Réseaux bayésiens et apprentissage ensembliste pour l'étude différentielle de réseaux de régulation génétique

Hoai-Tuong Nguyen

N° attribué par la bibliothèque

UNIVERSITÉ DE NANTES
ÉCOLE POLYTECHNIQUE DE
L'UNIVERSITÉ DE NANTES

ÉCOLE DOCTORALE STIM
SCIENCES ET TECHNOLOGIES DE
L'INFORMATION&MATHÉMATIQUES

T H È S E

pour obtenir le titre de

DOCTEUR EN SCIENCES
DE L'UNIVERSITE DE NANTES

Mention : Informatique

présentée par

Hoai Tuong NGUYEN

Réseaux bayésiens et apprentissage ensembliste pour l'étude différentielle de réseaux de régulation génétique

Directeur de thèse : **Philippe LERAY**
Co-encadrant : **Gérard RAMSTEIN**

soutenue **le 27 janvier 2012**

Jury :

Rapporteurs :

Salem BENFERHAT, Professeur, Université d'Artois
Sylvain PIECHOWIAK, Professeur, Université de Valenciennes

Examinateurs :

Philippe LERAY, Professeur, Université de Nantes
Gérard RAMSTEIN, Maître des Conférences, Université de Nantes

Président :

Salem BENFERHAT, Professeur, Université d'Artois

Cette recherche est soutenue par
Région PAYS DE LA LOIRE

Remerciements

Je tiens à remercier avant tout, **Philippe LERAY**, mon directeur de thèse. Il m'a toujours suggéré de bonnes directions de recherche et m'a laissé travailler de manière autonome quand il le fallait. Il a le talent de savoir expliquer les concepts les plus techniques en des termes très faciles à comprendre, surtout pour un étranger comme moi. Il m'a appris énormément.

La deuxième personne à qui je voudrais adresser mes remerciements est sans doute, **Gérard RAMSTEIN**. Etant mon encadrant depuis le master puis la thèse, il a grandement contribué à ma maturité scientifique. En effet, il me répondait avec patience aux questions d'un jeune naïf en bioinformatique. J'ai sincèrement apprécié de travailler avec lui et lui suis reconnaissant pour le temps qu'il m'a consacré.

Je remercie également **Salem BENFERHAT** et **Sylvain PIECHOWIAK** d'avoir accepté d'être rapporteurs de cette thèse pour leurs remarques et suggestions importantes.

Ce manuscrit n'aurait pas pu voir le jour sans la relecture de **Josiane** et **Alain MALEINGE**, **Marie-Noël PASTRE, Fernande et François CAVIGNAC**. Je suis très touché par leur dévouement et leur en remercie particulièrement.

Je voudrais remercier sincèrement **Henri BRIAND**, **Pascale KUNTZ-COSPEREC**, **Xuan Hiep HUYNH** qui sans être mes directeurs de thèse mais m'ont donné de bons conseils, des encouragements efficaces et des opportunités importantes pour venir faire des études en France.

Par ailleurs, toutes mes reconnaissances à **Yannick JACQUES**, directeur de l'U892 INSERM, pour m'avoir accueilli dans son laboratoire.

J'ai eu la chance tout au long de cette thèse d'être aidé par des assistantes administratives très efficaces. Merci beaucoup donc à **Sylvie LEROUX** et **Annie BOILOT** (LINA - Laboratoire d'Informatique de Nantes/Atlantique), **Corinne LORENTZ** (Polytech'Nantes), **Yvan LEMEUR** et **Stéphanie LEGEAY** (Ecole doctorale STIM - Sciences et Technologies de l'Information et Mathématiques), **Nathalie DE BROVES** (Université de Nantes), **François RESCHE**, **Caroline SEZESTRE** et **Soline PUENTE RODRIGUEZ** (Association Chercheurs Etrangers à Nantes), **Patricia TORRES**

et **Sébastien YOUINOU** (Maison des Echanges Internationaux et de la Francophonie).

Je remercie mes collègues de bureau à Polytech'Nantes, **Toader GHERASIM, Zohra BEN SAÏD, Claudia MARINICA, Raphaël MOURAD, Thomas PITON, Yasin AMANULLAH, Zineddine KOUAHLA, Lionel CHAUVIN, Michael LËON** qui sont beaux, intelligents et très sympathiques. Ils m'ont enrichi de connaissances sur leur culture.

Un grand merci à mes amis de l'Association des étudiants vietnamiens de Nantes (AEVN) et de l'Association des vietnamiens de Loire/Atlantique (AVLA) qui m'ont donné des occasions de travailler dans une ambiance très nostalgique.

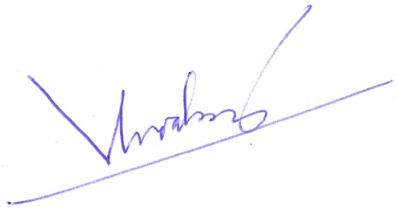
Je tiens à remercier spécialement aux moines **Nguyen Loc THICH** et **Nguyen Hung THICH** à la Pagode de Van Hanh (Nantes) qui m'ont donné beaucoup de conseils très utiles pour remonter le moral et travailler dans un esprit purement tranquille.

Un remerciement spécial à ma famille d'accueil **MALEINGE**, en particulier **Josiane MALEINGE, Alain MALEINGE, Jérôme MALEINGE et Julien MALEINGE**, qui m'ont beaucoup aidé dans l'intégration à la vie nantaise.

Un grand merci au fond du coeur à mon père **Van Khai NGUYEN**, ma mère **Thi Nguyet NGUYEN**, ma soeur **Thi To Nhu NGUYEN** dont l'affection m'a rendu la vie vraiment plus agréable.

Enfin mille mercis à ma femme **Phuong Nga NGUYEN** pour avoir toujours su me soutenir et parfois garder le moral pour moi.

Nantes, le 27 janvier 2012



Hoai Tuong NGUYEN

A ma grand-mère paternelle...

Résumé

Dans les dernières années, les réseaux Bayésiens (RB) sont devenus l'une des méthodes d'apprentissage automatique les plus puissantes permettant de modéliser graphiquement et de manière probabiliste différentes types de systèmes complexes. Un des problèmes communs dans l'apprentissage de la structure des RB est le problème des données de petites tailles. En effet, le résultat de l'apprentissage est sensible au nombre d'échantillons de données. En apprentissage automatique, les méthodes d'apprentissage ensemblistes telles que le bootstrap ou les algorithmes génétiques sont des méthodes souvent utilisées pour traiter le problème de la pauvreté de données. Toutefois, les méthodes existantes se limitent généralement à la fusion d'un ensemble de modèles, mais ne permettent pas de comparer deux ensembles de modèles. Inspiré par les résultats obtenus par les méthodes ensemblistes, nous proposons une nouvelle méthode basée sur le graphe quasi-essentiel (QEG - Quasi-Essential Graph) et l'utilisation d'un test multiple afin de comparer deux ensembles de RB. Le QEG permet de résumer et de visualiser graphiquement un ensemble de RB. Le test multiple permet de vérifier si les différences entre les deux ensembles de RB sont statistiquement significatives et de déterminer la position de ces différences. L'application sur des données synthétiques et expérimentales a démontré les différents intérêts de la méthode proposée dans la reconstruction des réseaux de régulation génétique et perspectivement dans les autres applications avec les données de petites tailles.

Mots-clés : *réseaux bayésiens, apprentissage ensembliste, étude différentielle, réseaux de régulation génétique*

Abstract

In the recent years, the Bayesian networks (BN) have become one of the most powerful machine learning methods to modeling graphically and probabilistically different kinds of complex systems. One of the common issues in BN structure learning is the small-data problem. In fact, the result of learning is sensible to sample size of dataset. In machine learning, the set-based learning methods such as Bootstrap ou genetic algorithms are the often used methods to dealing with the small-data problem. However, the existing methods limit generally to the fusion of a set of models, but do not allow to compare two set of models. Inspired from the obtained results of the set-based methods, we proposed a novel method based on the quasi-essential graph (QEG) and the usage of the multiple testing in order to compare two sets of BN. QEG allows to resume and visualize graphically a set of BN. The multiple testing allows to verify if the differences between two set of BN are statistically significative and to determine the position of the differences. The application on the synthetic and experimental data demonstrated the different interests of proposed method in gene regulatory networks reconstruction and perspectively in the other applications with the small dataset.

Keywords : *bayesian networks, set-based learning, differential study, gene regulatory networks*

Table des matières

Table des figures

Table de notations

D	jeu de données
d_i	ensemble de données obtenu par bootstrap
X, Y, Z, U, V	ensembles de variables aléatoires
χ	ensemble de configurations de X
X_i	variable aléatoire
x_i	valeur de X_i
$P(X)$	distribution de probabilité sur l'ensemble de X
$P(X_i)$	probabilité de X_i
$Pa(X_i)$	parents de X_i dans un graphe donné
$pa(X_i)$	valeur des parents de X_i
r_i	nombre de configurations de X_i
q_i	nombre de configurations des parents de X_i
N_{ijk}	nombre d'exemples dans D où $X_i = x_i^k$ et $Pa(X_i) = pa^j(X_i)$
N_{ij}	nombre d'exemples dans D où $Pa(X_i) = pa^j(X_i)$
N	nombre total d'exemples dans D

Introduction

L'informatique devient de plus en plus essentielle dans plusieurs domaines. La recherche biomédicale en fait partie. Alors que les données biologiques deviennent de plus en plus volumineuses, il y a de plus en plus de problèmes non résolus par des connaissances d'experts. La naissance de la bioinformatique [Baldi 2001] apporte un outils précieux pour chercher systématiquement des solutions pour ces problèmes. C'est un domaine résultant d'un croisement issu de *la biologie*, de *l'informatique* et de *la statistique*.

Les gènes sont à la base de toute vie terrestre. Les modifications sur l'expression des gènes peuvent poser des problèmes responsables éventuellement de maladies. En effet, les régulations entre gènes gouvernent et produisent des fonctions biologiques. Le changement de régulation génétique dans différentes conditions expérimentales permet de prédire des effets significatifs en terme de pathologie. *La comparaison de régulation génétique* entre conditions expérimentales donne des connaissances pour la prédiction biomédicale.

Comme le nombre de gènes dans le patrimoine génétique est très grand (des milliers de gènes) [Elati 2007], l'identification et l'interprétation de ces relations nécessitent par conséquent des apports fondamentaux d'outils de la bioinformatique pour :

- *le stockage des données,*
- *le traitement des données,*
- *l'extraction des connaissances à partir de données,*
- *l'évaluation et la visualisation des connaissances obtenues.*

D'après [Baldi 2001], le volume des données crée un besoin central en recherche théorique et expérimentale pour la récupération, le traitement, l'ana-

lyse, la navigation et la visualisation. Cependant, dans le cadre de notre travail, les problématiques ne reposent pas sur la grande quantité de données biologiques mais sur le nombre de variables observées et le nombre d'échantillons de données collectées. Car un des problèmes communs liés aux données biologiques est que *le nombre d'échantillons est trop petit par rapport au nombre des variables observées*. Ce problème pose par conséquent un grand défi pour les algorithmes d'apprentissage automatique qui sont sensibles au nombre de données.

Cette situation a, par exemple, donné lieu à plusieurs travaux hors du cadre de notre étude [Stevens 2005, Rhodes 2002] sur la méta-analyse de données dont le but est de travailler conjointement avec différents jeux de données.

Cette thèse présente un cadre théorique et expérimental de différentes approches qui permettent de *reconstruire*, *visualiser* et *comparer* des réseaux de régulation entre des gènes à partir de données, notamment dans le contexte de données de petites tailles.

L'inspiration de ce travail repose sur des solutions pour les problématiques dans l'apprentissage de réseaux bayésiens, l'un des outils qui répond à la majeure partie des questions posées ci-dessus.

Dans un premier temps, nous ciblons la modélisation (construction) de réseaux de régulation génétique. Comme la construction d'un modèle requiert une masse de connaissances sur les interactions entre gènes, l'utilisation de l'apprentissage automatique à partir de données expérimentales pour identifier des gènes potentiellement co-régulés devient appropriée. A ce titre, nous utilisons les réseaux bayésiens, une approche de l'apprentissage automatique, qui permettent d'obtenir à la fois un modèle graphique et probabiliste des variables observées.

Ensuite, dans le but d'augmenter la robustesse de la comparaison notamment lorsqu'il y a très peu de données disponibles, nous choisissons des approches ensemblistes pour l'apprentissage de la structure de réseaux bayésiens. Ces approches permettent d'obtenir un ensemble de bonnes structures. Cet ensemble nous permet ensuite de nous positionner dans le cadre d'une étude

différentielle de réseaux bayésiens. Cette étude se base sur l'utilisation d'un test multiple et d'une méthode de visualisation.

Les contributions de la thèse sont les suivantes :

- *l'évaluation et visualisation d'une population de réseaux bayésiens. Cette contribution a été présentée lors de la conférence ASMDA'2011 [Nguyen 2011b]*
- *l'étude différentielle de deux populations de réseaux bayésiens. Cette contribution a été présentée lors de la conférence KES'2011 [Nguyen 2011a]*
- *l'architecture générale de la reconstruction de réseaux de régulation génétique par réseaux bayésiens et de l'étude différentielle d'une population de réseaux de régulation génétique par des méthodes ensemblistes. Une première version de cette contribution a été présentée lors des conférences JOBIM'2009 [Nguyen 2009] et ACIIDS'2010 [Nguyen 2010]*
- *la validation sur des données synthétiques.*

La Figure 1 présente l'architecture générale de cette thèse :

Dans le *Chapitre 1,* nous présentons l'état de l'art sur les méthodes pour la reconstruction de réseaux de régulation génétique où les réseaux bayésiens sont présentés comme l'un des meilleurs outils qui nous permettent de résoudre des problèmes liés à la nature de données biologiques.

Dans le *Chapitre 2,* nous abordons l'état de l'art sur l'apprentissage de réseaux bayésiens et les méthodes d'évaluation des algorithmes d'apprentissage de la structure.

Dans le *Chapitre 3,* nous introduisons des méthodes existantes pour l'apprentissage ensembliste et proposons le QEG (Quasi-Essential Graph) pour évaluer et visualiser un ensemble de réseaux bayésiens.

Dans le *Chapitre 4,* nous analysons plusieurs solutions possibles et proposons une méthode basée sur un test multiple pour l'étude différentielle de deux ensembles de réseaux bayésiens.

Dans le *Chapitre 5,* nous appliquons les méthodes proposées sur un benchmark classique afin de valider nos algorithmes.

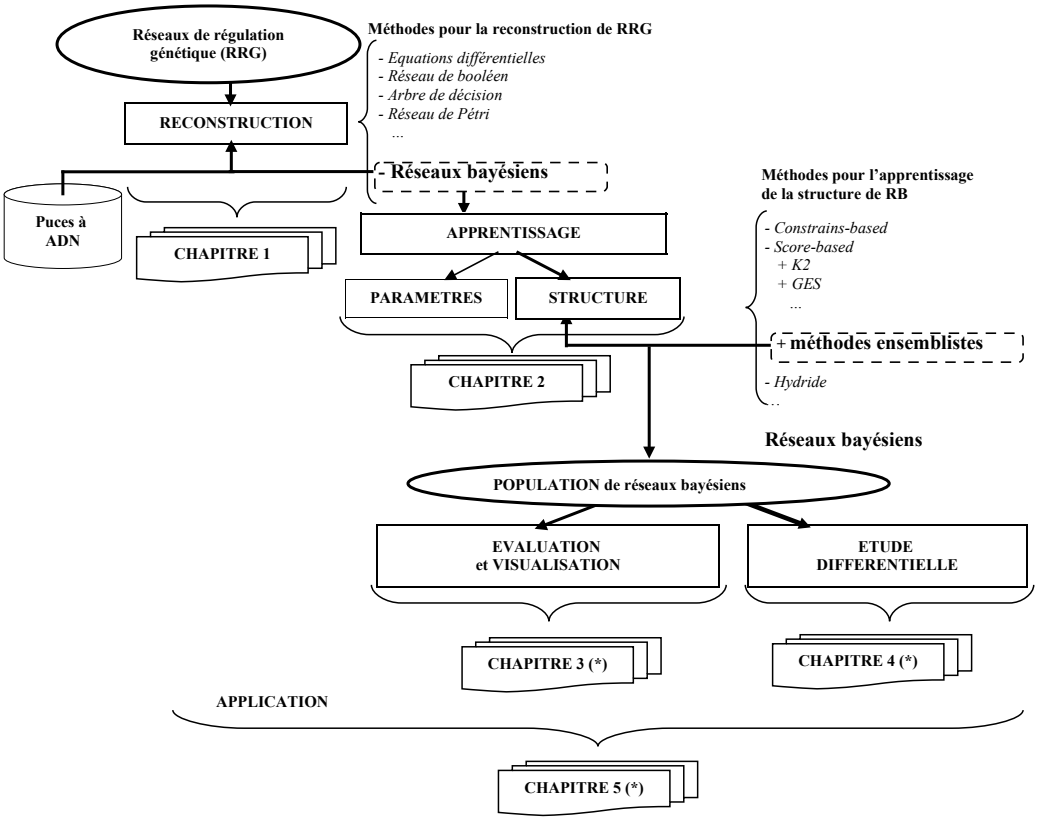


FIGURE 1 – Structure globale de la thèse. Les chapitres marqués en (*) sont des contributions principales de la thèse.

Chapitre

1

Reconstruction de réseaux de régulation génétique

Sommaire

Introduction

La régulation génétique coordonne tout processus de vie, y compris dans la différenciation cellulaire, le métabolisme, le cycle cellulaire et la transduction du signal [Karlebach 2008]. L'objectif général d'un réseau de régulation génétique (RRG) est de décrire la manière dont les gènes sont activés ou réprimés. La modélisation de RRG à partir de données d'expression génétique permet de comprendre plus finement les implications des gènes et est donc d'un apport capital pour l'industrie pharmaceutique et pour de nombreux autres domaines. C'est une des techniques qui permettent une meilleure compréhension de la génomique fontionnelle. Une grande partie des données expérimentales disponibles aujourd'hui porte sur ces réseaux. C'est la raison pour laquelle l'analyse

des RRG nécessite des outils de modélisation pour analyser visuellement des relations entre des gènes, mais aussi des techniques performantes pour traiter les données.

En effet, la nature des données d'expression pose des difficultés de reconstruction des RRG. Les valeurs d'expression sont entachées de bruit réduisant la précision des RRG inférés par des méthodes informatiques. Par ailleurs, les niveaux d'expression sont continus, et donc ne sont pas toujours compatibles avec les méthodes de modélisation. Enfin, le nombre de gènes est beaucoup plus grand par rapport au nombre d'échantillons, créant d'une part un problème de dimension, et, d'autre part, un problème de pauvreté de données. Cette difficulté provient des coûts de la technologie et parfois de la variété des échantillons.

Ce chapitre présente la reconstruction de RRG à partir de données d'expression de gènes. La reconstruction de réseau de régulation est un problème multi-dimensionnel, car pour une étude précise, un grand nombre de gènes peut être concerné simultanément. Ajoutons à cela le fait que les données d'expression mesurent le niveau d'expression d'un très grand nombre de gènes différents à partir d'un jeu réduit d'échantillons. Donc, l'extraction de connaissances pertinentes à partir des données d'expression génétiques est un véritable défi. Pour aborder cette question, il est nécessaire de mettre en oeuvre une démarche de développement des approches computationnelles pour modéliser les RRG. Quelle que soit la méthode utilisée, la difficulté principale consiste à trouver des paramètres qui permettent de décrire le plus précisément possible le caractère du réseau à partir des observations expérimentales. Par la suite, nous présentons quelques méthodes parmi les plus utilisées dans la littérature afin de mieux situer l'inspiration de notre proposition méthodologique.

Le RRG permet de mieux comprendre visuellement le fonctionnement des gènes dans une vue globale de notre patrimoine génétique. Donc, une meilleure connaissance du réseau de régulation d'un ensemble de gènes est une aide essentielle dans l'analyse de caractères complexes susceptibles d'induire des maladies. C'est la raison pour laquelle la recherche sur la reconstruction de

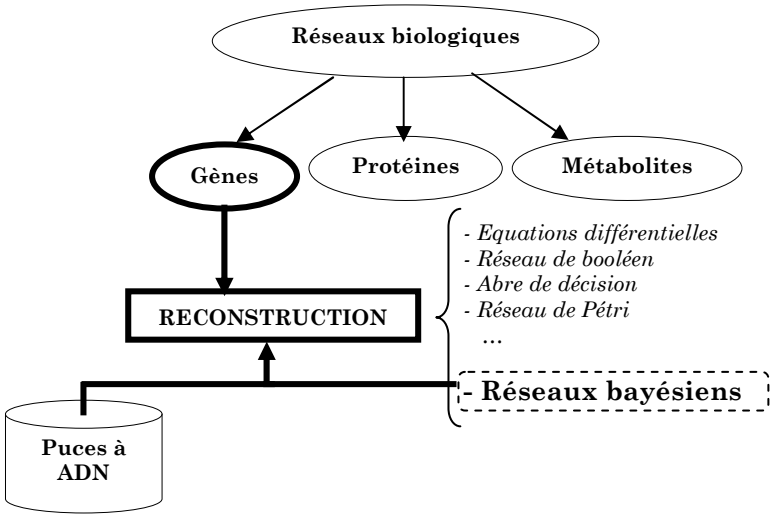


FIGURE 1.1 – L'organisation intuitive des notions-clés du Chapitre 1. Il existe plusieurs types de *Réseaux biologiques* dont *les réseaux de gènes* (aussi appelés *les réseaux de régulation génétique*), *les réseaux de protéines* et *les réseaux métaboliques*. Il y plusieurs méthodes pour la *reconstruction* de *réseaux de régulation génétique* : *Equations différentielles, Réseau de booléen, Abre de décision, Réseau de Pétri...*. Parmi ces méthodes, les *réseaux bayésiens* sont utilisés dans cette thèse.

réseaux de régulations génétiques attire particulièrement l'attention des scientifiques. En effet, la recherche sur la régulation génétique avance sans cesse de manière plus rapide. L'un des exemples les plus frappants de cette évolution est sans doute les *biopuces à ADN*. Elles permettent aux outils de modélisation de visualiser simultanément le niveau d'expression d'un ensemble de plusieurs milliers de gènes. Notons également l'avènement des techniques de séquençage à très haut débit qui permettent d'analyser les expressions des gènes. Bien qu'encore coûteuse, cette technique dénommée RNA-Seq est une évolution majeure en génomique. Toutefois, cette évolution technologique nécessite des méthodes de modélisation correspondant aux différentes problématiques posées. Cette nécessité renforce l'intérêt de la recherche interdisciplinaires non seulement des biologistes, mais aussi des statisticiens et des informaticiens. La Figure 1.1 résume une organisation intuitive des notions-clés de ce chapitre.

1.1 Réseaux de régulation génétique

Nous présentons brièvement dans cette sous-section les notions de base en génétique que nous utilisons dans la suite de cette thèse. Une présentation approfondie est disponible dans [Griffths 2007].

1.1.1 Notions de base

Nous rappelons brièvement dans cette section des notions essentielles de biologie moléculaire.

Définition 1. ***Cellule***
Une ***cellule*** *est une unité structurale, fonctionnelle et reproductrice constituant tout ou partie d'un organisme.*

Définition 2. ***ADN***
*L'****ADN*** *(acide désoxyribonucléique) est une molécule, présente dans toutes les cellules vivantes, qui renferme l'ensemble des informations nécessaires au développement et au fonctionnement d'un organisme. L'ADN porte l'information génétique.*

Définition 3. ***Chromosome***
Un ***chromosome*** *est un élément constitué de molécules d'ADN qui apparaît dans le noyau des cellules au moment de leur division.*

Définition 4. ***Gène***
Un ***gène*** *est une séquence d'ADN qui correspond à une unité d'hérédité : le gène est l'élément d'information codant un caractère particulier.*

Définition 5. ***Protéine***
Une ***protéine*** *est une macromolécule, présente dans toutes les cellules vivantes et indispensable à la vie et au maintien des organismes.*

Définition 6. ***ARN***
Un ***ARN*** *(acide ribonucléique) est une molécule, présente dans toutes les cellules vivantes, qui est une copie d'une région de l'un des brins de l'ADN.*

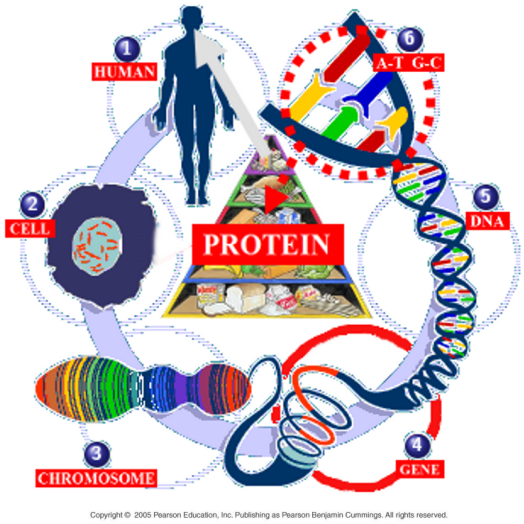


FIGURE 1.2 – De l'homme à l'ADN [Campbell 2005]. *Chez l'homme (1), dans le noyau d'une cellule (2) il y a plusieurs chromosomes (3) qui sont constitués par plusieurs gènes (4). Chaque gène est une séquence de l'ADN qui est construit par quatre lettres de bases "A" (adenine), "G" (guanine), "C" (cytosine) et "T" (thymine).*

Définition 7. ***ARNm***

Un ***ARNm*** *(acide ribonucléique messager) est une molécule qui porte l'information génétique d'un ou plusieurs gènes qui codent pour des protéines.*

Définition 8. ***Intron***

Un ***intron*** *est une région de l'ADN qui ne sert pas à coder des protéines.*

Définition 9. ***Exon***

Un ***exon*** *est une région de l'ADN qui sert à coder des protéines.*

1.1.2 De la cellule à l'ADN

Un organisme est une forme de vie individuelle comme une plante, un animal, une bactérie, ou un champignon. Les organismes sont divisés en deux grandes familles : les *procaryotes*, organismes unicellulaires sans noyau et les *eucaryotes*, organismes dont les cellules ont un noyau qui renferme l'*information génétique* porté par l'ADN. La Figure 1.2 illustre ces différentes cellules chez l'être humain.

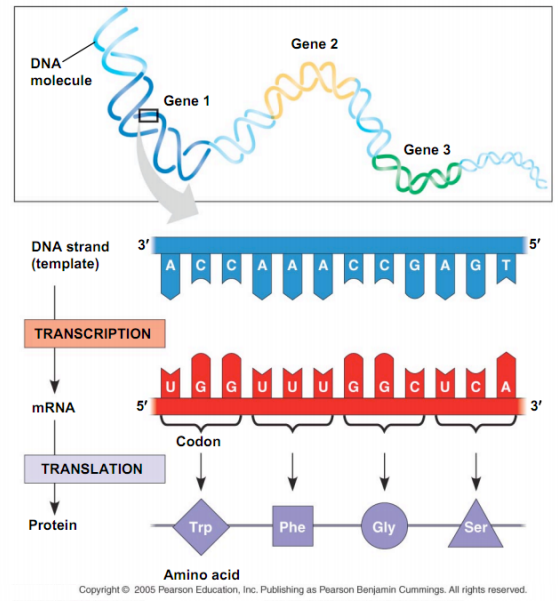


FIGURE 1.3 – Le processus de la synthèse de protéines [Campbell 2005].

1.1.3 De l'ADN à la protéine

Définition 10. ***Transcription***

La ***transcription*** *est un processus biologique qui consiste à copier des régions codantes de l'ADN en ARNm. Les ARNm sont des ARN qui contiennent des exons d'un ou plusieurs gènes qui codent pour des protéines.*

Définition 11. ***Traduction***

La ***traduction*** *est l'étape d'interprétation des exons de l'ARN messager en acides aminés pour produire les protéines.*

La transcription et la traduction sont les deux principaux processus dans la synthèse de protéines (voir Figure 1.3). Pour produire des protéines, la première étape consiste à transcrire l'ADN en ARNm . La deuxième étape consiste ensuite à traduire l'ARNm en protéine.

Dans la synthèse des protéines, le rôle des gènes est très important. Ils sont considérés comme les "recettes" qui contiennent les instructions en "langage" particulier utilisant un alphabet de 4 lettres (A, C, G et T). L'ordre de ces lettres, c'est-à-dire la séquence des gènes, détermine la forme et la fonction de la protéine dans l'organisme. Le processus de la transcription est aussi appelé

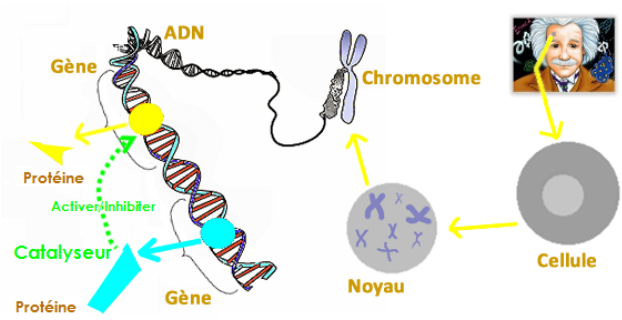


FIGURE 1.4 – Schéma de la régulation génétique. L'image extraite de Wikipedia

l'*expression génétique* car pour produire les protéines, les gènes doivent être *exprimés*. Dans la sous-section ci-après, nous présentons comment un gène devient "exprimé".

1.1.4 Régulation génétique

Le comportement d'une cellule est déterminé par la concentration de protéines particulières (voir la Figure 1.4). Ces protéines sont appelées des protéines régulatrices (ou simplement *régulateurs*). Ce sont des facteurs de transcription qui peuvent se fixer sur des sites spécifiques, appelé *sites de fixation* situés en amont des gènes (région appelée *"région promotrice du gène"*). Par l'existence d'un catalyseur (une substance qui augmente ou diminue la vitesse d'une réaction chimique), ces protéines *activent* ou *inhibent* l'expression d'un gène. Les régulateurs qui peuvent eux-mêmes être régulés, participent à une voie de régulation génétique.

En général, un gène cible peut être régulé par une combinaison de facteurs de transcription, et un facteur de transcription peut réguler plusieurs gènes cibles. Pour modéliser cette régulation, les biologistes ont adopté le concept d'un réseau. Ce réseau s'appelle *le réseau de régulation génétique*.

1.1.5 Réseaux de régulation génétique

La biologie moléculaire fait donc fortement appel à la notion de réseaux, dans des contextes différents et impliquant des entités biologiques distinctes. L'un des modèles les plus étudiés est probablement le réseau de régulation

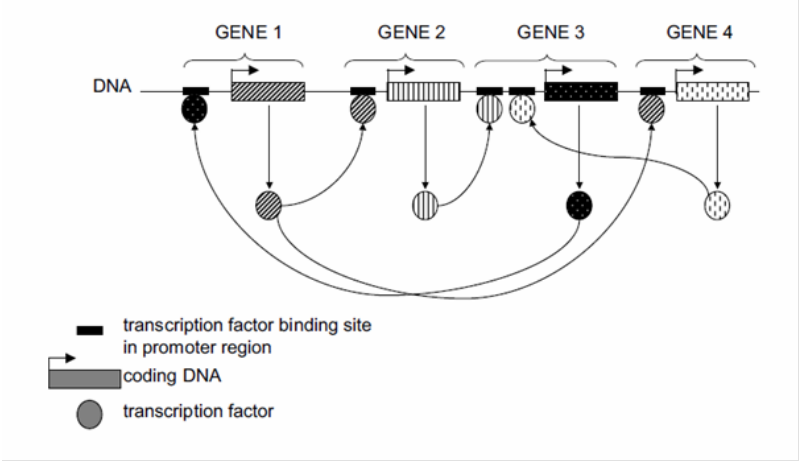


FIGURE 1.5 – Exemple fictif de réseau de regulation transcriptomique [Schlitt 2007]. *Les grands rectangles représentent la partie codante du génome, en amont desquels figurent les sites de liaison (région promotrice des gènes). Les cercles représentent les facteurs de transcription.*

transcriptomique se focalisant sur les relations existantes entre les gènes et les produits de gènes. Ce type de modélisation a pour but de définir la manière dont un gène est régulé en réponse à certains signaux. Dans les années 60, les biologistes ont montré que les gènes possèdent des séquences proches servant à leur régulation et que des protéines sont capables de se lier à ces séquences, permettant ainsi le contrôle de l'expression du gène selon deux modes, l'activation ou la répression. Comme ces protéines régulatrices sont elles-mêmes le produit de gènes, un réseau de régulation se forme, avec ses boucles de rétro-action positives et négatives. La figure 1.5 illustre ce mécanisme de régulation. Cet exemple de réseau est bien évidemment une simplification de l'activité biologique réelle qui possède d'autres contraintes (de régulation post-transcriptionnelle par exemple).

Définition 12. ***Réseau de régulation génétique***

Un réseau de régulation génétique (RRG) est un graphe orienté et signé qui indique quelles sont les activations et les inhibitions présentes entre les gènes du réseau. Dans le RRG, chaque noeud est un gène et chaque lien entre deux noeuds représente une interaction génétique entre deux gènes. Les interactions

se traduisent par des influences sur le niveau d'expression des gènes. Le RRG est aussi appelé le réseau d'interaction entre les gènes.

Outre les réseaux de régulation, il existe d'autres types de réseaux biologiques qui ne font pas l'objet de ces travaux mais qui possédent une importance capitale. Citons par exemple :

- *les réseaux d'interaction des protéines* : les protéines qui sont connectées à des interactions physiques ou métaboliques et les voies de signalisation de la cellule,
- *les réseaux métaboliques* : produits métaboliques et les substrats qui participent à une réaction.

Les interactions entre gènes dépendent de nombreux paramètres de différentes natures. C'est la raison pour laquelle la structure des réseaux de régulations génétiques est encore largement méconnue. La section suivante présente différentes problématiques ainsi que différentes solutions dans le cadre du problème de la reconstruction de réseaux de régulation génétique.

1.2 Reconstruction de réseaux de régulation génétique

1.2.1 Problématiques

Les RRG permettent de contrôler le fonctionnement et le développement des organismes vivants via des relations entre gènes. L'étude sur des RRG pose des problématiques importantes et diverses.

Premièrement, une proportion très importante de tous les génomes eucaryotes est composée de la classe d'ADN non-codant (Intron). Cela veut dire qu'il y a très peu de gènes qui peuvent éventuellement coder pour les protéines (voir Tableau 1.1).

Deuxièmement, tous les gènes ne peuvent être exprimés à la fois. Pour caractériser un organisme vivant, chaque gène s'exprime dans un environnement spécifique, à des moments précis et pendant une durée limitée en fonction de son niveau d'expression. En effet, seul un petit nombre de gènes fonctionne comme un activateur ou un inhibiteur, par conséquence leur identification est un problème important et difficile. Il faudrait également avoir un mécanisme

Organisme	Taille du génome	Nbre de gènes	Nbre de FT	ADN non-codant
Colibacille	$\sim 4.6Mb$	~ 4300	~ 100	$\sim 10\%$
Levure	$\sim 13Mb$	~ 6200	~ 250	$\sim 30\%$
Homme	$\sim 3200Mb$	~ 20000	~ 1700	$\sim 98\%$

TABLE 1.1 – Exemple des génomes eucaryotes, avec la taille du génome, le nombre de gènes, le nombre des facteurs de transcription (FT) connus et le pourcentage d'ADN non codant. D'après [Elati 2007].

pour identifier quels sont les gènes qui sont les cibles de ces facteurs.

Troisièmement, la fonction biologique est souvent sous-estimée et la structure de RRG est encore largement méconnue.

Par ailleurs, les problèmes liés aux données sont un autre problème classique dans le cadre d'analyse de réseaux de régulation génétique à partir de données. Nous présentons dans la section suivante la nature de données expérimentales et les techniques de traitement de données.

1.2.2 Données expérimentales et traitement des données

1.2.2.1 Puces à ADN

Définition 13. ***Puces à ADN*** *[Madan Babu 2004]*
*Une puce à ADN (aussi appelées puces à gènes, biopuces, ou en anglais "*DNA chip, DNA-microarray, biochip*") est un ensemble de molécules d'ADN fixées en rangées ordonnées sur une petite surface qui peut être du verre, du silicium ou du plastique. C'est une technique qui permet d'analyser et de quantifier simultanément l'expression de plusieurs millier de gènes.*

Les puces à ADN [Lander 1999] reposent sur le mécanisme d'hybridation de la double hélice d'ADN. Ce sont des lames de petite taille sur lesquels on peut synthétiser de milliers de gènes dont des séquences d'ADN. Ensuite, elles sont hydridées avec l'ensemble des transcrits issus d'une cellule cible pour marquer leur niveau d'expression grâce à leur fluorescente. L'hybridation est une interaction des deux chaînes de séquences complémentaires (liaisons hydrogènes)

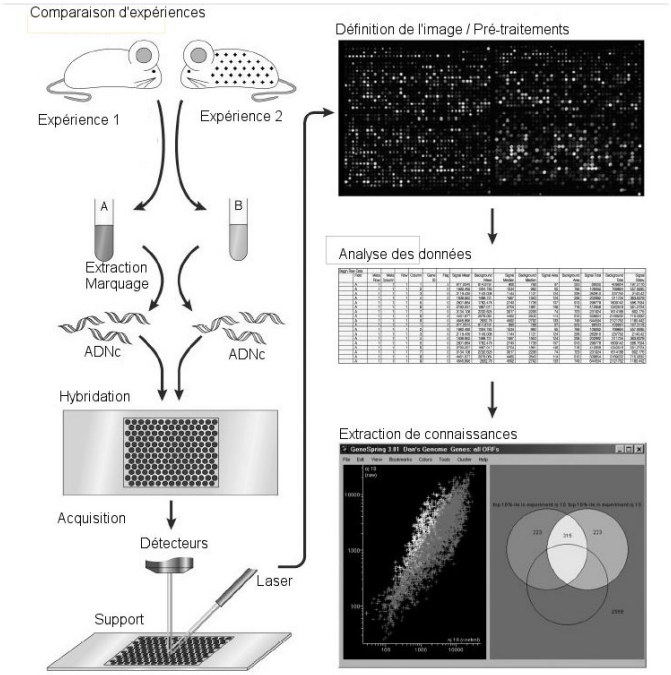


FIGURE 1.6 – Schéma de principe des expériences à base de puces à ADN. [Barra 2004]

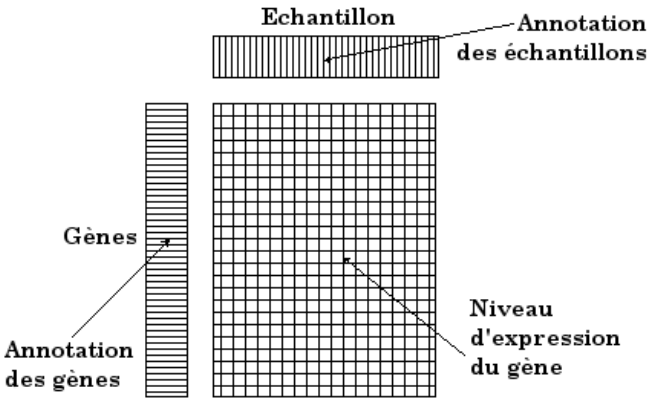


FIGURE 1.7 – Exemple d'une matrice de données d'expression des gènes issues de puces à ADN.

Dans les puces à ADN, les changements des niveaux d'expression des gènes provenant de différents échantillons fournissent des informations permettant des techniques d'ingénierie inverse (*"reverse engineering"* en anglais) pour reconstruire le réseau de régulation génétique. Pour les méthodes d'apprentissage à partir de données, l'utilisation de données de puces à ADN pose les

deux problématiques suivantes :

- *premièrement*, il est nécessaire d'avoir un nombre suffisant de données d'observations. Malheureusement, dans les données de puces à ADN, le nombre de gènes est très élevé (20.000 gènes) et excédant toujours celui des échantillons ($10 - 100$ échantillons).
- *deuxièmement*, comme toutes méthodes de recueil de données, le problème du bruit est inévitable.

Nous présentons dans la sous-section ci-après les méthodes de traitement de données de puces à ADN.

1.2.2.2 Traitement des données

Le traitement de ces données d'expression est nécessaire avant de les utiliser dans l'apprentissage. Les différents problèmes à considérer sont les suivants :

Normalisation

Les données d'expression souvent ne sont pas parfaites : valeurs manquantes, imprécises, non-homogènes. Le traitement du bruit dans les données n'est pas un problème facile, surtout dans le contexte de la pauvreté de données [1] ; il n'est pas facile de distinguer ce qui est le résultat d'une erreur ou d'une différence non significative d'une observation aléatoire. La motivation de la normalisation est de réduire ce bruit et de rendre comparable les données issues d'échantillons différents. Le résultat du traitement du bruit influe plus ou moins sur les résultats d'un modèle d'apprentissage. Donc les outils statistiques (traitement de bruit de fond, normalisation, etc.) sont apparues presque immédiatement après la naissance de la technologie de puces à ADN. Grâce à l'amélioration considérable des techniques de puces à ADN et des différentes techniques de traitement d'image et statistique proposées, nous disposons d'un grand nombre de méthodes éprouvées [Quackenbush 2002] (voir la Fig.1.8).

La correction de bruit repose sur différentes techniques (analyse de signal du voisinage du spot, usage de séquences de contrôle qui ne peuvent s'hydri-

1. le nombre de gènes est très élevé et excédant toujours celui des échantillons

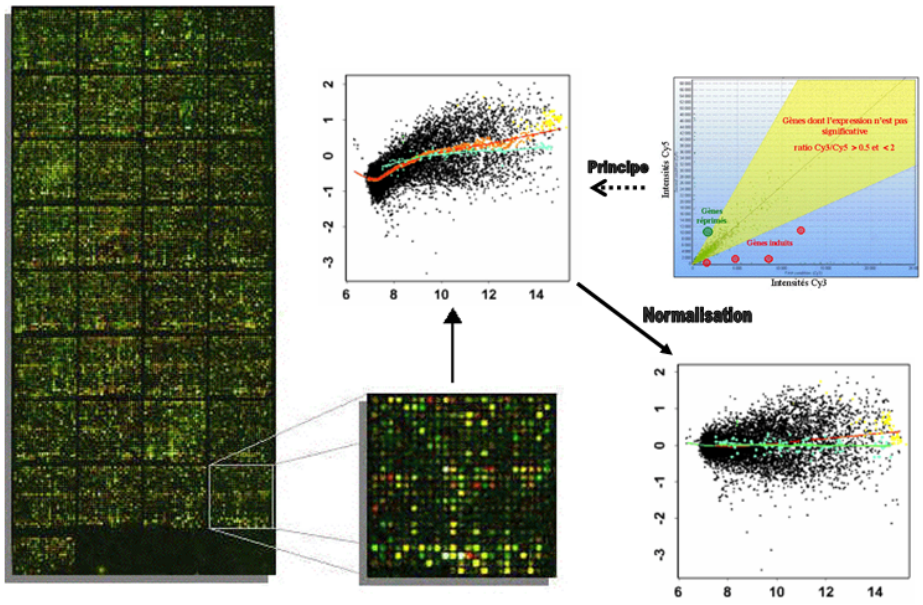


FIGURE 1.8 – Exemple de la normalisation de données de puces [Quackenbush 2002].

der avec les gènes de la puce,...). La normalisation proprement dite se base généralement sur soit un normalisation *LOWESS* (régression locale), soit une normalisation par *quantile*.

Réduction de dimensions

Les données de puces à ADN sont présentées par les matrices. Donc, la taille de données peut être mesurée selon deux dimensions, le nombre de variables (ligne) et le nombre d'exemples (colonne). Pour la plupart des bases de données disponibles, les données d'expression sont très dissymétriques, le nombre de gènes étant beaucoup plus élevé que celui des échantillons. Donc, il faudrait une réduction des dimensions pour sélectionner un sous-ensemble qui a deux objectifs : (1) réduire la complexité du calcul ; (2) choisir les gènes qui sont les plus pertinents.

Discrétisation

Certains algorithmes ne fonctionnent qu'avec des données catégorielles et surtout des données booléennes. Ainsi on devrait les transformer en variables discrètes par les méthodes de discrétisation. Les méthodes de discrétisation

sont nombreuses dans la littérature [Dougherty 1995]. Les modèles de régulation les plus souvent adoptés sont à 2 valeurs (0 : *normal* et 1 : *sur-exprimé*) ou à 3 valeurs (-1 : *sous-exprimé*, 0 : *normal* et 1 :*sur-exprimé*).

Il existe plusieurs types de discrétisation [Hartemink 2001b] :

- Discrétisation basée sur le quantile,
- Discrétisation basée sur l'intervalle,
- Discrétisation stochastique.

1.2.3 Méthodes de reconstruction de réseaux de régulation génétique

D'après [Schlitt 2007], il est possible d'établir une taxinomie des réseaux en fonction des critères suivants :

- *la description des entités biologiques* que l'on cherche à modéliser (par exemple, les facteurs de transcription, les sites de liaisons,...),
- *la topologie du réseau*. Ce critère désigne notamment le type d'interaction entre les entités biologiques,
- *la logique de contrôle*. Ce critère détermine comment se combinent les effets des noeuds sources sur un noeud cible du réseau,
- *le modèle dynamique*. Ce critère correspond à la modélisation temporelle du réseau et à la prédiction de la réponse des entités biologiques à certains stimuli.

1.2.3.1 Description des entités biologiques

La classe de réseaux qui nous intéresse se modélise sous la forme d'un graphe dont les noeuds sont des gènes ou des protéines et dont les arcs représentent une relation biologique.

1.2.3.2 Topologie du réseau

On trouve dans la littérature de nombreuses variétés de réseaux biologiques. Parmi ceux-ci, on peut citer :

- *les réseaux transcriptionnels* : les arcs sont dirigés ; le gène source est un facteur de transcription ; le gène cible est un gène activé par le gène source,

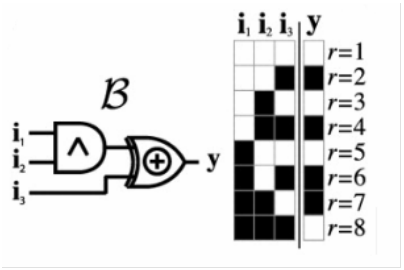


FIGURE 1.9 – Modélisation de la logique de contrôle par une fonction booléenne. La réponse de y est définie par une fonction logique. Figure extraite de [De Ridder 2010]

- *les réseaux d'interaction* : Les noeuds sont des protéines et les arcs non dirigés représentent des liaison entre protéines (par exemple : la transduction de signal),
- *les réseaux de mutation* : les noeuds sont des gènes et l'arc dirigé indique une mutation entre une séquence parente et une séquence fille,
- *les réseaux de co-citation* : un arc non dirigé est étiqueté selon la fréquence de colocation des deux gènes dans la littérature scientifique.

1.2.3.3 La logique de contrôle

Réseau transcriptionnel

La topologie du réseau permet d'appréhender les relations entre entités biologiques, mais elle n'indique pas le comportement d'un noeud cible en fonction de celui de ses noeuds sources. Concernant le réseau transcriptionnel, le modèle le plus simple compare la machinerie biologique à un circuit logique : le gène cible est activé selon une fonction booléenne, comme l'illustre la figure 1.9.

Ce modèle suppose une représentation des entités biologiques par des états discrets : par exemple, on peut associer une valeur booléenne au fait qu'un gène est exprimé. Une modélisation continue permet d'affiner le modèle.

Arbres de décision

Dans [Soinov 2003], une modélisation par arbres de décision a été proposée. Dans ce modèle, l'expression d'un gène est déterminée par les valeurs d'expression de certains gènes, comme le montre la figure 1.10.

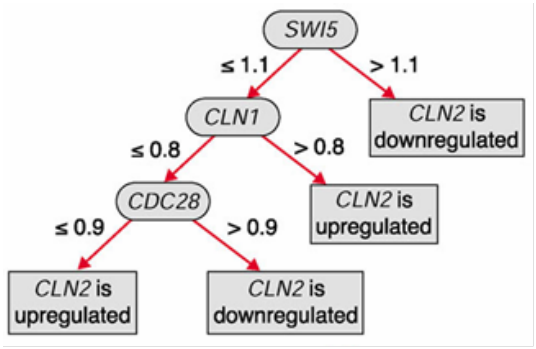


FIGURE 1.10 – Arbre de décision du gène CLN2 (d'après [Soinov 2003]). Dans cet exemple, la régulation du gène est prédite à partir des gènes SWI5, CLN1 et CDC28. Les étiquettes sur les arcs indiquent les seuils de valeur d'expression définis à partir de données de puces à ADN.

Réseaux de confiance

Les réseaux de confiance [Butte 2000, Carter 2004] calculent l'information mutuelle entre les niveaux d'expression pour chaque couple de gènes, puis génèrent un réseau de type d'interaction gène-gène. L'hypothèse est la suivante : *"une association avec une information mutuelle élevée signifie qu'un gène est non-aléatoirement associé à un autre ; dans ce cas les deux sont liés biologiquement"* ou *"deux gènes sont reliés si et seulement si ils sont co-exprimés"* (c'est la raison pour laquelle cette approche est aussi appelé "réseaux de co-expression").

Butte et Kohane [Butte 2000] ont développé une méthode basée sur l'information mutuelle. A partir d'un test de permutation, ils considèrent que deux gènes sont co-exprimés si et seulement si leur information mutuelle est supérieure à l'information mutuelle maximale obtenue dans les données permutées. Ils montrent que les composantes connexes du graphe obtenu, appelées *relevance networks*, contiennent des gènes (co-régulés) avec des fonctions biologiques proches. Pour calculer l'information mutuelle, Butte and Kohane ont discrétisé des valeurs de niveaux d'expressions. La Figure 1.11 montre un exemple de réseau obtenu avec cette méthode. Carter et al. [Carter 2004] se sont intéressés à la topologie de réseaux de co-expression des gènes. Ils utilisent la corrélation de Pearson pour mesurer la relation entre chaque paire de

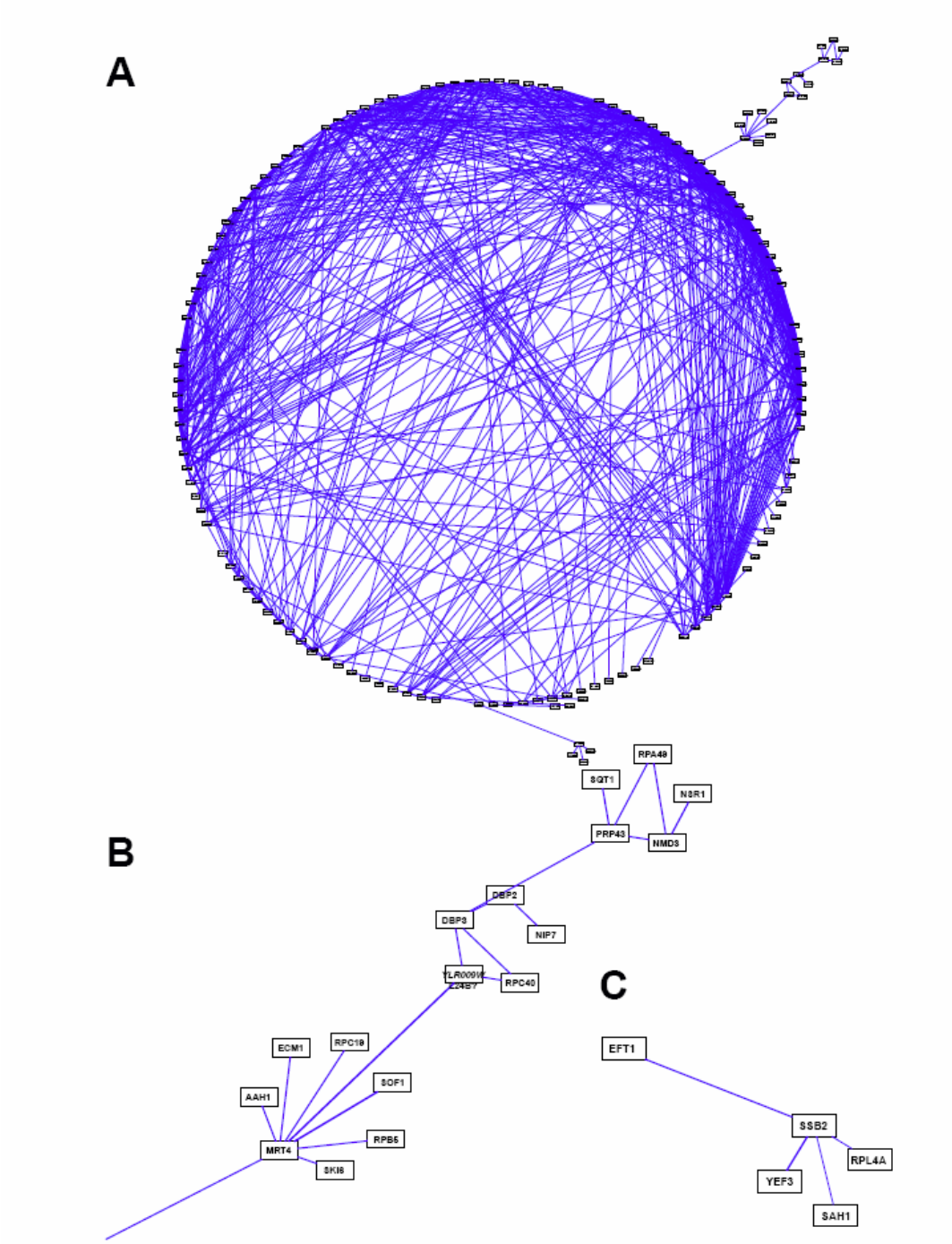


FIGURE 1.11 – Exemple d'un réseau obtenu par l'approche basée sur les réseaux de confiance. *(A) Le réseau le plus large possible. (B) et (C) Deux petites branches dans (A) en version ZOOM. [Butte 2000]*

gènes et éliminent les interactions ayant une faible significativité.

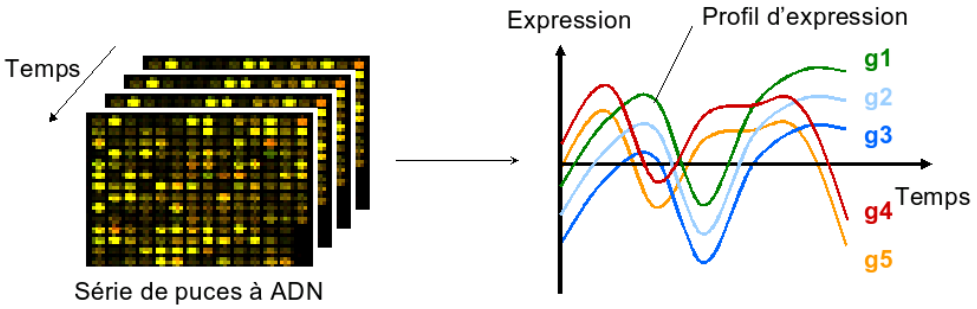


FIGURE 1.12 – Évolution des valeurs d'expression au cours du temps.

Cette méthode prend en compte l'association biologiques des gènes. Toutefois, la discrétisation entraine une perte d'information.

1.2.3.4 Le modèle statique/dynamique

Ce modèle décrit comment évolue le réseau dans le temps (voir la Figure 1.12). Il peut être absent de certains réseaux, purement statiques. Par exemple, l'arbre de décision (cf. Figure 1.10) ou les réseaux booléens (cf. Figure 1.14) définissent la régulation d'un gène en fonction de gènes informatifs, indépendamment de la notion de temps. De nombreux modèles dynamiques ont été proposés [Kauffman 1969, Ong 2002, Beal 2004, Kim 2004, Nachman 2004, Redestig 2007, Zou 2005].

Réseaux booléens

Stuart Kauffman [Kauffman 1969] a été l'un des premiers à modéliser la dynamique d'un réseau, en se basant sur le concept de réseau booléen. Dans ce modèle, on définit l'activité de chaque gène par une valeur binaire et on fait évoluer le réseau booléen par des pas de temps discret. Cette approche est synchrone : tous les gènes changent d'état simultanément (cf. Figure 1.12).

L'idée des réseaux booléens a été ensuite repris par [Akutsu 1999, Lahdesmaki 2003, Shmulevich 2002, Ribeiro 2008]. La modélisation des RRG est simple : on modélise un gène par une variable booléenne. Un gène est donc *exprimé*, ou *non-exprimé*. On représente ensuite les influences (régulations) *positives* (activation) ou *négatives* (inhibition) d'un gène sur les autres par des fonctions booléennes qui déterminent l'état d'un gène en fonction de

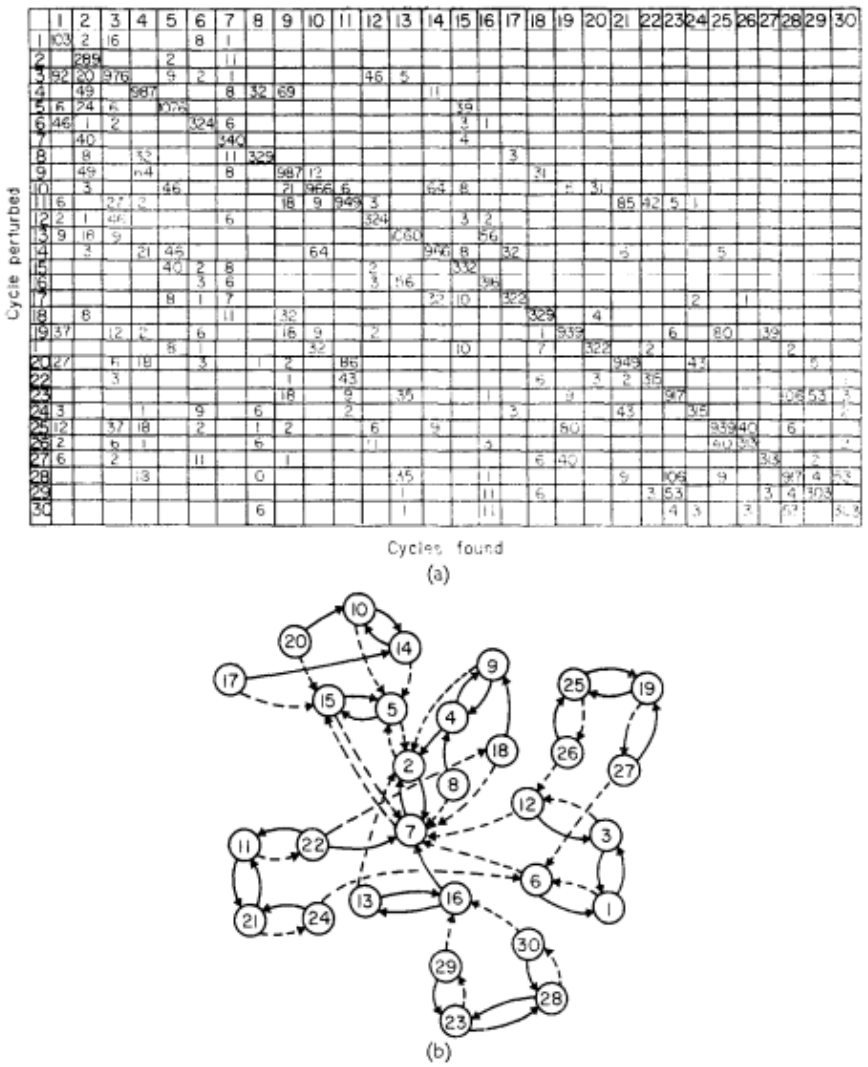


FIGURE 1.13 – Exemple d'un réseau dynamique pour représenter un RRG avec des données temptorelles [Kauffman 1969]. *(a) Matrice de transition de niveau de regulation dans 30 tranches temporelles. (b) Transition entre des tranches présentées dans (a).*

l'état des certains autres gènes.

Plus précisément, c'est un graphe orienté $G = (V, F)$ où $V = \{v_1, v_2, ..., v_n\}$ est un ensemble de noeuds et F est un ensemble de fonctions booléennes qui définit une topologie d'arêtes. n est appelé *taille* ou *dimension* du réseau. Un noeud représente un gène. A chaque noeud v, on associe une valeur booléenne $x(v)$ qui représente le niveau d'expression du gène correspondant. La valeur

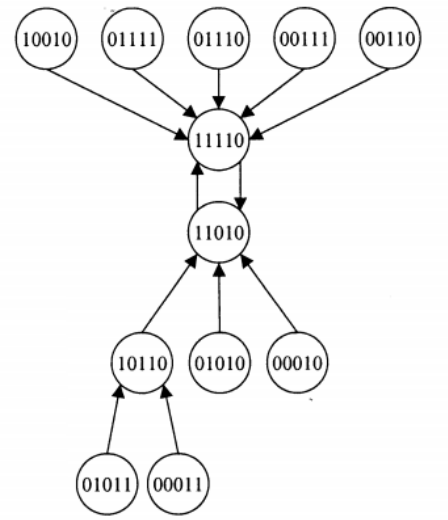


FIGURE 1.14 – Exemple d'un réseau booléen simple pour représenter un RRG [Shmulevich 2002].

de x sera 1 si le gène est *exprimé* et 0 sinon.

Cette méthode présente l'avantage d'être simple. Cependant, d'après [Lahdesmaki 2003] cette méthode accepte implicitement une hypothèse d'absence de bruit dans les mesures d'expression, hypothèse peu conforme à la réalité.

Réseaux de Pétri

Les réseaux de Pétri [Chaouiya 2004, Comet 2005, Matsuno 2000, Steggles 2007] sont des graphes bipartis utilisés pour la modélisation et le raisonnement sur les systèmes concurrentiels et distribués. Ils sont utilisés dans le contexte de RRG pour identifier leur structure et leur comportement dynamique.

Cette méthode prend en compte l'aspect dynamique des relations dans un RRG. Cependant, c'est une approche basée sur une grande part d'expertise [Comet 2005].

Equations différentielles

De manière générale, les approches basées sur les équations différentielles [Chan 2007, Hoon 2003] relient la valeur de chaque variable à la valeur de

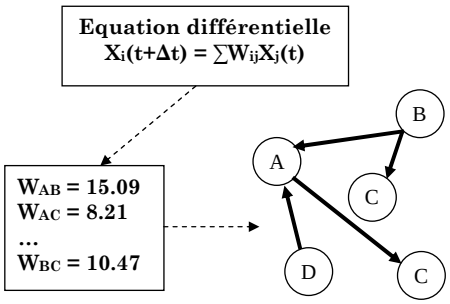
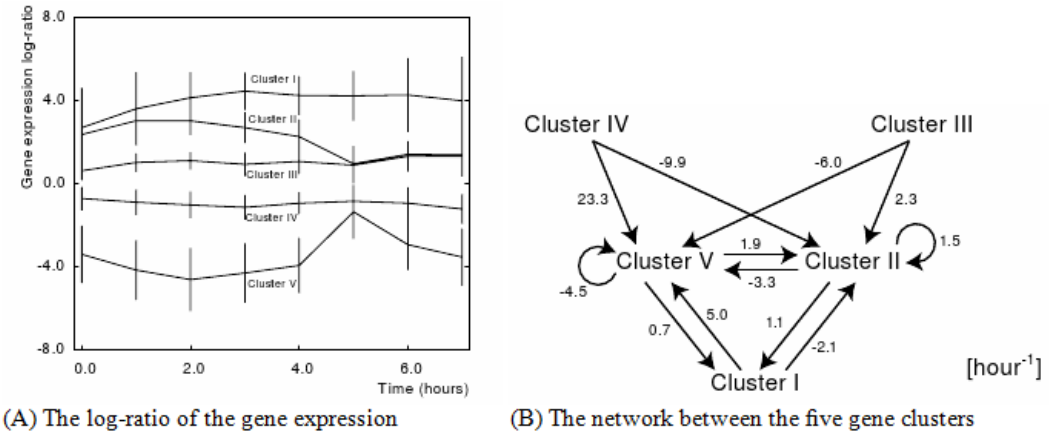


FIGURE 1.15 – Principe de la méthode basée sur les équations différentielles.

toutes les autres variables sous la forme d'une équation (voir la Figure 1.15). Ces équations peuvent être linéaires, non linéaires, et/ou des équations différentielles. Les interactions putatives sont identifiées par la résolution d'un ensemble d'équations incluant des poids comme paramètres. Ces pondérations représentent l'influence de chaque variable sur les autres. Généralement, seuls quelques poids dans l'équation pour une seule variable diffèrent considérablement de zéro et sont donc considérés comme des influenceurs putatifs.

Cette méthode considère un RRG comme un système d'équations différentielles. Le taux de changement d'une concentration particulière est fonction d'autres concentrations. Les concentrations de ces composants sont exprimées par des valeurs réelles positives évoluant dans le temps de manière continue. La variation de ces valeurs est décrite par une équation différentielle ayant comme paramètres les concentrations des molécules régulant l'entité étudiée. Ce type de modèle permet de prédire les valeurs de concentration des entités biologiques telles que les protéines, les molécules de signalisation, les ARN messagers.

L'avantage de cette approche est la précision du calcul. Pourtant, elle est limitée à un fonction simple à cause de la complexité des paramètres dans le cas où le nombre de variables est grand. Ce type de modélisation nécessite la connaissance précise des concentrations des composants moléculaires ainsi que de leur cinétique, information malheureusement souvent difficile à obtenir. En plus, cette méthode demande des données temporelles pour apprendre les



(A) The log-ratio of the gene expression (B) The network between the five gene clusters

FIGURE 1.16 – Exemple d'un réseau obtenu par l'approche basée sur les équations différentielles [Hoon 2003].

paramètres [Chan 2007] (cf. Figure 1.16).

Par ailleurs, cette modélisation ne tient pas compte des fluctuations aléatoires du système biologique qui peuvent induire des effets non négligeables sur les concentrations moléculaires. Des auteurs ont suggéré d'utiliser des modèles stochastiques [Paulevé 2011] pour modéliser la dynamique des réseaux. Ce type de modèle définit la probabilité qu'une variable atteint un état donné en fonction des états des molécules à un instant donné. Pour ce type de modélisation, les réseaux Bayésiens sont un des outils le plus répandus qui peuvent capturer les relations linaires, non-linaires, ou stochastiques...

Réseaux bayésiens

Les réseaux Bayésiens (RB) sont utilisés dans de nombreux domaines comme des outils de modélisation. En effet, ils sont un des représentants les plus connus des modèles graphiques probabilistes [Naïm 2007]. Ils rendent particulièrement intéressants la représentation de systèmes complexes. Pour la modélisation de réseaux de régulation génétique : *(i)* la partie graphique de RB donne un outil visuel qui indique non seulement les régulations potentielles entre les gènes, mais aussi l'orientation de ces régulations (arcs dirigés) ; *(ii)* la partie probabiliste permet de quantifier ces régulations. La Figure 3.8 présente un RB obtenu par la reconstruction du RRG présenté dans [Friedman 2004].

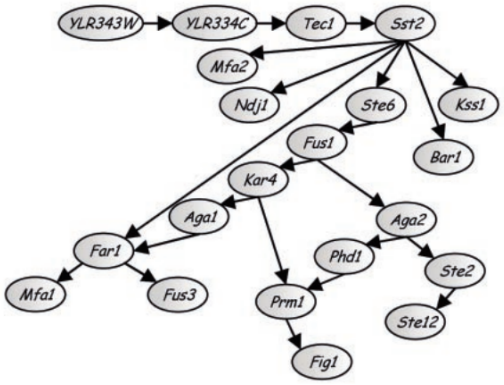


FIGURE 1.17 – Exemple d'un RB obtenu par la reconstruction du RRG présenté dans [Friedman 2004].

C'est une technique de modélisation de systèmes comportant un grand nombre de variables, au moyen de relations mesurées par la fonction de distribution des probabilités conditionnelles (paramètres) et présentées sur un graphe acyclique orienté (structure). L'apprentissage de RB à partir de données et de connaissances a priori offre un cadre théorique et méthodologique pour relier les problématiques de modélisation, d'identification, de simulation du fonctionnement, de l'interaction entre des gènes au sein d'un organisme. On peut citer quelques propositions de modélisation de réseaux de régulation génétique par RB depuis ces dernières années : [Friedman 2000, Pe'er 2001, Hartemink 2001a, Husmeier 2003, Friedman 2004, Auliac 2008].

La Table 1.2 nous permet de voir concrètement les avantages de réseaux bayésiens par rapport aux autres méthodes dans le contexte de données de petites tailles et incertaines. C'est un des contextes souvent rencontrés dans la modélisation de RRG à partir de données de puces à ADN.

1.3 Conclusion

Nous avons présenté dans ce chapitre différentes approches pour la modélisation de réseaux de régulation génétique. Quelles que soient les approches utilisées pour modéliser les réseaux de régulation génétique à partir de données de puces à ADN, la difficulté principale consiste à trouver la meilleure structure qui possède à la fois la *robustesse* (maximisation de précision de l'in-

Méthodes	Données petites tailles	Données incertaines	Visualisation
1. Réseau transcriptionnel	−	−	-
2. Arbres de décision	−	−	+
3. Réseaux de confiance	−	−	+
4. Réseaux booléens	−	−	+
5. Réseaux de Pétri	−	−	+
6. Equations différentielles	−	−	-
7. Réseaux bayésiens	+	+	+

(Données petites tailles) : si la méthode permet de remédier le problème de données petites tailles ;
(Données incertaines) : si la méthode permet de remédier le problème de données petites tailles ;
(Visualisation) : si l'interprétation graphique est facile à comprendre ;
(+) : plus favorable ; *(−)* : moins favorable

TABLE 1.2 – Vue d'ensemble des méthodes de modélisation de réseaux de régulation génétique

férence avec une bonne précision sur toute quantité/qualité de données) et la *simplicité* (minimisation de relations entre variables). Le réseau Bayésien est une méthode pertinente de modélisation de réseaux de régulation génétique.

Chapitre

2 Apprentissage et évaluation de réseaux bayésiens

Sommaire

Nous avons présenté dans le Chapitre 1 différentes approches pour la problématique de la reconstruction de réseaux de régulation génétique à partir de données. La nature incertaine des données nous a conduit à choisir les réseaux bayésiens pour résoudre cette problématique. Avant de mener à l'expérimentation, nous présentons dans ce chapitre le cadre théorique et méthodologique pour l'apprentissage de la structure de réseaux bayésiens à partir de données :

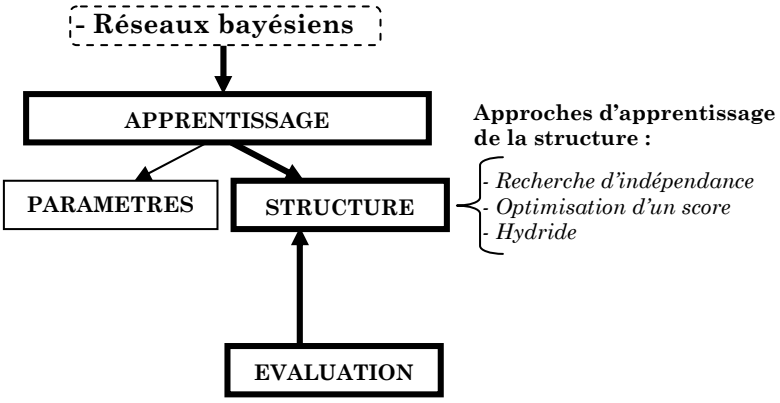


FIGURE 2.1 – L'organisation intuitive des notions-clés du chapitre 2. L'apprentissage de réseaux bayésiens consiste en deux phases : l'apprentissage des paramètres et l'apprentissage de la structure. Il y a trois types d'approches pour l'apprentissage de la structure : les méthodes basées sur la recherche d'indépendance, les méthodes basées sur l'optimisation d'un score et les méthodes hybrides.

- *le lien entre modèle d'indépendance et réseau bayésien qui est fondamental pour comprendre d'où vient et ce que fait un réseau bayésien,*
- *la notion d'équivalence de Markov qui pose un problème crucial et commun pour l'évaluation de la qualité d'un réseau bayésien*
- *l'apprentissage de réseaux bayésiens : apprentissage des paramètres et apprentissage de la structure.*
- *les méthodes d'évaluation d'un algorithme d'apprentissage de réseaux bayésiens*

La Figure 2.1 résume une organisation intuitive des notions-clés de ce chapitre.

2.1 Réseaux bayésiens

Nous définissons, dans cette section, une série de notions de base qui sont utilisées dans plusieurs parties de cette thèse.

2.1.1 Probabilité conditionnelle, théorème de Bayes et indépendance

Définition 14. ***(Probabilité conditionnelle)*** *[Dawid 1979]*
Etant donnés X et Y deux ensembles de variables aléatoires, sachant que la probabilité de Y est non nulle, la probabilité conditionnelle de X est définie par :

$$P(X|Y) = \frac{P(X,Y)}{P(Y)} \tag{2.1}$$

Theorème 1. ***(Théorème de Bayes) [Bayes 1763]***
Etant donnés X et Y deux ensembles de variables aléatoires, le théorème de Bayes permet de déterminer la probabilité conditionelle de X sachant Y à partir des probabilités de X, de Y et de Y sachant X, avec la formule suivante :

$$P(X|Y) = \frac{P(Y|X).P(X)}{P(Y)} \tag{2.2}$$

Définition 15. ***(Indépendance)*** *[Dawid 1979]*
Etant donnés X et Y deux ensembles de variables aléatoires, X et Y sont indépendantes, noté $<X \perp\!\!\!\perp_p Y>$, si et seulement si $P(X|Y) = P(Y)$.

Définition 16. ***(Indépendance conditionnelle)*** *[Dawid 1979]*
Etant donnés X, Y et Z trois ensembles de variables aléatoires, X est indépendante conditionnellement de Y sachant Z, noté $< X \perp\!\!\!\perp_p Y|Z >$ si seulement si $P(X|Y,Z) = P(X|Z)$.

2.1.2 Modèles d'indépendance

Définition 17. ***(Modèle d'indépendance)*** *- Dependency Model [Pearl 1985]*
Un modèle d'indépendance (MI) $\mathcal{M}$ sur un ensemble d'objets V est un sous-ensemble quelconque de triplets noté $I(X,Z,Y)$, aussi noté $< X \perp\!\!\!\perp Y|Z >_{\mathcal{M}}$, où X, Y, Z sont trois sous-ensembles de V. Les triplets dans $\mathcal{M}$ représentent des indépendances. $I(X,Z,Y) \in \mathcal{M}$ signifie que X et Y interagissent seulement via Z ou "X est indépendant de Y étant donné Z".

Définition 18. ***(semi-graphoïde)*** *[Pearl 1985]*
Un modèle d'indépendance $\mathcal{M}$ est un semi-graphoïde s'il satisfait les axiomes

d'indépendances conditionnelles suivants pour tous les (X, Y, Z, U) *sous-ensembles disjoints de* V *:*

Axiome	***Condition***		***Conclusion***
P0 *: Indépendance triviale :*	$< X \perp\!\!\!\perp Y \mid Z >_{\mathcal{M}}$	$\Rightarrow$	$< Y \perp\!\!\!\perp X \mid Z >_{\mathcal{M}}$
P1 *: Symétrie :*	$< X \perp\!\!\!\perp Y \mid Z >_{\mathcal{M}}$	$\Rightarrow$	$< Y \perp\!\!\!\perp X \mid Z >_{\mathcal{M}}$
P2 *: Décomposition :*	$< X \perp\!\!\!\perp (Y \cup U) \mid Z >_{\mathcal{M}}$	$\Rightarrow$	$< X \perp\!\!\!\perp Y \mid Z >_{\mathcal{M}}$
P3 *: Union faible :*	$< X \perp\!\!\!\perp (Y \cup U) \mid Z >_{\mathcal{M}}$	$\Rightarrow$	$< X \perp\!\!\!\perp Y \mid (Z \cup U) >_{\mathcal{M}}$
P4 *: Contraction :*	$< X \perp\!\!\!\perp Y \mid (Z \cup U) >_{\mathcal{M}}$ *et* $< X \perp\!\!\!\perp U \mid Z >_{\mathcal{M}}$	$\Rightarrow$	$< X \perp\!\!\!\perp (Y \cup U) \mid Z >_{\mathcal{M}}$

Définition 19. ***(Graphoide)*** *[Pearl 1985]*
Un modèle d'indépendance $\mathcal{M}$ *est un graphoïde s'il est un semi-graphoïde et s'il satisfait l'axiome "intersection" :*

$$\textbf{\textit{P5}} \text{ : Intersection : } \quad \begin{array}{l} < X \perp\!\!\!\perp Y) \mid (Z \cup U) >_{\mathcal{M}} \\ et < X \perp\!\!\!\perp U \mid (Z \cup Y) >_{\mathcal{M}} \end{array} \quad \Rightarrow \quad < X \perp\!\!\!\perp (Y \cup U) \mid Z >_{\mathcal{M}}$$

Remarque 1. *Les axiomes de* ***P0*** *à* ***P5*** *ont été premièrement introduits par [Dawid 1979, Spohn 1980, Pearl 1985]. Il y a bien d'autres axiomes d'indépendances conditionnelles importantes tel que "union forte", "transition forte", "transition faible", "chordalité" qui sont soigneusement présentés dans [Lucas 2007].*

2.1.2.1 Modèles d'indépendance et Modèle probabiliste

Définition 20. ***(Modèle probabiliste)*** *- Probabilistic Model [Pearl 1985]*
Un modèle probabiliste $\mathcal{M}_P$ *est défini par la distribution de probabilité* P *sur un ensemble de variables* V*.*

Theorème 2. ***(Pearl et Paz, 1985)*** *[Pearl 1985]*
Le modèle probabiliste $\mathcal{M}_P$ *satisfait tous les axiomes d'un semi-graphoïde. Si* P *est strictement positive alors* $\mathcal{M}_P$ *satisfait tous les axiomes d'un graphoïde.*

Remarque 2. *D'après le Théorème 2, le modèle d'indépendance issu d'une loi de probabilité a une structure de semi-graphoïde. Pourtant la question "Un semi-graphoïde représente-t-il nécessairement une loi de probabilité ?" est encore discutée autour de la conjecture "Completeness Conjecture" [Pearl 1985] qui donne encore lieu à de nombreux travaux [Kramosil 1988, Studený 1989, An 1992, Spohn 1994, Galles 1996, More 2010, More 2011].*

2.1.3 Réseaux bayésiens

Définition 21. ***(Condition de Markov)*** *[Pearl 1985]*
La condition de Markov peut être énoncée comme ceci : Chaque variable X_i est conditionnellement indépendante de l'ensemble de ses non-descendants, $NonDescendant(X_i)$, connaissant l'état de ses parents, $Pa(X_i)$, soit $P(X_i|Pa(X_i), NonDesc(X_i)) = P(X_i|Pa(X_i))$.

Définition 22. ***(Réseaux bayésiens)*** *[Pearl 1985]*
Les réseaux bayésiens sont des modèles graphiques probabilistes. Un réseau bayésien $\mathcal{B} = (G, \theta)$ est défini par :

- ***une structure*** *$G = (V, E)$ qui est un graphe orienté sans circuit (DAG : Directed Acyclic Graph) où V est l'ensemble des noeuds qui représentent un ensemble de variables aléatoires $X = (X_1, ..., X_n)$* [1] *et E est l'ensemble des arcs,*
- *et* ***des paramètres*** *$\theta = [P(X_i|Pa(X_i)]$ qui sont des distributions de probabilités pour que $\mathcal{B}$ vérifie la condition de Markov.*

De manière plus générale, les réseaux bayésiens (RB) sont des modèles graphiques pour représenter les relations probabilistes parmi un ensemble de variables aléatoires. Les RB offrent une représentation graphique de manière compacte des lois de probabilité jointes entre variables. La distribution de probabilités sur l'ensemble des variables est définie par :

$$P(X) = P(X_1, X_2, ..., X_n) = \prod_{i=1}^{n} P(X_i|Pa(X_i)) \tag{2.3}$$

Définition 23. ***(Chaîne bloquée)*** *[Geiger 1990]*
Soit $\mathcal{P}$ une chaîne du noeud X_1 au noeud X_2, $\mathcal{P}$ est dite bloquée par un ensemble de noeuds Z si et seulement si au moins une des conditions suivantes est satisfaite :

1. Nous nous plaçons pour ce travail dans le contexte de suffisance causale où il n'existe aucune variable latente parente de variables observées, sauf si cette variable latente est constante.

- $\mathcal{P}$ *est une chaîne linéaire :* $X_1 \rightarrow m \rightarrow X_2$ *ou* $X_1 \leftarrow m \leftarrow X_2$, *où* $m \in Z$.
- $\mathcal{P}$ *est une chaîne divergente :* $X_1 \leftarrow m \rightarrow X_2$, *où* $m \in Z$.
- $\mathcal{P}$ *est une chaîne convergente :* $X_1 \rightarrow m \leftarrow X_2$, *où* $m \notin Z$ *et* $Descendant(m) \notin Z$.

Définition 24. ***(d-séparation)*** - *Directed separation [Geiger 1990]*
Soit X, Y, Z *trois ensembles de variables aléatoires,* X *et* Y *sont* d-séparés *par* Z *si* Z *bloque tous les chemins entre* X *et* Y, *noté* $<X \perp\!\!\!\perp Y|Z>$.

Si une distribution de probabilités représente numériquement les indépendances conditionnelles d'un modèle d'indépendance, le graphe représente intuitivement ces relations. Les définitions suivantes (*D-map, I-map, P-map*) présentent le lien entre un graphe et un modèle d'indépendance.

Définition 25. ***(D-map)*** - *Dependency map [Pearl 1985]*
Soit un graphe $G = (V, E)$, G *est une D-map d'un modèle d'indépendance* M *si toute la séparation dans* G *est une indication d'une indépendance dans* M :

$$< X \perp\!\!\!\perp Y|Z >_{\mathcal{M}} \Rightarrow < X \perp\!\!\!\perp Y|Z >_G \tag{2.4}$$

Définition 26. ***(I-map)*** - *Independency map [Pearl 1985]*
Soit un graphe $G = (V, E)$, G *est une I-map d'un modèle d'indépendance* M *si toute indépendance dans* M *est une indication d'une séparation dans* G :

$$< X \perp\!\!\!\perp Y|Z >_{\mathcal{M}} \Leftarrow < X \perp\!\!\!\perp Y|Z >_G \tag{2.5}$$

Définition 27. ***(P-map)*** - *Perfect map [Pearl 1985]*
Soit un graphe $G = (V, E)$, G *est une P-map d'un modèle d'indépendance* M *si* G *est à la fois D-map et I-map de* M :

$$< X \perp\!\!\!\perp Y|Z >_{\mathcal{M}} \Leftrightarrow < X \perp\!\!\!\perp Y|Z >_G \tag{2.6}$$

2.1.4 Equivalence de Markov

Définition 28. ***(Equivalence de Markov)*** *[Verma 1991]*
Deux graphes sont dit équivalents s'ils ont le même modèle d'indépendance.

Définition 29. ***(Squelette)*** *[Verma 1991]*
Soit un graphe $G = (V, E)$, un graphe orienté sans circuit. Le squelette est un graphe non-orienté (ou non dirigé), obtenu en ignorant l'orientation de chaque arc de G.

Définition 30. ***(V-structure)*** *[Verma 1991]*
Soit $G = (V, E)$, un graphe orienté sans circuit. Une v-structure dans G est de la forme $X_1 \rightarrow X_3 \leftarrow X_2$ tel que X_1 et X_2 non-adjacents dans G.

Définition 31. ***(Arcs réversibles et non-réversibles)*** *[Castillo 1997]*
Soit $G = (V, E)$ un DAG, l'arc $E_{ij} = X_i \rightarrow X_i$ est un arc non-réversible si et seulement si $E_{ij} \in E'$ pour tous les DAG $G' = (V, E')$ équivalents à G, sinon c'est un arc réversible.

Définition 32. ***(Arcs inférés)***
Les arcs inférés sont des arcs non-réversibles qui n'appartiennent pas à une V-structure.

Theorème 3. ***(Verma and Pearl, 1991) [Verma 1991]***
Deux DAG sont équivalents si et seulement s'ils ont le même squelette et les mêmes v-structures.

Définition 33. ***(Classe d'équivalence de Markov)*** *[Chickering 2002b]*
La classe d'équivalence de Markov est l'ensemble de tous les graphes équivalents.

Définition 34. ***(Représentant de la classe d'équivalence de Markov)*** *[Chickering 2002b]*
Le représentant d'une classe d'équivalence de Markov possède le même squelette, *les* mêmes V-structures *et les mêmes arcs inférés que tous les graphes de la classe d'équivalence. Ce graphe est aussi appelé* pattern *par [Spirtes 1995],*

Algorithme 2.1 $DAG2EG(G)$ [Chickering 2002b]

ENTRÉES: Un DAG, G.
SORTIES: Le graphe essentiel EG.

1: $EG \leftarrow G$;
2: Ordonner les arcs de EG ;
3: $\forall arc, tiquette(arc) \leftarrow \emptyset$;
4: $\mathcal{A} \leftarrow$ liste des arcs non étiquetés ;
5: **Répéter**
6: $(X_i, X_j) \leftarrow min_A(arc)$ (plus petit arc non étiqueté) ;
7: $\forall X_k / tiquette(X_k, X_i) \leftarrow$ Non-réversible ;
8: $Fin \leftarrow Faux$;
9: **si** $X_k \notin pa(X_j)$ **alors**
10: $tiquette(x, X_j) \leftarrow$Non-réversible ;
11: $\mathcal{A} \leftarrow \mathcal{A} \setminus (x, X_j)$;
12: $Fin \leftarrow Faux$;
13: **sinon**
14: $tiquette(X_k, X_j) \leftarrow$ Non-réversible ;
15: $\mathcal{A} \leftarrow \mathcal{A} \setminus (X_k, X_j)$;
16: **finsi**
17: **si** $Fin = Faux$ **alors**
18: **si** $\exists arc(X_k, X_j) / X_k \notin pa(X_i) \cup X_1 i$ **alors**
19: **pour tout** $\forall (X_k, X_j) \in \mathcal{A}$, **faire**
20: $tiquette(X_k, X_j) \leftarrow$ Non-réversible ;
21: $\mathcal{A} \leftarrow \mathcal{A} \setminus (X_k, X_j)$;
22: **fin pour**
23: **sinon**
24: **pour tout** $\forall (X_k, X_j) \in A$, **faire**
25: $tiquette(X_k, X_j) \leftarrow$ Réversible ;
26: $\mathcal{A} \leftarrow \mathcal{A} \setminus (X_k, X_j)$;
27: **fin pour**
28: **finsi**
29: **finsi**
30: **Jusqu'à** $\mathcal{A} = \emptyset$
31: **retourner** EG

TABLE 2.1 – Algorithme transformant un DAG en EG

graphe essentiel - ***EG*** *(essential graph) par [Andersson 1995] ou* graphe orienté maximal *(maximally oriented graphs) par Meek [Meek 1995a]. Dans un EG, les arcs réversibles sont remplacés par des arêtes (non orientées) et les arcs non-réversibles qui restent orientés.*

L'Algorithme 2.1 proposé par [Chickering 2002b] permet d'obtenir d'un EG à partir d'un DAG donné. L'Algorithme 2.2 proposé par [Dor 1992] permet d'obtenir un DAG à partir d'un EG.

Algorithme 2.2 *EG2DAG(EG)* [Dor 1992]

Entrées: Un graphe essentiel EG.
Sorties: Un DAG G.

1: $G \leftarrow EG$;
2: $\mathcal{A} \leftarrow$ liste des arêtes de EG;
3: **Répéter**
4: Recherche d'un noeud X_i tel que;
5: il n'existe aucun arc $X_i \leftarrow X_j$ dans A
6: et pour tout X_j tel qu'il existe $X_i - X_j$ dans A,
7: (où X_j est adjacent à tous les autres noeuds adjacents à X_i),
8: **si** X_i n'existe pas **alors**
9: EG n'admet aucune extension complètement dirigée;
10: **sinon**
11: $\forall X_j$ tel que $X_i - X_j \in A$
12: $X_i \rightarrow X_j$ dans G
13: $\mathcal{A} \leftarrow \mathcal{A} \setminus (X_i, X_j)$;
14: **finsi**
15: **Jusqu'à** $\mathcal{A} = \emptyset$
16: **retourner** G

TABLE 2.2 – Algorithme transformant un EG en DAG

2.2 Apprentissage de réseaux bayésiens

2.2.1 Contexte et problématiques

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

1. Première 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.

Les méthodes d'apprentissage de réseaux bayésiens à partir de données font généralement l'hypothèse de *fidélité*.

Définition 35. ***(Fidélité)*** *[Meek 1995b]*
Un réseau bayésien $\mathcal{B} = (G, \theta)$ est dit fidèle à la distribution de probabilité P si et seulement si G est P-map du modèle d'indépendance associé à P.

2.2.2 Apprentissage des paramètres de réseaux bayésiens

2.2.2.1 Principes

Etant donnée une structure de RB G et un ensemble de données $\mathcal{D}$, le principe de l'algorithme d'apprentissage des paramètres est d'estimer les distribu-

tions de probabilités (paramètres) $\theta = \{\theta_i\}$ avec $\theta_i = P(X_i|Pa(X_i))$, table de probabilité conditionnelle constituée de $\theta_{ijk} = P(X_i = x_k|Pa(X_i = x_j))$. Cette estimation peut être effectuée différemment selon que les données disponibles soient complètes ou incomplètes.

2.2.2.2 Apprentissage avec des données complètes

Lorsque toutes les variables sont observées, la méthode simple et souvent utilisée est l'apprentissage statistique. Elle consiste à estimer la probabilité d'un événement par la fréquence d'apparition dans la base de données. Elle est aussi appelée *maximum de vraisemblance* (likelihood) :

$$\theta_{i,j,k}^{MV} = \frac{N_{i,j,k}}{\Sigma_k N_{i,j,k}} \tag{2.7}$$

L'estimation bayésienne consiste à trouver les paramètres les plus probables sachant que les données ont été observées, en utilisant des a priori sur les paramètres. Cette méthode est aussi appelée *le maximum a posteriori* (MAP) :

$$\theta_{i,j,k}^{MAP} = \frac{N_{i,j,k} + \alpha_{i,j,k} - 1}{\Sigma_k N_{i,j,k} + \alpha_{i,j,k} - 1} \tag{2.8}$$

où $\alpha_{i,j,k}$ sont les paramètres de la distribution de Dirichlet de la loi $P(\theta)$.

Une autre version de l'approche bayésienne est *l'espérance a posteriori* (EAP) :

$$\theta_{i,j,k}^{EAP} = \frac{N_{i,j,k} + \alpha_{i,j,k}}{\Sigma_k N_{i,j,k} + \alpha_{i,j,k}} \tag{2.9}$$

2.2.2.3 Apprentissage avec des données incomplètes

Lorsque les variables sont partiellement observées, la méthode d'estimation la plus utilisée est celle qui se base sur l'algorithme EM (Expectation Maximisation) [Dempster 1977]. Soit X un ensemble de variables observées et Z un ensemble de variables manquantes. L'espérance de la log-vraisemblance est définie par :

$$Q(\theta : \theta') = E_{Z|X,\theta'}[logP(\mathcal{D}|\theta)] = E_{Z|X,\theta'} logP(X, Z|\theta) \tag{2.10}$$

où $logP(\mathcal{D}|\theta) = logP(X, Z|\theta)$ est la log-vraisemblance des données et θ' sont les paramètres actuels.

Soit $\theta^{(t)}$ les paramètres à l'itération t. L'algorithme EM consiste en deux étapes :

- *expectation* : estimer les $Q(\theta : \theta')$ en utilisant X et les paramètres actuelles $\theta^{(t)}$,
- puis, *maximisation* : choisir la meilleure valeur des paramètres $\theta^{(t+1)}$ en maximisant $Q(\theta : \theta')$:

$$\theta^{(t+1)} = argmax_\theta Q(\theta : \theta') \tag{2.11}$$

- et, répéter ces deux étapes tant que l'on arrive à augmenter la valeur de $Q(\theta : \theta')$

2.2.3 Apprentissage de la structure

2.2.3.1 Principe

L'apprentissage de la structure d'un RB permet de déterminer automatiquement une structure optimale d'un RB à partir de l'information contenue dans les données observées.

On distingue trois grandes familles d'approches d'apprentissage de la structure :

1. *Les méthodes basées sur les contraintes et recherche d'indépendances (constraint-based)*
2. *Les méthodes basées sur le score et la recherche (score and search based)*
3. *Les méthodes hybrides*

Quelle que soit l'approche choisie, l'apprentissage automatique de la structure de RB à partir de données pose quelques problèmes communs :

1. la recherche de la structure optimale reste un problème NP-difficile [Chickering 1996]. Ces méthodes ne sont pas si simples, principalement à

cause de la taille super-exponentielle de l'espace de recherche en fonction du nombre de variables.

2. l'apprentissage de la structure est grande consommatrice de données. C'est-à-dire qu'il faut des bases de taille importante afin de pouvoir obtenir un modèle de qualité. Cette limite devient un problème crucial dans les domaines expérimentaux où le recueil de données est difficile, cher, etc.

Donc, un véritable défi est alors d'abord d'optimiser la recherche (soit transposer le problème en un espace plus petit, soit utiliser une stratégie de recherche optimisée), en travaillant sur un jeu de données d'apprentissage réduit.

2.2.3.2 Méthodes basées sur la recherche d'indépendance

Le but de ces méthodes est de trouver des indépendances conditionnelles avec l'aide de tests d'indépendance (souvent le test du χ^2), puis de construire le graphe à partir de ces connaissances.

Les méthodes telles que IC (Inductive Causality) [Pearl 2000], PC [Spirtes 2001], SGS (Sprites, Glymour and Scheines) [Spirtes 1990]) sont des exemples bien connus pour ce type de méthodes. Ces méthodes cherchent d'abord à identifier un graphe non orienté qui représente les différentes indépendances conditionnelles existantes entre les variables observées à l'aide du test d'indépendance conditionnelle. L'orientation est réalisée à l'aide de la détection des V-structures (en utilisant les résultats des tests d'indépendance conditionnelle) et puis la propagation des orientations de certains arcs (arcs inférés à cause des V-structures).

Certaines autres méthodes permettent de détecter des variables latentes lors de la phase d'orientation, telles que IC* [Pearl 2000], CI (Conditional Independence) [Cheng 2002], FCI (Fast Causal Inference) [Spirtes 2001]. Ces méthodes permettent de relaxer l'hypothèse de suffisance causale.

2.2.3.3 Méthodes basées sur l'optimisation d'un score

Chaque RB $\mathcal{B} = (G, \theta)$ est noté par un score $S(\mathcal{G}, D)$ qui désigne sa capacité à représenter les données D. Ce score doit prendre en compte deux propriétés :

- *décomposable* localement.

$$S(\mathcal{G}, D) = \Sigma_{X_i \in X} s(X_i, Pa(X_i)) \tag{2.12}$$

- *équivalent*, c'est-à-dire qu'il faut assurer que les réseaux équivalents (cf. Définition 28) ont le même score.

La plupart des scores proposés dans la littérature (cf. Annexe 2) appliquent le principe de parcimonie du rasoir d'Occam : "*la simplicité est la meilleure*", c'est-à-dire, trouver le modèle qui correspond le mieux aux données et le plus simple possible. Citons par exemple les scores basés sur les principes de sélection de modèles comme **AIC** (Akaike's Information Criterion) [Akaike 1970], **BIC** (Bayesian Information Criterion) [Schwarz 1978] et les scores bayésiens BDe (Bayesian Dirichlet equivalent) [Heckerman 1995]. Ces scores les plus utilisés sont décomposables et équivalents.

Soit $Dim(B)$ le nombre de paramètres nécessaires pour décrire toutes les distributions de probabilité du réseau $\mathcal{B}$:

$$Dim(B) = \sum_{i=1}^{n} (r_i - 1) q_i \tag{2.13}$$

et $LL(\mathcal{D}|\theta, G)$ la log-vraisemblance du réseau $\mathcal{B} = (G, \theta)$:

$$LL(\mathcal{D}|G, \theta) = logP(D|G, \theta) \tag{2.14}$$

Le score AIC est défini par :

$$AIC(G, \mathcal{D}) = LL(\mathcal{D}|G, \theta^{MV}) - Dim(G) \tag{2.15}$$

Le score BIC est défini par :

$$BIC(G, \mathcal{D}) = LL(\mathcal{D}|G, \theta^{MV}) - \frac{1}{2} Dim(G) log(N) \tag{2.16}$$

Le score BDe est défini par :

$$BDe(G,\mathcal{D}) = P(G)\prod_{i=1}^{n}\prod_{j=1}^{q_i}\frac{\Gamma(\alpha_{ij})}{\Gamma(N_{ij}+\alpha_{ij})}\prod_{k=1}^{r_i}\frac{\Gamma(N_{ijk}+\alpha_{ijk})}{\Gamma(\alpha_{ijk})} \tag{2.17}$$

$$\alpha_{ij} = \sum_{k=1}^{r_i}\alpha_{ijk} \tag{2.18}$$

$$\alpha_{ijk} = N' \times P(X_i = x_k, Pa(X_i) = x_j | G_c) \tag{2.19}$$

G_c est le graphe complètement connecté et N' est un nombre d'exemples "équivalent" défini par l'utilisateur.

Le principe des méthodes à base de score consiste à parcourir de manière heuristique l'espace des DAG ou celui des EG :

1. en cherchant dans l'espace des arbres (MWST - *Maximum Weighted Spanning Tree* [Chow 1968]) ;
2. en cherchant dans l'espace des réseaux bayésiens avec un contrainte sur l'ordre des noeuds (K2 [Herskovits 1991]) ;
3. en faisant une recherche gloutonne dans l'espace des DAG (GS - *Greedy Search* [Chickering 1995]) ou dans l'espace des EG (GES - *Greedy Equivalence Search* [Chickering 2002a]).
4. en optimisant la recherche par des métaheuristiques par exemple avec des approches évolutionnaires comme les algorithmes génétiques [Pelikan 2000, Pelikan 2001, Blanco 2003, Sun 2005], ou avec PSO - *Particle Swarm Optimization* [Cowie 2007] pour l'apprentissage de la structure de RB.

2.2.3.4 Méthodes hybrides

Le but de ces méthodes est de combiner les avantages des méthodes basées sur les contraintes et celles basées sur les scores. Inspirées par le principe *"diviser pour régner"*, les méthodes hybrides consistent en deux étapes :

- une recherche locale, qui permet d'obtenir un voisinage contenant toutes les dépendances (indépendances) locales intéressantes avec l'aide de tests d'indépendances. Citons par exemple MMMB [Tsamardinos 2006], MMPC [Tsamardinos 2006], PCMB [Peña 2007], MBOR [Rodrigues De Morais 2008].
- une optimisation globale, qui permet de faire une recherche sur l'espace des DAG en se restreignant aux dépendences locales trouvées précédemment. Citons par exemple MMHC [Tsamardinos 2006] qui combine MMPC et la recherche gloutonne.

Remarque 3. *Comme pour l'apprentissage des paramètres, l'apprentissage de la structure est aussi confronté aux problèmes de données complètes ou incomplètes. Comme ce n'est pas l'objectif de notre travail, nous ne présentons pas le cas des données incomplètes, pour lequel le lecteur peut par exemple consulter [François 2006].*

2.3 Evaluation d'un algorithme d'apprentissage de la structure

Dans les sections précédentes, nous avons présenté les méthodes d'apprentissage de réseaux bayésiens, dont les méthodes d'apprentissage de la structure. Dans cette section nous présentons les méthodes pour évaluer la qualité du réseau bayésien obtenu par cet apprentissage de la structure.

Il y a deux scénarios pour l'évaluation d'un algorithme d'apprentissage de la structure :

- Etant donnés un RB théorique $\mathcal{B}_0 = (G_0, \theta_0)$ et des données D générées à partir de ce RB, les mesures évaluent la qualité de l'algorithme en comparant la qualité du graphe appris $\mathcal{B} = (G, \theta)$ et celle du réseau théorique $\mathcal{B}_0$
- Il n'y a pas de RB $\mathcal{B}_0$. Il faut comparer la qualité du graphe appris par rapport au résultat d'un autre algorithme dit "standard".

Dans ces deux scénarios, il y a besoin de méthodes pour mesurer la qualité d'un réseau bayésien. Nous allons passer en revue différentes méthodes en

résumant les avantages et les inconvénients de chaque approche.

Cette évaluation peut être biaisée par le fait que les algorithmes d'apprentissage produisent tous une structure optimale, à une classe d'équivalence près (cf. Définition 28). Toutes les structures de cette classe d'équivalence représentent le même modèle d'indépendance, ont le même score, etc...Il faudrait donc prendre cette caractéristique en compte dans l'évaluation, quel que soit le scénario choisi.

Nous allons maintenant passer en revue les méthodes d'évaluation les plus couramment utilisées en vérifiant entre autres si le point précédent est bien pris en compte.

2.3.1 Méthode basée sur le score

2.3.1.1 Principe

Comme précédemment présenté dans la sous-section 2.2.3.3, le score évalue la qualité d'un réseau bayésien. Pour évaluer un algorithme d'apprentissage de la structure, il suffit d'évaluer le score du graphe appris. L'algorithme d'apprentissage est bon si $S(\mathcal{B}, D) \simeq S(\mathcal{G}_0, D)$ où S est un score équivalent.

2.3.1.2 Avantages et limitations

Cette approche prend bien en compte équivalence de Markov lors de l'évaluation. Si un algorithme retourne un graphe G équivalent à $\mathcal{G}_0$, les deux structures ont le même score. Cet avantage est aussi un inconvénient, puisque s'il y a peu de données, il est possible d'obtenir un graphe G de même score que $\mathcal{G}_0$, mais qui ne soit pas dans la même classe d'équivalence.

2.3.2 Méthode basée sur la divergence Kullback-Leibler

2.3.2.1 Principe

La divergence de Kullback-Leibler est une mesure de dissimilarité entre deux distributions de probabilités [Kullback 1951]. Elle est souvent considérée comme une distance, pourtant elle n'en remplit pas tous les axiomes : elle n'est pas symétrique et ne respecte pas l'inégalité triangulaire. Toutefois, il existe également une version symétrique de la divergence Kullback-Leibler (voir l'équation 2.20).

Dans le contexte des RB, la divergence Kullback-Leibler est utilisée pour mesurer la différence entre la distribution de probabilité du réseau appris, $P_{\mathcal{B}}$ et celle du réseau "cible", $P_{\mathcal{B}_0}$.

Si les variables sont discrètes, cette divergence est définie par :

$$D_{KL}(\mathcal{B}, \mathcal{B}_0) = \sum_{x \in \chi} P_{\mathcal{B}}(x) log \frac{P_{\mathcal{B}}(x)}{P_{\mathcal{B}_0}(x)} \tag{2.20}$$

Cette formule est utilisée avec une convention où : *0 log 0 = 0*. La valeur de KL est non-négative et est égale à *zéro* lorsque les lois $P_{\mathcal{B}}$ et $P_{\mathcal{B}_0}$ sont identiques.

Remarques :

Une version symétrique de la divergence KL entre deux distributions P_G et P_{G_0} est définie par :

$$D^s_{KL}(G, G_0) = [D_{KL}(G, G_0) + D_{KL}(G_0, G)]/2 \tag{2.21}$$

2.3.2.2 Avantages et limitations

Comme la méthode basée sur le score, cette approche prend bien en compte l'équivalence de Markov lors de l'évaluation. Cependant, il y a des contextes où la divergence KL souffre de limitations. En effet, la complexité des calculs est exponentielle par rapport au nombre de variables. Dans ce cas, le principe d'échantillonnage peut être utilisé pour réduire cette complexité.

2.3.3 Méthode basée sur la sensibilité/la spécificité

2.3.3.1 Notions

La sensibilité est la proportion de bons arcs découverts. A *contrario*, **la spécificité** est la proportion de mauvais arcs découverts par erreur.

2.3.3.2 Méthode

Etant donné le graphe du réseau théorique $G_0 = (V, E_0)$ et le graphe du réseau appris $G = (V, E)$, la méthode basée sur la sensibilité/la spécificité commence par calculer les indices suivants :

- TP (*true positive*) = nombre d'arcs présents à la fois dans G et G_0 ;
- TN (*true negative*) = nombre d'arcs absents à la fois dans G et G_0 ;

– FP (*false positive*) = nombre d'arcs présents dans G mais pas dans G_0 ;

– FN (*false negative*) = nombre d'arcs absents dans G mais pas dans G_0 ;

Ensuite, la sensibilité et la spécificité peuvent être calculées comme suit :

– *Sensibilité*, également appelée *précision* ou TPR (True Positive Rate)

$$TPR = sensibilité = \frac{TP}{TP + FN} \tag{2.22}$$

– *Spécificité* également appelée *rappel* ou TNR (True Negative Rate) :

$$TNR = spécificité = \frac{TN}{TN + FP} \tag{2.23}$$

En outre, il y a d'autres indices qui permettent aussi de mesurer la qualité d'un algorithme d'apprentissage :

– *Taux de faux positifs* généralement noté FPR (False Positive Rate) :

$$FPR = 1 - spécificité = \frac{FN}{TP + FN} \tag{2.24}$$

– *Valeur de prédiction positive* généralement notée $PPV+$:

$$PPV+ = \frac{TP}{TP + FP} \tag{2.25}$$

– *Valeur de prédiction négative* généralement notée $PNV-$:

$$PNV- = \frac{TN}{TN + FN} \tag{2.26}$$

2.3.3.3 Avantages et limitations

Ces mesures sont faciles à calculer. Elles sont souvent utilisées dans la littérature. Toutefois, les différences d'orientation entre deux graphes de la même classe d'équivalence sont comptées comme des erreurs.

2.3.4 Méthode basée sur la distance d'édition

2.3.4.1 Notions

Si l'approche basée sur la sensibilité/la spécificité présentée ci-dessus permet de qualifier un algorithme d'apprentissage au niveau de *similarité* entre le RB appris et le RB "cible", la distance d'édition s'intéresse à la *dissimilarité* entre eux en calculant le coût des opérations de modifications nécessaires

pour transformer le graphe appris vers le graphe "cible". Ces deux mesures sont très proches en terme de principe.

Plus formellement, la distance d'édition de graphes est une mesure inspirée de la distance d'édition (ou distance de Levenshtein) entre deux chaînes de caractères. Elle est définie comme l'ensemble le moins coûteux d'opérations nécessaires pour transformer les graphes de l'un à l'autre. Dans le contexte des RB, la distance d'édition initialement utilisée étant celle entre deux DAG de même noeuds. Nous présentons dans la suite une version améliorée la distance d'édition SHD (*SHD : Structural Hamming Distance*) proposée par [Tsamardinos 2006]. Une distance d'édition entre graphes possédant des ensembles de noeuds différents est décrite dans [Bunke 1997].

2.3.4.2 Méthodes

[Tsamardinos 2006] propose une distance d'édition entre les graphes essentiels EG et EG_0 de deux DAG G et G_0. Pour tous les arcs différents entre deux graphes essentiels, le coût d'édition est augementé de 1 si :

- l'arc est absent dans EG mais présent dans EG_0
- l'arc est présent dans EG mais absent dans EG_0
- l'arc est mal orienté dans EG par rapport au EG_0

L'algorithme SHD entre deux graphes essentiels est présenté dans Algorithme 2.3. La Figure 2.2 donne un exemple d'application de cette distance.

2.3.4.3 Avantages et limitations

Comme l'approche basée sur la sensibilité/la spécificité, l'avantage de cette approche est la simplicité du calcul. Par contre, cette méthode prend aussi en compte l'équivalence de Markov.

2.3.5 Méthodes basées sur la visualisation

2.3.5.1 Principe

En complément des méthodes d'évaluation qui mesurent numériquement la qualité d'un réseau bayésien, les méthodes basées sur la visualisation peuvent être utilisées dans l'évaluation de réseaux bayésiens.

Algorithme 2.3 $SHDEG(EG, EG_0)$ [Tsamardinos 2006]

ENTRÉES: Deux graphes essentiels EG et EG_0.
SORTIES: La valeur SHD entre EG et EG_0.

```
1: shd ← 0 ;
2: A ← liste des arêtes de EG ;
3: pour tout Tous les arcs E différents dans EG et EG0 faire
4:     si E est absent dans EG alors
5:         shd ← shd + 1 ;
6:     finsi
7:     si E est présent dans EG mais absent dans EG0 alors
8:         shd ← shd + 1 ;
9:     finsi
10:    si E est mal orienté dans EG par rapport au EG0 alors
11:        shd ← shd + 1 ;
12:    finsi
13: fin pour
14: retourner G
```

TABLE 2.3 – Algorithme SHD pour deux EG

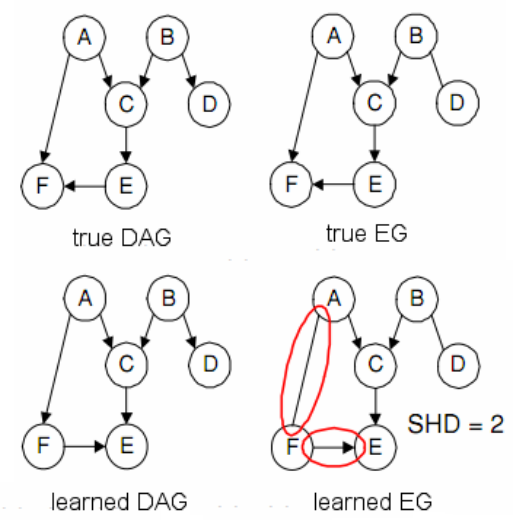


FIGURE 2.2 – Exemple de distance SHD entre deux graphes essentiels [Tsamardinos 2006].

2.3.5.2 Méthodes

La première approche possible [Ciaccio 2010] consiste en deux étapes : tout d'abord utiliser un algorithme de placement, comme par exemple "Force-based algorithms" [Eades 1984], sur l'union des deux graphes G et G_0 (cf. Figure 2.3). Les couleurs des arcs représentent la nature d'arcs : présent uniquement

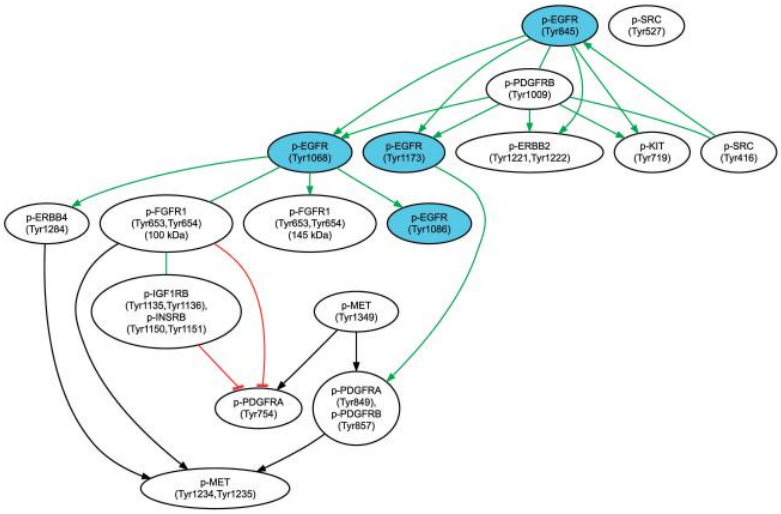


FIGURE 2.3 – La visualisation par le graphe union [Ciaccio 2010].

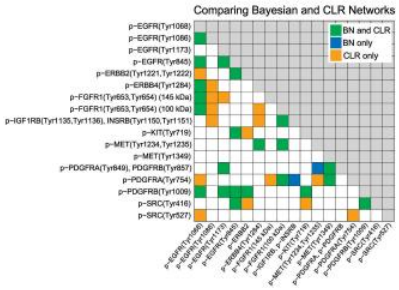


FIGURE 2.4 – La visualisation par carte de chaleur [Ciaccio 2010].

dans le graphe appris, présent uniquement dans le graphe théorique, présent à la fois dans les deux graphes.

La seconde approche proposée par [Ciaccio 2010] utilise une carte de chaleur (cf. Figure 2.4) où chaque arc est représenté par une cellule de la matrice d'adjacence et les couleurs différentes représentent la nature des arcs comme dans la première approche.

2.3.5.3 Avantages et limitations

Les avantages de ces méthodes sont la lisibilité et la compréhension. En effet, toutes les méthodes précédemment présentées donnent les résultats d'évaluation numériques qui ne sont pas très bien interprètables pour les utilisateurs non-experts. Avec une visualisation colorée, il est possible de repérer facile-

Méthodes	Classe d'équiv.	Complex. de calculs	Interp. Graphique
1. Score	+	−	-
2. Divergence KL	+	−	-
3. Sensib./Spécif.	−	+	-
4. Distance d'édition	+	+	-
5. Visualisation	−	−	+

(Classe d'équiv.) : si c'est une solution pour le problème de classe d'équivalence;
(Complex. de calcul) : si le calcul est complexe;
(Interp. Graphique) : si l'interprétation graphique est facile à comprendre;
(+) : plus favorable; *(−)* : moins favorable

TABLE 2.4 – Vue d'ensemble des méthodes d'évaluation

ment les différences entre graphes appris et graphes théorique. Par contre, cette méthode n'est pas applicable lorsque le nombre de variables est important par manque de lisibilité. De plus, cette méthode ne tient pas compte de l'équivalence de Markov.

2.4 Conclusion

Après avoir défini la notion de réseau bayésien, nous avons passé en revue les différences méthodes d'apprentissage de ces modèles. Nous avons aussi abordé la question de l'évaluation de ces algorithmes d'apprentissage.

La Table 2.4 présente une vue d'ensemble des méthodes d'évaluation.

La méthode basée sur *le score* prend bien en compte l'équivalence de Markov lors de l'évaluation. Pourtant si les données sont réduites, il est difficile de distinguer deux réseaux différents par score.

La méthode basée sur *la divergence KL* est aussi capable de prendre en compre l'équivalence de Markov, pourtant son calcul est coûteux lors que le nombre de variables est important.

Les méthodes basées sur *la sensibilité/la spécificité* et sur *la distance d'édition* reposent sur un principe assez proche. Sensibilité/Spécificité sont simples en terme de calculs et indépendantes des données. Cependant elles ne prennent pas en compte l'équivalence de Markov à la différence de la distance d'édition.

Les méthodes basées sur *la visualisation* donnent une représentation intuitive, mais sont limitées par des réseaux de grande taille et ne prennent pas en compte l'équivalence de Markov.

Chapitre

3 Apprentissage ensembliste de réseaux bayésiens

Sommaire

Comme mentionné dans le chapitre 2, l'apprentissage de la structure de réseaux Bayésiens à partir de données est un problème NP-complet [Chickering 1996], les heuristiques sont souvent utilisées pour trouver un bon optimum local. De plus, dans plusieurs applications réelles, la quantité de données disponibles est faible par rapport aux nombre de variables.

Pour ces raisons, il existe plusieurs algorithmes d'apprentissage de structure qui proposent d'utiliser les approches évolutionnaires [Opitz 1999] (e.g. Bootstrapping [Friedman 1999b, Rodin 2005], algorithmes evolutionnaires [Larrañaga 1994, Delaplace 2007, Muruzbal 2007, Auliac 2007, Wang 2010],

etc...) afin d'apprendre un ensemble de structures candidates au lieu d'une seule.

L'évaluation de qualité d'une seule structure obtenue avec un algorithme d'apprentissage classique est simple, comme abordé dans le Chapitre 2.

Le problème posé dans ce chapitre est relativement plus compliqué : nous ne voulons pas évaluer la qualité d'un seul RB, mais la qualité d'un ensemble de RB. Une première solution triviale est d'estimer la qualité de cet ensemble en utilisant une mesure de qualité (distance d'édition, divergence KL, score...) pour chaque modèle et ensuite d'estimer la moyenne ou de visualiser la distribution de ces résultats. La visualisation de graphes n'est pas appropriée quand le nombre de RB dans l'ensemble est trop élevé.

Nous proposons dans cette partie une approche basée sur le principle inverse : nous devons tout d'abord trouver le modèle graphique "le plus représentatif" pour l'ensemble de RB, appelé **le graphe quasi-essentiel** (QEG - Quasi Essential Graph), et puis nous appliquons les opérateurs d'évaluation usuels pour ce nouvel objet. Un inconvénient de cette approche est que le QEG stocke plus d'informations par rapport à un graphe simple, donc, les opérateurs d'évaluation de qualité doivent être redéfinis pour ce nouvel objet. Par contre, un grand avantage (l'inspiration de la contribution de cette partie) est que le QEG est plus facilement visualisable étant donné une métaphore graphique qui est présentée ci-après.

La Figure 3.1 résume une organisation intuitive des notions-clés de ce chapitre.

3.1 Méthodes ensemblistes pour l'apprentissage de la structure de RB

3.1.1 Approches basées sur le Bootstrap

3.1.1.1 Principes

Ces approches proviennent de la famille des méthodes de sélection de modèles basées sur le principe du Bootstrap proposé par Efron [Efron 1993]. Le principe de ces méthodes est de re-générer des sous-ensembles de données d'ap-

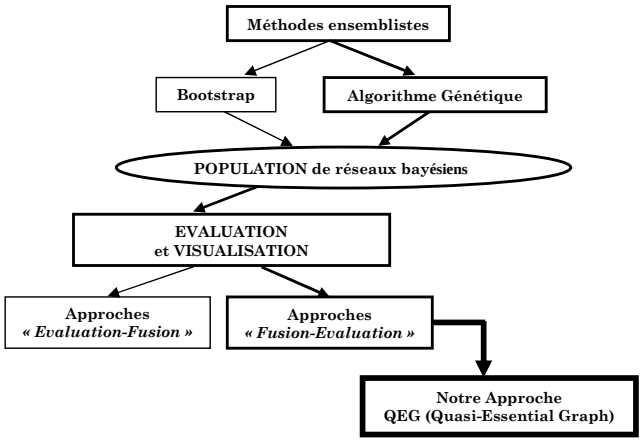


FIGURE 3.1 – L'organisation intuitive des notions-clés du chapitre 3. Les méthodes d'apprentissage de la structure de réseaux bayésiens basées sur *les méthodes ensemblistes* telles que *le bootstrap*, ou *les algorithmes génétiques* permettent d'obtenir *une population de réseaux bayésiens*. Il y a deux approches d'évaluation et de visualisation d'une population de réseaux bayésiens : *approche "Evaluation-Fusion"* et *approche "Fusion-Evaluation"*. Nous proposons une approche de type *"Fusion-Evaluation"* qui s'appelle "QEG" (Quasi-Essential Graph) pour résumer une population de réseaux bayésiens en un représentant graphique de toute la population.

prentissage. Ensuite, l'algorithme d'apprentissage cherche la meilleure structure à partir de chaque sous-ensemble. On obtient donc plusieurs meilleures structures en fonction du nombre de sous-ensembles de données fixé par l'utilisateur.

3.1.1.2 Bootstrap paramétrique et Bootstrap non-paramétrique

Soit un jeu de données d'apprentissage D et un algorithme d'apprentissage de la structure A, la procédure du Bootstrap *non-paramétrique* (cf. Figure 3.2) consiste en deux étapes :

- *Etape 1 :* Pour $i = 1, 2...n$
  - Re-échantillonner avec remise un nouveau jeu de données (noté d_i) à partir de D ;
  - Appliquer A sur d_i : $A(d_i)$;

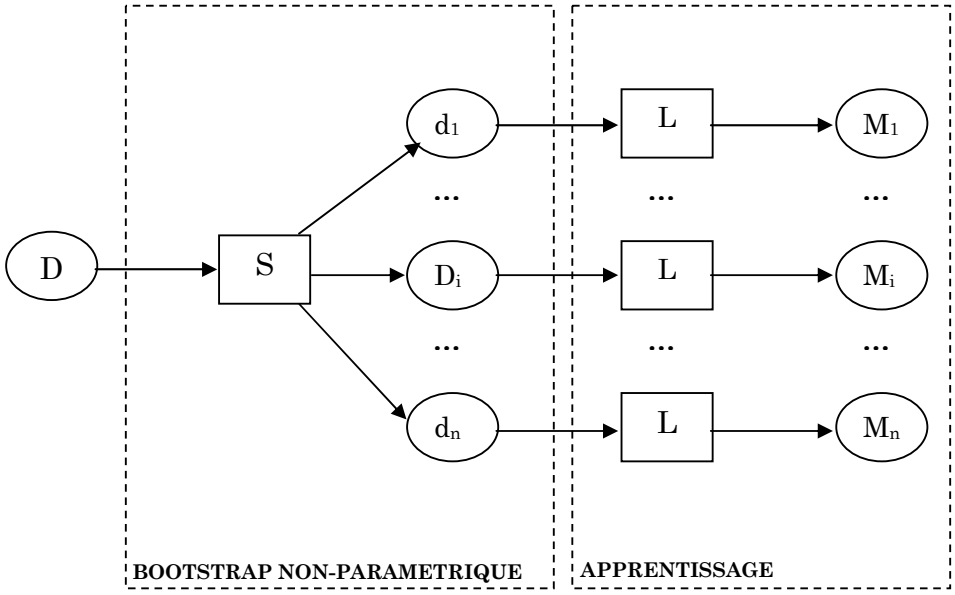


FIGURE 3.2 – Architecture des approches basées sur le Bootstrap non-paramétrique. *D (Data)* : Jeu de données initial pour l'apprentissage ; *S* (Sampling) : l'étape d'échantillonnage ; d_i : des jeux de données ré-échantillonnées à partir de *D* ; *L (Learning)* : un algorithme d'apprentissage de la structure de réseaux bayésiens ; M_i *(Model)* : des modèles obtenus par *L*.

– *Etape 2 :* Pour chaque $A(d_i)$, on définit un score.

Au lieu de re-échantillonner des données à partir du jeu de données initial, l'approche *paramétrique* (cf. Figure 3.3) génère des jeux de données à partir du réseau qui est obtenu par l'apprentissage avec le jeu de données initial. La procédure détaillée consiste en trois étapes :

– *Etape 1 :* Apprendre le réseau initial M_0 à partir de *D* ;
– *Etape 2 :* Pour $i = 1, 2...n$
  – Générer un nouveau jeu de données (noté d_i) à partir de M_0 ;
  – Appliquer *A* sur d_i : $A(d_i)$;
– *Etape 3 :* Pour chaque $A(d_i)$, on définit un score.

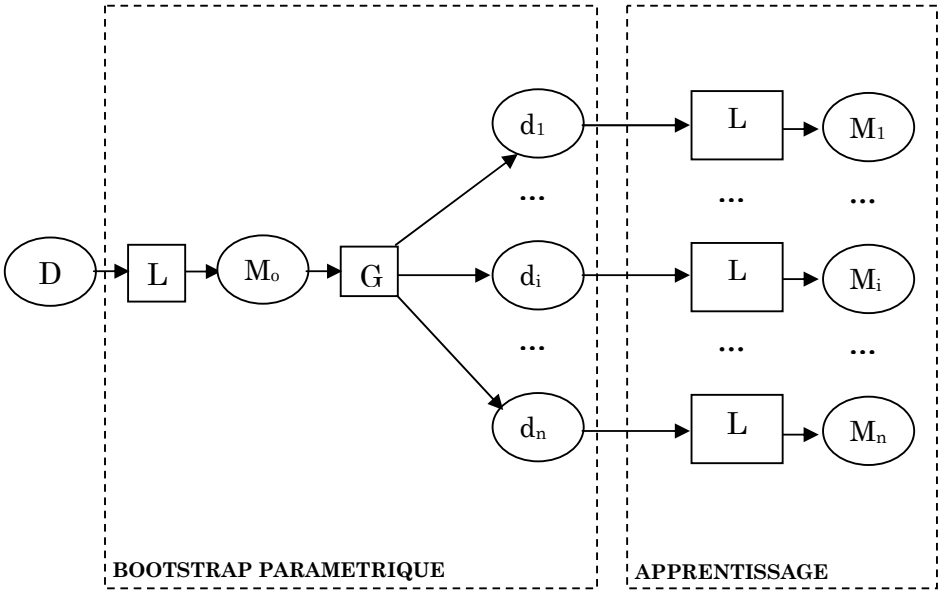


FIGURE 3.3 – Architecture des approches basées sur le Bootstrap paramétrique. *D (Data)* : Jeu de données initial pour l'apprentissage ; *S* (Sampling) : un outils d'échantillonnage ; d_i : des jeux de données ré-échantillonnées à partir de *D* ; *L (Learning)* : un algorithme d'apprentissage de la structure de réseaux bayésiens ; M_i *(Model)* : des modèles obtenus par *L*.

Le but de ces deux types d'approches est le même : générer un ensemble de jeux de données pour l'apprentissage. Cependant, la façon dont chaque approche génère des données est différente : le Bootstrap *non-paramétrique* se base sur le ré-échantillonnage aléatoire avec remise appliqué sur le jeu de données d'apprentissage et le Bootstrap *paramétrique* se base sur le génération de données avec un réseau appris à partir du jeu de données d'apprentissage.

3.1.1.3 Travaux existants

Dans le cadre de l'apprentissage de la structure de réseaux bayésiens avec peu de données, [Friedman 1999a] (*paramétrique* et *non-paramétique*), [Friedman 1999b] (*non-paramétique*), [Friedman 2000] (*non-paramétique*), [Pe'er 2001] (*non-paramétique*), [Djebbari 2008] (*non-paramétique*) font partie

des premiers travaux utilisant une approche basée sur le bootstrap.

Quelque soit l'approche choisie, le résultat à la fin de la procédure du Bootstrap est un ensemble de réseaux associés avec un score. En général, ce score permet ensuite de choisir la meilleure structure.

3.1.1.4 Avantages et inconvénients

Ces méthodes présentent l'avantage d'être simple à programmer. Ce ne sont, en effet, que des tirages aléatoires. Toutefois, le temps d'apprentissage devient grand si le nombre de sous-ensembles de données est grand. Car il faut apprendre plusieurs fois (en fonction du nombre de sous-ensembles échantillonnés) pour choisir la meilleure structure.

3.1.2 Approches basées sur les algorithmes génétiques

3.1.2.1 Principes

Les **algorithmes génétiques** (AG) sont des algorithmes évolutionnaires pour l'optimisation heuristique s'appuyant sur l'évolution au fil des générations d'un ensemble d'individus, chaque individu représentant une solution potentielle au problème. La qualité d'une solution est évaluée par une fonction de satisfaction (fitness). La recherche des meilleures solutions est guidée par la maximisation de cette fonction.

3.1.2.2 Méthode

Plus techniquement, les AG prennent chaque RB à l'entrée comme un individu (candidat) afin de constituer la population initiale. Ensuite, ils continuent à appliquer des opérateurs évolutionaires (*la combinaison/le croisement* et *la mutation*) afin d'obtenir à la fin un ensemble des meilleurs RB en fonction de leur valeur de score (fitness) (cf. Figure 3.4). Le principe très général des AG est décrit dans la Table 3.1. La Figure 3.5 présente un exemple de différentes étapes de l'apprentissage de la structure de RB par l'algorithme génétique.

3.1.2.3 Travaux existants

Considérés comme les premiers travaux proposant une approche évolutionnaire pour trouver la meilleure structure de RB, les articles de Larrañaga *et al.* [Larrañaga 1996, nga 1997] sont remarquables en terme de

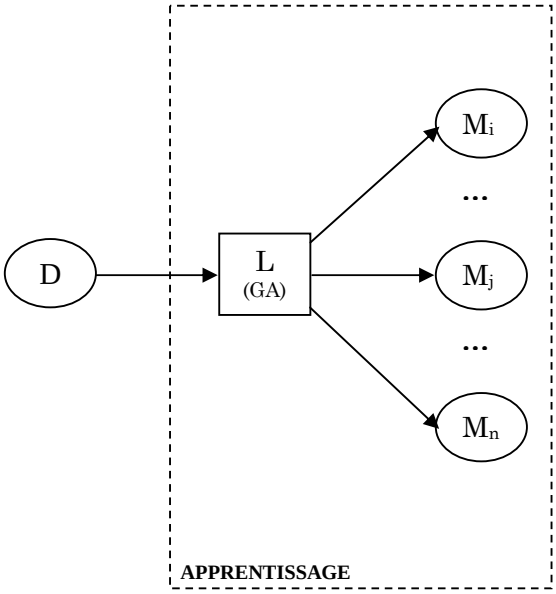


FIGURE 3.4 – Architecture des approches basées sur les algorithmes génétiques. *D (Data)* : Jeu de données initial pour l'apprentissage ; *L*(*GA*) *Learning (Genetic Algorithm)* : apprentissage de la structure de réseaux bayésiens avec les algorithmes génétiques ; M_i *(Model)* : modèles obtenus par *GA*.

Algorithme. Algorithme Génétique

ENTRÉES: Une population d'individus.
SORTIES: Une population de meilleurs individus.

1: Générer alératoirement la population initiale
2: **Répéter**
3: Sélectionner les meilleures solutions.
4: Produire une nouvelle génération en combinant une partie de ces paires.
5: Introduire des mutations à cette nouvelle génération.
6: **Jusqu'à** la condition d'arrêt est vérifiée

TABLE 3.1 – Algorithme génétique

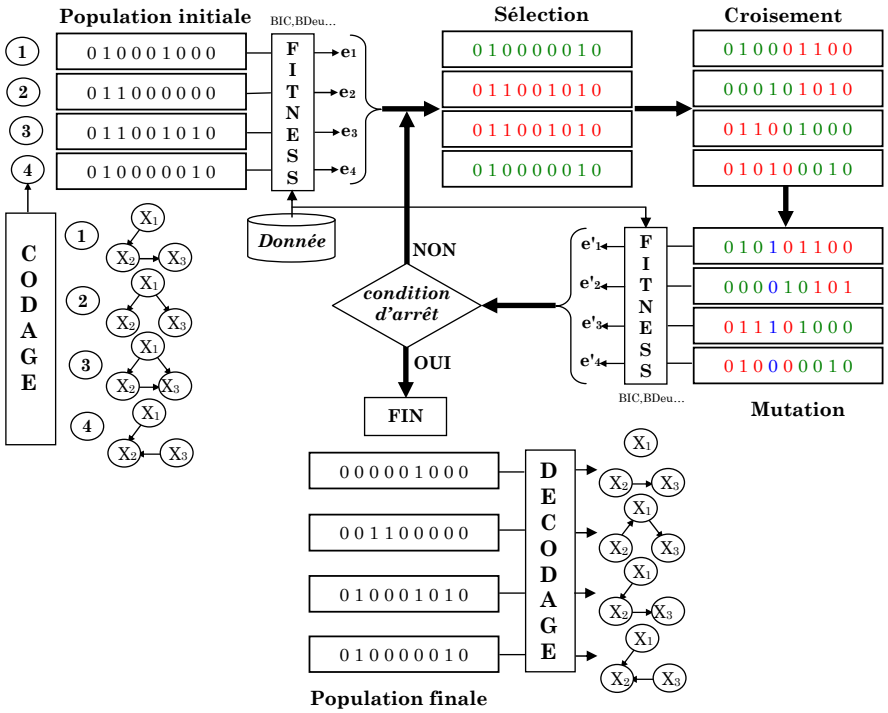


FIGURE 3.5 – Exemple de l'apprentissage de la structure de RB basé sur les AG. *(1,2,3,4)* : la génération aléatoire des DAG ; *(Codage)* : le codage des DAG en individus (candidats) binaires ; *(Fitness)* : l'évaluation des individus par leur fitness (score) ; *(Sélection)* : la sélection des meilleurs individus ; *(Croisement)* le croisement de meilleurs individus ; *(Mutation)* : la mutation des individus croisés ; l'itération à l'étape "*Sélection*" jusqu'à la condition d'arrêt est vérifiée. A la fin de cet apprentissage, on obtient une population finale des meilleurs individus.

solution. Il y a nombreux travaux dans la littérature qui utilisent les algorithmes génétiques pour l'apprentissage de la structure de réseaux bayésiens [Pelikan 2001, Hsu 2002, Haiyang 2008, Lee 2010], avec des applications par exemple la reconstruction de réseaux de régulation génétique tel que [nga 1997, Hsu 2002, Auliac 2007, Auliac 2008]. Dans ces travaux, les individus sont différemment codés : *par une chaîne binaire* [Pelikan 2001, Hsu 2002], *par une matrice d'adjacent binaire* [nga 1997, Hsu 2002, Lee 2010], *par un vecteur ternaire* [Auliac 2007]. Il y a une variété de façons dont les travaux appliquent le principe des AG dans l'apprentissage de la structure de

RB, par exemple : (i) après la mutation, [nga 1997] a proposé un opérateur d'ajout des individus parents et un opérateur de réduction des individus de faibles fitness ; (ii) [Hsu 2002] utilise les AG pour définir l'ordre optimal des variables à l'entrée pour un algorithme d'apprentissage (e.g. K2). (iii) [Pelikan 2001, Auliac 2007] ont mise en revue une approche de "*niching*" qui permet d'obtenir plusieurs optima locaux ; (iv) [Auliac 2007] a proposé différents stratégies de croisement.

3.1.2.4 Algorithme génétique et algorithme d'estimation de distribution

Il est également important de remarquer que dans la littérature il existe une extension des algorithmes génétiques qui s'appelle EDA (Estimation of Distribution Algorithms) [Larrañaga 2002, Sun 2005, Lozano 2006] qui sert éventuellement à faire l'apprentissage de la structure de réseaux bayésiens. C'est une optimisation au niveau des opérateurs évolutionnaires. L'EDA garde quasiment les étapes traditionnelles de l'algorithme génétique, sauf l'étape "*Croisement*" et "*Mutation*". L'EDA remplace ces deux étapes par l'apprentissage d'un modèle probabiliste comme les réseaux bayésiens pour représenter la distribution de probabilité des individus puis générer une nouvelle population selon cette loi.

3.2 Evaluation de méthodes ensemblistes pour l'apprentissage de la structure de réseaux bayésiens

Nous venons de voir plusieurs méthodes permettant de générer un ensemble de RB à partir de données. Comment faire maintenant pour évaluer la qualité de cet ensemble ?

3.2.1 Principes

Il y a deux grandes familles d'approches d'évaluation d'un ensemble de réseaux bayésiens. Il s'agit de l'approche "*Evaluation-Fusion*" et de l'approche "*Fusion-Evaluation*".

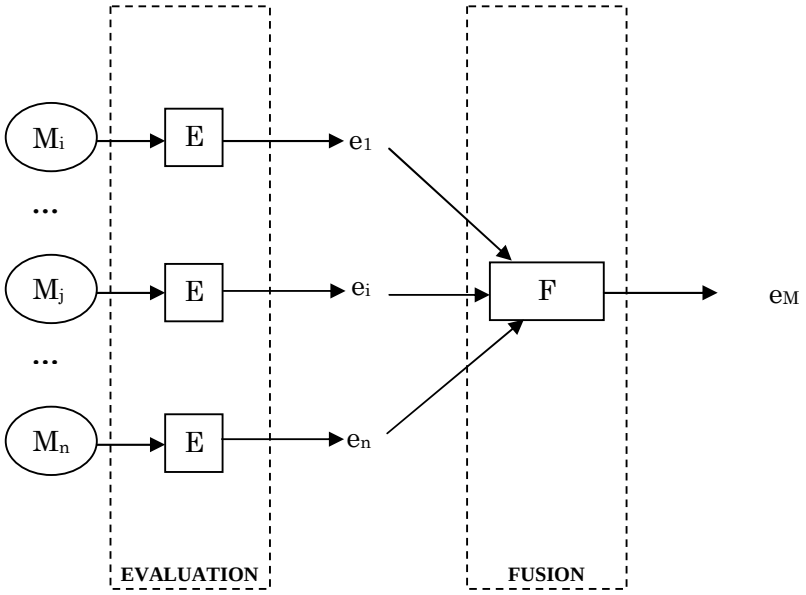


FIGURE 3.6 – Architecture de l'approche "Evaluation-Fusion". M_i : modèle de réseau bayésiens appris par l'apprentissage ; E (Evaluation) : Méthode d'évaluation d'un réseau bayésien ; e_i : la qualité du modèle M_i ; F : la fusion de qualité de modèles par des techniques statistiques simples ; e_M : la qualité globale de la population.

3.2.1.1 Approche "*Evaluation-Fusion*"

L'approche "*Evaluation-Fusion*" consiste *d'abord* à évaluer la qualité de chaque réseau par une mesure existante (*score, KL, SHD... cf. Section 2.3*). *Ensuite*, ces valeurs sont fusionnées ou synthétisées par des techniques statistiques simple (*moyenne, écart-type, boxplot...*). La Figure 3.6 est un schéma intuitif qui présente les étapes de l'approche "*Evaluation-Fusion*".

L'avantage de cette approche est la simplicité de calcul. L'inconvénient est qu'il n'y a pas de modèle final facile à visualiser.

3.2.1.2 Approche "*Fusion-Evaluation*"

La deuxième approche "*Fusion-Evaluation*" (cf. Figure 3.7) consiste *d'abord* à fusionner l'ensemble de réseaux bayésiens en un seul modèle. *Ensuite*,

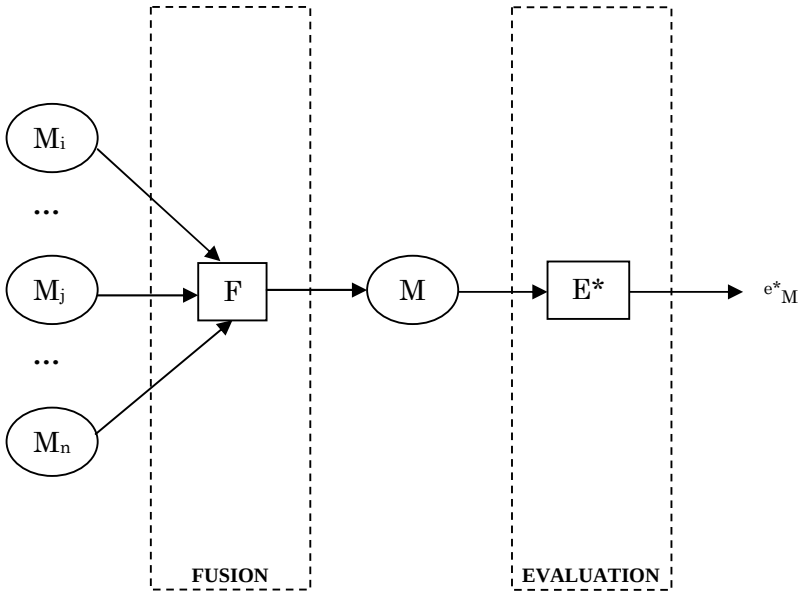


FIGURE 3.7 – Architecture de l'approche "Fusion-Evaluation". M_i : modèle de réseau bayésiens appris par l'apprentissage ; F (Fusion) : Une méthode de fusion de modèles ; M : modèle représentant de la population de réseaux bayésiens ; $E*$: méthode d'évaluation de la qualité du modèle représentant M ; $e*_M$: la qualité globale de la population.

il faut évaluer ce nouveau modèle pour estimer la qualité de tout ensemble.

Il existe différentes approches de fusion de RBs dans la littérature

1. [Benferhat 2011] ont d'abord proposé un opérateur "produit" pour la fusion de RB de même structure. Cette solution est ensuite adaptée dans le cas de structures différentes en proposant aussi un moyen de gérer les informations conflictuelles.

2. [Santos Jr. 2011] ont proposé une union des règles de connaissances caractérisées par la fiabilité de la source et la probabilité de chaque règle. Le réseau obtenu est une base de connaissances bayésiennes (Bayesian Knowledge Bases - BKB), et les dépendances sont formées par les états des variables aléatoires.

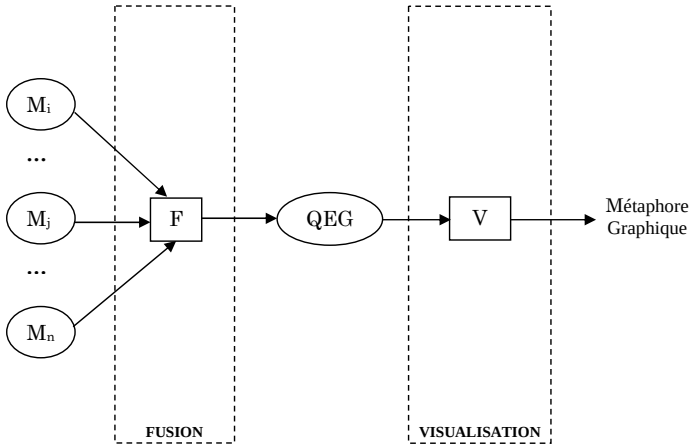


FIGURE 3.8 – Architecture de l'approche basée sur le QEG.

Il est plus difficile de fusionner des modèles que des indicateurs numériques. Par contre, le résultat de la fusion est facilement visualisable.

Nous présentons dans la suite notre approche qui permet à la fois d'obtenir un représentant de l'ensemble de réseaux et une visualisation intuitive sous forme graphique.

3.3 Notre approche : *QEG (Quasi-Essentiel Graph)*

Nous proposons une méthode de fusion de graphes décrite dans la Figure 3.8. Nous avons vu dans la section qu'un ensemble de DAG équivalents peut être exactement résumé par le graphe essentiel associé à la classe d'équivalence. Nous définissons ici la notion de QEG (Quasi Essential Graph), *graphe quasi-essentiel*, afin de résumer approximativement un ensemble de DAG quelconques. Puis, nous donnons quelques propriétés intéressantes du QEG. Ensuite, nous décrivons un algorithme pour construire le QEG à partir un ensemble de DAG donné. Et enfin, nous proposons une métaphore graphique permettant de visualiser ce nouvel objet graphique.

3.3.1 Définition

Un QEG *(Quasi Essential Graph)* $\langle V, G, w_u, w_a \rangle$ est un graphe pondéré défini par $V = \{X_1, \ldots, X_n\}$, un ensemble de variables aléatoires, un graphe G, où chaque noeud représente une variable dans V, un ensemble de poids w_u associé à chaque arc non-dirigé dans le squelette de G, et un ensemble de poids w_a associé à chaque arc dirigé de G.

Un QEG Q peut résumer un ensemble de DAG $\mathcal{G}$, pour un seuil donné $\alpha > 0.5$, si et seulement si ces trois conditions suivantes sont vérifiées :

(C_0) Q et tous les DAG dans $\mathcal{G}$ sont définis pour le même ensemble de variables V,

(C_1) deux noeuds X_i et X_j sont adjacents dans Q ssi leur probabilité d'être adjacents dans $\mathcal{G}$ est égale à $w_u(X_i – X_j)$ et supérieure à α,

(C_2) un arc dirigé $X_i \rightarrow X_j$ existe dans Q ssi sa probabilité d'être présent dans $EG(\mathcal{G})$ (l'ensemble de graphes équivalents associés à $\mathcal{G}$), quand X_i et X_j sont adjacents, est égale à $w_a(X_i \rightarrow X_j)$ et supérieure à α.

3.3.2 Propriétés

Le QEG résume l'information contenue dans l'ensemble de DAG $\mathcal{G}$ donné : le graphe vide quand il n'y a aucune information (distribution aléatoire) ou le graphe essentiel quand les informations sont consistantes (tous les DAG dans la même classe d'équivalence, ou avec de petites perturbations autour du graphe donné).

3.3.2.1 QEG pour les RB de la même classe d'équivalence

Le QEG résumant $\mathcal{G}$ un ensemble de DAG appartenant à la même classe d'équivalence $G = EG$ est défini par un graphe partiellement dirigé EG et les poids w_u et w_a sont égaux à 1.

Comme tous les DAG dans $\mathcal{G}$ ont le même squelette, $w_u = 1$ pour toutes les arêtes de ce squelette, et comme ils ont aussi les mêmes arcs non réversibles, $w_a = 1$ pour toutes les flèches concernées et G est donc défini par ce squelette commun et ces arcs non réversibles, ce qui est la définition du graphe essentiel d'une classe d'équivalence.

3.3.2.2 QEG pour les RB quelconque à partir d'une distribution uniforme dans l'espace DAG

Le QEG résumant un ensemble $\mathcal{G}$ de DAG générés par une distribution uniforme dans l'espace DAG est défini par un DAG partiellement dirigé G. Ce dernier est égal au graphe non orienté complètement connecté G_c ou un graphe vide G_0 selon la valeur de α, les poids w_u sont égaux à 1 si $G = G_c$, 0 sinon et $w_a = 0$.

Il y a deux manières pour connecter deux noeuds (orienté à gauche ou à droite) pour trois configurations. Même si on enlève toutes les configurations guidant vers un circuit (impossible pour un graphe orienté sans circuit), la probabilité de la connexion entre deux noeuds est inférieure à 2/3 et supérieure à 0.5. Donc, les arcs vont être présents dans G (guidant vers $G = G_c$) si α est faible. Si α augmente, les arcs vont être enlevés et $G = G_0$.

3.3.3 Détermination

L'algorithme *DAG2QEG* décrit dans la table 3.2 comment déterminer le QEG Q associé à un ensemble de DAG $\mathcal{G}$ pour un seuil donné α.

Afin d'éviter des problèmes concernant l'équivalence de Markov, l'étape pré-traitement consiste à considérer le graphe essentiel correspondant aux éléments de $\mathcal{G}$ et à estimer la fréquence de chaque arc non orienté de son squelette(étape 3 à 6).

Cet algorithme comprend deux phases : nous déterminons (au cours des étapes 8 à 10) le squelette de Q et estimons les poids w_u de chaque arc non dirigé en utilisant la condition C_1 de définition d'un QEG (cf. section 3.3.1). Ensuite nous estimons (étapes 11 à 19) le poids potentiel de chaque arc et son orientation dans l'ensemble $\mathcal{B}$ et nous utilisons la condition C_2 pour garder les flèches consistantes possibles et leurs poids w_a correspondants.

3.3.4 Métaphore graphique pour la visualisation

L'inspiration de la contribution de cette partie est liée au fait que la visualisation d'un ensemble de DAG n'est pas très pratique. Le QEG (Quasi Essential Graph) est défini comme un nouvel objet graphique. Nous propo-

sons dans cette partie une métaphore graphique pour visualiser cet objet.

Pour un graphe pondéré classique, les poids sont seulement définis pour les arêtes (liens non-dirigé), et la métaphore graphique est simplement obtenue en variant la taille du trait correspondant.

Comme le QEG associe aussi un poids spécifique aux arcs (liens dirigés), nous allons aussi faire varier l'épaisseur des flèches correspondantes. Nous utilisons différentes couleurs pour mieux différencier ces deux éléments graphiques.

3.3.5 Evaluation de la qualité d'un QEG

Nous proposons une approche d'évaluation de la qualité du QEG basée sur la méthode d'évaluation d'un algorithme d'apprentissage par visualisation présenté dans la Section 2.3.5. Cette approche consiste à construire un graphe union du graphe théorique G_0 et du QEG mais sans prise en compte du poids des arêtes (cf. Figure 3.9). Les couleurs des arêtes et des flèches représentent les différences entre les deux graphes : vert = présent dans le QEG et absent dans le G_0, rouge = absent dans le QEG et présent dans le G_0, noire = présent à la fois dans le QEG et le G_0.

3.3.6 Expérimentations

3.3.6.1 Protocole expérimental

Nous proposons dans cette partie un jeu d'exemple pour illustrer l'intérêt de notre approche QEG. Nos algorithmes ont été implémentés sous C++ avec la bibliothèque Boost [1] et des APIs fournis par la plateforme ProBT [2]. Les outils de visualisation sont fournis par Tulip [3] et Graphiz [4]..

3.3.6.2 Avec une distribution uniforme

Nous avons fait une expérimentation avec une population de DAG de 8 variables qui est générée avec une distribution uniforme. Le QEG défini avec $\alpha = 0.5$ ainsi obtenu est un graphe vide.

1. `http://www.boost.org`
2. `http://bayesian-programming.org`
3. `http://tulip.labri.fr`
4. `http://graphiz.org`

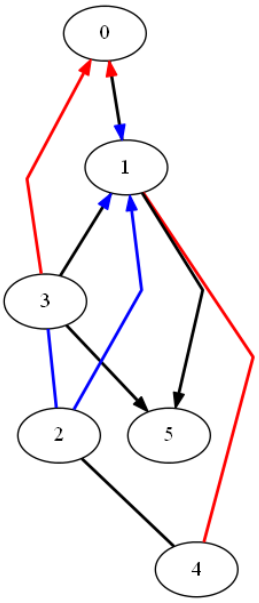


FIGURE 3.9 – Graphe union du graphe théorique G_0 et du QEG mais sans pris en compte du poids des arêtes. Les couleurs des arêtes et des flèches représentent les différences entre : vert = présent dans le QEG et absent dans le G_0, rouge = absent dans le QEG et présent dans le EG_0, noire = présent à la fois dans le QEG et le G_0.

3.3.6.3 Avec une distribution autour d'un graphe fixé

La Figure 3.10 donne un ensemble de 12 DAG générés aléatoirement autour du premier DAG. Ces variations sont des additions, suppressions ou inversions entre un et trois arcs dans le graphe initial.

La Figure 3.11(a) décrit la squelette du QEG obtenu après la première étape de notre algorithme (avec $\alpha = 0.55$). La Figure 3.11(b) présente des arcs non réversibles consistants ajoutés lors de la deuxième étape de notre algorithme. Nous pouvons noter que ce graphe correspond au graphe essentiel du graphe initial. Ce résultat est cohérent par rapport à la définition du QEG. La Figure 3.11(c) propose le QEG final en utilisant notre métaphore graphique.

3.3.6.4 Avec une population apprise par méthode ensembliste

La Figure 3.12 montre le QEG obtenu par l'apprentissage de la structure de RB par bootstrap et recherche glouton sur les données INSULIN décrites

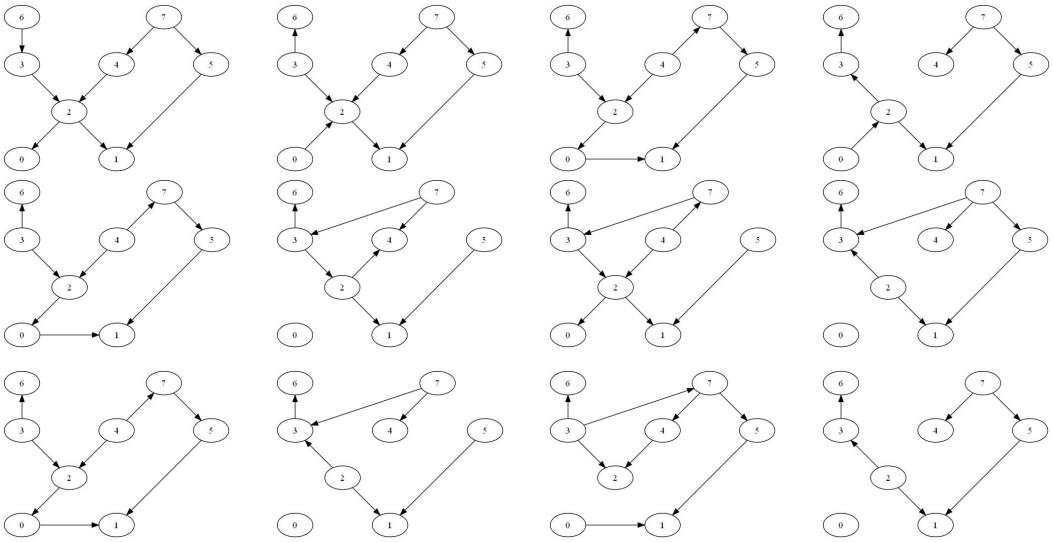


FIGURE 3.10 – L'ensemble de 12 DAG obtenus par la perturbation aléatoire à partir du premier graphe.

dans le Chapitre 5.

3.4 Conclusion

L'apprentissage ensembliste permet de construire un ensemble de modèles. Cette méthode est utilisée lorsque le nombre de données est réduit. Nous avons passé en revue deux approches ensemblistes, les algorithmes génétiques et le bootstrap. Pour évaluer ensuite la qualité de l'ensemble des modèles générés, nous avons proposé une approche de type fusion-évaluation. La fusion des DAG nous donne notre Quasi-Essential Graph (QEG), qu'il est plus facile de visualiser.

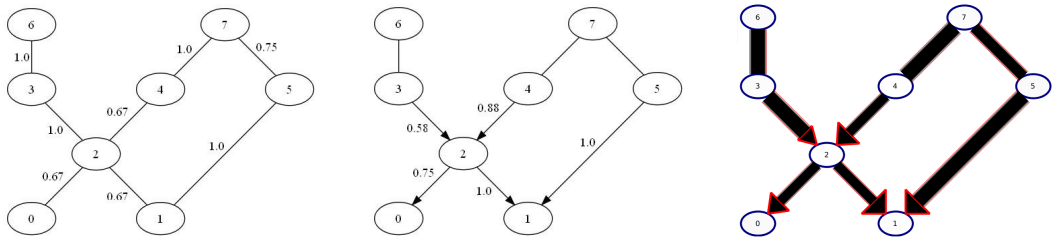


FIGURE 3.11 – (a) Le squelette du QEG obtenu après la première phase de notre algorithme ; (b) Les arcs non réversibles ajoutés au squelette précédent après la deuxième étape de notre algorithme ; (c) La métaphore graphique du QEG utilisant la plateforme Tulip

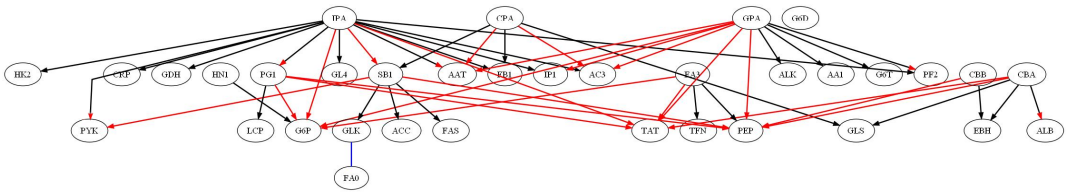


FIGURE 3.12 – *Le graphe union de graphe essentiel du graphe théorique et du QEG obtenu par l'apprentissage de la structure avec Bootstrap dans le contexte du réseau INSULIN.*

Algorithme 3.2 $DAG2QEG(\mathcal{G}, \alpha)$

Entrées: Un ensemble de DAG, $\mathcal{G}$, et un seuil, α ($\alpha > 0.5$).
Sorties: Un graphe quasi-essentiel QEG Q.

1: $N \leftarrow card(\mathcal{G})$;
2: $Q \leftarrow \emptyset$;
3: **pour** $i = 1$ to N **faire**
4: $EG_i \leftarrow EG(\mathcal{G}(i))$; {Phase de traitement}
5: **fin pour**
6: $[UG, w(u)] \leftarrow Union\{Skeleton(EG_i)\}$;
7: **pour** chaque arc $u \in UG$ **faire**
8: **si** $w(u) > \alpha$ **alors**
9: $add\_edge(u, Q)$; {Détermination de squelette}
10: $w_u(u) = w(u)$;
11: $w(\overleftarrow{u}) = card(\{EG_i | \overleftarrow{u} \in EG_i\})/(N * w(u))$; {Détermination de flèches}
12: $w(\overrightarrow{u}) = card(\{EG_i | \overrightarrow{u} \in EG_i\})/(N * w(u))$;
13: **si** $w(\overleftarrow{u}) > \alpha$ **alors**
14: $orient\_edge(\overleftarrow{u}, Q)$;
15: $w_a(\overleftarrow{u}) \leftarrow w(\overleftarrow{u})$;
16: **sinon si** $w(\overrightarrow{u}) > \alpha$ **alors**
17: $orient\_edge(\overrightarrow{u}, Q)$;
18: $w_a(\overrightarrow{u}) \leftarrow w(\overrightarrow{u})$;
19: **finsi**
20: **finsi**
21: **fin pour**
22: **retourner** Q

Notations :
- $EG(\mathcal{G}(i))$: algorithme retournant le graphe essentiel d'un DAG (cf. Algorithme 2.1)
- $Union\{Skeleton(EG_i)\}$: algorithme génère un graphe non-orienté qui est l'union d'un ensemble de squelettes. Cet algorithme compte la fréquence de chaque arête dans cet ensemble
- pour une arête $u = X_i$–X_j, deux orientation possibles sont $\overleftarrow{u} = X_i \leftarrow X_j$ et $\overrightarrow{u} = X_i \rightarrow X_j$

Table 3.2 – Algorithme retourne le graphe quasi-essentiel QEG Q pour un ensemble de DAG $\mathcal{G}$ et pour un seuil α donné

Chapitre

4

Etude différentielle de deux populations de réseaux bayésiens

Sommaire

Un système complexe est souvent représenté sous la forme de graphe. Le graphe d'un réseau bayésien est une représentation d'un ensemble d'objets, qui se compose essentiellement d'un ensemble fini de paires ordonnées d'objets (appelés *noeuds*/*sommets*) reliés par des liens (appelés *arêtes*/*arcs*). Dans l'apprentissage automatique, les graphes peuvent être appris à partir de données observées. Normalement, la différence entre les graphes peut prédire la différence entre deux systèmes. Cependant, il est difficile de décrire exactement cette différence. En effet, dans de nombreuses applications réelles, la

taille des échantillons de données disponibles est beaucoup moins grande que le nombre de variables observées. Pour cette raison, les méthodes ensemblistes (cf. Chapitre 3) proposent d'apprendre à partir de données observées de chaque système un ensemble de graphes. Cet ensemble peut représenter le plus exactement possible le système complexe. Ensuite, pour la comparaison de deux systèmes, il faut un mécanisme permettant de comparer deux ensembles de graphes correspondant de ces deux systèmes.

La contribution principale de cette section est une approche basée sur le test multiple pour comparer deux ensembles de modèles basés sur les graphes. Un jeu de *réseaux Bayésiens* simulé nous permet de démontrer la performance de l'approche proposée.

Dans ce travail, nous avons trois questions à traiter :

Tout d'abord, un des enjeux majeurs pour les approches de comparaison de graphes est l'équivalence de Markov. En effet, certaines arêtes peuvent être inversées sans changer le modèle d'indépendance. Autrement dit, deux graphes structurellement différents peuvent être équivalents. Ainsi, une comparaison directe de deux graphes est impossible. Une solution est d'utiliser une propriété de l'équivalence de Markov : tous les graphes équivalents peuvent être résumés par un graphe essentiel (cf. Sectoion 2.1.4). Cela signifie que pour comparer deux graphes, nous comparons les graphes essentiels de ces deux graphes.

Ensuite, à l'heure actuelle, il n'y a pas de manière simple de formaliser une distribution de probabilité des graphes orientés sans circuits. Par conséquent, afin de comparer des graphes, nous devrons transposer le problème sur les arcs de graphes. Pour cela, comme il y a de nombreuses arêtes dans chaque graphe, nous appliquons un test multiple pour valider si la différence est bien significative. Lorsque plusieurs hypothèses sont testées, le risque de commettre des erreurs de Type I (faux positifs) augmente, souvent fortement et relativement par rapport au nombre d'hypothèses. Il faut donc corriger le seuil de signification global du test.

Finalement, afin de limiter le nombre de tests, nous avons proposé une approche qui consiste à éliminer certains arcs inutiles à tester. Ces arcs ont

une faible probabilité d'occurrence. Cette approche se base sur le graphe quasi-essentiel (QEG - Quasi Essential Graph) qui résume statistiquement chaque ensemble de RB dans un graphe représentant unique. Le QEG nous permet de trouver les arcs les plus pertinents dans chaque ensemble original de graphes.

La Figure 4.1 résume une organisation intuitive des notions-clés de ce chapitre.

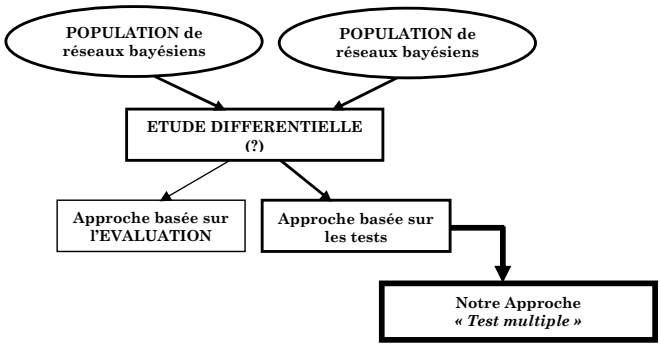


FIGURE 4.1 – L'organisation intuitive des notions-clés du chapitre 4. *L'étude différentielle* de *deux populations de réseaux bayésiens* consiste à identifier les différences entre ces deux populations. Il y a deux approches possibles : la première approche se base sur *les méthodes d'évaluation*, c'est-à-dire qu'on évalue la qualité de chacune de ces deux populations puis on compare leur qualité, et la deuxième approche consiste à utiliser *un test statistique* pour valider si les différences entre deux populations sont significatives. Nous proposons une nouvelle approche basée sur *le test multiple* qui permet de répondre (1) si les différences sont significatives et (2) où sont ces différences.

4.1 Méthodes pour l'étude différentielle de deux populations de réseaux bayésiens

4.1.1 Principe

La méthode classique décrite dans la Figure 4.2 consiste à utiliser les mesures existantes pour l'évaluation de réseau bayésiens (voir chapitre 3). L'idée

est simple : *évaluer la qualité de chaque population et ensuite comparer leur qualité par un test statistique.*

Cette méthode reprend toutes les étapes d'évaluation d'une population de réseaux bayésiens par l'approche "*Evaluation-Fusion*" présentés dans 4.2.1. La méthode d'évaluation permet d'évaluer localement la qualité de chaque réseau bayésien. Pour chaque population, nous obtenons donc un ensemble de valeurs qui estiment la qualité de chaque réseau.

Une fois que nous obtenons deux ensembles de qualités des populations, l'utilisation d'un test statistique permet de comparer la distribution entre ces deux ensembles. Il est possible d'utiliser par exemple un test paramétrique comme le test sur la moyenne, ou un test non paramétrique comme le test du rang.

4.1.2 Avantages et inconvénients

Cette approche est simple à mettre en oeuvre, et permet d'identifier si les deux populations sont différentes. Cependant, elle ne permet pas de localiser les différences.

4.2 Notre proposition : *Test multiple*

4.2.1 Principe

L'approche précédente a un désavantage important : *l'impossibilité d'identifier où sont les différences.* Par conséquent, nous proposons de transposer l'évaluation au niveau des graphes au niveau des arcs. C'est-à-dire qu'il faut vérifier s'il y a des différences au niveau des arcs, et si oui, où sont elles ? Nous avons choisi le test multiple qui peut répondre à ces deux questions indiquées.

La Figure 4.3 présente une vue globale sur les étapes de l'approche que nous proposons et que nous allons détailler en reprenant les étapes classiques de construction d'un test statistique :

1. Définir la population et les variables observées.
2. Choisir le type de test.
3. Définir l'hypothèse nulle H_0.
4. Calculer la variable de décision.

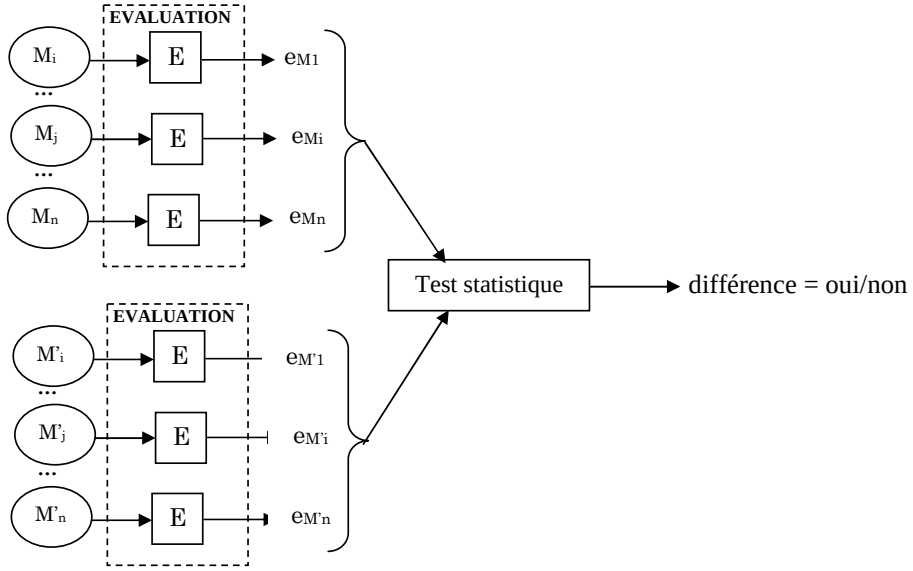


FIGURE 4.2 – Schéma de l'approche basée sur un test statistique. M_i : modèle de réseau bayésiens appris par l'apprentissage ; *E (Evaluation)* : Méthode d'évaluation d'un réseau bayésien ; e_{Mi} et $e_{M'i}$: la qualité du modèle M_i et M'_i ;

5. Choisir le risque de première espèce α (cf. Annexe 1).
6. Conclure le test en comparant α à α_{seuil}.

4.2.2 Population et variables observées

Soit M_i un RB $\mathcal{B} = (V, E_i)$. Deux populations de RB $\{M_i\}$ et $\{M'_i\}$ sont définies sur le même ensemble de noeuds V. Les variables aléatoires qui vont nous intéresser sont les états "présent"/"absent" d'un arc $E_i = \{X_i \rightarrow X_j\}$ ou une arête $E_i = \{X_i - X_j\}$, $\forall X_i \in V et X_i \neq X_j$. Ces deux populations sont non-appariées.

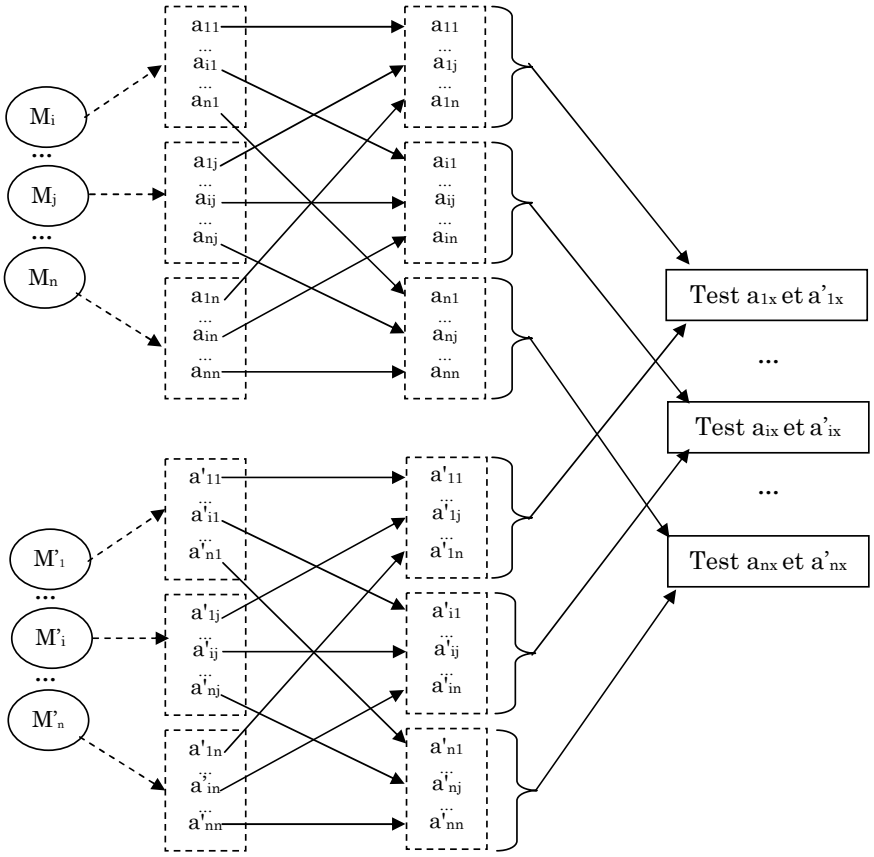


FIGURE 4.3 – Schéma de l'approche basée sur un test multiple. M_i : modèle de réseau bayésiens appris par l'apprentissage ; a_ij : les noeuds du réseaux concernés, a_ij prend deux états "*absence*"/"*présence*" ou 0/1 ; la valeur d'état de chaque arc dans tous les réseaux d'une population est réunie dans un sous-ensemble, on obtient donc deux ensembles de valeurs pour chaque arc ; On applique simultanément une série de tests (test multiple) sur chaque couple d'ensembles de valeurs.

4.2.3 Choix de test

La variable d'observation est l'apparition d'arcs qui prend deux valeurs *"absent"/"présent"* (variable qualitative). Dans le contexte de la comparaison de deux ensembles de graphes, nous proposons d'étudier la fréquence de l'apparition d'arcs dans chaque ensemble de graphes. C'est pourquoi, dans ce travail, nous avons choisi des tests sur les données de fréquence dans laquelle

	Pop_1	Pop_2	Total
Absent	a	b	$a+b$
Present	c	d	$c+d$
Total	$a+c$	$b+d$	n

TABLE 4.1 – La table de contingence représente la différence d'occurrence d'un arc entre deux différent ensembles de graphes. *Légende :* Pop_1 et Pop_2 représente respectivement le premier et le deuxième ensemble de graphes.

le test exact de Fisher est un test approprié pour notre problème. En fait, nous n'avons pas toutes les informations sur la distribution de données de graphes, nous ne pouvons pas utiliser le test binomial. D'une part, le test de Fisher donne toujours la valeur exacte p. D'autre part, le test du Chi-deux est simple à calculer mais les résultats ont seulement une valeur approximative. Par conséquent, nous préférons le test exact de Fisher (cf. Equation 4.1) pour assurer la précision de résultats.

$$p = \frac{\binom{a+b}{a}\binom{c+d}{c}}{\binom{n}{a+c}} = \frac{(a+b)!(c+d)!(a+c)!(b+d)!}{a!b!c!d!n!} \tag{4.1}$$

avec a, b, c, d et $n = a+b+c+d$ définis dans la Table 4.1

4.2.4 Correction de seuil de signification

Normalement, l'approche la plus commune du problème des tests multiples se réalise en deux étapes : (1) *calculer une statistique* T_i *pour chaque test* i ; (2) appliquer une procédure de tests multiples pour déterminer quelles hypothèses rejeter convenablement sous un contrôle d'erreurs de type I (faux positifs) au niveau de signification α. Si nous appliquons classiquement un seul test avec $\alpha = 0.05$, la probabilité d'obtenir un résultat faux positif est 0.05 et la probabilité de ne pas obtenir un résultat faux positif est égale à $\alpha - 0.05 = 0.95$. Maintenant, supposons que, nous effectuons $m = 10$ tests, chacun avec $\alpha = 0.05$. La probabilité que nous obtiendrons au moins un faux

Nom	Définition	Formule
PCER (Per-comparison error rate)	La valeur espérée du nombre de d'erreurs de Type I divisé par le nombre d'hypothèses	$PCER = E(V)/m$
PFER (Per-family error rate)	La valeur espérée du nombre d'erreurs de Type I	$PFER = E(V)$
FWER (Family-wise error rate)	La probabilité de commettre au moins une erreur de Type I	$FWER = Pr(V \geq 1)$
FDR (False discovery rate)	La proportion espérée d'erreurs de Type I parmi des hypothèses nulles rejetées	$FDR = E(Q)$

TABLE 4.2 – La liste de types de correction de seuil de signification. *Légende* : V est le nombre d'hypothèses nulles vraies rejetées ; m est le nombre d'hypothèses ; $Q = V/R$, où (R = le nombre d'hypothèses nulles rejetées), si $R > 0$; $Q = 0$ si $R = 0$

résultat positif est égale à $1 - 0.95^m = 1 - 0.95^{10} = 0.4$. Par conséquence, le problème est que la probabilité d'obtenir au moins un résultat faux positif est quasiment *certaine* lorsque nous testons 1000 tests avec $\alpha = 0.05$. Cela signifie que nous ne pouvons trouver aucune évidence pour rejeter l'hypothèse nulle. Ainsi, nous devons corriger le seuil de signification de manière moins conservatrice dans le but d'augmenter la possibilité de rejeter l'hypothèse nulle. Cette procédure est appelée *le contrôle d'erreur de première espèce* ou *la correction de seuil de signification* α . Il y a quatre catégories d'erreurs de Type I présentées dans le Table 4.2 [Dudoit 2003].

Dans la littérature, les méthodes de correction de seuil de signification sont quasiment toutes basées sur le FWER (tel que la correction de Bonferroni et la correction de Bonferroni-Holm) et FDR (la correction de Benjamini-Hochberg). La correction de Bonferroni [Abdi 2007] propose de diviser localement le seuil classique α par le nombre de tests effectués. Par exemple, si nous voulons appliquer $k = 1000$ tests, le seuil dit "local" $\alpha_i = \alpha/k = 0.05/1000 = 0.00005$. Si la p-*value* de chaque test i est inférieure à α_i corrigée, on rejette l'hypothèse nulle.

La correction de Bonferroni est appelé un méthode *"étape-seule"* (single-step). α_i est corrigée une seule fois pour tous les tests. A partir de la correction de Bonferroni, [Holm 1979] a proposé une méthode *"progressive"* (stepwise) (α_i est corrigé, étape par étape, pour chaque test), cette méthode s'appelle la

correction de Bonferroni-Holm.

La correction de Bonferroni-Holm examine chaque hypothèse dans une séquence ordonnée, et la décision d'accepter ou de rejeter l'hypothèse nulle dépend des résultats de l'hypothèse nulle du test précédent (en commençant par la p-*value* la plus petite, et continue jusqu'à ce qu'il échoue à rejeter une hypothèse nulle).

Par rapport à la correction Bonferroni simple, celle de Bonferroni-Holm est plus puissante, c'est-à-dire que $1 - \beta$ est plus grande (cf. Table A1.1) et moins conservatrice, c'est-à-dire qu'elle rejette moins d'hypothèses nulles qui sont vraies. Puisque la correction Bonferroni simple compare toutes les p-*values* pour α/k. La méthode de Bonferroni-Holm rejette plus d'hypothèses nulles par rapport à la méthodes Bonferroni simple. Cependant, ces approches sont encore très conservatrices. C'est pourquoi [Benjamini 1995] a proposé la correction Benjamini-Hochberg, qui est moins conservatrice que les méthodes présentées ci-dessus. La correction Benjamini-Hochberg est basée non seulement sur la probabilité d'obtenir au moins un faux positif (FWER), mais aussi sur la proportion de faux positifs parmi les hypothèses nulles rejetées (FDR). Cette proportion peut être pré-définie par l'utilisateur.

Dans le contexte de tests multiples, nous nous attendons à ce que la valeur de FDR soit inférieure à un seuil global. Cela nécessite de corriger chaque α_i local afin d'éviter une prise de conclusion fausse positive dans chaque test.

La correction de Benjamini-Hochberg est une des premières méthodes développées et souvent utilisées dans la littérature. La procédure de la correction de Benjamini-Hochberg est présentée dans l'Algorithme 4.3.

Pour chaque hypothèse H_i, l'algorithme calcule la p-*value* correspondant p_i. k est le nombre d'hypothèses nulles simultanément testées. Puis, l'algorithme ordonne les p-*values* $p_1, ..., p_k$ et les hypothèse correspondantes $H_1, ..., H_k$ de manière ascendante. Pour un FDR souhaité, l'algorithme compare la p-value ordonnée p_i à la valeur du seuil de signification corrigé localement $\frac{\alpha * i}{k}$ pour rejeter H_i.

Algorithme 4.3 La correction Benjamini-Hochberg

ENTRÉES: Une liste de *p-value* $P = \{p_1, ..., p_k\}$, k est le nombre de tests et α est la seuil de signification.
SORTIES: Une liste des indices des hypothèses nulles rejetées.

1: $listIndex \leftarrow \emptyset$;
2: $P' \leftarrow order(P, ASC)$;
3: $Index \leftarrow getIndex(P', P)$;
4: **pour** $i = 1$ to k **faire**
5: **si** $P'[i] \leq \alpha * i/k$ **alors**
6: $listIndex[i] \leftarrow Index[i]$;
7: **finsi**
8: **fin pour**
9: **retourner** $listIndex$;

Notations :
$order(P, ASC)$: fonction qui retourne une liste des *p-value* ordonnées ascendantes, P ;
$getIndex(P', P)$: fonction qui retourne des indices des tests selon la liste ordonnée des *p-value*, P' ;

TABLE 4.3 – L'algorithme de correction Benjamini-Hochberg [Abdi 2007].

4.2.5 Réduction de nombre de tests

Dans les tests multiples, le nombre de tests est souvent important. Pour cela, nous devons non seulement corriger le seuil de signification α, mais aussi diminuer le nombre de tests. Une approche naïve qui peut être appliquée est de ne tester que les arcs trouvés dans au moins un des deux ensembles de graphes. Cela signifie que nous ne testons pas les $\frac{n(n-1)}{2}$ arêtes possibles, où n est le nombre de variables. Dans cette section, nous présentons une autre approche qui peut être utilisée pour réduire le nombre de tests. En effet, dans une section précédente (voir la Section 3.3), nous avons proposé un nouvel objet nommé QEG (Quasi-Essential Graph), *le graphe quasi-essentiel* pour résumer statistiquement chaque ensemble de RB dans un graphe représentant unique et ensuite nous avons utilisé cet objet pour éliminer des arcs bruyants qui ont une faible probabilité d'occurrence. Cela peut aider à réduire le nombre de tests pour les tests multiples sur les arcs du graphe.

Etant donné un ensemble de RB $\mathcal{B}$ et un seuil β, le QEG Q est un représentant de $\mathcal{B}$ sii : (1) Q a le même ensemble de variables par rapport aux tous les RB ; (2) la probabilité d'occurrence dans $\mathcal{B}$ de chaque arête (non-dirigé) de

Q est supérieure à β ; (3) la probabilité d'occurrence dans $EG(\mathcal{B})$ de chaque arc (dirigé) de Q est supérieure à β ;

Notre objectif est de trouver les arcs les plus pertinents en utilisant le QEG représentant de chaque ensemble de RB. Dans ce travail, nous nous concentrons sur les arêtes et mettons les arcs dans le cadre de travaux en perspective. Nous construisons d'abord une union U des squelettes de deux QEG Q_1 et Q_2 ; Ensuite, pour chaque arête non orienté e_i dans U, nous comparons son poids dans le squelette de Q_1 et Q_2 en calculant $D_i = w_u^i(Q_1, e_i) - w_u^i(Q_2, e_i)$. Si $\{D_i\} \neq 0$, e_i est marqué comme un arc "pertinent".

Après avoir éliminé des arêtes inutiles, nous pouvons appliquer un test multiple (cf. Section 4.2.3) pour toutes les arêtes pertinentes des graphes essentiels des deux ensembles de RB.

4.3 Expérimentation

4.3.1 Génération de données expérimentales

L'étude expérimentale a été conçue sur des RB simulés. Soit $S_1 = \{G_1^i\}$ et $S_2 = \{G_2^i\}$ deux populations de graphes selon une loi $Poisson(\Lambda)$ (cf. Table 4.4) et $n = Poisson(\Lambda)$, la procédure de génération de S_1 et S_2 consiste en quatre étapes suivantes :

1. Générer aléatoirement G_1^0
2. Construire S_1 en applicant n ajouts/suppressions d'arcs sur G_1^0
3. Générer G_2^0 en applicant n ajouts/suppressions d'arcs sur G_1^0
4. Construire S_2 en applicant n ajouts/suppressions d'arcs sur G_2^0

Afin d'assurer une différence significative entre les deux ensembles de graphes, nous avons appliqué l'opération "ajout d'arc" à partir d'une liste d'arcs "fixés". Cette liste peut être générée de façon aléatoire en prenant un nombre fixe d'arcs ($nRndE$) qui exclurait les arcs présentés du graphe initial. La Table 4.4 décrit l'ensemble des paramètres de la procédure de génération de DAG.

Dans ce travail, nous avons choisi $nG = 100$, $nV = 100$, $nE = 100$, $nRndE = \{20, 50, 100\}$, $\Delta = \{7, 15\}$ et $\Lambda = 5$.

Nom	Définition
nG	Nombre de graphes par population
nV	Nombre de noeuds par graphe
nE	Nombre d'arcs par graphe
$nRndE$	Nombre d'arcs aléatoires pour les opérations d'ajout
Δ	Moyenne de la distribution Poisson utilisée pour générer aléatoirement le nombre de différences entre deux graphes initiaux de deux populations par les opérations d'ajout et de suppression
Λ	Moyenne de la distribution Poisson utilisée pour générer aléatoirement le nombre de différences entre le graphe initial et les autres graphes dans une population par les opérations d'ajout et de suppression

TABLE 4.4 – Liste de paramètres pour la procédure de génération de données.

Il est important de noter que nous essayons de fixer certains paramètres. Par exemple nG, nV, nE or Λ, $nRndE$ et Δ jouent un rôle important pour générer les différences entre les deux ensembles de graphes. Cela permet aussi de contrôler le taux d'erreurs commises dans les résultats de tests (cf. Section 4.4).

4.3.2 Méthodes d'implémentation

Pour faire l'expérimentation, nous avons utilisé des méthodes suivantes :

- Test multiple : Fisher-exact
- Correction de seuil de signification : Bonferroni, Benjamini-Hochberg
- Réduction du nombre de tests : QEG

Ces méthodes ont été codées en C++ avec l'aide de la bibliothèque Boost [1] et des API fournies par la plate-forme ProBT [2].

4.4 Résultats

Le Table 4.5 décrit le résultat de chaque procédure de tests dans plusieurs contextes expérimentaux (sans correction, correction de Bonferroni ou correction de Benjamini-Horchberg, et *avec* ou *sans* filtrage QEG).

Nous pouvons observer le comportement général suivant : toutes les approches peuvent détecter les différences réelles entre les deux graphes initiaux

1. `http://www.boost.org`
2. `http://bayesian-programming.org`

(tp) mais diffèrent dans leurs erreurs. L'approche "sans correction" a un très haut taux de faux positifs. Cette valeur diminue d'abord avec l'approche de correction de Benjamini-Horchberg, BH et ensuite avec l'approche de Bonferroni, BON qui est la procédure la plus conservatrice . L'utilisation de filtrage QEG diminue clairement le taux de faux positifs quelque soit la méthode de correction appliquée.

Dans un travail récent, [Dudoit 2003] a prouvé que les procédures basées sur FDR présentent une alternative prometteuse par rapport aux approches qui contrôlent le FWER. Ce travail a aussi constaté que les méthodes basées sur FDR sont également les approches les plus utilisées dans l'application réelle où des milliers de tests sont effectués simultanément, comme la différenciation de l'expression des gènes, etc... Pourtant, dans nos jeux de RB simulés, les résultats expérimentaux montrent que la correction de Bonferroni (basée sur FWER) est plus performante que la correction de Benjamini-Horchberg (basée sur FDR) au niveau du taux d'erreurs commises (cf. Table 4.5). Ceci est un résultat normal parce qu'il y a beaucoup des différences réelles entre les deux échantillons. Ainsi, il est facile de rejeter une hypothèse nulle quelle que soit la méthode de correction appliquée. En effet, comme la génération de données n'est pas complètement aléatoire (c'est notre choix, voir la Section 4.3.1), si une arête a été choisie pour générer la différence entre les ensembles, ses fréquences dans les deux ensembles de graphes sont vraiment différentes. C'est pourquoi la p-*value* pour test sur cette arête est vraiment petite et donc, le taux de rejet de l'hypothèse nulle est effectivement élevé.

Les résultats expérimentaux montrent également que la correction de Bonferroni accepte plus fréquemment l'hypothèse nulle (aucune différence) pour les arêtes avec une petite différence de fréquence entre les ensembles ($< \frac{20}{100}$, cf. Figure 4.4) par rapport à la correction de Benjamini-Horchberg. Le tableau 4.5 montre que pour 8 expériences, nous avons besoin d'appliquer seulement 836 tests en tout au lieu de 1534. Par ailleurs, les tests ne commettent presque aucune erreur. Nous voyons que la correction de Benjamini-Horchberg semble donc plus intéressant ici que la correction de Bonferroni.

EXP	Réel	sans QEG												N
		sans correction				BH				BON				
		tp	tn	fp	fn	tp	tn	fp	fn	tp	tn	fp	fn	
100-15	17	17	218	32	0	17	247	3	0	17	250	0	0	267
50-15	18	18	204	45	0	19	94	26	0	19	104	16	0	139
20-15	19	19	91	29	0	6	172	14	0	6	185	1	0	192
100-7	5	5	243	24	0	18	226	23	0	18	249	0	0	267
50-7	6	6	143	43	0	5	267	0	0	5	267	0	0	272
20-7	9	9	94	36	0	9	96	34	0	9	108	22	0	139
20-3	5	5	95	39	0	5	99	35	0	5	112	22	0	139
10-3	1	1	98	20	0	1	99	19	0	1	100	18	0	119
Total	80	80	1186	268	0	80	1300	154	0	80	1375	79	0	1534

(a) Résultats sans QEG

EXP	Réel	Avec QEG												N
		sans correction				BH				BON				
		tp	tn	fp	fn	tp	tn	fp	fn	tp	tn	fp	fn	
100-15	17	17	89	2	0	17	91	0	0	17	91	0	0	108
50-15	18	18	116	5	0	19	89	1	0	19	90	0	0	109
20-15	19	19	87	3	0	6	97	0	0	6	97	0	0	103
100-7	5	5	95	2	0	18	90	0	0	18	90	0	0	108
50-7	6	6	94	3	0	5	97	0	0	5	97	0	0	102
20-7	9	9	93	2	0	9	95	0	0	9	95	0	0	104
20-3	5	5	93	4	0	5	97	0	0	5	97	0	0	102
10-3	1	1	98	1	0	1	99	0	0	1	99	0	0	100
Total	80	80	1186	268	0	80	1300	154	0	80	1375	79	0	836

(a) Résultat avec QEG

TABLE 4.5 – Les résultats de tests multiples. **EXP** : Contexte experimental ($nRndE, \triangle$) ; **Réel** : Le nombre de différences réelles entre le graphe initial de chaque population ; **BH** : La correction Benjamini-Horchberg ; **BON** : La correction Bonferroni ; **tp** : True Positive (rejette H_0 si une arête existe dans un graphe initial mais pas dans l'autre) ; **tn** : True Negative (accepte H_0 si une arête existe dans les deux graphes initiaux) ; **fp** : (rejette H_0 si une arête existe dans un graphe initial mais pas dans l'autre) ; **fn** : (accepte H_0 si une arête existe dans un graphe initial mais pas dans l'autre) ; **N** : nombre de tests.

4.5 Conclusion

Nous avons présenté dans ce chapitre une architecture permettant d'affectuer une étude différentielle de deux ensembles de réseaux bayésiens, représentant deux contextes différents. Cette solution s'avève intéressante lorsque le nombre de données est faible et ne permet pas forcément d'estimer un seul bon modèle par contexte.

Notre approche effectue un test multiple sur les arêtes, permettant ainsi de localiser les différences.

L'utilisation d'une correction du seuil de signification permet de garantir

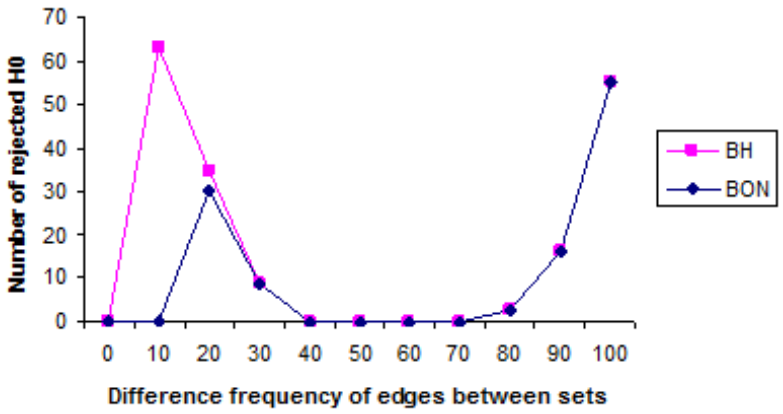


FIGURE 4.4 – La comparaison entre la correction Bonferroni vs. la correction Benjamini-Horchberg sur la capacité de rejeter des hypothèses nulles selon la différence de fréquence d'arêtes entre deux populations.

un contrôle sur les erreurs de Type I.

L'utilisation du graphe quasi essentiel (QEG) permet aussi de réduire le nombre de tests à effectuer.

Chapitre

5

Application à l'étude différentielle de réseaux de régulation génétique

Sommaire

5.1 Contexte et problématique

Nous présentons dans ce chapitre les différents tests que nous avons effectués afin de valider nos propositions proposées. Nous nous plaçons dans le cadre d'une étude différentielle sur le changement de régulation génétique entre des gènes, en utilisant le réseau "benchmark" INSULIN proposé par [Philip P. 2004].

Pour ce faire, nous avons examiné la régulation génétique dans deux contextes de réseaux différents. Le premier est celui du réseau INSULIN. Le deuxième est générés à partir du réseau INSULIN en modifiant certaines relations entre des gènes. L'étude différentielle doit nous permettre de vérifier deux objectifs :

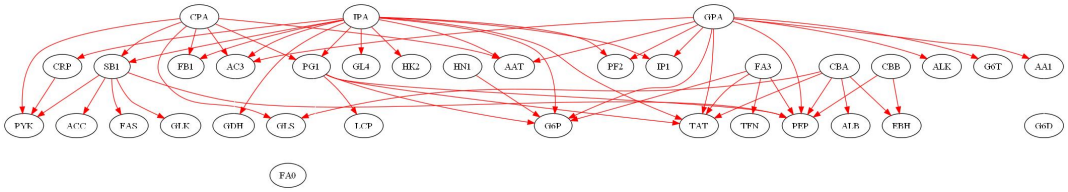


FIGURE 5.1 – Le réseau INSULIN proposé par [Philip P. 2004]

- *vérifier s'il y a des différences entre deux contextes de réseaux*
- *et localiser où sont des différences*

5.2 Protocole expérimental

5.2.1 Réseau INSULIN

Le réseau INSULIN est un réseau de régulation génétique proposé par [Philip P. 2004] (cf. Figure 5.1 et Annexe 3). Il comporte 35 noeuds, figurant soit des gènes, soit les voies de signalisation relatives à l'insuline (IPA), le glucagon (CPA) et les glucocorticoïdes (GPA). Les noeuds racines autres qu'IPA, CPA et GPA sont des facteurs de transcription connus pour jouer un rôle important dans la régulation de la glycémie. Ce réseau permet de contrôler le métabolisme du glucose dans les hépatocytes périnatale en observant les effets de différents facteurs. Le RB INSULIN et les tables de probabilités sont accessibles via les ressources supplémentaires de [Philip P. 2004].

Ce réseau est représentatif de la problématique applicative qui nous intéresse, et est aussi utilisé comme réseau de référence pour évaluer les algorithmes d'apprentissage de la structure des RB.

5.2.2 Simulation de données

Dans le cadre d'une étude différentielle, il nous faut des jeux de données représentatifs de deux contextes différents où le nombre de données est faible. Pour ce faire, nous utilisons le réseau INSULIN qui permet de générer des échantillons synthétiques de cellules "normales". A cette condition de référence, nous avons ajouté un autre modèle, relatif à une condition dite "pathologique", liée à une défaillance du mécanisme de régulation. Plus particulièrement, nous avons considéré que la voie de signalisation de l'insuline

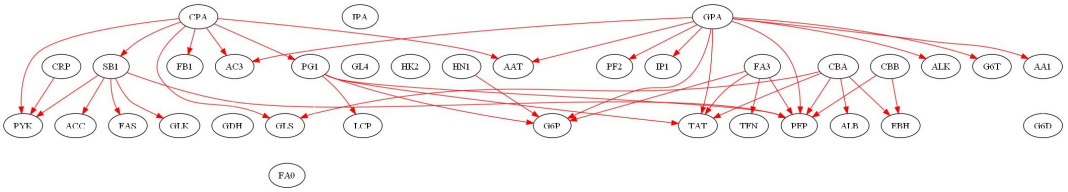


FIGURE 5.2 – *Le réseau INSULIN* généré à partir du réseau INSULIN en supprimant rôle du gène TPA.*

est défaillante, sans décrire l'origine biologique de ce dysfonctionnement. Ce nouveau réseau, nommé INSULIN* (cf. Figure 5.2), est le même qu'INSULIN, à deux différences près :

- le noeud IPA n'est plus capable d'exercer son influence sur les noeuds fils qu'ils possèdent dans le réseau de référence INSULIN,
- par conséquence, les noeuds fils ne subissent plus que l'influence des autres parents si ceux-ci existent, ou sont désormais des noeuds racines.

Chacun de ces réseaux nous permet de générer des bases de données de 200 échantillons.

5.2.3 Méthodes utilisées

5.2.3.1 Apprentissage ensembliste

Pour chaque contexte, nous avons utilisé l'algorithme d'apprentissage par bootstrap (cf. Section 3.1.1) de 100 réplications appliqué à la recherche glouton. Pour fusionner l'ensemble de RB obtenus par bootstrap, nous avons utilisé l'approche QEG (cf. Section 3.3) avec un seuil de 0.5.

5.2.3.2 Etude différentielle

Pour l'étude différentielle des ensembles de modèles obtenus à partir des deux contextes, nous avons utilisé notre approche proposée dans la Section 4.2. Nous avons choisit le test de Fisher exact sur les tests locaux et la correction Benjamini-Hochberg pour corriger le seuil de signification. Nous avons étudié l'influence de l'utilisation du QEG pour réduire le nombre de tests.

Tous ces algorithmes ont été implémentés sous C++ avec la bibliothèque

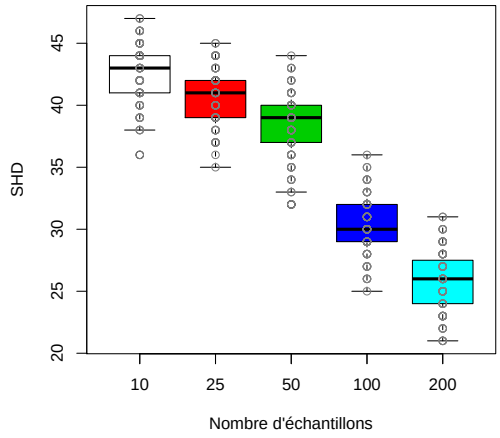


FIGURE 5.3 – *L'évolution de des distances SHD des RB obtenu par Bootstrap où le nombre d'échantillons dans chaque jeu est variable et le nombre de réplications est fixé à 100.*

Boost [1], des API fournis par la plateforme ProBT [2] et la visualisation est réalisée avec Graphviz [3].

5.2.4 Critères d'évaluation

Pour évaluer la qualité de résultats obtenu par l'algorithme d'apprentissage, nous avons utilisé la méthode d'évaluation basée sur la distance SHD (cf. Section 2.3.4) et la visualisation du QEG en comparant avec le graphe essentiel du graphe théorique (cf. Section 3.3.5).

Pour valider le résultat obtenu par l'étude différentielle, nous avons utiliser la liste des arcs différents entre INSULIN et INSULIN* et vérifier si nous retrouvons bien ces différences sans commettre trop d'erreurs.

5.3 Résultats et interprétation

5.3.1 Qualité de l'apprentissage ensembliste

En applicant la mesure de qualité d'un algorithme d'apprentissage par la distance d'édition SHD, le résultat obtenu montre que la qualité de l'apprentis-

1. `http://www.boost.org`
2. `http://bayesian-programming.org`
3. `http://graphiz.org`

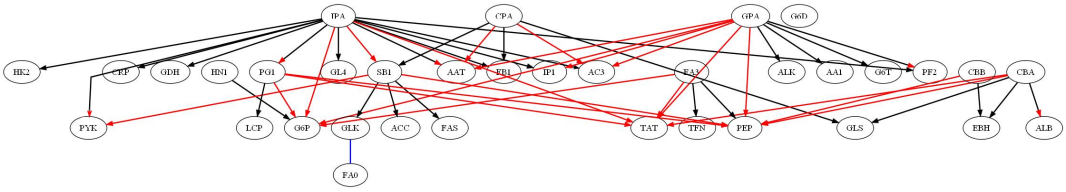
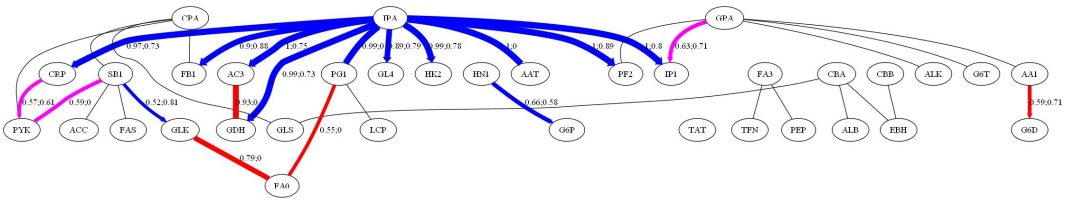


FIGURE 5.4 – *Le graphe union de graphe essentiel du graphe théorique et du QEG obtenu par l'apprentissage de la structure avec Bootstrap dans le contexte du réseau INSULIN.*

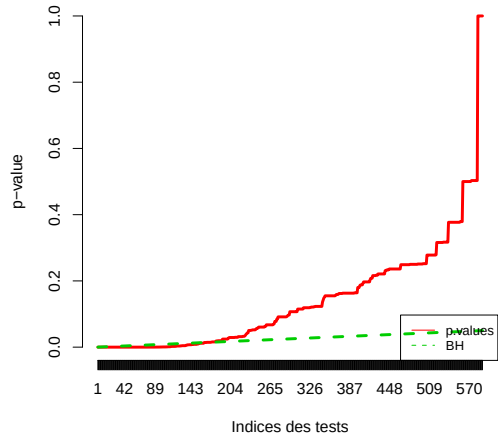


TP	TN	FP	FN	SEN	SPE	ACC	FDR	PPV
16	559	12	8	66.7%	97.9%	96.6%	42.9%	57.1%

FIGURE 5.5 – Le graphe union obtenu dans le contexte 1 et 2. *Les arcs en noir sont des arcs présents dans le QEG obtenu de deux contextes. Les arcs en bleu sont des arcs présents dans le QEG obtenu dans le contexte 1 mais absents dans le QEG obtenu dans le contexte 2. Les arcs en rouge sont des arcs présents dans le QEG obtenu dans le contexte 2 mais absents dans le QEG obtenu dans le contexte 1. Les arcs en violet sont des arcs présents à la fois dans le QEG obtenu dans le contexte 2, le graphe INSULIN et le graphe INSULIN*, mais absent dans le QEG obtenu dans le contexte 1.* SEN *: Sensibility ;* SPE *: Specificity ;* ACC *: Accuracy ;* FDR *: False Discovery Rate.*

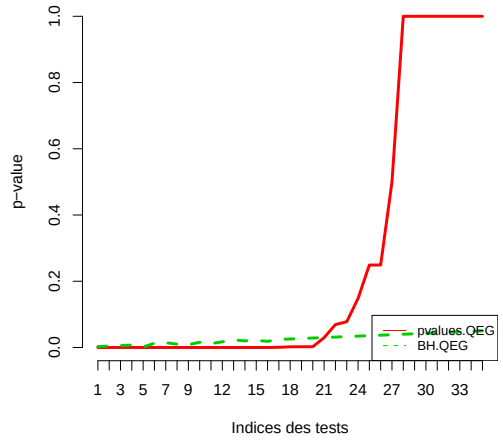
sage de la structure de réseaux bayésiens par bootstrap augmente en fonction de nombre d'échantillons dans chaque jeu de données (cf. Figure 5.3).

La Figure 5.5 montre que l'apprentissage dans le contexte INSULIN* a réussi à détecter les arcs supprimés au tour du gène TPA du graphe INSULIN. Cependant, il trouve 8 nouveaux arcs qui ne sont pas présents dans le QEG du contexte INSULIN : $AC3-GDH$, $CPA-FB1$, $GLK-FA0$, $PG1-FA0$, $AA1-G6D$, $GPA-IP1$, $CRP-PYK$, $SB1-PYK$. Parmi ces 7 arcs, il y a 3 arcs qui sont présents dans le graphe INSULIN. Il s'agit de arcs suivants : $GPA-IP1$, $CRP-PYK$, $SB1-PYK$. Cela veut dire que l'apprentissage dans INSULIN* arrive à retrouver des arcs que celui dans INSULIN n'arrive pas à retrouver.



Nombre de tests	H_0 rejecté	H_0 accepté	Taux de H_0 rejetés
590	167	432	28.3%

FIGURE 5.6 – *Le résultat du test multiple sur contexte 1 & 2 avec la correction Benjamini-Horchberg sans QEG.*



Nombre de tests	H_0 rejecté	H_0 accepté	Taux de H_0 rejetés
35	20	15	0.571

FIGURE 5.7 – *Ligne de décision du test multiple sur contexte 1 et 2 avec la correction Benjamini-Horchberg avec QEG.*

5.3.2 Qualité de l'étude différentielle

La Table 5.1 représente la liste des arêtes testées par le test multiple avec QEG et le résultat de décision de chaque test. Parmi 13 arêtes supprimés au

Arête	p-values	BH	Décision
IPA - GDH	1.12E-57	0.001428571	rejeté
IPA - IP1	1.1E-59	0.002857143	rejeté
IPA - PF2	1.1E-59	0.004285714	rejeté
IPA - AC3	1.1E-59	0.005714286	rejeté
AAT - IPA	1.1E-59	0.007142857	rejeté
AC3 - GDH	2.88E-49	0.008571429	rejeté
IPA - CRP	1.95E-54	0.01	rejeté
IPA - GL4	5.23E-45	0.011428571	rejeté
IPA - PG1	1.12E-57	0.012857143	rejeté
IPA - HK2	1.12E-57	0.014285714	rejeté
IPA - FB1	5.18E-46	0.015714286	rejeté
GLK - FA0	1.87E-36	0.017142857	rejeté
PG1 - FA0	7.96E-22	0.018571429	rejeté
AA1 - G6D	6.71E-24	0.02	rejeté
CRP - PYK	7.55E-23	0.021428571	rejeté
GPA - IP1	4.31E-26	0.022857143	rejeté
HN1 - G6P	0.000548	0.024285714	rejeté
CPA - SB1	0.00209	0.025714286	rejeté
CPA - FB1	0.0022	0.027142857	rejeté
CBA - EBH	0.00232	0.028571429	rejeté
PG1 - LCP	0.0297	0.03	accepté
SB1 - GLK	0.0692	0.031428571	accepté
PF2 - GPA	0.0777	0.032857143	accepté
FA3 - PEP	0.148	0.034285714	accepté
CPA - GLS	0.249	0.035714286	accepté
GPA - G6T	0.249	0.037142857	accepté
CBA - GLS	0.5	0.038571429	accepté
GPA - AA1	1	0.04	accepté
GPA - ALK	1	0.041428571	accepté
CBB - EBH	1	0.042857143	accepté
FA3 - TFN	1	0.044285714	accepté
SB1 - ACC	1	0.045714286	accepté
SB1 - FAS	1	0.047142857	accepté
ALB - CBA	1	0.048571429	accepté
PYK - CPA	1	0.05	accepté

TABLE 5.1 – Résultat de l'étude différentielle, avec l'utilisation du QEG pour limiter le nombre de tests. Arête *: Listes des arêtes ;* p-values *: p-values obtenus par le test multiple ;* BH *: les seuils de signification corrigés avec la méthode de Benjamini-Horchberg.* ***Décision*** *: la décision accepter/rejecter l'hypothèse nulle.*

tour du gène TPA du graphe INSULIN pour obtenir le réseau INSULIN*, l'étude différentielle a réuissi à détecter 10 arêtes (en gras dans la Table 5.1) comme des différences entre les QEG obtenus dans les contextes INSULIN et INSULIN*. C'est-à-dire, le test multiple rejette tous les arcs présents dans un QEG mais pas dans l'autre et accepte les arcs présents à la fois dans les deux QEG. Cependant, il y a 3 arêtes $CPA - SB1$, $CPA - FB1$ et $CBA - EBH$ qui ne vérifient pas ce principe. Une arête peut être présente dans les deux

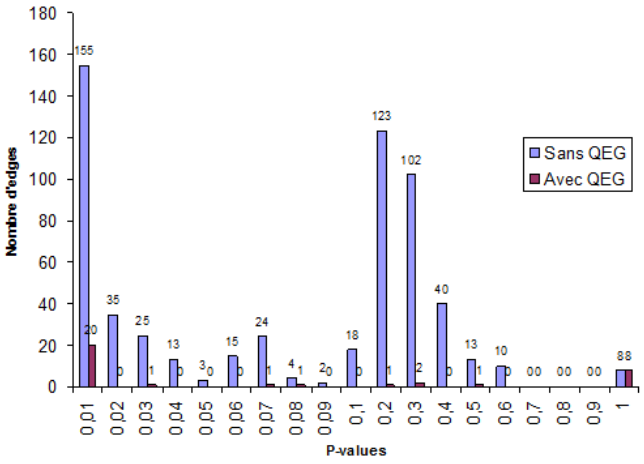


FIGURE 5.8 – *Histogram des p-values obtenu avec le test multiple sur contexte 1 et 2 dans les deux approches de correction Benjamini-Horchberg sans et avec QEG.*

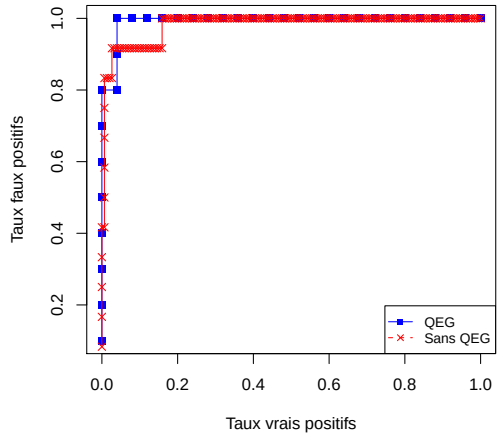


FIGURE 5.9 – *La courbe ROC du test multiple avec QEG et sans QEG.*

QEG mais ses fréquences dans deux contextes sont suffisamment différentes pour que le test décide que cette différence est significative.

L'histogramme de la Figure 5.8 compare le nombre de tests entre l'approche sans QEG et avec QEG. Dans l'approche sans QEG, il y a beaucoup plus de tests avec p-value qui est supérieure au seuil de signification maximum 0.05 par rapport à celle avec QEG. Cela veut dire que l'approche sans QEG

risque d'effectuer plus de test sur les arêtes unutiles par rapport à l'approche avec QEG.

La courbe ROC du test multiple avec QEG et sans QEG présentée dans la Figure 5.9 montre que le test multiple arrive à découvrir les différences réelles entre deux contextes avant de commettre quelques erreurs et l'approche avec le QEG semble plus intéressant par rapport celle sans QEG.

5.4 Conclusion

Nous avons présenté dans ce chapitre différents expérimentations sur le réseau de régulation génétique synthétique INSULIN afin de valider (i) la méthode basé sur le QEG qui fusionne un ensemble de réseaux bayésiens issus de d'apprentissage ensembliste ; (ii) l'architecture générale pour l'étude différentielle des ensembles de graphes.

Les résultats obtenus montrent l'intérêt du QEG dans la visualisation d'un ensemble de graphes et dans la réduction de l'étude différentielle des ensembles de graphes. L'étude différentielle basé sur un test multiple nous permet de détecter des différences significatives dans deux contextes d'apprentissage avec des données de tailles réduites et aussi de localiser ces différences.

Conclusions et Perspectives

Nous avons présenté dans le *Chapitre 1* les différentes approches pour la modélisation de réseaux de régulation génétique. Quelles que soient les approches utilisées pour reconstruire un réseau de régulation génétique ? partir de puces à ADN, la difficulté principale est le nombre limité de données disponibles. Les réseaux bayésiens dégagent un fort intérêt dans la reconstruction de réseaux de régulation génétique.

Le *Chapitre 2* a permis de montrer qu'il existe plusieurs méthodes d'apprentissage de la structure de réseaux bayésiens et des méthodes d'évaluation variées de la qualité de ces algorithmes. Une caractéristique importante à prendre en compte pour toutes ces méthodes est l'équivalence de Markov.

Parmi les méthodes d'apprentissage les plus utilisées, nous avons présenté dans *le Chapitre 3* des méthodes ensemblistes qui permettent d'obtenir un ensemble de meilleures structures. C'est cet intérêt important qui est l'inspiration de ce travail. Cette méthode d'apprentissage permet de remédier au problème de la pauvreté des données. Pour utiliser cette approche, il faut être capable d'évaluer, de comparer et de visualiser un ensemble de réseaux bayésiens. A notre connaissance, les méthodes existantes dans la littérature permettent de fusionner un ensemble de réseaux bayésiens, mais elles ne permettent d'évaluer et de comparer deux ensembles de réseaux bayésiens. C'est la raison pour laquelle nous avons proposé un nouvel objet, qui s'appelle QEG (Quasi-Essential Graph), qui permet de représenter et de visualiser un ensemble de réseaux bayésiens. Le QEG généralise le graphe essentiel, représentant de tous les graphes d'une classe d'équivalence de Markov. Il a en plus des poids spécifiques pour chaque arête et chaque orientation d'un arc. Par consé-

quent, le QEG peut être visualisé avec une métaphore graphique qui rend plus lisible les propriétés d'un représentant d'une population de réseaux bayésiens.

Pour comparer deux ensembles de réseaux bayésiens, dans *le Chapitre 4*, nous avons proposé d'utiliser un test multiple qui permet non seulement de vérifier si la différence est significative mais également de localiser où sont les différences. Nous avons montré que l'utilisation d'une méthode de correction du seuil de signification permet de contrôler les erreurs de type I. Nous avons aussi mis en avant que l'utilisation du QEG permet de réduire le nombre de tests réalisés.

Pour analyser et valider les mérites des méthodes retenues et proposées, nous avons fait des expérimentations sur un benchmark spécifique classiquement utilisé dans l'évaluation des méthodes de reconstruction de réseaux de régulation génétique. Les résultats obtenus ont permis de montrer l'intérêt des méthodes d'apprentissage ensemblistes de la structure de réseaux bayésiens et de l'architecture proposée pour l'étude différentielle dans le cas de la pauvreté de données.

A l'issue de ce travail, toutes les propriétés théoriques et les résultats expérimentaux devraient être étendues. Nous avons identifié plusieurs pistes perspectives :

1. Proposition des mesures d'évaluation d'un QEG : adaptation de la distance SHD, de la divergence KL par exemple, pour estimer directement la qualité d'un QEG donné.

2. Amélioration de la métaphore graphique permettant de comparer le QEG et le graphe théorique, pour essayer de prendre en compte les poids du QEG.

3. Utilisation d'autres cadres théoriques (théorie des possibilités [Dempster 1967, Shafer 1976]) pour définir le graphe représentant d'un ensemble de DAG.

4. Etude théorique et expérimentale sur l'intérêt de combiner AG avec *niching* [Pelikan 2001, Auliac 2007] pour garantir variabilité des solutions,

et Bootstrap, pour améliorer la robustesse des méthodes ensemblistes, lorsqu'il y a peu de données.

5. Prise en compte de l'orientation des arcs dans l'étude différentielle, puisque notre solution actuelle compare uniquement les fréquences d'apparition des arêtes.
6. Application de l'architecture proposée à des données réelles de régulation génétique.
7. Extension de l'approche générale à d'autres domaines d'application par exemple à l'étude différentielle des réseaux sociaux grâce à l'observation de leur comportement.

ANNEXES

Annexe 1 : Type d'erreur d'un test statistique

Le risque de première et deuxième espèce sont définis comme suivants :

- Le risque de première espèce α indique la probabilité de rejeter H_0 dans le cas ou H_0 est vrai (cf. Table A1.1).
- Le risque de deuxième espèce β indique la probabilité d'accepter H_0 dans le cas ou H_0 est fausse (cf. Table A1.1) :

Décision/Vérité	H_0 est vraie	H_0 est fausse
accepter H_0	$1-\alpha$	β
rejeter H_0	α	$1-\beta$

TABLE A1.1 – Table de décision du test statistique : *α est souvent appelé "le seuil de la signification du test, p-value". α est souvent pris dans l'intervalle [0.001, 0.05]; $1-\beta$ est souvent appelé "la puissance du test".*

Annexe 2 : Score

Nom	Formule	Cat.	Car.
LL [Anderson 1970]	$LL(\mathcal{B},\mathcal{D}) = \sum_{i=1}^{n}\sum_{j=1}^{q_i}\sum_{k=1}^{r_i} N_{ijk} log(\frac{N_{ijk}}{N_{ij}})$	I	D, E
AIC [Akaike 1970]	$AIC(\mathcal{B},\mathcal{D}) = LL(\mathcal{B},\mathcal{D}) - Dim(\mathcal{B})$	I	D, E
BIC [Schwarz 1978]	$BIC(\mathcal{B},\mathcal{D}) = LL(\mathcal{B},\mathcal{D}) - \frac{1}{2} log(N) Dim(\mathcal{B}) - 1) q_i$	I	D, E
MDL [Rissanen 1978]	$MDL(\mathcal{B},\mathcal{D}) = LL(\mathcal{B},\mathcal{D}) - \frac{1}{2} log(N) Dim(\mathcal{B}) + O(1)$	I	D, E
NML [Rissanen 1978]	$NML(\mathcal{B},\mathcal{D}) = LL(\mathcal{B},\mathcal{D}) - C(\mathcal{B})$	I	D, E
$K2$ [Cooper 1992]	$K2(\mathcal{B},\mathcal{D}) = log(P(\mathcal{B})) + \sum_{i=1}^{n}\sum_{j=1}^{q_i}(log(\frac{(r_i-1)!}{(N_{ij}+r_i-1)}) + \sum_{k=1}^{r_i} log(N_{ijk}!))$	B	D
BD [Heckerman 1995]	$BD(\mathcal{B},\mathcal{D}) = log(P(\mathcal{B})) + \sum_{i=1}^{n}\sum_{j=1}^{q_i}(log(\frac{\Gamma(\alpha_{ij})}{\Gamma(N_{ij}+\alpha_{ij})}) + \sum_{k=1}^{r_i} log(\frac{\Gamma(N_{ijk}+\alpha_{ijk})}{\Gamma(\alpha_{ijk})}))$	B	D
$BDeu$ [Buntine 1991]	$BDeu(\mathcal{B},\mathcal{D}) = log(P(\mathcal{B})) + \sum_{i=1}^{n}\sum_{j=1}^{q_i}(log(\frac{\Gamma(\frac{\alpha_{ij}}{q_i})}{\Gamma(N_{ij}+\frac{\alpha_{ij}}{q_i})}) + \sum_{k=1}^{r_i} log(\frac{\Gamma(N_{ijk}+\frac{\alpha_{ijk}}{r_i q_i})}{\Gamma(\frac{\alpha_{ijk}}{r_i q_i})}))$	B	D, E

TABLE A1.2 – Classification des principaux scores. (*Cat.*) Catégories : **I** : *Les scores basés sur la théorie de l'information*, **B** : *Les scores basés sur l'approche Bayésienne*. (*Car.*) Caractères : **D** : *Décomposable*, **E** : *Equivalent*

La Table A1.2 présente une liste de scores bien connus dans la littérature. Il est important de noter que MDL et BIC sont distingués par le terme O(1). $O(1) \to 0$ si $N \to \infty$ [Grünwald 2007, Foster 2004]. Par ailleurs, ce terme est décrit plus concrètement comme la longueur de codage de données pour construire le RB [Nicandro 2006].En terme des composants du score, apparemment, la fonction généralisé de score Bayésien BD présente implicitement la complexité du modèle. Cependant, Borgelt et al. [Borgelt 2002] ont démontré que ce score prend en compte également la tendance à sélectionner des structures simples. K2 est un cas particulier de BD avec $\alpha_{ijk} = 1$ et $\Gamma(c) = (c-1)!$ où c est un nombre entier. DBe est un score BD où N' est le nombre d'exemples "équivalent". BDe est appelé BDeu en utilisant α_{ijk} défini

par : $\alpha_{ijk} = \frac{N'}{r_i q_i}$. Au niveau des caractères du score, jusqu'à présent, tous les scores présentés sont décomposables. Les scores LL, AIC, BIC, MDL, NML, BDe, BDeu sont des scores équivalents. Le reste des scores comme K2, BD ne sont pas des scores équivalents. En terme de performance, pour une donnée de petite taille, les méthodes de la famille NML présente une performance prometteuse [Carvalho 2009]. Le score K2 donne le meilleur résultat par rapport aux autres scores [Carvalho 2009] dans le contexte de données de grandes tailles.

Annexe 3 : Réseau INSULIN

Le réseau de régulation modélise l'homéostasie du glucose dans les cellules du foie (hépatocytes). La régulation de la glycémie joue un rôle capital dans l'organisme pour apporter l'énergie nécessaire aux cellules. Ce contrôle du glucose est effectué par un processus complexe impliquant plusieurs organes (foie, pancréas, rein essentiellement) ainsi que des mécanismes de signalisation. Le foie agit sur ce système en stockant le glucose sous la forme de glycogène quand le taux de glucose est élevé et en le libérant quand le taux de glucose est faible (en période de jeune par exemple). Cette action est coordonnée par des hormones spécifiques : principalement l'insuline, le glucagon et les glucocorticoïdes. L'insuline, secrétée par le pancréas, a un effet hypoglycémiant (elle assure une baisse du taux de glucose dans le sang). Le glucagon, également secrété par le pancréas, a une action inverse (hormone hyperglycémiante) tandis que les glucocorticoïdes sont des hormones stéroïdiennes sécrétées par le cortex de la glande surrénale. L'action des glucocorticoïdes est également antagoniste à celle de l'insuline. L'insuline est donc indispensable à la régulation de la glycémie : c'est la seule substance capable de réduire le glucose.

Le réseau INSULIN proposé par [Philip P. 2004] comporte 35 noeuds, soit des gènes, soit les voies de signalisation relatives à l'insuline (IPA), le glucagon (CPA) et les glucocorticoïdes (GPA). Les noeuds racines autres qu'IPA, CPA et GPA sont des facteurs de transcription connus pour jouer un rôle dans la régulation de la glycémie. Le réseau a été constitué à partir d'une étude bibliographie étendue sur le domaine :

- la topologie du réseau décrit au total 52 interactions. Un arc dirigé d'un noeud parent vers un noeud fils a été inféré à partir d'un corpus d'articles,
- les tables de probabilités conditionnelles ont été également définies à partir de la littérature, selon le rôle plus ou moins activateur ou inhibiteur des gènes et des voies de signalisation. Ces tables déterminent la probabilité qu'un noeud fils soit en mode 'activé' ou non en fonction des

niveaux binaires d'expression des noeuds parents.

Bibliographie

[Abdi 2007] H. Abdi. *Bonferroni and Sidak corrections for multiple comparisons.* Encyclopedia of measurement and statistics, vol. 1, pages 103–107, 2007.

[Akaike 1970] H. Akaike. *Statistical predictor identification.* Ann. Inst. Statist. Math., vol. 22, pages 203–217, 1970.

[Akutsu 1999] T Akutsu, S Miyano et S Kuhara. *Identification of genetic networks from a small number of gene expression patterns under the Boolean network model.* Pacific Symposium On Biocomputing, vol. 28, no. 4, pages 17–28, 1999.

[An 1992] Z. An, D.A. Bell et J.G. Hughes. *On the Axiomization of Conditional Independence.* Kybernetes, vol. 21(7), pages 48 – 58, 1992.

[Anderson 1970] E. B. Anderson. *Asymptotic Properties of Conditional Maximum Likelihood Estimators.* Journal of the Royal Statistical Society, vol. 32, pages 283–301, 1970.

[Andersson 1995] S. Andersson, D. Madigan et M. Perlman. *A Characterization of Markov Equivalence Classes for Acyclic Digraphs.* Rapport technique 287, Department of Statistics, University of Washington, 1995.

[Auliac 2007] C. Auliac, F. d'Alché Buc et V. Frouin. *Learning Transcriptional Regulatory Networks with Evolutionary Algorithms Enhanced with Niching.* 7th International Workshop on Fuzzy Logic and Applications Lecture Notes in Computer Science, pages 612–619, 2007.

[Auliac 2008] C. Auliac, V. Frouin, X. Gidrol et F. d'Alche Buc. *Evolutionary approaches for the reverse-engineering of gene regulatory networks : A*

study on a biologically realistic dataset. BMC Bioinformatics, vol. 9, no. 1, page 91, 2008.

[Baldi 2001] P. Baldi et S. Brunak. Bioinformatics : The machine learning approach. MIT Press, 2001.

[Barra 2004] Vincent Barra. *Modélisation, classification et fusion de données biomédicales.* PhD thesis, Université Blaise Pascal - Clermont-Ferrand II, 2004.

[Bayes 1763] Thomas Bayes. *An Essay towards solving a Problem in the Doctrine of Chances.* Philosophical Transactions of the Royal Society of London, vol. 53, pages 370–418, 1763.

[Beal 2004] Matthew J. Beal, Francesco Falciani, Zoubin Ghahramani, Claudia Rangel et David L. Wild. *A Bayesian approach to reconstructing genetic regulatory networks with hidden factors.* Bioinformatics, vol. 21, no. 3, pages 349–356, 2004.

[Benferhat 2011] Salem Benferhat et Faiza Titouna. *On the Fusion of Probabilistic Networks.* In Kishan G. Mehrotra, Chilukuri K. Mohan, Jae C. Oh, Pramod K. Varshney et Moonis Ali, editeurs, IEA/AIE (1), volume 6703 of *Lecture Notes in Computer Science*, pages 49–58. Springer, 2011.

[Benjamini 1995] Y. Benjamini et Y. Hochberg. *Controlling the false discovery rate : a practical and powerful approach to multiple testing.* Journal of the Royal Statistical Society, vol. 57, no. 1, pages 125–133, 1995.

[Blanco 2003] R. Blanco, I. Inza et P. Larrañaga. *Learning Bayesian Networks in the space of structures by Estimation of Distribution Algorithms.* International Journal of Intelligent Systems, vol. 18, pages 205–220, 2003.

[Borgelt 2002] C. Borgelt et R. Kruse. *Graphical Models - Methods for Data Analysis and Mining.* John Wiley & Sons, 2002.

[Bunke 1997] H. Bunke. *On a relation between graph edit distance and maximum common subgraph.* Pattern Recogn. Lett., vol. 18, no. 9, pages 689–694, 1997.

[Buntine 1991] W. L. Buntine. *Theory refinement on Bayesian networks.* In Proc. UAI'91, pages 52–60, 1991.

[Butte 2000] Atul J. Butte, Isaac S. Kohane et I. S. Kohane. *Mutual Information Relevance Networks : Functional Genomic Clustering Using Pairwise Entropy Measurements.* Pacific Symposium on Biocomputing, vol. 5, pages 415–426, 2000.

[Campbell 2005] N.A. Campbell et J.B. Reece. Biology : Concepts and connections with mybiology. Biology. Pearson Benjamin-Cummings Publishing Company, 2005.

[Carter 2004] Scott L. Carter, Christian M. Brechbuhler, Michael Griffin et Andrew T. Bond. *Gene co-expression network topology provides a framework for molecular characterization of cellular state.* Bioinformatics, vol. 20, no. 14, pages 2242–2250, Septembre 2004.

[Carvalho 2009] A. M. Carvalho. *Scoring functions for learning Bayesian networks.* Inesc-id Tec. Rep., 2009.

[Castillo 1997] E Castillo, J M Gutierrez et A S Hadi. Expert systems and probabilistic network models. Springer-Verlag, 1997.

[Chan 2007] Zeke S H Chan, Lesley Collins et N Kasabov. *Bayesian learning of sparse gene regulatory networks.* Bio Systems, vol. 87, no. 2-3, pages 299–306, 2007.

[Chaouiya 2004] Claudine Chaouiya, Elisabeth Remy, Paul Ruet et Denis Thieffry. *Qualitative Modelling of Genetic Networks : From Logical Regulatory Graphs to Standard Petri Nets.* In Jordi Cortadella et Wolfgang Reisig, editeurs, Applications and Theory of Petri Nets 2004, volume 3099 of *Lecture Notes in Computer Science*, pages 137–156. Springer Berlin – Heidelberg, 2004.

[Cheng 2002] J. Cheng. *Learning Bayesian networks from data : An information-theory based approach.* Artificial Intelligence, vol. 137, no. 1-2, pages 43–90, Mai 2002.

[Chickering 1995] D. Chickering. *A Transformational Characterization of Equivalent Bayesian Network Structures.* In Philippe Besnard et Steve Hanks, editeurs, Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence (UAI'95), pages 87–98, San Francisco, CA, USA, Août 1995. Morgan Kaufmann Publishers.

[Chickering 1996] D.M. Chickering. *Learning Bayesian Networks is NP-Complete.* Learning from Data : Artificial Intelligence and Statistics V, Springer-Verlag, pages 121–130, 1996.

[Chickering 2002a] D. Chickering et C. Meek. *Finding Optimal Bayesian Networks.* In Adnan Darwiche et Nir Friedman, editeurs, Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence (UAI-02), pages 94–102, S.F., Cal., 2002. Morgan Kaufmann Publishers.

[Chickering 2002b] D.M. Chickering. *Learning equivalence classes of bayesian-network structures.* Journal of Machine Learning Research, vol. 2, pages 445–498, 2002.

[Chow 1968] C.K. Chow et C.N. Liu. *Approximating discrete probability distributions with dependence trees.* IEEE Transactions on Information Theory, vol. 14, no. 3, pages 462–467, 1968.

[Ciaccio 2010] Mark F Ciaccio, Joel P Wagner, Chih-Pin Chuu, Douglas A Lauffenburger et Richard B Jones. *Systems analysis of EGF receptor signaling dynamics with microwestern arrays.* Nature Methods, vol. 7, no. 2, pages 148–155, 2010.

[Comet 2005] J.P. Comet, H. Klaudel et S. Liauzu. *Modeling Multivalued Genetic Regulatory Networks Using High Level Petri Nets.* In G. Ciardo and P. Darondeau (eds), Proc. of the Int. Conf. on the Application and Theory of Petri Nets, Lecture Notes in Computer Science 3536, pages 208–227. Springer–Verlag, 2005.

[Cooper 1992] G.F. Cooper et E. Herskovits. *A Bayesian Method for the Induction of Probabilistic Networks from Data.* Machine Learning, vol. 9, pages 309–347, 1992.

[Cowie 2007] J. Cowie, L. Oteniya et R. Coles. *Particle Swarm Optimisation for learning Bayesian Networks.* In World Congress on Engineering'07, pages 71–76, 2007.

[Dawid 1979] A.P. Dawid. *Conditional Independence in statistical theory.* Journal of the Royal Statistical Society, vol. 41, pages 1–31, 1979.

[De Ridder 2010] Jeroen De Ridder, Alice Gerrits, Jan Bot, Gerald De Haan, Marcel Reinders et Lodewyk Wessels. *Inferring combinatorial association logic networks in multimodal genome-wide screens.* Bioinformatics, vol. 26, no. 12, pages i149–i157, 2010.

[Delaplace 2007] A. Delaplace, T. Brouard et H. Cardot. *Two Evolutionary Methods for Learning Bayesian Network Structures.* International Conference on Computational Intelligence and Security, pages 288–297, 2007.

[Dempster 1967] A P Dempster. *Upper and lower probabilities induced by a multivalued mapping.* Annals of Mathematical Statistics, vol. 38, no. 2, pages 325–339, 1967.

[Dempster 1977] A. P. Dempster, N. M. Laird et D. B. Rubin. *Maximum likelihood from incomplete data via the EM algorithm.* Journal of the Royal Statistical Society : Series C, vol. 39, no. 1, pages 1–38, 1977.

[Djebbari 2008] Amira Djebbari et John Quackenbush. *Seeded Bayesian Networks : Constructing genetic networks from microarray data.* BMC Systems Biology, vol. 2, no. 1, page 57, 2008.

[Dor 1992] Dorit Dor et Michael Tarsi. *A simple algorithm to construct a consistent extension of a partially oriented graph.* Rapport technique R-185, Cognitive Systems Laboratory, Computer Science Department, University of California, Los Angeles, CA, USA, Octobre 1992.

[Dougherty 1995] James Dougherty, Ron Kohavi et Mehran Sahami. *Supervised and Unsupervised Discretization of Continuous Features.* In International Conference on Machine Learning, pages 194–202, 1995.

[Dudoit 2003] S. Dudoit, J.P. Shaffer et J.C. Boldrick. *Multiple Hypothesis Testing in Microarray Experiments.* Statist. Sci., vol. 18, no. 1, pages 71–103, 2003.

[Eades 1984] Peter Eades. *Graph Drawing by Force-Directed Placement.* Congressus Numerantium, vol. 42, no. 11, pages 149–160, 1984.

[Efron 1993] B. Efron et R.J. Tibshirani. An introduction to the bootstrap. London : Chapman and Hall, 1993.

[Elati 2007] M. Elati. *Apprentissage de réseaux de régulation génétique à partir de données d'expression.* PhD thesis, Université Paris Nord, France, 2007.

[Foster 2004] D. P. Foster et R. A. Stine. *The contribution of parameters to stochastic complexity.* In P. D. Grünuwald, I. J. Myung, and M. A. Pitt (Eds.), Advances in Minimum Description Length : Theory and Applications. MIT Press, 2004.

[François 2006] O. François. *De l'identification de structure de réseaux bayésiens à la reconnaissance de formes à partir d'informations completes où incompletes.* PhD thesis, Institut National des Science Appliquées de Rouen, 2006.

[Friedman 1999a] Goldszmidt M. Wyner A. J. Friedman N. *On the Application of the Bootstrap for Computing Confidence Measures on Features of Induced Bayesian Networks.* In Proceedings of the Seventh International Workshop on Artificial Intelligence and Statistics, pages 197–202, San Francisco, CA, 1999. Morgan Kaufmann, San Francisco.

[Friedman 1999b] N. Friedman, M. Goldszmidt et A.J. Wyner. *Data analysis with bayesian networks : A bootstrap approach.* Proceedings of the 15th Annual Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann Publishers, vol. 24, pages 206–215, 1999.

[Friedman 2000] N. Friedman, M. Linial, I. Nachman et D. Pe'er. *Using Bayesian Networks to Analyze Expression Data.* In Fourth Annual International Conference in Computational Molecular Biology, pages 127–135, 2000.

[Friedman 2004] N. Friedman. *Inferring Cellular Networks Using Probabilistic Graphical Models.* Science, vol. 303, pages 799–805, 2004.

[Galles 1996] David Galles et Judea Pearl. *Axioms of Causal Relevance.* Artificial Intelligence, vol. 97, pages 97–1, 1996.

[Geiger 1990] Dan Geiger, Thomas Verma et Judea Pearl. *Identifying independence in Bayesian Networks.* Networks, vol. 20, pages 507–534, 1990.

[Griffths 2007] J. F. Griffths, S. R. Wessler, R. C. Lewontin et S. B. Carroll. An introduction to genetic analysis. W. H. Freeman and Company, 2007.

[Grünwald 2007] P. D. Grünwald. The minimum description length principle. MIT Press, 2007.

[Haiyang 2008] J. Haiyang, L. Dayou, C. Juan, G. Jinghua et L. Key. *Learning Markov equivalence classes of Bayesian Network with immune genetic algorithm.* pages 197 – 202, 2008.

[Hartemink 2001a] A J Hartemink, D K Gifford, T S Jaakkola et R A Young. *Using graphical models and genomic expression data to statistically validate models of genetic regulatory networks.* Pacific Symposium On Biocomputing, vol. 6, pages 422–433, 2001.

[Hartemink 2001b] Alexander John Hartemink. *Principled Computational Methods for the Validation and Discovery of Genetic Regulatory Networks by.* PhD thesis, 2001.

[Heckerman 1995] D. Heckerman et D.M. Chickering. *Learning Bayesian networks : The combination of knowledge and statistical data.* Machine Learning, pages 20–197, 1995.

[Herskovits 1991] E.H. Herskovits. *Computer-based probabilistic network construction.* PhD thesis, Medical Information Sciences, Stanford University, Stanford, CA., 1991.

[Holm 1979] S. Holm. *A simple sequentially rejective multiple test procedure.* Scandinavian Journal of Statistics, vol. 6, no. 2, pages 65–70, 1979.

[Hoon 2003] Michiel De Hoon, Seiya Imoto et Satoru Miyano. *Inferring gene regulatory networks from time-ordered gene expression data of bacillus subtilis using differential equations.* In Pac. Symp. Biocomput, pages 17–28, 2003.

[Hsu 2002] W.H. Hsu, H. Guo, B.B. Perry et J.A. Stilson. *A Permutation Genetic Algorithm For Variable Ordering In Learning Bayesian Networks From Data.* GECCO '02 : Proceedings of the Genetic and Evolutionary Computation Conference, pages 383–390, 2002.

[Husmeier 2003] D. Husmeier. *Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic Bayesian networks.* Bioinformatics, 2003.

[Karlebach 2008] Guy Karlebach et Ron Shamir. *Modelling and analysis of gene regulatory networks.* Nature Reviews Molecular Cell Biology, vol. 9, no. 10, pages 770–780, Septembre 2008.

[Kauffman 1969] S. Kauffman. *Metabolic stability and epigenesis in randomly constructed genetic nets.* Journal of Theoretical Biology, vol. 22, no. 3, pages 437–467, Mars 1969.

[Kim 2004] Sunyong Kim, Seiya Imoto et Satoru Miyano. *Dynamic Bayesian network and nonparametric regression for nonlinear modeling of gene networks from time series gene expression data.* Biosystems, vol. 75, no. 1-3, pages 57 – 65, 2004.

[Kramosil 1988] I. Kramosil. *A Note on Nonaxiomatizability of Independence Relations Generated by Certain Probabilistic Structures.* Kybernetika, vol. 24(6), pages 439–446, 1988.

[Kullback 1951] S. Kullback et R.A. Leibler. *On Information and Sufficiency.* Annals of Mathematical Statistics, vol. 22(1), pages 79–86, 1951.

[Lahdesmaki 2003] Harri Lahdesmaki, Ilya Shmulevich et Olli Yli-Harja. *On Learning Gene Regulatory Networks Under the Boolean Network Model.* Machine Learning, vol. 52, pages 147–167, 2003.

[Lander 1999] Eric S Lander. *Array of hope.* Nature Genetics, vol. 21, pages 3–4, 1999.

[Larrañaga 1994] P. Larrañaga, M. Poza, Y. Yurramendi, R. H. Murga et C. M. H. Kuijpers. *Structure learning of bayesian networks by genetic algorithms : A performance analysis of control parameters.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 18, pages 912–926, 1994.

[Larrañaga 1996] P. Larrañaga, C.M.H. Kuijpers, R.H. Murga et Y. Yurramendi. *Learning Bayesian Network structures by searching for the best ordering with Genetic Algorithms.* IEEE Transactions on Systems, Man and Cybernetics, vol. 26, no. 4, pages 487–493, 1996.

[Larrañaga 2002] P. Larrañaga. Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation, chapitre An introduction to probabilistic graphical models, pages 25–54. Kluwer Academic Publishers, Boston/Dordrecht/London, 2002.

[Lee 2010] J. Lee, W. Chung, E. Kim et S. Kim. *A new genetic approach for structure learning of Bayesian networks : Matrix genetic algorithm.* International Journal of Control, Automation and Systems, vol. 8, pages 398–407, 2010.

[Lozano 2006] Jose A. Lozano, Pedro Larrañaga, Iñaki Inza et Endika Bengoetxea. Towards a new evolutionary computation : Advances on estimation of distribution algorithms (studies in fuzziness and soft computing). Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[Lucas 2007] P Lucas et J A Gmez. *Advances in Probabilistic Graphical Models.* Reliable Computing, vol. 2, no. 2, pages 195–203, 2007.

[Madan Babu 2004] M. Madan Babu. Computational genomics : Theory and application, chapitre An introduction to microarray data analysis, pages 225–249. Horizon Scientific Press, Norwich, UK, 2004.

[Matsuno 2000] H Matsuno, A Doi, M Nagasaki et S Miyano. *Hybrid Petri net representation of gene regulatory network.* Pacific Symposium On Biocomputing, vol. 349, no. 338-349, pages 341–352, 2000.

[Meek 1995a] C. Meek. *Causal inference and causal explanation with background knowledge.* In Proceedings of 11th Conference on Uncertainty in Artificial Intelligence, pages 403–418, 1995.

[Meek 1995b] C. Meek. *Strong-completeness and faithfulness in belief networks.* In S. Hanks, P. Besnard (Eds.), Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence. in : , Montreal, QU, Morgan Kaufmann, San Mateo, CA, pages 411–418, 1995.

[More 2010] Sara Miner More et Pavel Naumov. *An independence relation for sets of secrets.* Studia Logica, vol. 94(1), pages 73–85, 2010.

[More 2011] Sara Miner More, Pavel Naumov et Benjamin Sapp. *Concurrency Semantics for the Geiger-Paz-Pearl Axioms of Independence.* In CSL, pages 443–457, 2011.

[Muruzbal 2007] J. Muruzbal et C. Cotta. *A study on the evolution of Bayesian network graph structures.* Advances in Probabilistic Graphical Models, Studies in Fuzziness and Soft Computing, vol. 214, pages 193–213, 2007.

[Nachman 2004] I. Nachman, A. Regev et N. Friedman. *Inferring quantitative models of regulatory networks from expression data.* Bioinformatics, vol. 20, no. suppl 1, pages i248–i256, 2004.

[Naïm 2007] P. Naïm, P. Wuillemin, P. Leray, O. Pourret et A. Becker. *Les réseaux bayésiens.* 3e Édition Eyrolles, 2007.

[nga 1997] P. Larra nga, B. Sierra, M. J. Gallego, M. J. Michelena et J. M. Picaza. *Learning Bayesian Networks by Genetic Algorithms : A case*

study in the prediction of survival in malignant skin melanoma. Prob. Models and Fuzzy Logic., vol. 1211, pages 261–272, 1997.

[Nguyen 2009] Hoai-Tuong Nguyen, Philippe Leray, Gérard Ramstein et Yannick Jacques. *Reconstruction de réseaux de régulations génétiques par l'approche évolutionnaire sur les réseaux Bayésiensk.* In Proceedings of MODGRAPH - Journées Ouvertes en Biologie, Informatique et Mathématiques (JOBIM), June, 9-11, 2009, pages 42–46, 2009.

[Nguyen 2010] Hoai-Tuong Nguyen, Philippe Leray, Gérard Ramstein et Yannick Jacques. *Differential study of the cytokine network in the immune system by the evolutionary algorithm based on the bayesian network.* In Proceedings of Doctoral Colloquium of the 2nd Asian Conference on Intelligent Information and Database Systems (ACIIDS), pages 173–182, 2010.

[Nguyen 2011a] Hoai-Tuong Nguyen, Philippe Leray et Gérard Ramstein. *Multiple hypothesis testing and quasi essential graph for comparing two sets of bayesian networks.* In Proceedings of the 15th international conference on Knowledge-based and intelligent information and engineering systems - Volume Part II, KES'11, pages 176–185. Springer-Verlag, 2011.

[Nguyen 2011b] Hoai-Tuong Nguyen, Philippe Leray et Gérard Ramstein. *Summarizing and visualizing a set of bayesian networks with quasi essential graphs.* ASMDA (Applied Stochastic Models and Data Analysis) International Society, Rome, Italy, 2011.

[Nicandro 2006] C.R. Nicandro, A.M. Hector-Gabriel, B.M. Rocio-Erandi et N.F. Luis-Alonso. *How good are the Bayesian information criterion and the minimum description length principle for model selection ? A Bayesian network analysis.* MICAI 2006 : Advances in Artificial Intelligence. LNCS Springer, vol. 4293, pages 494–504, 2006.

[Ong 2002] Irene M. Ong, Jeremy D. Glasner et David Page. *Modelling regulatory pathways in E. coli from time series expression profiles.* Bioin-

formatics, vol. 18, no. suppl 1, pages S241–S248, 2002.

[Opitz 1999] D. Opitz et R. Maclin. *Popular ensemble methods : An empirical study.* Journal of Artificial Intelligence Research, vol. 11, pages 169–198, 1999.

[Paulevé 2011] Loïc Paulevé, Morgan Magnin et Olivier Roux. *Refining Dynamics of Gene Regulatory Networks in a Stochastic π-Calculus Framework.* In Corrado Priami, Ralph-Johan Back, Ion Petre et Erik de Vink, editeurs, Transactions on Computational Systems Biology XIII, volume 6575 of *Lecture Notes in Computer Science*, pages 171–191. Springer Berlin / Heidelberg, 2011.

[Pearl 1985] J. Pearl et A. Paz. *GRAPHOIDS : A graph-based logic for reasoning about relevance relations.* Rapport technique (R–53–L, Cognitive Systems Laboratory, University of California, Los Angeles, 1985.

[Pearl 2000] Judea Pearl. *Causality : Models, Reasoning and Inference.* MIT Press, 2000.

[Pe'er 2001] D. Pe'er, E. Regev, G. Elidan et N. Friedman. *Inferring subnetworks from perturbed expression profiles.* Bioinformatics, vol. 17, pages S215–S224, 2001.

[Pelikan 2000] M. Pelikan, D.E. Goldberg et E. Cantu-Paz. *Linkage problem, distribution estimation, and Bayesian Networks.* Evolutionary Computation, vol. 8, no. 3, pages 311–340, 2000.

[Pelikan 2001] Martin Pelikan et David. E. Goldberg. *Hierarchical Bayesian Optimization Algorithm = Bayesian Optimization Algorithm + Niching + Local Structures.* pages 525–532. Morgan Kaufmann, 2001.

[Peña 2007] Jose M. Peña, Roland Nilsson, Johan Björkegren et Jesper Tegnér. *Towards scalable and data efficient learning of Markov boundaries.* Int. J. Approx. Reasoning, vol. 45, pages 211–232, July 2007.

[Philip P. 2004] Le Philip P., Amit Bahl et Lyle H Ungar. *Using prior knowledge to improve genetic network reconstruction from microarray data.* In Silico Biology, vol. 4, no. 3, pages 335–353, 2004.

[Quackenbush 2002] J. Quackenbush. *Microarray data normalization and transformation.* Nature, vol. 32, pages 123–148, 2002.

[Redestig 2007] Henning Redestig, Daniel Weicht, Joachim Selbig et Matthew Hannah. *Transcription factor target prediction using multiple short expression time series from Arabidopsis thaliana.* BMC Bioinformatics, vol. 8, no. 1, page 454, 2007.

[Rhodes 2002] Daniel R. Rhodes, Terrence R. Barrette, Mark A. Rubin, Debashis Ghosh et Arul M. Chinnaiyan. *Meta-analysis of microarrays : interstudy validation of gene expression profiles reveals pathway dysregulation in prostate cancer.* Cancer research, vol. 62, no. 15, pages 4427–4433, Août 2002.

[Ribeiro 2008] A. S. Ribeiro, S.A. Kauffman, J. Lloyd-Price, B. Samuelsson et J. E. S. Socolar. *Mutual information in random Boolean models of regulatory networks.* Phys. Rev. E, vol. 77, no. 1, page 011901, Jan 2008.

[Rissanen 1978] J. Rissanen. *Modeling by shortest data description.* Automatica, vol. 14(5), pages 465–471, 1978.

[Rodin 2005] A.S. Rodin et E. Boerwinkle. *Mining genetic epidemiology data with Bayesian networks I : Bayesian networks and example application (plasma apoE levels).* Bioinformatics, vol. 21(15), pages 3273–3278, 2005.

[Rodrigues De Morais 2008] Sergio Rodrigues De Morais et Alex Aussem. *A Novel Scalable and Data Efficient Feature Subset Selection Algorithm.* In Proceedings of the European conference on Machine Learning and Knowledge Discovery in Databases - Part II, ECML PKDD '08, pages 298–312, Berlin, Heidelberg, 2008. Springer-Verlag.

[Santos Jr. 2011] Eugene Santos Jr., John T. Wilkinson et Eunice E. Santos. *Fusing multiple Bayesian knowledge sources.* Int. J. Approx. Reasoning, vol. 52, pages 935–947, October 2011.

[Schlitt 2007] Thomas Schlitt et Alvis Brazma. *Current approaches to gene regulatory network modelling.* BMC Bioinformatics, vol. 8, no. Suppl 6, page S9, 2007.

[Schwarz 1978] G. Schwarz. *Estimating the dimension of a model.* Annals of Statistics, vol. 6, pages 461–464, 1978.

[Shafer 1976] G. Shafer. A mathematical theory of evidence. Princeton University Press, 1976.

[Shmulevich 2002] I Shmulevich et E R Dougherty. *From Boolean to probabilistic Boolean networks as models of genetic regulatory networks.* Proceedings of the IEEE, vol. 90, no. 11, pages 1778–1792, 2002.

[Soinov 2003] Lev A. Soinov, Maria A. Krestyaninova et Alvis Brazma. *Towards reconstruction of gene networks from expression data by supervised learning.* Genome biology, vol. 4, no. 1, 2003.

[Spirtes 1990] P. Spirtes, C. Glymour et R. Scheines. *Causality from probability.* In : G. McKee ed. : Evolving knowledge in natural and artificial intelligence, London : Pitman, pages 181–199, 1990.

[Spirtes 1995] P. Spirtes et C. Meek. *Learning Bayesian networks with discrete variables from data.* In Proceedings of First International Conference on Knowledge Discovery and Data Mining, pages 294–299. AAAI Press, 1995.

[Spirtes 2001] P. Spirtes, C. Glymour et R. Scheines. *Causation, Prediction and Search.* MIT Press, 2001.

[Spohn 1980] W. Spohn. *Stochastic Independence, Causal Independence, and Shieldability.* Journal of Philosophical Logic, vol. 9, pages 73–99, 1980.

[Spohn 1994] Wolfgang Spohn. *On the Properties of Conditional Independence.* In Patrick Suppes, : Scientific Philosopher Vol. 1 : Probability and Probabilistic Causality., pages 173–194. Kluwer, 1994.

[Steggles 2007] L. Jason Steggles, Richard Banks, Oliver Shaw et Anil Wipat. *Qualitatively modelling and analysing genetic regulatory networks : a Petri net approach.* Bioinformatics, vol. 23, no. 3, pages 336–343, 2007.

[Stevens 2005] John R Stevens et R W Doerge. *Meta-Analysis Combines Affymetrix Microarray Results Across Laboratories.* Comparative and Functional Genomics, vol. 6, no. 3, pages 116–122, 2005.

[Studený 1989] M Studený. *Multiinformation and the problem of characterization of conditional independence relations.* Problems of Control and Information Theory, 1989.

[Sun 2005] J. Sun, Q. Zhang et E. P.K. Tsang. *DE/EDA : A new evolutionary algorithm for global optimization.* Information Sciences, vol. 169, no. 3-4, pages 249 – 262, 2005.

[Tsamardinos 2006] I. Tsamardinos, L.E. Brown et C.F. Aliferis. *The max-min hill-climbing Bayesian network structure learning algorithm.* Machine Learning, vol. 65, no. 1, pages 31–78, 2006.

[Verma 1991] T. Verma et J. Pearl. *Equivalence and Synthesis of Causal Models.* In M. Henrion, R. Shachter, L. Kanal et J. Lemmer, editeurs, Proceedings of the Sixth Conference on Uncertainty in Artificial Intelligence, pages 220–227, San Francisco, 1991. Morgan Kaufmann.

[Wang 2010] T. Wang et J. Yang. *A heuristic method for learning bayesian networks using discrete particle swarm optimization.* Knowl. and Info. Sys., vol. 24, pages 269–281, 2010.

[Zou 2005] Min Zou et Suzanne D. Conzen. *A new dynamic Bayesian network (DBN) approach for identifying gene regulatory networks from time course microarray data.* Bioinformatics, vol. 21, no. 1, pages 71–79, 2005.