Table des matières

Résumé 4

Abstract 5

ملخص 6

Table des matières 7

Liste des figures 10

Liste des abréviations 11

Liste des algorithmes 11

INTRODUCTION GENERALE 13

1. Contexte d'étude 14

2. Problématique 14

3. Contribution 15

4.Objectifs 15

5.Organisation du mémoire 16

PARTIE 1 : ETUDE THEORIQUE

CHAPITRE I : Réseaux bayésiens 17

1. Introduction : Pourquoi les RB ? 18

2. Intelligence artificielle 18

3. Relation entre RB et IA 19

4. Définition des RB 20

4.1 Exemple illustratif de RB 21

4.2 Représentation graphique

4.3 Pourquoi choisir les modèles orientés ? 24

5. Inférence 24

5.1 Définition 24

5.2 Méthode d'inférence : Jonction Tree……25

5.3 Les étapes de JT 25

6. Conclusion 28

CHAPITRE II : Apprentissage bayésien 29

1. Introduction 30

2. Apprentissage des paramètres 30

2.1. Apprentissage des paramètres à partir de donnée complètes 30

2.2. Apprentissage des paramètres à partir de données incomplètes 32

2.3 L'algorithme EM 32

3. Apprentissage de structure 33

3.1 L'algorithme PC-IC 33

4. Outils d'implémentation 34

5. Conclusion 35

PARTIE 2 : PARTIE PRATIQUE 36

CHAPITRE I : Analyse et conception 37

1. Introduction 38

2. Présentation de l'application 38

3. Besoins techniques 38

4. Identification des cas d'utilisation 39

5. Documentation des cas d'utilisation 40

6. Diagramme de séquence 43

7. Documentation des diagrammes de séquences 44

8. Architecture globale 45

9. Configuration matérielle du système 45

10. Architecture globale de l'application 46

11. Conclusion 47

CHAPITRE II : Réalisation et mise en oeuvre 48

1. Introduction 49

2. Concepts de développement 49

2.1. Langage de programmation 49

2.2. Base de données **49**

2.2.i. Base de données MySQL **50**

2.2.ii. Système de gestion de base de données **50**

2.2.iii. Système de gestion de Base de donnée phpMYadmin **51**

2.2.iv Langage de manipulation de bases de données **51**

2.3. Environnement de développement **52**

2.4. Outils utilisés **53**

2.4.i. XAMP **53**

2.4.ii. HeidiSQL **54**

2.4.iv. Netica Java-API **54**

2.4.v. MySQL connector Java **55**

3. Description de l'application **55**

4. Conclusion **68**

CONCLUSION GENERALE & PERSPECTIVES **69**

Conclusion générale **70**

Perspectives **71**

BIBLIOGRAPHIE **72**

ANNEXES **76**

Table des figures

Partie théorique :

Figure II.1 : Graphe réseau bayésien
Figure II.2 : Exemple d'un graphe
Figure II.3 : Supports graphiques
Figure II.4 : Graphe dirigé
Figure II.5 : Types de graphes
Figure II.6: étapes arbre de jonction
Figure II.7 : La moralisation du graphe « Arbre de jonction»
Figure II.8 La triangulation « Arbre de jonction»
Figure II.9 arbre de jonction
Figure III.1 : Tableau comparatif entre les outils

Partie pratique :

Figure I.1 : Diagramme de cas d'utilisation.
Figure I.2: Représentation des classes dans le diagramme de séquence.
Figure I.3: Diagramme de séquence (processus d'authentification)
Figure I.4: Diagramme de séquence (processeur d'application)
Figure I.5 : la configuration matérielle de notre système
Figure I.6 : l'architecture générale de l'application
Figure II.1. Interface de Netbeans
Figure II.2. Interface de XAMPP
Figure II.3. Interface de HeidiSQL
Figure II.4. Logo Application
Figure II.5. Interface de l'application
Figure II.6. Interface de création de nouveau compte
Figure II.7. Message d'erreur – Nom d'utilisateur existe déjà –
Figure II.8. Message d'erreur – Les 2 mots de passe ne sont pas identiques –
Figure II.9. Interface d'authentification
Figure II.10. Message d'erreur – Les informations de connexion sont incorrectes –
Figure II.11. Interface principale
Figure II.12. Interface de données initiales
Figure II.13. Interface de graphe du réseau bayésien
Figure II.14. Exemple de déclaration d'un nœud du réseau bayésien
Figure II.15. Exemple de déclaration des éléments dans un fichier dne

Figure II.16. Interface de saisie d'informations de cas de simulation

Figure II.17. Le contenu de fichier infocas.cas

Figure II.18. Le contenu de fichier ChestClinic.cas

Figure II.19. Le code source qui permet de faire l'apprentissage

Figure II.20. L'interface graphique pour saisir les informations d'un patient

Figure IV.21. L'interface graphique qui indique la probabilité d'avoir le diabète

Figure II.22. Liste des patients

Figure II.23. À propos de l'application

Liste des abréviations :

RB : Réseaux bayésiens

JT : Jonction tree (arbre de jonction)

EM: Expectation Maximisation

PC-IC: « PC » pour Peter et Clark « IC » algorithme de principe similaire

API: Application Programming Interface

BDD: Base De Données

SGBD: Système de Gestion de Base de Données

SQL: Structured Query Language

IA: Intelligence artificielle

Liste des algorithmes :

- Algorithme Expectation Maximisation
- Algorithme PC-IC

INTRODUCTION GENERALE

Partie 01

ETUDE THEORIQUE

Rapport-Gratuit.com

1. Contexte d'étude :

Une des grandes problématiques de notre époque est de traiter la grande quantité de données qui est mise à notre disposition (notamment grâce à l'informatique) pour en extraire de l'information : un statisticien qui veut analyser un tableau de mesures pour une population donnée se retrouve face à une immense masse d'informations de laquelle il doit extirper de la connaissance. Il serait donc intéressant d'avoir un modèle effectuant le lien entre les observations et la réalité pour un objectif précis, et cela, même lorsque les observations sont incomplètes et/ou imprécises.

2. Problématique :

L'objectif de ce travail est de proposer comme méthode : « les réseaux bayésiens » pour extraire automatiquement de la connaissance à partir de données statistiques c'est-à-dire réelles. Dans ce travail il s'agit, entre autre, d'identifier automatiquement un environnement. Ce dernier sera modélisé par la distribution de probabilités qui a généré les exemples. Une fois cet environnement reconnu, la machine sera alors capable de générer de nouvelles données indiscernables de celles créées par la nature, ou encore d'extraire de la connaissance du modèle obtenu.

Le but étant de s'appuyer sur une base d'exemples de fonctionnement d'un système donné (complexe, chaotique etc…), cette démarche peut alors avoir un intérêt dans une problématique d'intelligence artificielle puisqu'une fois un robot aura acquis une connaissance sur son environnement grâce à la méthode des réseaux bayésiens, il l'utilisera pour interagir plus efficacement avec celui-ci. [11]

Les réseaux bayésiens sont parmi les nombreux formalismes qui ont été créés pour cet objectif prenant en compte l'incertitude ; enjeu important dans le domaine de l'intelligence artificielle. En effet, il faut être capable de déterminer les causes ou les conséquences les plus vraisemblables à partir d'un état particulier du système pour en déduire les meilleures actions possibles. [12]

Outre cet avantage, afin de mieux illustrer leur intérêt, nous allons prendre l'exemple du corps humain : Lorsqu'un individu est malade, nous allons observer pour celui-ci un certain nombre de grandeurs (tension, fièvre, etc…) en fonction desquelles a priori le médecin va donner son diagnostic. Ce diagnostic est ici évalué en fonction d'un nombre restreint de paramètres, or il se peut que deux individus aient la même forme (les mêmes valeurs pour toutes les grandeurs observées) et que l'un d'eux soit malade, tandis que l'autre est sain.

Les modèles probabilistes ont cet avantage de fournir systématiquement une probabilité à chaque état (ici sain ou malade). Celle-ci peut alors être considérée comme un indice de confiance dans le résultat (par exemple, cet individu a 85% de (mal) chance d'être malade). Bien sûr un tel diagnostic n'est pas satisfaisant d'un point de vue éthique pour décider d'administrer ou non un traitement. Pour ce type d'exemple, il est évident que n'importe quel système ayant ces propriétés servira essentiellement à aider le praticien dans sa prise de décision, par exemple, en le confortant sur son avis.

3. Contribution :

Les réseaux bayésiens que nous allons définir un peu plus loin dans ce mémoire sont des outils permettant alors de résoudre le type de tâche énoncée précédemment (exemple du corps humain). Seulement pour qu'une tâche soit résolue efficacement, il faut qu'un modèle soit correctement construit. Or, comme les systèmes sont gérés par un grand nombre de paramètres, nous ne pouvons demander à un homme seul, ou encore à une équipe de les construire [22]. Nous nous proposons donc de développer certaines méthodes d'apprentissage automatique de ce modèle probabiliste à partir de la disponibilité de données massives, de connaissances expertes et d'un historique : une base d'exemples de fonctionnement et/ou de non fonctionnement du système ou de systèmes similaires. Nous avons pris comme exemple explicatif « le diabète » qui est une maladie courante.

4. Objectifs :

Les réseaux bayésiens prennent une ampleur dans la modélisation de données, nos objectifs à travers notre mémoire sont multiples :

- Aide à la compréhension des réseaux bayésiens et montrer leur utilité comme outil pour modéliser toute sorte de phénomènes même les plus complexes.
- Explication simple et détaillée de la démarche à suivre pour l'élaboration d'un réseau bayésien.
- Donner par la suite une image concrète à toutes ces étapes via l'application conçue.
- Montrer en aval l'intérêt de notre choix quant aux réseaux bayésiens en citant nos perspectives ultérieures.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

5. Organisation du mémoire :

Notre mémoire est organisé comme suit :

Partie 01 : ETUDE THEORIQUE

- Chapitre I : Introduction générale.
- Chapitre II : Se propose tout d'abord de définir la notion de réseaux bayésiens comme modèle graphique probabiliste et qui s'est révélé être un outil très pratique pour la représentation de connaissances incertaines et le raisonnement à partir d'informations incomplètes puis de donner un aperçu de l'utilisation de ces modèles pour répondre à différentes requêtes (notion d'inférence ou de raisonnement probabiliste).
- Chapitre III : Nous abordons dans cette partie le problème de l'apprentissage des réseaux bayésiens à partir de données complètes ou incomplètes, en commençant par la détermination des distributions de probabilité conditionnelles définies par un graphe donné (apprentissage des paramètres), et en essayant ensuite de déterminer le graphe même à partir des données (apprentissage de la structure).

Partie 02 : ETUDE PRATIQUE

Pour finir, la partie pratique concerne la présentation et se divise en deux phases :

- Chapitre I : Analyse et conception pour l'aspect graphique.
- Chapitre II : Mise en œuvre et réalisation pour l'aspect physique.

CHAPITRE I

Réseaux Bayésiens

1. Introduction : Pourquoi les réseaux bayésiens :

Les réseaux Bayésiens sont actuellement une des techniques les plus intéressantes de l'intelligence artificielle car ils allient la lisibilité d'une représentation de la connaissance par un graphe causal intuitif à l'efficacité d'une représentation « distribuée » des données qui tient compte de l'incertitude dans le raisonnement. Ils sont utilisés dans bon nombre d'applications. Nous définissons, dans cette section, quelques notions de base qui nous seront utiles dans cette thèse.

2. L'intelligence artificielle :

Avant toutes choses, on peut se demander ce qu'est l'intelligence artificielle, quelle relation a-t-elle avec l'intelligence humaine, comment ce domaine de recherche a vu le jour et a émergé. Afin de comprendre la chronologie des choses, nous allons ici donner un aperçu historique avant d'expliquer les différents principes de ce concept nécessaire.

Le terme « intelligence artificielle », créé par John McCarthy, est souvent abrégé par le sigle « IA ». Il est défini comme « la construction de programmes informatiques qui s'adonnent à des tâches qui sont, pour l'instant, accomplies de façon plus satisfaisante par des êtres humains car elles demandent des processus mentaux de haut niveau tels que : l'apprentissage perceptuel, l'organisation de la mémoire et le raisonnement critique ». [27] On y trouve donc le côté « artificiel » atteint par l'usage des ordinateurs ou de processus électroniques élaborés et le côté « intelligence » associé à son but d'imiter le comportement humain. Cette imitation peut se faire dans le raisonnement, par exemple dans les jeux ou la pratique de mathématiques, dans la compréhension des langues naturelles, dans la perception visuelle (interprétation des images et des scènes), auditive (compréhension du langage parlé) ou par d'autres capteurs, dans la commande d'un robot dans un milieu inconnu ou hostile.

C'est donc une science expérimentale dont le but est d'étudier le cerveau humain pour tenter de reproduire ses pensées. Encore, des scientifiques recherchent à créer des systèmes qui atteignent un idéal d'*intelligence humaine*. Cependant il existe un certain nombre de définitions de l'IA qui différent :

Des définitions qui lient l'IA à un aspect humain de l'intelligence, et celles qui la lient à un modèle idéal d'intelligence, non forcément humaine, nommée rationalité : C'est une logique qu'approfondissent les scientifiques. Par exemple, la perception de tout ce qui l'entoure en fait partie. C'est ici que la technologie est repoussée à ses limites. De plus, le comportement rationnel reprend l'idée d'une pensée rationnelle de l'IA mais est plus générale car raisonner

logiquement peut être une façon d'agir rationnellement. Néanmoins, les scientifiques recherchent ce standard de rationalité qui ne peut être atteint en pratique, car la technologie actuelle ne permet pas de réaliser des calculs aussi complexes.

3. Relation entre intelligence artificielle et réseaux bayésiens :

L'un des enjeux principaux dans le domaine de la recherche en Intelligence Artificielle est d'être capable de concevoir et de développer des systèmes dynamiques et évolutifs. De ce fait, ces derniers doivent être équipés de comportements intelligents qui peuvent apprendre et raisonner. Mais dans la plupart des cas, la connaissance acquise n'est pas toujours adéquate pour permettre au système de prendre la décision la plus appropriée. Pour répondre à ce genre de questions, plusieurs méthodologies ont été proposées, mais seules les approches probabilistes s'adaptent mieux non seulement au raisonnement avec la connaissance et la croyance incertaine, mais aussi à la structure de la représentation de la connaissance. Ces approches probabilistes sont appelées "réseaux bayésiens", mais sont aussi connues sous le nom de "belief networks", "causal networks".

Les réseaux bayésiens sont la combinaison des approches probabilistes et de la théorie de graphes. Autrement dit, ce sont des modèles qui permettent de représenter des situations de raisonnement probabiliste à partir de connaissances incertaines. Ils sont une représentation efficace pour les calculs d'une distribution de probabilités, de plus ils constituent une technique d'acquisition, de représentation et de manipulation de connaissances, et on les utilise pour leur capacité d'effectuer des **inférences** dans un contexte d'incertitude et aussi pour leurs algorithmes **d'apprentissage** afin de prévoir, contrôler et simuler le comportement d'un système, analyser des données et prendre des décisions[19]. Leurs domaines d'application sont variés :

• Marketing/Finance (modélisation de risques) : - Détection de fraudes - Optimisation de portefeuilles

• Informatique : - Microsoft : différents assistants de résolution de problèmes, l'assistant Office - Evaluation de la qualité pour des développements logiciels - Reconnaissance de la parole, d'objets 3D… - Diagnostic de pannes

• Médecine : - Aide au diagnostic de problèmes précis, - Surveillance transfusionnelle

• Industrie : - NASA Vista : aide au diagnostic de pannes en temps réel pour les systèmes de propulsion - Lockheed Martin : système de contrôle d'un véhicule sous-marin autonome - EDF : modélisation de groupes électrogènes

• Environnement : - Prédiction des pics d'ozone - Prévisions météo

Dans le présent travail, les RB sont appliqués au domaine médical. [15]

4. Définition des RB :

Probabilité conditionnelle : Etant donné que l'événement B s'est produit, la probabilité que l'événement A se produise est x et s'écrit : P(A|B) = x.

Les réseaux bayésiens sont initialement connus comme « réseaux de relations », mais sont nommés "Bayes" après la mort de Révérend Thomas Bayes, 1702-1761, théologien et mathématicien britannique qui a écrit une loi de base de probabilité, maintenant appelée le théorème de Bayes. Les RB sont basés sur ce théorème, ce sont des modèles qui permettent de représenter des situations de raisonnement probabilistes exprimés par la formule :

$$P(A/B) = \frac{P(B/A).P(B)}{P(A)}$$

Ainsi, les RB associent une partie qualitative que sont les graphes et une partie quantitative représentant les probabilités conditionnelles associées à chaque nœud du graphe relativement au parent. La partie qualitative exprime des indépendances conditionnelles entre variables et des liens de causalités et ce grâce à un graphe orienté acyclique dont les nœuds correspondent à des variables aléatoires. La partie quantitative est constituée de tables de probabilités dans le cas discret ou de distribution gaussiennes dans le cas continu. Un réseau bayésien B = {G , P} est donc défini par un graphe dirigé, un espace probabiliste est un ensemble de variables aléatoires. Le graphe est sans circuit G = (X, E) où X est l'ensemble des nœuds (ou sommets) et E, l'ensemble des arcs. L'espace probabiliste est tel que (**Ω,** P) où **Ω** est l'univers des probabilités et P l'ensemble de variables aléatoires X= $\{X_{1,\ldots,}X_n\}$ associées aux noeuds du graphe et tel que:

$$P(X_1,\ldots,X_n) = \prod_{i=1}^{n} P(X_i / Pa(X_i))$$

Dans cette expression, Pa(Xi) est l'ensemble des parents du nœud X_i dans G.[10]

Ainsi, un réseau bayésien est un modèle probabiliste graphique permettant d'acquérir, de capitaliser et d'exploiter des connaissances, né du besoin de créer des systèmes experts à base de probabilités

Dans l'exemple ci-contre, la probabilité jointe de toutes les variables est :

P(A,B,C) = P(A| B)× P(B)× P(C | B)

En général, étant donné les nœuds X= X_1,..., X_n, la probabilité jointe pour les réseaux bayésiens est :

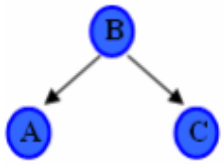


Figure II.1 : Graphe réseau bayésien

La probabilité jointe de toutes les variables est le produit de toutes les probabilités de chaque variable et ainsi les valeurs de ses parents.

- L'aspect graphique des modèles graphiques permet de représenter les relations entre les attributs clairement et intuitivement.
- Leurs orientations (si elles existent) peuvent représenter des relations de cause à effet.
- Les modèles probabilistes sont capables de gérer l'incertain et l'imprécis.
- Le rôle des graphes dans les modèles probabilistes est triple :
- fournir un moyen simple et efficace d'exprimer des hypothèses,
- donner une représentation économique des fonctions de probabilité jointe ,
- faciliter l'inférence à partir d'observations.
- Ces modèles deviennent incontournables lorsque nous avons affaire à un problème sujet à l'incertain. Les modèles probabilistes sont alors utiles pour
- l'extraction de connaissance probabiliste : c'est-à-dire trouver quelles variables sont corrélées, dépendantes ou conditionnellement indépendantes,
- le diagnostic : l'évaluation de P(causes|symptômes),
- la prédiction : l'évaluation de P(symptômes|causes),
- la classification : le calcul de max classes P(classe|observations).

$$P(X) = \prod_{i=1}^{n} P(X_i \mid parents(X_i)) \cdot$$

4.1 Exemple illustratif des RB :

Afin de mieux comprendre le concept, nous avons instauré un exemple qui conférera au modèle bayésien une image plus réelle. Il s'agit d'une modélisation des connaissances d'un médecin spécialiste des maladies pulmonaires Pour mieux comprendre en quoi les réseaux bayésiens constituent un bon outil d'aide à la décision, voyons sur un exemple concret une de leurs applications possibles. Considérons une clinique spécialisée dans le traitement des

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

maladies pulmonaires. Lorsqu'un patient entre dans cette clinique, un médecin doit diagnostiquer une maladie, et donc un traitement, en fonction des symptômes et de l'histoire du patient. Pour l'aider dans sa décision, il pourrait modéliser ses connaissances par un réseau bayésien. La construction d'un réseau bayésien (figure ci-contre) se fait en quatre étapes détaillées ci-dessous :

1- Création des variables représentant l'univers On suppose que les trois maladies possibles sont le cancer du poumon, la bronchite et la tuberculose. On suppose qu'on peut faire un examen de radiographie, et on considère que les caractéristiques importantes du patient sont son âge, son aspect fumeur ou non-fumeur, un éventuelle voyage en Asie et ses éventuels problèmes respiratoires chroniques. On introduit de plus une variable TbOuCa déterministe pour factoriser le graphe, elle vaut oui s'il le patient a un cancer ou une tuberculose.

2- Définition de l'ensemble des valeurs possibles pour chaque variable Par exemple, l'âge pourrait prendre trois valeurs : jeune, adulte, senior; et la bronchite par oui ou non. [15]

3- Définition des relations de dépendances entre les variables Par exemple, une éventuelle tuberculose est en relation directe avec un éventuel voyage en Asie.

4- Attribution des probabilités conditionnelles Par exemple, un patient de la clinique qui a voyagé en Asie a 1% de risque d'avoir la tuberculose. Si le patient dit qu'il a été en Asie, alors la variable Visite Asie a une probabilité de 100% d'être oui, ensuite on recalcule la nouvelle distribution de probabilités pour les autres variables qui en découlent, ce qui donne des hypothèses plus ou moins probables sur la pathologie réelle du patient.

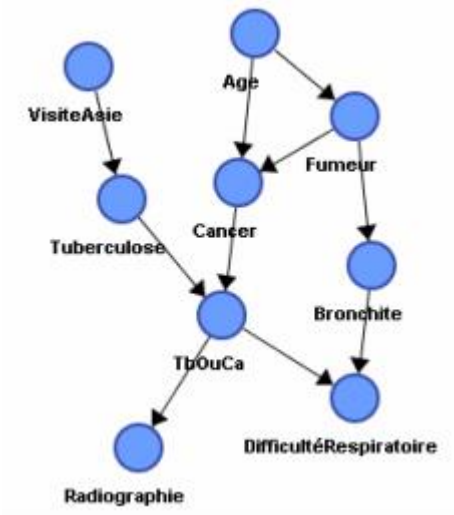


Figure II.2 : Exemple d'un graphe

4.2 Les différents supports graphiques :

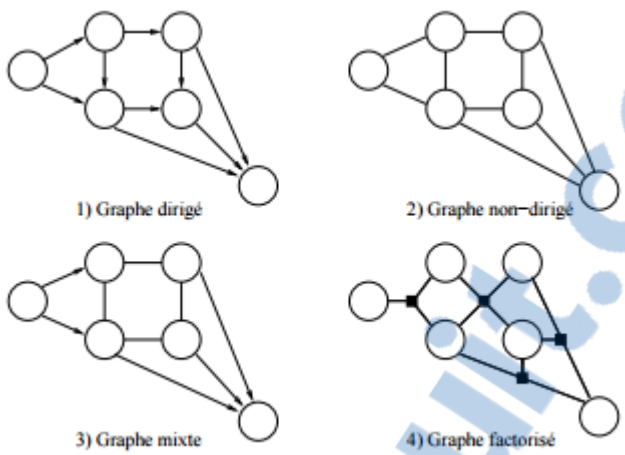


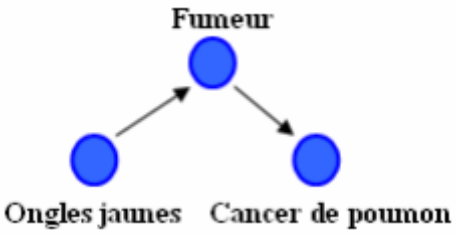
Figure II.3 : Supports graphiques

Les supports des modèles graphiques probabilistes peuvent être de plusieurs types, qui ont des pouvoirs expressifs différents.

Les graphes dirigés sans circuit :

Ils peuvent, grâce à leurs V-structures modélisé des variables qui sont indépendantes, mais qui deviennent dépendantes étant donné l'état d'une troisième variable. On parle aussi de notion de causalité : Dans les réseaux bayésiens, deux nœuds qui représentent deux faits différents peuvent être en relation causale sans que l'un implique l'autre. Graphiquement, chaque parent a un effet sur ses fils. La notion de causalité joue un rôle très important pour construire les réseaux bayésiens.

- Graphe de causalité : Par exemple sur un patient, si on constate qu'il a des ongles jaunes, alors on pourrait dire qu'il fume et en même temps, s'il fume, il a une forte chance d'avoir le cancer de poumon. Cela peut être représenté sous forme d'un graphe de causalité.,



- **Figure II.4 :** Graphe dirigé

Rapport-Gratuit.com

- Les graphes non dirigés : Ils ne peuvent pas modéliser le cas précédent, par contre ils peuvent modéliser une loi qui vérifierait A⊥⊥C|{B, D} et B⊥⊥D|{A, C}. Or un graphe orienté ne peut pas modéliser ces deux indépendances conditionnelles en même temps.
- Les graphes mixtes (ou chain graphs) : Ils ont l'avantage de cumuler les avantages des graphes dirigés et non dirigés.
- Les graphes factorisés : Ils sont une autre manière de modéliser les graphes non dirigés. Ils partagent alors le même pouvoir expressif.

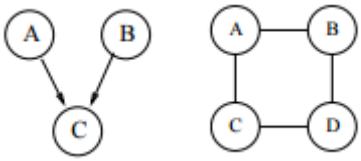


Figure II.5 : Types de graphes

4.3 Pourquoi choisir des modèles orientés ?

Le pouvoir expressif des modèles graphiques orientés, par rapport à ceux non dirigés, n'est ni meilleur, ni moins bon du point de vue des dépendances conditionnelles mais ils peuvent se prêter à une interprétation causale. Nous avons préféré étudier les modèles orientés car ils sont plus intuitifs et plus visuels. Les orientations guident sur la manière d'interpréter la structure, tout comme elles guident lors de l'inférence pour savoir comment faire circuler l'information. Ces modèles sont plus simples à construire pour les experts d'un domaine précis qui raisonnent souvent par mécanisme de causes à effets. Il serait alors dommage de s'abstraire de cette information en leur demandant de construire des modèles non orientés. [20]

5. Inférence :

5.1 Définition :

Une fois qu'un réseau bayésien a été construit pour rendre compte d'un domaine, on cherche souvent à l'utiliser pour déterminer des probabilités correspondantes à certains événements, certaines questions, certaines dépendances. Dans ces cas, les inférences permettent effectivement de calculer la probabilité de chaque nœud lorsque les autres variables sont inconnues: En effet, l'inférence bayésienne est basée sur l'utilisation d'énoncés probabilistes, qui dans le cas général sont trouvés par des experts étudiant un système qui leurs ait connu. Ces énoncés doivent être clairs et précis afin d'éviter toute confusion dans les relations de dépendance qui en découleront.

L'inférence bayésienne est particulièrement utile dans les problèmes d'induction, car se basant sur des cas particuliers et n'a de validité qu'en terme probabiliste. Les méthodes bayésiennes se distinguent des méthodes dites standard par l'application systématique de règles formelles de transformation des probabilités. [16]

On cherche à induire sur un système bayésien aussi bien par le haut que par le bas, aussi bien les conséquences que les causes, du graphe de dépendance car le théorème de Bayes permet d'inverser les probabilités. [18] C'est-à-dire que si l'on connaît les conséquences d'une cause, l'observation des effets permet de remonter aux causes, c'est l'effet d'induction « bottom-up ». Sachant aussi qu'une lecture littéral du théorème de Bayes permet une induction « top-down », c'est à dire à partir des causes en déduire les conséquences.

5.2 Méthodes inférence :

Il existe cependant quelques méthodes permettant d'effectuer ce calcul tels que : le conditionnement global, méthode approchée et Arbre de jonction « Jonction Tree » qui sont les premiers algorithmes d'inférence pour les réseaux bayésiens basés sur une architecture à passage de messages limités aux arbres. Dans cette technique, à chaque nœud est associé un processeur qui peut envoyer des messages de façon asynchrone à ses voisins jusqu'à ce qu'un équilibre soit atteint, en un nombre fini d'étapes. Cette méthode a été depuis étendue aux réseaux quelconques pour donner l'algorithme de l'arbre de jonction (JT).

5 .3 Etapes de JT :

Cet algorithme s'applique en plusieurs étapes de transformation du graphe:

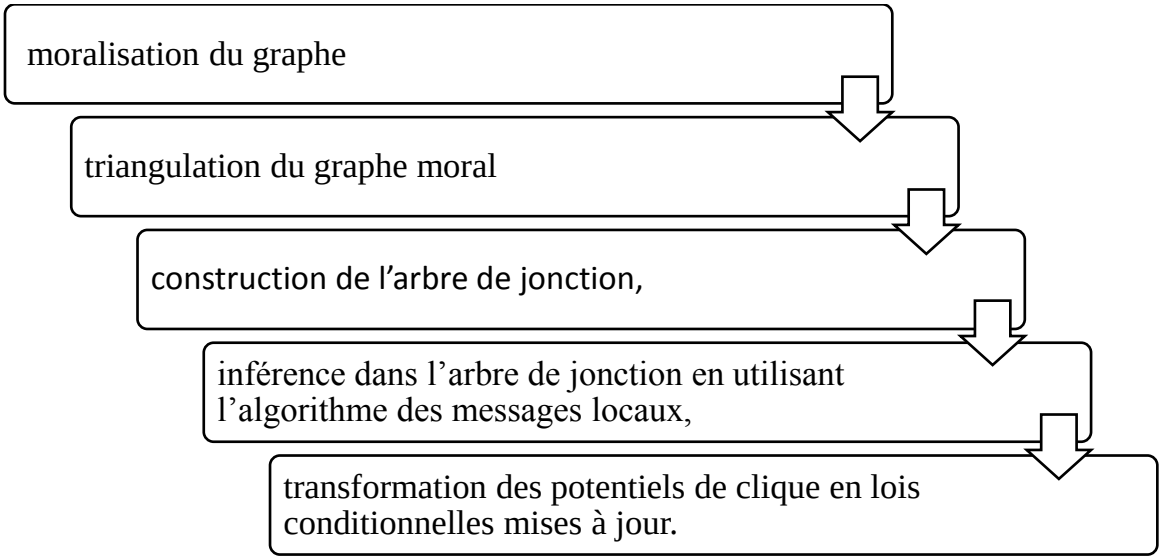


Figure II.6: étapes arbre de jonction.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Les étapes sont organisées dans les deux phases suivantes :

- **la phase de construction** : elle nécessite un ensemble de sous-étapes permettant de transformer le graphe initial en un arbre de jonction, dont les nœuds sont des clusters (regroupement) de nœuds du graphe initial. Cette transformation est nécessaire, d'une part pour éliminer les boucles du graphe et d'autre part, pour obtenir un graphe plus efficace quant au temps de calcul nécessaire à l'inférence, mais qui reste équivalent au niveau de la distribution de probabilité représentée.
- Cette transformation se fait en trois étapes :
- la moralisation du graphe
- la triangulation du graphe et l'extraction des cliques qui formeront les nœuds du futur arbre
- la création d'un arbre couvrant minimal, appelé arbre de jonction
- **la phase de propagation** : il s'agit de la phase de calcul probabiliste à proprement parler où les nouvelles informations concernant une ou plusieurs variables sont propagées à l'ensemble du réseau, de manière à mettre à jour l'ensemble des distributions de probabilités du réseau. Ceci ce fait en passant des messages contenant une information de mise à jour entre les nœuds de l'arbre de jonction précédemment construit. A la fin de cette phase, l'arbre de jonction contiendra à la distribution de probabilité sachant les nouvelles informations, c'est-à-dire P(Uje) où U représente l'ensemble des variables du réseau bayésien et e l'ensemble des nouvelles informations sur lesdites variables.[24]

- **La moralisation**

La moralisation se décompose suivant les étapes suivantes :

- Mariage des nœuds parents : pour les nœuds possédant plusieurs parents, liaison des parents deux à deux avec des arcs supplémentaires.
- Récupération du squelette du graphe ainsi obtenu, Nous obtenons alors un graphe non dirigé dit moralisé.

Le déroulement de cet algorithme sera illustré sur l'exemple précédent :

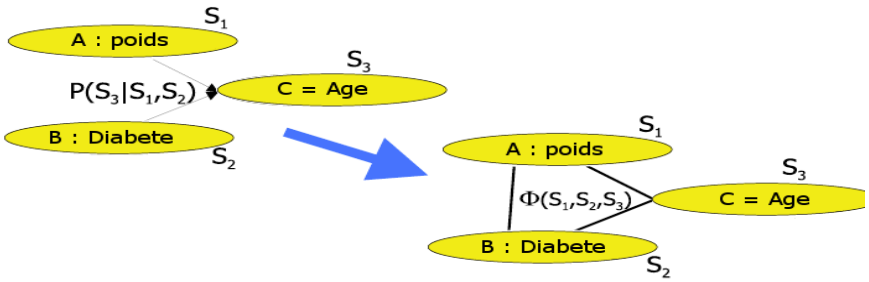


Figure II.7 : La moralisation du graphe «Arbre de jonction»

- **Triangulation**

La deuxième étape consiste à trianguler le graphe moral Gm et en extraire des cliques de nœuds, qui sont des sous graphes complets de G. Ces cliques formeront les nœuds de l'arbre de jonction utilisé pour l'inférence. Il faut donc ajouter suffisamment d'arcs au graphe moral G_m afin d'obtenir un graphe triangulé G_T .

L'algorithme de triangulation opère d'une manière très simple. Un graphe est triangulé si est seulement si l'ensemble de ses nœuds peuvent être éliminés. Un nœud peut être éliminé si tous ses voisins sont connectés deux à deux. Donc un nœud peut être éliminé s'il appartient à une clique dans le graphe. Une telle clique forme un nœud pour le futur arbre de jonction qui est en train d'être construit. Ainsi, il est possible de trianguler le graphe et de construire les nœuds de l'arbre de jonction en même temps en éliminant les nœuds dans un certain ordre. Si aucun nœud n'est éliminable, il faut en choisir un parmi les nœuds restants et rajouter les arcs nécessaires entre ses voisins pour qu'il devienne éliminable. Le nœud choisi sera celui pour lequel l'espace d'état de la clique formée sera le plus petit possible. En effet, plus les cliques sont petites, plus l'espace de stockage, et a fortiori le temps de calcul, sont réduits.

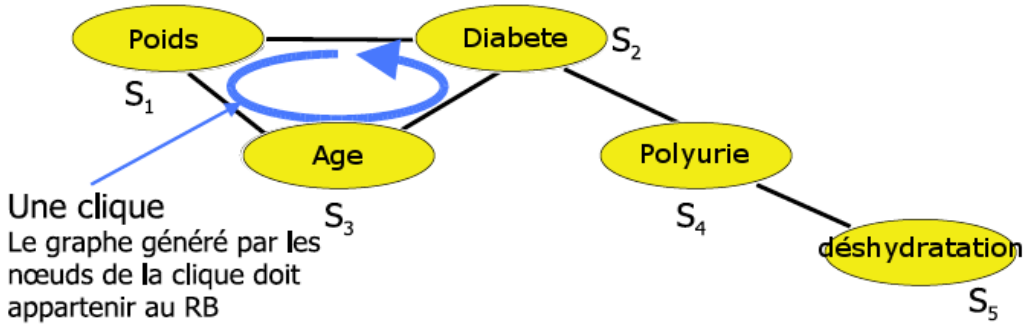
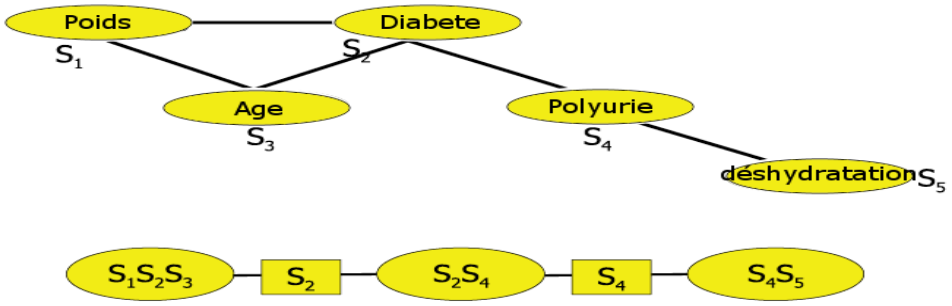


Figure II.8 La triangulation « Arbre de jonction»

Si le graphe est moralisé et triangulé, alors les cliques peuvent être organisées en un arbre de jonction.



$$P(S) = \Phi(S1, S2, S3)\Phi(S2, S4)\Phi(S4, S5)$$

Figure II.9 arbre de jonction

6. Conclusion :

Cette introduction très générale sur les modèles graphiques existants et les méthodes d'inférence qui leur sont associées permet de comprendre ce qu'est un réseau bayésien ainsi que les avantages que peuvent posséder ces modèles. Nous avons également passé en revue différentes déclinaisons de ces derniers. Ceci nous a permis de comprendre que les réseaux bayésiens sont un outil souple et performant de gestion de l'incertitude :

- ils permettent de séparer aspects qualitatifs et aspects quantitatifs des modèles probabilistes grâce à leur présentation graphique et fournissent ainsi des méthodes simples de construction ;
- ils ont la souplesse suffisante pour fusionner différentes sources d'informations (connaissances à partir de bases de données ou d'experts) ;
- ils ont la capacité d'intégrer les modifications de l'environnement d'étude ;
- ils peuvent gérer des séquences temporelles (virtuellement infinies) d'évaluations et de prises de décision ;
- ils permettent de grandes rapidités de réaction car, bien que les propagations requièrent des calculs assez fins, ceux-ci sont bien explicités dans la littérature et peuvent être effectués en temps réel.

Néanmoins, ces modèles peuvent alors s'avérer difficiles à construire manuellement lorsque le nombre de nœuds augmente. Pour cela, des méthodes de créations automatiques de réseaux bayésiens doivent être développées. Certaines heuristiques existent déjà avec des avantages et inconvénients issus du choix d'un compromis entre l'exhaustivité des heuristiques et leur complexité. Dans la prochaine partie, nous nous proposons de comparer empiriquement certaines techniques existantes pour découvrir la structure d'un réseau bayésien à partir de bases d'exemples complètement observées. Nous proposons et testons alors une méthode d'initialisation de certaines méthodes classiques permettant de garantir leur stabilité tout en conservant, voire améliorant, la qualité de leurs résultats.

CHAPITRE II

Apprentissage Bayésien

1. Introduction :

La construction de réseaux bayésiens consiste à trouver une structure ou graphe et estimer les paramètres (probabilités conditionnelles). Cependant il n'est pas toujours évident devant une très grande base de données qu'un expert extrait une structure adoptée à une telle quantité de données nécessaires à l'inférence dans un graphe. Il est intéressant dans certains cas de déterminer ces paramètres à partir d'une base d'exemples existante. C'est la méthode d'apprentissage qui permet de pallier ce problème.

L'apprentissage d'un RB à partir de données se décompose en deux phases :

1. Premier phase : Apprentissage de la structure : trouver la meilleure structure par apprentissage automatique à partir de données observées.
2. Deuxième phase : Apprentissage des paramètres : estimer des paramètres à partir de données observées.

2. Apprentissage des paramètres :

2.1 Apprentissage à partir de donnée complètes :

L'apprentissage de paramètres consiste à supposer que la structure du réseau est fixe et donc à déterminer les probabilités conditionnelles de chaque variable qui se trouve dans le réseau. Les données disposées peuvent être complètes ou incomplètes, discrètes ou continues. Pour chaque cas, l'algorithme d'apprentissage des paramètres diffère. Dans le cas où toutes les variable sont observées et discrètes, la méthode la plus simple et la plus utilisée pour estimer les paramètres est l'estimation statique de la probabilité d'un évènement par la fréquence d'apparition de l'évènement dans la base de donnée. Cette méthode est appelée maximum de vraisemblance, donnée par l'expression suivante :

$$P(X_i = x_k / \Pi_i = \pi_{ij}) = \hat{\theta}^{MV}_{ijk} = \frac{N_{ijk}}{\sum_{k=1}^{ri} N_{ijk}}$$

N_{ijk} : le nombre d'occurrence simultané dans la base de $X_i=x_k$ et $\Pi_i = \pi_{ij}$

Avec : $k \in 1 \ldots r$ et $j \in 1 \ldots j$

L'inconvénient de cette méthode est qu'on peut avoir une probabilité nulle à cause de la non apparition d'un évènement dans la base de donnée, ce qui est faux en réalité. Pour remédier à ce problème on fait recours à d'autres approches dites méthodes bayésiennes. Ces méthodes sont connues sous le non maximum a posteriori (MAP) et espérance a posteriori (EAP).

Supposons que les paramètres θ, admettent une densité de probabilité exponentiel de Dirichlet alors on peut écrire l'expression ci-dessous :

$$P(\theta_i / \alpha_1, \ldots \alpha_n) = \frac{\Gamma(\sum_{j=1}^{ri} \alpha_i}{\prod_{i=1}^{ri} \Gamma \alpha_i} \times \prod_{i=1}^{r} \theta^{\alpha_i - 1}$$

Avec : Γ la fonction gamma d'Euler

Γ(x+1)=x Γ(x) et Γ(1)=1 dans R

Le maximum a posteriori est donnée par :

$$P(X_i = x_k / \Pi_i = \pi_{ij}) = \hat{\theta}_{ijk}^{MAP} = \frac{N_{ijk} + \alpha_{ijk} - 1}{\sum_{k=1}^{ri} N_{ijk} + \alpha_{ijk} - 1}$$

L'espérance a posteriori est donnée par :

$$P(X_i = x_k / \Pi_i = \pi_{ij}) = \hat{\theta}_{ijk}^{EAP} = \frac{N_{ijk} + \alpha_{ijk}}{\sum_{k=1}^{ri} N_{ijk} + \alpha_{ijk}}$$

Un autre estimateur appelé estimateur de la place qui suppose que les paramètres de Dirichlet égale à 1, est défini par :

$$P(X_i = x_k / \Pi_i = \pi_{ij}) = \hat{\theta}_{ijk}^{EL} = \frac{N_{ijk} + 1}{\sum_{k=1}^{ri} N_{ijk} + 1}$$

Dans la plupart des travaux, les calculs de probabilités effectués reposent sur l'utilisateur de variables discrètes dans les distributions de probabilités conditionnelles par rapport aux parents sont des tables. L'extension de ces résultats à des variables décrites par des distributions connues est possible, il est également possible de mixer des variables continues et des variables discrètes. Pour ce faire, on utilise généralement des distributions gaussiennes et la distribution d'un nœud conditionnellement à ses parents et une distribution gaussiennes dont la moyenne et une combinaison linéaire de la valeur des parents et dont la variance et indépendante de la valeur des parents et dont la variance et indépendante de la valeur des parents. [22]

$$p\left(X_i = \underset{k}{x} \mid \Pi_i = \underset{ij}{\pi}\right) = N(m_i, v_i)$$

$$m_i = \mu_i + \sum_{j=1}^{ni} \beta_{ij}(x_j - \mu_j)$$

$$\beta_{ji} = \frac{\sigma_{ij}}{\sigma_j^2} \qquad \text{et} \qquad v_i = \sigma_i^2$$

Où : Π_i sont les parents de X_i

μ_i et μ_j Sont la moyenne et la variance des attributs de X_i et X_j respectivement sans prendre en compte leurs parents

X_j : respectivement sans prendre en compte les parents

m_i et υ_i : sont la moyenne de la variance de l'attribut X_i en connaissant ses parents X_j

n_i : est le nombre de parents X_i

β_{ij} : appelé matrice de régression des poids ou matrice de poids

2.2 Apprentissage de paramètres à partir de données incomplètes :

On notera dans ce genre de situation que la démarche d'apprentissage des paramètres dépend de la nature plus au moins aléatoires des données manquantes. Elles peuvent être totalement aléatoires (ne dépendent pas de la base de données) pseudo aléatoires (dépendent des données observées) ou non aléatoires. Dans les deux premier cas, on a l'avantage de pouvoir estimer une distribution de données manquantes ; dans le dernier cas, il faut disposer d'autres informations.

Plusieurs approches ont été proposées pour cet apprentissage, nous ne citerons ici que la plus connue, basée sur l'algorithme EM permettant l'estimation de la log-vraisemblance des valeurs manquantes. L'algorithme EM est basé sur la répétition de deux étapes jusqu'à convergence du maximum de vraisemblance, après avoir initialisé les valeurs manquantes :

1. Etape E, « expectation » : estimation des valeurs manquantes en calculant leurs espérance selon les paramètres du modèle (ces valeurs sont initialisées aléatoirement lors de la première itération) ;
2. Etape M, « maximisation » : estimation des paramètres par maximum de vraisemblance (de la même façon qu'avec les données complètes). [23]

2.3 L'algorithme EM :

Algorithme EM structurel générique
- Initialiser $i \leftarrow 0$
- Initialisation du graphe $\mathcal{G}^0$
(*Graphe vide, aléatoire, donné par un expert ou arbre obtenu par MWST-EM*)
- Initialisation des paramètres Θ^0)
- Répéter
  - $i \leftarrow i+1$
  - $(\mathcal{B}^i, \Theta^i) = \underset{\mathcal{B},\Theta}{\operatorname{argmax}}\ Q(\mathcal{B}, \Theta : \mathcal{B}^{i-1}, \Theta^{i-1})$

Tant Que $| Q(\mathcal{B}^i, \Theta^i : \mathcal{B}^{i-1}, \Theta^{i-1}) - Q(\mathcal{B}^{i-1}, \Theta^{i-1} : \mathcal{B}^{i-1}, \Theta^{i-1}) | > \epsilon$

Notations :	
$Q(\mathcal{B}, \Theta : \mathcal{B}^*, \Theta^*)$	Espérance de la vraisemblance d'un réseau bayésien $< \mathcal{B}, \Theta >$ calculée à partir de la distribution de probabilité des données manquantes $P(\mathcal{D}_m \mid \mathcal{B}^*, \Theta^*)$

3. Apprentissage de la structure :

Dans la première partie de ce chapitre, nous avons examiné différentes méthodes d'apprentissage des paramètres d'un réseau bayésien, à l'aide d'un expert, en supposant que la structure de ce réseau était déjà connue. Se pose maintenant le problème d'apprentissage de cette structure : comment trouver la structure qui représentera le mieux de notre problème.

Deux types de score : les scores locaux et les scores globaux. La deuxième famille utilise les tests statistiques afin de déterminer les indépendances entre les variables dans le réseau. On peut citer l'algorithme « PC» pour Peter et Clark « IC » algorithme de principe similaire, pour inductive causation « BNPC» nommée « BN-PC-B » pour bayes net power construction et B.

Cette matrice de poids une fois définie, il suffit d'utiliser un des algorithmes de résolution du problème de l'arbre de poids maximal comme l'algorithme de Prim ou celui de kruskal.

Puis l'arbre non dirigé retourné par cet algorithme doit ensuite être dirigé en choisissant une racine et en parcourant l'arbre par une recherche en profondeur. La racine peut être choisie aléatoirement ou à l'aide de connaissance à priori. Comme dans l'apprentissage de paramètre, si nous ne possédons pas toutes les mesures qui sont issues des observations réelles, on utilise les algorithmes d'apprentissage de structure avec des données incomplètes, l'algorithme le plus usité est celui de EM (Expectation maximisation ou Esperance Maximisation). [25]

3.1. L'algorithme PC-IC:

Algorithme PC

- Construction d'un graphe non orienté
  - Soit $\mathcal{G}$ le graphe reliant complètement tous les nœuds $\mathcal{X}$
  - $i \leftarrow 0$
  - Répéter
    - Recherche des indépendances cond. d'ordre i
    - $\forall \{X_A, X_B\} \in \mathcal{X}^2$ tels que $X_A - X_B$ et $Card(Adj(\mathcal{G}, X_A, X_B)) \geq i$
    - $\forall \mathcal{S} \subset Adj(\mathcal{G}, X_A, X_B)$ tel que $Card(\mathcal{S}) = i$
      - si $X_A \perp X_B \mid \mathcal{S}$ alors
        - suppression de l'arête $X_A - X_B$ dans $\mathcal{G}$
        - $SepSet(X_A, X_B) \leftarrow SepSet(X_A, X_B) \cup \mathcal{S}$
        - $SepSet(X_B, X_A) \leftarrow SepSet(X_B, X_A) \cup \mathcal{S}$
    - $i \leftarrow i + 1$
  - Jusqu'à $Card(Adj(\mathcal{G}, X_A, X_B)) < i, \forall \{X_A, X_B\} \in \mathcal{X}^2$
- Recherche des V-structures
  - $\forall \{X_A, X_B, X_C\} \in \mathcal{X}^3$ tels que $\neg \overline{X_A X_B}$ et $X_A - X_C - X_B$,
    - si $X_C \notin SepSet(X_A, X_B)$ alors on crée une V-structure :
      $X_A \rightarrow X_C \leftarrow X_B$
- Ajout récursif de $\rightarrow$
  - Répéter
  - $\forall \{X_A, X_B\} \in \mathcal{X}^2$,
    - si $X_A - X_B$ et $X_A \rightsquigarrow X_B$, alors ajout d'une flèche à X_B :
      $X_A \rightarrow X_B$
    - si $\neg \overline{X_A X_B}$, $\forall X_C$ tel que $X_A \rightarrow X_C$ et $X_C - X_B$ alors $X_C \rightarrow X_B$
  - Tant qu'il est possible d'orienter des arêtes

Notations :

$\mathcal{X}$	ensemble de tous les nœuds
$Adj(\mathcal{G}, X_A)$	ensemble des nœuds adjacents à X_A dans $\mathcal{G}$
$Adj(\mathcal{G}, X_A, X_B)$	$Adj(\mathcal{G}, X_A) \setminus \{X_B\}$
$X_A - X_B$	il existe une arête entre X_A et X_B
$X_A \rightarrow X_B$	il existe un arc de X_A vers X_B
$\overline{X_A X_B}$	X_A et X_B adjacents $X_A - X_B$, $X_A \rightarrow X_B$ ou $X_B \rightarrow X_A$
$X_A \rightsquigarrow X_B$	il existe un chemin dirigé reliant X_A et X_B

Rapport-Gratuit.com

4. Outils d'implémentation :

Pour être efficace, les RB requièrent des implémentations fines, aussi le temps nécessaire à la programmation des inférences probabilistes dans un RB peut être un point pénalisant pour l'utilisation de tels outils. Cependant, un nombre assez important de logiciels existent aujourd'hui. Leurs différences sont nombreuses et il n'est pas toujours facile de trouver le programme adéquat à ses besoins.

Outre le langage de programmation utilisé (qui peut avoir son importance pour l'utilisateur), les plateformes et les fonctionnalités sont variées, autant du point de vue des calculs menés, que de ceux de l'interface utilisateur et du type de réseaux gérés. Il faut aussi distinguer les logiciels offerts dans le domaine public et les logiciels payants (cependant, la plupart des logiciels commerciaux ont des versions de démonstration gratuites utilisables dans de nombreux cas).

La plupart des logiciels utilise une méthode fondée sur celle de l'arbre de jonction. La liste de logiciels présentée ci-dessous ne se veut pas exhaustive. Tous les logiciels listés sont capables de gérer des réseaux probabilistes simples et de mener des inférences. NeticaJ, dont nous allons prendre mieux connaissance plus tard, est l'un des produits commerciaux de référence dans le domaine, tant par ses capacités que par ses nombreuses applications et extensions ; il possède une version de démonstration très élaborée. [13]

Logiciel	Système / machine	Remarques	Adresse
Graphical-Belief	Common Lisp		http://bayes.stat.washington.edu/almond/gb/graphical-belief.html
HUGIN	SunOS, Sun Solaris, Windows 95, NT (pour les dernières versions)	Produit commercial "phare". Version de démonstration	http://www.hugin.dk/
IDEAL	Common Lisp (interface GLIM)	Utilisation non-commerciale	http://www.rpal.rockwell.com/ideal.html
MacEvidence	MacIntosh		Prakash P. Shenoy School of Business University of Kansas Summerfield Hall Lawrence, KS 66045-2003 USA
MSBN32	Windows 95, Windows NT	Utilisation non-commerciale, licence requise	Microsoft Decision Theory Group http://www.research.microsoft.com/dtas/msbn/default.htm dtg-msbn@microsoft.com
Netica	Windows 95/NT, Macintosh, UNIX	Commercial mais version de démonstration gratuite	http://www.norsys.com/netica.html
Pulcinella	Common Lisp	Utilisation non-commerciale	Alessandro Saffiotti (asaffio@ulb.ac.be) IRIDIA, Université Libre de Bruxelles 50 av. F. Roosevelt, BP 194-6 1050 Brussels, Belgium
Strategist	Windows 95/NT	Commercial	http://www.prevision.com/strategist.html

Figure III.1 : Tableau comparatif entre les outils

5. Conclusion :

L'utilisation des réseaux bayésiens pour la modélisation de systèmes passe inévitablement par la détermination de la structure et des paramètres du réseau. De multiples méthodes existent, mais beaucoup de problèmes restent encore à résoudre concernant la prise en compte de données incomplètes, par exemple, ou l'apprentissage de modèles comme les réseaux bayésiens temporels. Il est de plus en plus fréquent d'être confronté à des applications où les données sont nombreuses mais incomplètes, et que les utilisateurs aimeraient exploiter, de manière à en extraire le plus d'informations possible. [26] Il ne suffit plus d'apprendre automatiquement un modèle qui réponde à une tâche précise, il faut arriver à un modèle permettant de découvrir des relations, des explications dont les utilisateurs pourront tirer profit.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Partie 02

ETUDE PRATIQUE

Chapitre I :

Analyse et conception

1. Introduction :

Nous passons à présent à l'analyse de l'étude de l'existant ainsi que l'identification des besoins suivant la démarche de conception adoptée pour cela nous utilisons UML.

UML ou Langage de Modélisation Unifié est un langage de modélisation graphique à base de pictogrammes conçu pour fournir une méthode normalisée pour visualiser la conception d'un système. Il est couramment utilisé en développement logiciel et en conception orientée objet.

Cette étape est très délicate dans n'importe quel projet informatique. Bien que cette phase paraisse facile à première vue, elle reste très complexe et nécessite beaucoup d'attention et de travail.

Elle nous permet aussi de bien comprendre les besoins des utilisateurs, et de déterminer les différentes fonctionnalités du système dont ils ont besoin. Par ailleurs, dans cette partie nous parlerons des principaux acteurs, nous illustrons leurs fonctions à travers les diagrammes de cas d'utilisations et de séquence.

2. Présentation application :

Dans cette partie nous présenterons brièvement et globalement le fonctionnement et but de notre application ainsi que ses tâches élémentaires que nous détaillerons dans le chapitre suivant.

Notre étude nous a permis d'établir la formalisation d'un système d'aide au diagnostic des pathologies de la glycémie à base de RB duquel nous pouvons extraire un certain nombre d'information. En premier lieu, nous avons accès à la structure du réseau, celui-ci nous permet de savoir quels sont les attributs qui sont dépendants ou non, de plus nous avons accès aux probabilités conditionnelles. Le calcul de ces informations constitue l'inférence assurant un diagnostic efficace et facile pour le modèle graphique.

Ce système est appliqué dans le diagnostic des maladies les plus fréquentes « le diabète » pour aider le médecin qui sera l'utilisateur dans notre application, possédant un compte propre à lui dans son processus de diagnostic. Pour ce faire il doit accéder à son compte, introduire les données concernant chaque patient, ensuite après analyse et calculs effectués par notre RB, le système lui fournira le diagnostic exact c'est-à-dire avec un pourcentage.

Les réseaux bayésiens présentent une méthodologie de représentation des connaissances permettant d'exprimer les relations entre les variables en utilisant la théorie des probabilités pour gérer l'incertitude liée au problème médicale afin d'apporter une vision compréhensible pour un non spécialiste.

3. Besoins techniques :

- **Traitement rapide :** Les données doivent être traitées rapidement afin de pouvoir satisfaire toutes les requêtes.

- **Temps de réponse réduit :** Les requêtes doivent être satisfaites le plus rapidement possible, pour ne pas faire attendre les utilisateurs
- **Ergonomie :** L'interface du système doit être claire et facile à utiliser sans fournir d'efforts.

4. Identification des cas d'utilisation :

L'objectif des cas d'utilisation est de définir une manière d'utiliser le système et permet d'en décrire les exigences fonctionnelles. Il représente aussi l'ensemble des séquences d'actions qui sont réalisées par le système et qui sont déclenchées par des acteurs.

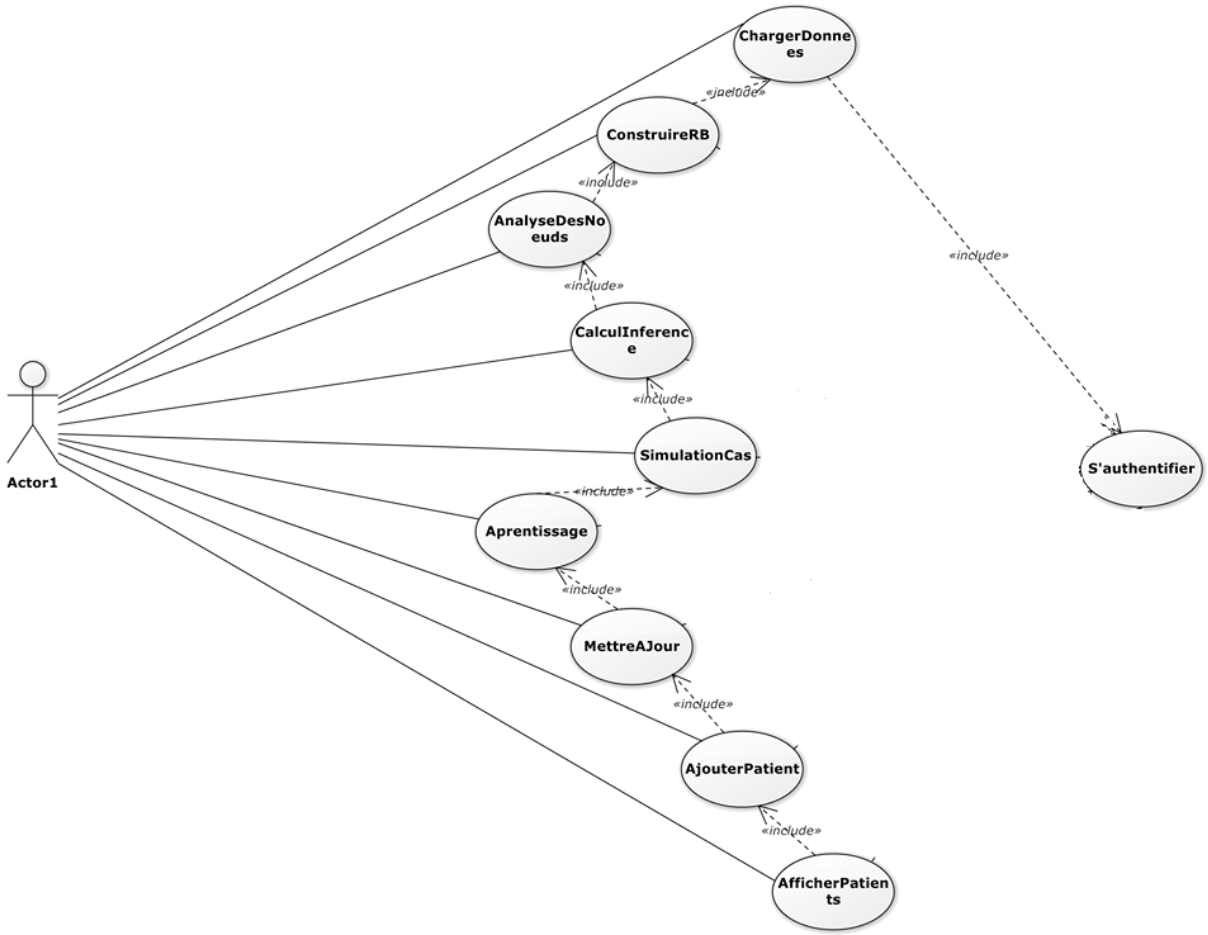


Figure I.1 : Diagramme de cas d'utilisation

5. Documentation des cas d'utilisation :

- **Cas d'utilisation « s'authentifier » :**

CU : S'authentifier
ID : 01

Description : Afin de pouvoir accéder au menu principal, il doit d'abord s'authentifier.
Acteur Primaire : utilisateur
Acteur Secondaire : /
Préconditions : - Utilisateur possédant un compte. - Connexion au serveur établie.
Enchainement principal : Ce cas commence lorsqu'un utilisateur muni d'un compte souhaite accéder à son compte. 1. L'utilisateur lance l'application 2. La page d'authentification apparait 3. Il introduit son nom d'utilisateur et son mot de passe. 4. Le serveur vérifie la validité du compte 5. La page de l'utilisateur s'affiche
Post-conditions : Utilisateur connecté
Enchainement alternatif : **E1 : La page d'authentification n'apparait pas**. 2- L'application affiche un message d'erreur. **E2 : L'identifiant ou le mot de passe de l'utilisateur ne sont pas valides.** 5- L'application affiche un message d'erreur.

- **Cas d'utilisation « AjouterUtilisateur» :**

CU : AjouterUtilisateur
ID : 4
Description : Afin qu'un nouvel utilisateur puisse bénéficier d'un accès à l'application, il faut que l'administrateur lui crée un compte.
Acteur Primaire : utilisateur
Acteur Secondaire : /
Préconditions : - Nouveau décideur sans compte.
Enchainement principal : Ce cas commence lorsque l'utilisateur souhaite créer un nouveau compte décideur 1. L'utilisateur lance l'application 2. Inclure (s'authentifier) 3. Il accède à l'option « s'enregistrer » 4. Une nouvelle page s'affiche contenant des cases à remplir 5. L'utilisateur remplie ses informations 6. L'application crée un nouveau compte utilisateur
Post-conditions : Nouveau compte décideur créé.
Enchainement alternatif : **E1 : L'application n'enregistre pas le nouveau compte** 7- L'application affiche un message d'erreur.

- **Cas d'utilisation « ChargerDonnees » :**

CU : ChargerDonnees
ID : 2
Description : Afin de construire le réseau bayésien, l'utilisateur doit charger les données à partir de la base de données contenant l'historique des patients.

Acteur Primaire : Utilisateur
Acteur Secondaire : /
Préconditions : - Utilisateur connecté
Enchainement principal : Ce cas commence lorsque l'utilisateur souhaite charger les données dans l'application 1. L'utilisateur lance l'application 2. Inclure (s'authentifier) 3. Il lance l'importation des données 4. Une fois les données chargées, l'application donne la possibilité à l'utilisateur pour effectuer les étapes suivantes.
Post-conditions : Données chargées dans l'application
Enchainement alternatif: /

- **Cas d'utilisation « ConstruireRB »**

CU : ConstruireRB
ID : 3
Description : Afin que l'utilisateur puisse analyser la structure du réseau bayésien et voir le graphe probabiliste construit à partir de données.
Acteur Primaire : Utilisateur
Acteur Secondaire : /
Préconditions : - Données chargées dans l'application.
Enchainement principal : Ce cas commence lorsque l'utilisateur souhaite construire le réseau bayésien 7. L'administrateur lance l'application 8. Inclure (s'authentifier) 9. Inclure (ChargerDonnees) 10. Il accède à l'option « Réseau Bayésien » 11. Une nouvelle page s'affiche contenant le graphe probabiliste du réseau bayésien.
Post-conditions : Réseau bayésien créé.
Enchainement alternatif : /

- **Cas d'utilisation « AjouterPatient »**

CU : AjouterPatient
ID : 9
Description : afin de pouvoir identifier un potentiel risque de diabète, l'utilisateur ajoute un nouveau patient et détermine quel est son pourcentage de diabète grâce au réseau

bayésien
Acteur Primaire : Utilisateur
Acteur Secondaire : /
Préconditions : - Mise à jour effectuée
Enchainement principal : Ce cas commence lorsqu'un utilisateur souhaite examiner un nouveau patient 1. L'utilisateur lance l'application 2. Inclure (s'authentifier) 3. Inclure (MettreAJour) 4. L'utilisateur sélectionne la partie « Ajouter un nouveau patient » 5. Un nouveau formulaire s'affiche pour introduire les informations du nouveau patient 6. L'utilisateur introduit les informations relatives au patient (nom, prénom, photo, taille, poids, âge, conditions sportives, antécédents familiaux, sexe, nourriture, glycémie, fumeur, obésité) 7. L'utilisateur confirme l'ajout du patient 8. L'application calcule le taux du diabète du nouveau patient en se basant sur le réseau bayésien 9. Une nouvelle page s'affiche contant la fiche du client ainsi que le pourcentage du diabète.
Post-conditions : le nouveau patient est enregistré dans la base de données.
Enchainement alternatif : /

Cas d'utilisation « AfficherPatients »

CU : AfficherPatients
ID : 10
Description : Cette fonctionnalité permet de lister toute la liste des patients
Acteur Primaire : Utilisateur
Acteur Secondaire : /
Préconditions : /
Enchainement principal : Ce cas commence lorsqu'un utilisateur souhaite consulter la liste des patients. 1. L'utilisateur lance l'application 2. Inclure (s'authentifier) 3. L'utilisateur sélectionne le volet « affichage des patients » 4. Une nouvelle page s'affiche contant la liste des patients.
Post-conditions : /
Enchainement alternatif : /

6. Diagrammes de séquences :

Les diagrammes de classes sont la représentation graphique des interactions entre les acteurs et les différentes classes du système selon un certain ordre chronologique.

La représentation des différentes classes se fait de la manière suivante :

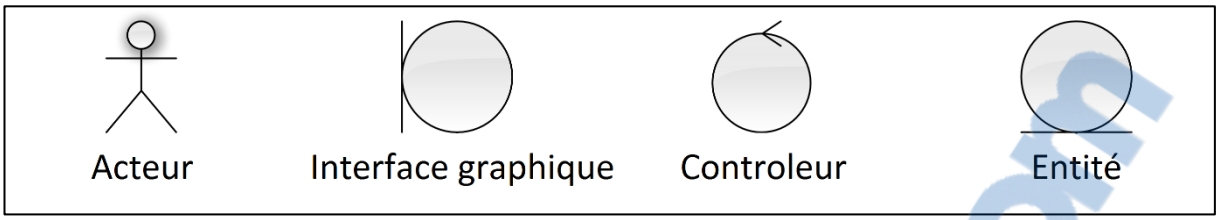


Figure I.2*:* Représentation des classes dans le diagramme de séquence

- **Diagramme de séquence – partie authentification :**

Cette partie concerne le processus d'authentification des utilisateurs du système Le diagramme d'authentification présente aussi les manipulations que peut effectuer l'administrateur sur la liste des décideurs (ajout et suppression).

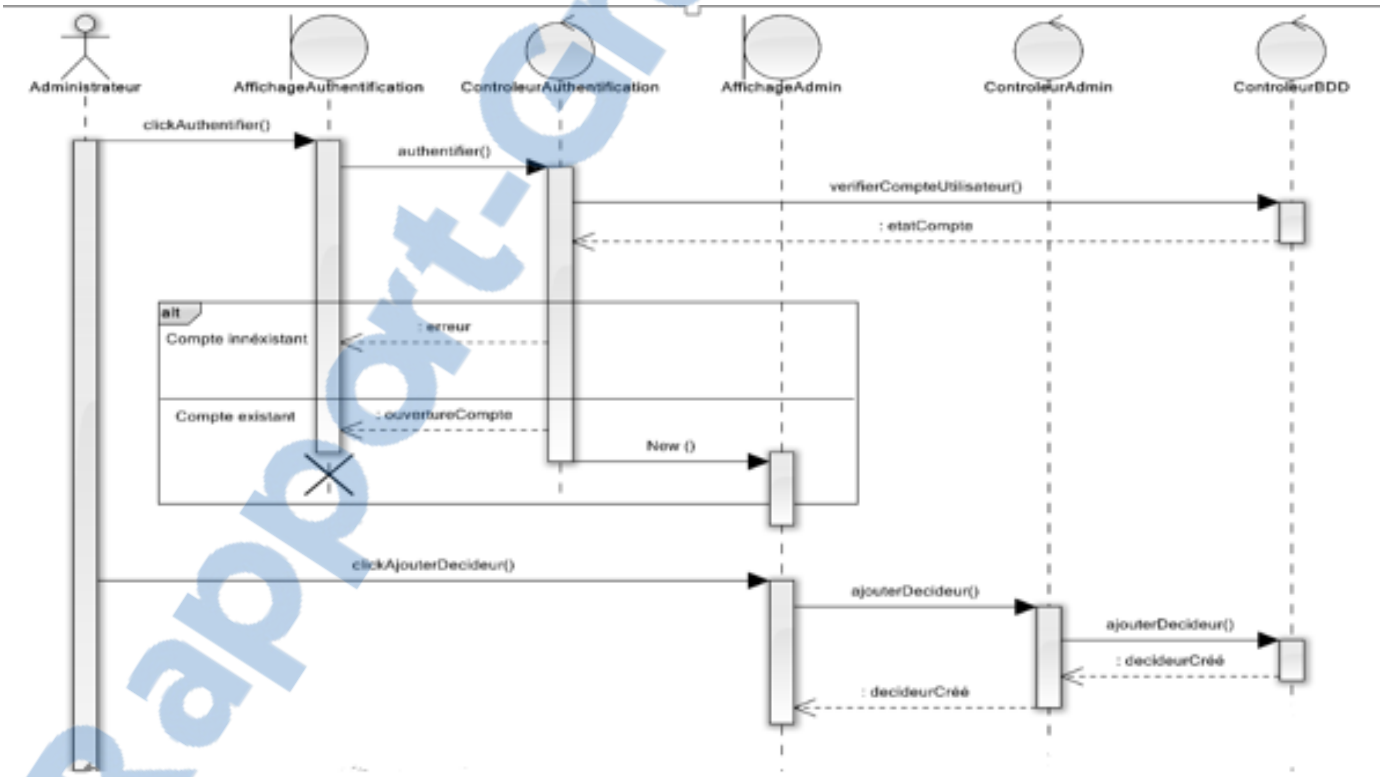


Figure I.3 : Diagramme de séquence (processus d'authentification)

Rapport-Gratuit.com

- **Diagramme de séquence :**

Cette partie concerne le processus de l'application, elle décrit les différentes opérations réalisées lorsqu'un utilisateur effectue un ajout de patient.

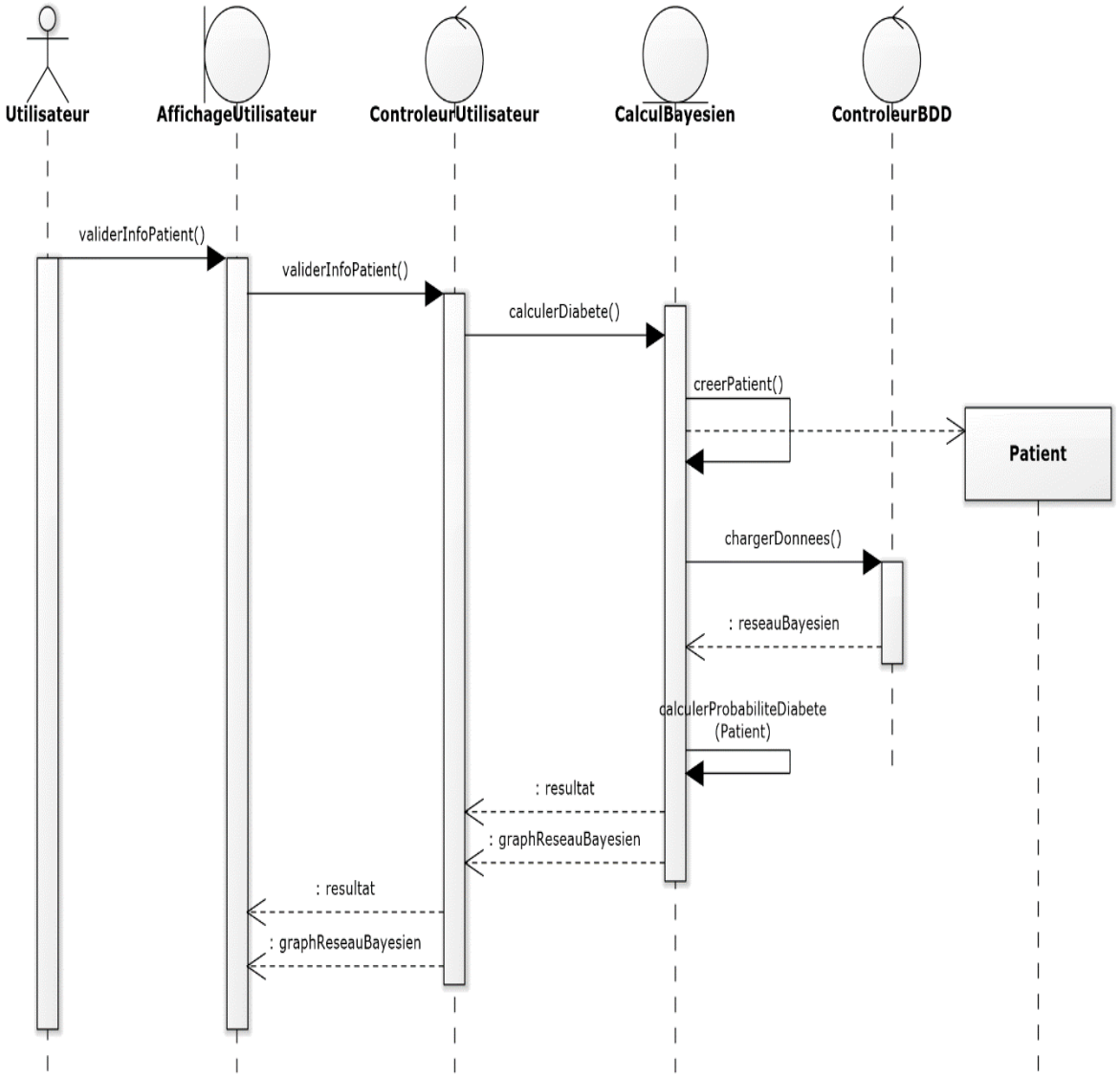


Figure I.4 : Diagramme de séquence (processeur d'application)

7. Documentation des diagrammes de séquence :

La séquence débute lorsque l'utilisateur accède à l'interface graphique "AffichageUtilisateur" et remplie le formulaire du patient, il clique sur le bouton de validation pour valider les informations saisies.

L'interface graphique "affichageUtilisateur" renvoie les informations validées après un contrôle au contrôleur "ControleurUtilisateur" et ce dernier renvoie les informations à l'entité qui permet de calculer la probabilité d'atteinte du diabète.

L'entité "CalculBayésien" crée un nouveau patient dans la base de données avec les informations récupérées du contrôleur précédent.

Il charge par la suite le réseau bayésien, puis calcule la probabilité du diabète selon les informations du patient.

Les résultats sont ensuite retournés jusqu'à ce qu'ils arrivent à l'interface graphique et s'affichent finalement devant les utilisateurs.

8. Architecture globale :

Dans cette partie nous allons présenter l'architecture globale de notre système mais avant parler de notre architecture, présentons d'abord les différentes architectures n_tiers.

- **Architecture simple tiers :**

Cette architecture présente l'interface utilisateur, la couche métier, et de la base de données en un seul logiciel, toutes les fonctionnalités sont donc comprises dans une seule couche logicielle. Les applications logicielles telles que MS Access, MS Excel ont tous le même point commun : l'application gère les trois niveaux (logique métier, l'assurance-chômage et l'accès aux données). Cette architecture est inadaptée à notre système puisque les données que l'utilisateur exploite sont stockées sur une machine à part.

- **Architecture 2-tiers :**

Cette architecture est appelée aussi architecture client/serveur : le client exécute l'application et le serveur qui gère la base de données autrement dit, le client gère l'interface utilisateur et le processus métier et le serveur gère l'accès aux données. Ce type d'architecture permet à plusieurs utilisateurs d'accéder à la base de données en même temps tant. Ceci nous donne un énorme avantage et cette architecture semble conforme à notre système.

Cependant, le besoin de l'exécution en temps réel des demandes des clients ainsi que le besoin de sécuriser la base de données nous oblige à séparer le serveur web et le serveur de base de données ce qui va nous amener à utiliser une

- **Architecture 3-tiers :**

En nous offrant à la fois une exécution instantanée et une sécurité exceptionnelle.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

9. Configuration matérielle du système :

Comme nous avons mentionné précédemment, l'architecture 3-tiers représente l'architecture la plus appropriée à notre système en nous assurant la disponibilité de données en temps réel avec des meilleures performances grâce à la répartition des charges de travail.

La figure suivante illustre la configuration matérielle de notre système :

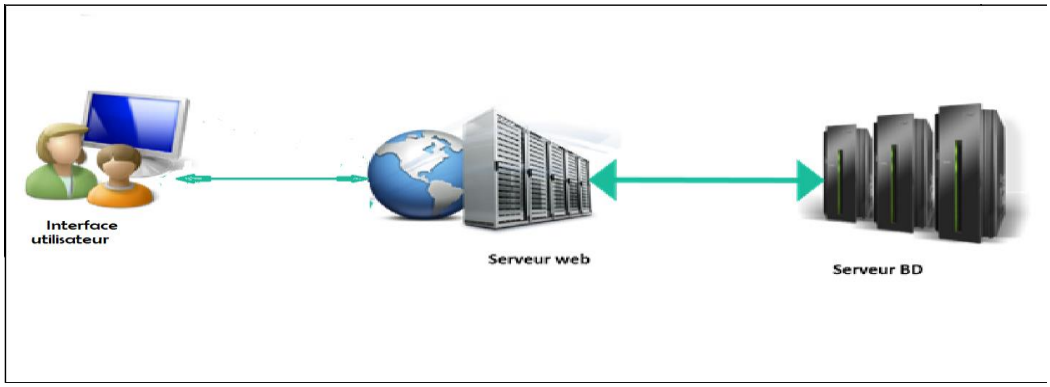


Figure I.5 : la configuration matérielle de notre système

Notre système est équipé de :

- **Serveur de gestion de base de données :** En raison du changement continuel des données, l'accès au serveur BD doit être disponible à tout moment, il doit aussi avoir une grande capacité de stockage.
- **Serveur web :** Est un serveur qui répond aux demandes des utilisateurs.
- **Utilisateurs :** Sont toutes sortes de machine qui peuvent avoir accès à l'application.

10. Architecture Globale de l'application :

L'architecture de notre application est formée comme suite :

1- La fenêtre d'édition du graphe

- Création de nœuds
- Création de relations entre nœuds
- Calcul de probabilité et de décision
- Chargement de données

2- Le menu : Il faut dans un premier temps :

- Enregistrement
- Connexion

Pour avoir accès à :

- Données

- Réseau bayésien
- Analyse
- Inférence
- Simulation des cas
- Apprentissage
- Mise à jour
- Ajouter un patient
- Liste des patients

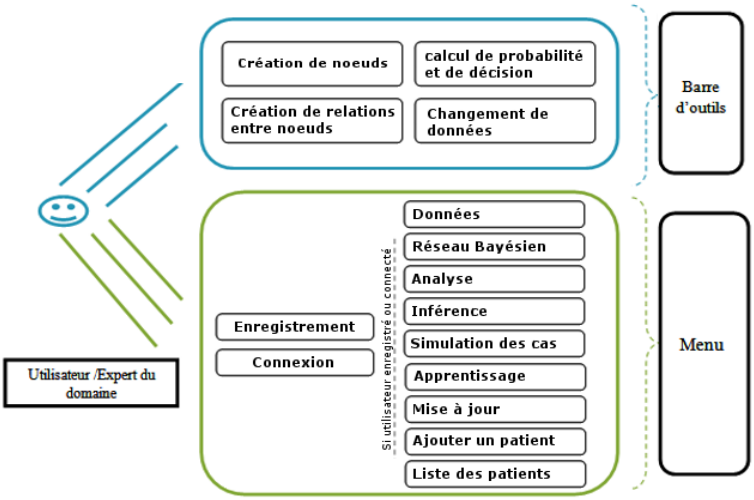


Figure I.6 : l'architecture générale de l'application

11. Conclusion :

Cette phase d'expression des besoins est une étape très importante pour la bonne conception du projet. Lors de ce chapitre, nous avons identifié les différents acteurs de notre système et nous les avons interrogés de différentes méthodes afin d'obtenir les informations nécessaires pour comprendre leurs besoins.

Nous avons su identifier à travers ce chapitre les fonctionnalités de notre système, ceci va constituer la base de la conception qui sera détaillée dans le prochain chapitre.

CHAPITRE II

Mise en œuvre et réalisation

1. Introduction :

Ce chapitre est basé sur la réalisation et la mise en œuvre de notre application. Nous allons utiliser le JAVA comme langage de programmation. Une base de données MySQL, le système de gestion de base de données est phpMYadmin et le langage de manipulation de notre base de données est SQL. Enfin nous montrons les principales interfaces et fenêtres de notre application.

2. Concepts de développement :

2.1. Langage de programmation :

Java est un langage de programmation et une plate-forme informatique qui ont été créés par Sun Microsystems en 1995. Beaucoup d'applications et de sites Web ne fonctionnent pas si Java n'est pas installé et leur nombre ne cesse de croître chaque jour. Java est rapide, sécurisé et fiable. Des ordinateurs portables aux centres de données, des consoles de jeux aux superordinateurs scientifiques, des téléphones portables à Internet, la technologie Java est présente sur tous les fronts ! [1]

La particularité et l'objectif central de Java est que les logiciels écrits dans ce langage doivent être très facilement portables sur plusieurs systèmes d'exploitation tels que UNIX, Windows, Mac OS ou GNU/Linux, avec peu ou pas de modifications. Pour cela, divers plateformes et frameworks associés visent à guider, sinon garantir, cette portabilité des applications développées en Java. [1]

2.2. Base de Données :

Une base de données est composée de données stockées dans des mémoires de masse sous une forme structurée, et accessibles par des applications différentes et des utilisateurs différents. Une base de données doit pouvoir être utilisée par plusieurs utilisateurs en même temps. [2]

2.2.i. Bases de Données MySQL :

Développée par le monde libre sous licence GNU (racheté par Sun en 2007 qui garde la méthode de distribution), les bases de données MySQL sont principalement utilisées sur les hébergements Internet de type Linux, en coopération avec le langage PHP mais utilisables sur de nombreux systèmes d'exploitation. [3]

MySQL est une base de données **Client - serveur**. Le serveur SQL est multithread (il accepte plusieurs requêtes simultanées) et intègre différents clients, librairies.

Chaque base de données peut inclure plusieurs tables. Chaque table inclue les informations brutes, comme entrées par l'utilisateur. Via différentes commandes, MySQL est relationnel, il permet de créer des relations entre différentes tables via des requêtes dans son propre langage de programmation. La base de données en entière est stockée sous forme de dossier. Les tables sont enregistrées sous forme de fichiers. [3]

2.2.ii. Système de Gestion de Bases de Données :

Un système de gestion de base de données '**SGBD**' est un logiciel de haut niveau qui permet de manipuler ses informations. Il peut être perçu comme un ensemble de logiciels systèmes permettant aux utilisateurs d'insérer, de modifier et de rechercher efficacement des données spécifiques dans une masse d'informations partagées par de multiples utilisateurs .Les recherches peuvent être exécutées à partir de la valeur d'une donnée désignée par un nom dans un ensemble d'objet , mais aussi à partir de relations entre objets . [4]

En résumé, un SGBD peut donc apparaître comme un outil informatique permettant la sauvegarde, l'interrogation, la recherche et la mise en forme de données stockées en mémoire pour :

- Permettre l'accès aux données de façon simple.
- Autoriser un accès aux informations à de multiples utilisateurs.

- Manipuler les données présentes dans la base de données (insertion, suppression, modification).
- Le contrôle d'intégrité des données accédées.
- Le maintien de la cohérence des données entre elles.

2.2.iii Système de Gestion de Bases de Données phpMYadmin :

phpMyAdmin est une application de gestion pour les systèmes de gestion de base de données MySQL.

Il s'agit de l'une des plus célèbres interfaces pour gérer une base de données MySQL sur un serveur.

Cette interface pratique permet d'exécuter, très facilement et sans grandes connaissances en bases de données, des requêtes comme les créations de table de données, insertions, mises à jour, suppressions et modifications de structure de la base de données, ainsi que l'attribution et la révocation de droits et l'import/export. Ce système permet de sauvegarder commodément une base de données sous forme de fichier .sql et d'y transférer ses données, même sans connaître SQL.

Les requêtes SQL restent possibles, ce qui permet de les tester interactivement lors de la création d'un site pour les utiliser ensuite en batch (c'est-à-dire en différé) une fois au point. [5]

2.2.iv Langage de manipulation de bases de données :

SQL '**Structured Query Language**' est un langage de manipulation de bases de données mis au point dans les années 70, et il permet trois types de manipulations :

- La maintenance des tables : création, suppression, modification de la structure des tables.
- La manipulation des données : sélection, modification, suppression d'enregistrements.
- La gestion des droits d'accès aux tables : contrôle des données; droits d'accès, validation des modifications. [4]

2.3. L'Environnement de Développement :

NetBeans est un environnement de développement intégré (EDI), placé en open source par Sun en juin 2000 sous licence CDDL (Common Development and Distribution License) et GPLv2. En plus de Java, NetBeans permet la prise en charge native de divers langages tels le C, le C++, le JavaScript, le XML, le Groovy, le PHP et le HTML, ou d'autres (dont Python et Ruby) par l'ajout de greffons. Il offre toutes les facilités d'un IDE moderne (éditeur en couleurs, projets multi-langage, refactoring, éditeur graphique d'interfaces et de pages Web). Compilé en Java, NetBeans est disponible sous Windows, Linux, Solaris, Mac OS X ou sous une version indépendante des systèmes d'exploitation. Un environnement Java Development Kit JDK est requis pour les développements en Java.

NetBeans constitue par ailleurs une plate-forme qui permet le développement d'applications spécifiques (bibliothèque Swing (Java)). L'IDE NetBeans s'appuie sur cette plate forme. L'IDE Netbeans s'enrichit à l'aide de greffons. [6]

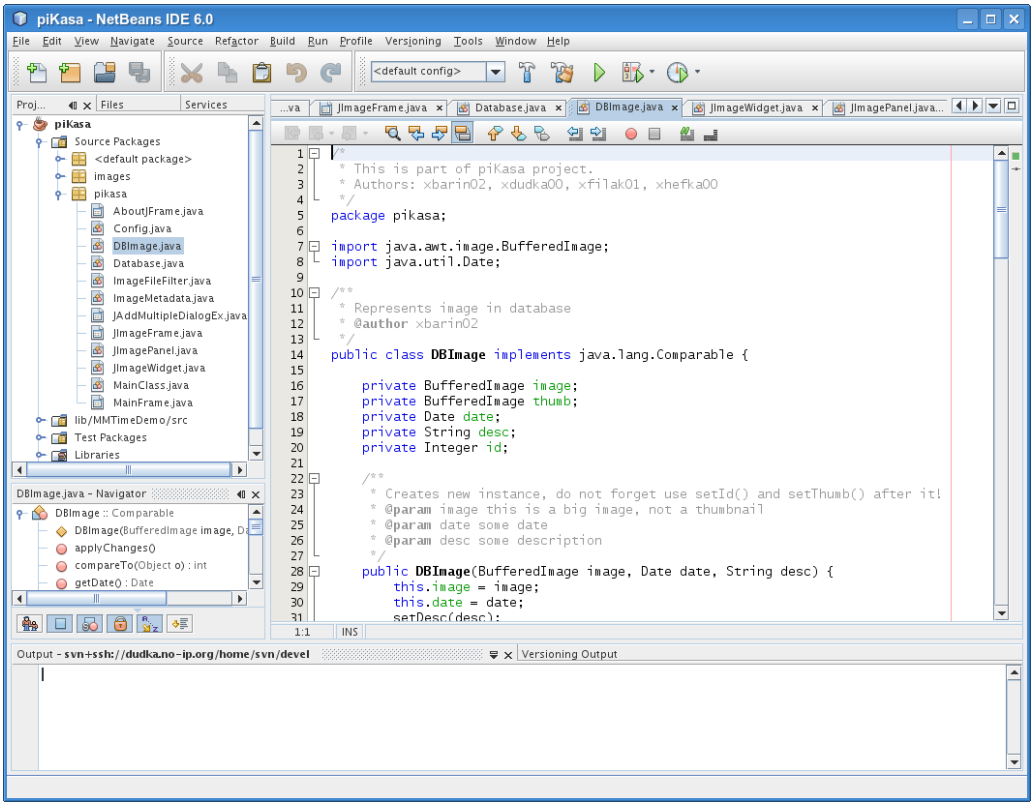


Figure II.1. Interface de Netbeans

2.4. Outils utilisés :

2.4.i. XAMPP :

XAMPP est un ensemble de logiciels permettant de mettre en place facilement un serveur local.

Il s'agit d'une distribution de logiciels libres (X (cross) Apache MariaDB Perl PHP) offrant une bonne souplesse d'utilisation, réputée pour son installation simple et rapide. Ainsi, il est à la portée d'un grand nombre de personnes puisqu'il ne requiert pas de connaissances particulières et fonctionne, de plus, sur les systèmes d'exploitation les plus répandus. [7]

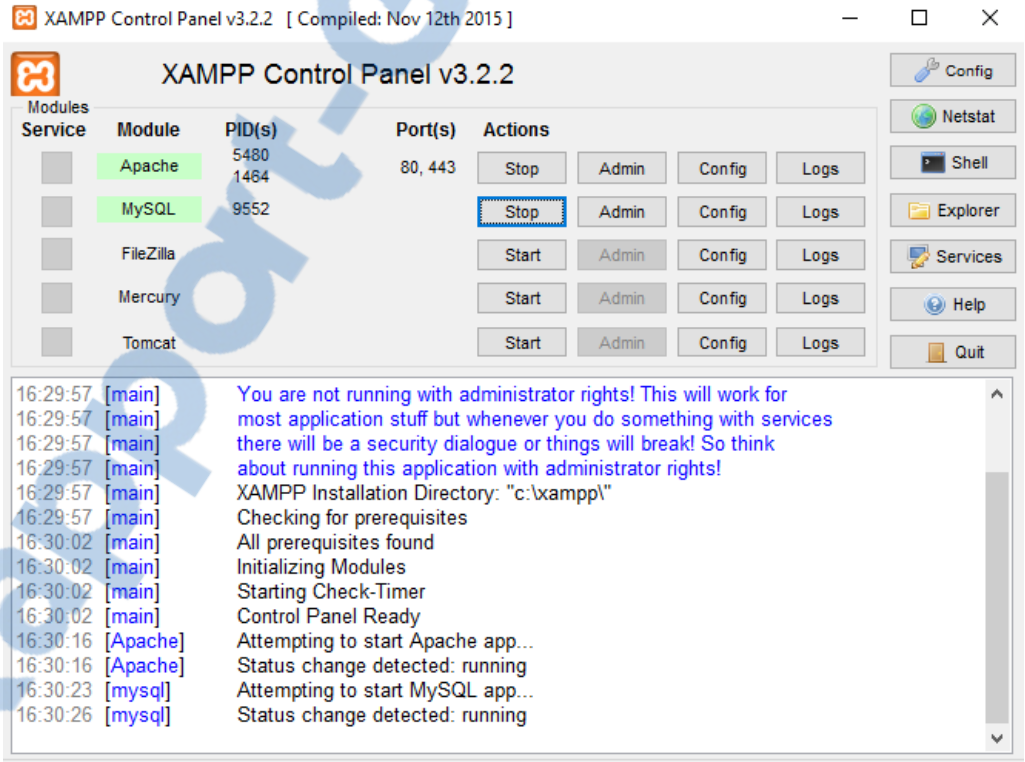


Figure II.2. Interface de XAMPP

Rapport-Gratuit.com

2.4.ii. HeidiSQL :

HeidiSQL est un outil d'administration de base de données possédant un éditeur SQL et un constructeur de requête. Il a été développé et optimisé pour être utilisé avec le SGBD relationnel MySQL. [8]

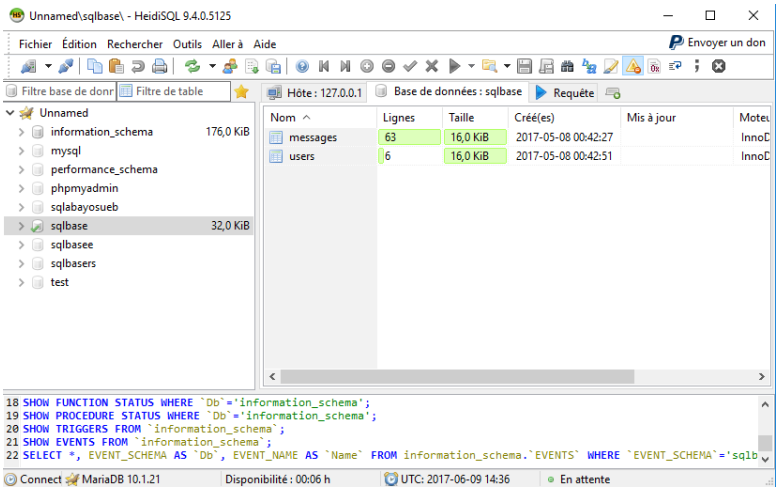


Figure II.3. Interface de HeidiSQL

2.4.iii Netica-Java API :

L'API Netica-J est une bibliothèque complète de classes Java pour travailler avec des réseaux bayésiens (également connus sous le nom de réseaux Bayes, réseaux de croyance, modèles graphiques ou modèles causaux probabilistes) et des diagrammes d'influence (également appelés réseaux de décision). Il contient des fonctions à construire, à apprendre à partir de données, à modifier, à transformer, à tester le rendement, à sauvegarder et à lire des filets, ainsi que d'un puissant moteur d'inférence. Il peut gérer des "cas" et des ensembles de cas, et peut se connecter directement à la plupart des logiciels de base de données. Les réseaux Bayes peuvent être utilisés pour le diagnostic, la prédiction, la classification, la fusion des capteurs, l'analyse des risques, l'analyse des décisions, combinant des informations incertaines et de nombreuses tâches d'inférence probabilistes. [10]

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

2.4.iv. MySQL Connector-Java :

MySQL connector est une bibliothèque Java qui permet l'insertion et la récupération de données dans une base de données MySQL.

3. Description de l'application

Figure II.4. Logo Application

- **L'entrée de l'application**

C'est la première fenêtre qui s'affiche au lancement de notre application, toute personne qui veut bénéficier des services du logiciel doit s'authentifier avec un login et mot de passe. Cette page comporte deux boutons principaux dont le premier est « Connexion » qui permet l'accès à la fenêtre de connexion si l'utilisateur a un compte. Le deuxième bouton est « S'enregistrer » pour créer un nouveau compte. Il existe aussi deux boutons, le premier est «À propos » qui permet l'accès à la fenêtre « A propos de nous » et le deuxième bouton est « Quitter » pour quitter l'application.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Figure II.5. Interface de l'application

La figure suivante présente l'interface de la création d'un nouveau compte, l'utilisateur doit introduire son nom d'utilisation pour l'application et le mot de passe ainsi qu'une photo.

Figure II.6. Interface de création de nouveau compte

Si le nom d'utilisateur introduit existe déjà, alors l'application revoit le message d'erreur suivant :

Figure II.7. Message d'erreur – Nom d'utilisateur existe déjà –

Si les deux mots de passe ne sont pas identiques, l'application vous demandera d'entrer de nouveau les mots de passe une autre fois avec le message suivant :

Figure II.8. Message d'erreur – Les 2 mots de passe ne sont pas identiques –

Une fois l'utilisateur s'enregistre dans le système, il peut se connecter pour accéder au menu principal, un formulaire s'affiche à l'écran, il lui demandera d'introduire le nom d'utilisateur et le mot de passe d'authentification.

Figure II.9. Interface d'authentification

Si le nom d'utilisateur ou le mot de passe introduit n'est pas valide, alors l'application renvoit message d'erreur suivant :

Figure II.10. Message d'erreur – Les informations de connexion sont incorrectes –

Après connexion, la fenêtre ci-après s'affiche, elle comporte le menu principal où l'utilisateur pourra sélectionner la tâche à effectuer.

Le menu de notre application contient neuf (09) boutons qui sont : Données, Réseau Bayésien, Analyse, Inférence, Simulation des cas, Apprentissage, Mise à jour, Ajout d'un patient, prise de la décision et la liste des patients.

Figure II.11. Interface principale

✓ Données

La première étape de notre travail consiste à choisir la topologie du réseau. Dans cette topologie on a localisation 08 nœuds : Poids, Taille, Sport, Fumeur, Nourriture, Obésité, Glycémie, Antécédent.

Dans cette étape le système récupère des données ou des probabilités initiales « d'après l'algorithme bayésien » du réseau qui représente les cas réels, par exemple pour le cas poids normal 0.6 et abnormal 0.4 donc 60% de personnes ont un poids normal… comme le présente la figure suivante :

Figure II.12. Interface de données initiales

✓ Réseau Bayésien

La deuxième étape consiste à construire le graphe du réseau bayésien. La construction de ce graphe nécessite l'utilisation des 08 nœuds déjà définies dans l'étape précédente. Nous procédons dans cette partie à la construction de réseaux bayésiens à partir de bases de données initiales.

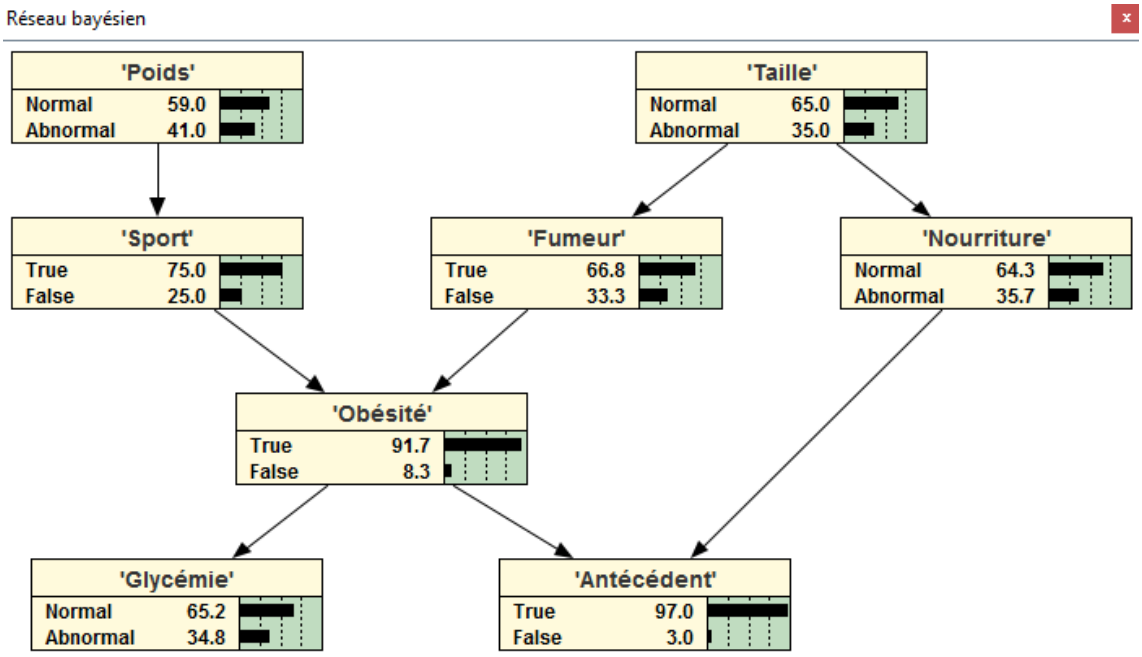


Figure II.13. Interface de graphe du réseau bayésien

La figure suivante illustre la déclaration du nœud Poids du réseau bayésien

```
node poids {
kind = NATURE;
discrete = TRUE;
states = (Normal, Abnormal);
parents = ();
probs =
(0.74,0.26);
title = 'Poids';
whenchanged = 904512863;
belief = (0.74,0.26);
visual V1 {
center = (126, 54);
height = 7;
};
};
```

Figure II.14. Exemple de déclaration d'un nœud du réseau bayésien

✓ Analyse

Cette étape consiste à analyser le réseau bayésien et la relation qui existe entre les différents nœuds avec leurs probabilité pour que l'application se familiarise avec le réseau bayésien.

✓ Inférence

Ensuite l'étape d'inférence pour déduire les relations entre les classes (poids, taille..) et leurs probabilités.

Dans cette étape nous indiquons à Netica-Java de lire le fichier ChestClinic.dne, ce fichier contient les informations de notre réseaux bayésien (les noeuds, les relations, et les probabilités).

Les probabilités sont récupérées à partir d'une base de données, donc ce fichier sera mis à jour lorsqu'il y a un changement de ces valeurs.

```
node poids {
kind = NATURE;
discrete = TRUE;
states = (Normal, Abnormal);
parents = ();
probs =
(0.59,0.41);
title = 'Poids';
whenchanged = 904512863;
belief = (0.59,0.41);
visual V1 {
center = (126, 54);
height = 7;
};
};

node faire_sport {
kind = NATURE;
discrete = TRUE;
states = (True, False);
parents = (poids);
probs =
((0.59,0.41),
 (0.98,0.02));
title = 'Faire_sport';
whenchanged = 904522713;
belief = (0.02,0.98);
visual V1 {
center = (126, 156);
```

Figure II.15. Exemple de déclaration des éléments dans un fichier.dne

Ensuite nous indiquons a Netica-Java les nœuds et nous demandons de faire une compilation, cette phase permet de faire les inférences.

Après cette étape Netica-Java va comprendre notre réseau : les nœuds, leurs probabilités, les relations entre eux…, en résumé il va comprendre tout ce qui concerne notre réseau.

✓ Simulation des cas

Netica-Java utilise un fichier avec extension .cas pour indiquer les cas sur lesquelles nous allons faire notre apprentissage.

Pour Netica-Java il utilise le principe suivant :

- On lui indique un exemple (ou un ensemble d'exemples) cet exemple doit avoir une valeur pour chaque nœud de notre réseau bayésien. Pour cela nous sélectionnons ces valeurs avec l'interface suivante :

Figure II.16. Interface de saisie d'informations de cas de simulation

Après confirmation, ces informations sont enregistrées dans le fichier infocas.cas, comme montre la figure suivante :

```
IDnum,poids,faire_sport,taille,fumeur,plasmasanguin,resultatanalyse,nuriture,maladiegenetique
0,normal,false,abnormal,true,false,normal,normal,false
```

Figure II.17. Le contenu de fichier infocas.cas

Le fichier infocas.cas est composé de deux lignes, la ligne 1 contient les noms des noeuds, la ligne 2 contient la valeur de chaque noeud.

Rapport-Gratuit.com

- Ensuite nous demandons à Netica-Java de générer 500 cas aléatoires, les cas sont enregistrés dans le fichier ChestClinic.cas, comme montre la figure suivante :

```
462,Abnormal,True,Abnormal,True,True,Normal,Abnormal,True
463,Abnormal,True,Normal,True,True,Normal,Abnormal,True
464,Normal,False,Normal,True,True,Normal,Normal,True
465,Normal,True,Normal,True,True,Normal,Normal,True
466,Normal,True,Normal,True,True,Abnormal,Abnormal,True
467,Abnormal,True,Normal,True,True,Normal,Normal,True
468,Normal,True,Abnormal,True,True,Normal,Abnormal,True
469,Normal,False,Normal,False,False,Abnormal,Normal,True
470,Normal,True,Normal,True,True,Normal,Normal,True
471,Normal,False,Abnormal,True,True,Normal,Abnormal,True
472,Normal,True,Abnormal,True,True,Abnormal,Normal,True
473,Normal,True,Abnormal,True,True,Normal,Abnormal,True
474,Abnormal,True,Normal,True,True,Normal,Abnormal,True
475,Abnormal,True,Abnormal,True,True,Normal,Abnormal,True
476,Abnormal,True,Normal,True,True,Normal,Normal,True
477,Normal,True,Abnormal,True,True,Abnormal,Abnormal,True
478,Normal,True,Abnormal,False,True,Abnormal,Abnormal,True
479,Abnormal,True,Normal,True,True,Normal,Abnormal,True
480,Normal,True,Normal,True,True,Normal,Normal,True
481,Normal,False,Abnormal,True,True,Normal,Normal,True
482,Normal,False,Abnormal,True,True,Normal,Abnormal,True
483,Normal,False,Abnormal,True,True,Abnormal,Normal,True
484,Abnormal,True,Normal,True,True,Abnormal,Normal,True
485,Abnormal,True,Normal,True,True,Abnormal,Normal,True
486,Normal,True,Normal,True,True,Abnormal,Normal,True
487,Abnormal,True,Normal,True,True,Abnormal,Normal,True
488,Abnormal,True,Abnormal,True,True,Normal,Normal,True
489,Abnormal,True,Abnormal,True,True,Normal,Abnormal,True
490,Normal,False,Normal,False,False,Abnormal,Abnormal,False
491,Normal,False,Normal,True,True,Normal,Abnormal,True
492,Normal,True,Abnormal,True,True,Normal,Abnormal,True
493,Abnormal,True,Normal,True,True,Normal,Normal,True
494,Abnormal,True,Abnormal,True,True,Normal,Normal,True
```

Figure II.18. Le contenu de fichier ChestClinic.cas

✓ Apprentissage

Dans l'étape apprentissage l'application applique toutes les notions qu'elle a déduit sur ces cas pour les classifier et prendre les décisions.

Donc pour Netica-Java il va procéder comme suivant :

- Comparer l'exemple dans le fichier infocas.cas avec les exemples dans le fichier ChestClinic.cas.

- Ensuite générer le fichier Learned_ChestClinic.dne ce fichier est le même que ChestClinic.dne, sauf que les probabilités ont changé par apport à l'apprentissage qui est appliqué sur les 500 cas aléatoire.

- Comme conclusion Netica-Java aide à bien comprendre notre réseau bayésien. La figure suivant montre l'étape d'apprentissage :

```
try {
    //Environ env = new Environ (null);
    Net       net       = new Net (new Streamer ("Data/ChestClinic.dne"));
    NodeList nodes      = net.getNodes();
    int       numNodes  = nodes.size();

    for (int n = 0;  n < numNodes;  n++) {
        Node node = (Node) nodes.get (n);
        node.deleteTables();
    }

    Streamer caseFile = new Streamer ("Data/ChestClinic.cas");
    net.reviseCPTsByCaseFile (caseFile, nodes, 1.0);

    net.write (new Streamer ("Data/Learned_ChestClinic.dne"));
```

Figure II.19. Le code source qui permet de faire l'apprentissage

✓ Mise à jour

L'étape M.A.J permet d'effectuer les M.A.J par rapport aux résultats obtenus sur les cas simulés.

Désormais Netica-Java a bien compris notre réseau bayésien, après simulation des cas et apprentissage, cet outil permet de proposer des mises à jour sur les probabilités, il permet aussi de bien comprendre les relations et les dépendances entre les nœuds de notre réseau bayésien.

Dans cette étape nous disons à Netica-Java de faire les MAJ nécessaire.

✓ Ajout d'un patient

A présent Netica-Java permet de prendre les décisions, sur des cas réels, pour cela nous mettons une interface graphique pour la saisie les informations d'un patient afin de générer la décision finale.

La figure suivant montre l'interface graphique pour saisir les informations d'un patient comme nom, prénom, sexe, poids, taille …

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

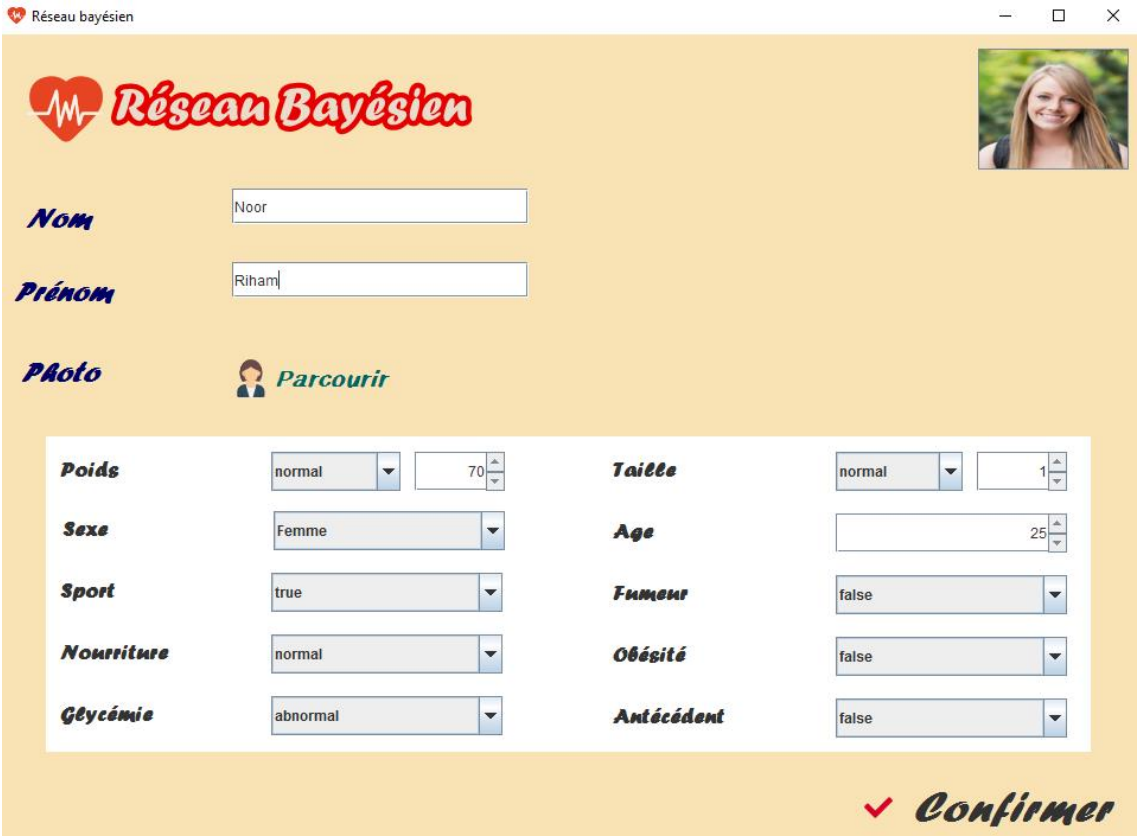


Figure II.20. L'interface graphique pour saisir les informations d'un patient

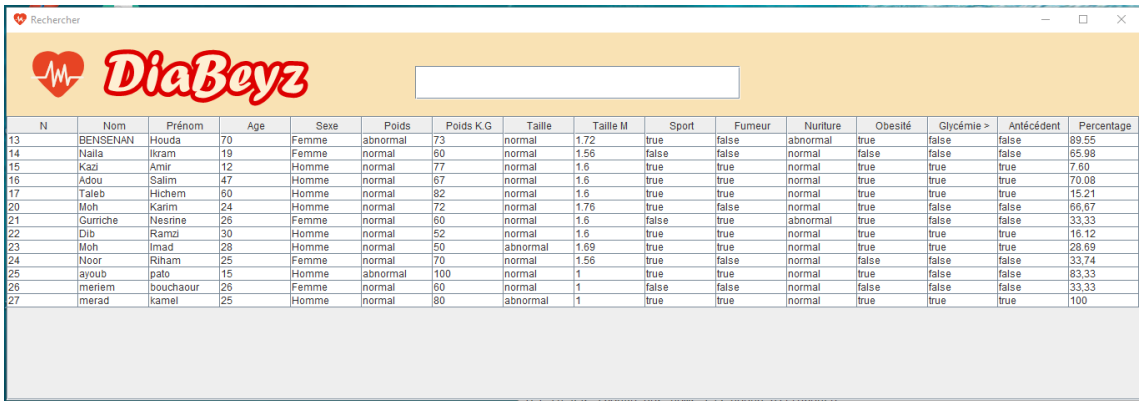
Enfin le système va prendre la décision, nous obtenons une interface graphique indiquant la probabilité d'avoir le diabète, et nous générons le PDF de rapport médicale.

Figure IV.21. L'interface graphique qui indique la probabilité d'avoir le diabète

✓ Liste des patients

La liste des patients permet de garder un historique des patients dans la base de données.

La figure suivante illustre la liste des patients avec leurs informations et la décision finale, l'application donne la possibilité de faire une recherche d'un patient.



N	Nom	Prénom	Age	Sexe	Poids	Poids K.G	Taille	Taille M	Sport	Fumeur	Nuriture	Obesité	Glycémie >	Antécédent	Percentage
13	BENSENAN	Houda	70	Femme	abnormal	73	normal	1.72	true	false	abnormal	true	false	false	89.55
14	Naila	Ikram	19	Femme	normal	60	normal	1.56	false	false	normal	false	false	false	65.98
15	Kazi	Amir	12	Homme	normal	77	normal	1.6	true	true	normal	true	true	true	7.60
16	Adou	Salim	47	Homme	normal	67	normal	1.6	true	true	normal	true	true	true	70.08
17	Taleb	Hichem	60	Homme	normal	82	normal	1.6	true	true	normal	true	true	true	15.21
20	Moh	Karim	24	Homme	normal	72	normal	1.76	true	false	normal	true	false	false	66,67
21	Gurriche	Nesrine	26	Femme	normal	60	normal	1.6	false	true	abnormal	true	false	false	33,33
22	Dib	Ramzi	30	Homme	normal	52	normal	1.6	true	true	normal	true	true	true	16.12
23	Moh	Imad	28	Homme	normal	50	abnormal	1.69	true	true	normal	true	true	true	28.69
24	Noor	Riham	25	Femme	normal	70	normal	1.56	true	false	normal	false	false	false	33,74
25	ayoub	pato	15	Homme	abnormal	100	normal	1	true	true	normal	true	false	false	83,33
26	meriem	bouchaour	26	Femme	normal	60	normal	1	false	false	normal	false	false	false	33,33
27	merad	kamel	25	Homme	normal	80	abnormal	1	true	true	normal	true	true	true	100

Figure II.22. Liste des patients

✓ À propos de l'application

Figure II.23. À propos de l'application

4. Conclusion :

Dans cette dernière partie de notre projet, nous avons présenté les différents outils de développement de notre application ainsi que ses interfaces essentielles.

Conclusion générale et Perspectives

Conclusion générale :

Les réseaux bayésiens, modèle très structuré mathématiquement, offrent des voies nouvelles dans les domaines de l'évaluation et de l'accompagnement automatisé de l'apprenant. Il prend de plus en plus d'ampleur privilégiée au sein des techniques d'Intelligence Artificielle susceptibles d'être applicables en Modélisation de l'Utilisateur.
En effet si l'on prend l'exemple du secteur médical constituant un vaste champ d'application pour l'informatique qui se retrouve de surcroît encouragé à s'intéresser davantage aux outils informatiques et c'est le cas de plusieurs travaux de recherches ayant été menés dans le but de la gestion des établissements de soins et des cabinets médicaux, à la mise au point des systèmes experts, en passant par les systèmes d'aide au diagnostic, nous pouvons constater que les solutions les plus intéressantes et les plus simples, les réseaux bayésiens sur données médicales. Cette dernière permet de faire non seulement un diagnostic précis mais de plus son efficacité reste durable.

Tenant compte de la contribution des réseaux bayésiens dans la facilitation du diagnostic, cette technique devient de plus en plus adoptée dans le monde jusqu'à ce qu'elle remplacera, peut-être, d'ici quelques années nos médecins !

Dans ce contexte, nous avons essayé de réaliser une application à partir de données réelles.

Notre première mission impliquait à orienter nos données dans un domaine bien précis selon le besoin et la préférence des utilisateurs.

Nous avons procédé à cela à travers des questionnaires, interviews, …etc. Nous avons ensuite analysé différents exemples de données afin d'élaborer et synthétiser les solutions.

L'élaboration de la solution représente la phase la plus délicate dans le projet, nous avons consacré beaucoup de temps et d'efforts pour la bonne conception de l'application, de ses fonctionnalités, de son interface. Pour cela, nous avons essayé de choisir ce qui s'adapte le mieux avec notre problématique, par la suite, nous avons développé notre méthode et son architecture de façon à ce qu'elle soit bien adaptée et exploitable par différents utilisateurs.

Enfin, nous conviendrons de dire que les RB sont encore un domaine de recherche mais ils sont d'ores et déjà à l'œuvre dans de nombreux systèmes opérationnels. Il nous semble que leur richesse et leur souplesse devraient leur assurer un large développement dans les années à venir.

Perspectives :

L'application des réseaux bayésiens est une thématique large et sans limites, qui ne cesseront d'évoluer, sans que cela ne se focalise sur un certain type de données ou un domaine rigoureux. A travers notre application, notre premier essai constitue un modèle qui répond à une tâche bien précise, nous nous sommes basés sur des données simples et claires afin de montrer l'enjeu des RB dans un premier temps mais en réalité il est plus fréquent d'être confronté à des applications où les données sont nombreuses, incomplètes mais plus complexes et fortuites, et que les utilisateurs aimeraient exploiter, de manière à en extraire le plus d'informations possible.

C'est pourquoi, maintenant que nous avons une connaissance approfondie sur les réseaux bayésiens, nous avons comme perspectives :

- Tenter d'améliorer l'application en arrivant à un modèle permettant de découvrir des relations, des explications dont les utilisateurs pourront tirer profit.
- Nous avons pensé également qu'il serait avantageux d'étendre l'usage de l'application à d'autres types de maladies.
- Il serait aussi envisageable d'exploiter d'autres domaines, et pourquoi pas parvenir à confluer plusieurs en même temps.

Finalement, nous avons songé à un autre aspect bien que complexe mais fort intéressant d'unifier des approches développées auparavant de façon concurrente, Il est donc tout à fait concevable d'utiliser les principes développés pour les réseaux bayésiens à la découverte de la structure d'autres modèles graphiques.

.

Webographie et Bibliographie

Webographie :

[1] Site officiel de JAVA: https://www.java.com/fr/download/faq/whatis_java.xml - 10/06/2017

[2] Package pédagogique multimédia - Utilisation des bases de données: http://discala.univ-tours.fr/LesChapitres.html/Cours6/Chap6.9.htm - 10/06/2017

[3] Site de mossaique informatique: http://www.mosaique-info.fr/glossaire-web-referencement-infographie-multimedia- informatique/m-glossaire-informatique-et-multimedia/448-mysql-definition.html - NANCY, 10/06/2017

[4] C.SOUTOU. De UML à SQL - La conception de base de données. 2ème édition Eyrolles 2004.

[5] Dictionnaire sensagent le parisien: http://dictionnaire.sensagent.leparisien.fr/PhpMyAdmin/fr-fr/ - 10/06/2017

[6] Site officiel de Netbeans: https://netbeans.org/index_fr.html - 10/06/2017

[7] Site DesGeeksEtDesLettres: https://desgeeksetdeslettres.com/programmation-java/xampp-plateforme-pour-heberger-son-propre-site-web - 10/06/2017 - 10/06/2017

[8] Dictionnaire sensagent leparisien: http://dictionnaire.sensagent.leparisien.fr/HeidiSQL/de-de/ - 10/06/2017

[9] Netica-J Manual Version 4.18 and Higher

Bibliographie:

[10] P.NAIM, P-H. WULLEMIN, Ph.LERAY, O.LERAY, A.BECKER, livre Réseaux bayésiens. EYROLLES.2002

[11] PYNADATH, D. V., WELLMAN, M. P., “Accounting for context in plan recognition, with application to traffic monitoring”, Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence, 1995

[12] Z.ZAABOT. Les Réseaux Bayésiens. Application en reconnaissance de formes à partir d’informations complètes ou incomplètes. Université MOULOUD MAMMERI, TIZI-OUZOU.2012

[13] Ch. GONZALES, P-H.WUILLEMIN Reseaux bayésiens en modélisation d’utilisateurs. Laboratoire d’Informatique de Paris VI (LIP6) Université Paris 6. 1998

Rapport-Gratuit.com

[14] P.LERAY. Réseaux bayésiens : Apprentissage et diagnostic de systèmes complexes. Modélisation et simulation. Université de Rouen, 2006.

[15] O.PARENT, J.EUSTACHE. Les Réseaux Bayésiens A la recherche de la vérité. Université Claude Bernard Lyon 1. Master 2 Recherche Connaissance et Raisonnement, 2006 – 2007.

[16] H-T NGUYEN. Réseaux bayésiens et apprentissage ensembliste pour l'étude différentielle de réseaux de régulation génétique. Intelligence artificielle [cs.AI]. Université de Nantes, 2012.

[17] D. CRAM, M. MAY, R. GUELTON, S. TOUCH. Résumé : Réseaux bayésien. Novembre, 2005

[18] A.BOUZAIE. Réseaux Bayésiens et quelques applications en traitement d'images. Université de Nancy 2. Mars 2014

[19] G. FLORIAN. Développement Entre prouesse technologique et volonté de l'homme, l'intelligence artificielle dépassera elle l'homme.2015/2016.

[20] PETRUSHIN, V., SINITSA, K., "Using probabilistic reasoning techniques for learner modeling", Artificial Intellignece in Education : Proceedings of AIED 1993

[21] L'informatique et la médecine : Diagnostic d'une application.L'economiste.com. Edition N°:22 Le 26/03/1992

[22] E. PRESTAT. Les réseaux bayésiens : classification et recherche de réseaux locaux en cancérologie. Sciences agricoles. Université Claude Bernard - Lyon I, 2010. Français.

[23] F.OLIVIER : de l'identification de structure des réseaux bayésiens à la reconnaissance de forme à partir d'information complètes ou incomplètes. Thèse doctorat.2006.

[24] P.LERAY : Réseau Bayésien. Introduction et apprentissage, modélisation et découverte de connaissances. Université de NANTES Sophia Antipolis. 2008.

[25] P.LERAY, O.FRANCOIS : Etude comparative d'algorithmes d'apprentissage de structure dans les réseaux bayésiens. January 2004

[26] E.LEPAGE, M.FIESCHI, R.TRAINEAU, J. GOUVERNET, C.CHASTANG : Système d'aide à la décision fondé sur un modèle de réseau bayésien application à la surveillance transfusionnelle. Département de Bio statistique et Informatique Médicale, Centre d'Enseignement et de Recherche du Traitement de l'Information Médicale. 1992.

[27] P.LARAY, O.FRANCOIS : Réseaux bayésiens pour la classification méthodologique dans le cadre du diagnostic médical. PhEngRS, vol. 80, no. 3, 2002.

[28] D.BELLOT : Fusion de données avec des réseaux bayésiens pour la modélisation des systèmes dynamiques et son application et télémédecine. Thèse doctorat. Université de Bruxelles. 2002.

[28] L.SMAIL : Algorithmes pour les Réseaux Bayésiens et leurs extensions. Thèse doctorat de l'université de Polytech Nantes. 2004.

[29] J. SACHA, L. GOODENDAY, and K. CIOS. Bayesian learning for cardiac spect image interpretation. Artificial Intelligence in Medecine, 26 :109– 143, 2002.

[30] L. van der Gaag, S. Renooij, C. Witteman, B. Aleman, and B. Taal. Probabilities for a probabilistic network : a case study in oesophageal cancer. Artificial Intelligence in Medicine, 25(2) :123–148, june 2002.

[31] O. FRANCOIS and P. LERAY. Evaluation d'algorithmes d'apprentissage de structure pour les réseaux bayésiens. In Proceedings of 14ème Congrès Francophone Reconnaissance des Formes et Intelligence Artificielle, RFIA 2004, pages 1453–1460, Toulouse, France, 2004.

[32] J.ZHANG. Causal Inference and Reasoning in Causally Insufficient Systems. PhD thesis, Carnegie Mellon University, July 2006.

[33] S. MEGANCK, P. Leray, S. Maes, and B. Manderick. Apprentissage des réseaux bayésiens causaux à partir de données d'observation et d'expérimentation. In Proceedings of 15ème Congrès Francophone Reconnaissance des Formes et Intelligence Artificielle, RFIA 2006, page (CDROM), Tours, France, 2006.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

ANNEXES

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

ANNEXE 1 : Exemple de EM.

Soit la base de données suivante :

M	F	R
m0	Bad	O
m0	Bad	O
?	Bad	O
m0	Bad	O
?	Bad	N
m0	Ok	O
m1	Bad	O
m1	Bad	N
?	Ok	O
m1	Ok	N
m1	Ok	O
m1	Ok	N
m1	?	O
m1	Ok	N
m2	Ok	N

Pour estimer P (M).

1) initialisation

$p^{\wedge(0)}(m) = [1/3 \;\; 1/3 \;\; 1/3]$.

1ère itération [E] :

M	F	R	P(M=m0)	P(M=m1)	P(M=m2)
m0	Bad	O	1	0	0
M0	Bad	O	1	0	0
?	Bad	O	1/3	1/3	1/3
M0	Bad	O	1	0	0
?	Bad	N	1/3	1/3	1/3
M0	Ok	O	1	0	0
M1	Bad	O	0	1	0
M1	Bad	N	0	1	0
?	Ok	O	1/3	1/3	1 /3
M1	Ok	N	0	1	0
M1	Ok	O	0	1	0
M1	Ok	N	0	1	0
M1	?	O	0	1	0
M1	Ok	N	0	1	0
M2	Ok	N	0	0	1
	total		5	8	2

1er itérarion [M] :

$P^{(1)}(m0)=5/15=0.333$

$P^{(1)}(m1)=8/15=0.533$

$P^{(1)}(m2)=2/15=0.133$

2eme itérarion [E] :

M	F	R	P(M=m0)	P(M=m1)	P(M=m2)
m0	Bad	O	1	0	0
M0	Bad	O	1	0	0
?	Bad	O	0.333	0.533	0.133
M0	Bad	O	1	0	0
?	Bad	N	0.333	0.533	0.133
M0	Ok	O	1	0	0
M1	Bad	O	0	1	0

?• O	?	O	0	1	0
?• O					
?• O					
?• O					
?• O					
?• O					
?• O					
?• O					
?• O					
?• O					