REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE MOHAMED BOUDIAF - M'SILA
FACULTE DES MATHEMATIQUES 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 et Génie Logiciel (SIGL)

Par: Hamma Manel

SUJET

DATA MINING DES BASES DE DONNEES DISTRIBUEES

Soutenu publiquement le : / /2019 devant le jury composé de :

Dr.Ouldmohamedi Najib	**Université de M'sila**	**Président**
Dr. Meheni Tahar	**Université de M'sila**	**Encadreur**
Dr.Barkat Abdelbasset	**Université de M'sila**	**Examinateur**

Promotion : 2018 /2019

Résumé :

Le Data Mining est une technologie dont le but est la valorisation de l'information et l'extraction de connaissances d'un grand nombre de données, et dans la majorité des cas ces données ne résident pas dans un emplacement centralisé, ce qui complique l'application des techniques de Data Mining sur des données distribuées. L'objectif de notre projet est de présenter les différentes approches utilisées pour extraire la connaissance à partir des données distribuées. Nous utilisons l'arbre de décision comme technique de Data Mining et nous essayons d'implémenter l'approche par agrégation et d'effectuer des comparaisons d'un ensemble d'agrégats.

Mots clé : Data Mining, base de données distribuées, agrégation.

Abstract

Data Mining is a technology whose purpose the exploitation of information and the extraction of knowledge from a large number of data, and in most cases this data does not reside in a centralized location, which makes it difficult to application of Data Mining techniques on distributed data. The objective of our project is to present the different approaches used to extract knowledge from distributed data. We use the decision tree as a Data Mining technique and we try to implement the aggregation approach and make comparisons of a set of aggregates.

Key word: Data Mining, distributed data bases, aggregation.

ملخـــص :

استخراج البيانات هي تقنية هدفها استغلال المعلومات واستخراج المعرفة من عدد كبير من البيانات ، وفي معظم الحالات لا توجد هذه البيانات في موقع مركزي ، مما يجعل من الصعب تطبيق تقنيات استخراج البيانات على البيانات الموزعة. الهدف من مشروعنا هو تقديم الأساليب المختلفة المستخدمة لاستخراج المعرفة من البيانات الموزعة. نحن نستخدم شجرة القرار كأسلوب لاستخراج البيانات ونحاول تنفيذ نهج التجميع وإجراء مقارنات لمجموعة من المجاميع.

الكلمات الرئيسية: استخراج البيانات، قاعدة البيانات الموزعة ،التجميع.

Tables des matières

Sommaire **Pages**

Chapitre 2 : Approche de Data Mining Distribuées

Liste des figures

Figures **Pages**

Chapitre I

Chapitre II

Liste Des Tables

Tables **Pages**

Chapitre III

INTRODUCTION GENERALE

Avec la progression technologique des outils de collecte et de stockage des données, la plupart des gros systèmes sont submergés par un flot de données continu qui est quotidiennement stocké dans les bases de données. Et avec les révolutions technologiques intervenues dans le domaine des réseaux de communications qui ont permis à l'approche base de données distribuée de devenir une solution alternative à la centralisation .

Ces méga bases qui ne cessent de s'accumuler et de s'accroître d'une façon exponentielle au fil du temps sont devenues une mine d'information qui alimente le Data Mining qui ce charge d'extraction d'informations intéressantes, non triviales, implicites, préalablement inconnues et potentiellement utiles on utilisant des techniques spéciales.

L'application classique des algorithmes de Data Mining dans ces environnements distribués sont inadéquats pour leurs traitements devenus de plus en plus complexes. Pour palier à ce problème, on a eu recours aux approches d'extraction de données distribuées.

L'objectif de notre projet est de présenter les différentes approches utilisées pour extraire la connaissance à partir des données distribuées. Ensuite, nous essayons d'implémenter l'approche par agrégation et d'effectuer des comparaisons d'un ensemble d'agrégats.

Le mémoire est composé des chapitres suivants :

- Le chapitre 1 présente les bases de données distribuées.
- Le chapitre 2 décrit les différentes approches de data mining des bases de données distribuées.
- Le chapitre 3 présente l'approche d'agrégation utilisé ainsi que quelques résultats obtenus en utilisant les arbres de décision comme algorithmes de classification.
- Enfin une conclusion du projet est présentée.

CHAPITRE I

BASE DE DONNEES DITRIBUEES

1-Introduction

Depuis ces dernières années, les techniques informatiques évoluent vers le traitement de grande masse d'informations de nature diverse, intégrées dans un environnement géographiquement réparti ou ils doivent cohabiter du matériel généralement hétérogène.Dans ce contexte, et vue la souplesse des SGBDs d'une part et les performances des réseaux d'autre part, les bases de données réparties sont une solution importante pour parvenir à maîtriser la distribution des données.

2-Définition

Une base de données répartie est une collection de bases de données localisées sur différents sites, généralement distants, mises en relations les unes avec les autres à travers un réseau d'ordinateurs, perçues pour l'utilisateur comme une base de données unique. Elle permet de rassembler des données plus ou moins hétérogènes, disséminées dans un réseau sous forme d'une base de données globale, homogène et intégrée.[12]

3-Le besoin de la distribution

La distribution est devenue impérative à cause des raisons suivantes :

- La décentralisation de l'information (cas des multinationales),
- Augmentation du volume de l'information (14 fois de 1990 à 2000),
- Augmentation du volume des transactions (10 fois dans les 5 prochainesannées).[9]

4-Avantages :

-Plus de fiabilité: les bases de données réparties ont souvent des données répliquées. La panne d'un site n'est pas très importante pour l'utilisateur, qui s'adressera à un autre site.[1]

-Meilleure performance : réduire le trafic sur le réseau est une possibilité d'accroître les performances. Le but de la répartition des données est de les rapprocher de l'endroit où elles sont accédées. Répartir une base de données sur plusieurs sites permet de répartir la charge sur les processeurs et sur les entrées/sorties.[5]

-Faciliter l'accroissement: l'accroissement se fait par l'ajout de machines sur le réseau. [5]

5-Les Objectifs des systèmes distribués :

Les principaux objectifs sont :

- Transparence pour l'utilisateur.
- Autonomie de chaque site
- Absence de site privilégié
- Continuité de service
- Transparence vis à vis de la localisation des données
- Transparence vis à vis de la fragmentation
- Transparence vis à vis de la réplication
- Traitement des requêtes distribuées
- Indépendance vis à vis du matériel
- Indépendance vis à vis du système d'exploitation
- Indépendance vis à vis du réseau
- Indépendance vis à vis du SGBD. [9]

6-SGBD réparti

Une base de données centralisée est gérée par un seul SGBD, est stockée dans satotalité à un emplacement physique unique et ses divers traitements sont confiés à uneseule et même unité de traitement. Par opposition, une base de données distribuéeestgérée par plusieursprocesseurs, sites ou SGBD.[9]

6.1 Définition d'un Système de gestion de bases de données

Le SGBD (Système de Gestion des Bases de Données) est l'outil principal de gestion d'une base de données. Il permet d'insérer, de modifier et de rechercher efficacement des données spécifiques dans une grande masse d'informations. C'est une interface entre les utilisateurs et la mémoire de masse. Il facilite ainsi le travail des utilisateurs en leur donnant

l'impression que l'information est organisée comme ils le souhaitent. Le SGBD est composé de plusieurs couches :

-Le SGBD externe (user interface handler). Sa tâche est d'interpréter les commandes utilisateurs.

-Le contrôleur sémantique des données (sémanticdata controler). Il utilise les différentes contraintes définies sur la base de données afin de vérifier qu'une requête d'un utilisateur peut être effectuée.

-Le processeur de requêtes (query processor). Il détermine une stratégie afin de minimiser le temps d'exécution d'une requête.

-Le gestionnaire de transactions (transaction manager). Il assure la coordination des différentes demandes des utilisateurs.

-Le gestionnaire de reprise (recovery manager). Il s'occupe d'assurer la cohérence des données lorsque des pannes surviennent.

-Le système de gestion des fichiers (run-time support processor). Il gère le stockage physique de l'information. Il est dépendant du matériel utilisé.[5]

Un SGBD réparti doit rendre la répartition des bases de données transparentes aux utilisateurs. La base de données étant répartie, il faut également répartir certaines fonctionnalités du SGBD.

6.2 Rôle d'un SGBD

Le logiciel de gestion d'un système de base de données (SGBD) à pour rôle :

-d'assurer la confidentialité des données: implémentation d'un mécanisme d'authentification par compte avec un mot de passe, attribution de rôles aux utilisateurs permettant d'ouvrir ou de réduire la surface d'exposition des données.

-d'assurer la cohérence des données : vérifier les connaissances d'unicité (clés primaires) et les contraintes d'intégrité fonctionnelles (s'assurer qu'une clé étrangère référence bien un clé primaire et que la suppression d'une clé primaire ne crée pas d'enregistrement orphelins).

-d’assurer la gestion des incidents : le système doit s’assurer que l’échec d’une requête ne remet pas en cause l’intégrité des données. Il doit également pouvoir procéder aux reprises sur incident suite à une panne du serveur hébergeant la base de données, par exemple.[5]

7-Conception d'une base de données distribuées

La définition du schéma de répartition est la partie la plus délicate de la phase de conception d'une BDR, car il n'existe pas de méthode miracle pour trouver la solution optimale. L'administrateur doit donc prendre des décisions dont l'objectif est de minimiser le nombre de transferts entre sites, les temps de transfert, le volume de données transférées, les temps moyens de traitement des requêtes, et le nombre de copies de fragments, ... etc.

7.1-Méthodes de conception

Deux approches fondamentales sont à l'origine de la conception des bases de données réparties : la conception descendante 'Top down design' et la conception ascendante 'Bottom up design'.

7.1.1-Conception ascendante :

Cette approche se base sur le fait que la répartition est déjà faite, mais il faut réussir à intégrer les différentes BDs existantes en une seule BD globale. En d'autre terme, les schémas conceptuels locaux existent et il faut réussir à les unifier dans un schéma conceptuel global. [12]

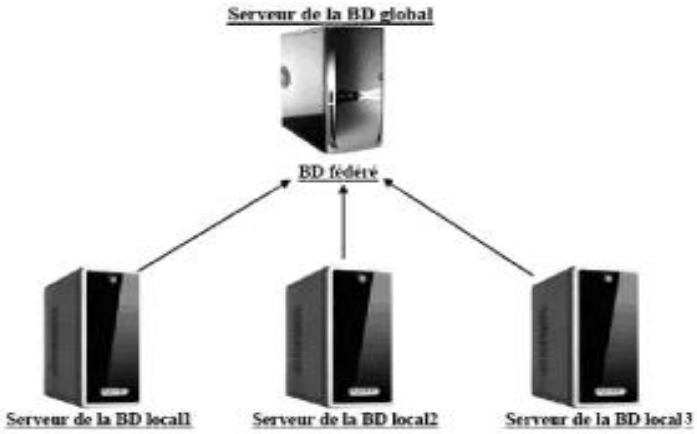

FIG. 1.1 Conception ascendante [12]

7.1.2-Conception descendante :

On commence par définir un schéma conceptuel global de la base de données répartie, puis on le distribue sur les différents sites en des schémas conceptuels locaux. La répartition se fait donc en deux étapes, en première étape la fragmentation et en deuxième étape l'allocation de ces fragments aux sites.[12]

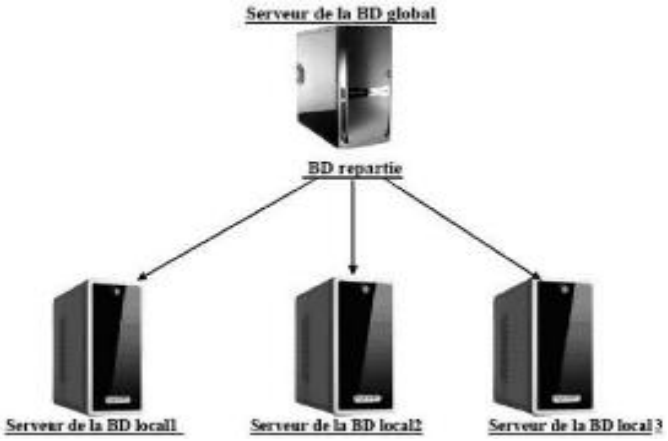

FIG. 1.2 Conception descendante [12]

8- La fragmentation

8.1 Définition :

La fragmentation est le processus de décomposition d'une base de données en un ensemble de sous bases de données. Cette décomposition doit être sans perte d'information.

8.2 Objectif de la fragmentation :

Les applications ne travaillent que sur des sous-ensembles des relations. Une distribution complète des relations générerait soit beaucoup de trafic, soit une réplication des données avec tous les problèmes que cela occasionne : problèmes de mises à jour, problèmes de stockage. Il est donc préférable de mieux distribuer ces sous-ensembles.

L'utilisation de petits fragments permet de faire tourner plus de processus simultanément, ce qui entraîne une meilleure utilisation des capacités du réseau d'ordinateurs.

8.3 Les problèmes de la fragmentation :

La fragmentation peut être coûteuse s'il existe des applications qui possèdent des besoins opposés. On est en quelque sorte dans le cas d'une exclusion mutuelle qui empêche une fragmentation correcte.

Par ailleurs, la vérification des dépendances sur différents sites peut être une opération très longue.

8.4 Types de fragmentation

1.8.4.1-<u>La fragmentation horizontale</u> :

C'est un découpage d'une table en sous tables par utilisation de prédicats permettant de sélectionner les lignes appartenant à chaque fragment. L'opération de fragmentation est obtenue grâce à la sélection des tuples d'une table selon un ou des critères bien précis et la reconstitution de la relation initiale se fait grâce a l'union (U) des sous-relations. [13]

Client	NoCli	NomCli	VilleCli
	C1	Marius	Marseille
	C2	Durant	Lyon
	C3	Duval	Paris
	C4	Olive	Marseille

Client1	NoCli	NomCli	VilleCli
	C1	Marius	Marseille
	C4	Durant	Marseille

Client2	NoCli	NomCli	VilleCli
	C2	Durant	Lyon
	C3	Duval	Paris

FIG.1.3 Exemple de fragmentation horizontale [13]

8.4.1-<u>La fragmentation verticale</u> :

Elle est le découpage d'une table en sous tables par projection permettant de sélectionner les colonnes composant chaque fragment. La relation initiale doit pouvoir être recomposée par la jointure des fragments. [13]

Produit	NoProd	DesProd	Prix	NoFour
	P1	Chaise	80	F1
	P2	Bureau	150	F1
	P3	Table	100	F2
	P4	fauteuil	200	F2

Produit	NoProd	DesProd	prix
	P1	Chaise	80
	P2	Bureau	150
	P3	Table	100
	P4	fauteuil	200

Produit	NoProd	NoFour
	P1	F1
	P2	F1
	P3	F2
	P4	F2

FIG.1.4 Exemple de fragmentation horizontale [13]

8.5-Les règles de la fragmentation

Un problème qui se pose pour la fragmentation est comment définir un bon degré de fragmentation. Il existe trois règles pour la fragmentation :

1. **Complétude** : pour toute donnée d'une relation globale R, il existe au moins un fragment Ri de la relation R qui possède cette donnée.[13]

2. **Reconstruction** : pour toute relation R décomposée en un ensemble de fragments Ri, il existe une opération de reconstruction à définir en fonction de la fragmentation. Pour les fragmentations horizontales, l'opération de reconstruction est une union. Pour les fragmentations verticales c'est la jointure. [13]

3. **Disjonction** : une donnée n'est présente que dans un seul fragment, sauf dans le cas de la fragmentation verticale pour la clé primaire qui doit être présente dans l'ensemble des fragments issus d'une relation.[13]

9-Allocation

L'affectation des fragments sur les sites est décidée en fonction de l'origine prévue des requêtes qui ont servi à la fragmentation. Le but est de placer les fragments sur les sites où ils sont les plus utilisés, et ce pour minimiser les transferts de données entre les sites.

L'allocation peut se faire avec réplication ou sans réplication. Sachant que la réplication favorise les performances des requêtes et la disponibilité des données, mais est coûteuse en considérant les mises à jour des fragments répliqués. [5]

10- Réplication

Certaines informations ne subissent pas souvent de modification (comme le nom de famille, l'adresse ou le nombre d'enfants des employés) et par conséquent une copie même ancienne de ces informations est, dans sa grande majorité, exacte.

La réplication consiste en l'utilisation de clichés (ang. *snapshot*). Un cliché représente un état de la base de données à un instant donné.La pertinence d'un cliché diminue donc au fur et à mesure que le temps passe.[5]

11-Conclusion

Dans ce chapitre nous avons présenté les déférents concepts des bases de données distribuées puisque ils sont un sujet très vaste qui nécessite plus pour être traitées en profondeur, et nous avons conclu que les bases de données distribuées sont une solution adéquate pour parvenir à maîtriser la distribution des ressources informatiques sur plusieurs processeurs interconnectés dans un contexte où l'on gère des données structurées et où il est fondamental de garantir la cohérence des données.

CHAPITRE II

APPROCHE DE DATA MINING DITRIBUEES

1-introduction

Ce chapitre nous commençons par présenter les concepts de Data Mining (DM), où les différentes étapes du processus d'extraction de connaissances à partir des données sont décrites, ensuite nous présentons les différentes approches de Data Mining Distribuées (DDM).

2-Historique

Le *"Data Mining"* que l'on peut traduire par "fouille de données" apparaît au milieu des années 1990 aux États-Unis comme une nouvelle discipline à l'interface de la statistique et des technologies de l'information: bases de données, intelligence artificielle, apprentissage automatique« *machine learning*».[2]

Les premières applications se sont faites dans le domaine de la gestion de la relation client qui consiste à analyser le comportement de la clientèle pour mieux la fidéliser et lui proposer des produits adaptés. La recherche d'information dans les grandes bases de données médicales ou de santé (enquêtes, données hospitalières etc.) par des techniques de *Data Mining* est encore relativement peu développée, mais devrait se développer très vite à partir du moment où les outils existent.

La communauté de *"data mining "* a initié sa première conférence en 1995 à la suite denombreux ateliers (workshops) sur le *KDD* entre 1989 et 1994. La première revue du domaine" *Data mining and knowledgediscovery journal "* publiée par "*Kluwers*" a été lancée en1997.[2]

3-Définitions

-Définition1 : « Le data mining, ou fouille de données, est l'ensemble des méthodes et techniques destinées à l'exploration et l'analyse de bases de données informatiques (souvent grandes), de façon automatique ou semi-automatique, en vue de détecter dans ces données des règles, des associations, des tendances inconnues ou cachées, des structures particulières restituant l'essentiel de l'information utile tout en réduisant la quantité de données».[2]

-Définition 2 : «Le Data mining est un processus non trivial qui consiste à identifier, dans desdonnées, des schémas nouveaux, valides, potentiellement utiles et surtout compréhensibles

et utilisables».[2]

4-processus d'extraction de connaissances à partir des données

Le processus d'extraction de l'information consiste à parcourir les données volumineuses contenues dans une base, à la recherche de connaissance. Ce processus est décrit dans le schéma suivant. (FIG. 2.1). Ce processus comprend des étapes de définition du problème (définition du domaine, but de l'utilisateur final), de préparation des données (sélection, préparation, transformation), de fouille de données (sélection, des outils de data mining appropriés, recherche des patrons) et d'évaluation des résultats pour aboutir aux nouvelles connaissances. [3]

Le processus présenté est itératif et plusieurs retours en arrière dans les différentes étapes peuvent être nécessaires pour affiner les résultats.

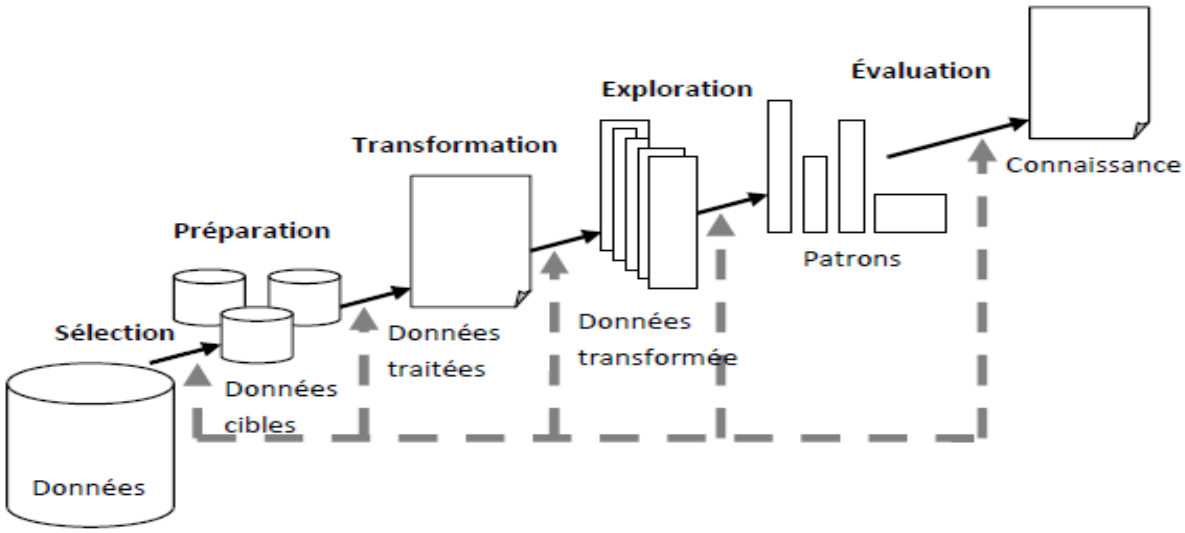

Figure 2.1 : processus d'extraction de données [6]

4.1-Nettoyage et intégration des données

Le nettoyage des données consiste à les supprimer, soit à les modifier de manière à tirer le meilleur profit. L'intégration est la combinaison des données provenant de plusieurs sources (base de données, sources externes, etc.). Le but de ces deux opérations est de générer des entrepôts de données

et/ou des magasins de données spécialisés contenant les données retravaillées pour faciliter leurs exploitations futures.[2]

4.2- Pré-traitement des données

Il peut arriver parfois que les bases de données contiennent des données incomplètes et/ou bruitées. Ces données erronées, manquantes ou inconsistantes doivent être retravaillées si cela n'a pas été fait précédemment. Dans le cas contraire, durant l'étape précédente, les données sont stockées dans un entrepôt. Cette étape permet de sélectionner et transformer des données de manière à les rendre exploitables par un outil de fouille de données.

4.3- Fouille de données

La fouille de données c'est trouver des pépites de connaissances à partir des données. Tout le travail consiste à appliquer des méthodes intelligentes dans le but d'extraire cette connaissance.

4.4 Evaluation et présentation

Cette phase est constituée de l'évaluation, qui mesure l'intérêt des motifs extraits, et de la présentation des résultats à l'utilisateur grâce à différentes techniques de visualisation.

Il y a principalement deux techniques de validation qui sont la technique de validation statistique et la technique de validation par expertise.

La validation statistiqueconsiste à utiliser des méthodes de base de statistique descriptive. L'objectif est d'obtenir des informations qui permettront de juger le résultat obtenu, ou d'estimer la qualité des données d'apprentissage. Cette validation peut être obtenue par :

•le calcul des moyennes et variances des attributs,

•si possible, le calcul de la corrélation entre certains champs,

•ou la détermination de la classe majoritaire dans le cas de la classification.

La validation par expertiseest réalisée par un expert du domaine qui jugera la pertinence des résultats produits. Par exemple pour la recherche des règles d'association, c'est l'expert du domaine qui jugera la pertinence des règles.

5- Tâches du Data Mining

De nombreuses tâches peuvent être associées au Data Mining, parmi elles nous pouvons citer: la classification. La classification consiste à examiner les caractéristiques d'un élément nouvellement présenté afin de l'affecter à une classe d'un ensemble prédéfini. Elle se rapporte à des événements discrets (homme/femme, client bon payeur/ mauvais payeur, etc.) .[3]

- La prédiction :

La prédiction vise à prédire la valeur future d'un champ. Tout comme la tâche précédente, elle s'appuie sur le passé et le présent mais son résultat se situe dans un futur généralement précisé. [2]

- **L'optimisation** :

C'est le problème qui consiste à optimiser un ou plusieurs paramètres du système selon un ensemble de contraintes. Pour résoudre de nombreux problèmes, il est courant pour chaque solution potentielle d'y associer une fonction d'évaluation. Le but de l'optimisation est de maximiser ou minimiser cette fonction.[3]

-**La segmentation** :

(analyse des clusters) L'analyse des clusters consiste à segmenter une population hétérogène en sous populations homogènes (groupes). Il appartient ensuite à un expert du domaine de déterminer l'intérêt et la signification des groupes ainsi constitués. Contrairement à la classification, les sous-populations ne sont pas préétablies.

- L'association:

L'association est la tâche qui consiste à rechercher les relations ou les dépendances existantes entre plusieurs caractéristiques d'un individu.[3]

6- Les méthodes de data mining

Dans cette partie, nous proposons des techniques d'extraction de données qui sont utilisées et nous mettons en avant la méthode de l'arbre de décision qui sera utilisée dans notre étude

6-1- Segmentation (Clustering)

La segmentation est l'opération qui consiste à regrouper les individus d'une population enun nombre limité de groupes, les segments (ou clusters, ou partitions), qui ont deuxpropriétés : D'une part, ils ne sont pas prédéfinis, mais découverts automatiquement au coursde l'opération, contrairement aux classes de la classification. D'autre part, les segmentsregroupent les individus ayant des caractéristiques similaires et séparent les individus ayantdes caractéristiques différentes (homogénéité interne et hétérogénéité externe).

La segmentation est une tâche d'apprentissage *"non supervisée"* car on ne dispose d'aucune autre information préalable que la description des exemples. Après application de l'algorithme et donc lorsque les groupes ont été construits, d'autres techniques ou une expertise doivent dégager leur signification et leur éventuel intérêt,(*k*-moyennes).[2]

6-2- Règles d'association

Les règles d'association sont traditionnellement liées au secteur de la distribution car leurprincipale application est *"l'analyse du panier de la ménagère (market basket analysis)"* quiconsiste en la recherche d'associations entre produits sur les tickets de caisse. Le but de laméthode 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 peut être appliquée à toutsecteur d'activité pour lequel il est intéressant de rechercher des groupements potentiels deproduits ou de services: services bancaires, services de télécommunications, par exemple.Elle peut être également utilisée dans le secteur médical pour la recherche de complicationsdues à des associations de médicaments ou à la recherche de fraudes en recherchant desassociations inhabituelles.[6]

6-3- Les plus proches voisins

La méthode des plus proches voisins *(PPV* en bref, *nearestneighbor*en anglais*)* est uneméthode dédiée à la classification qui peut être étendue à des tâches d'estimation. Laméthode *PPV* est une méthode de raisonnement à partir de cas. Elle part de l'idée de prendredes décisions en recherchant un ou des cas similaires déjà résolus en mémoire[3]

6-4- Les réseaux de neurones

Les réseaux de neurones sont apparus dans les années cinquante avec les premiersperceptrons, et sont utilisés industriellement depuis les années quatre-vingt. Un réseau deneurone *"ou réseau neuronal"* a une architecture calquée sur celle du cerveau, organisée enneurones et synapses, et se présente comme un ensemble de noeuds "ou neurones formels, ouunités" connectés entre eux, chaque variable prédictive continue correspondant à un nœud d'un premier niveau, appelé *couche d'entrée,* et chaque variable prédictive catégorique (ouchaque modalité d'une variable catégorique) correspondant également à un noeud de lacouche d'entrée.[2]

6-5- Les arbres de décision

La méthode des arbres de décision est l'une des plus intuitives et des plus populaires du Data Mining, d'autant plus qu'elle fournit des règles explicites de classement et supporte bienles données hétérogènes, manquantes et les effets non linéaires. Pour les applicationsrelevant du marketing de bases de données, cette méthode étant préférée dans laprédiction du risque en raison de sa plus grande robustesse.

Cette technique est employée en classement pour détecter des critèrespermettant de répartir les individus d'une population en n classes (souvent n=2) prédéfinies.On commence par choisir la variable qui, par ses modalités, sépare le mieux les individus dechaque classe, de façon à avoir des sous-populations, que l'on appelle noeuds, contenantchacune le plus possible d'individus d'une seule classe, puis on réitère la même opérationsur chaque nouveau noeud obtenu jusqu'à ce que la séparation des individus ne soit pluspossible ou plus souhaitable.[2]

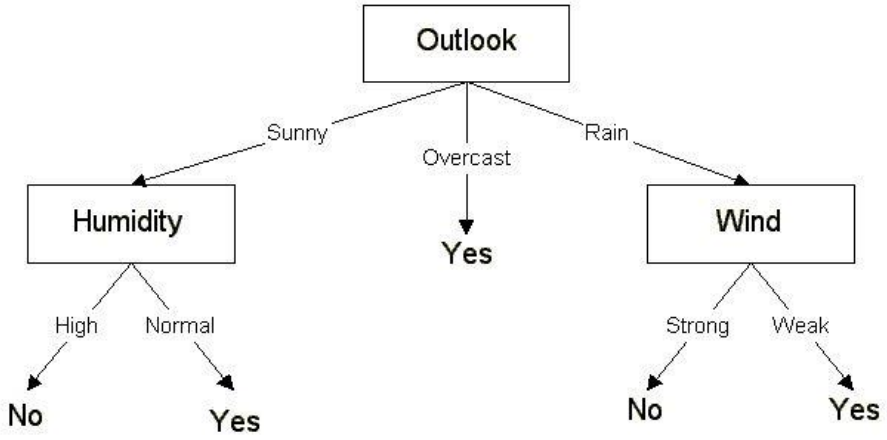

FIG 2.2 : arbre de décision

-une **règle** est générée pour chaque **chemin** de l'arbre (de la racine à une feuille)

-Les paires attribut-valeur d'un chemin forment une conjonction

-Le noeud terminal représente la classe prédite

Exemple d'une règle :

Sioutlook = sunny

Ethumidity = normal

Alorsplayball

a.Avantages

•Adaptabilité aux attributs de valeurs manquantes: les algorithmes peuventtraiter les valeurs manquantes (descriptions contenant des champs non renseignés)

•Bonne lisibilité du résultat: un arbre de décision est facile à interpréter et à lareprésentation graphique d'un ensemble de règles.

•Traitement de tout type de données : l'algorithme peut prendre en compte tous lestypes d'attributs et les valeurs manquantes. Il est robuste au bruit.

•Sélectionne des variables pertinentes : l'arbre contient les attributs utiles pour laclassification. L'algorithme peut donc être utilisé comme pré-traitement quipermet de sélectionner l'ensemble des variables pertinentes pour ensuite appliquerune autre méthode.

•Donne une classification efficace: l'attribution d'une classe à un exemple à l'aided'un arbre de décision est un processus très efficace (parcours d'un chemin dans unarbre).

•Disponibilité des outils: les algorithmes de génération d'arbres de décision sontdisponibles dans tous les environnements de fouille de données.

•Méthode extensible et modifiable: la méthode peut être adaptée pour résoudre destâches d'estimation et de prédiction. Des améliorations des performances desalgorithmes de base sont possibles grâce aux techniques qui génèrent un ensembled'arbres votant pour attribuer la classe.[2]

b. Inconvénients

•Méthode sensible au nombre de classes: les performances tendent à se dégraderlorsque le nombre de classes devient trop important.

•Manque d'évolutivité dans le temps: l'algorithme n'est pas incrémental, c'est-àdire,que si les données évoluent avec le temps, il est nécessaire de relancer unephase d'apprentissage sur l'échantillon complet (anciens exemples et nouveauxexemples).[2]

7- <u>Intérêt du data mining distribué</u> :

L'application classique du processus de découverte de connaissances dans des environnements distribués nécessite la collecte de données distribuées dans un entrepôt de données central pour le traitement. Cependant, elle est généralement soit inefficace, soit irréalisable pour les raisons suivantes:

-<u>Coût de communication</u> : Le transfert de grandes quantités de données sur un réseau peut prendre non seulement beaucoup de temps mais aussi exiger une charge financière

importante. Même un petit volume de données peut créer des problèmes dans les environnements réseau sans fil avec une largeur de bande limitée. On note également que la communication peut être coûteuse, car les bases de données distribuées ne sont pas toujours constantes et inchangeables. [3]

- **Coût de calcul** : Le coût de calcul à extraire d'un entrepôt central de données est beaucoup plus grand que la somme du coût d'analyse des plus petites parties des données qui pourraient également être faites en parallèle. Dans une grille, par exemple, il est plus facile de recueillir les données à un endroit central. Cependant, une approche data mining distribuée ferait une meilleure exploitation des ressources disponibles. [3]

-**Des données privées sensibles** : Il existe un grand nombre d'applications data mining qui traitent des données sensibles, telles que les dossiers médicaux et les dossiers financiers. La collection centrale de ces données n'est pas souhaitable car elle met leur vie privée en danger. Dans certains cas (par exemple les banques, télécommunications) les données sont susceptibles d'appartenir à différents organismes qui veulent échanger des connaissances sans échanger des informations confidentielles brutes.[3]

8-Domaines d'application du Data Mining

Les domaines d'application du data mining sont très nombreux. On cite par exemple :

-la prévision de parts de marché

- la gestion et la segmentation de la clientèle.

-l'analyse du portefeuille « clientèle ».

- l'analyse de données scientifiques

- le contrôle de données financières en temps réel.

-la détection de comportements frauduleux.

- le diagnostic et la maintenance préventive.

9-Approches de Data Mining des bases de données distribuées :

9.1- Intégration des liens utiles dans les arbres de décision pour la classification des bases de données distribuées

Cette approche présente une méthode de classification par les arbres de décision permettant de réduire le coût de communication en intégrant une méthode de prédiction des attributs utiles à la construction de l'arbre ainsi qu'une méthode de sélection du site de la partie de la base de données le plus économique.

Le principe de cette approche consiste à échanger des informations entre les différentes tables de la base de données distribuée par le biais de liens déterminés, et cela dans le but d'accomplir une tâche relative à l'algorithme de fouille de données, ces liens ne sont que des attributs issus de différents sites et qui jouent le rôle de ponts pour le transfert des informations.[11]

9.1.1-Notion d'utilité des liens :

Un lien est considéré utile s'il donne un gain-en-information significatif et vice-versa. Le gain-en-information est défini comme suit :

Définition (Gain-en-information): Soient *P* tuples positifs et *N* tuples négatifs d'une relation. Supposons qu'un attribut *A* divise ces tuples en *k* partitions, chacune contient *Pi* tuples positifs et *Ni* tuples négatifs.[11]

$$gain_{info(A)} = entropie(P,N) - \sum_{i=1}^{k} \frac{Pi + Ni}{P + N} \,.entropie(Pi,Ni)$$

Avec

$$entropie(P,N) = -(\frac{P}{P+N} \,.log\, \frac{P}{P+N} + \frac{N}{N+P} \,.\log \frac{dNy}{P+N})$$

9.2- Approche de La statistique suffisante

Cette approche d'apprentissage à partir de données distribuées repose sur une décomposition de la tâche d'apprentissage en deux composants: extraction de statistiques suffisantes à partir de données et génération d'hypothèses.

Cette approche provient de la révision de la formulation traditionnelle du problème de l'apprentissage à partir des données et en observant que la plupart des algorithmes d'apprentissage utilisent uniquement certaines statistiques calculées à partir des données dans le processus de génération des hypothèses qu'ils produisent . Cela donne une décomposition naturelle d'un algorithme d'apprentissage en deux composants: un composant d'extraction d'information qui formule et envoie une requête statistique à une source de données et à un composant de génération d'hypothèses qui utilise la statistique pour modifier une hypothèse partiellement construite (et invoque en outre les informations composant d'extraction si nécessaire). [1]

À la lumière de cette observation, un algorithme d'apprentissage à partir de données distribuées peut être utilisé. Également décomposé en deux composants: (1) extraction de l'information à partir de données distribuées et (2) génération d'hypothèses.

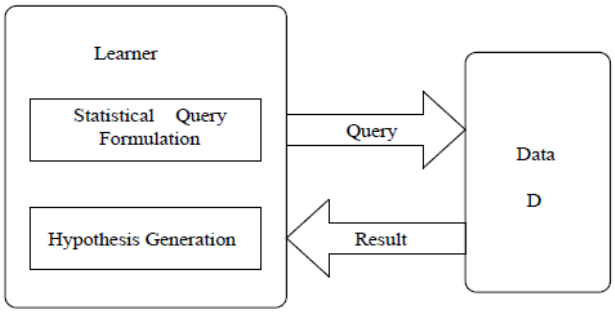

Fig2.3: Déterminer des statistiques suffisantes, rassembler les statistiquessuffisantes et générer les hypothèse [1]

L'extraction d'informations à partir de données distribuées implique la décomposition de chaque statistique requête q posée par le composant d'extraction d'informations de l'apprenant dans les sous-requêtes q1, · · ·, qK auxquelles les sources de données individuelles D1, · · ·, DK, respectivement, et un processus procédure pour combiner les réponses aux sous-requêtes en une réponse pour la requête d'origine q (voir la figure 1.3). Cela donne une stratégie générale pour transformer les algorithmes d'apprentissage à partir dedonnées centralisées en algorithmes exacts d'apprentissage à partir de données distribuées (un algorithme Ld pour apprendre des jeux de données distribués D1, · · ·, DK est exacte par

rapport à son équivalent centralisé L si l'hypothèse produite par Ld est identique à celle obtenue par L à partir des données complètes ensemble D obtenu en combinant de manière appropriée les ensembles de données D1, · · ·, DK).[1]

9.3-Approche de propositionalisation

La propositionalisation consiste à transformer un problème de fouille de données contenues dans plusieurs tables en un problème de fouille d'une seule table. La complexité du problème est alors réduite et le problème peut être résolu en appliquant une des méthodes de fouille de données classiques et efficaces pour la fouille dans une seule table. L'objectif dans cette transformation est de capturer l'information utile issue des différentes tables afin de la restituer dans la table qui sera fouillée. Cette approche sera présenté en détail dans le 3eme chapitre.[4]

10-Conclusion

Dans ce chapitre, nous avons présenté les principaux concepts de fouille de données, les processus, les tâches et les méthodes les plus utilisés en datamining et aussi nous avons présenté quelques approches de distribution du DM. Les nouvelles approches apportent beaucoup d'avantage par rapport aux méthodes distribuées traditionnelles qui évite le déplacement des données qui engendre des coûts de communication très élevés, contrairement aux méthodes traditionnelles qui exigent le déplacement de grandes quantités de données vers un seul noeud central.

CHAPITRE III

UTILISATION DE L'AGREGATION POUR LA CLASSIFICATION DES DONNEES DISTRIBUEES

1-Introduction :

Nous présentons dans ce chapitre l'approche d'agrégation que nous avons utilisée pour appliquer l'algorithme de classification dans les bases de donnes distribuées. Nous commençons par présenter cette approche ainsi que les agrégats que nous avons choisis, ensuite nous décrivons la base de données utilisées, et enfin nous présentons les différents résultats obtenus.

2-Propositionalisation

L'application des méthodes classiques de data mining sur des bases de données distribuées rencontre de sévères problèmes. Les données étant distantes, il est judicieux de les amener dans une seule table locale pour pouvoir appliquer un des algorithmes de fouille de données. Cette opération s'appelle la propositionalisation. Ce chapitre présente cette approche ainsi que ces différentes techniques.

Nous définissons la propositionalisation par le processus qui transforme une base de données en une table qui contient des valeurs d'attributs dérivés. L'objectif de la propositionalisation est de permettre d'avoir une seule table pour pouvoir appliquer un des algorithmes classiques de data mining.[7]

Il est à noter que la propositionalisation permet de dériver un attribut numérique en différents attributs appelés features (le mot est laissé en anglais). Un feature est un attribut obtenu par propositionalisation d'un ou plusieurs attributs de la base de données.[7]

Traditionnellement, la propossitionalisation a été réalisée par la programmation logique inductive, en utilisant des features binaires exprimées par la logique du premier ordre. Récemment, différentes approches sont utilisées.[8]

La dérivation des attributs peut être obtenue par plusieurs façons. Deux techniques peuvent êtres décrites. L'une s'appelle l'agrégation et l'autre la summarisation.

3-Summarisation

On peut toujours voir le processus de propositionalisation comme une série d'étapes permettant de projeter successivement l'information d'une table via des enregistrements vers la table cible.

Considérons deux tables P et Q, reliées par une association A. La summarisation sur A permet d'ajouter des informations à la table P qui proviennent de la structure de l'association A est ses propriétés ainsi que la Table Q. [8]

Pour réaliser la summarisation de Q, un ensemble d'agrégats de complexité 1 est nécessaire. Cependant, la multiplicité de l'association A peut influencer l'espace de recherche des patterns multi-relationnels relatifs à A. Le choix des agrégats dépend donc de la multiplicité de A. En particulier si on réalise la summarisation de Q sur A, seulement la multiplicité du côté de Q est nécessaire car l'association est généralement décrite de part et d'autre des deux tables qui la relient. Les quatre options possibles sont présentées comme suit.

- **1:** Pour chaque enregistrement dans A, Il existe un et un seul enregistrement dans Q. Dans ce cas, aucun agrégat n'est nécessaire. Une simple jointure doit permettre d'ajouter tous les attributs de Q vers A.
- **0..1 :** Similaire comme le cas précédent, sauf que un enregistrement dans P peut ne pas avoir une correspondance dans Q. Une jointure de valeur NULL peut être considérée.
- **1..n :** Pour chaque enregistrement dans P, il existe au moins un enregistrement dans Q. Dans ce cas, l'agrégation est nécessaire pour capturer les informations qui proviennent de l'ensemble des enregistrements de Q qui correspondent à un seul enregistrement de P.
- **0..n :** Similaire que le cas précédent, sauf que certains agrégats peuvent être indéfinis à cause de l'absence des enregistrements. La valeur NULL doit être utilisée avec précaution.[8]

4- Agrégation des attributs

L'agrégation est l'opération qui utilise certaines fonctions qui réduisent un ensemble de valeurs d'un ou plusieurs attributs en une seule valeur. Cette opération de réduction est généralement faite en gardant le plus possible la structure et la sémantique des attributs. Ces fonctions sont souvent appelées agrégats.

Nous pouvons définir l'agrégation par la fonction qui prend en entrée (input) un ensemble d'enregistrements de la base de données obtenus des différentes tables reliées, et produit une seule valeur certains attributs en sortie (output). [7]

Nous pouvons imaginer l'agrégation comme une projection de certaines informations stockées dans plusieurs tables sur une seule table par l'ajout d'attributs virtuels (features) à cette table.

L'agrégation inclut des agrégats de différentes complexités. Un aspect important de la complexité d'un agrégat est le nombre de tables utilisées. On peut trouver trois types de complexité selon le nombre de tables utilisées.

- **La complexité de type 0 :** fait appel à la même table par transformation de certains attributs.
- **La complexité de type 1 :** utilise les fonctions SQL sur une seule table, comme : sum, count, min,….
- **La complexité de type >1 :** utilise plusieurs tables associées. [7]

Voici quelques exemples qui illustrent ces trios types de complexité.

- **Complexité type 0:**
  - Propositions (adult == (age ≥ 18)
  - Fonctionsarithmétiques (area == width × length)
- **Complexité type 1:**
  - Count, count avec condition
  - Count distinct
  - Min, max, sum, avg
  - Exists, exists avec condition
  - Select record
  - Valeur dominante (ou fréquente)
- **Complexité de type > 1:**
  - Exists substructure
  - Count substructure
  - Conjonctiond'agrégats (maximum count of children).[7]

La liste des types d'agrégats est certainement longue. Pour cela, on fixe généralement les fonctions d'agrégation pour un certain nombre d'enregistrements de la base de données. Ce choix est souvent fait d'une manière expérimentale, mais l'utilisation d'heuristiques et des méthodes déterministes pour certainement permettre d'en sélectionner les meilleures. [7]

Dans notre projet nous avons sélectionné deux agrégats :

- **L'agrégat qui permet de déterminer les valeurs dominantes (ou fréquentes) :** cet agrégat est obtenu simplement on déterminant les enregistrements qui présentent les valeurs fréquentes d'un certain attribut. Ces enregistrements sont ensuite migrées vers la table cible de la base de données locales.
- **Un agrégat hybride (Min et Max) :** qui permet d'assembler les deux enregistrements qui présentent la valeur maximale et la valeur minimale parmi les valeurs de l'attribut en question.

5-Description générale de la base de données utilisée :

5.1-Le contexte

Cet ensemble de données est à l'origine de l'Institut national du diabète et des maladies digestives et rénales. L'objectif de l'ensemble de données est de prédire de manière diagnostique si un patient est diabétique ou non, en fonction de certaines mesures de diagnostic incluses dans l'ensemble de données. Plusieurs contraintes ont été placées sur la sélection de ces instances dans une base de données plus grande. En particulier, tous les patients ici sont des femmes d'au moins 21 ans et d'origine indienne Pima. [14]

5.2-Titre du fichier

Ce fichier a été créé par George John, octobre 1994 sou le nom de « Pima Indians Diabetes Database » Base de données sur le diabète des Indiens Pima. [14]

5.3-Source de la base de données :

- Propriétaires d'origine : National Institute of Diabetes and Digestive and Kidney Diseases.

-Doneur de la base de données : Vincent Sigillito (vgs@aplcen.apl.jhu.edu)

Centre de recherche, Chef de groupe RMI Laboratoire de physique appliquée Université Johns Hopkins Johns Hopkins Road Laurel, MD 20707 | (301) 953-6231. [14]

5.4- Description de la base de données :

-Nombre d'exemples (lignes) : 786

-Nombre de class : 2 Classes

Classe 1 : la valeur de classe 1 est interprétée comme « test positif pour le Diabète »

Classe 2 : la valeur de classe 2 est interprétée comme « test négatif pour le Diabète »

-Distribution des Classes :

Classe 1 : 500 (65.1%)

| Classe 2 :268 (34.9%)

-Nombred'attribut : 8 attributs

1. Number of times pregnant
2. Plasma glucose concentration a 2 hours in an oral glucose tolerance test
3. Diastolic blood pressure (mm Hg)
4. Triceps skin fold thickness (mm)
5. 2-Hour serum insulin (mu U/ml)
6. Body mass index (weight in kg/(height in m)^2)
7. Diabetes pedigree function
8. Age (years) [14]

6-Utilisation de la base de donné :

Ecarter le 1/3 de la table pour le test (262 enregistrements).

- Fragmenter la table en deux tables différentes on utilisant la fragmentation horizontale, Une table située sur le serveur local nommée « tabdiabete1 » (274 enregistrements) et une table située au serveur distant nommée « tabdiabete2 » (250enregistrements).

7-Scénarios d'agrégations:

Le choix d'agrégation a été établi pour obtenir les meilleurs résultats représentatifs de l'ensemble de données.

1er scénario : agrégation par « valeur fréquente » pour retourner toutes les lignes qui contiennent la valeur fréquente.

Le tableau suivant représente un ensemble de lignes obtenu après agrégation par valeur fréquente pour l'attribut « Number_pregnant » de la table distante

Number_ pregnant	plasma_glucose_ concentration	Diastolic_ blood _pressure	Triceps_skin_ fold_ thickness	Hour_serum_ insulin	Body_mass _index	Diabetes_ Pedigree _function	Age	Class
9	140	94	0	0	32,7	0,734	45	2.
9	130	70	0	0	34,2	0,652	45	2.
9	119	80	35	0	29	0,263	29	2.
9	72	78	25	0	31,6	0,28	38	1.
9	89	62	0	0	22,5	0,142	33	1.
9	91	68	0	0	24,2	0,2	58	1.
9	164	78	0	0	32,8	0,148	45	2.
9	124	70	33	402	35,4	0,282	34	1.
9	164	84	21	0	30,8	0,831	32	2.
9	112	82	24	0	28,2	1,282	50	2.
9	184	85	15	0	30	1,213	49	2.

Tab 3.1 : Exemple d'agrégation par valeur fréquente

On remarque dans le tableau 3.1 que la valeur la plus fréquente pour l'attribut « Number_pregnant » est la valeur 9.

2ème scénario : agrégation par « Max et Min » pour retourner toutes les lignes qui contiennent les valeurs minimales et les valeurs maximales.

Le tableau suivant représente un ensemble de lignes obtenu après agrégation Max et Min pour l'attribut « Number_pregnant » de la table distante

Number_ pregnant	plasma_glucose_ concentration	Diastolic_ blood _pressure	Triceps_skin_ fold_ thickness	Hour_serum_ insulin	Body_mass _index	Diabetes_ Pedigree _function	Age	Class
0	151	90	46	0	42,1	0,371	21	2.
0	125	96	0	0	22,5	0,262	21	1.
0	126	86	27	120	27,4	0,515	21	1.
0	137	68	14	148	24,8	0,143	21	1.
0	138	60	35	167	34,6	0,534	21	2.
0	102	52	0	0	25,1	0,078	21	1.
0	198	66	32	274	41,3	0,502	28	2.
15	136	70	32	110	37,1	0,153	43	2.
0	101	64	17	0	21	0,252	21	1.
0	151	90	46	0	42,1	0,371	21	2.
0	125	96	0	0	22,5	0,262	21	1.

Tab 3.2 : Exemple d'agrégation par Max et min

On remarque dans le tableau 3.2 que la valeur minimale pour l'attribut « Number_pregnant » est la valeur 0 est cette valeur apparu plusieurs fois et la valeur maximale est 15 et cette valeur est apparu une seul fois

8-Outils de développement

Pour réaliser notre projet, nous avons utilisé différents outils. Comme outils de développement, on a choisi **Visuel studio Community 2015** qui est Un environnement de développement intégré et riche permettant de créer des applications époustouflantes pour Windows, Android et iOS, ainsi que des applications Web et des services cloud modernes. Outil gratuit, complet et extensible pour les développeurs qui construisent des applications non professionnelles, est avec comme langage

de programmation C# (C sharp) qui est un langage de programmation orientée objet, commercialisé par Microsoft depuis 2002. Il est un langage dont la syntaxe ressemble un peu au C++ ou au Java qui sont d'autres langages de programmation très populaires. Le C# est le langage phare de Microsoft. Il fait partie d'un ensemble plus important. Il est en fait une brique de ce qu'on appelle le « Framework .NET » ,et le système de gestion de base de données **SQL server 2014**.

9-Résultat et évaluation :

L'évaluation des résultats est faite en utilisant la matrice de confusion et en calculant l'Accuracy, Recall et la Précision puis les comparer avec les résultats obtenus par la fusion des deux table fragmentées.

-**Matrice de confusion** : Une Matrix de Confusion est un résumé des résultats de prédictions sur un problème de classification. Les prédictions correctes et incorrectes sont mises en lumière et réparties par classe. Les résultats sont ainsi comparés avec les valeurs réelles.[15]

	Predicted	
	Positive	Negative
Actuel True	**TP**	**FN**
Actuel False	**FP**	**TN**

Tab 3.3 : Matrice de confusion [15]

- **TP (True Positives)** : les cas où la prédiction est positive, et où la valeur réelle est effectivement positive.
- **TN (TrueNegatives)** : les cas où la prédiction est négative, et où la valeur réelle est effectivement négative.
- **FP (False Positive)** : les cas où la prédiction est positive, mais où la valeur réelle est négative.
- **FN (False Negative)** : les cas où la prédiction est négative, mais où la valeur réelle est positive.[15]

-Accuracy= $\frac{TP+TN}{TP+TN+FP+FN}$

-Recall = $\frac{TP}{TP+FN}$

-Precision = $\frac{TP}{TP+FP}$

Le tableau suivant représente une comparaison entre les résultats obtenu après fusion et les résultats obtenus après agrégation (Max et Min)

	Fusion	Aggregation
Accuracy	0.73	0.71
Recall	0.86	0.82
Precision	0.78	0.73

Tab 3.4:comparaison entre les résultats obtenu après fusion et les résultats obtenus après agrégation (Max et Min)

Le tableau suivant représente une comparaison entre les résultats obtenu après fusion et les résultats obtenus après agrégation (Valeur fréquente)

	Fusion	Aggregation
Accuracy	0.90	0.87
Recall	0.94	0.89
Precision	0.92	0.91

Tableau 3.5:comparaison entre les résultats obtenu après fusion et les résultats obtenus après agrégation (Valeur fréquente)

D'après les deux tableaux : tableau 3.2 et tableaux 3.3 on peut remarquer clairement que les résultats obtenus par fusion des tables et par agrégation sont approximativement proche. On remarque notamment que l'approche d'agrégation en utilisant la valeur fréquente a données des résultats meilleurs que ceux de l'approche d'agrégation en utilisant Max et Min. Ce ci peut-être expliqué par le fait que la valeur fréquente est plus proche de la réalité et permet d'éviter les valeurs exceptionnelles ce qui donne un meilleur résultat.

8-Conclusion :

Dans ce chapitre nous avons présenté une évaluation pour valider l'approche d'agrégation de base de données distribuées pour le data Mining. Nous avons appliqués notre approche sur un base de données distribuées qui contient deux classes puis tester les résultats de classification obtenu et comparé les résultat d'évaluation de notre approche avec les résultats d'évaluation obtenus après fusion. Les résultats obtenus après le calcul des différents critères d'évaluations Accuracy, Reccal et precision indiquent que l'approche proposée est plus efficace que l'approche par fusion qui souffre d'un majeur inconvénient qui est le temps d'exécution.

CONCLUSION GENERALE

L'application classique des algorithmes de Data Mining dans les environnements distribués n'est pas sans difficultés. La distribution des données via différents sites géographiquement distants rend la tâche de data mining difficile voire impossible si on applique la méthode traditionnelle qui consiste à réaliser des jointures de ces bases de données.

Plusieurs méthodes ont été proposées, dont ladite propositionalisation, qui consiste à appliquer des agrégations des différents tuples des bases de données distantes pour ensuite les projeter à la base de donné locale.

Nous avons appliqué deux agrégats sur un ensemble de données distribuées, à savoir l'agrégation par valeurs dominantes et l'agrégation par MinMax. Les comparaisons des résultats sur un algorithme de classification, en l'occurrence les arbres de décision, ont aboutit à une conclusion raisonnable : l'agrégation par valeurs dominantes est plus efficace car elle comporte la projection des tuples représentant une certaine force à l'opération de classification et rejettent ainsi les valeurs les moins fréquentes (dont celles représentant les valeurs exceptionnelles et le bruit) qui affectent négativement le résultat de classification.

L'objectif de notre projet étant de présenter les différentes approches utilisées pour extraire la connaissance à partir des données distribuées. Ensuite, d'implémenter l'approche par agrégation et d'effectuer des comparaisons d'un ensemble d'agrégats. Nous pouvons conclure que nous avons abouti à cet objectif, cependant, les travaux futurs peuvent toujours continuer par l'ajout et la comparaison d'autres agrégats pour savoir la meilleure technique d'agrégation efficace pour le data mining des bases de données distribuées.

Bibliographie

[1]- D Caragea, ASilvescu, V Honavar. A Framework for Learning from Distributed Data Using Sufficient Statistics and its Application to Learning Decision Trees. *Int J HybridIntellSyst*. 2004;1(1-2):80–89.

[2]- L. Chaabane,« fusion et fouille de donneesguidees par les connaissances : application a l'analyse d'image »,Doctorat,Universite Mohamedkhider – biskra,2013.

[3]- B. Fatiha, « DATA MINING DISTRIBUE», Magister, Université Mohamed Boudiaf (USTO-MB) Oran ,

[4]- L.Florence « Fouille de données relationnelles et spatiales – application au domaine hydroécologique »,Magister, Université de Strasbourg – INSA -- ENGEES,équipe BFO.

[5] - B. Hanane. Suivi et gestion répartie des clients d'une banque Cas de la Banque BADR-diplôme de Master en Informatique-Université de Université Abou BakrBelkaid– Tlemcen, 19 /09/ 2013.

[6]- B. HOUMADI, « Etude exploratoire d'outils pour le data mining », Master ,l'Université du Québec aTrois-Rivières,2007.

[7]-Knobbe, Arno & de Haas, Marc & P. J. M. Siebes, A. (2001). Propositionalisation and Aggregates. LNAI. 2168. 10.1007/3-540-44794-6_23.

[8]- Krogel MA., Rawles S., Železný F., Flach P.A., Lavrač N., Wrobel S. (2003) Comparative Evaluation of Approaches to Propositionalization. In: Horváth T., Yamamoto A. (eds) Inductive Logic Programming. ILP 2003. Lecture Notes in Computer Science, vol 2835. Springer, Berlin, Heidelberg

[9]- M.Rim. 2006. Systèmes de Gestion de Bases de Données Réparties & Mécanismes de Répartition avec Oracle-Ecole Supérieure de Technologie et d'Informatique à Carthage,2005-2006

[10]-G.Souhila «Proposition d'une Approche de Data Mining Distribuée Basée sur OPTICS »,Magister , Université Abderrahmane Mira de Bejaia,2008-2009

[11]- M.Tahar– « Intégration des liens utiles dans les arbres de décision pour la classification des bases de données distribuées »

Site web :

[12]-https://www.memoireonline.com/02/11/4278/m_Conception-et-realisation-dune-base-de-donnees-repartie-sous-oracle--cas-de-lhebergement-d2.html

[13]-https://www.memoireonline.com/05/10/3459/m_Bases-de-donnees-reparties-sous-Oracle1.html

[14]-https://www.kaggle.com/uciml/pima-indians-diabetes-database#diabetes.csv

[15]-https://www.lebigdata.fr/confusion-matrix-definition