



UNIVERSITE MOHAMED BOUDIAF - M'SILA
FACULTE DES MATHÉMATIQUES ET
DE L'INFORMATIQUE



DEPARTEMENT D'INFORMATIQUE

MEMOIRE de fin d'étude

Présenté pour l'obtention du diplôme de MASTER

Domaine : Mathématiques et Informatique

Filière : Informatique

Spécialité : Systèmes d'Informations Avancé

Par: Benslimane Afaf

SUJET

Qualité des données dans le processus ECD

Soutenu publiquement le : 1 / 06 /2016 devant le jury composé de :

**A.amroun
T.Mehenni
M.bounif**

**Université de M'sila
Université de M'sila
Université de M'sila**

**Président
Rapporteur
Examineur**

Promotion : 2016 /2017

Dédicace

Je dédie ce mémoire à :

Mes parents :

Ma mère Sabah , qui a œuvré pour ma réussite, de par son amour, son soutien, tous les sacrifices consentis et ses précieux conseils, pour toute son assistance et sa présence dans ma vie, reçois à travers ce travail aussi modeste soit-il, l'expression de mes sentiments et de mon éternelle gratitude.

Mon père Ben slimane saleh, qui peut être fier et trouver ici le résultat de longues années de sacrifices et de privations pour m'aider à avancer dans la vie. Puisse Allah faire en sorte que ce travail porte son fruit ; Merci pour les valeurs nobles, l'éducation et le soutien permanent venu de toi.

Mon frère Oussama et mes sœurs Yasmine ,Inas,Malak et Hanaa qui n'ont cessé d'être pour moi des exemples de persévérance, de courage et de générosité.

Mes professeurs de MI qui doivent voir dans ce travail la fierté d'un savoir bien acquis.

Afaf

REMERCIEMENT

Je remercie en premier lieu Allah tout puissant de m' avoir permis de mener à terme ce travail .

Je témoigne une reconnaissance très particulière pour mon encadreur et je lui renvoie l'expression de toute mon reconnaissance pour ses conseils et orientations et l'aide apporté pour mon projet. Merci monsieur Mehenni Tahar

Je remercie monsieur Benazi Makhelouf pour ses précieux conseils et son aide

Je remercie tous les professeurs qui ont contribué de près ou de loin à mon formation universitaire.

Enfin, tenant à remercier tout ce qui m'ont aidé de près ou de loin pour la réussite de ce travail.

Table de matière

INTRODUCTION GENERALE.....1

CHAPITRE1 : PROCESSUS ECD ET DATA MINING

1. Introduction.....	3
2. Le processus ECD.....	3
2.1. Présentation	3
2.2. Données, information, connaissance	4
2.3. L'acquisition des données	5
2.4. Le prétraitement des données	5
2.5. La transformation des données	6
2.6. La fouille de données (ou Data mining).....	6
3. Méthodes et techniques de data mining.....	7
3.1. La description	8
3.2. Le clustering	9
3.2.1. Clustering hiérarchique	9
3.2.2. Clustering descendante	10
3.2.3. Clustering par la methode K-means	12
3.3. Les règles associatives	15
3.3.1. Présentation.....	15
3.3.2. Avantages et inconvénients.....	15
3.3.3. Les algorithmes des règles associatives	16
3.4. L'estimation.....	17
3.4.1. Principe	17
3.4.2. La régression linéaire.....	17
3.5. Segmentation.....	19
3.5.1. Principe.....	19
3.5.2. Clustering supervisé (Classification).....	19
3.5.3. Clustering bayésien.....	22
3.5.4. Réseaux de neurone	23
3.5.5. SupportVector Machine	24
4. Conclusion.....	25

CHAPITRE 2 : METHODES DE PRETRAITEMENT DES DONNEES

1. Introduction	26
2. Concepts fondamentaux des bases de données.....	27
2.1. Définition	27
2.2. Intérêt des bases de données.....	27
2.3. Schéma conceptuel d'une base de données.....	28
3. Préparation des données.....	28
3.1. Définition et compréhension du problème.....	29
3.2. Collecte des données.....	30

3.3.	Prétraitement.....	30
4.	Nettoyage des données	31
4.1.	Etapas de nettoyage de données	31
4.1.1.	Analyse es données :	32
4.1.2.	Définition des flux de transformation	32
4.1.3.	Vérification.....	32
4.1.4.	Transformation	32
4.1.5.	Feedback des données nettoyées	32
4.2.	Les techniques de nettoyage de données	32
4.2.1.	Mesure de la qualité de la base	32
4.2.2.	Détection des fautes de frappe	32
4.2.3.	Valeurs manquantes.....	32
4.2.4.	Valeurs saillantes.....	35
4.2.5.	Codes inconsistants.....	35
4.2.6.	Extraction des valeurs concaténées d'un attribut.....	35
4.2.7.	Conflits de typage et de nommage.....	35
4.2.8.	Elimination des redondances (minimalité et complétude).....	35
5.	Intégration de données	36
6.	Transformation de données	36
7.	Sélection des données	37
8.	Conclusion	37

CHAPITRE 3 : SYSTEME DE NETTOYAGE DES DONNEES

1.	Introduction.....	38
2.	Le langage de programmation et de développement c#.....	38
2.1.	Présentation de c#.....	38
2.2.	Composants élémentaires du C#.....	39
2.3.	Visual Studio c#.....	39
3.	Présentation des données du système de nettoyage.....	39
4.	Description des fonctionnalités du système	41
4.1.	Description générale du système.....	41
4.2.	Traitement des valeurs manquantes	41
4.2.1.	Traitement des valeurs manquantes par moyenne	41
4.2.2.	Traitement des valeurs manquantes par suppression.....	43
4.2.3.	Traitement des valeurs manquantes par régression.....	45
4.3.	Elimination des mots étrangers.....	45
4.4.	traitement de l'incohérence	47
5.	Evaluation du système de nettoyage.....	49
6.	interface du système de nettoyage.....	49
7.	conclusion.....	50

CONCLUSION GENERALE.....57

Liste des figures

Fig. 1.1: Etapes principales du ECD.....	4
Fig. 1.2 : Etats éventuels de changements des données brutes.....	5
Fig 1.3 : Exemple d'un dendrogramme.....	12
Fig 1.4: Exemple de Clustering par K-means.....	13
Fig. 1.5 : La droite de régression.....	19
Fig.. 1.6: Exemple d'arbre de décision.....	25
Fig. 1.7 : Support Vector Machine	24
Fig. 2.1: Emplacement de la base de données et ses clients.....	27
Fig. 2.2 : Exemple d'une base de données (bibliothèque).....	28
Fig. 2.3 : Processus de data minig.....	29
Fig 2.4 : Résultat de nettoyage des données.....	31
Fig. 2.5 : Exemples d'intégration de données.....	36
Fig. 3.1: Interface visual studio c#.....	39
Fig. 3.2 : Description de structure de la table des données.....	40
Fig. 3.3: Un extrait des données (non traitées) de la table.....	41
Fig. 3.4 : Organigramme de traitement des valeurs manquante par moyenne.....	42
Fig. 3.5 : Exemple de donnée manquante de l'attribut âge.....	43
Fig. 3.6: Etat de la base après remplissage de la valeur manquante.....	43
Fig. 3.7 : Organigramme de traitement des valeurs manquantes par suppression.....	44
Fig 3.8 : Organigramme de la méthode d'élimination des mots étrangers.....	46
Fig 3.9 : Organigramme de la méthode d'élimination de l'incohérence.....	48
Fig.3.10: Interface du système de nettoyage.....	49

La liste des table :

Tab. 1.1: Classes des méthodes de data	8
Tab. 1.2 : Quelques algorithmes des règles d'association	16
Tab. 2.1: Méthodes d'imputation.....	34

Chapitre 1 : processus ECD et data mining

CHAPITRE 1

PROCESSUS ECD ET DATA MINING

1. Introduction :

L'accroissement du volume des données stockées dans les bases de données et les entrepôts de données et l'émergence du Big Data, a fait qu'il est devenu urgent et vital de recourir à des techniques et méthodes pour extraire de l'information à partir de cette masse volumineuse de données. Le processus d'Extraction des Connaissances à partir des Données (ECD) et en particulier le Data mining sont alors devenus rapidement des thèmes de recherche suscitant l'intérêt de la communauté scientifique.

Dans ce chapitre nous allons tout d'abord faire un tour d'horizon sur le processus ECD, ensuite nous présentons sommairement les techniques et les applications du data mining.

2. Le processus ECD

2.1. Présentation :

Le processus d'extraction des connaissances à partir des données (ECD) constitue un champ de recherche important dans lequel de nombreux problèmes restent à résoudre. L'ECD est un processus à plusieurs étapes, il commence généralement par l'importation des données à partir des différentes sources de production des données, ensuite vient le stockage et l'organisation des ces données dans des entrepôts de données (EDs) où les données sont prêtes pour la fouille (le data mining) en utilisant certains algorithmes de fouille de données.

Le processus d'ECD, sous la supervision d'un spécialiste, se déroule en cinq phases : l'acquisition des données, le prétraitement et le nettoyage des données, la transformation et la mise en forme des données, la fouille de données, et finalement la validation et la mise en forme des connaissances produites (voir Figure 1.1). L'entreposage des données a pour objet d'organiser des très grands volumes de données, de les structurer et de les préparer à l'analyse en les stockant dans les entrepôts de données (EDs) qui sont des composants fondamentaux du processus ECD. [13]

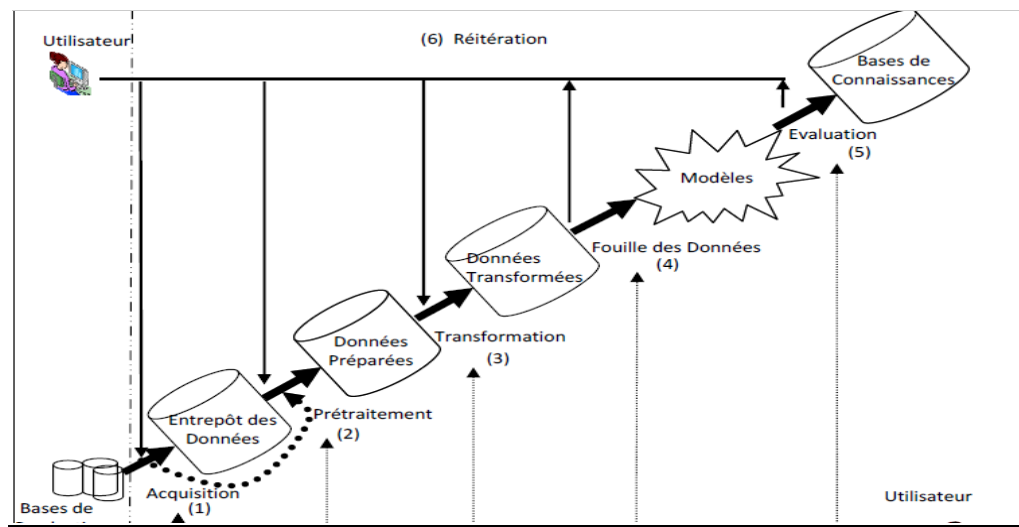


Fig. 1.1: Etapes principales du processus ECD [13]

2.2. Données, information, connaissance :

Avant de continuer l'explication des différentes étapes de l'ECD, il est préférable de donner la différence qui existe entre : données, information et connaissance.

Une donnée est un élément brut qui peut être collectée par un outil de supervision, par une personne ou être déjà présente dans une base de données par ex. Une donnée seule ne permet pas de prendre une décision sur une action à lancer.[3]

Une information est une donnée à laquelle un sens et une interprétation ont été donnés. Une information permet à un responsable opérationnel de prendre une décision sur une action à mener.[3]

Une Connaissance est une information évaluée et organisée de telle sorte qu'elle peut être utilisée délibérément.[3].

La figure 1.2 illustre les étapes de transformation des données brutes jusqu'à l'obtention de la connaissance et effectivement de la sagesse.

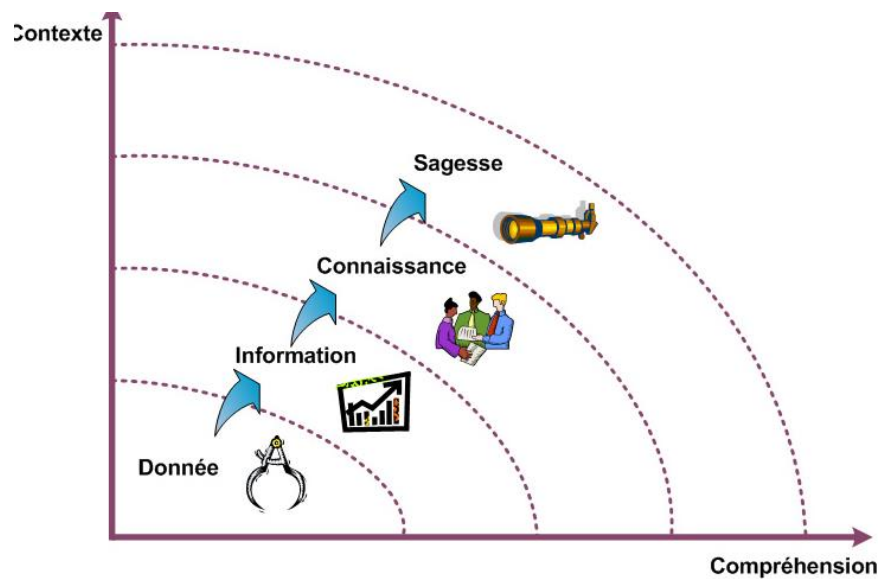


Fig. 1.2 : Etats éventuels de changements des données brutes [3]

2.3 L'acquisition des données :

Cette étape est la première dans le processus ECD. Elle consiste au regroupement, l'intégration, la fusion, la mise en forme, bref à la préparation des données afin de mieux les organiser et les structurer.

Les données peuvent être de différentes sources : bases de données, fichiers de texte, templates, fichiers XML, ... Leur acquisition consiste à mieux les organiser, les structurer et enfi les stocker dans des récipients adéquats, généralement des entrepôts des données.

2.4 Le prétraitement des données :

Une fois acquises, les données doivent passer par une autre étape très délicate et très indispensable : le prétraitement. En effet, les données issues de différentes sources présentent souvent diverses anomalies : doublons, champs vides, bruit, ..., d'où l'étape de prétraitement qui va s'occuper de résoudre toutes ces anomalies afin d'avoir des données de bonne qualité.

Les connaissances produites à partir de données de mauvaise qualité ont des conséquences graves sur les décisions prises par les utilisateurs, quel que soit le domaine d'application. Pour cela, la qualité des données et des connaissances est un sujet d'intérêt dans le processus d'ECD. Toutes les applications dédiées à l'ECD requièrent différentes formes de préparation des données avec de nombreuses techniques de traitement, afin que les données ne contenant pas d'incohérences, de doublons, de valeurs manquantes ou incorrectes. Le processus d'ECD s'intéresse à :

1. l'acquisition des données ou il procède à la sélection, l'intégration et le nettoyage des données cibles.

2. le prétraitement des données où l'ECD exécute des opérations complémentaires du nettoyage et sélection des données

L'étape de prétraitement s'avère très intéressante, car tout simplement, la qualité de la fouille de données dépend très étroitement de la qualité des données. Pour plus de détails sur cette étape, le chapitre suivant sera pleinement consacré aux différentes techniques de prétraitement des données.

2.5 La transformation des données :

Cette étape regroupe les différentes transformations que les données doivent subir avant d'être disponibles pour l'analyse: l'agrégation, la normalisation, la réduction des dimensions d'analyse, ...

La transformation des données est souvent nécessaire pour que les algorithmes de fouille de données (data mining), étape suivante, soient plus pertinents. Les données sont transformées ou consolidées dans un format approprié et adéquat à la tâche de fouille de données.

La préparation de données signifie généralement l'ensemble des trois étapes précédentes, à savoir : l'acquisition, le prétraitement et la transformation des données. La préparation des données occupe environ 60 à 80% du temps impliqué dans le processus ECD. La difficulté de cette préparation peut être comprise à partir de deux perspectives : les problèmes relatifs aux données, ainsi que les problèmes relatifs au processus.

2.6. La fouille de données (ou Data mining):

L'étape de fouille de données (FD) du processus d'ECD est la plus importante. Elle consiste à l'application d'algorithmes de FD sur les données. Il existe différentes tâches de la FD, chacune répondant à un problème différent. Il n'existe pas de technique de FD supérieure à toutes les autres pour tous les problèmes. Il faut choisir une technique en fonction des besoins des utilisateurs ainsi que des avantages et inconvénients de chacune d'elles. [15]

Le data mining est un ensemble des méthodes scientifiques destinées à l'exploration et l'analyse des données issues de grandes bases de données informatiques, en vue de détecter dans ces données des profils, des comportements récurrents, des règles, des liens, des tendances inconnues, qui devront nous aider à la prise des décisions. [23]

Le data mining signifie littéralement « fouille de données » ou « forage de données ». Ce procédé, basé sur une série d'algorithmes ou modèles de data mining, permet d'extraire des informations à partir des données, informations qui, grâce à l'analyse, se convertissent en connaissances. [23]

Le data mining est l'analyse d'un ensemble d'observations qui a pour but de trouver des relations insoupçonnées et résumer les données d'une nouvelle manière, de façon qu'elles soient plus compréhensibles et utiles pour leurs détenteurs. Autrement dit, il consiste à analyser des informations collectées dans des entrepôts de données afin d'y détecter des relations qu'il serait a priori impossible d'identifier sans cet outil. C'est un élément essentiel dans la relation client et dans les systèmes d'aide à la décision. [23]

Pour plus de détails, la section suivante sera consacrée aux différentes méthodes et techniques de la FD.

3. Méthodes et techniques de data mining

Pour arriver à exploiter ces quantités importantes de données, le data mining utilise des méthodes d'apprentissages automatiques. Ces méthodes sont de deux types : les méthodes descriptives et les méthodes prédictives, selon qu'il existe ou non une variable "cible" que l'on cherche à expliquer.

Les méthodes descriptives (recherche de patterns) visent à mettre en évidence des informations présentes mais cachées dans le volume de données dans l'objectif de construire un modèle et de découvrir de relations entre les données.[4]. Nous citons entre autres: Le clustering(k-Mean, hiérarchique) et les règles d'association.[4]

Les méthodes prédictives (modélisation) visent à extrapoler de nouvelles informations à partir des informations présentes. Nous citons entre autres : la classification et la régression [4].

La table 1.1 illustre en détail ces deux types de méthodes. Il en découle six (06) classes de méthodes selon le degré de complexité et le type de prédiction. Il est également possible de combiner plusieurs méthodes (appelées dans ce cas méthodes hybrides) selon la nature des données et pour chercher une meilleure qualité des résultats du data mining.

Technique descriptives		Techniques prédictives		
Méthode simple	Méthode complexe	Présent		Futur
		Variable cible Numérique	Variable cible catégorielle	
Description	Clustering Assosiation	Estimation	Segmentation (ou Classification)	Prévision

Tab. 1.1: Classes des méthodes de data mining [15]

3.1 La description :

Principe :

La description consiste à mettre au jour

- Pour une variable donnée : la répartition de ses valeurs (tri, histogramme, moyenne, minimum, maximum, etc.).
- Pour deux ou trois variables données : des liens entre les répartitions des valeurs des variables.

Intérêt :

- Favoriser la connaissance et la compréhension des données.

Méthode :

- Méthodes graphiques pour la clarté : analyse exploratoire des données.

3.2 Le clustering :

Principe :

Le clustering consiste à créer des classes (c'est-à-dire des sous-ensembles) de données similaires entre elles et différentes des données d'une autre classe (autrement dit, l'intersection des classes entre elles doit toujours être vide).

Autrement dit, il s'agit pour n variables de créer des sous-ensembles disjoints de données. On dit aussi « **segmenter** » l'ensemble entier des données.

Le clustering définit les grands types de regroupement et de distinction : on parle de métatypologie (type de type). Elle permet une vision générale de l'ensemble (de la clientèle, par exemple).[15]

Intérêt :

- Favoriser, grâce à la métatypologie, la compréhension et la prédiction.
- Fixer des segments qui serviront d'ensemble de départ pour des analyses approfondies.
- Réduire les dimensions, c'est-à-dire le nombre d'attributs, quand il y en a trop au départ.

Méthodes :

- Clustering hiérarchique
- Clustering des K moyennes
- Règles d'association.

3.2.1. Clustering hiérarchique :

Le clustering hiérarchique constitue depuis longtemps une forme de clustering très populaire. Il a l'avantage d'être interprétable visuellement à l'aide des graphes ou dendrogrammes. Il est utilisé dans différents domaines : la taxinomie, la biologie, les réseaux de télécommunications, la phytosociologie, ... etc.

C'est une méthode de clustering automatique utilisée en analyse des données, à partir d'un ensemble de n objets, son but est de répartir ces individus dans un certain nombre de classes. [16]

On distingue deux types de clustering hiérarchiques :

- Le clustering ascendant hiérarchique : noté (C.A.H) qui se déroule comme suit : à partir des éléments terminaux, on forme de petites classes ne comportant que les individus les plus semblables, et à partir de celles-ci, on construit des classes de moins en moins homogène jusqu'à obtenir la classe tout entière qui réunit tous les éléments terminaux.[16]. Avec le CAH on travaille à partir des dissimilarités entre les objets que l'on veut regrouper. On peut donc choisir un type de dissimilarité adapté au sujet étudié et à la nature des données. L'un des résultats est le dendrogramme, qui permet de visualiser le regroupement progressif des données. On peut alors se faire une idée d'un nombre adéquat de classes dans lesquelles les données peuvent être regroupées.

- Le clustering descendant hiérarchique : noté (C.D.H), il s'agit d'une dichotomie de la classe entière jusqu'à obtenir tous les éléments terminaux.

Principe de clustering ascendant :

On commence par calculer la dissimilarité entre les N objets.

1. Puis on regroupe les deux objets dont le regroupement minimise un critère d'agrégation donné, créant ainsi une classe comprenant ces deux objets.
2. On calcule ensuite la dissimilarité entre cette classe et les N-2 autres objets en utilisant le critère d'agrégation. Puis on regroupe les deux objets ou classes d'objets dont le regroupement minimise le critère d'agrégation.

On continue ainsi jusqu'à ce que tous les objets soient regroupés.

Ces regroupements successifs produisent un arbre binaire de clustering, dont la racine correspond à la classe regroupant l'ensemble des individus. Ce dendrogramme représente une hiérarchie de partitions. On peut alors choisir une partition en tronquant l'arbre à un niveau donné, le niveau dépendant soit des contraintes de l'utilisateur (l'utilisateur sait combien de classes il veut obtenir), soit de critères plus objectifs.

Mesure de dissimilarité inter classe :

La dissimilarité de deux classes $C_1 = \{x\}, C_2 = \{y\}$ contenant chacune un individu se définit simplement par la dissimilarité entre ces individus.

$$dissim(C_1, C_2) = dissim(x, y)$$

Lorsque les classes ont plusieurs individus, il existe de multiples critères qui permettent de calculer la dissimilarité. Les plus simples sont les suivants :

- Le saut minimum retient le minimum des distances entre individus de C_1 et C_2 :

$$dissim(C_1, C_2) = \min_{x \in C_1, y \in C_2} (dissim(x, y))$$
- Le saut maximum est la dissimilarité entre les individus de C_1 et C_2 les plus éloignés :

$$dissim(C_1, C_2) = \max_{x \in C_1, y \in C_2} (dissim(x, y))$$
- Le lien moyen consiste à calculer la moyenne des distances entre les individus de C_1 et C_2 :

$$dissim(C_1, C_2) = moyenne_{x \in C_1, y \in C_2} (dissim(x, y))$$
- La distance de Ward vise à maximiser l'inertie inter-classe :

$$dissim(C_1, C_2) = \frac{n_1 * n_2}{n_1 + n_2} dissim(G_1, G_2)$$

avec n_1 et n_2 les effectifs des deux classes, G_1 et G_2 leurs centres de gravité respectif

Le dendrogramme

C'est la représentation graphique d'une clustering ascendante hiérarchique ; Il se présente souvent comme un arbre binaire dont les feuilles sont les individus alignés sur l'axe des abscisses. Lorsque deux classes ou deux individus se rejoignent avec l'indice d'agrégation τ , des traits verticaux sont dessinés de l'abscisse des deux classes jusqu'à l'ordonnée τ , puis ils sont reliés par un segment horizontal. À partir d'un indice d'agrégation τ , on peut tracer une droite d'ordonnée τ qui permet de voir une clustering sur le dendrogramme. La Figure 1.2 donne un exemple de dendrogramme

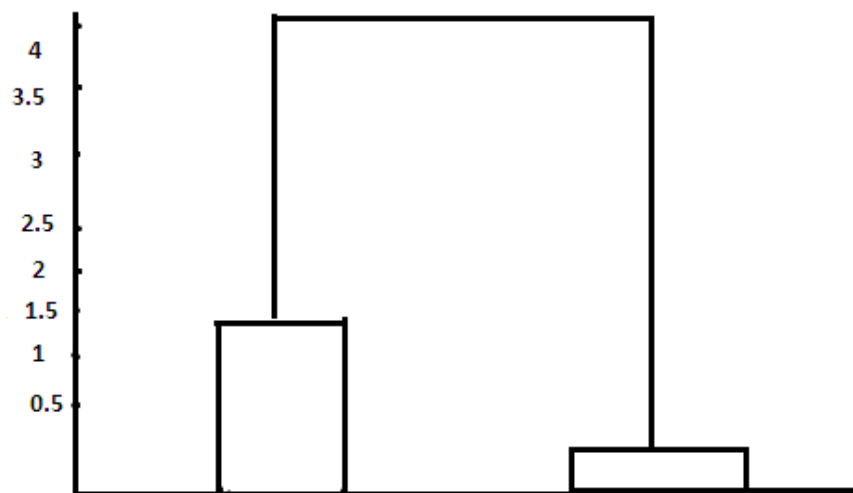


Fig 1.3 : Exemple d'un dendrogramme

3.2.2 Clustering descendante :

Les méthodes de construction descendante hiérarchique sont plus satisfaisantes à cet égard. Nous partons des toutes les observations qui forment au départ un seul et même groupe.

1. On choisit d'abord l'observation la plus éloignée de toutes les autres, elle formera le noyau d'un nouveau groupe
2. On rattache à ce groupe toutes les observations qui sont plus proches de ce groupe que du groupe initial. Nous avons donc deux groupes.
3. On recommence cette opération pour chaque sous-groupe ainsi obtenu jusqu'à obtenir une unique observation par groupe

Cette procédure algorithmique est donc à l'opposé de la précédente. On part d'un ensemble pour arriver à des singletons. Cette méthode de clustering est appelée méthode de Diana.

3.2.3. Clustering par la méthode K-means :

La méthode des "K-means" reste actuellement la méthode la plus utilisée surtout pour les grands fichiers de données qui contiennent plus de 40 000 individus. En effet, Ceux-ci ont répondu à une enquête sur les ventes par correspondance d'une entreprise afin d'obtenir des profils types de clientèle.

K-means est un algorithme de quantification vectorielle (clustering en anglais). K-means est un algorithme de minimisation alternée qui, étant donné un entier K, va chercher à séparer un ensemble de points en K clusters [18] (voir Figure 1.4).

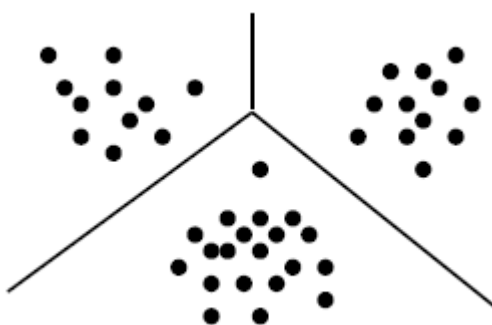


Fig 1.4: Exemple de Clustering par K-means [18]

Cette méthode à l'instar de la méthode hiérarchique, a l'avantage d'être efficace et très rapide. La clustering hiérarchique a l'inconvénient d'utiliser toutes les ressources de l'ordinateur. Elle procède par le calcul pour chaque point sa distance à tous les autres. Elle effectue ensuite un tri et enfin elle agrège les individus les plus proches. La méthode hiérarchique est itérative et elle est inefficace pour les grands fichiers de données. Le principe de la méthode des "k-means" c'est que la clustering se fait sur la base du critère des plus proches voisins. Celui-ci signifie que chaque individu est affecté à une classe s'il est très proche de son centre de gravité. [18]

La particularité de la méthode des "k-means" c'est que le nombre de classes doit être spécifié préalablement. La méthode la plus utilisée pour estimer ce nombre c'est de mener une clustering hiérarchique sur un échantillon représentatif de l'ensemble I des individus. Une autre manière de procéder est de se baser sur le nombre de classes obtenues par des clusterings ayant les mêmes objectifs que la présente étude.

Algorithme de la methode K-means :

Dans la méthode des "k-means", le choix des centres initiaux s'effectue sur la base d'un tirage aléatoire sans remise de k individus à partir de la population à classifier. La partition des classes est modifiée avec chaque affectation d'un individu i de I . [18]

L'algorithme classique des k -means laisse un paramètre libre. Généralement le choix de K est fait empiriquement en sélectionnant la valeur de k qui minimise l'énergie.

Les individus sont géométriquement représentés dans l'espace vectoriel P muni d'une distance notée d . L'algorithme de "k-means" se déroule comme suit :

Etape 0 :

1. On choisit par un tirage aléatoire sans remise k individus parmi n individus composant l'ensemble I . Ces k centres notés sont provisoires.
2. Chaque individu i de I est affecté à une seule classe. Chacune de ces classes est localisée par son centre. La procédure d'affectation est la suivante : i est affecté à la classe notée de centre c_i si et seulement si $d(i, c_i) < d(i, c_j)$ pour tout $j \neq i$.

Après avoir affecté tous les individus on obtient k classes notées de centres respectifs.

Etape 1 : En considérant les k classes obtenues à l'étape -0-, on calcule ses centres de gravité. On obtient donc k nouveaux centres notés. On utilise la même règle d'affectation qu'à l'étape -0-, on obtient k nouvelles classes de centres respectifs.

Etape N : On détermine k nouvelles classes en calculant les centres de gravité des classes obtenues à l'étape $(n-1)$. La règle d'affectation reste la même qu'à l'étape précédente et on obtient par la suite une nouvelle typologie de l'ensemble I : de centres respectifs

L'arrêt de l'algorithme

L'arrêt de l'algorithme de la méthode des "k-means" se fait

- Lorsque deux itérations successives conduisent à une même partition.
- Lorsqu'on fixe un critère d'arrêt tel que le nombre maximal d'itérations.

Les avantages et inconvénients :

Avantages :

1. La méthode résolve une tâche non supervisée, donc elle ne nécessite aucune information sur les données.
2. Technique facile à mettre en oeuvre.

3. La méthode est applicable à tout type de données (mêmes textuelles), en choisissant une bonne notion de distance.

Inconvénients

1. La difficulté de trouver une bonne fonction de distance.
2. Un bon choix du nombre k est nécessaire, un mauvais choix de k produit de mauvais résultats.
3. La difficulté d'expliquer certains clusters (i.e. attribuer une signification aux groupes constitués)

3.3 Les règles associatives :

3.3.1 Présentation :

Les règles associatives sont des règles extraites d'une base de données transactionnelles et qui décrivent des associations entre certains éléments. Elles sont fréquemment utilisées dans le secteur de la distribution des produits dont le principe est l'extraction d'associations entre produits sur les tickets de caisse. Le but de la méthode est l'étude de ce que les clients achètent pour obtenir des informations sur qui sont les clients et pourquoi ils font certains achats. La méthode recherche quels produits tendent à être achetés ensemble.

Une règle d'association est de la forme : Si condition alors résultat. Dans la pratique, nous nous limitons généralement à des règles où la condition se présente sous la forme d'une conjonction d'apparition d'articles et le résultat se constitue d'un seul article. Par exemple, une règle à trois articles sera de la forme : Si X et Y alors Z ; règle dont la sémantique peut-être énoncée : Si les articles X et Y apparaissent simultanément dans un achat alors l'article Z apparaît [16]

5.3.2. Avantages et inconvénients

Avantages

- Résultats clairs : règles faciles à interpréter.
- Simplicité de la méthode et des calculs (calculs élémentaires des fréquences d'apparition).
- Aucune hypothèse préalable (Apprentissage non supervisé).
- Méthode facile à adopter aux séries temporelles.

Inconvénients

- la méthode est coûteuse en temps de calcul.
- Qualité des règles : production d'un nombre important de règles triviales (des règles évidentes qui, par conséquent, n'apportent pas d'information) ou inutiles (des règles difficiles à interpréter provenant de particularités propres à la liste des achats ayant servi à l'apprentissage).
- Méthode non efficace pour les articles rares

3.3.3 Les algorithmes des règles associatives :

Il existe une multitude d’algorithmes qui construisent les règles d’association. La table 1.2 présente la majorité des ces algorithmes, leur développeurs, ainsi que l’année de publication. Nous présentons par la suite en détail un de ces algorithmes, qui est l’algorithme Apriori.

<i>Nom de l’algorithme</i>	<i>Développeur</i>	<i>Année</i>
APRIORI	Agrawal, et al.	1993
FP-GROWTH	Han, et al.	2000
ECLAT	Zaki	2000
SSDM	Escovar, et al.	2005
KDCI	Orlando, et al.	2003

Tab. 1.2 : Quelques algorithmes des règles d’association [16]

Algorithme a priori :

Apriori est un algorithme classique de recherche de règles d’association. Comme tous les algorithmes de découvertes d’associations, il travaille sur des bases de données transactionnelles (des enregistrements de transactions). Pour révéler la pertinence d’une règle on utilise deux concepts qui sont le support (12) et la confiance(13). Afin d’être retenue, chaque règle devra avoir un support supérieur à *minSupport* et une confiance supérieure à *minConf*. Ces deux valeurs étant définies empiriquement par l’utilisateur du système.

Support : $\#(a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_k)$

Confiance:
$$\frac{\#(a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_k)}{\#(a_1, a_2, \dots, a_n)}$$

Amiliorite:
$$\frac{\text{Confiance}(\#(a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_k))}{\#(a_1, a_2, \dots, a_n)}$$

L’algorithme démarre avec la liste des produits les plus fréquents dans la base de données respectant les seuils de support. Un ensemble de règles (des candidats) est généré à partir de cette liste. Les candidats ont testés sur la base de données (ie. on recherche les instances des règles générées et leurs occurrences) et les candidats ne respectant pas *minSupp* et *minConf* sont retirées. L’algorithme réitère ce processus en augmentant à chaque fois la dimension des candidats d’une unité tant que des règles pertinentes sont découvertes. A la fin, les ensembles de règles découvertes sont fusionnés.

La génération des candidats se fait en deux étapes : la jointure et l'élagage. La jointure consiste en une jointure d'un ensemble de règles à $k-1$ éléments sur lui-même qui aboutit à la génération d'un ensemble de candidats éléments. Enfin, l'élagage supprime les candidats dont au moins une des sous-chaînes à $k-1$ éléments n'est pas présente dans l'ensemble des règles à $k-1$ éléments.

Cet algorithme est très performant, mais souffre si les ensembles d'éléments fréquents sont trop grands. De plus, scanner la base de donnée à la recherche d'un motif de façon répétée devient rapidement un frein aux performances sur de grosses bases de données.[16]

3.4 L'estimation :

3.4.1 Principe :

L'estimation consiste à définir le lien entre un ensemble de prédicteurs et une variable cible. Ce lien est défini à partir de données « complètes », c'est-à-dire dont les valeurs sont connues tant pour les prédicteurs que pour la variable cible. Ensuite, on peut déduire une variable cible inconnue de la connaissance des prédicteurs.

À la différence de la segmentation (technique prédictive suivante) qui travaille sur une variable cible catégorielle, l'estimation travaille sur une variable cible numérique.

Intérêt :

- Permettre l'estimation de valeurs inconnues.

Méthodes :

- Analyse statistique classique : régression linéaire simple, corrélation, régression multiple, intervalle de confiance, estimation de points.

3.4.2 La régression linéaire

Régression linéaire multiple

La régression linéaire multiple permet de représenter la relation de dépendance entre une variable cible continue et des prédicteurs continus [21]. Cette relation est une connaissance extraite d'une base d'exemples composée de n observations supposées être indépendantes. Le modèle linéaire ou équation linéaire peut s'écrire sous forme matricielle comme suit :

Régression linéaire multiple

$$Y = X\beta + \varepsilon$$

Y : le vecteur colonne de dimension n contenant les observations de la variable cible, n représentant la taille de la base d'exemples.

X : la matrice de taille $n \times (p+1)$ composée d'une colonne de 1 suivies de p colonnes des vecteurs observations des variables explicative

p représente le nombre des variables explicatives

β : est le vecteur de dimension $(p+1)$ des paramètres à estimer. β_0 est l'ordonnée à l'origine qui représente la valeur de la variable cible Y quand les variables explicatives sont nulles. Les β_j sont les coefficients de régression pour estimer Y à partir de X . Chaque β_j représente le taux de changement en unité de Y par unité de changement dans la variable explicative X_j .

ε : vecteur de dimension n des erreurs (erreur standard).

La méthode la plus utilisée pour résoudre cette équation est la méthode des moindres carrés. Le principe des moindres carrés consiste à rechercher les valeurs des paramètres qui minimisent la somme des carrés des erreurs, à savoir :

$$S = \sum_{i=1}^n \varepsilon_i^2$$

En général, il n'y a pas de solution pour le système $Y = X\beta$ car il est surdéterminé, il comporte plus d'équations que d'inconnues. Un système d'équations équivalent appelé « les équations normales » peut être utilisé pour trouver la solution. Si $X'X$ est inversible alors ces équations normales admettent une solution :

$$\beta = (X'X)^{-1} (X'Y)$$

Lorsque le nombre des prédicteurs est limité à 1, nous avons une régression linéaire simple ;

$$Y = X\beta + \varepsilon = \beta_0 + \beta_1 X + \varepsilon$$

L'estimation de la régression linéaire est représentée par une droite linéaire Y . Pour chaque point de données du diagramme, une erreur ε est associée à la distance entre le point Y et la droite de régression (Voir Figure 1.5)

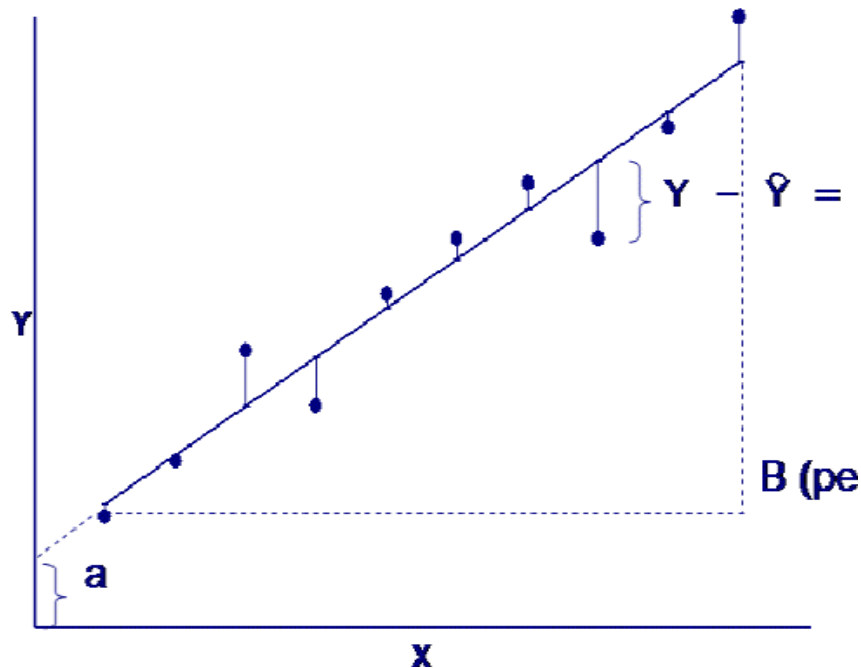


Fig. 1.5 : La droite de régression[21]

3.5 Segmentation (ou Classification) :

3.5.1.Principe :

La segmentation, dite aussi classification, est une estimation qui travaille sur une variable cible catégorielle. On parle de segmentation car chaque valeur possible pour la variable cible va définir un segment (ou type, ou classe, ou catégorie) de données. La segmentation peut être vue comme un clustering supervisé.

L'Intérêt de la segmentation est de permettre l'estimation de valeurs inconnues.

3.5.2 Clustering supervisé (Classification):

Le clustering supervisé est utilisé pour classer des données par rapport au partitionnement sûr qu'on appelle ensemble d'apprentissage.

Si nous posons $Y1$ l'ensemble d'apprentissage préalablement rangé en k classe. On souhaite classer un ensemble d'observations $Y2$ de la même manière qu' $Y1$. L'objectif recherché est d'automatiser un clustering à partir de cas déjà traités et validés.

Ce clustering fait intervenir la notion de probabilité puisque on classe une donnée en fonction de la probabilité qu'elle a d'être dans une classe.[17]

L'objectif du clustering supervisé est principalement de définir des règles permettant de classer des objets dans des classes à partir de variables qualitatives ou quantitatives caractérisant ces objets. [17]

Les méthodes de clustering supervisé sont nombreuses, nous en citons :

- Les arbres de décision
- Les Réseaux de neurone

- La méthode Support Vector Machine

Les arbres de décision :

Les arbres de décision sont des règles de clustering qui basent leurs décisions sur une suite de tests associée aux variables. Dans le cas d'une clustering supervisé, on en tire les informations afin de pouvoir classer nos nouvelles observations. Chaque nœud correspond à un "carrefour" où l'on effectue un test permettant de savoir le long de quelle branche nous allons descendre, une fois arrivée en bas, l'observation est classée. La Figure 1.6 donne un exemple d'arbre de décision.

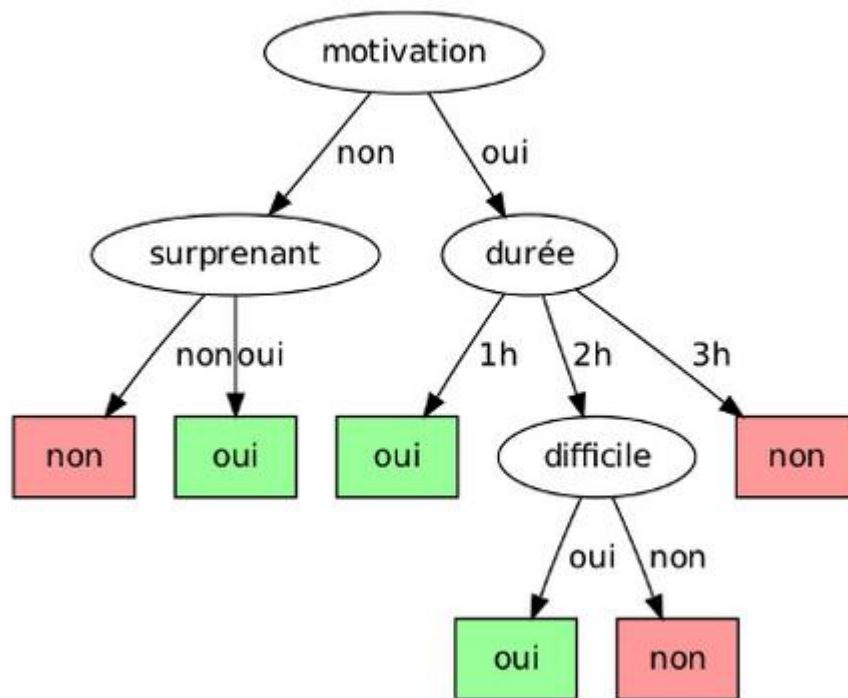


Fig.. 1.6: Exemple d'arbre de décision[24]

Construction de l'arbre de décision :

- La construction de cet arbre se fait de la façon suivante :
- On initialise l'arbre courant à vide, la racine de l'arbre est le nœud courant.
- On sélectionne un test permettant de mesurer l'impureté et on crée autant de nouveaux sous arbres qu'il y a de réponses possibles au test
- On passe au nœud suivant non étudié (s'il existe).
- Si un nœud est terminal (si l'homogénéité est parfaite), on lui affecte une classe.
- On recommence ces opérations jusqu'à ce qu'il n'y a plus de nœud sans classe.

Cette construction est la base de deux algorithmes de références dans la construction d'arbre de décision, CART et C4.5. La différence entre les deux, se situe au niveau du choix de test t qui fait le mieux progresser la discrimination des données de S : gain en information.

- Indice de Gini (CART),
- Critère d'entropie (C4.5),

L'indice de Gini est défini par :

$$gini(D) = 1 - \sum_{j=1}^n p_j^2$$

Où p_j est la fréquence relative de la casse j dans D

- Si un ensemble de données D est séparé selon A en deux sous-ensembles D_1 et D_2 , l'index de Gini, $gini(D)$ est défini par

$$gini_A(D) = \frac{|D_1|}{|D|} gini(D_1) + \frac{|D_2|}{|D|} gini(D_2)$$

- Réduction d'impureté:

$$\Delta gini(A) = gini(D) - gini_A(D)$$

- L'attribut qui donne le plus petit index de Gini, (D) (ou la plus grande réduction d'impureté) est choisi comme un nœud de séparation

Gain en Information (C4.5)

- Sélectionner l'attribut ayant le plus grand gain en information
- Soit p_i la probabilité qu'un tuple quelconque de D appartient à la classe C_i , estimée par $|C_i, D|/|D|$
- L'information attendue (entropie) nécessaire pour classer un tuple de D :

$$Info(D) = - \sum_{i=1}^m p_i \log_2(p_i)$$

- L'information nécessaire (après utilisation de A pour diviser D en v partitions) pour classer D :

$$Info_A(D) = \sum_{j=1}^v \frac{|D_j|}{|D|} \times I(D_j)$$

- L'Information gagnée (Gain en information) en branchant sur l'attribut A

$$Gain(A) = Info(D) - Info_A(D)$$

3.5.3 Clustering bayésien:

La clustering bayésien est un type de clustering probabiliste basée sur le théorème de Bayes avec une forte indépendance (dite naïve) des hypothèses. Elle est peu utilisée par les praticiens du datamining au détriment de méthodes plus traditionnelles comme les arbres de décision.

Un avantage de cette méthode est la simplicité de programmation, la facilité d'estimation des paramètres et sa rapidité (même sur de très grandes bases de données). Malgré ses avantages, son peu d'utilisation en pratique vient en partie du fait que, ne disposant pas d'un modèle explicite simple (l'explication de probabilité conditionnelle à priori), l'intérêt pratique d'une telle technique est remise en question

Ce clustering met en œuvre un classifieur bayésien naïf. Vulgairement, un classifieur bayésien naïf suppose que l'existence d'une caractéristique pour une classe est indépendante de l'existence d'autres caractéristiques. Par exemple, un restaurant peut être considéré comme gastronomique s'il a plusieurs étoiles sur le guide Michelin, s'il a à sa tête un grand chef étoilé et si ses prix sont très élevés. Ces caractéristiques sont liées dans la réalité mais ce type de classifieur déterminera que le restaurant est gastronomique en considérant indépendamment ces caractéristiques du nombre d'étoile, de la qualification du cuisinier et de gamme de prix. [17]

L'objectif de ce clustering est identique au clustering par arbre de décision, on souhaite classer une nouvelle observation avec la même logique qu'un ensemble préalablement observé et classifié.

Le théorème de Bayes (dans sa version probabiliste) est le suivant :

$$\mathbb{P}(A_i|B) = \frac{\mathbb{P}(B|A_i)\mathbb{P}(A_i)}{\sum_j \mathbb{P}(B|A_j)\mathbb{P}(A_j)}$$

On suppose maintenant que dans notre ensemble d'apprentissage on a n classes ($C1;C2,...,Cn$) et on considère X_{new} la nouvelle observation que l'on souhaite classifier.[3]

Pour mettre en place un classifieur naïf de Bayes :

1. On détermine un ensemble d'apprentissage
2. On détermine des probabilités à priori de chaque classe (par exemple en observant les effectifs)
3. On applique la règle de Bayes :

$$\mathbb{P}(C_i|X_{new}) = \frac{\mathbb{P}(X_{new}|C_i)\mathbb{P}(C_i)}{\sum_j \mathbb{P}(X_{new}|C_j)\mathbb{P}(C_j)}$$

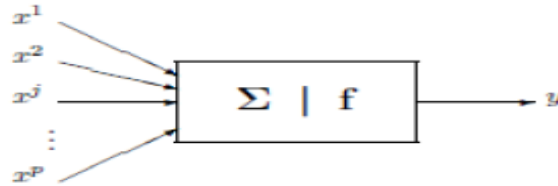
4. Pour obtenir la probabilité à posteriori des classes ($C1;C2,...,Cn$) au point X_{new} .
5. On choisit la classe la plus probable et on classe le point X_{new} dans cette classe.
6. On réitère le procédé pour tous les points à classer

3.5.4 Réseaux de neurone :

Cette méthode repose sur la notion de neurone formel. Un neurone formel est un modèle caractérisé par des signaux d'entrée (les variables explicatives par exemple), une fonction d'activation f .

$$f\left(\alpha_0 + \sum_i \alpha_i \times x_i\right)$$

f peut être linéaire, à seuil, stochastique et le plus souvent sigmoïde. Le calcul des paramètres se fait par apprentissage.



Les neurones sont ensuite associés en couche. Une couche d'entrée lit les signaux entrant, un neurone par entrée x_j , une couche en sortie fournit la réponse du système. Une ou plusieurs couches cachées participent au transfert. Un neurone d'une couche cachée est connecté en entrée à chacun des neurones de la couche précédente et en sortie à chaque neurone de la couche suivante.

De façon usuelle et en régression (Y quantitative), la dernière couche est constituée d'un seul neurone muni de la fonction d'activation identité tandis que les autres neurones (couche cachée) sont munis de la fonction sigmoïde.

En clustering binaire, le neurone de sortie est muni également de la fonction sigmoïde tandis que dans le cas d'une discrimination à m classes (Y qualitative), ce sont m neurones avec fonction sigmoïde, un par classe, qui sont considérés en sortie..

On minimise une fonction objective $Q(\alpha)$ (perte quadratique si Y est quantitative ou une fonction entropie en clustering). A partir des gradients de cette fonction, on utilise un algorithme d'optimisation.

Le modèle dépend de plusieurs paramètres :

- l'architecture du réseau : nombre de couches cachées (une ou deux en général) et le nombre de neurones par couche,
- le nombre d'itération, l'erreur maximale tolérée et un terme de régularisation (decay).

Les paramètres de réglage sont difficiles à définir correctement. On peut utiliser Library (e1071) par exemple pour rechercher les valeurs optimales

3.5.5 Support Vector Machine :

Les entrées X sont transformées en un vecteur dans un espace de Hilbert F . Dans le cas d'un classement en 2 classes, on détermine un hyperplan dans cet espace F . La solution optimale repose sur la propriété que les

objets sont les plus éloignés possibles de l'hyperplan, on maximise ainsi les marges (voir la figure illustrative 1.6).

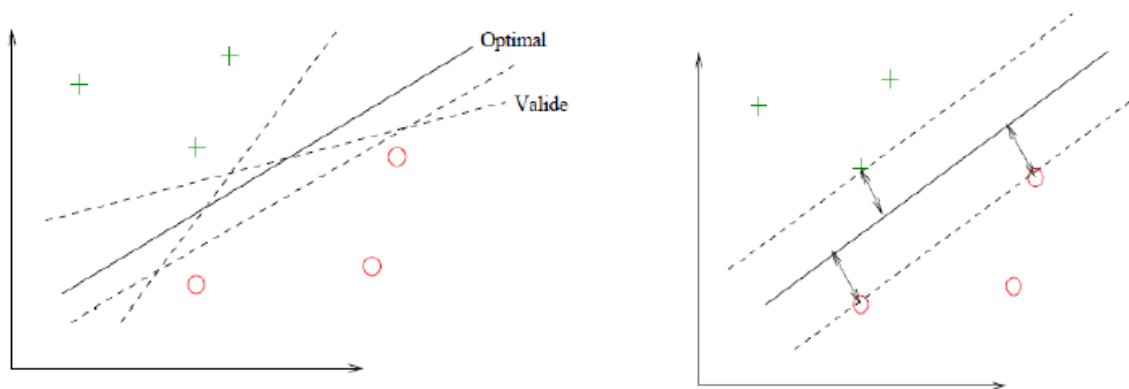


Fig. 1.7 :Support Vector Machine

Soit x le vecteur associé. On définit $f(x) = \omega x + \beta$ et l'hyperplan a pour équation $\omega x + \beta = 0$. La distance d'un point au plan est donnée par

$$d(x) = \frac{|\omega x + \beta|}{\|\omega\|}$$

Le classement est correct si $yf(x) > 0$ ou à un coefficient près $yf(x) \geq 1$.

Maximiser la marge revient à minimiser $\|w\|$ ou $\|w\|^2/2$ sous les contraintes $y_i f(x_i) \geq 1$. On utilise la méthode des multiplicateurs de Lagrange en ne conservant que les vecteurs x_i les plus proches de l'hyperplan (vecteurs supports). Lorsque tous les cas ne sont pas séparables, on introduit un terme d'erreur : $y_i f(x_i) \geq 1 - \xi_i$. La transformation en vecteur ne fait intervenir que l'expression du produit scalaire dans F . On recherche en fait directement l'expression du produit scalaire à partir des coordonnées initiales à l'aide d'une fonction k appelée noyau. On distingue les noyaux linéaire, polynômiaux, gaussien ...s'adaptant aux différentes problématiques rencontrées.[17]

4. Conclusion

Dans ce chapitre, un aperçu a été donné sur le processus ECD ainsi que ces étapes. Le data mining, une étape primordiale de l'ECD, est ensuite présenté, en détaillant la définition de presque toutes ses méthodes et techniques.

Cependant une étape d'importance similaire doit précéder l'application des algorithmes de data mining, c'est la préparation des données. Cette dernière joue un rôle essentiel dans la qualité des résultats du data mining. Le chapitre suivant sera consacré totalement aux méthodes et techniques de prétraitement.

CHAPITRE 2

METHODES DE PRETRAITEMENT

DES DONNEES

CHAPITRE 2

METHODES DE PRETRAITEMENT DES DONNEES

1. Introduction :

L'ECD est un processus complexe qui se déroule suivant une série d'opérations, dont principalement le prétraitement (cf. section 2.4 du chapitre 1), qui, après la collecte des données, doit être effectuée avec le plus de soin et de bonne pratique.

En effet, la mesure de la qualité des connaissances extraites du data mining est une étape clef qui a donné lieu à de nombreux travaux de recherche. Ces travaux ont montré que la qualité de connaissances est fortement liée à la qualité des données à partir desquelles sont extraites ces connaissances. [14]

Des étapes de prétraitement doivent avoir lieu avant le data mining en tant que tel. Le prétraitement porte sur l'accès aux données en vue de construire des données spécifiques d'une qualité telle que, traitées par les algorithmes de data mining, doivent donner une qualité de connaissance la meilleure.

Le prétraitement concerne la mise en forme des données entrées selon leur type (numériques, symboliques, images, textes, sons), ainsi que le nettoyage des données, le traitement des données manquantes, la sélection d'attributs ou la sélection d'instances. Cette première phase est cruciale car du choix des descripteurs et de la connaissance précise de la population va dépendre la mise au point des modèles de prédiction. L'information nécessaire à la construction d'un bon modèle de prévision peut être disponible dans les données mais un choix inapproprié de variables ou d'échantillon d'apprentissage peut faire échouer l'opération.

Malgré la quantité croissante de données disponibles et l'émergence du Big Data, les problématiques de données manquantes, intégration des données, fautes de frappe, .., restent très répandues dans les problèmes statistiques et nécessitent une approche particulière. Ignorer cette problématique peut entraîner, outre une perte de précision, de forts biais dans les modèles d'analyse.

Dans ce chapitre, nous allons présenter les méthodes de prétraitement pour avoir des données de bonne qualité. Nous commençons par rappeler quelques concepts fondamentaux des bases de données, ensuite nous abordons les étapes préliminaires de préparation des données avant d'introduire les différentes techniques de prétraitement.

2. Concepts fondamentaux des bases de données

2.1 Définition :

Une base de données (BD) est un ensemble structuré de données enregistrées avec le minimum de redondance pour satisfaire simultanément plusieurs utilisateurs de façon sélective en un temps opportun.

Une base de données est une entité dans laquelle il est possible de stocker des données de façon structurée et avec le moins de redondance possible. Ces données doivent pouvoir être utilisées par des programmes (oracle, SQL server), par des utilisateurs différents. La Figure 2.1 donne l'emplacement de la base de données et ses utilisateurs (clients).

Une base de données permet de mettre des données à la disposition d'utilisateurs pour une consultation, une saisie ou bien une mise à jour [1].

Une BD peut être locale, c'est-à-dire utilisable sur une machine par un utilisateur, ou bien répartie, c'est-à-dire que les informations sont stockées sur des machines distantes et accessibles par réseau.

Les bases de données sont gérées par des logiciels spécialisés appelés systèmes de gestion de bases de données (SGBD).

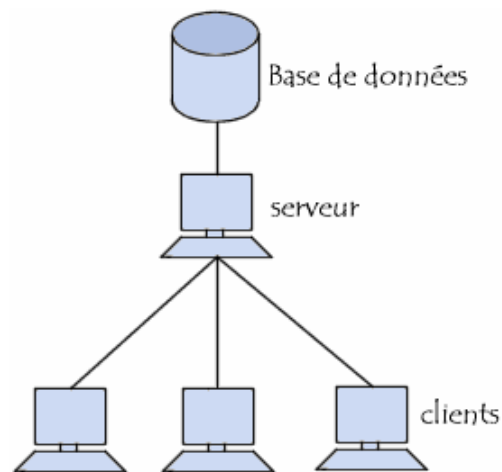


Fig. 2.1: Emplacement de la base de données et ses clients.[2]

2.2 Intérêt des bases de données :

L'intérêt d'une BD est de regrouper les données communes à une application dans le but :

- d'éviter les redondances et les incohérences qu'entraînerait fatalement une approche où les données seraient réparties dans différents fichiers sans connexions entre eux,
- d'offrir des langages de haut niveau pour la définition et la manipulation des données,
- de partager les données entre plusieurs utilisateurs,
- de contrôler l'intégrité, la sécurité et la confidentialité des données,
- d'assurer l'indépendance entre les données et les traitements.

2.3 Schéma conceptuel d'une base de données

Le schéma conceptuel est une représentation du monde réel auquel se rapporte la BD. Les principaux concepts du schéma conceptuel d'une base de données sont :

- L'entité (ou l'objet) : une personne, un livre
- La propriété (ou l'attribut) : le titre d'un livre, l'adresse d'une personne
- L'association : entre une personne et le livre dont elle est l'auteur
- L'agrégat : une adresse composée d'une rue et d'un code postal
- La collection : un ensemble de personnes, une liste de prénoms

Les principaux modèles conceptuels des bases de données sont : le modèle relationnel, entité-association, orienté-objet, ...etc. La Figure 2.2 montre un exemple d'une base de données en modèle relationnel.

■ Schéma

```
■ livre(cote: texte, titre: texte)
  auteur(nom: texte,
        prénom: texte,
        année_naissance: entier)
  écrire(cote: texte, nom: texte)
```

■ Extension

livre		auteur			écrire	
<u>cote</u>	titre	<u>nom</u>	prénom	année_naissance	<u>cote</u>	<u>nom</u>
BD/46	Les BD en BD	Dupont	Jean	1960	BD/46	Dupont
		Durand	Pierre	1953	BD/46	Durand

Fig. 2.2 : Exemple d'une base de données (bibliothèque) [22]

3. Préparation des données:

Il est très important de comprendre que le datamining n'est pas seulement le problème de découverte de modèles dans un ensemble de donnée. Ce n'est qu'une seule étape dans tout un processus suivi par les scientifiques, les ingénieurs ou toute autre personne qui cherche à extraire les connaissances à partir des données. En 1996 un groupe d'analystes définit le datamining comme étant un processus composé de cinq étapes sous le standard CRISP-DM (Cross-Industry Standard Process for Datamining) comme schématisé sur la Figure 2.3 [5]

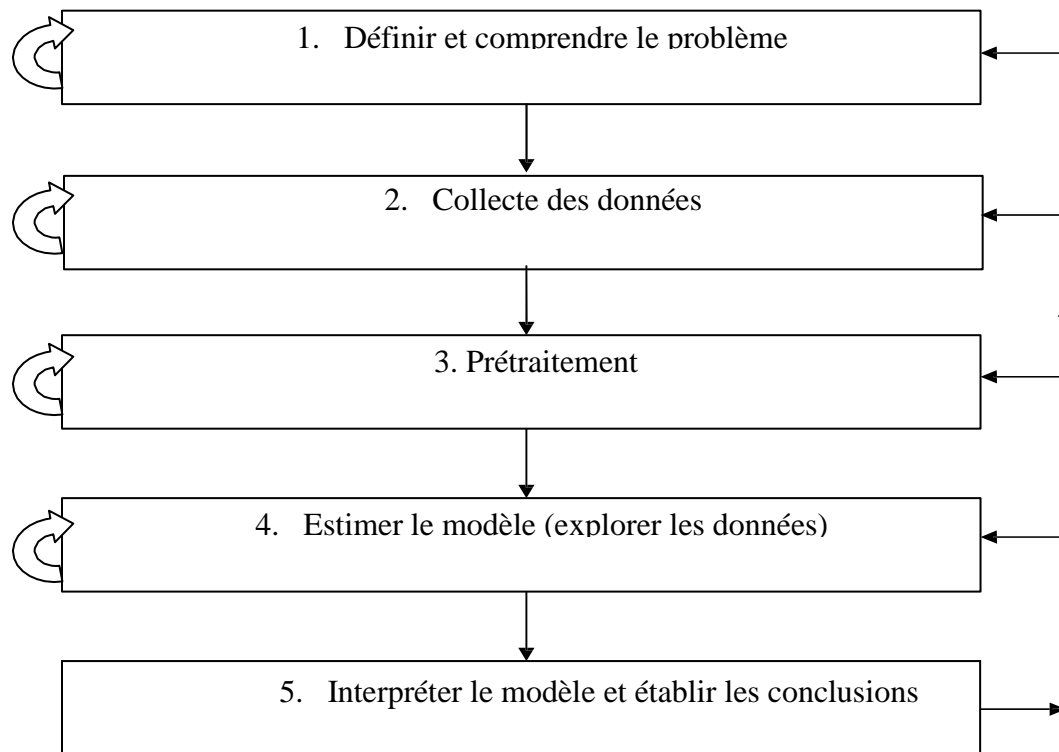


Fig. 2.3 : Processus de data minig [6]

Ce processus, composé de cinq étapes, n'est pas linéaire, on peut avoir besoin de revenir à des étapes précédentes pour corriger ou ajouter des données. Nous nous intéressons aux trois premières étapes de ce processus, qui peuvent être regroupées sous une seule appellation : Etape de préparation des données.

3.1 Définition et compréhension du problème :

Dans la plus part des cas, il est indispensable de comprendre la signification des données et le domaine à explorer. Sans cette compréhension, aucun algorithme ne va donner un résultat fiable. En effet, avec la compréhension du problème, on peut préparer les données nécessaires à l'exploration et interpréter correctement les résultats obtenus.

Généralement, le datamining est effectué dans un domaine particulier (banques, médecine, biologie, marketing, ...etc.) où la connaissance et l'expérience dans ce domaine jouent un rôle très important dans la définition du problème, l'orientation de l'exploration et l'explication des résultats obtenus. Une bonne compréhension du problème comporte une mesure des résultats de l'exploration, et éventuellement une justification de son coût. C'est-à-dire, pouvoir évaluer les résultats obtenus et convaincre l'utilisateur de leur rentabilité.[6]

3.2 Collecte des données :

Dans cette étape, on s'intéresse à la manière dont les données sont générées et collectées. D'après la définition du problème et des objectifs du datamining, on peut avoir une idée sur les données qui doivent être utilisées. Ces données n'ont pas toujours le même format et la même structure. On peut avoir des textes, des bases de données, des pages web, ...etc.

Parfois, on est amené à prendre une copie d'un système d'information en cours d'exécution, puis ramasser les données de sources éventuellement hétérogènes (fichiers, bases de données relationnelles, temporelles, ...).

Quelques traitements ne nécessitent qu'une partie des données, on doit alors sélectionner les données adéquates. Généralement les données sont subdivisées en deux parties : une utilisée pour construire un modèle et l'autre pour le tester. On prend par exemple une partie importante (suffisante pour l'analyse) des données (80 %) à partir de laquelle on construit un modèle qui prédit les données futures. Pour valider ce modèle, on le teste sur la partie restante (20 %) dont on connaît le comportement. [6]

3.3 Prétraitement :

Les données collectées doivent être "préparées". Avant tout, elles doivent être **nettoyées** puisqu'elles peuvent contenir plusieurs types d'anomalies : des données peuvent être bruité ou omises à cause des erreurs de frappe "par exemple ville=parais" ou à cause des erreurs dues au système lui-même, dans ce cas il faut remplacer ces données ou éliminer complètement leurs enregistrements. Des données peuvent être **incohérentes** c.-à-d. qui sortent des intervalles permis, on doit les écarter ou les normaliser. Parfois on est obligé de faire des transformations sur les données pour unifier leur poids. Un exemple de ces transformations est la normalisation des données qui consiste à la projection des données dans un intervalle bien précis [0,1] ou [0,100] par exemple.

Le prétraitement comporte aussi la réduction des données qui permet de réduire le nombre d'attributs pour accélérer les calculs et représenter les données sous un format optimal pour l'exploration. Une méthode largement utilisée dans ce contexte est l'analyse en composantes principales (**ACP**).

Une autre méthode de réduction est celle de la sélection et suppression des attributs dont l'importance dans la caractérisation des données est faible, en mesurant leurs variances. On peut même réduire le nombre de données utilisées par le datamining en écartant les moins importantes.

Dans la majorité des cas, le prétraitement doit préparer des informations globales sur les données pour les étapes qui suivent tel que la tendance centrale des données (moyenne, médiane, mode), le maximum et le minimum, le rang, les quartiles, la variance, ... etc.

Plusieurs techniques de visualisation des données telles que les courbes, les diagrammes, les graphes,... etc., peuvent aider à la sélection et le nettoyage des données. Une fois les données collectées, nettoyées et prétraitées on les appelle entrepôt de données (data warehouse). [6]

4. Nettoyage des données :

Le nettoyage de données est l'opération de détection et de correction ou suppression d'erreurs présentes sur des données stockées dans des [bases de données](#) ou dans des [fichiers](#). (voir Figure 2.4)

Les données présentes dans les bases de données peuvent avoir plusieurs types d'erreurs comme des erreurs de frappe, des informations manquantes, des données bruitées ou les données inconsistantes.

La partie impropre de la donnée traitée peut être remplacée, modifiée ou supprimée. Le processus de nettoyage identifie les données erronées et les corrige automatiquement avec un [programme informatique](#) ou les propose à un humain pour qu'il effectue les modifications.



Fig 2.4 : Résultat de nettoyage des données [7]

4.1 Etapes de nettoyage de données :

4.1.1 Analyse des données :

Une analyse approfondie des relations est nécessaire afin de déterminer quels genres de problèmes sont à corriger. Ceci signifie, en plus d'une inspection manuelle, un parcours automatisé des données pour extraire des méta-données qui peuvent servir de base pour établir des règles de correction et évaluer la qualité de la base.

4.1.2 Définition des flux de transformation :

Un nombre généralement élevé de transformations doivent être envisagé afin d'unifier les différentes sources de données en un seul schéma sur lequel les futures corrections peuvent être effectuées et qui sera intégré dans l'entrepôt. Les transformations préliminaires servent à nettoyer les problèmes de source unique puis les manipulations de détection de duplicates sont mises en œuvre.

4.1.3 Vérification :

L'efficacité du nettoyage doit être vérifiée, en générale sur un échantillon de données pour améliorer éventuellement les définitions précédentes. En effet, de manière itérative, plusieurs passages sur les données

permettent d'améliorer pas à pas la qualité du schéma unifié. De plus, certains problèmes sous-jacents n'apparaissent qu'après avoir corrigé ceux qui peuvent être mis en évidence dès le début.

4.1.4 Transformation :

Exécution des étapes de transformation afin de charger ou mettre à jour le contenu de la base de données (ou l'entrepôt de données).

4.1.5 Feedback des données nettoyées :

Après avoir effectué les étapes de nettoyage dans une base, il est nécessaire de rediriger ce flux et de recharger les données propres dans la base initiale afin d'assurer sa consistance et d'éviter de refaire les mêmes étapes à chaque mise à jour de l'entrepôt.

4.2 Les techniques de nettoyage de données :

4.2.1 Mesure de la qualité de la base :

Avec les techniques de profiling, il est possible d'avoir rapidement une vue globale du niveau de qualité de la base. Ce processus consiste à analyser chaque attribut des n-uplets dans sa globalité. Ainsi des indicateurs de qualité peuvent être déterminés tels que le type, la longueur, le domaine, la cardinalité, les valeurs discrètes et leur fréquence, l'unicité, les valeurs nulles d'un attribut. Ces méta-données révéleront donc un certain nombre de défaillance et fournissent de bons indicateurs. [10]

4.2.2 Détection des fautes de frappe :

Il est possible de les corriger avec une recherche dans un dictionnaire plus ou moins exhaustif mais il existe une autre solution dite fenêtre glissante *n-gram* ($n=1, 2, 3, \dots$ lettres) (*n-gram sliding window*). Une chaîne de n caractères est sélectionnée sur le mot vérifié. Ce n -gram est ensuite recherché dans la table des *n-grams* (avec des fréquences d'apparition associée). Si le mot comporte des n -grams très inhabituels voire non-existants, le mot est erroné.[10]

4.2.3 Valeurs manquantes

Les valeurs manquantes d'un attribut peut-être éventuellement déduites en analysant les dépendances de plusieurs attributs. Des méthodes statistiques ou des règles de production établies à partir des correspondances des attributs peuvent prédire donc ces valeurs.[10]

Les différents traitements des données manquantes :

1. ne rien faire
2. utiliser uniquement les enregistrements pour lesquels les données sont complètes
3. utiliser une méthode de repondération
4. imputer une valeur

1 : ne rien faire :

Cela oblige à travailler avec un fichier de données incomplet qui ressemble à un morceau de fromage gruyère (plein de trous !!).

2 : utiliser uniquement les enregistrements complets :

Si les données sont présentées sous forme de tableau, cela revient à oublier une ligne dès qu'il manque une valeur dans cette ligne : on oublie donc aussi les autres valeurs de cette ligne, qui sont effectivement présentes. Bien que cette option soit simple et permette d'utiliser un fichier complet, elle présente certains risques. En effet :

1. l'échantillon de ceux qui ont répondu à toutes les questions peut être
 - soit trop réduit pour être significatif
 - soit non représentatif de la population globale.
2. elle peut mener à des estimateurs fortement biaisés, à moins que la non-réponse ne dépende d'aucune des variables d'intérêts.

3 : utiliser une méthode de repondération :

- **Non-réponse totale:** Les méthodes de repondération augmentent le poids de sondage appliqué aux répondants pour compenser pour les non-répondants. L'objectif est de produire des estimations approximativement sans biais.
- **Non-réponse partielle:** On peut appliquer des méthodes de repondération mais le principal inconvénient est qu'il faut créer un nouveau poids ajusté pour chaque variable d'intérêt.

4 : utiliser l'imputation :

L'imputation consiste à produire une « valeur artificielle » pour remplacer la valeur manquante, avec pour objectif de produire des estimations approximativement sans biais. L'imputation par règle consiste à appliquer à une valeur manquante une valeur déterminée suivant une réglementation : Exemple : Calcul montant TTC à partir du montant HT [11].

On distingue deux classes de méthodes d'imputation (Voir table 2.1)

1. Méthodes déterministes :
 - la moyenne
 - le ratio
 - la régression
2. Méthodes stochastiques et aléatoires :
 - le hot-deck aléatoire
 - le plus proche voisin

Méthode d'imputation	Moyenne	Ratio	Régression	Hot-deck aléatoire	Plus proche voisin
Valeur imputée	$y_i^* = \frac{1}{r} \sum_{j \in s_r} y_j = \bar{y}_r$	$y_i^* = \frac{\bar{y}_r}{\bar{x}_r} x_i$	$y_i^* = \hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \dots + \hat{\beta}_q x_{iq}$	$y_i^* = y_j$ pour certains $j \in s_r$ tels que $P(y_i^* = y_j) = 1/r$	$y_i^* = y_j$ pour certains $j \in s_r$ tels que $dist(x_i, x_j)$ soit minimal
Variable(s) auxiliaire(s)?	NON	OUI (une)	OUI (une ou plus)	NON	OUI (une ou plus)

Tab. 2.1: Méthodes d'imputation [11]

Description des méthodes d'imputation :

Imputation par la moyenne : On remplace chacune des valeurs manquantes par la valeur moyenne de l'ensemble de réponses obtenues.

imputation par le ratio : chaque valeur manquante y_i est remplacée par la valeur prévue \hat{y}_i obtenue par régression de y sur x .

imputation par régression : C'est une extension naturelle de l'imputation par la méthode du ratio où l'on se sert de q variables auxiliaires $x_1 \dots x_q$.

imputation par la méthode hot-deck aléatoire : Cela consiste à attribuer la valeur de y fournie par un répondant (donneur), sélectionné au hasard avec remise parmi l'ensemble des répondants, pour remplacer la valeur manquante pour l'unité non-répondante (receveur).

imputation par la méthode par le plus proche voisin : On attribue à l'enregistrement pour lequel la réponse à une question manque la valeur figurant pour cette question dans l'enregistrement obtenu pour le répondant le plus proche, où l'expression « le plus proche » est habituellement définie par une fonction de distance basée sur une ou plusieurs variables auxiliaires.[11]

4.2.4 Valeurs saillantes

La détection ces valeurs consiste en particulier à parcourir les valeurs d'un attribut dans la base et établir les limites $\pm 3\sigma$. Des règles de production sont établies à partir des correspondances des attributs pour confirmer l'éventuelle correction/élimination de ces valeurs. [10]

4.2.5 Codes inconsistants

Comme les codes utilisés pour un attribut sont en général un sous-ensemble restreint, une table de hachage est utilisée pour vérifier leur existence. Par exemple, les codes postaux sont vérifiés avec une table de hachage. [10]

4.2.6 Extraction des valeurs concaténées d'un attribut

Il faut procéder à un ordonnancement dû aux transpositions et au découpage de l'attribut pour extraire les informations nécessaires. Les attributs les plus souvent affectés par cette opération sont les champs d'adresse et de nom.[10]

4.2.7 *Conflits de typage et de nommage*

Quand différents types d'attribut représentent le même concept, un type commun intégrera toutes les valeurs de ces attributs que l'on obtient par des étapes de transformation et standardisation. Les conflits de nommage sont résolus par des changements de nom.

4.2.8 *Elimination des redondances (minimalité et complétude)*

L'objectif de la minimalité signifie la détection et l'élimination des données redondantes ainsi il s'oppose en partie à l'objectif de la complétude qui cherche à intégrer le plus d'informations utiles possibles dans l'entrepôt de données.

En effet, l'élimination des duplicats d'un schéma demeure un des problèmes les plus complexes qui a donné suite à de nombreuses recherches scientifiques. L'élimination se décompose en 3 phases :

- identification des duplicats
- fusion des attributs des duplicats en un n-uplets en gardant uniquement les attributs non-redondants
- suppression des duplicats

Dans le cas le plus simple, il y a un attribut ou une combinaison d'attributs par n-uplet qui peut être utilisé pour apparier les enregistrements, par exemple, si les différentes sources partagent la même clef primaire ou s'il y a d'autres attributs communs uniques. Dans ce cas, un tri sur la clef ou sur ces les attributs concernés permet de trouver les duplicats dans différentes bases par une opération de jointure naturelle. Dans une base unique, il suffit d'effectuer un tri sur les attributs précédents et analyser si le voisinage d'un n-uplet est redondant.

5. Intégration de données :

L'intégration est la combinaison des données provenant de plusieurs sources (base de données, sources externes, etc.). Le but d'opérations est de générer des bases de données et/ou des entrepôts de données spécialisés contenant les données retravaillées pour faciliter leurs exploitations futures. La Figure 2.5 donne quelques exemples d'intégration de données.

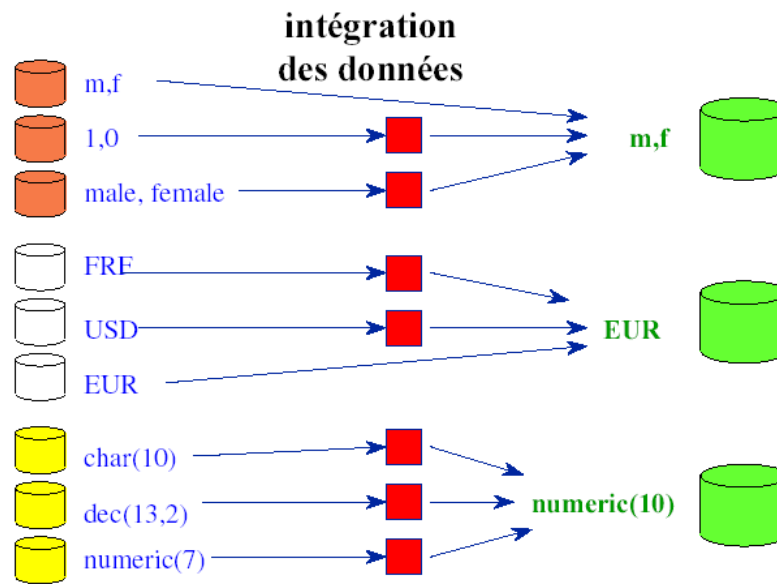


Fig. 2.5 : Exemples d'intégration de données [12]

6. Transformation de données :

Lissage de données : utilisation de technique de régression.

Normalisation des données : normaliser certains Attributs numériques afin qu'ils varient entre 0 et 1. Pour ne pas privilégier les attributs ayant les plus grands domaines de variation (salaire/Âge).

Agrégation des données : opérations OLAP (On-LineAnalytical Processing) permettant une analyse multidimensionnelle sur les BD volumineuses afin d'émettre en évidence une analyse particulière des données. Calculer les niveaux de ventes réalisées de tel produit par mois plutôt que par jour.

Généralisation des données : remplacer les données finies par des données de plus haut niveau.

- Remplacer les adresses précisées des clients par leur code postal
- Remplacer l'âge des clients par « jeune », « adulte », « senior »

7. Sélection des données :

L'objectif de cette opération est de garder uniquement les données pertinentes pour l'étude à réaliser [10].

8. Conclusion :

Dans ce chapitre, nous avons présenté la majorité des techniques de prétraitement des données. Ces techniques vont permettre certainement d'offrir une bonne qualité des données.

Le chapitre suivant sera consacré à la description du système de nettoyage des données que nous avons conçu et implémenté.

CHAPITRE 3

SYSTEME DE NETTOYAGE DES DONNEES

CHAPITRE 3

SYSTEME DE NETTOYAGE DES DONNEES

1. Introduction

Dans les chapitres précédents, nous avons exploré les différentes étapes du processus ECD, en présentant en détail, les méthodes de prétraitement dans l'objectif d'améliorer la qualité des données pour pouvoir en extraire des connaissances pertinentes.

Dans ce chapitre, il est question de décrire le système de nettoyage que nous avons implémenté. Nous commençons par présenter les outils de développement de notre système, avant de procéder à la description des ses différents composants.

2. Le langage de programmation et de développement c#

2.1 Présentation de c#

C# est un langage récent. Il a été disponible en versions beta successives depuis l'année 2000 avant d'être officiellement disponible en février 2002 en même temps que la plate-forme .NET 1.0 de Microsoft à laquelle il est lié. C# ne peut fonctionner qu'avec cet environnement d'exécution. Celui-ci rend disponible aux programmes qui s'exécutent en son sein un ensemble très important de classes. En première approximation, on peut dire que la plate-forme .NET est un environnement d'exécution analogue à une machine virtuelle Java. On peut noter cependant deux différences :

- Java s'exécute sur différents OS (windows , unix, macintosh) depuis ses débuts. En 2002, la plate-forme .NET ne s'exécutait que sur les machines Windows. Depuis quelques années le projet Mono [<http://www.mono-project.com>] permet d'utiliser la plate-forme .NET sur des OS tels que Unix et Linux. La version actuelle de Mono (février 2008) supporte .NET 1.1 et des éléments de .NET 2.0.
- la plate-forme .NET permet l'exécution de programmes écrits en différents langages. Il suffit que le compilateur de ceux-ci sache produire du code IL (Intermediate Language), code exécuté par la machine virtuelle .NET. Toutes les classes de .NET sont disponibles aux langages compatibles .NET ce qui tend à gommer les différences entre langages dans la mesure où les programmes utilisent largement ces classes. Le choix d'un langage .NET devient affaire de goût plus que de performances.

2.2 Composants élémentaires du C#

Un programme en langage C# est constitué des six groupes de composants élémentaires suivants :

- les identificateurs,
- les mots-clefs : sont réservés pour le langage lui-même et ne peuvent pas être utilisés comme identificateurs.
- les constantes : Une constante est une valeur qui apparaît littéralement dans le code source d'un programme,
- les chaînes de caractères,
- les opérateurs,
- les signes de ponctuation.

On peut ajouter à ces six groupes les commentaires, qui sont ignorés par le préprocesseur.

2.3 Visual Studio c#:

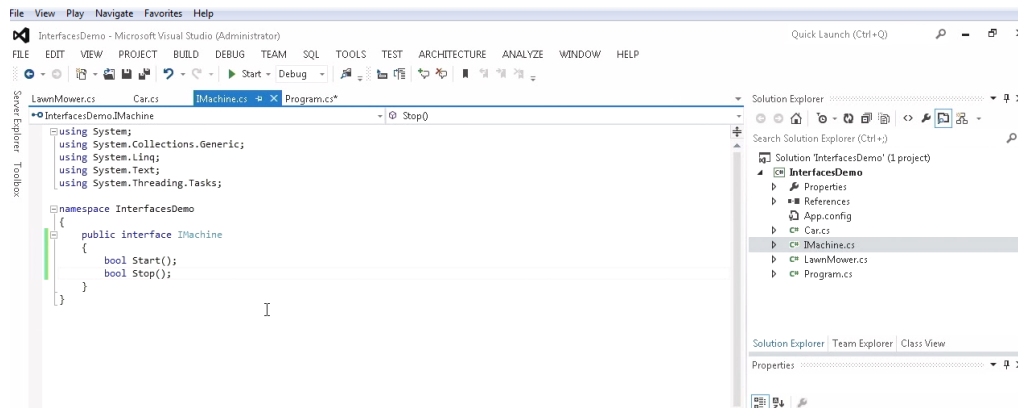


Fig. 3.1: Interface visual studio c#

3. Présentation des données du système de nettoyage:

Notre système de nettoyage des données travaille sur un extrait d'une base de données issue d'une agence de location de voitures. Les données sélectionnées ont été intégrées dans une table unique pour un meilleur traitement.

La structure de la table créée est composée de cinq attributs qui sont :

- *nom*, *prénom* de type string, qui désignent les noms et prénoms des locataires de véhicules
- *âge* de type entier qui désigne l'âge du locataire
- *fidélité* de type string, qui désigne le degré de fidélité du locataire
- *région* de type string, qui désigne la région d'où appartient le locataire
- *clas* de type string, cet attribut est l'étiquette de classe à laquelle appartient le locataire, qui désigne si l'agence peut avoir confiance à cette personne (ou société) et lui louer le véhicule ou non. Cette information est généralement donnée par l'agence selon l'historique des ses différents locataires.

Voici un morceau de programme qui déclare la structure de la table des données.

```
struct base
{
    string nom;
    string prenom;
    int age
    string fidelite ;
    string region;
    string clas ;
};
```

Les deux figures 3.2 et 3.3 donnent respectivement la structure ainsi qu'un extrait des données de la table.

Nom De Colonne	Type De Données	Valeur Nullable	Valeur Par Défaut	Clé Primaire
NOM	VARCHAR2(4000)	Yes	-	-
PRENOM	VARCHAR2(4000)	Yes	-	-
REGION	VARCHAR2(4000)	Yes	-	-
FIDELITE	VARCHAR2(4000)	Yes	-	-
CLAS	VARCHAR2(4000)	Yes	-	-
AGE	NUMBER	Yes	-	-
1 - 6				

Fig. 3.2 : Description de structure de la table des données

NOM	PRENOM	REGION	FIDELITE	CLAS	AGE
afaf	benslimane	boussada	no	yes	5
inas	inas	boussada	no	no	12
oussama	oussama	msila	yes	yes	-
ahlem	barka	msif	yes	yes	23
kanza	layadi	bordj	no	no	23
meriem	meriem	ain meleh	yes	no	25
yasmine	yesmine	hamam dal3a	no	yes	18
yasmine	yesmine	hamam dal3a	no	yes	18
hanaa	benslimane	boussada	yes	yes	5

Fig. 3.3: Un extrait des données (non traitées) de la table

4. Description des fonctionnalités du système :

4.1 Description générale du système :

Le système de nettoyage des données que nous avons implémenté applique certaines méthodes de prétraitement sur les données afin de les rendre de meilleure qualité pour le data mining.

Nous avons sélectionné trois types de prétraitement auxquels nous avons implémenté des algorithmes pour leur mise en œuvre. Ces types de prétraitement sont : traitement des valeurs manquantes, élimination des mots étrangers et traitement de l'incohérence des données.

4.2 Traitement des valeurs manquantes :

Pour traiter les valeurs manquantes d'une base de données, nous avons choisi d'implémenter deux algorithmes appartenant aux méthodes déterministes : par moyenne et par suppression, ainsi qu'un algorithme de type stochastique appliquant la notion de distance.

4.2.1 Traitement des valeurs manquantes par moyenne :

Le principe du traitement des valeurs manquantes par la moyenne est très simple, il permet de calculer la moyenne de l'attribut de la base de données où se trouve la valeur manquante, ensuite remplir cette dernière par la moyenne calculée. La Figure 3.4 donne l'organigramme de fonctionnement de cette méthode.

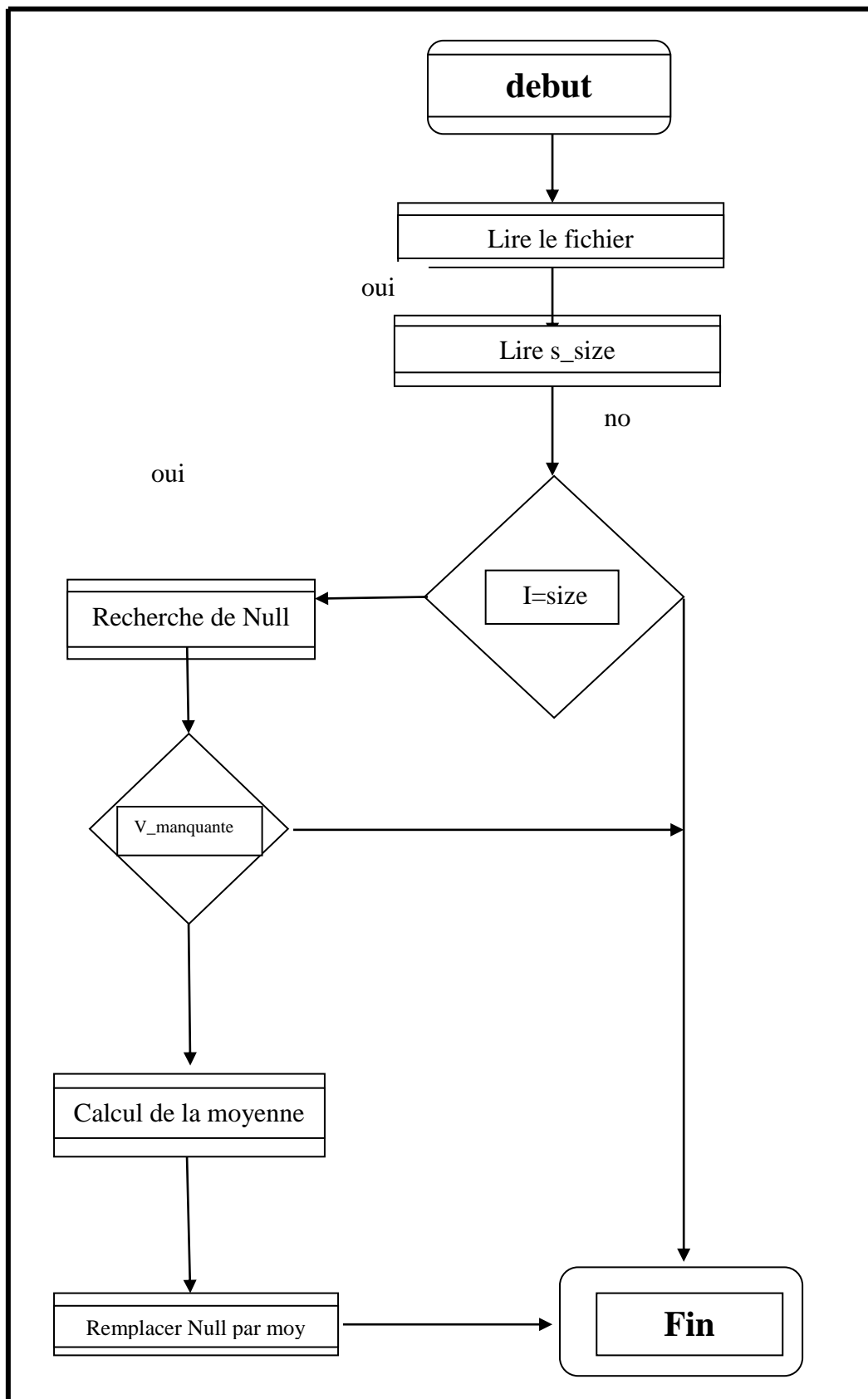


Fig. 3.4 : Organigramme de traitement des valeurs manquante par moyenne

Les Figures 3.5 et 3.6 montrent respectivement un exemple d'une valeur manquante, à savoir l'âge d'un locataire et le remplissage de cette valeur après application de la méthode de traitement par moyenne.

NOM	PRENOM	REGION	FIDELITE	CLAS	AGE
afaf	benslimane	boussada	yes	yes	-

Fig. 3.5 : Exemple de donnée manquante de l'attribut âge

NOM	PRENOM	REGION	FIDELITE	CLAS	AGE
afaf	benslimane	boussada	yes	yes	26

Fig. 3.6: Etat de la base après remplissage de la valeur manquante

4.2.2 Traitement des valeurs manquantes par suppression :

Le principe du traitement des valeurs manquantes par suppression permet tout simplement de supprimer tout l'enregistrement où se trouve la valeur manquante. Cette méthode peut s'avérer efficace dans certains domaines et pour certaines données volumineuses où la suppression d'une partie de la base de données (enregistrement dans les quels des valeurs manquantes existent) n'a pas d'influence sur les résultats de la fouille des données. La Figure 3.7 présente l'organigramme de cette méthode.

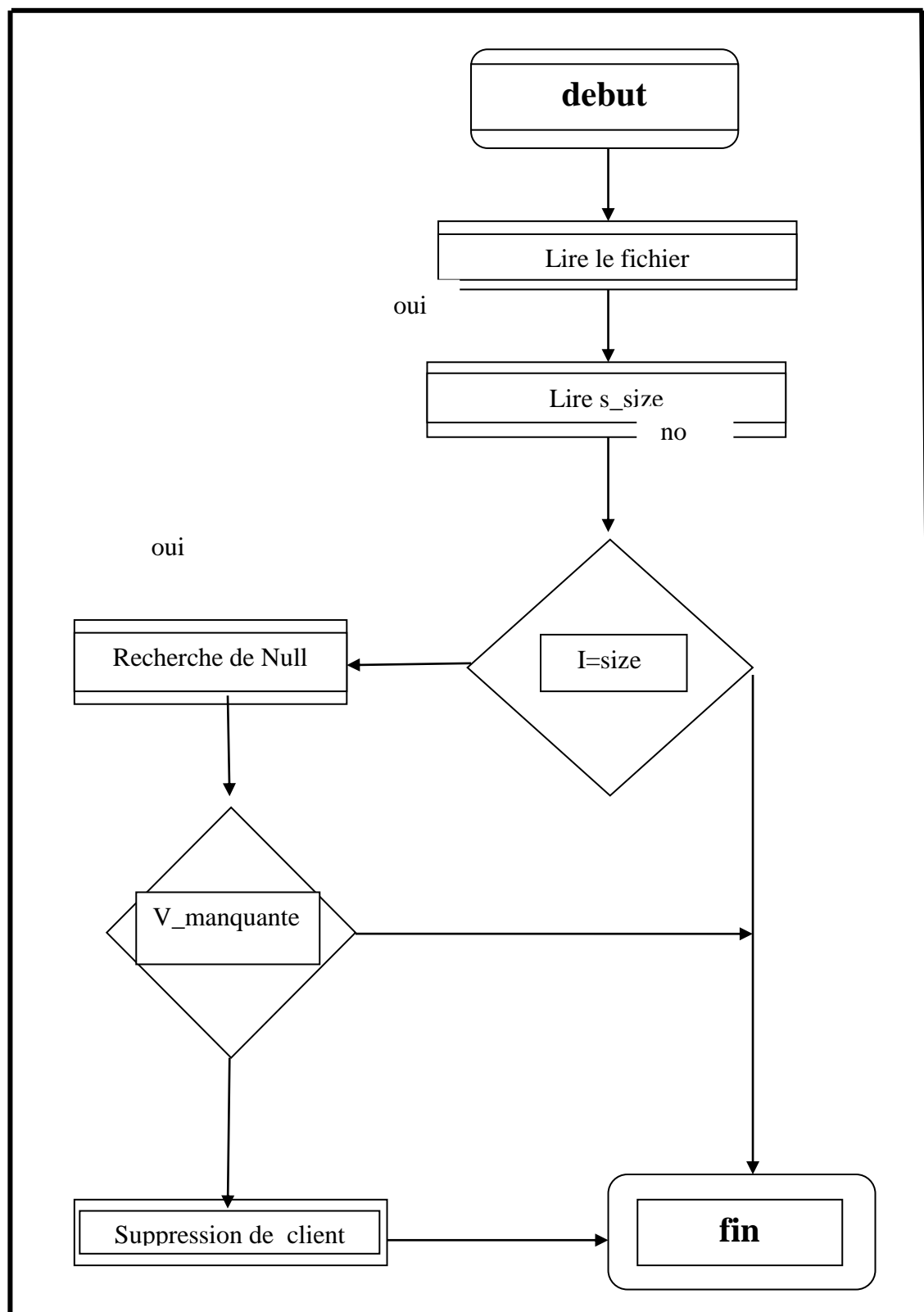


Fig. 3.7 : Organigramme de traitement des valeurs manquantes par suppression

4.2.3 Traitement des valeurs manquantes par régression

La technique de traitement des valeurs manquantes par régression consiste à appliquer la chercher un modèle de régression sur l'ensemble des données ensuite remplir la valeur manquante en appliquant le modèle trouvé. On peut trouver plusieurs modèles de régression dans la littérature : régression linéaire, régression multiple, etc. Par ailleurs, il faut mentionner que cette méthode ne peut s'appliquer que si les valeurs manquantes sont numériques.

Dans notre système, nous avons appliqué la régression linéaire, qui consiste à déterminer un modèle linéaire (droite pour deux variables, plan pour trois variables, etc).

L'exemple que nous donnons consiste à trouver le système linéaire qui lie les trois variables : âge du locataire (*Age*), durée de location (*Duree*) et montant d'entretien après retour (*Entret*). Si nous supposons que la valeur manquante appartient à l'attribut *Entret*, le système linéaire de la régression aura la forme suivante :

$$Entret = \alpha + \beta_1.Age + \beta_2.Duree$$

Les étapes de cette méthode sont :

- 1) Transformer la base (les données des trois attributs sus-mentionnés) en une matrice (p,p)
- 2) Résoudre le système linéaire de taille p.

Après résolution de l'équation pour trouver les coefficients α , β_1 et β_2 , les valeurs manquantes de l'attribut *Entret* sont trouvées par l'équation.

4.3 Elimination des mots étrangers:

L'élimination des mots étrangers dans une base de données est une opération semi- automatique. Elle consiste à demander à l'utilisateur de signaler le mot étranger parmi la liste des différentes valeurs de l'attribut de la base de données. Une fois, le mot étranger localisé, le système de nettoyage peut appliquer l'une des deux méthodes suivantes :

- La méthode aléatoire : le principe de cette méthode est de remplacer le mot étranger par un autre mot de la liste d'une façon aléatoire.
- La méthode du plus courant : le principe de cette méthode est de remplacer le mot étranger par le mot le plus courant (ayant la plus grande fréquence) de la liste des mots qui existent dans la base.

La Figure 3.8 donne l'organigramme de la méthode d'élimination des mots étrangers.

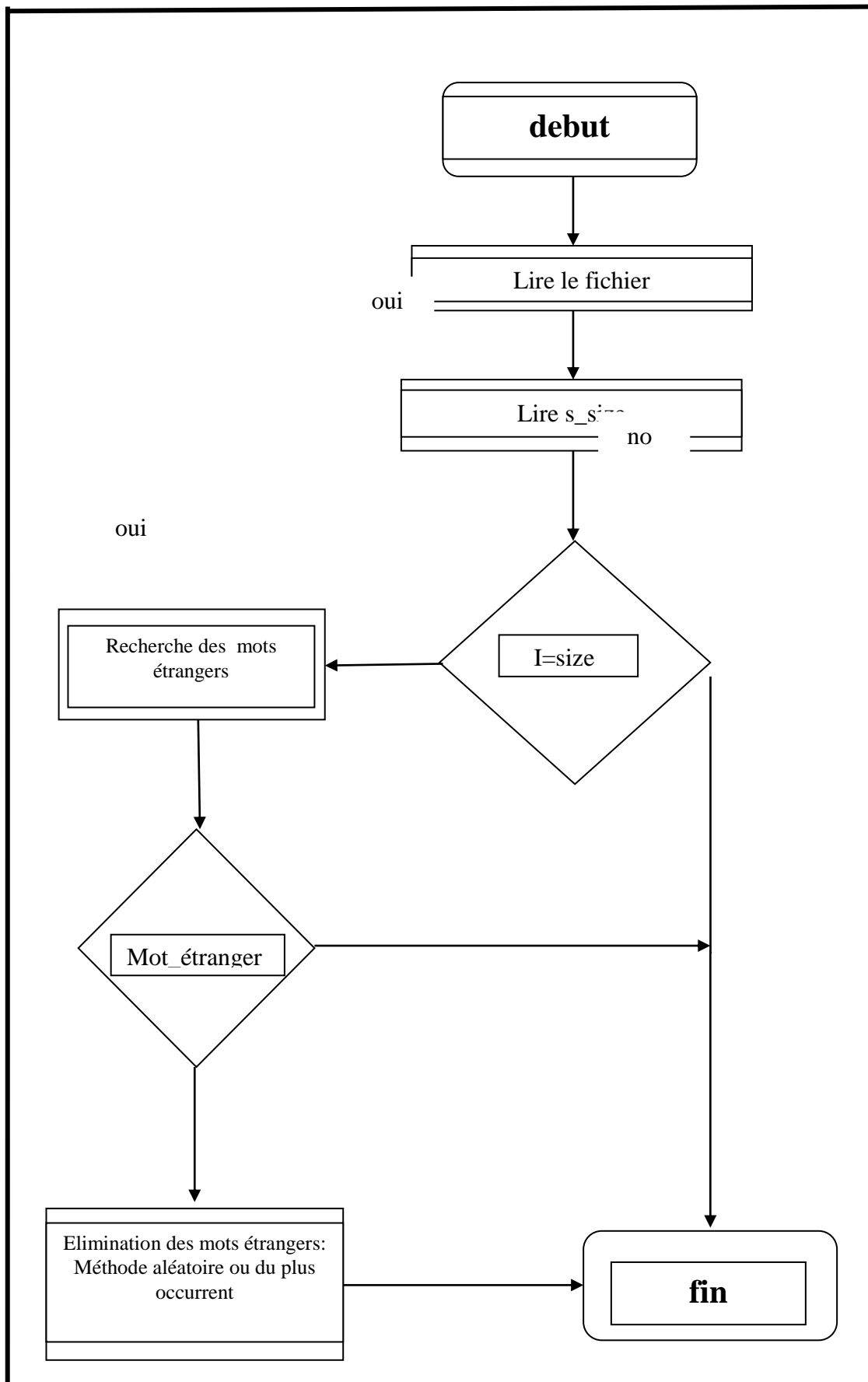


Fig 3.8 : Organigramme de la méthode d'élimination des mots étrangers

4.4. traitement de l'incohérence :

cette fonction détecter le nombre d'incohérence et me donne le nom et prenom de client et aussi corrige la , a l'aide d'un expert nous concluons l'incohérence en 3 type suivant :

1. Incohérences au niveau de l'âge si on trouve quelqu'un ayant zero ou inferieur de 0.
2. Incohérence au niveau de nom et prénom si on trouve deux identique.
3. Incohérences entre âge et fidélité chaque personne ayant un âge inferieur de 18 son fidélité sera automatique no

Chaque type d'incohérence a un traitement spécial :

1. Le premier type sera traite en donnant a l'âge la valeur nulle et sera considéré comme une valeur manquante.
2. Le deuxième type sera directement supprimé.
3. Le troisième type sera traite automatique en donne la valeur NO a fidelité

La Figure 3.9 donne l'organigramme de la méthode d'élimination des mots étranger

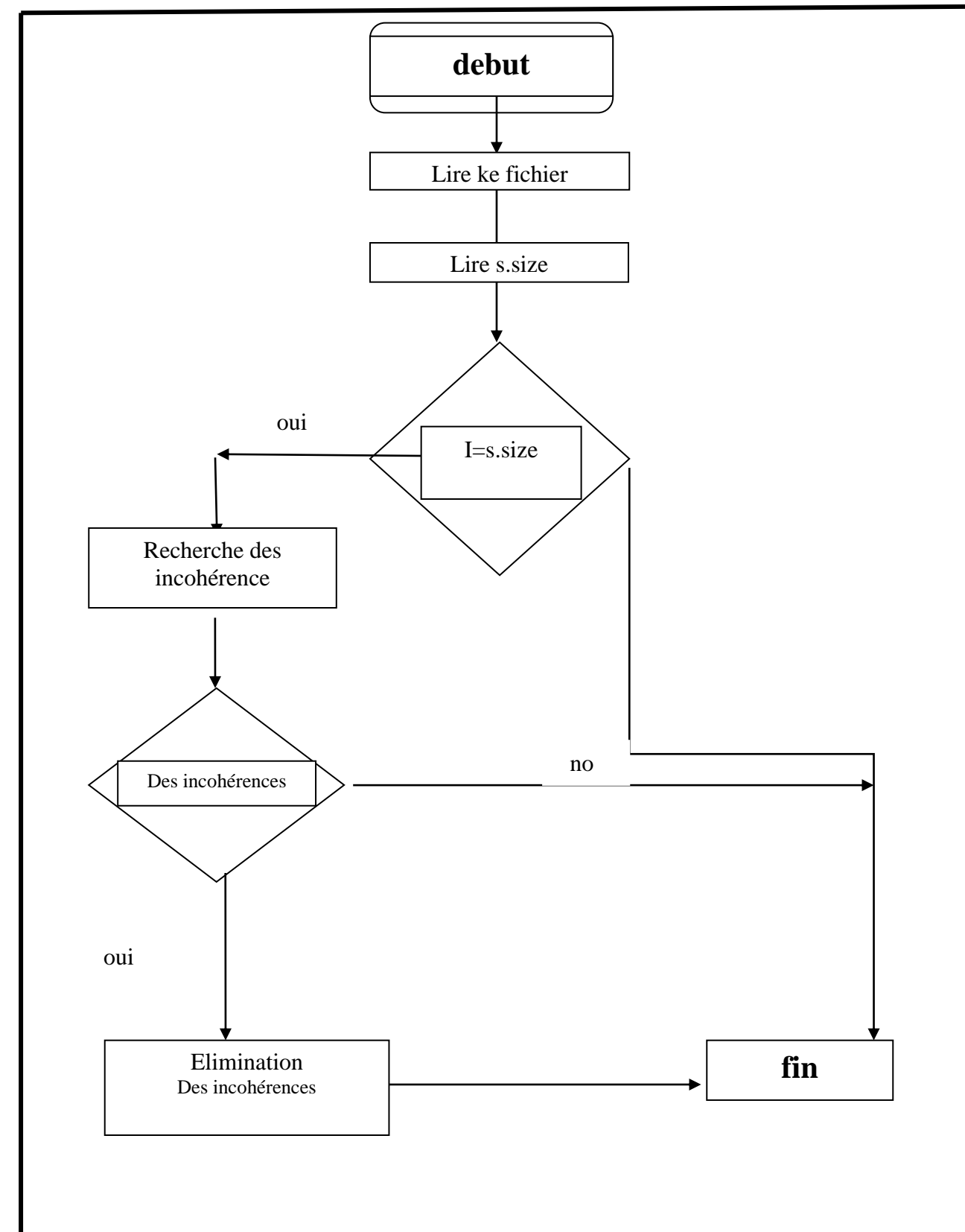


Fig 3.9 : Organigramme de la méthode d'élimination de l'incohérence

5. Evaluation du système de nettoyage

L'objectif de notre système de nettoyage est d'avoir des données plus propres pour pouvoir, lorsqu'une méthode de data mining est appliquée, d'obtenir une meilleure qualité des connaissances.

Pour cela, nous avons procédé à l'évaluation de l'ensemble des méthodes de nettoyage implémenté dans le système, en appliquant une méthode de data mining sur les données de l'agence de location avant prétraitement et après prétraitement par le système. Le but de cette évaluation est de comparer les résultats obtenus pour montrer que la qualité des données influe largement sur le résultat du data mining.

Comme les méthodes de data mining sont nombreuses, nous avons choisi la méthode de classification dite : bayésienne naïve. Cette méthode est appliquée sur les données brutes (avant prétraitement) ensuite sur les données propres (après nettoyage effectué par notre système).

Les résultats obtenus après plusieurs comparaisons montrent clairement que notre système de nettoyage contribue largement aux nettoyages des données brutes de l'agence de location, ceci pour obtenir un taux = 3/5 des résultats de classification efficaces, et que la qualité des données joue un rôle très important dans le processus de l'ECD.

6. interface du système de nettoyage:



Fig.3.10: Interface du système de nettoyage

7. conclusion

Dans ce chapitre, il a été question de décrire les fonctionnalités du système de nettoyage que nous avons implémenté. Nous avons présenté, en premier lieu la structure des données de notre application qui sont issues d'une agence de location de véhicules. Ensuite, les différentes méthodes de prétraitements ont été présentées. Trois types de traitement ont été implémentés avec plusieurs variantes, à savoir : traitement des valeurs manquantes, élimination des mots étrangers et traitement de l'incohérence des données.

Pour évaluer la qualité des données obtenues après nettoyage, nous avons appliqué une méthode de data mining : la classification bayésienne, sur les données avant prétraitement ensuite après, et comparer les résultats obtenus dans les deux cas. L'objectif était de montrer que la qualité des données influe largement sur la qualité des connaissances obtenues après la fouille de données.

CONCLUSION GENERALE

Il va sans dire que la qualité d'un travail est l'objectif de tous. La qualité du résultat obtenu d'un algorithme est sans aucun doute la plus chère des recommandations de n'importe quel informaticien.

Notre travail s'est focalisé sur la qualité des données en vue d'obtenir une bonne qualité des résultats rendus d'un algorithme de fouille de données. Pour cela, nous nous sommes intéressés aux méthodes de prétraitement des données. Nous avons implémenté un certain nombre de ces méthodes et les ont appliquées sur un extrait de base de données d'une agence de location des véhicules.

Toutefois, avant d'aboutir à l'implémentation du système, nous avons présenté le processus ECD et les différentes méthodes de data mining, ceci pour montrer l'intérêt d'avoir une bonne qualité des données. Nous avons ensuite décrit les différentes étapes de prétraitement des données ainsi que la majorité des méthodes et techniques y afférentes.

Dans notre système de nettoyage, nous avons sélectionné trois types de prétraitement auxquels nous avons implémenté des algorithmes pour leur mise en œuvre. Ces types de prétraitement sont : traitement des valeurs manquantes, élimination des mots étrangers et traitement de l'incohérence des données.

Comme l'objectif de notre système de nettoyage est d'avoir des données plus propres pour avoir les meilleurs résultats de fouille de données, nous avons procédé à l'évaluation de l'ensemble des méthodes de nettoyage implémenté dans le système, en appliquant une méthode de data mining (classification bayésienne naïve) sur les données de l'agence de location avant prétraitement et après prétraitement par le système. Le but de cette évaluation est de comparer les résultats obtenus pour montrer que la qualité des données influe largement sur les résultats du data mining. Les évaluations ont montré qu'une partie importante des résultats ont été amélioré après application des méthodes de prétraitement.

Le système de nettoyage que nous avons implémenté est sujet à des améliorations. Plusieurs autres méthodes de prétraitement que nous n'avons pas implémentées peuvent être intégrées. Aussi, nos perspectives est d'avoir un système indépendant de la nature des données, ainsi qu'un ensemble de méthodes de data mining pour mieux évaluer la qualité des données.

BIBLIOGRAPHIE

- [1] D. Kindjangu, l'informatisation de la gestion des abonnés de la SNEL, Société nationale d'électricité en RDC, 2012
- [3] T. Mehenni, Data mining, cours de Master, Université Mohamed Boudiaf de M'sila, 2015/2016
- [4] C. Bernard et P. Craveski, Classification de données, Cours de Master, Université Claude Bernard, Lyon, 2014
- [5] R. Elamin, Data mining : techniques de DM pour la GRC dans les Banques, Thèse de doctorat ,Université de Biskra, 2015
- [6] A. Djeflal , Fouille de données avancée, Cours de Master, Université Mohamed Khider Biskra, 2015/2016
- [7] Google, https://simple-mail.fr/fonctionnalites_delivrer consulté avril 2016
- [8] Wikipidia, https://fr.wikipedia.org/wiki/Nettoyage_de_donn%C3%A9es consulté mai 2016
- [9] Comment ça marche, <http://www.commentcamarche.net/contents/104-bases-de-donnees-introduction>, consulté mai 2016
- [10] T. Gatid , Mini-mémoire de BDA , dernière mise à jour le 22 Mars 2002
- [11] F. Weber, Etude Statistique ,en 2005-2006
- [12] D.Donsez, Intégration des données, Université Joseph Fournier, Cours de Doctorat, 2002
- [13] M. Boufaïda , Adaptation de technique de l'extraction des connaissances à partir de données , Thèse de doctorat, Université Mentouri Constantine, 2012
- [14] D. Abdelkader et R. Rakotomalala, Extraction des connaissances à partir de données, Techniques de l'ingénieur, 2002
- [15] B. Liaudet, Cours de data mining : Modélisation et présentation générale, Université de Finance, Septembre 2008
- [16] D. Chami, La plate forme orientée agent pour le data mining, Mémoire de Magister, Université Hadj Lakhedar Batna, 2009/2010
- [17] G. Calas, Etude des principaux algorithmes de data minig, Ecole ingénieur en informatique, Le Kremlin-Bicêtre, France, 2009
- [18] F. Bash, K-means et théorie des graphes, Cours, Université Alexandre Boulch, 2010
- [20] M.N. Mami, Extraction des connaissances dans l'environnement distribué, Mémoire de Master, Ecole Nationale des Sciences de l'Informatique, 2013

- [21] A. El Mhamdi, Gestion proactive du changement dans les projets de réingénierie des processus métiers, Thèse de Doctora , Université de Paris 8, 2009
- [22] M. Jacques, Cours de Bases de données, Université du Sud Toulon-var, 2014
- [23] M. Parmami, Mémoire fin d'études, 2013
- S. Caron, Une introduction aux arbres de décision, Mémoire de License, France, 2011 [24]**
- [25] J.Han , M. Kamber ,Data Mining Concepts and Techniques deuxième édition, Stanley B. Zdonik et D.Maier . Etats-Unis d'Amerique,2006
- [26] J. Balding, Peter Bloomfield, Noel A. C. Cressie, Nicholas I. Fisher, Iain M. Johnstone, J. B. Kadane, Louise M. Ryan, David W. Scott, Adrian F. M. Smith, Jozef L. Teugels, Exploratory Data Mining and Data Cleaning , J.Wiley ,Sons, Inc., Hoboken, N. Jersey, Canada., 2003

ملخص: إن عملية استخراج المعلومات من خلال البيانات تتم عبر البحث في ثنایا الكم الهائل من البيانات عن المعلومات المختبئة فيها. وتتم هذه العملية عبر مراحل عديدة بدءا من الفهم الدقيق لميدان هذه المعلومات إلى مرحلة تفسير النتائج المتحصل عليها، مرورا بعدة مراحل ثانوية يتم فيها اختيار البيانات وتحضيرها والتي تبين أنها مهمة للغاية لأجل ضمان نتائج ذات جودة عالية. إن عملية تطهير البيانات هي حتما أصعب العمليات ضمن مرحلة التحضير. إن مشروعنا يتمثل في تسليط الضوء على هذه العملية وذلك بتقديم معظم الطرق والتقنيات المستخدمة لأجل ذلك. وقد أنجزنا تبعا لذلك نظاما لتطهير البيانات من خلال برمجة عدة خوارزميات وقمنا بعد ذلك بتطبيقها على مجموعة حقيقية من البيانات.

كلمات مفتاحية: استخراج المعلومات، جودة البيانات، المعالجة القبلية، التطهير، تصنيف بايز (Bayes)

Abstract : Knowledge extraction from data consists in browsing huge volumes of data contained in a database, in order to search knowledge. This process it composed of several operations, which begins by understanding the domain studied until the interpretation of the obtained results, while passing by several stages of selection and preprocessing of data that prove to be very important to guarantee efficient results. The data cleaning is certainly the most complex phase of data preparation. Our work consists in highlighting the cleaning phase of data, by presenting several methods and techniques that perform this task. We describe and develop a data cleaning system that performs a set of cleaning methods and apply them on real data.

Keywords: Knowledge extraction, data quality, preprocessing, cleaning, Bayesian classification.

Résumé : L'Extraction de Connaissances à partir de Données (ECD) consiste à parcourir d'immenses volumes de données contenues dans une base, à la recherche de connaissances. Il se décompose en plusieurs opérations, allant de la phase de compréhension du domaine étudié jusqu'à l'interprétation des résultats, en passant par plusieurs étapes de sélection et de préparation des données qui s'avèrent très importantes pour garantir des résultats efficaces. La phase de prétraitement est certainement l'une des phases de préparation des données la plus complexe. Notre travail consiste à mettre en évidence la phase de prétraitement des données, en présentant les méthodes et techniques y afférentes. Nous décrivons et implémentons un ensemble de ces méthodes que nous appliquons sur un extrait de données réelles.

Mots clés : Extraction des connaissances, qualité des données, prétraitement, nettoyage, classification bayésienne