que l'on note par X = $\{x_1, x_{2,\ldots\ldots,} x_N\}$caractérisé par un ensemble de descripteurs D, l'objectif du clustering est de trouver les groupes auxquels appartiennent chaque objet x qu'on note par C = $\{C_1, C_2, \ldots\ldots C_n\}$Ce qui revient à déterminer une fonction notée Y_s qui associe à chaque élément de X

un ou plusieurs éléments de C. Il faut pouvoir affecter une nouvelle observation à une classe. Les disponibles ne sont pas initialement identifiées comme appartenant à telle ou telle population. L'absence d'étiquette de classe est un lourd handicap qui n'est que très partiellement surmontable. Seule l'analyse de la répartition spatiale des observations peut permettre de "deviner" où sont les véritables classes.

Parmi les méthodes non-supervisées les plus utilisées, citons deux types d'approches : les centres mobiles (k-means) et la classification hiérarchique [11].

2.8.2.1. Classification non hiérarchique

Classification non hiérarchique ou partitionnement, aboutissant à la décomposition de l'ensemble de tous les individus en m ensemble disjoints ou classes d'équivalence ; le nombre m de classes est fixé.

Le résultat obtenu est alors une partition de l'ensemble des individus, un ensemble de parties, ou classes de l'ensemble I des individus telles que :

- Toute classe soit non vide ;
- Deux classes distinctes sont disjointes ;
- Tout individu appartient à une classe.

Cet algorithme porte le nom de "agrégation autour de centres variables". Une version légèrement différente, connue sous le nom de "nuées dynamiques" consiste à représenter chaque groupe non pas par son centre, mais par un ensemble de points (noyau) choisis aléatoirement à l'intérieur de chaque groupe. On calcule alors une distance "moyenne" entre chaque observation et ces noyaux et l'on procède à l'affectation [23].

2.8.1.2.1. Méthode de k-means

C'est une méthode dont le but est de diviser des observations en k partitions dans lesquelles chaque observation appartient à la partition avec la moyenne la plus proche.

Nous citons deux méthodes connues sur le principe de k-means sont :

→ Méthodes de centres mobiles ;

→ Méthodes des nuées dynamiques.

- **Méthode de entres mobiles**

  Cette méthode consiste à construire une partition en k classes en sélectionnant k individus commence, des classes tirés au hasard de l'ensemble d'individus. Après cette sélection, on affecte chaque individu au centre le plus proche en créant k classes, les centres des classes seront remplacer par les centres de gravité et nouveaux classes seront crées par le même principe.

  Généralement la partition obtenue est localement optimale car elle dépend du choix initial des centres. Pour cela les résultats entre deux exécutions de l'algorithme sont significativement variés.

- **Méthode de nuées dynamiques**

  Dans ce cas, le problème posé est la recherche d'une partition en k (k fixé) classes d'une ensemble de n individus. C'est un algorithme itératif.

  Soit *I* une population d'individus, cette population est représentable sur R et forme un nuage de n points.

  On cherche à constituer une partition en k classes sur i. chaque classe est représentée par son centre, également appelé noyau, constitué du petit sous-ensemble de la classe qui minimise le critère de dissemblance.

2.8.2.2. La classification hiérarchique

La classification hiérarchique : pou un niveau de précision donné, deux individus peuvent être confondus dans un même groupe, alors qu'a un niveau de précision plus élevé, ils seront distingués et appartiendront à deux sous-groupes différents. Le résultat d'une classification hiérarchique n'est pas une partition de l'ensemble des individus. C'est une hiérarchie de classes telle que :

- Toute classe est non vide.
- Tout individu appartient à une (et même plusieurs) classes.
- Deux classes distinctes sont disjointes, ou vérifient une relation d'inclusion (l'une d'elle est incluse dans l'autre)
- Toute classe est la réunion des classes qui sont incluse dans elle.

L'avantage de cette méthode est qu'elle n'est soumise à aucune initialisation particulière de paramètre(s) ce qui la rend déterministe, et en outre, que le nombre de classe n'a pas à être fixé a priori. Cependant, ce type de méthode impose le calcul de la matrice des distances de tous les points d'observation avec tous les autres, et cette masse de calculs est beaucoup trop importante compte tenu du temps que nous voulons consacrer à cette étape [23].

Parmi les méthodes non-supervisées les plus utilisées, citons deux types d'approches :

2.8.2.2.1. Classification hiérarchique ascendante

La CAH permet de construire une hiérarchie entière des objets sous la forme d'un "arbre" dans un ordre ascendant. On commence en considérant chaque individu comme une classe et on essaye de fusionner deux ou plusieurs classes appropriées (selon une similarité) pour former une nouvelle classe. Le processus est itéré jusqu'à ce que tous les individus se trouvent dans une même classe. Cette classification génère un arbre que l'on peut couper à différents niveaux pour obtenir un nombre des classes plus ou moins grand.

Différentes mesures de la distance interclasses peuvent être utilisées : la distance euclidienne, la distance inferieur (qui favorise la création de classes de faible inertie) ou la distance supérieure (qui favorise la création de classes d'inertie plus importante) etc.

le cas de la classification ascendante hiérarchique, à partir des éléments, on forme des petites classes ne comprenant que des individus très semblables, puis à partir de celle-ci, on construit des classes de moins en moins homogènes, jusqu'à obtenir la classe tout entière [4].

2.8.2.2.2. Classification hiérarchique descendante

Dans la CDH, on considérant tous les individus comme une seule classe au début, on divise successivement les classes en classes plus raffinées. Le processus marche jusqu'à ce que chaque classe contienne un seul point ou bien si l'on atteint un nombre de classes désiré [4].

2.9.La qualité d'une classification

Une bonne classification produire des classes avec une grande similarité à l'intérieur de chaque classe et une petite similarité entre les différentes classes. Sa

qualité dépend à la fois de la mesure de similarité utilisée par la méthode et de son implémentation.

Elle est aussi mesurée par sa capacité à mettre à jour quelques unes ou toutes les formes cachées.

Une méthode de classification est valable si elle permet de :

- Prendre en compte simultanément des variables de nature différente
- Utiliser un nombre minimum de paramètres à fixer au départ de l'algorithme
- S'appliquer à des données de taille importante et hautement multidimensionnelles sans nécessité un temps ordinateur prohibitif, c'est-à-dire de complexité raisonnable.
- Découvrir des classes de forme quelconque (convexe ou non) et de détecter du bruit ou des points isolés
- Ne pas être sensible à l'ordre des observations dans le fichier à analyser

Conduire à une interprétation et une utilisation faciles des résultats [3].

3. Conclusion

Nous avons dans ce chapitre présenté l'analyse des données qui est une famille de méthode statistiques dont les principales caractéristiques sont d'être multidimensionnelles et descriptives. L'analyse des données permet de traiter un nombre très important de données (quantitatives, qualitatives) et de dégager les aspects les plus intéressant. Elle comprend l'analyse en composantes principales (ACP), employée pour des données quantitatives, et ses méthodes dérivées ; l'analyse factorielle correspondances (AFC) utilisée sur des données qualitatives et l'analyse factorielle des correspondances multiples (AFCM) généralisant la précédence et l'analyse canonique.

Toutes ces méthodes sont réduites sous la méthode factorielle.

Ensuite les méthodes de classification ont de but de diviser un ensemble des données en plusieurs classes homogènes. Avec différentes méthodes : supervisée et non supervisée : hiérarchie (comme CAH et CDH) et non hiérarchie (méthode de k-mean).

Nous allons étudier ce type de classification hiérarchique ascendante dans le chapitre suivant.

Chapitre2 :

Classification ascendante hiérarchique

1. Introduction

La classification hiérarchique ascendante permet de construire une hiérarchie entière des objets dans un ordre ascendant. On commençant par considérer chaque individu comme une classe et on essaye de fusionner deux ou plusieurs classes appropriées (selon la similarité) pour former une nouvelle classe. Le processus est itéré jusqu'à ce que tous les individus se trouvent dans une même classe. Cette classification génère un arbre que l'on peut couper à différents niveaux pour obtenir un nombre des classes plus ou moins grand [5].

2. Similarité

La similarité définie sur un ensemble E est une application du produit cartésien E×E dans $\mathbb{R}^+$satisfaisant aux axiomes suivant :

$S(X_i,X_{i'}) = S(X_{i'}, X_i) \geq 0 \qquad \forall X_i \in E, \forall X_{i'} \in E.$

$S(X_i,X_i) = S(X_{i'}, X_{i'}) = S_{max} > S(X_i,X_{i'}) \qquad \forall X_i \in E, \forall X_{i'} \in E.$

Ou S_{max} est la plus grande similarité possible[3].

3. La dissimilarité

Définition : On appelle indice de dissimilarité sur M toute application d définit comme suit :

$$d : M \times M \rightarrow \mathbb{R}^+$$

et vérifiant la propriété suivante :

$$\forall (x, y) \in M^2 \; d(x, y) = 0 \Longleftrightarrow x = y$$

Cela signifie que deux points ont une dissimilarité égale à zéro si et seulement si ils sont confondus [6].

3.1. Distance définie sur un ensemble E

C'est une application du produit cartésien E×E dans $\mathbb{R}^+$ satisfaisant aux axiomes suivants :

Symétrie $d(X_i,X_{i'}) = d(X_{i'}, X_i), \forall X_i \in E, \forall X_{i'} \in E.$

Positivité stricte $d(X_i,X_{i'}) > 0$ si $X_i \neq X_{i'}$ et $d(X_i,X_{i'}) = 0 \Longleftrightarrow$ si $X_i = X_{i'}$,

$\forall X_i \in E, \forall X_{i'} \in E.$

Inégalité triangulaire $d(X_i,X_{i'}) \leq d(X_i, X_{i''}) + d(X_{i''}, X_{i'}), \forall X_i \in E, \forall X_{i'} \in E$ et, $\forall X_{i''} \in E.$ [4]

3.1.1. Les mesures de distance [19]

La classification ascendante hiérarchique CAH utilise des mesures de dissemblance ou de distance entre les objets pour former des classes. Ces distances peuvent être basées sur une ou plusieurs dimensions. La méthode la plus directe pour calculer des distances entre objets dans un espace multidimensionnel consiste à calculer les distances euclidiennes. Si nous avons un espace à deux ou trois dimensions, cette mesure est celle des distances géométrique normales entre les objets dans l'espace.

La classification permet de calculer de nombreux types de mesures de distances, afin de l'utiliser directement dans la procédure.

- **Distance Euclidienne**

  C'est probablement le type de distance le plus couramment utilisé. Il s'agit simplement d'une distance géométrique dans un espace multidimensionnel. Elle se calcule ainsi :

$$\text{Distance(X, Y)} = \left(\sum_i (X_i - Y_i)^2\right)^{1/2}$$

- **Distance Euclidienne au carré**

  Vous pouvez élever la distance Euclidienne standard au carré afin de surpondérer les objets atypiques (éloignés). Cette distance se calcule ainsi :

$$\text{Distance (X, Y)} = \sum_i (X_i - Y_i)^2$$

Remarque

Les distances Euclidiennes (et Euclidiennes au carré) sont calculées à partir des données brutes, et non des données centrées-réduites. C'est la méthode de calcul qui est habituellement utilisée, et elle présente certains avantages (en particulier, la distance entre deux objets quelconques n'est pas affectée par l'introduction de nouveaux objets dans l'analyse, qui peuvent être des points atypiques). Toutefois, les distances peuvent être largement affectées par les différences d'unités de mesure des dimensions pour lesquelles ces distances sont calculées. Ainsi, si l'une des dimensions représente une taille en centimètres, que vous décidez de convertir en millimètres (en multipliant les valeurs par 10), les distances Euclidiennes ou distances Euclidiennes au carré résultantes (calculées sur de multiples dimensions) pourront s'en trouver largement affectées, et par conséquent, les résultats de la

classification pourront être très différents. Naturellement, vous avez la possibilité d'effectuer tout type de standardisation ou de changement d'échelle en utilisant les fonctionnalités de gestion des données de STATISTICA

- **Distance du city-block (Manhattan)**

  Cette distance est simplement la somme des différences entre les dimensions. Dans la plupart des cas, cette mesure de distance produit des résultats proches de ceux obtenus par la distance euclidienne simple. En revanche, notez qu'avec cette mesure, l'effet des différences simples importantes (points atypiques) est atténué (puisque ces distances ne sont pas élevées au carré). Cette distance se calcule ainsi :

  $$\text{Distance (X, Y)} = \sum_i |X_i - Y_i|$$

- **Distance de Tchebychev**

  Cette mesure de distance est adaptée lorsque nous considérons deux objets comme étant différents à partir du moment où ils sont différents sur l'une des dimensions. La distance de Tchebychev se calcul ainsi :

  $$\text{Distance (X, Y)} = \text{Maximum } |X_i - Y_i|$$

- **Distance de puissance**

  Nous pouvons parfois souhaiter augmenter ou diminuer la pondération progressive associée à des dimensions pour lesquelles les objets respectifs sont très différents. Cette opération est rendue possible par la distance à la puissance. La distance à la puissance se calcule ainsi :

  $$\text{Distance(X, Y)} = \left(\sum_i |X_i - Y_i|^P\right)^{\frac{1}{r}}$$

  Où r et p sont des paramètres définis par l'utilisateur. Le paramètre p contrôle la pondération progressive affectée aux différences entre les dimensions individuelles, tandis que le paramètre contrôle la pondération progressive affectée aux grandes différences entre les objets. Si r et p sont égaux à 2, cette distance équivaut à la distance euclidienne.

- **Percent disagreement**

Cette mesure est particulièrement utile si les données des dimensions utilisées dans l'analyse sont de nature catégorielle. Cette distance se calcule ainsi :

Distance(X, Y)= (Nombre de$X_i \neq Y_i$)/i

3.2. La dissimilarité définie sur un ensemble E

Elle est définie par à partir de l'indice de similarité : dis (X_i ,$X_{i'}$)= 1- S ($X_{i'}$, X_i) [3].

Les critères d'agrégation [19]

De nombreux critères d'agrégation ont été proposés les plus connus sont :

- **Le critère du saut minimal**

La distance entre 2 classes C_1 et C_2 est définie par la plus courte distance séparant un individu de C_1 et un individu deC_2.

$$D(C_1, C_2) = min\ (\{d(x, y)\}, x \in C_1\ , y \in C_2)$$

- **Le critère du saut maximal**

La distance entre 2 classes $C_1 et\ C_2$ est définie par la plus grande distance séparant un individu de C_1 et un individu de C_2

$$D\ (C_1\ , C_2) = max\ (\{d(x, y)\}, x \in C_1, y \in C_2)$$

- **Le critère de la moyenne**

Ce critère consiste à calculer la distance moyenne entre tous élément de C_1 et tous les éléments deC_2.

$$D\ (C_1\ , C_2) = \frac{1}{nC_1\ n\ C_2} \sum_{x \in C_1} \sum_{y \in C_2} d(x, y)$$

Avec :

nC_1 : Le cardinal deC_1

nC_2 : Le cardinal de C_2

- **Le critère de Ward**

Ce critère ne s'applique que si on est muni 'un espace euclidien. La dissimilarité entre 2 individus doit être égale à la moitié du carré de la distance euclidienne d, le critère de Ward consiste à choisir à chaque étape le regroupement de classes tel que l'augmentation de l'inertie intra-classe soit minimal.

$$D(C_1, C_2) = \frac{nC_1 nC_2}{nC_1 + nC_2} d^2(gC_1, gC_2)$$

Avec :

gC_1 : Centre de gravité de C_1

gC_2 : Centre de gravité de C_2

℘ **Le critère de centre de gravité**

La distance entre 2 classes C_1et C_2 est définie par la distance entre leurs centres de gravité.

$$D(C_1, C_2) = d(gC_1, gC_2)$$

La difficulté du choix du critère d'agrégation réside dans le fait que ces critères peuvent déboucher sur des résultats différents.

Remarque

Dans le cas de stratégie du saut minimal, plus une classe contient d'éléments, plus elle est attractive pour des éléments isolés. Au contraire, dans le cas d'une stratégie du saut maximal, plus une classe contient d'élément, mois elle est attractive pour les éléments isolés.

Le critère de Ward, aisée à mettre en œuvre lorsque la classification est effectuée après une analyse factorielle (les objets à classer étant repérés par leurs coordonnées sur les premiers axes factoriels). Constitue une excellente méthode de classification ascendante hiérarchique. En effet, il est basé sur la réunion des 2 classes qui minimise l'augmentation de l'inertie intra-classes.

4. Le principe générale de la CAH

On suppose que les distances entre tous les objets, deux à deux, ont été calculées suivant l'une des formules (citées dans le sous titre 3.1.1.). On procède alors par étapes successives, chacune d'elles consistant à réunir les deux objets les plus proches. A la fin de chaque étape on recalcule les distances entre le groupe nouvellement créé et le reste des objets. Cela permet de réitérer le processus jusqu'à ce que tous les objets aient été réunis dans un seul groupe. Lorsque cela est achevé on dresse un arbre hiérarchique dont les nœuds représentent les fusions successives, la hauteur de ces nœuds étant égale à la valeur de la distance entre les deux objets, ou groupes, fusionnés. Le niveau des nœuds a donc ainsi une signification concrète ; on dit dans ce cas qu'on obtient une hiérarchie indicé.

4.1. Hiérarchie totale de parties d'un ensemble E [3]

H_E est un sous-ensemble de l'ensemble ρ(E) des parties de E ayant les propriétés suivantes :

- L'ensemble E est un élément de H_E
- (i) ∈ H_E pour tout élément de E
- h∩ h' ∈ {Ø, h, h'} pour h et h' deux parties quelconque de H_E autrement dit h et h' sont soit disjointes, soit incluse l'une dans l'autre.

4.2. Hiérarchie de partie indicée [3]

C'est une hiérarchie de parties H à laquelle est associée une échelle d'indices qui satisfait la propriété suivante :

A tout h élément de E est associé un nombre v(h) ≥ 0 tel que si h ⊂ h' alors v(h) < (h') .

4.3. L'algorithme CAH

L'algorithme de la classification ascendante hiérarchique est très simple. Il est dû à Lance et William (1967) [16].

Initialisation construction du tableau des distances, peu importe la formule utilisée pour le construire car l'algorithme de CAH est indépendant de la métrique utilisé. Ainsi, entre chaque couple de point (*x, y*) de M, nous disposons d'une valeur d(*x, y*). La partition initiale est la plus fine ρ_0 de M.

Regroupement parcourir le tableau de distance pour déterminer le couple d'élément (x^*,y^*) les plus proches :

$$d\,(x^*,y^*) \leq \min_{x,y\epsilon M} \{d(x, y)\}$$

On réunit les deux éléments dans une même classe A=$x^* \cup y^*$ les autres classes restent inchangées. Nous obtenons une nouvelle partition ρ_i moins fie que la précédente.

Tableau des distances la classe A sera vue comme un seul point. Il faut donc calculer les distances qu'il y a entre le point A qui est un ensemble de cardinal supérieur à un, et tous les autres points qui ne sont pas dans A et peuvent être des singleton. Par souci de généralité, nous les notons B.

$$d\,(A, B)\ ;\ B \nsubseteq A$$

Pour cela, on peut utiliser l'un des cinq critères proposés plus haut. Nous disposons alors d'un nouveau tableau des distances ayant une ligne et une colonne de moins que le précédent dont il ne diffère que par ligne et colonne qui correspond au point A.

Condition d'arrêt si nous avons atteint la partition du niveau souhaité, généralement c'est la partition grossière, celle qui ne comporte qu'une seule classe réunissant la totalité des points, alors, c'est terminé. Dans le cas contraire, nous repartons de l'étape regroupement à partir du tableau des distances calculé à la suite du précédent regroupement.

4.4. Dendrogramme (arbre hiérarchique)

Puisque les méthodes hiérarchiques fusionnent les groupes à des degrés décroissants de ressemblance, il est naturel de représenter les résultats de la classification au moyen d'une structure arborescente que l'on appelle dendrogramme

Il peut être intéressant de fournir ici l'algorithme pour la construction d'un dendrogramme qui consiste à ordonner les observations de telle sorte qu'il n'y ait aucun croisement entre les diverses branches du dendrogramme.

1. placer les observations selon un ordre quelconque de gauche à droite
2. s'il ne reste qu'un seul groupe, on termine.
3. prendre les groupes compris entre les groupes qui fusionnent et les déplacer rigidement à la droite de la dernière observation du groupe fusionné situé le plus à droite.
4. retourner à 2 [18].

4.5. Exemple [16]

Nous disposons d'une population Ω composée de cinq points du plan notés $\omega_1, \omega_2, \omega_3, \omega_4, \omega_5$ dont les coordonnées figurent dans le tableau 1.1.

Ω	X_1	X_2
ω_1	2	2
ω_2	7.5	4
ω_3	3	3
ω_4	0.5	5
ω_5	6	4

Tableau 2.1. Tableau initial des données

La représentation graphique de ces points est donnée par la figure 1.2.

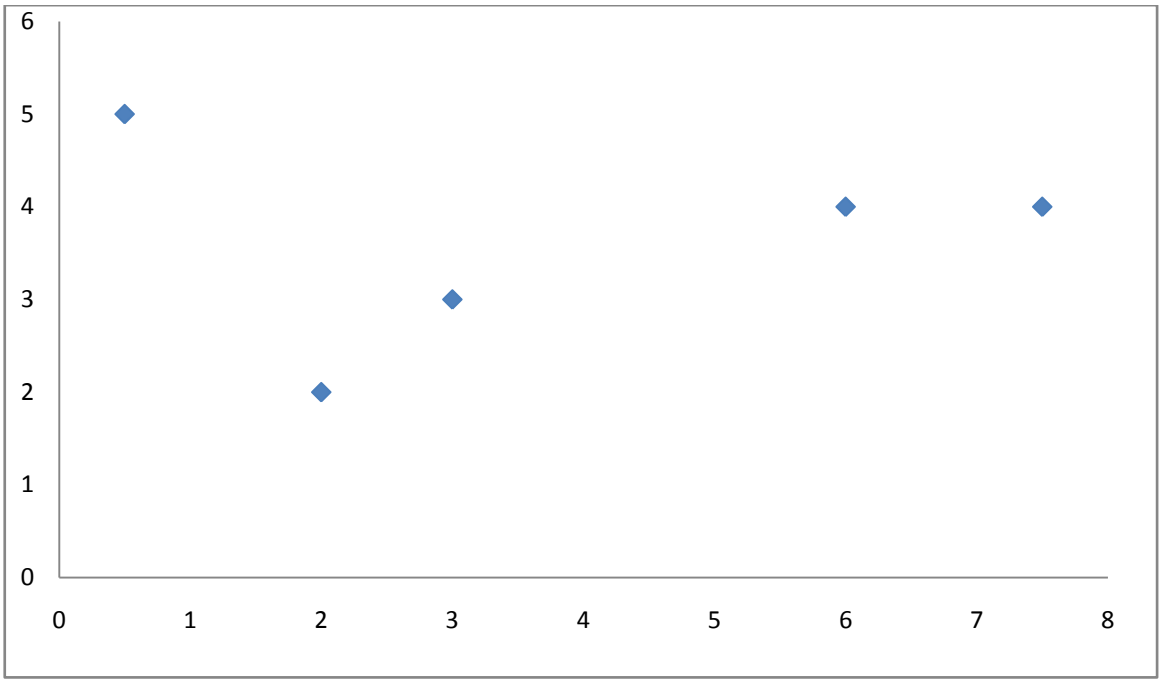

Figure 2.1 : Nuage des points

Chaque point est ici représenté par 2 coordonnées, pour évaluer la dissimilarité entre les points nous utiliserons la distance euclidienne. Autrement dit, si les coordonnées des point ω_i et ω_j sont données par : (X_{i_1},X_{i_2}) et (X_{j_1},X_{j_2}) on a :

$$d\,(\omega_i, \omega_j) = \sqrt{(X_{i_1} - X_{j_1})^2 + (X_{i_2} - X_{j_2})^2}$$

Ainsi, la distance entre les points ω_1 et ω_2 est donnée par :

$$d\,(\omega_1, \omega_2) = \sqrt{(2 - 7.5)^2 + (2 - 4)^2}$$

$d(\omega_i \omega_j)$	ω_1	ω_2	ω_3	ω_4	ω_5
ω_1	0	5.85	1.41	3.35	4.47
ω_2	5.85	0	4.61	7.07	1.5
ω_3	1.41	4.61	0	3.20	3.16
ω_4	3.35	7.07	3.20	0	5.34
ω_5	4.47	1.5	3.16	5.34	0

Tableau2.2. Tableau des distances

On notera $\rho_0 = (\{\omega_1\}, \{\omega_2\}, \{\omega_3\}, \{\omega_4\}, \{\omega_5\})$ la partition la plus fine sur Ω. Chaque classe deρ_0 ne contient qu'un singleton. Imaginons que nous souhaitons construire une nouvelle partition ρ_1 sur Ω composée seulement de 4 classes. Il faut donc regrouper deux points parmi les cinq. L'objectif en classification est de chercher des partitions dans lesquelles tous les individus d'une même classe sont les plus semblables possible et deux individus appartenant à deux classes différentes soient les plus dissemblables. Dans ce cas, il semble naturel de ressembler les deux point qui sont les plus proches dans l'espace de représentation. En considérant la distance euclidienne comme mesure de proximité.

Les points les plus proches sont ω_1 et ω_3 car pour toute couple $(i, j) \epsilon \{1, 2, 3, 4,5\}^2$; $d_e(\omega_1, \omega_3) \leq d_e(\omega_i, \omega_j)$. Nous les regroupons dans une même classe A= $\{\omega_1\omega_3\}$.

$\rho_1 = \{\{\omega_1\omega_3\}, \{\omega_2\}, \{\omega_4\}, \{\omega_5\}\}$ et le tableau de distance devient :

$$d(A, \omega_j) = \max_{j \epsilon \{2.4.5\}} \{d(\omega_1\omega_j), d(\omega_3\omega_j)\}$$

$$= \max \{5.85 ; 3.35 ; 4.47\}$$

de	A= $(\omega_1\omega_3)$	ω_2	ω_4	ω_5
A= $(\omega_1\omega_3)$	0	5.85	3.35	4.47
ω_2	5.85	0	7.07	1.5
ω_4	3.35	7.07	0	5.34
ω_5	4.47	1.5	5.34	0

Tableau2.3. Tableau de distance de la partition ρ_1

Poursuivons notre processus en construisant une nouvelle partition ρ_2 formée de trois classes. Il faut donc, à nouveau regrouper deux éléments parmi les quatre de la partition ρ_1 . En observant la figure 1.4 précédente, il semble évident de

mettre les points ω_2 et ω_5 dans une même classe d'où, $\rho_2 = \{\{\omega_1\omega_3\}, \{\omega_2\omega_5\},\{\omega_4\}\}$ avec : B = (ω_2,ω_5)

d (B, A) = max {d (A, ω_2), d (A,ω_5)}=d (A, ω_5) = 5.85

d (B,ω_4) =max {d(ω_2,ω_4), d(ω_4,ω_5)}=d(ω_2,ω_4) =7.07

D	A= $(\omega_1\omega_3)$	B= $(\omega_2\omega_5)$	ω_4
A= $(\omega_1\omega_3)$	0	5.85	3.35
B= $(\omega_2\omega_5)$	5.85	0	7.07
ω_4	3.35	7.07	0

Tableau2.4.Tableau de distance de partition ρ_2

Cherchons maintenant une partition ρ_3 en deux classes. Nous l'obtenons par le regroupement de $\{\omega_1\omega_3\}$ avec $\{\omega_4\}$ (les éléments les plus proches), d'où $\rho_3 = (\{(\omega_1\omega_3), \omega_4\}, \{\omega_2\omega_5\})$ la classe C= $\{\omega_1, \omega_3, \omega_4\}$

D	C= (A,ω_4)	B= $(\omega_2\omega_5)$
C= (A,ω_4)	0	7.07
B= $(\omega_2\omega_5)$	7.07	0

Tableau2.5 Tableau de distance de la partition ρ_3

Enfin, nous obtenons la partition grossière ρ_4= $(\{\omega_1, \omega_2, \omega_3, \omega_4, \omega_5\})$ en réunissant les deux classe B et C dans une même classe D = (B, C)

Le processus s'arrête parce que nous avons obtenu la partition grossière

L'axe gradué permet de repérer la distance à laquelle deux classes de points ont été regroupées. On peut voir ainsi que les classes C= (A,ω_4) et B= $(\omega_2\omega_5)$ ont été regroupées à une distance 7.07, les classes A= $\{\omega_1, \omega_3\}$ et $\{\omega_4\}$ ont été réunies à une distance de 5.85, les classes $\{\omega_2\}$ et $\{\omega_5\}$ à une distance 1.5 et enfin $\{\omega_1\}$ et $\{\omega_3\}$ à la distance 1.41.

Cette structure induit une nouvelle matrice de distance. En effet, puisque la distance entre les classes $\{\omega_1\omega_3\}$ et $\{\omega_4\}$ est de 3.35. On pourrait alors dire que la distance entre, d'une part $\{\omega_1\}$ et $\{\omega_4\}$ et d'autre part $\{\omega_3\}$ et $\{\omega_4\}$ sont

Egales : $d'(\omega_1, \omega_4) = d'(\omega_3, \omega_4)$

En suivant le même raisonnement on détermine $d'(\omega_1, \omega_5) = 7.07$ car, c'est à cette distance que ces deux point ont été regroupés. La nouvelle matrice des distances entre chaque paire de point qui résulte de cette construction est la suivante :

d'	ω_1	ω_2	ω_3	ω_4	ω_5
ω_1	0	7.07	1.41	3.35	7.07
ω_2	7.07	0	7.07	7.07	1.5
ω_3	1.41	7.07	0	3.35	7.07
ω_4	3.35	7.07	3.35	0	7.07
ω_5	7.07	1.5	7.07	7.07	0

Tableau2.6.Tableau des distances ultramétriques

Il s'agit bien d'une matrice de distance car elle est nulle sur toute la diagonale, elle est symétrique et l'inégalité triangulaire est vérifiée pour toute paire de points. Mais cette matrice possède une propriété supplémentaire. Si nous considérons un triplet de points quelconques, disons *x*, *y* et *z* alors nous avons nécessairement, au moins, l'une des trois égalités suivantes qui est vérifiée :

$$d'(x, y) = d'(x, z)$$
$$d'(y, x) = d'(y, z)$$
$$d'(z, x) = d'(z, y)$$

Par conséquent, tout triplet de points forme ainsi, dans l'espace de représentation un triangle isocèle puisque au moins deux de ses cotés sont égaux. Cette propriété résulte d'une propriété dite ultra- métrique.

Définition on appelle ultra métrique sur M toute application d' : M*x*M $\longrightarrow \mathbb{R}_+$

Possédant les quatre propriétés suivantes :

$(p_1) : \forall (x, y) \in M^2 \; d'(x, y) = 0 \Leftrightarrow x = y$

$(p_2) : \forall (x, y) \in M^2 \; d'(x, y) = d'(y, x)$

$(p_3) : \forall x \in M, \; \forall y \in M, \forall z \in M, \; d'(x, y) \leq d'(x, z) + d'(y, z)$

$(p_4) : \forall x \in M, \; \forall y \in M, \forall z \in M, \; d'(x, y) \leq \max \{d'(x, z), d'(y, z)\}$

4.6. Interprétation

Il semble naturel d'accepter l'idée selon laquelle, la meilleure partition est celle où les dissimilarités entre individus d'une même classe sont les plus faibles et les dissimilarités entre individus de classes différentes sont les lus fortes. Dans ce contexte, l'indice de la hiérarchie va nous aider à déterminer la meilleure partition. En effet, si l'indice de la hiérarchie fait un saut important par passage de la partition ρ_i à la partition ρ_{i+1} cela signifie que les deux classes que l'on vient de réunir sont relativement éloignées.

Sur l'exemple nous pouvons constater que le saut le plus important a été effectué pour passer de ρ_3 à ρ_4 car l'indice de la hiérarchie est passé de 3.35 à 7.07. Comparativement aux précédentes valeurs, il s'agit d'une variante brusque. Cela traduit donc l'existence d'amas $(\{\omega_1, \omega_3, \omega_4\}, \{\omega_4, \omega_5\})$ de points relativement éloignés. La meilleure partition qui apparait est celle pour laquelle l'indice de la hiérarchie h est tel que $3.35 \leq 7.07$ c'est-à-dire $\rho_3 = (\{\omega_1, \omega_3, \omega_4\}, \{\omega_4, \omega_5\})$. En général, nous sélectionnons quelques partitions suffisamment bien contrastées et nous regardons si l'une ou plusieurs ont un sens pratique. Cette notion de sens pratique est entièrement subjective sur notre exemple, les deux partitions qu'il faut chercher à interpréter sont ρ_2 qui possède trois classes et ρ_3 qui est moins fine et qui ne possède que deux classes.

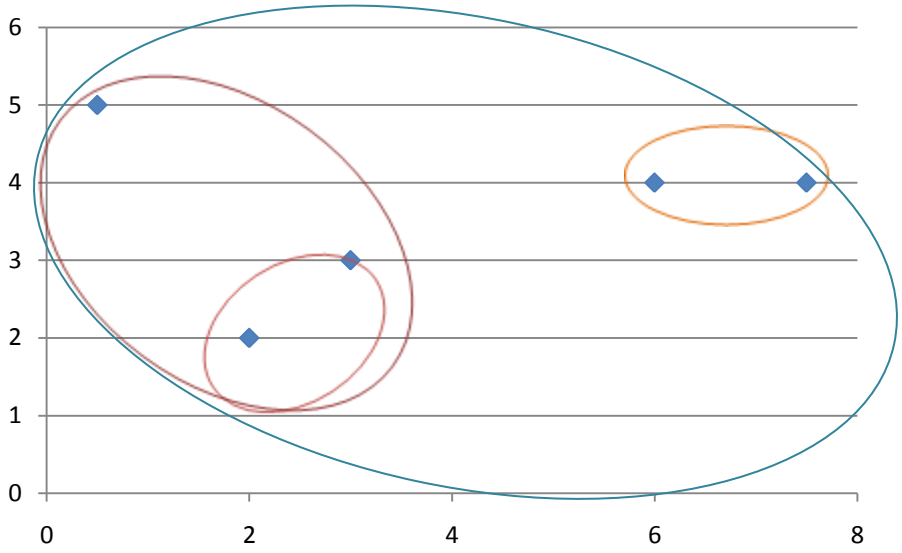

Figure 2.3. Hiérarchie de partitions obtenues par C.A.H (nuage de points)

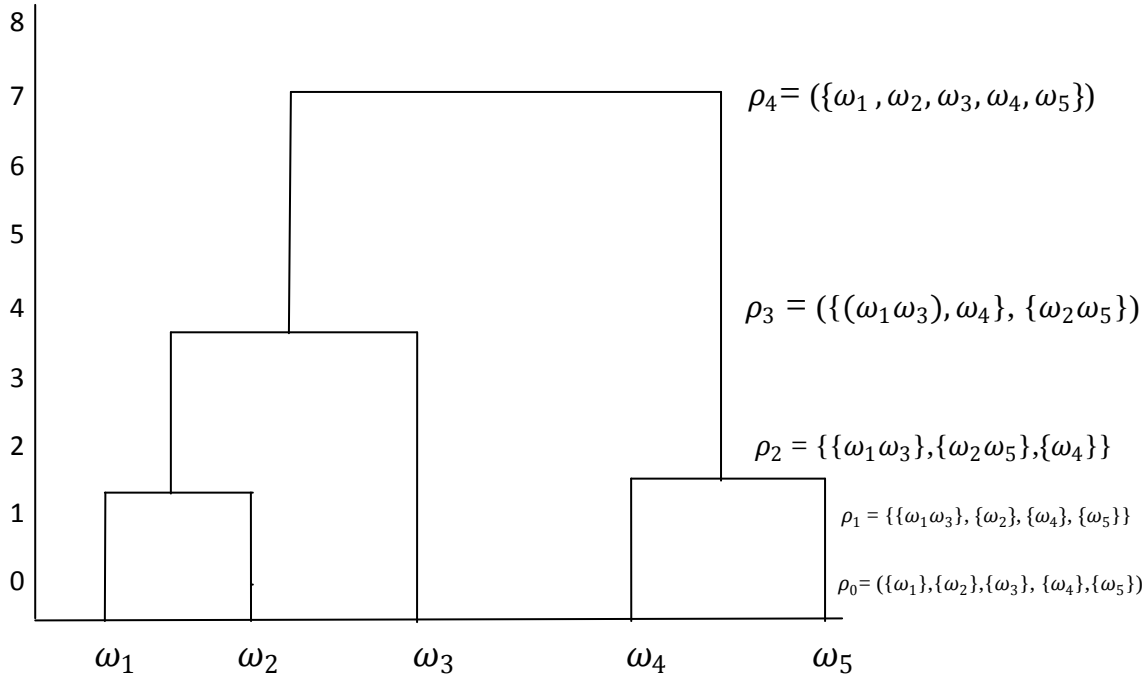

Figure 2.4. Hiérarchie de partitions obtenues par C.A.H (dendrogramme)

5. Conclusion

La classification hiérarchique ascendante (CAH) consiste à agréger progressivement les individus selon leur ressemblance, mesurée à l'aide d'un indice de similarité ou de dissimilarité. Elle nécessite la définition d'une mesure de similarité ou de distance entre les objets à classifier, échantillons; et d'un critère d'agrégation des classes qui peut être défini comme une mesure de similarité ou de distance entre les classes d'objets Elle produit une suite de partitions emboîtées de l'ensemble d'objets à classifier. Au départ, on a une partition en n classes, chaque classe étant composée d'un seul objet (partition la plus fine).L'algorithme commence par rassembler les couples d'individus les plus ressemblants, puis à agréger progressivement les autres individus ou groupes d'individus en fonction de leur ressemblance, jusqu'à ce que la totalité des individus ne forme plus qu'un seul groupe.

La CAH produit un arbre binaire de classification (dendrogramme), dont la racine correspond à la classe regroupant l'ensemble des individus. L'ensemble des nœuds définit une «hiérarchie» sur l'ensemble d'objets.

Chapitre3

Application

1. Présentation de java

Java est un langage de programmation à usage général, évolué et orienté objet dont la syntaxe est proche du C. Ses caractéristiques ainsi que la richesse de son écosystème et de sa communauté lui ont permis d'être très largement utilisé pour le développement d'applications de types très disparates. Java est notamment largement utilisé pour le développement d'applications d'entreprises et mobiles[22].

1.1. Edition de Java

- Java SE : Java Standard Edition (JDK = Java SE development Kit)
- Java EE : Entreprise Edition qui ajoute les API pour écrire des applications installées sur les serveurs dans des applications servelet JSP, JSF, EJB.
- Java ME : Micro Edition, version pour écrire des programmes embarqués (java à puce, java.card, téléphone portable) [22].

1.2. Environnement de programmation

Java est un environnement de programmation objet composé de :

- **Langage orienté objet** JAVA
- **JVM** (machine virtuelle Java) : permettant d'interpréter et d'exécuter le bytecode Java
- **API** (Application Programming Interface) : un ensemble de classes standards (bibliothèque standard)
- **JRE** (Java Runtime Environment) : l'environnement d'exécution Java désigne un ensemble d'outils permettant l'exécution de programmes Java sur toutes les plates-formes supportées
- **JDK (**Java Development Kit) le nouveau terme c'est SDK (Standard Dvelopment Kit) qui est l'environnement dans lequel le code java est compilé pour être transformé en bytecode afin que la machine virtuelle Java (JVM) puisse l'interpréter [17].

1.3. Les caractéristiques de java

- **Simple** : plus simple que le C ou C++ car on lui retiré les caractéristique peu utilisées ou difficile à utiliser : [17]
  - Pointeurs
  - Héritage multiple

- Le mécanisme de libération de la mémoire (garbage-collector) est transparent contrairement au C++.

℘ **Orienté objet** : une classe contient des attributs et des méthodes

℘ **Distribué** : il est facile de mettre en place une architecture Client-serveur pour travailler avec des fichiers situés sur un ordinateur distant.

℘ **Multithread** : pour l'exécution simultanée de plusieurs processus. Java est fourni avec un jeu de primitives qui facilitent l'écriture de ce genre de programmes.

℘ **Robuste** : le typage des données est très strict, tant à la compilation qu'a l'exécution.

℘ **Dynamique** : les classes de java peuvent être modifiées sans modification du programme qui les utilise.

℘ **Portable** : le compilateur java fabrique du bytecode « universel ». pour l'exécuter sur une machine quelconque, il faut qu'un interpréteur java (machine virtuelle) existe pour cette machine. Les types de données sont indépendants de la plate forme.

℘ **Haute performance** : le compilateur java génère du bytecode optimisé en incluant des instructions qui permettront à chaque interpréteur de tirer la meilleure performance possible du code.

1.4. Les IDE (integreted development environnement)

Le monde Java bénéfice sans doute des meilleurs IDE qui offrent une palette de fonctionnalités nous permettant d'atteindre un excellent niveau de productivité. Les IDE existent en java :

♦ **Eclipse** : est une plate-forme de développement écrite en Java, fruit du travail d'un consortium de grandes entreprises (IBM, Borland, Rational Rose, HP) il en résulte d'un IDE performant et Open Source qui a su trouver sa place comme l'un des environnements de développement Java les plus populaires.

♦ **Netbeans** : créé à l'initiative de Sun Microsystems (Noyau de Forte4j/Sun One), présente toutes les caractéristiques indispensables à un EDI de qualité, que ce soit pour développer en Java, Ruby, C/C++ ou même PHP.

Les effets négatifs de la migration de Netbeans développement vers un seul environnement. Comme d'autres environnements de développement intégrés, l'EDI NetBeans fournit une interface utilisateur graphique pour les outils en ligne de commande qui gère la compilation, le débogage et le packaging des applications.

L'IDE Netbeans sera utilisé dans l'application de Classification Hiérarchique Ascendante C.H.A [22] .

1.5. Programmation avec l'interface graphique

L'utilisateur peut interagir à tout moment avec plusieurs objets graphiques (bouton, liste déroulante, menu, champ de texte…).

Ces articles peuvent modifier totalement le cheminement du programme

L'ordre d'exécution des instructions ne peut pas être prévu à l'écriture du code [22].

1.6. Programmation conduite par les événements

Une interface graphique pose une façon particulière de programme

La programmation conduite par les événements est du type suivant

- Les actions de l'utilisateur (déplacement, clic souris, frappe de touche du clavier)
- Engendrement des événements qui sont mis dans une file d'attente

Le programme récupère un à un ces événement et les traites[17].

1.7. Boite à outils graphique

Les boites à outils graphique offrent des facilités pour utiliser et gérer la file d'attente des événements en particulier pour associer les événements avec les traitements qu'ils doivent déclencher.

1.8. Les API

2 Bibliothèques :

- AWT (Abstract Windows Toolkit) tous les composants de AWT ont leurs équivalents dans SWING
- SWING offre des facilitées pour construire des interfaces graphiques
  - les noms de composants commencent par un J.
  -Tous héritent du composant graphique JComponent.[22]

2. Présentation de l'interface graphique Netbeans

JFrame permettant d'obtenir des éléments graphiques dans Java Gui Forms.

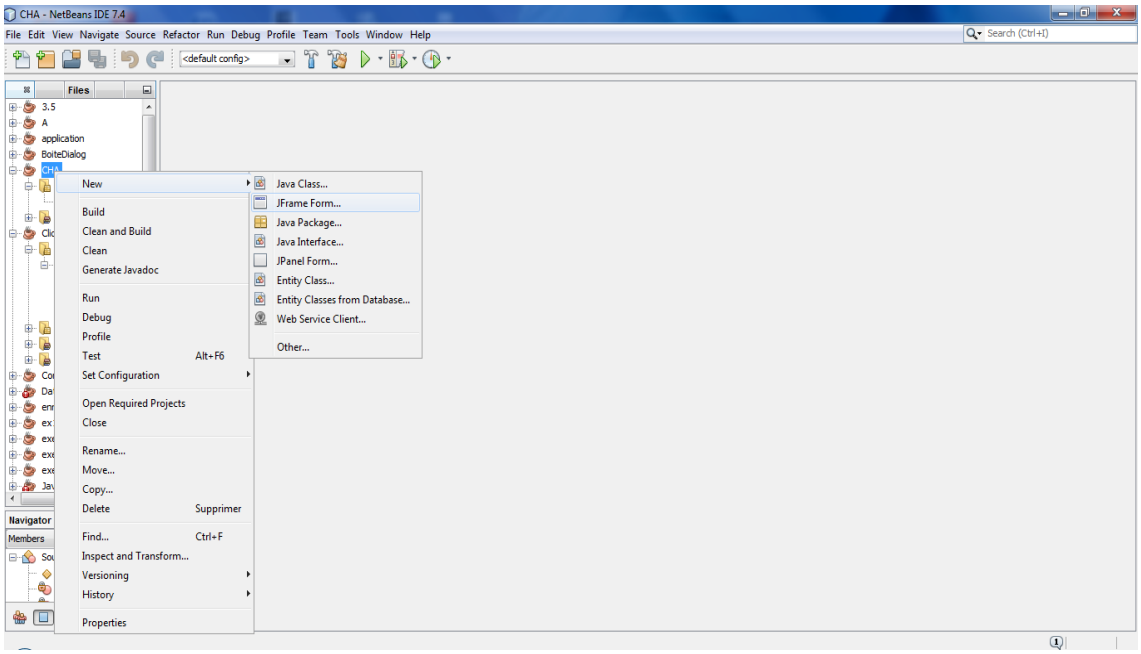

Figure3.1. créer nouveau JFrame

Le projet se présente alors comme suit :

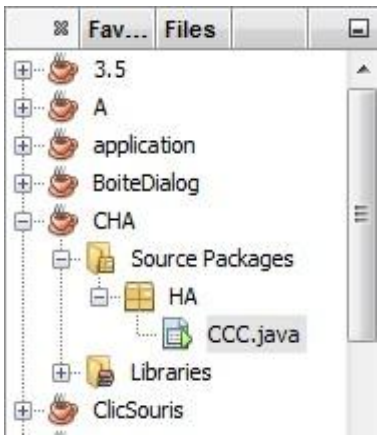

Figure3.2. emplacement de JFrame dans le projet

Une fenêtre graphique peut se manipuler sous forme graphique (utilisez l'ongle Design) ou sous forme textuelle (onglet Source). La construction d'une interface se fait en mode Design.

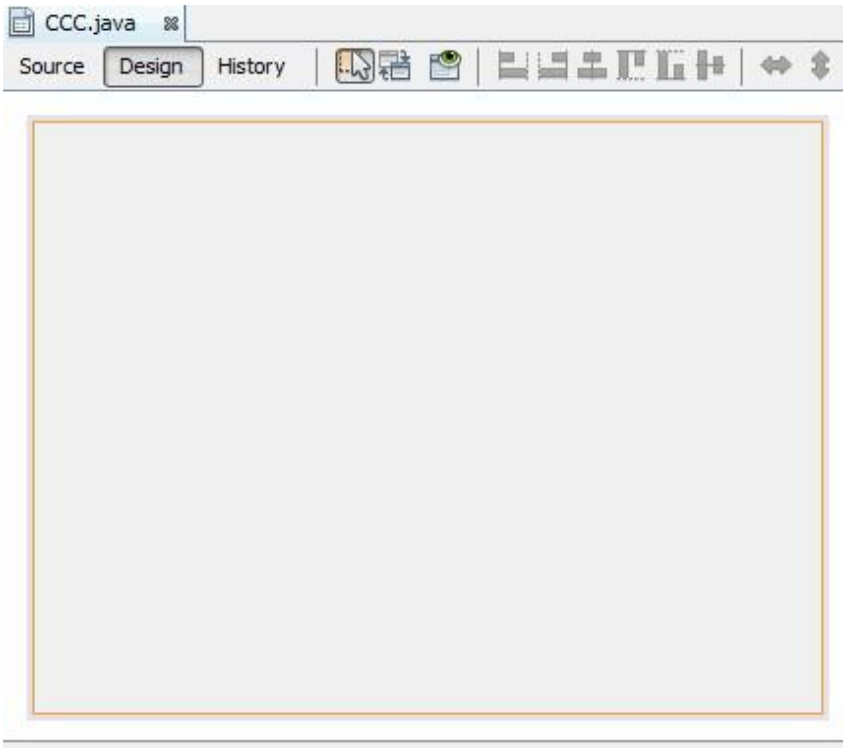

Figure 3.3. Modèle d'une interface graphique Design

Tous les objets graphiques nécessaires à une interface sont regroupés dans l'onglet Palette.

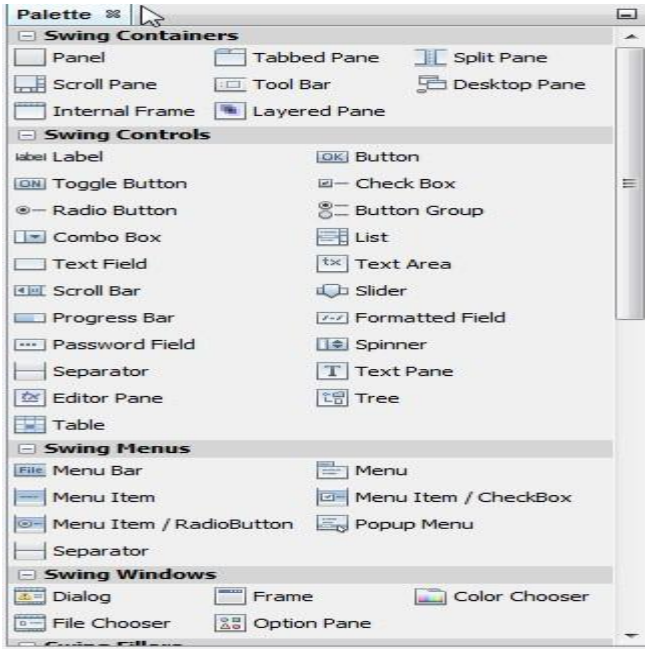

Figure3.4. palette des composants

L'onglet Propriétés permet de manipuler ses propriétés (couleur, position, forme,….).

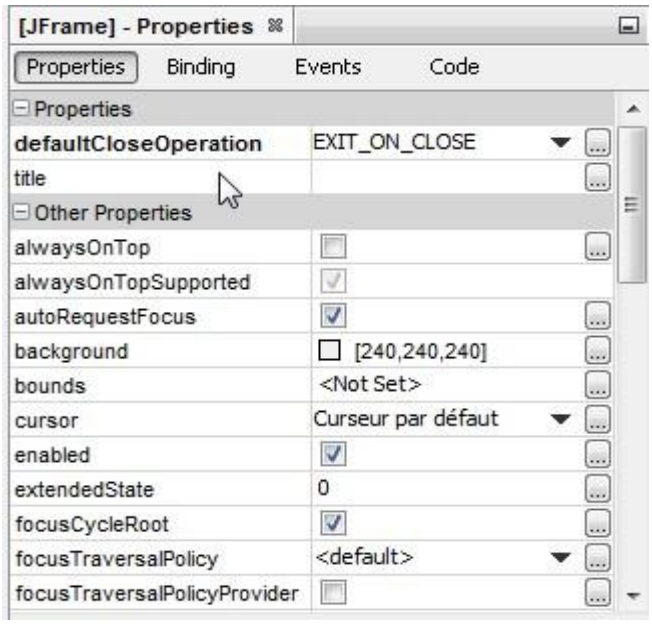

Figure3.5. onglet propriétés

L'inspecteur fait la représentation graphique de l'application (component).

L'inspecteur est utilisé pour changer les titres des variables.

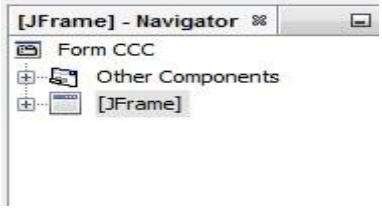

Figure3.6. inspecteur

Terminal Windows : tous les sorties de l'exécution de program seront afficher dans le terminal Windows.

Figure3.7. le terminal

3. Application

L'interface principale de notre application est comme suit

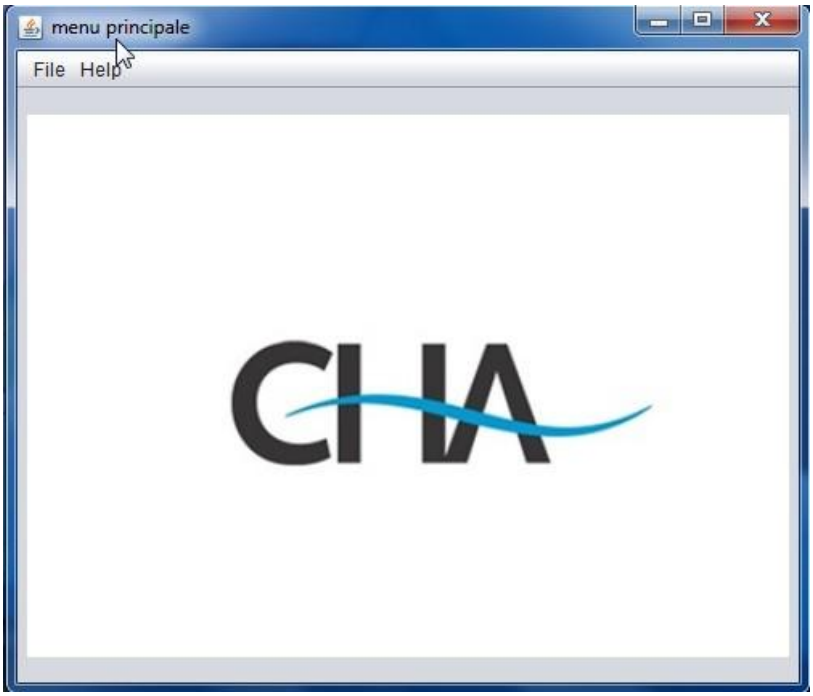

Figure3.8. La page d'accueil

Dans la page d'accueil, on a deux Menu « File »et « Edit ».
Chaque menu est composé de deux jMenu item comme suit :

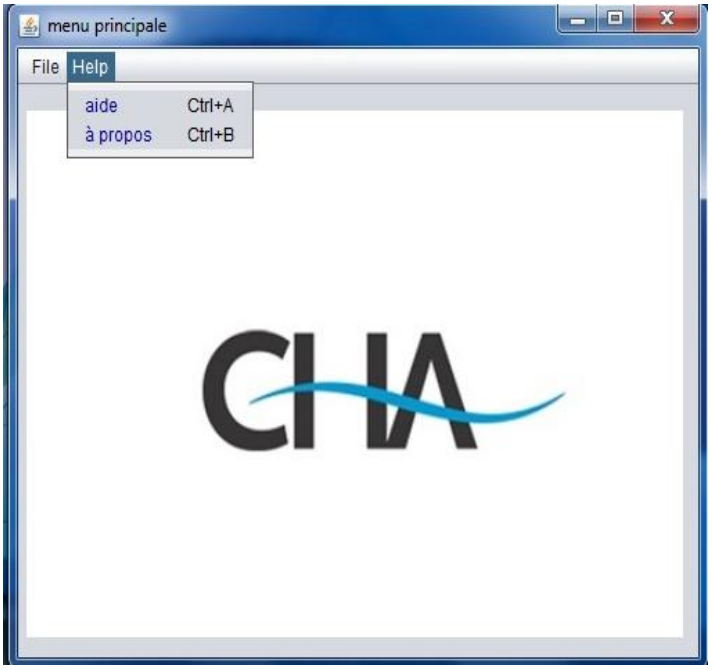

Figure 3.9. Le menu Help

- **Aide** : donne une idée générale sur le rôle de chaque bouton ;

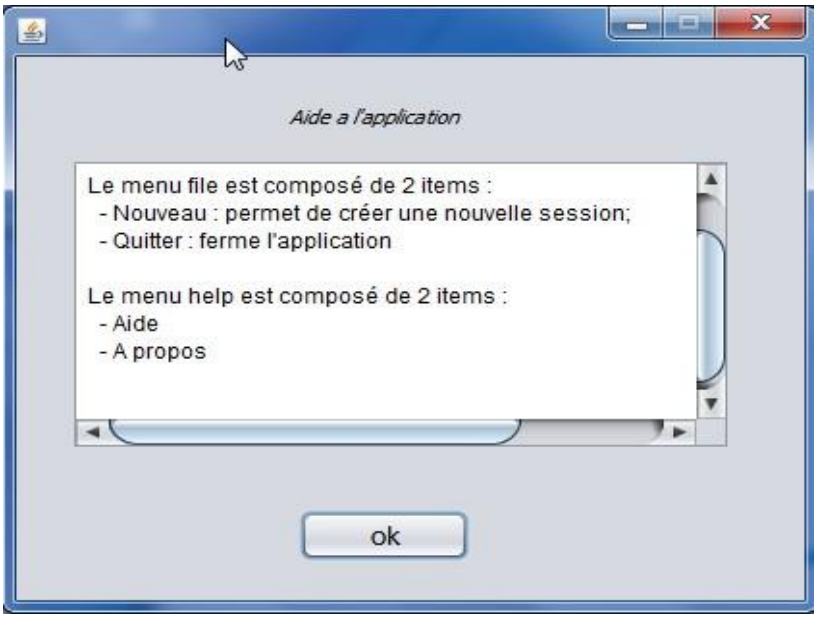

Figure3.10. page d'aide

- **A propos** : nous informe sur la réalisation de se projet

Le menu File :

Figure3.11. Le menu File

- **Nouveau :** permet d'accéder a la fenêtre « clics souris » ;
- **Quitter :** pour sortir sans effectuer aucune simulation ;

  La fenêtre clics souris est une fenêtre composée d'un panneau, 2 labels, un bouton et un tableau.

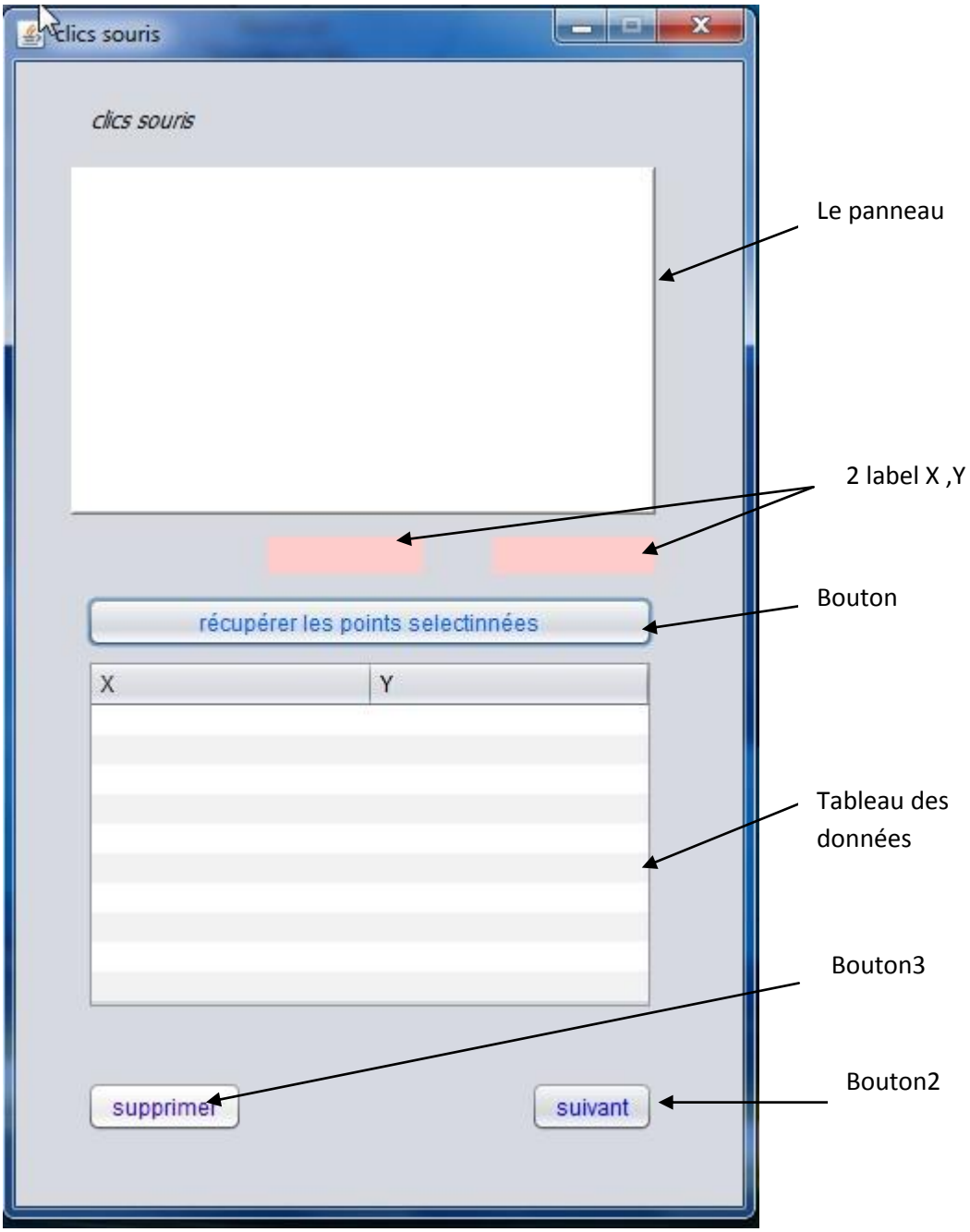

Figure3.12. la fenêtre clics souris

- **Le panneau :** pour traiter l'événement clics souris, on associe à l'objet jPanel un écouteur de souris la méthode correspondante à cet évènement : *mouseClicked* correspond à un clic usuel (appui suivi d'un relâchement).
- **Les labels :** les 2 labels permettent de récupérer les coordonnées. lors de du clic un certain nombre d'information transmis vers l'écouteur comme les cordonnées du curseurs de souris au moment du clic par les méthodes *getX ,getY* , et pour les afficher dans les labels par *setTexte*

- **Le bouton « récupérer les point sélectionnées » :** ce bouton permet de récupérer les coordonnées des points choisis (sélectionnées) dans un jTable « tableau de données » par la méthode getModel() et les afficher dans le tableau par setValueAt .
- **Le tableau des données :** c'est un objet de type jTable qui contient les coordonnées des points sélectionnées (X, Y)
- **Le bouton « suivant » :** permet d'accéder a la fenêtre « mesure de distances »
- **Le bouton supprimer :** pour supprimer les points insérés dans le panneau et récupérer dans le tableau.

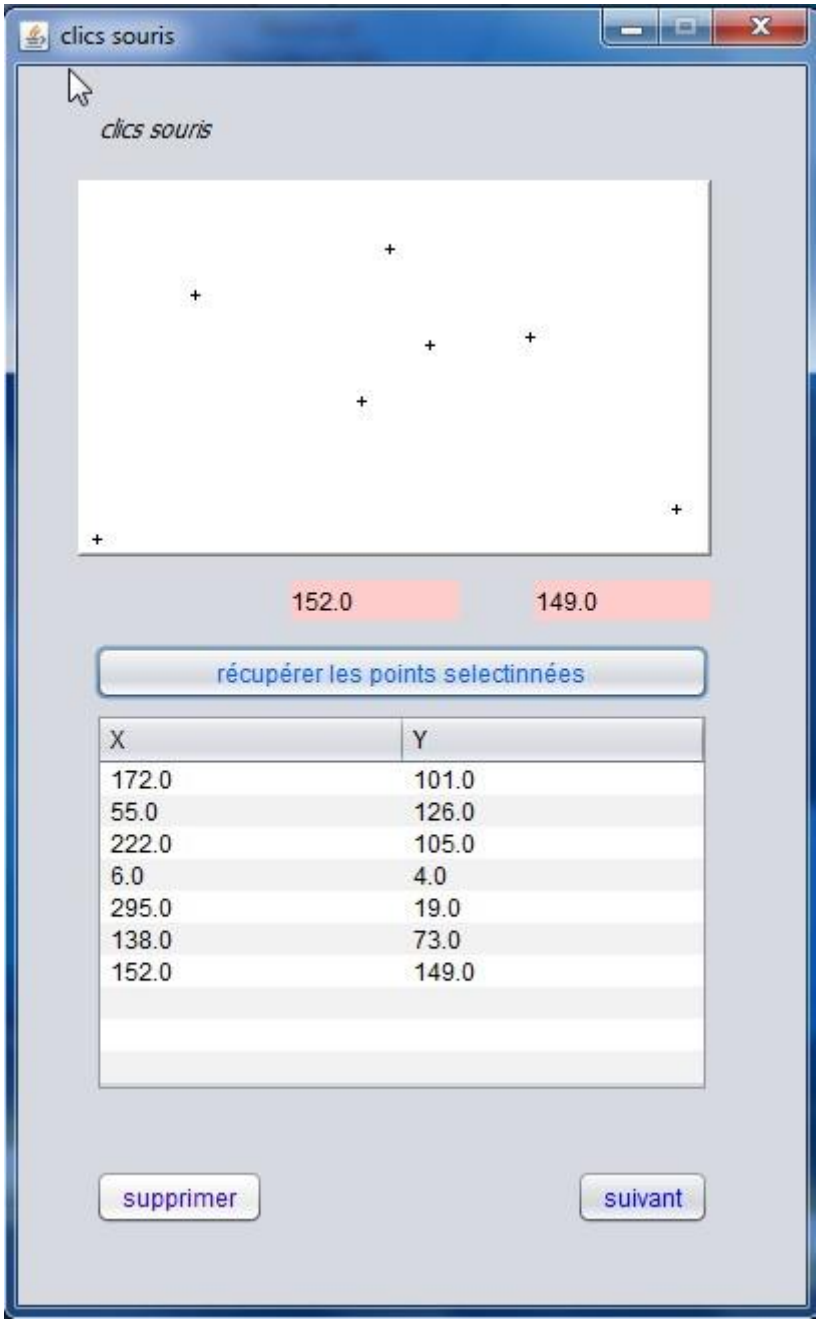

Figure3.13. exemple de récupération des données

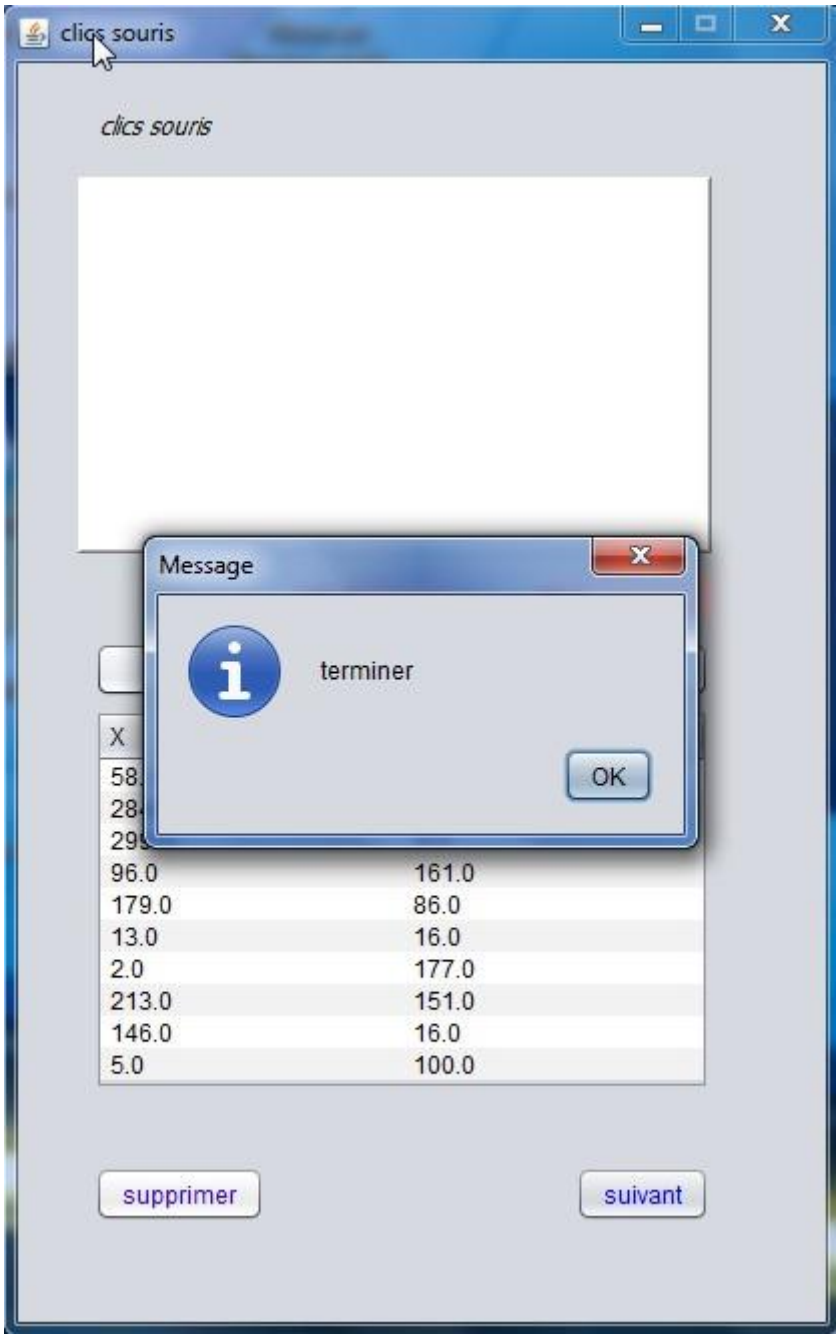

Figure3.14. affichage de message "terminer"

Quand le nombre de points récupérer dépasse le nombre de lignes du tableau le programme affiche un message « terminer » dans une boite de dialogue avec un bouton "ok" pour retourner à la fenêtre clics souris pour suivi l'opération de CHA.

Après le clic sur le bouton suivant on accède à la fenêtre "mesure de distances"

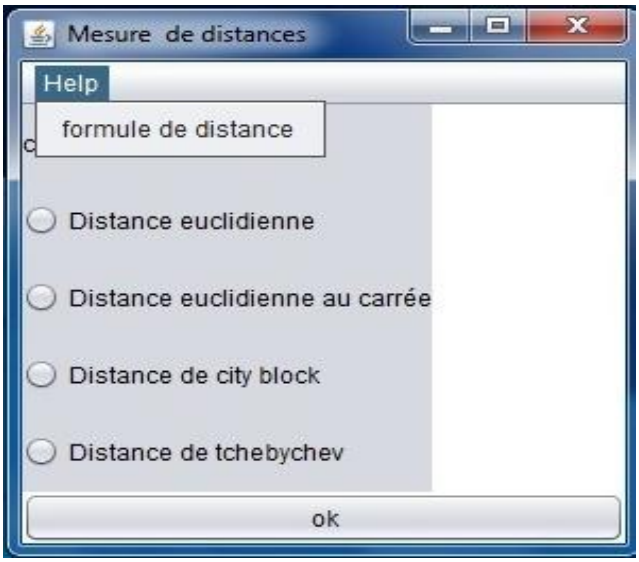

Figure3.15. fenêtre des distances

Le menu Help permet afficher les formule de calcule des distances

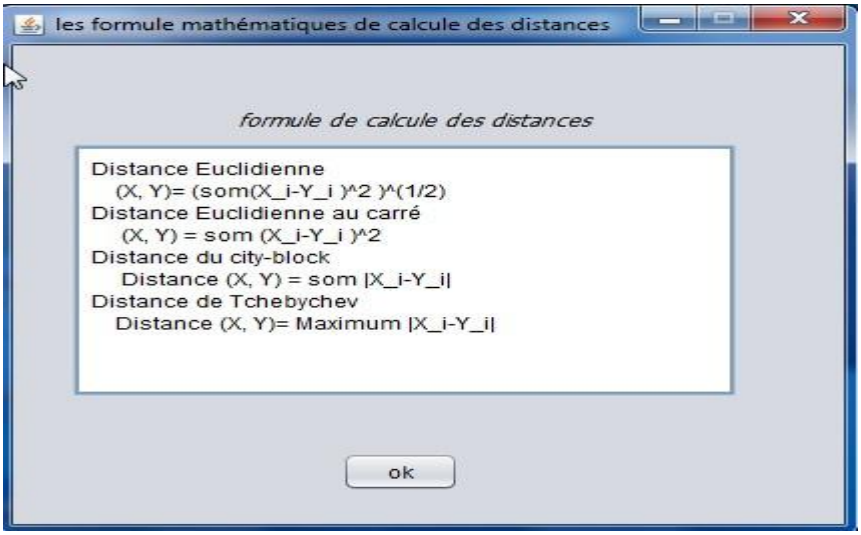

Figure3.16. le menu Help (formule de calcule des distances)

Si l'utilisateur n'a sélectionnée aucune distance le programme affiche le message :

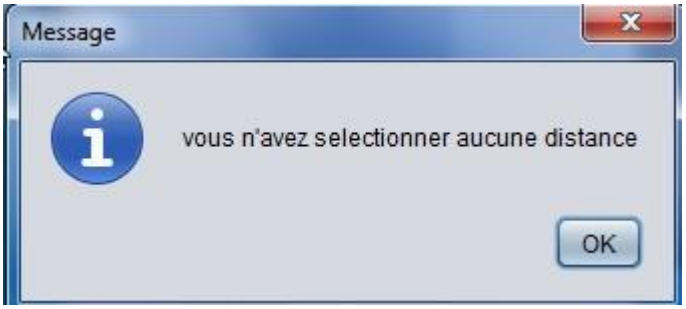

Figure3.17. message "demande de sélection"

Sinon :

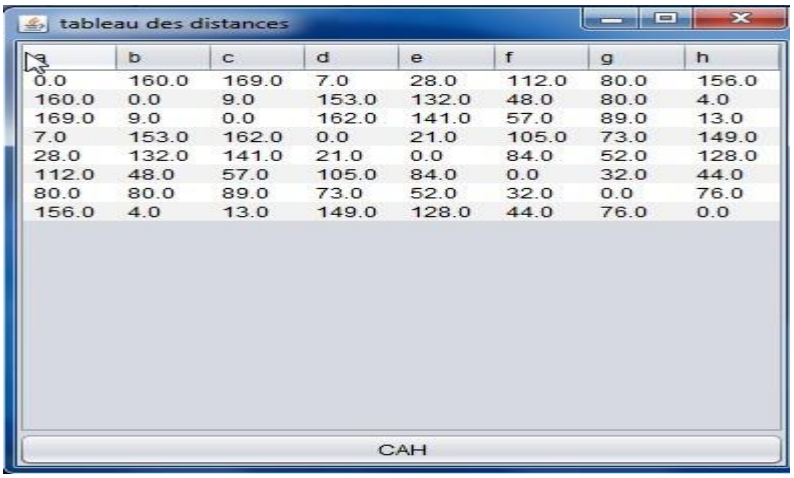

Figure3.18. tableau de distance

Le bouton CAH permet d'afficher la fenêtre des critères :

Figure3.19. fenêtre des critères d'agrégation

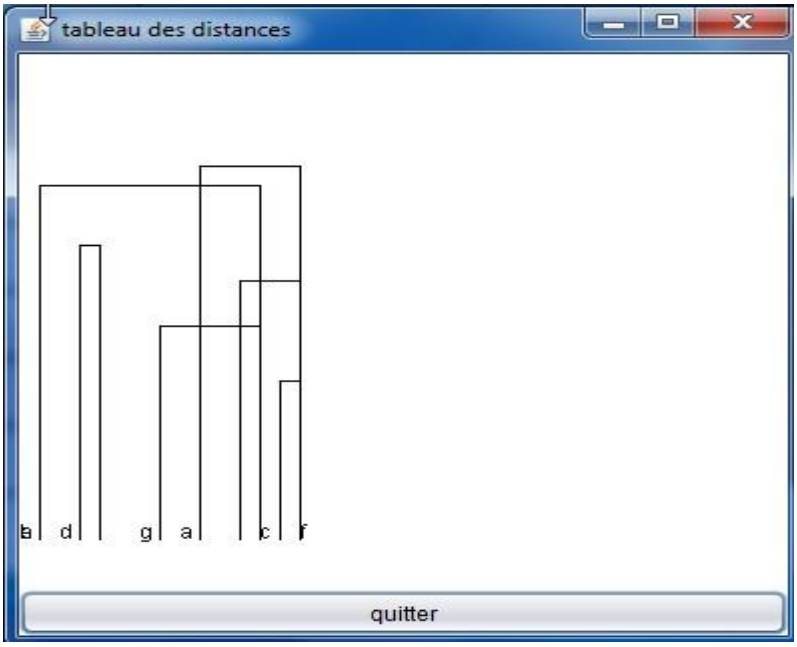

Figure3.21. dendrogramme

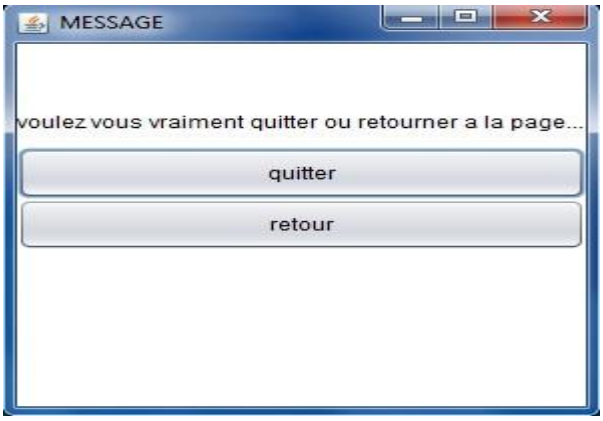

Figure3.22. message (quitter, retour)

4. Conclusion

Dans ce chapitre nous avons présenté la partie essentielle de notre projet de fin d'étude. L'idée de notre travail consiste à appliquer la théorique de CAH sur l'interface graphique Netbeans par la création d'un panneau (événement clic souris) et appliquer l'algorithme de CAH sur les données extraite de cet événement.

Les résultats se changent lorsque les critères et les mesures de distances se changent.

Conclusion générale

La classification est une méthode d'analyse des données qui vise à regrouper en classes homogènes un ensemble d'observations. Ces dernières années, les besoins d'analyse de données et en particulier la classification ont augmenté significativement. En effet, de plus en plus de domaines scientifiques nécessitent de catégoriser leurs données dans un but descriptif ou décisionnel. Nous avons vue notamment que la classification se divise généralement en deux sous-problèmes distincts : la classification supervisée, et la classification non supervisée.

Nous avons intéressées dans ce travail à la classification hiérarchique ascendante dont le procédé consiste à grouper les observations en classe par agrégation successive jusqu'à ce que toutes les observations fassent partie de la même classe. Pour cela nous avons traité les différents critères de dissimilarité, alléger les différents aspects mathématique de calcul, comme nous avons cité les étapes nécessaires pour obtenir les classes de partitionnement.

Pour finir nous avons décrit notre application avec les différents exemples et résultats.
Ce travail nous a apporté dans un premier temps des nouvelles connaissances dans le domaine de la classification hiérarchique ascendante et l'utilisation de ces propriétés.

Comme perspectives on peut traiter d'autres approches de classifications comme le supervisé et le non supervisé.

Références Bibliographiques

[1] F.SEYTE, M.TERRAZZA : statistique appliquées à la gestion.

[2] Arnaud MARTIN : l'analyse des données.

[3] Jean-Pierre NAKACHE approche pragmatique de la classification.

[4] Samuel AMBAPOUR Introduction à l'analyse des données.

[5] Mounzer Boubou contribution de classification non supervisée via des approches pré topologiques et d'opinion.

[6] Christophe Biernacki Pourquoi les modèles de mélange pour la classification ?

[7] Maurice ROUX algorithmes de classification.

[8] Yves Tillé Cours de Statistique Descriptive

[9] Ricco Rakotomalala Étude des dépendances - Variables qualitatives

[10] E. Lebarbier, T. Mary-Huard Classification non supervisée

[11] Fatma Karem, Mounir Dhibi, Arnaud Martin Combinaison de classification supervisée et non-supervisée par la théorie des fonctions de croyance

[12] F.-X. Jollois Classification

[13] Philippe Cibois Principe de l'analyse factorielle

[14] Lebart, L., Morineau, A., Piron M., Analyse exploratoire multidimensionnelle

[15] G. DREYFUS LES RÉSEAUX DE NEURONES

[16] support de cours (chapitre2).

[17] cours de Mr Benmamar

[18] classification automatique chapitre 6

[19] www.statsoft.fr/concepts-statistiques/classification

[20] stadon.voila.net/classif.htm

[21] www.math.univ-toulouse.fr

[22] developpez.com

[23] PSY83A - Analyses multidimensionnelles et applications informatiques

[24]http://tutoriels-data-mining.blogspot.com/2010/02/discretisation-comparaison-de-logiciels.html.

[25] http://www.dbmsmag.com/9807m05.html

[26] www.fxpal.com/people/denoue/teaching/classification.doc.

Annexe A

```
// déclaration de tableau des distances
float [][] dist;
   dist= new float[nbl][nbl];
  for( int i=0;i<nbl;i++  ){
        for (int j=0; j<nbl; j++){
           dist[i][j]=0;
         }
        }
// calcule de distance euclidienne
for( int i=0;i<nbl;i++  ){
       for (int j=0; j<nbl ; j++){
          for (int k=0; k <nbc; k++){
            if(i==j)
               dist[i][j]=0;
        else   dist[i][j]=(float)((float)Math.pow((tableau[i][k]-tableau[j][k]),2  )+ dist[i][j]);
                  }
                    }
                      }
   for( int i=0;i<nbl;i++  ){
       for (int j=0; j<nbl ; j++){
          dist[i][j]= (float)Math.sqrt(dist[i][j]);
       }
   }
//calcule de distance euclidienne au carrée
for( int i=0;i<nbl;i++  ){
       for (int j=0; j<nbl ; j++){
          for (int k=0; k <nbc; k++){
            if(i==j)
               dist[i][j]=0;
          else   dist[i][j]=(float)(Math.pow((tableau[i][k]-tableau[j][k]) , 2 )+ dist[i][j]);
                 }
                   }
                      }
// calcule  de distance de city block
for( int i=0;i<nbl;i++  ){
       for (int j=0; j<nbl ; j++){
          for (int k=0; k <nbc; k++){
            if(i==j)
               dist[i][j]=0;
          else   dist[i][j]=(float)(Math.abs((tableau[i][k]-tableau[j][k])) + dist[i][j]);
                 }
                   }
                      }
// calcule de distance de  tchebychev
float [][]max;
         max = new float[nbl][nbl];
  for( int i=0;i<nbl;i++  ){
   for (int j=0; j<nbl ; j++){
     for (int k=0; k <nbc; k++){
            if(i==j)
            max[i][j] =0;
```

```
          else {
       if   ( (float) (Math.abs (tableau[i][k]-tableau[j][k]) ) > dist[i][j] );
        max[i][j]=(float) (Math.abs(tableau[i][k]-tableau[j][k]));

       }
         }
}
}
```

Figure A.1. Calcule de distance implémentée par Java

```
// critère du  plus  loin voisin
int l=0;
  l=nbl;   // nbl c'est le nombre de ligne de tableau initial des données
float v [];    // déclaration de vecteur v vecteur de plus loin valeur de la valeur min
 v= new float[nbl];

float v 1[];    // déclaration de vecteur v vecteur de plus proche valeurs de la valeur min
 v1= new float[nbl];
float de[][];   //tableau qui reçoit les nouvelle valeurs  après calcule de min
       de= new float[l][l];
       for(int i=0; i<l; i++){
          for(int j=0;j<l; j++){
             de [i][j]=0;
          }
          }
float min =0;  // calcule le minimum dans le tableau des distances
  int s =0,sauv  = 0;  // deux variable locale pour sauvegarder les indice de min dans le
tableau des distances
for( int n=0;n<nbl-2;n++){
     for (int i=0 ; i < nbl; i++ ){
           for (int j=0; j<nbl; j++){
              if ((dist[i][j]!=0)){
                        min=dist[i][j] ;
                       s=i;
                        sauv=j ;break;
              }
              }
             }
for (int i=0 ; i < nbl; i++ ) {
                              for (int j=0; j<nbl; j++)
                        if ((dist[i][j]!=0)& (dist[i][j]<min))
                         {
                        min=dist[i][j];
                        s=i;
                        sauv=j ;
                     }
```

```
                        }
System.out.println("min["+s+"]["+sauv+"]=" + min);

       // initialisée le vecteur a 0
        for (int i=0; i<nbl; i++){
        v[i]=0;
           }
      for (int i=0; i< nbl;i++)  {
         for (int j=0; j< nbl; j++){
            if((i!=s)&&(i!=sauv))
            {
               if (dist[s][i]>dist[sauv][i])
         v[i] =dist[s][i];
               else v[i]= dist[sauv][i];
                  }
      }
      }
      for( int i=0;i<nbl; i++){
      System.out.println("plus loin c'est v["+i+"]"+v[i]);
       }


    for(int i=0; i<l; i++){
        for(int j=0;j<l; j++){
     if((i!=s)&&(j!=sauv)&&(i!=sauv)&&(j!=s))
           dist [i][j]=dist[i][j];
           else  dist [i][j]=0;
        }
        }
      for(int i=0; i<l; i++){
         for(int j=0;j<l; j++){
       if((i==sauv)| (j==sauv))
          dist[sauv][j]=v[i];
                    }
      dist[i][sauv]=v[i];
      }
   for(int i=0; i<l; i++){
         for(int j=0;j<l; j++){
              if((i==sauv)| (j==sauv))

               dist[sauv][j]=v[j];

      }
      dist[i][sauv]=v[i];
      }

      for(int i=0; i<nbl; i++){
         for(int j=0;j<nbl; j++){
            de[i][j]=dist[i][j];
            System.out.print("de["+i+"]["+j+"]"+dist[i][j]+"   ");
```

```
          }
          System.out.println("");
          }



      // plus proche voisin
for( int n=0;n<nbl-2;n++){
     for (int i=0 ; i < nbl; i++ ){
          for (int j=0; j<nbl; j++){
             if ((dist[i][j]!=0)){
                      min=dist[i][j] ;
                     s=i;
                      sauv=j ;break;
             }
             }
            }
for (int i=0 ; i < nbl; i++ ) {
                          for (int j=0; j<nbl; j++)
                     if ((dist[i][j]!=0)& (dist[i][j]<min))
                      {
                     min=dist[i][j];
                     s=i;
                     sauv=j ;
                  }
                      }
 System.out.println("min["+s+"]["+sauv+"]=" + min);

       // initialisée le vecteur a 0
        for (int i=0; i<nbl; i++){
        v1[i]=0;
           }
      for (int i=0; i< nbl;i++)  {
         for (int j=0; j< nbl; j++){
            if((i!=s)&&(i!=sauv))
            {
               if (dist[s][i]<dist[sauv][i])
         v1[i] =dist[s][i];
               else v1[i]= dist[sauv][i];
                  }
      }
      }
      for( int i=0;i<nbl; i++){
      System.out.println("plus loin c'est v1["+i+"]"+v1[i]);
       }
```

```
  for(int i=0; i<l; i++){
     for(int j=0;j<l; j++){
   if((i!=s)&&(j!=sauv)&&(i!=sauv)&&(j!=s))
        dist [i][j]=dist[i][j];
        else  dist [i][j]=0;
     }
     }
   for(int i=0; i<l; i++){
     for(int j=0;j<l; j++){
    if((i==sauv)| (j==sauv))
      dist[sauv][j]=v1[i];
                }
   dist[i][sauv]=v1[i];
   }
for(int i=0; i<l; i++){
     for(int j=0;j<l; j++){
         if((i==sauv)| (j==sauv))

          dist[sauv][j]=v1[j];

  }
   dist[i][sauv]=v[i];
   }

   for(int i=0; i<nbl; i++){
     for(int j=0;j<nbl; j++){
        de[i][j]=dist[i][j];
        System.out.print("de["+i+"]["+j+"]"+dist[i][j]+"   ");
     }
     System.out.println("");
     }
```

Figure A.2. Calcule des critères d'agrégation

Résumé

La classification hiérarchique ascendante est une méthode de classification qui permet de mettre en évidence un regroupement « naturel » d'un ensemble d'individus décrits par des caractéristiques (les variables). Elle propose une série de partitions emboîtées représentées sous forme d'arbres appelés dendrogrammes. L'algorithme procède par agrégations successives, partant de la partition la plus fragmentaire, un individu est égal à une classe, jusqu'à la partition triviale, le regroupement de tous les individus dans une et une seule classe.

Mots clés : analyse des données, classification hiérarchique ascendante, dendrogramme, similarité, dissimilarité.

Abstract

Ascending hierarchical classification is a classification method that allows to identify a "natural" grouping of a set of individuals described by characteristics (variables). It proposes a series of nested partitions represented as trees called dendrograms. The algorithm proceeds by successive aggregations, starting from the most fragmentary partition, an individual is equal to a class, to the trivial partition, grouping all individuals in one and only one class.

Key words : data analysis, hierarchical cluster analysis, dendrogram, similarity, dissimilarity.

ملخص

. التصنيف الهرمي التصاعدي هو طريقة التصنيف الذي يسمح بتحديد مجموعة الأفراد يقترح سلسلة من الأقسام المتداخلة ممثلة بالأشجار يسمى تمثيل بالأعمدة . عائدات الخوارزمية من قبل المجموعات المتتالية، بدءا من القسم الأكثر تجزئة، فرد يساوي فئة، إلى القسم الأكبر تجمع جميع الأفراد في واحدة وفئة واحدة فقط.

الكلمات المفتاحية التصنيف الهرمي التصاعدي ,تحليل البيانات, تمثيل بالأعمدة ,التشابه، الاختلاف .