Inférence de réseaux causaux à partir de données interventionnelles

Gilles Monneret

École doctorale nº 386 : Sciences Mathématiques de Paris Centre

THÈSE

pour obtenir le grade de docteur délivré par

l'Université Pierre et Marie Curie

Spécialité doctorale "Statistiques"

présentée et soutenue publiquement par

Gilles MONNERET

le 15 février 2018

Inférence de réseaux causaux à partir de données interventionnelles

Directeurs de thèse :
Grégory NUEL, FLORENCE JAFFRÉZIC et **Andrea RAU**

Jury

M. Christophe AMBROISE,	Professeur	Examinateur
M. Ismaël CASTILLO,	Professeur	Examinateur
M. Hervé ISAMBERT,	Chef d'équipe de recherche	Examinateur
Mme. Florence JAFFRÉZIC,	Directrice de recherche	Directrice de thèse
M. Grégory NUEL,	Directeur de recherche	Directeur de thèse
M. Matthieu VIGNES,	Senior lecturer	Rapporteur
Mme. Nathalie VILLA-VIALANEIX,	Chargée de recherche	Rapporteur

Université Pierre et Marie Curie
4 Place Jussieu, 75005 Paris, France

Table des matières

Table des matières iii

1 Introduction générale 5
- 1.1 Contexte biologique . . . 6
- 1.2 Réseaux bayésiens . . . 9
- 1.3 Causalité . . . 14
- 1.4 Organisation du manuscrit . . . 28

2 Hypothèse d'acyclicité 29
- 2.1 Introduction . . . 30
- 2.2 Méthode . . . 31
- 2.3 Applications et exemples . . . 35
- 2.4 Discussion . . . 37
- 2.5 Annexe . . . 38

3 Détection de relations causales marginales 41
- 3.1 Introduction . . . 43
- 3.2 Méthode . . . 44
- 3.3 Résultat . . . 47
- 3.4 Discussion . . . 53
- 3.5 Références . . . 54

4 Estimation d'effets causaux par pénalisation L_2 57
- 4.1 Introduction . . . 58
- 4.2 Causalité dans l'expression des gènes . . . 59
- 4.3 Navigation dans l'espace des ordres : MCMC-Mallows . . . 61
- 4.4 Vers la grande dimension : Pénalisation ridge & squelette . . . 64
- 4.5 Simulations et résultats . . . 67
- 4.6 Conclusion . . . 73

5 Parallel tempering 75
- 5.1 Introduction . . . 76
- 5.2 Résultats . . . 80
- 5.3 Conclusion . . . 83

6 Approximation de Laplace 85
- 6.1 Introduction . . . 86
- 6.2 Méthode . . . 87
- 6.3 Résultats . . . 91
- 6.4 Discussion-Ouverture . . . 96

7 Discussion **97**

A Annexes **I**
A.1 Approche bayésienne I

B Bibliographie **V**
B.1 Références V

Remerciements

Ce travail n'aurait pas pû aboutir sans le soutien de nombreuses personnes.

Je veux tout d'abord remercier mes encadrants, sans qui évidemment ce travail n'aurait pas pû aboutir. Les nombreux entretiens du mercredi avec Grégory m'ont souvent permis d'enrichir continuellement mon travail de nouvelles idées. L'expertise et la bienveillance d'Andrea furent tout aussi précieuse pour me permettre de m'approprier ce sujet. Je n'aurait pas réussit à finir ce travail sans Florence, qui m'a notamment portée sur la dernière ligne droite, et qui a relu de très nombreuses fois les différentes versions de ce manuscrit.

Je veux remercier l'université Pierre et Marie Curie de m'avoir permis de faire cette thèse grâce à l'allocation doctorale que j'ai pû bénéficier durant ces trois ans.

J'ai mené ces thèses au sein de deux laboratoires. L'ambiance à été très positive à l'intérieur de ceux-ci. Je veux remercier l'équipe du laboratoire de probabilité et statistique de Jussieu et notamment Florence pour ses conseils avisées, ainsi que l'ensemble des doctorants partageant mon bureau pour leur bienveillance. Je remercie également l'équipe de génétique animale et biologie intégrative, et en particulier Jean-Noël, Mathieu, Mélina et Denis.
Je voudrais enfin remercier ma conjointe pour ses relectures et sa patience durant ces trois longues années.

Je dédie ce travail à ma famille de cœur.

Articles, Communications et logiciels

Articles publiés dans une revue à comité de lecture

MONNERET, Gilles, JAFFREZIC, Florence, RAU, Andrea, et Grégory Nuel. Estimation d'effets causaux dans les réseaux de régulation génique : vers la grande dimension. *Revue des Sciences et Technologies de l'Information-Série RIA : Revue d'Intelligence Artificielle*, 2015, vol. 29, no 2, p. 205-227.

MONNERET, Gilles, JAFFRÉZIC, Florence, RAU, Andrea, et al. Identification of marginal causal relationships in gene networks from observational and interventional expression data. *PLoS One*, 2017, vol. 12, no 3, p. e0171142.

Communications orales

MONNERET, Gilles, JAFFREZIC, Florence, RAU, Andrea, et al. Etude de l'hypothèse de non cyclicité dans les modèles graphiques orientés causaux, *Journée des Statistiques*, 2016, Lille, France.

MONNERET, Gilles, JAFFREZIC, Florence, RAU, Andrea, et al. Identification of marginal causal relationships in gene networks, from observational and interventional expression data, *JFRB 16*, 2016, Clermont-Ferrand, France.

MONNERET, Gilles, JAFFREZIC, Florence, RAU, Andrea, et al. Identification of marginal causal relationships in gene networks, from observational and interventional expression data, *Statistical Methods for Post Genomic Data*, 2016, Lille, France.

MONNERET, Gilles, JAFFREZIC, Florence, RAU, Andrea, et al. Identification of marginal causal relationships in gene networks, *XXVIIIth International Biometric Conference*, 2016, Victoria, Canada.

MONNERET, Gilles, JAFFREZIC, Florence, RAU, Andrea, et al. Identification of causal relationships in gene networks, from observational and interventional expression data, *ETH Research Seminar on Statistics*, 2016, Zürich, Switzerland.

MONNERET, Gilles, JAFFREZIC, Florence, RAU, Andrea, et al. Identification of causal relationships in gene networks, *Oldenburg University Seminar on Physics*, 2017, Oldenburg, Germany.

Communications affichées

MONNERET, Gilles, RAU, Andrea, JAFFREZIC, Florence, NUEL, Gregory. Estimation of causal effects in high dimensional gene regulatory networks from a mixture of observational and intervention data, *Workshop on Statistical Learning of Biological Systems from Perturbations*, 2015, Ascona, Suisse.

FIETH, Pascal, MONNERET, Gilles, et al. Improving Causal Gaussian Bayesian Network Inference using Parallel Tempering, *RSG Dream*, 2016, Phoenix, USA.

Packages R

`MarginalCausality`, 2017, Github (https ://github.com/Monneret/MarginalCausality). Package dont le but est d'établir si un gène, via une intervention, a effectivement un lien sur un autre groupe de gènes et d'en calculer l'effet causal.

`bandsolve`, 2017, Github (https ://github.com/Monneret/bandsolve). Package dont le but est de résoudre rapidement le problème d'algébre linéaire des systèmes linéaires symétriques à bandes via une décomposition LDL.

Séjour au sein d'un laboratoire international

Séjour de 3 mois au sein de l'équipe de statistiques de l'ETH Zürich, financé par une bourse de la Fondation Science Mathématiques de Paris.

Chapitre 1

Introduction générale

Cette thèse a pour objet la découverte, par des moyens statistiques, des liens *causaux* régissant le génome. Dans ce chapitre d'introduction, nous commencerons par rappeler le contexte biologique de ce travail. Nous continuerons par les outils statistiques que nous utiliserons tout au long de la thèse. Nous finirons ce chapitre par une discussion sur le terme de causalité, qui peut être ambigu mais qui constitue pourtant l'objectif de ce travail.

1.1 Contexte biologique

1.1.1 Génomique

Les êtres vivants, de la bactérie à l'être humain, sont tous bâtis à partir de cellules. Celles-ci contiennent notamment du matériel génétique, ARN ou ADN protégé par une membrane qui définit la limite de la cellule. De plus, ces cellules peuvent posséder un noyau où le matériel génétique y est protégé. Le dogme fondamental de la biologie pose la relation entre ADN, ARN et protéine comme le pilier central des organismes vivants. Par divers mécanismes, l'ADN est transcrit en ARN, qui lui-même est traduit en protéine, avec des variations nombreuses dans ces deux étapes en fonction de l'environnement cellulaire. Ces protéines assurent les différentes fonctions cellulaires et extracellulaires, comme les voies de signalisation ou de transport au sein de la cellule, mais aussi la défense contre les attaques extérieures via les anticorps ou la transformation moléculaire via les enzymes. Ces différentes catégories de molécules font chacune l'objet d'études approfondies, pour en comprendre le fonctionnement, parmi lesquelles :

— la génomique, qui est l'information disponible sur l'ADN d'un individu,

— la transcriptomique, qui est l'ensemble des transcrits présents dans la cellule,

— la protéomique, qui est l'ensemble des protéines exprimées par une cellule à un instant donné,

— la métabolomique, qui est l'ensemble des petites molécules, les métabolites, présentes dans la cellule à un instant donné,

— la phénomique, qui est l'ensemble des phénotypes, c'est-à-dire les caractéres observables,

— la métagénomique, qui concerne l'information génétique disponible dans un milieu donné (par exemple, l'intestin).

Ces différents champs *-omics* ont par nature de nombreux liens entre eux, et varient en fonction du temps, de l'environnement cellulaire et extra-cellulaire. Ainsi dans un organisme complexe comme un mammifère, le transcriptome issu d'une cellule de foie à une étape embryonnaire sera tout à fait différent de celui d'une cellule de cœur d'un adulte, du fait de phénomènes épigénétiques comme la méthylation ou l'acétylation. Un exemple célèbre d'influence de l'épigénétique concerne la couleur des poils de rats : celle-ci varie en fonction de la méthylation d'un certain gène, allant du jaune au gris [24].

Il faut bien comprendre que ces différentes études ne sont possibles que par les formidables progrès techniques réalisés ces dernières décennies. En effet, les avancées se sont succédées avec une vitesse folle : avec la découverte de l'ADN en 1953 [104], ou de la structure de la première protéine toujours en 1953 [88], ou du séquençage complet d'un génome en 1977 [87]. Le passage à l'échelle industrielle de ce séquençage au début du XXI^{e} siècle a mis à la portée du plus grand nombre la lecture du génome. Ce sont des données qui ont l'avantage d'être stable pour un individu donné, peut-être plus encore que le

phénotype, contrairement aux concentrations ou même la présence d'ARN, de protéines ou de métabolites. Cependant cette stabilité se paye par la perte des des informations épigénétiques plus difficiles à obtenir, celles-ci modulant l'expression du génome dans le temps et dans l'espace. Ce cadre étant précisé, nous travaillerons uniquement avec des données transcriptomiques, en prenant comme hypothèse qu'elles décrivent fidèlement l'état d'activation ou d'inactivation des différents gènes présents dans le génome.

Ces données massives nécessitent des outils particuliers, et notamment des outils mathématiques. Cela donne lieu à la biologie des systèmes, qui s'entend regarder l'ensemble des éléments conjointement, permettant une interprétation qui ne peut être révélée par l'étude de chacun de ces éléments [59; 83]. Dans notre cadre particulier, il s'agira de l'étude de *réseaux* de régulation de gènes plutôt que celle de chacun des gènes pris indépendamment. Ces modèles mathématiques ne peuvent souvent être utilisés et appliqués que grâce à l'outil informatique : ainsi la biologie computationnelle concerne cette interface entre les données biologiques et l'informatique, où les problématiques de temps de calcul ou de stockage mémoire sont souvent de mise. Cette thèse s'inscrit dans ce cadre : nous allons tout au long de cette thèse présenter des méthodes capables d'utiliser de grands jeux de données issues de la biologie, dans le but d'évaluer les mécanismes du vivant par des moyens informatiques.

La lecture massive du génome étant actée, se pose la question de la compréhension des fonctions et des interactions à l'intérieur du génome. Cette compréhension nécessite souvent de faire varier un unique paramètre, ici un unique gène, toutes choses égales par ailleurs. La sélection a pourtant été longtemps l'unique manière de faire varier cette composante génétique, avec son lot de variables confondantes. Cela peut par exemple prendre la forme d'une invalidation fonctionnelle, c'est-à-dire qu'une lignée sélectionnée porte une version mutée d'un gène, amenant la perte de la fonction d'une protéine. La technique de l'ARN interférent, dont la découverte date des années 90, a permis de limiter quelque peu ces variables confondantes [92]. Cette technique consiste à injecter dans les cellules concernées de petits fragments qui s'apparient spécifiquement avec des ARN complémentaires dont l'on souhaite la destruction. Cet appariement provoque la destruction des ARN ciblés. Néanmoins cette spécificité n'est pas totale, et il y a toujours potentiellement des *off-target*, c'est-à-dire un appariement et donc une destruction avec d'autres ARN. Avec cette technique, on parlera de *knock-down* d'un gène, c'est-à-dire de la diminution de l'expression d'un gène, sans toutefois faire disparaitre cette expression. Récemment de nouvelles techniques sont apparues permettant l'édition précise et directe du génome. Parmi elles se trouve la très prometteuse CRISPR-Cas9 [93]. Cette technique consiste, par l'intermédiaire du complexe moléculaire CRISPR-Cas9, à détruire précisément une portion du génome. Cela permet de réduire considérablement ce phénomène de *off-target*. On parlera dans ce cas de *knock-out* ou d'invalidation génétique : il n'y a plus du tout d'expression pour le gène concerné, contrairement au *knock-down*.

Dans cette thèse, nous utiliserons exclusivement des données transcriptomiques, c'est-à-dire provenant d'expression de gènes. Celles-ci ont l'avantage de décrire si un gène est effectivement exprimé ou non, contrairement au code brut de l'ADN. En effet dans ce dernier, l'état de compaction, la méthylation et divers autres phénomènes biologiques peuvent amener un gène à ne jamais être exprimé dans un tissu donné.

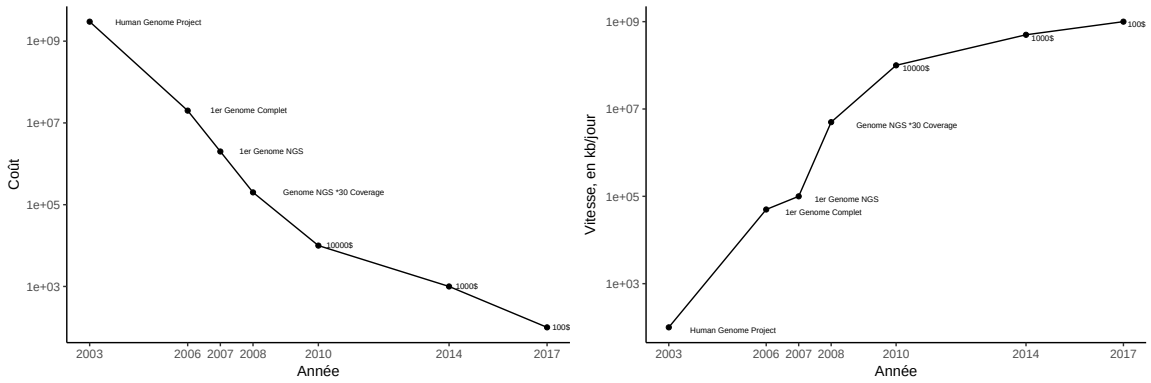


FIGURE 1.1 – Évolution du coût, en dollars, et de la vitesse, en kilobases par jour, du séquençage ADN pour l'être humain, 2003 a 2017. Source : National Human Genome Research Institute

1.1.2 Transcriptomique

Recueil des données

L'acquisition des données transcriptomiques s'effectue essentiellement par deux techniques : celle basée sur les *microarrays* ou puces à ADN, et celle, plus récente, basée sur le *high throughput sequencing* ou séquençage à haut débit. La première à été découverte dans les années 1970, même si le développement moderne et industriel date de la fin des années 1990 [10].

Dans les deux cas, puce ou séquençage, on commence par extraire les ARN des cellules, puis on synthétise un ADN complémentaire pour chaque molécule d'ARN, notée ADNc. Pour permettre la lecture, on démultiplie le nombre de chaque ADNc via la technique d'amplification en chaîne par polymérase, ou *polymerase chain reaction*.

Les procédures diffèrent ensuite en fonction de la technique utilisée : pour les puces à ADN, on retransforme les ADNc en ARNc, que l'on colore différemment en fonction de conditions expérimentales différentes. On prépare un gel où sont fixés des ADN complémentaires à des gènes dont l'on souhaite mesurer l'activité, et on laisse ensuite s'apparier librement les ARNc avec ces brins d'ADN. Le résultat est une puce possédant des milliers de lieux de depôts délivrant une couleur différente. C'est la mesure de cette fluorescence, en lien avec les colorations des différentes conditions expérimentales, qui permet d'obtenir une quantification relative du niveau d'expression, ou absolue si une seule condition est utilisée.

Le séquençage quant à lui utilise directement le brin d'ADNc, qu'il va séquencer, c'est-à-dire lire les éléments constitutifs du brin d'ADN. Une des techniques majeure, le pyroséquençage, à été découverte dans les années 1980. L'apparition des premiers séquenceurs industriels à haut-débit ne se fera que dans la fin des années 2000 [46]. Ceux-ci sont dits de deuxième génération, succédant à des séquenceurs à bas-débit. Ceci signifie principalement que ces machines sont capables de lire l'intégralité du génome extrêmement rapidement, en quelques heures seulement, là ou la précédente génération avait pris des années pour lire un seul génome. Les progrès technologiques se poursuivant, une troisième génération de séquenceurs, encore plus rapides, précis et moins chers arrive sur le marché. On peut apprécier l'évolution de la vitesse et du coût de ces nouvelles générations successives de séquenceurs dans la figure 1.1.

Etapes de pré-traitement

Quelle que soit la technique utilisée, les données ne sont généralement pas directement exploitables. Les différentes quantités obtenues ne sont pas comparables, du fait de la réponse propre à chaque puce. En plus de cette variabilité de mesure, il y a aussi celle inhérente à l'expérience : chaque observation induit un bruit propre sur l'ensemble des gènes qu'il convient de normaliser. Nous utiliserons principalement des bruits gaussiens, et ce faisant les données doivent respecter cette structure. Pour les données de puces, une intensité lumineuse, strictement positive, est donnée en sortie. Pour les données de séquençage, ce sont des comptages. Ceux-ci peuvent poser problème lorsque nombre de comptage est trop faible. On exclura donc ces cas là, traitables en utilisant une autre modélisation, par loi de Poisson [41] ou par une loi binomiale négative [3]. Pour revenir vers une distribution gaussienne, le principe est d'utiliser une transformation $\log_2$. Des packages R permettent d'affiner cette normalisation, `limma` [94] pour les données de puces, ou `voom` [57] pour celles de séquençage.

En fonction de la méthode d'acquisition choisie, d'autres prétraitements sont nécessaires. Par exemple, plusieurs ARN peuvent traduire l'expression d'un même gène [69]. Deux solution sont utilisés en pratique : choisir un représentant parmi ces différents ARN ou effectuer une moyenne. Le choix dépend des problématiques biologiques sous-jacente ainsi que des informations supplémentaires disponible. A défaut, on choisira ici de prendre la moyenne dans le but de rendre comparable les différents niveau d'activations des gènes.

1.1.3 Réseaux de régulation génique

Pour étudier l'effet des différents gènes, les biologistes ont tout d'abord travaillé selon une approche réductionniste : chaque élément est étudié à la loupe indépendamment du reste [95]. Cela a donné lieu à quantité d'informations, ce qui a permis d'établir les premières voies de signalisation métaboliques. Cependant, dans de nombreux cas, il n'y a pas ou peu d'informations sur une éventuelle interaction entre plusieurs constituants. Les données massives de transcriptomique peuvent quant à elles être utilisées pour effectuer une approche exploratoire, via l'utilisation des réseaux de régulation de gènes.

Un réseau de régulation de gènes, ou *gene regulatory network*, est un réseau que l'on infère à partir de données d'expression de gènes [27]. En ce sens, ce réseau traduit les interactions entre les différents gènes. Celles-ci sont de multiples natures : activation / inhibition via un facteur de transcription, méthylation d'un promoteur, transcription de micro-ARN. Dans tous ces cas, l'expression d'un gène a une influence sur l'expression d'un autre gène. Bien sûr, ne disposant que d'une infime partie de l'information, les différents intermédiaires liant les gènes sont oubliés, et l'on dira qu'un gène active ou inhibe un autre gène si l'on peut détecter une telle action à partir des données.

Cela pose donc la question de l'inférence de ces réseaux. Dans la partie suivante, nous allons étudier les différents outils statistiques le permettant.

1.2 Réseaux bayésiens

1.2.1 Modèles graphiques probabilistes

La théorie des graphes trouve son origine chez le généticien Wright en 1918, via la méthode de l'analyse de chemin [108]. Il s'agit à l'origine de décrire les liens causaux entre différentes variables, tout en liant les multiples quantités statistiques à ces descriptions

causales. Par la suite cette association entre dans une représentation graphique et une loi de probabilité va être reprise et développées dans les années 80, en créant les réseaux bayésiens et les champs de Markov [73]. Nous allons commencer par indiquer quelques méthodes utilisées dans le cadre de modèles non-orientés, puis nous définirons la terminologie de la théorie des modèles graphiques probabilistes [53; 99].

Modèles graphiques non-orientés

Une première approche pour modéliser un réseau de régulation de gènes est de travailler sur la similarité du profil d'expression entre les différentes variables. Les réseaux de co-expression sont des réseaux non-orientés qui décrivent ces profils d'expression [19; 26]. L'idée générale est la suivante : tout d'abord on se fixe un critère de co-expression. Cela peut être une corrélation de Pearson [13], de Spearman [105] ou de l'information mutuelle [11; 12]. Ensuite il s'agit de seuiller sur ce critère pour sélectionner les interactions les plus crédibles : chaque mesure donne donc lieu à un réseau différent. Le package R `WGCNA` [109] fait la synthèse entre ces différentes mesures en tentant de se rapprocher le plus d'un réseau biologique [1]. En travaillant seulement avec les corrélations partielles, on peut aussi utiliser l'idée de sélection de modèle et d'inférer un réseau de co-expression directement à partir des corrélations partielles [30]. Bien que ces méthodes soient capables de prendre en charge la présence de variables latentes ou de boucles de rétro-actions, on ne peut pas tirer d'informations précises sur le *sens* des relations entre les différents gènes.

Modèles graphiques orientés

Un graphe $\mathscr{G}$ est défini par un ensemble de noeuds V et par un ensemble d'arêtes E liant les différents nœuds. Si les arêtes sont orientées, on parlera de graphe orienté, et dans un tel cas si on enlève toutes les orientations, nous obtenons le squelette de $\mathscr{G}$ figure 1.2. Un graphe orienté acyclique est un graphe qui ne possède aucun cycle orienté, c'est-à-dire aucun cycle du type $X \to Y \to Z \to X$.

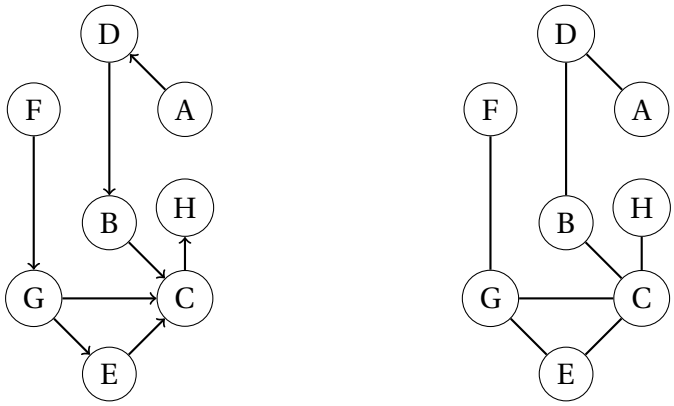


FIGURE 1.2 – Exemple de graphe orienté acyclique (ou DAG, pour *Directed Acyclic Graph*) à gauche. Squelette associé à droite.

La terminologie du champs de la théorie des graphes puise allégrement dans celle de la filiation (figure 1.3) : ainsi on dira que X est un parent de Y, noté pa si $X \to Y$, et inversement on dira que Y est un enfant de X. De même, on dira que X est un ancêtre de Y s'il existe une chaîne orientée du type $X \to A \to B \to .. \to F \to Y$, et dans ce cas Y est un descendant de X. On dira que des nœuds X et Y sont adjacents s'ils sont directement reliés. Un

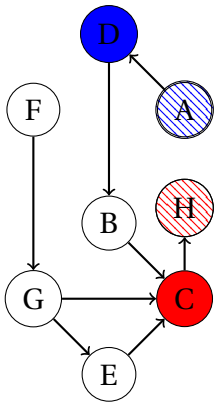


FIGURE 1.3 – Famille du nœud B. En bleu sont représentés les ancêtres de B, en rouge les descendants. Les versions pleines correspondent respectivement aux parents et aux enfants du nœud B.

chemin est une succession de nœuds distincts liés par des arêtes. Dans cette thèse et sauf mention contraire nous nous consacrerons uniquement aux modèles orientés acycliques, c'est-à-dire qu'aucune variable ne descend d'elle-même.

Lorsque l'on ajoute au graphe une loi de probabilité compatible $\mathbb{P}$, on définit un modèle graphique probabiliste $\mathrm{M} = (\mathcal{G}, \mathbb{P})$. Cette loi de probabilité est compatible si les indépendances conditionnelles respectent un critère de séparation graphique, la d-séparation. Si ces indépendances conditionnelles sont respectées dans le graphe, on dira alors que la loi $\mathbb{P}$ est Markov par rapport au graphe $\mathcal{G}$. En particulier, dans le cadre des réseaux bayésiens, on a alors le critère de factorisation suivant :

$$\mathbb{P}(\boldsymbol{x}) = \prod_{i=1}^{n} \mathbb{P}\left(x_i | x_{\mathtt{pa}_i}\right).$$

Ce lien entre loi de probabilité et graphe permet de relier des indépendances conditionnelles à une représentation graphique séduisante car facile à interpréter. En effet, le critère de d-séparation suivant relie la structure du graphe aux indépendances conditionnelles de la loi de probabilité associée.

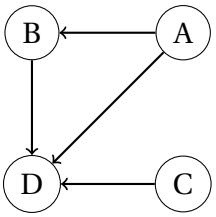
Definition 1 *Soit $\mathcal{G}$ un graphe orienté acyclique. Soit* A,B *et* C *trois sous-ensembles de noeuds disjoints. On dit qu'un chemin* $\pi = (i_1, \ldots, i_{\mathrm{N}})$ *entre* A *et* B *est bloqué par* C *si l'une des deux conditions suivantes est vraie :*

- *il existe un noeud* $i_k \in \mathrm{C}$ *tel que* $i_{k-1} \to i_k \to i_{k+1}$, $i_{k-1} \leftarrow i_k \leftarrow i_{k+1}$ *ou* $i_{k-1} \leftarrow i_k \to i_{k+1}$.
- *on a* $i_{k-1} \to i_k \leftarrow i_{k+1}$ *tel que* i_k *ou un de ses descendants appartiennent à* C.

Si tous les chemins entre A *et* B *sont bloqués par* C*, on dira que* C *d-sépare* A *et* B*, noté* $A \perp\!\!\!\perp_{d\text{-}sep} B | C$.

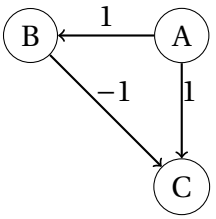
Par construction, ce critère de d-séparation implique les différentes indépendances conditionnelles.

Toutes les lois de probabilités ne peuvent pas se factoriser dans un graphe orienté acyclique sans perte d'information sur les différentes relations d'indépendance conditionnelle. Dans le cas où aucune information n'est perdue, on dira alors que la loi $\mathbb{P}$ est *fidèle* au graphe $\mathcal{G}$. Dans le cas de dépendance linéaire entre les variables assujetties de lois standards, la fidélité est une condition vraie presque sûrement.



Exemple 1 *On pose* $\mathbf{X} = (X_A, \ldots, X_D)$ *un vecteur aléatoire suivant une loi jointe* $\mathbb{P}$*. Cette loi sera Markov par rapport au graphe ci-contre si et seulement si elle admet une factorisation de la forme :*

$$\mathbb{P}(\mathbf{X}) = \mathbb{P}(X_A)\,\mathbb{P}(X_B|X_A)\,\mathbb{P}(X_C)\,\mathbb{P}(X_D|X_A, X_B, X_C)$$



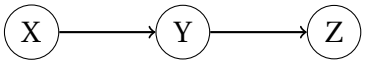
Exemple 2 *Le graphe ci-contre représente différents liens linéaires, avec la valeur de ces liens linéaires indiquée sur les arêtes. On pose* X_A, X_B, X_C *des variables aléatoires qui suivent une loi Markov par rapport au graphe ci-contre. On peut pourtant construire une loi qui n'est pas fidèle en induisant une indépendance entre* X_B *et* X_C*. Celle-ci n'est pas traduite par une d-séparation dans le graphe.*

$$\begin{array}{rcl} X_A & \sim & \mathcal{N}(0,1) \\ X_B|X_A & \sim & \mathcal{N}(X_A, 1) \\ X_C|X_A, X_C & \sim & \mathcal{N}(X_A - X_B, 1) = \mathcal{N}(0,1) \end{array}$$

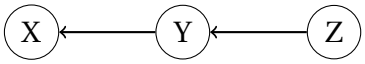
Ainsi pour pouvoir effectuer une inférence du réseau à partir de données, nous devons donc faire deux hypothèses : l'hypothèse de Markov et celle de fidélité. Dans ce cas, on va pouvoir associer la représentation graphique via le critère de d-séparation et la loi de probabilité.

Malheureusement de nombreux DAGs sont statistiquement indistinguables, comme le montre l'exemple suivant :

Exemple 3 *Soit* $\mathbb{P}$ *une mesure de probabilité se factorisant dans le graphe A suivant. Alors elle se factorise dans les deux graphes.*



Graphe A



Graphe B

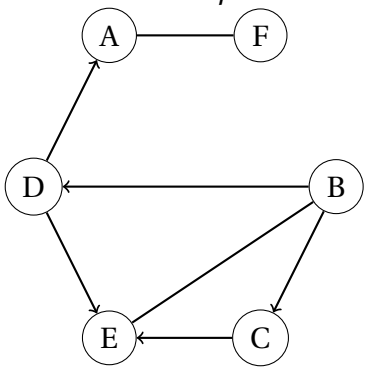
En effet, sans perte de généralité, supposons que $\mathbb{P}$ *se factorise dans le graphe* A*.*

$$\begin{array}{rcl} \mathbb{P} & = & \mathbb{P}(Z|Y)\,\mathbb{P}(Y|X)\,\mathbb{P}(X)\,, \\ \mathbb{P} & = & \mathbb{P}(Z,Y)\,\mathbb{P}(Y,X)\,\frac{\mathbb{P}(X)\,\mathbb{P}(Z)}{\mathbb{P}(X)\,\mathbb{P}(Y)\,\mathbb{P}(Z)}\,, \\ \mathbb{P} & = & \mathbb{P}(X|Y)\,\mathbb{P}(Y|Z)\,\mathbb{P}(Z)\,. \end{array}$$

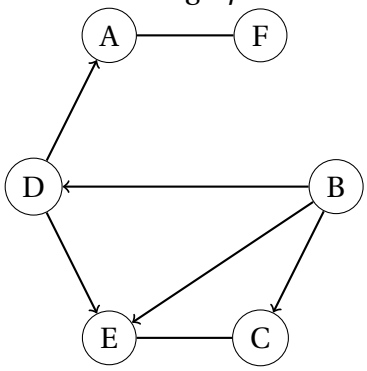
Ainsi $\mathbb{P}$ *se factorise dans le graphe* B*.*

Pour traduire ce type d'équivalence, la notion de graphe acylique partiellement orienté, en anglais PDAG pour *Partially Directed Acyclic Graph*, ou aussi nommé *chain graph* doit être introduite. Un PDAG est un graphe mixte, contenant à la fois des arêtes orientées et non orientées. De plus, un graphe G est PDAG s'il ne contient pas de cycle orienté et si ses composantes connexes orientées forment une partition de sous-graphes non-orientés [56]. Une composante connectée S est un sous-ensemble maximal de sommets tels que pour tout $(x, y) \in S$, il existe un chemin partiellement orienté reliant x à y, c'est-à-dire du type $x \to \ldots - \ldots \to \ldots - y$, dans lequel le sens de la flèche compte. Le chemin peut aussi n'être constitué que d'arêtes non-orientées ou orientées, du moment que l'intégralité des nœuds sont reliés, deux à deux et dans les deux sens. Une conséquence de cette définition est la suivante : si on essaye de remplacer chaque arête non-orientée par une arête orientée, alors il existe au moins une configuration telle que l'on obtienne un DAG. L'exemple suivant explicite cette définition du PDAG :

Exemple 4 *Ci-dessous, le graphe A n'est pas un PDAG. En effet, si on décompose le graphe en composantes connectées, on obtient les éléments :* (A, F) , (B, C, D, E). *Le sous graphe obtenu par la deuxième composante n'étant pas un graphe non-orienté, il ne s'agit pas d'un PDAG. Le graphe B est bien un PDAG. Les composantes connectées sont :* (A, F) , (B) , (D) , (C, E). *Chacune de ces composantes connectées induisent bien un sous-graphe non-orienté.*



Graphe A



Graphe B

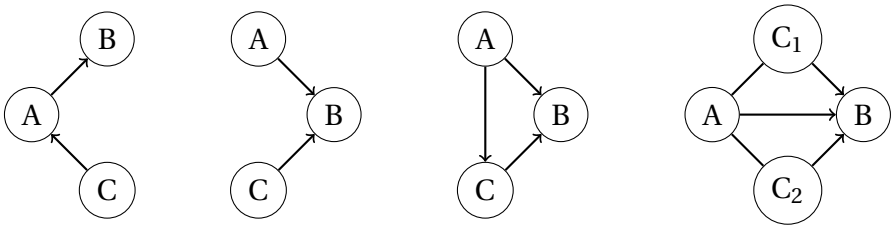
Ceux-ci peuvent être utilisés pour représenter l'ensemble des graphes appartenant à la même classe d'équivalence. Un théorème permet d'utiliser cette représentation, en se basant sur les v-structures, c'est à dire les structures du type A → B ← C, mais tel qu'il n'existe pas de lien direct entre A et C.

Proposition 1 *[102] Deux DAGs sont équivalents s'ils ont le même squelette et les mêmes v-structures.*

Les v-structures spécifiant en partie les différentes classes d'équivalences, le PDAG peut représenter un graphe incluant la classe d'équivalences. Une version plus précise, le CPDAG, pour *Completed Partially Directed Acyclic Graph* est utilisée pour représenter spécifiquement cette classe d'équivalence. Il est nécessaire pour l'introduire de définir la notion de graphe cordal. La définition la plus simple pour ce type de graphes non-orientés est graphique : un graphe cordal est un graphe tel que pour tout cycle comprenant 4 nœuds ou plus, il existe une corde, c'est-à-dire une arête entre 2 nœuds non adjacents. De ce fait, si un graphe est cordal et qu'il contient des cycles, ceux-ci sont en fait une succession de petits triangles, lui donnant aussi le nom de graphe triangulé. D'ailleurs, le processus qui consiste à transformer un graphe non-orienté quelconque en graphe cordal s'appelle la triangulation. On peut maintenant définir le CPDAG.

Définition 1 *[4] Un CPDAG* G *est un PDAG tel que :*

- G *est cordal, c'est-à-dire que chaque cycle non-orienté comprenant 4 sommets ou plus possède une corde.*
- *la configuration* A → B – C *n'apparaît pas comme un sous graphe de* G.
- *Chaque arête orientée* A → B ∈ G *apparaît dans au moins l'une des configurations suivantes :*



A noter que le nombre de DAGs et de CPDAGs croît de manière superexponentielle en fonction du nombre de noeuds p, et celle des CPDAGs croît, en moyenne, 4 fois moins vite [36; 82]. La classe d'équivalence d'un CPDAG peut contenir plus de $p!$ DAG, pour le CPDAG complet, à un seul, par exemple pour le CPDAG vide. Sur ces bases, nous pouvons passer à la question de l'inférence de ces réseaux. Celle-ci peut se faire directement sur le DAG, sur le CPDAG ou sur l'ordre topologique des noeuds. Des données d'interventions peuvent être adjointes à cette inférence : dans ce cas la classe d'équivalence interventionnelle peut être réduite [43].

1.3 Causalité

Le terme de causalité est relégué au second plan depuis les fondations statistiques modernes. Bien que ce soit toujours l'objectif ultime dans bon nombre de domaines associés, comme l'économie ou la génétique, on lui préfère notamment le terme de corrélation, qui jouit d'instruments théoriques bien établis. Néanmoins depuis les années 80 on voit se développer une réelle théorie statistique de la causalité [74]. Dans cette partie, j'explique pourquoi la notion de causalité est importante en me basant sur des travaux philosophiques, ce qu'elle apporte de plus comparée à la corrélation et les différentes écoles statistiques qui s'attaquent à cette notion de causalité.

Fondement philosophique

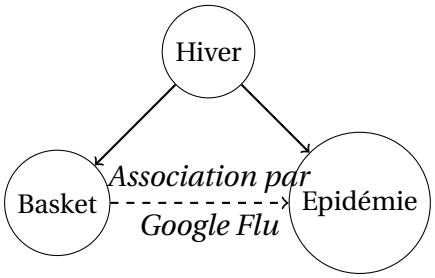
La notion de causalité a été longuement interrogée en philosophie. D'abord évidente, elle fut progressivement attaquée en même temps que le développement des sciences modernes. En effet dans ces dernières, et notamment en physique, la notion de cause a disparu progressivement, laissant place à des lois purement symétriques ou probabilistes. L'un des grands penseurs contemporain de la causalité, Bertrand Russel, juge que cette notion de causalité n'est fondée que sur la persistance d'anciennes croyances [86]. Pour lui, la notion de cause induit un objet actif et un objet passif. Or les interactions moléculaires et biologiques sont des interactions symétriques. Par exemple, le développement du microbiote intestinal se fait via des synergies et des antagonismes, à travers la modification du milieu. Un récepteur hormonal n'est pas moins actif qu'une hormone. Comment justifier alors de l'utilisation du terme cause ?

L'idée de la causalité est pourtant séduisante pour une double raison : d'abord par son côté intuitif, que tout le monde saisit, et ensuite par la possibilité que cela nous donne d'interagir avec le système. Nancy Carwright suggère de justifier le terme de cause en proposant de se fonder sur notre capacité d'intervention sur les systèmes [14]. La cause n'est alors plus un objet actif, mais une variable contrôlée par nos interventions. Cela donne lieu à une des branches moderne de la causalité en philosophie : la causalité interventionniste [52]. C'est dans cette conception que les théories causales en statistiques ont été développées.

Les différentes démarches causales

L'analyse de chemin, en tant qu'ancêtre de la théorie des graphes, cherchait déjà à différencier la causalité et la corrélation [107]. En effet, après le développement rapide des probabilités, d'aucuns ont pu estimer qu'une corrélation entre deux variables indiquait un lien de cause à effet. Or cela est bien sûr faux dans l'immense majorité des cas. L'exemple suivant le souligne.

Exemple 5 Google flu trends *fut un algorithme développé par Google qui tentait de prévoir des épidémies en se basant sur les réseaux sociaux. Cet algorithme a rapidement été abandonné du fait de la grande approximation des résultats. Parmi les erreurs rencontrées, celle issue des variables confondantes est majeure.*



On voit très bien par cet exemple qu'une corrélation entre deux variables n'implique en rien une causalité. En l'espèce, décider de ne pas organiser le tournoi de basket une année n'influerait en rien la probabilité d'éclosion de l'épidémie.

En médecine, on recherche pourtant précisément ce lien de cause à effet. Les études randomisées en double aveugle permettent d'éviter l'écueil de ces variables confondantes. Le principe est le suivant : on souhaite tester une molécule. On va préparer deux médicaments : l'un avec la molécule, et l'autre sans, par exemple juste de l'eau. La distinction entre le médicament avec principe actif et le placébo n'est pas connue des thérapeutes : ce sont les premiers aveugles. On va ensuite attribuer de façon aléatoire les produits aux cobayes, les seconds aveugles : ceux-ci ne savent pas s'ils ont le placébo ou le médicament actif. De cette façon, ni l'équipe thérapeutique, ni les cobayes ne peuvent influencer les résultats. On peut ensuite travailler sur les résultats, par exemple en regardant une différence d'effets entre les groupes placébo et principe actif. Cette façon de procéder est le reflet d'une première école statistique [85].

Dans de rares cas, nous disposons de données temporelles. En économie, de nombreuses séries chronologiques sont disponibles. On peut alors être tenté de regarder s'il y a une causalité temporelle entre ces différentes séries : c'est la causalité au sens de Granger [38].

Cette thèse suit une autre voie, qui a été développée par Pearl [74]. Celle-ci est notamment basée sur l'introduction d'un nouvel opérateur statistique, l'opérateur *do*. Cet opérateur sert à modéliser les interventions effectuées, que ce soit les *knock-down*, dans le cadre d'ARN interférents, ou des *knock-out*, par exemple via CRISPR-Cas9. Elle a l'avantage d'être plus interprétable que les deux précédentes approches, au prix d'un effort de modélisation plus grand.

Modèle d'équations structurelles

Un réseaux bayésien ne traduit pas nécessairement des relations causales. Pour travailler dans ce cadre, il faut spécifier ces liens. L'approche la plus naturelle est d'utiliser les modèles d'équations structurelles (SEM, pour *Structural Equation Modeling*).

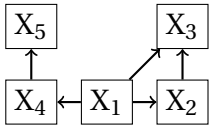
Définition 2 *Un modèle d'équations structurelles est un couple* $\left(\mathbf{S}, \mathbb{P}^{\mathrm{N}}\right)$, *tel que* $\mathbf{S} = (\mathrm{S}_1, \ldots, \mathrm{S}_{\mathrm{N}})$ *est une collection de* N *équations. Ces équations lient les variables :*

$$\forall i \ in \ \{1, \mathrm{N}\} \quad \mathrm{X}_i = f_i\left(pa(\mathrm{X}_i), \epsilon_i\right)$$

où $\boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_{\mathrm{N}}) \sim \mathbb{P}^{\mathrm{N}}$.

De plus, si aucune boucle de rétroaction n'existe, on parle d'un modèle d'équations structurelles récursif. On peut alors associer ce modèle avec un graphe orienté acyclique. Le triplet $(\mathcal{G}, \mathbb{P}, \mathbf{S})$ forme alors un réseau bayésien causal. Dans la suite, on utilisera uniquement des relations linéaires entre les différentes variables.

Exemple 6 *Soit* $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathrm{I}_5)$. *Les équations structurelles couplées aux erreurs résiduelles déterminent une loi jointe. Cette loi jointe est factorisable dans le graphe ci-contre. On remarquera qu'aucun cycle n'est présent.*



$$\begin{aligned}
\mathrm{X}_1 &= \epsilon_1 \\
\mathrm{X}_2 &= 0.3\mathrm{X}_1 + \epsilon_2 \\
\mathrm{X}_3 &= -1.9\mathrm{X}_1 + 0.3\mathrm{X}_2 + \epsilon_3 \\
\mathrm{X}_4 &= 2\mathrm{X}_1 + \epsilon_4 \\
\mathrm{X}_5 &= -4\mathrm{X}_4 + \epsilon_5
\end{aligned}$$

Ce modèle permet d'illustrer l'opérateur *do* : cet opérateur fixe la loi marginale d'une variable, indépendamment des autres.

Définition 3 *Soit* $\mathrm{X}_1, \ldots, \mathrm{X}_n$ *des variables aléatoires suivant une loi* $\mathbb{P}$ *absolument continue par rapport à la mesure de Lebesgue. Soit* $\mathrm{J} \in \{1, \ldots, \mathrm{N}\}$ *et* π *une loi de probabilité. On définit la loi interventionnelle* $\mathrm{X}|\, do\left(\mathrm{X}_\mathrm{J} = \pi\right)$ *de la façon suivante :*

$$\mathbb{P}\left(\mathrm{X}_1, \ldots, \mathrm{X}_\mathrm{N}|\, do\left(\mathrm{X}_\mathrm{J} = \pi\right)\right) = \prod_{i \notin \mathrm{J}} \mathbb{P}\left(\mathrm{X}_i |\, pa(\mathrm{X}_i)\right) \prod_{j \in \mathrm{J}} \pi\left(\mathrm{X}_j\right)$$

La loi π dépend de la nature de l'intervention : en particulier s'il s'agit d'une invalidation génique par exemple, il s'agira d'une masse de Dirac. Les méthodes présentées dans la suite de ce manuscrit sont basées sur cet opérateur.

1.3.1 Inférence

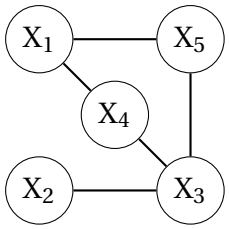
Graphical Lasso

Avant de passer aux algorithmes d'inférence spécifique aux réseaux orientés, il est nécessaire de parler du *graphical Lasso*. Cette méthode consiste à estimer une matrice de précision parcimonieuse, et par voie de conséquence, un modèle graphique gaussien parcimonieux. Dans ce cas, le graphe non-orienté peut être traduit directement de la matrice de précision $\Omega = \Sigma^{-1}$ qui donne les indépendances conditionnelles d'intérêt pour les modèles non-orientés. En effet en considérant $\mathbf{X} = (\mathrm{X}_1, ..., \mathrm{X}_n)$, si $\left(\Sigma^{-1}\right)_{i,j} = 0$ alors $\mathrm{X}_i \perp\!\!\!\perp \mathrm{X}_j | \mathbf{X} - \left\{\mathrm{X}_i, \mathrm{X}_j\right\}$. La représentation graphique sous forme de graphe non-orienté est issue de ces indépendances conditionnelles.

Exemple 7 *Soit* Σ^{-1} *la matrice de précision associée aux variables aléatoires* $(\mathrm{X}_1, \mathrm{X}_2, \mathrm{X}_3, \mathrm{X}_4, \mathrm{X}_5)$ *suivante :*

$$\Sigma^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0.5 & 0.25 \\ 0 & 4 & 1 & 0 & 1 \\ 0 & 1 & 0.5 & 0.5 & 0 \\ 0.5 & 0 & 0.5 & 2 & 0 \\ 0.25 & 1 & 0 & 0 & 1 \end{pmatrix}$$

Le modèle graphique gaussien correspondant est le suivant.



Graphe non-orienté associé à $\Sigma-1$

Ce résultat est intéressant même lorsque l'on souhaite travailler avec un modèle orienté du fait du mécanisme de moralisation. On dit qu'on moralise un graphe orienté lorsque l'on ajoute une arête entre les parents de tous les noeuds, puis qu'on enlève l'orientation des arêtes. Une loi se factorisant dans un graphe orienté $\mathcal{G}$ se factorisera dans le graphe non-orienté issue de la moralisation $\mathcal{G}_M$, même si quelques indépendances conditionnelles ne sont pas plus traduites. Ainsi le squelette d'un réseau bayésien sera strictement dans le graphe issu d'un modèle graphique gaussien.

L'algorithme du *graphical Lasso* propose d'utiliser une pénalité de type L1, c'est-à-dire une pénalité sur les valeurs absolues des éléments de la matrice de précision. Ceci dans le but d'effectuer une sélection sur les éléments de cette matrice, et au final d'obtenir la représentation graphique la plus parcimonieuse possible. L'avantage de cette représentation parcimonieuse est qu'elle est interprétable, au risque de simplifier les mécanismes d'un éventuel phénomène physique [106], socio-économique [22] ou biologique [54].

Friedman et al. [30] proposent donc de maximiser la fonction suivante :

$$\log\det\left(\Sigma^{-1}\right) - \mathrm{Tr}\left(S\Sigma^{-1}\right) - \lambda||\Sigma^{-1}||_1, \tag{1.1}$$

sur l'ensemble des matrices symétriques définies positives. Σ^{-1} est la matrice de précision, et S la matrice de covariance empirique. La première partie de cette fonction correspond à la log-vraisemblance, et traduit l'attachement aux données. La seconde partie sert à favoriser les modèles parcimonieux. En particulier, le paramètre $\lambda \in \mathbb{R}^+$ est un paramètre de régularisation à calibrer. Lorsqu'il est égal à 0, on maximise la vraisemblance et en pratique l'ensemble des paramètres sont différents de 0. Dans ce cas, le graphe associé sera plein. Si λ tend vers l'infini, le terme de pénalisation l'emporte largement sur la vraisemblance, et la matrice de précision est estimée diagonale. Dans ce cas, le modèle vide est considéré.

On peut transformer le problème exprimé en (1.1) via l'utilisation de la norme duale. Dans le cas présent, on a notamment : $||X||_1 = \max_Z \mathrm{Tr}\,(XZ)$ sous la contrainte suivante : $||Z||_\infty \leq 1$. En injectant cette formulation dans (1.1), le problème est donc de maximiser en X dans le domaine des matrices symétriques définies positives la fonction suivante :

$$\min_{||Z||_\infty<\lambda} \log\det(X) - \mathrm{Tr}\,(X(S+Z))\,. \tag{1.2}$$

En échangeant la minimisation et la maximisation, le problème peut-être résolu en fonction de Z et via l'étude de la dérivée. En intégrant la solution $X^* = (S+Z)^{-1}$ dans l'expression restante, et en posant $W = S + Z$, on obtient le problème dual :

$$\min_{||W-S||_\infty<\lambda} \log\det W. \tag{1.3}$$

Ceci est alors résolu itérativement sur les colonnes de W, W représentant la matrice de covariance. Il s'agit de la méthode de descente par bloc de coordonnées. En effet on peut

effectuer la décomposition suivante, où $W_{1,1}$ est une matrice de taille $(n-1)\times(n-1)$ et $w_{2,2}$ un réel, de même pour la décomposition de S :

$$W = \begin{pmatrix} W_{1,1} & w_{1,2} \\ w_{1,2}^T & w_{2,2} \end{pmatrix} \quad S = \begin{pmatrix} S_{1,1} & s_{1,2} \\ s_{1,2}^T & s_{2,2} \end{pmatrix}.$$

En utilisant ensuite le complément de Schur, on obtient la fonction coût suivante :

$$\log\det\left(w_{2,2} - w_{1,2}^T W_{1,1}^{-1} w_{1,2}\right) + \log\det W_{1,1}. \tag{1.4}$$

Ainsi si on ne considère que le vecteur $w_{1,2}$, on s'aperçoit qu'il est la solution du problème :

$$w_{1,2} = \text{argmin}_{y;||y-s_{1,2}<\lambda||} y^T W_{1,1}^{-1} y. \tag{1.5}$$

En regardant à nouveau le dual, on obtient pour chaque coordonnée le problème suivant :

$$\min_{\beta} \frac{1}{2}||W_{11}^{1/2}\beta - b||^2 + \rho||\beta||_1. \tag{1.6}$$

Il s'agit donc d'une régression Lasso pour la coordonnée en question, en utilisant les autres éléments de la matrice de covariance comme variables. C'est ce problème qui est résolu itérativement pour chaque coordonnée. On fait ensuite la correspondance avec $w_{1,2} = W_{1,1}\beta$. La preuve de dualité forte n'est pas triviale mais n'est pas importante pour comprendre la façon dont s'opère l'algorithme, les conditions d'existence et d'unicité sont satisfaites du fait de la convexité du problème. De plus la solution obtenue est bien garantie d'être symétrique définie positive. L'algorithme de descente par coordonnée s'écrit alors :

Algorithme 1 : Graphical Lasso

Entrées : Matrice de covariance empirique S, paramètre de régularisation λ
début
  Commencer avec $W = S + \lambda I$.
tant que Il n'y a pas convergence **faire**
  Résoudre itérativement le problème Lasso sur chaque coordonnée.
  Modifier W en remplaçant $w_{1,2}$ et $w_{1,2}^T$ par la solution trouvée.
Sorties : Matrice de covariance empirique pénalisée W.

L'algorithme est très rapide, même pour une centaine de nœuds. Cependant en pratique se pose un double problème. D'une part le choix de la constante de régularisation nécessite d'avoir une idée de la solution voulue, par exemple en terme de parcimonie ou en terme de garantie sur des aspects de la solution. D'autre part, la solution obtenue n'est pas robuste : deux jeux de données issus d'un même modèle peuvent donner des résultats drastiquement différents, en fonction de la taille des jeux de données et de la parcimonie attendue. Une validation est alors éventuellement nécessaire.

Algorithme PC

L'idée est d'utiliser les indépendances conditionnelles et sa traduction graphique, la d-séparation. Supposons que nous disposions de p variables suivant une loi de probabilité jointe factorisable dans un DAG. Si nous disposons de l'ensemble des indépendances conditionnelles, il est facile de retrouver un graphe correspondant, plus exactement la classe d'équivalence associée, le CPDAG.

Étudier l'ensemble des indépendances conditionnelles est cependant une tâche fastidieuse. Ainsi pour un réseau de n variables le nombre d'indépendances conditionnelles différentes à regarder peut s'établir de la sorte : il existe $\begin{pmatrix} 2 \\ n \end{pmatrix}$ couples de variables à regarder, auquel multiplier $\sum_{k=0}^{n-2} \begin{pmatrix} k \\ n-2 \end{pmatrix}$ potentiels ensembles séparateurs. Il peut y avoir cependant une certaine redondance dans l'information fournie. Dans un graphe orienté, la propriété de Markov nous dit par exemple qu'un nœud d'intérêt est entièrement indépendant du reste du graphe conditionnellement à ses parents. Se faire une idée des différents parents des différentes nœuds permet donc de limiter le nombre d'indépendances conditionnelles testées. Connaître les enfants de chaque nœud est aussi judicieux, notamment pour détecter les v-structures. On s'intéressera donc à l'ensemble adj(X) des nœuds adjacents du nœud x, pour tout x. L'algorithme PC proposé par Spirtes et al. [98] se décrit donc comme suit, avec **EnsSep** (X, Y) l'ensemble de séparations entre les nœuds X et Y.

Algorithme 2 : PC

Entrées : Oracle pour les différentes indépendances conditionnelles
début
  Initialiser C comme le graphe complet, n=0.
pour tout couple ordonné (X, Y) adjacent tel que |adj(X) /Y| ≥ n **faire**
  pour tout ensemble S ∈adj(X) tel que |S| = n **faire**
    si X ⫫ Y|S **alors**
      Retirer l'arête X − Y dans C et ajouter S dans **EnsSep** (X, Y) et **EnsSep** (Y, X).
  n=n+1
Marquer les v-structures, c'est-à-dire :
pour les triplets (X, Y, Z) tels que X, Y sont adjacents dans C, Y, Z sont adjacents mais X et Z ne le sont pas **faire**
  si Y ∉ ***EnsSep***(X, Z) **alors**
    Remplacer dans C la chaine X − Y − Z par X → Y ← Z.
Compléter le PDAG obtenu, c'est-à-dire :
répéter
  Si A → B, B et C sont adjacents, A et C ne sont pas adjacents, alors remplacer B − C par B → C. S'il existe un chemin orienté entre A et B, et que A et B sont adjacents, alors remplacer A−B par A → B.
jusqu'à jusqu'à ce qu'aucune arête ne puisse être orientée;
Sorties : CPDAG C

A partir d'un oracle pour les indépendances conditionnelles, l'algorithme PC commence par créer le squelette du CPDAG : il s'agit de la partie gourmande en temps de calcul. Pour se faire, en commençant par le graphe plein, on cherche itérativement des ensembles séparateurs de plus en plus grands. Lorsqu'un tel ensemble est trouvé, on le garde en mémoire et on peut affiner le graphe en enlevant une arête, et il n'est alors plus nécessaire de tester ce couple pour des ensembles séparateurs plus grands. Ceux-ci étant inclus dans l'ensemble des noeuds adjacents, on explore l'intégralité des possibilités. Dans un deuxième temps, on passe du squelette à un PDAG en orientant les v-structures. Ceci peut être fait en gardant les ensembles séparateurs en mémoire. Enfin, on passe dans la dernière partie de l'algorithme du PDAG au CPDAG en complétant le PDAG.

En pratique, nous ne disposons pas d'oracles pour les indépendances conditionnelles. Par contre, on peut utiliser des tests sur les corrélations partielles. Dans le cas Gaussien, si

un coefficient de corrélation partielle est égal à 0, cela traduit une indépendance conditionnelle. En posant r la corrélation partielle empirique par rapport à S et n le nombre d'observations, on peut écrire la transformation de Fisher suivante :

$$Z = \frac{\sqrt{n - |S| - 3}}{2} \log \frac{1 + r}{1 - r}. \tag{1.7}$$

Celle-ci est distribuée asymptotiquement selon une loi normale sous l'hypothèse que le coefficient de corrélation partielle est égal à 0. On peut donc utiliser cette quantité pour effectuer un test d'un certain niveau α. Cependant ce contrôle n'est pas réellement une garantie statistique. Les tests devant être faits successivement, ils dépendent de l'ordre dans lequel ils sont réalisés. En particulier, si on change l'ordre des variables, les ensembles séparateurs admissibles pour chaque nœud sont différents et les résultats des tests peuvent être en conséquence différents. Il s'agit donc plus d'un paramètre de forme plus que d'un paramètre de contrôle de l'erreur statistique. Cet algorithme dispose d'une implémentation dans un package R, `pcalg`. Son utilisation est simple : il suffit d'entrer dans les arguments une statistique exhaustive résumant les données, ainsi que les tests d'indépendance que l'on souhaite réaliser. Une fonction annexe est déjà proposée dans le cas de données gaussiennes, il s'agit de `gaussCItest`. Deux autres arguments sont nécessaires : le paramètre de forme α ainsi que le nombre de nœuds. L'expression est alors la suivante :

```
pc.graph<-pc(suffStat=list(C=cor(data),n=n),gaussCItest,0.15,p).
```

On peut voir l'influence du paramètre α dans la figure 1.4.

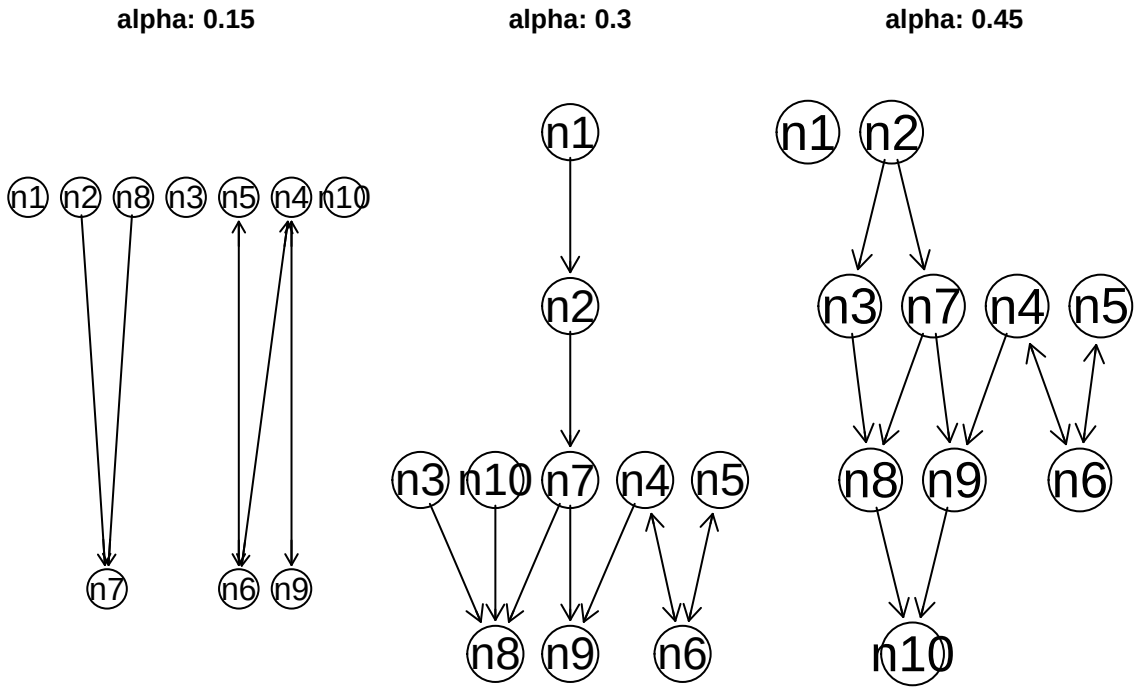


FIGURE 1.4 – Différents graphes obtenus selon différentes valeurs du paramètre de forme.

Comme on peut le voir, on peut moduler à l'envie l'aspect parcimonieux du graphe de sortie. Aussi, nous n'avons pas d'indice de confiance pour choisir entre ces différentes représentations.

Greedy Equivalence Search

Une façon peut-être plus naïve de choisir un DAG est de simplement choisir un score, puis de chercher exhaustivement le graphe maximisant ce score. Un bon score est par exemple le score BIC, transformant la log-vraisemblance ℓ en log-vraisemblance pénalisée ℓ_{pen} :

$$\ell_{\text{pen}} = \ell - \frac{1}{2} n \log|\mathrm{E}|,$$

où n est le nombre d'observations, et $|\mathrm{E}|$ est le nombre d'arêtes. Cette pénalité BIC est largement documentée et utilisée en pratique pour effectuer de la sélection de modèles [91]. Malheureusement le nombre de DAG croit superexponentiellement avec le nombre de noeuds, comme le montre le tableau 1.1.

# Nœuds	# DAGs*
1	1
2	3
3	25
4	543
5	29281
6	3781503
7	1138779265
8	783702329343
9	1213442454842881
10	4175098976430598143
11	31603459396418917607425
12	521939651343829405020504063
13	18676600744432035186664816926721
14	1439428141044398334941790719839535103

TABLEAU 1.1 – Nombre de DAGs en fonction du nombre de noeuds

Il est donc illusoire, pour un nombre de noeuds supérieur à une dizaine, d'utiliser cette procédure. Chickering [15] propose un algorithme glouton pour rechercher le meilleur graphe. Un tel algorithme est basé sur des heuristiques et n'a aucune garantie de trouver le maximum global.

Le fonctionnement est le suivant : On commence par un graphe C vide que l'on met à jour progressivement selon une procédure progressive-rétrogressive. Dans la partie progressive, on tente d'ajouter une arête à C. Cet ajout ne peut être fait que sous la contrainte que C reste un DAG. Cela nous donne un ensemble $\mathscr{E}^+$ de DAG qui appartiennent à la

classe d'équivalence de C, plus une arête. Ainsi, on peut obtenir le déroulement suivant :

Algorithme 3 : Greedy Equivalence Search

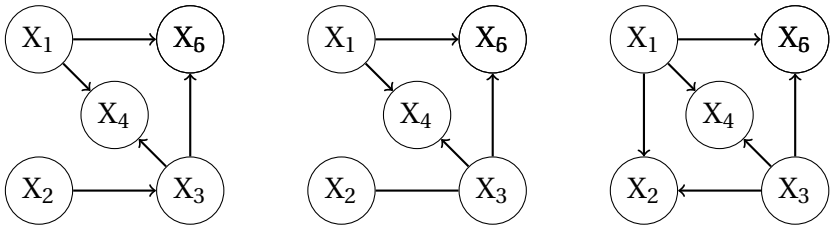
Entrées : Données D, Fonction de score S
Sorties : CPDAG C
début
  Initialiser C comme le graphe vide.
répéter
  Ajouter une arête e dans C tel que cela résout :

$$\text{argmax}_e S(C+e, D)$$

jusqu'à ce que plus aucune amélioration n'est possible;
répéter
  Enlever une arête e dans C tel que cela résout :

$$\text{argmax}_e S(C-e, D)$$

jusqu'à ce que plus aucune amélioration n'est possible;
retourner C



Itération i Classe d'équivalence Itération $i+1$

On remarque dans ce cas que l'arête entre les nœuds X_2 et X_3 a changé d'orientation, en plus de l'ajout de l'arête $X_1 \rightarrow X_2$. Ceci est dû au fait que la classe d'équivalence admet les deux orientations, et donc l'ensemble des graphes admissibles pour l'étape $i+1$ admettent l'une ou l'autre orientation. Lorsque l'algorithme n'arrive plus à améliorer le score, alors la phase rétrogressive commence : il s'agit du procédé symétrique. A chaque itération, une nouvelle arête est enlevée de manière à maximiser le score. Les graphes candidats appartiennent alors à $\mathscr{E}^-$, l'ensemble des DAGs appartenant à la classe d'équivalence moins une arête. Le résultat est alors retourné.

Max-min Hill Climbing

Tsamardinos et al. [101] ont proposé un algorithme mixte, qui marie à la fois l'approche gloutonne et contrainte. L'objectif ici est de trouver dans une première partie les nœuds adjacents, c'est-à-dire soit les parents, soit les enfants, de chaque variable. En effet, on peut remarquer que dans les classes d'équivalence le squelette reste constant, et ainsi les adjacences restent constantes. Pour pouvoir trouver celles-ci, on se base sur l'association minimale, c'est-à-dire la quantité $\forall T, X \in V, \quad Z \subset V \quad \min_{S \subset Z} \texttt{Assoc}(X;T|S)$, où `Assoc` est une mesure de l'association, par exemple la corrélation. L'algorithme se compose de deux phases et se présente comme suit.

Lors de la première phase, on ajoute toutes les variables pouvant être adjacentes à T. On utilise pour ça l'heuristique max-min : on ajoute, tant que c'est possible, les variables qui maximisent l'association minimale avec T. Il s'agit de l'association maximum que l'on

Algorithme 4 : MMPC

Entrées : Noeud d'intérêt T, Données $\mathscr{D}$
Sorties : $\mathscr{A}$ =adj(T)
début
  Initialiser $\mathscr{A} = \emptyset$
répéter
  $\texttt{assocF} = \max_X \min_{S \in \mathscr{A}} \texttt{Assoc}(T, X|S)$
  $F = \text{argmax}_X \min_{S \in \mathscr{A}} \texttt{Assoc}(T, X|S)$
  si $assocF \neq 0$ **alors**
    $\mathscr{A} = \mathscr{A} \cup F$
jusqu'à ce que $\mathscr{A}$ n'évolue plus;
pour $X \in \mathscr{A}$ **faire**
  si $\exists S \in \mathscr{A}$ tel que $X \perp\!\!\!\perp T|S$ **alors**
    $\mathscr{A} = \mathscr{A} \setminus X$
retourner $\mathscr{A}$

a dans le pire des cas, c'est-à-dire lorsqu'on tente de choisir un sous-ensemble de nœuds adjacents afin d'être au plus proche de l'indépendance. Dans un second temps, on retire les nœuds qui sont indépendants conditionnellement à un sous-ensemble de nos nœuds adjacents. Cela donne en sortie un ensemble de variables adjacentes, pour une variable donnée.

L'idée est ensuite d'appliquer cet algorithme sur chaque nœud, puis de lancer un algorithme glouton pour estimer notre réseau.

Algorithme 5 : MMHC

Entrées : Données $\mathscr{D}$
Sorties : DAG D
pour $X \in V$ **faire**
  $A_X = \text{MMPC}(X, \mathscr{D})$
pour $X \in V$ **faire**
  pour $T \in \mathscr{A}_X$ **faire**
    si $X \notin \mathscr{A}_T$ **alors**
      $\mathscr{A}_X = \mathscr{A}_X \setminus T$
Lancer un algorithme glouton sur l'espace restreint par les adjacences.
retourner Le DAG D obtenant le score maximum

Une fois les adjacences calculées, une phase corrective appliquée : pour pouvoir s'assurer théoriquement d'une possible convergence, les adjacences doivent respecter le principe de symétrie. En effet, de potentiels faux positifs peuvent apparaître dans le cas où cette symétrie n'est pas effective. Dans ce cas, on retire le potentiel lien entre les deux nœuds. L'algorithme glouton appliqué ensuite est similaire à l'algorithme GES : on propose d'abord des arêtes maximisant le score. Une fois le score maximisé, on itère de façon analogue en enlevant des arêtes successivement. La principale différence vient ici de l'espace considéré : ici on y ajoute les contraintes sur les adjacences. De ce fait l'algorithme converge plus vite.

Z-score

Lorsque des interventions sur chaque nœud sont disponibles, comme c'est le cas dans le benchmark DREAM4 [66], la méthode suivante semble donner les meilleurs résultats. Il s'agit ici d'estimer un réseau biologique, la littérature donnant une bonne idée de la structure d'un tel réseau. Proposée par Pinna et al. [76], elle ne nécessite pas l'hypothèse DAG mais est contrainte par la nécessité d'obtenir des données très particulières. Celles-ci doivent absolument contenir une intervention pour chaque nœud et de manière indépendante. Cela permet de calculer un score, à partir des moyennes empiriques et variances empiriques calculées comme suit, $\forall p$:

$$\begin{aligned} \mu_j &= \frac{1}{p}\sum_{i\in\{1,\dots,p\}} x_j^i, \\ \sigma_j^2 &= \frac{1}{p}\sum_{i\in\{1,\dots,p\}} \left(x_j^i - \mu_j\right)^2. \end{aligned}$$

En partant de ces quantités, on peut alors calculer le score suivant :

$$W_{i,j} = \frac{|x_j^i - \mu_j|}{\sigma_j}.$$

On remarquera que ce score ne nécessite aucune donnée "observationnelle", c'est-à-dire sans intervention. Ce score permet de choisir quelles arêtes seront présentes, grâce à un seuil λ fixé à l'avance. Ce seuil pourra être calculé par validation croisée par exemple. Une fois ce seuil appliqué, une phase d'élagage est lancée sur les arêtes détectées. Cette phase est inspirée du principe biologique suivant : si deux gènes sont connectés directement, alors il est très peu probable qu'une connexion indirecte existe entre ces deux gènes. On considérera préalablement tout cycle comme un nœud unique, puis, fort de cette affirmation, on enlèvera tout lien indirect entre deux nœuds adjacents. L'algorithme complet est décrit ci-après.

On remarquera qu'en pratique, il est illusoire d'avoir de telles données : les réseaux de régulation de gènes comportent plusieurs milliers de gènes en fonction de l'espèce. Même en prenant un sous-réseau, effectuer une intervention pour plus d'un ou deux gènes reste très rare.

MC3

L'algorithme MC3, proposé par Madigan et al. [63] dans le cas de variables discrètes, se présente sous la forme d'un algorithme de Métropolis-Hastings. Cette méthode de Monte Carlo par chaine de Markov est ici intéressante à deux égards. En premier lieu, le nombre de DAG étant trop grand, même pour un nombre raisonnable de noeuds, il est impossible d'estimer un score sur chacun d'entre eux. En second lieu, il n'y a généralement pas de garantie que le processus que l'on observe soit généré par un seul DAG, ou généré par un DAG tout court. En effet, plusieurs états d'équilibre peuvent se superposer, et des boucles de rétroactions peuvent prendre place. L'algorithme de Métropolis-Hastings peut répondre à ces deux questions : d'une part, en n'échantillonnant qu'une partie de l'espace on contourne le problème de la dimension. D'autre part, cet échantillonnage permet d'estimer une loi de probabilité plutôt qu'une réponse univoque.

Rappelons en détail l'algorithme de Métropolis-Hastings. L'objectif est de simuler une certaine loi de probabilité π.

L'algorithme est construit de telle sorte à vérifier la balance détaillée, c'est-à-dire l'équation :

$$\pi(x)\mathcal{T}_{x,y}q(x|y) = \pi(y)\mathcal{T}_{y,x}q(y|x).$$

Algorithme 6 : Z-Score

Entrées : Données $\mathscr{D}$, Seuil λ
Sorties : Réseau R
début
  Calculer W, Initialiser N comme un graphe vide
pour $i \in \{1, ..., p\}$ **faire**
  pour $j \in \{1, ..., p\}$ **faire**
    si $W_{i,j} > \lambda$ **alors**
      Ajouter l'arête $i \rightarrow j$ dans N.
pour tous les cycles $\mathscr{C}$ dans N **faire**
  Fusionner les nœuds appartenant à $\mathscr{C}$ dans N
$N' = N$
pour $i \in \{1, ..., p\}$ **faire**
  pour $j \in \{1, ..., p\}$ **faire**
    si i et j sont directement liés dans N **alors**
      S'il existe un chemin indirect entre i et j dans N, l'enlever dans N'
Ajouter $\max_{i,j \in N \setminus N'} W_{i,j}$ pour tous les éléments de W correspondant à N'.
pour $i \in \{1, ..., p\}$ **faire**
  pour $j \in \{1, ..., p\}$ **faire**
    si $W_{i,j} > \lambda$ **alors**
      Ajouter l'arête $i \rightarrow j$ dans R.
retourner R

Cela permet de s'assurer que π est la distribution stationnaire de la chaîne de Markov construite ainsi.

Dans notre cas d'intérêt, nous estimons une loi de probabilité sur l'espace des DAGs. Etant donné un DAG courant, la loi de proposition est la loi uniforme sur l'ensemble des DAGs qui diffèrent d'une arête, qu'elle soit ajoutée ou retirée.

Ce type de procédure souffre cependant d'un problème de mélange. En particulier, il est difficile de naviguer entre plusieurs configurations équivalentes.

Order-MCMC

Plutôt que chercher directement le graphe orienté, qui peut sembler être une tâche bien trop ardue, on peut tenter de déconstruire le problème analytiquement. C'est ce que

Algorithme 7 : Metropolis-Hastings

Entrées : Données $\mathscr{D}$, Loi de proposition q,
Initialisation $x^{(0)}$, Iteration maximale $i_{\max}$
Sorties : Echantillons issus de π
pour $i \in \{1, ..., i_{\max}\}$ **faire**
  $x^{(i)} \sim q(x|x^{(i-1)})$
  $\mathscr{T}_{x^{(i-1)},x^{(i)}} = \min\left(\frac{\pi(x^{(i)})q(x^{(i)}|x^{(i-1)})}{\pi(x^{(i-1)})q(x^{(i-1)}|x^{(i)})}, 1\right)$
  Avec probabilité $1 - \mathscr{T}_{x^{(i-1)},x^{(i)}}$, refuser le mouvement et fixer $x^{(i)} = x^{(i-1)}$
retourner $\boldsymbol{x} = \left(x^{(0)}, ..., x^{(i_{\max})}\right)$

propose [31] dans le cas purement observationnel, puis [77] en présence d'interventions. L'idée est la suivante : chaque DAG peut être décomposé en un squelette d'une part, et d'un ordre topologique d'autre part. Résoudre chacun de ces deux sous-problèmes, plus simples, permet de répondre à la question portant sur le DAG. On se concentre ici sur l'ordre topologique, et plus particulièrement sur une distribution sur les ordres topologiques. En adoptant une formulation bayésienne, on cherche à trouver la loi a posteriori des ordres $\mathbb{P}(\boldsymbol{o}|\mathscr{D})$. Celle-ci peut être formulée de manière naïve en utilisant un maximum a posteriori, où θ représente les paramètres en jeu dans le modèle, et $\hat{\theta}$ le maximum de vraisemblance.

$$\begin{aligned}\mathbb{P}(\boldsymbol{o}|\mathscr{D}) &= \int \mathbb{P}(\mathscr{D}|\boldsymbol{o},\theta)\,\mathbb{P}(\theta|\boldsymbol{o})\,\mathbb{P}(\boldsymbol{o})\,d\theta \\ &\approx \mathbb{P}(\mathscr{D}|\boldsymbol{o},\theta)\,\mathbb{P}\left(\hat{\theta}|\boldsymbol{o}\right)\mathbb{P}(\boldsymbol{o})\,.\end{aligned}$$

On utilise pour la déterminer de nouveau un algorithme de Métropolis-Hastings.

Algorithme 8 : Order-MCMC

Entrées : Données $\mathscr{D}$, Ordre initial $\boldsymbol{o}^{(0)}$
Sorties : Echantillons issus de $\mathbb{P}(\boldsymbol{o}|\mathscr{D})$
pour $i \in \{1,\dots,i_{\max}\}$ **faire**
  $\boldsymbol{o}^{(i)} \sim q(\boldsymbol{o}|\boldsymbol{o}^{(i-1)})$
  $\mathscr{T}_{\boldsymbol{o}^{(i-1)},\boldsymbol{o}^{(i)}} = \min\left(\frac{\pi(\boldsymbol{o}^{(i)})q(\boldsymbol{o}^{(i)}|\boldsymbol{o}^{(i-1)})}{\pi(\boldsymbol{o}^{(i-1)})q(\boldsymbol{o}^{(i-1)}|\boldsymbol{o}^{(i)})},1\right)$
  Avec probabilité $1-\mathscr{T}_{\boldsymbol{o}^{(i-1)},\boldsymbol{o}^{(i)}}$, refuser le mouvement et fixer $\boldsymbol{o}^{(i)} = \boldsymbol{o}^{(i-1)}$
retourner $\left(\boldsymbol{o}^{(0)},\dots,\boldsymbol{o}^{(i_{\max})}\right)$

IDA

Cette dernière méthode ne concerne pas exactement l'inférence de réseau, elle vise plutôt à palier l'absence de données interventionnelles. Tout l'intérêt de ces dernières est de permettre une meilleure spécification du modèle, en discriminant la cause de l'effet. Lorsque seules des données observationnelles sont à disposition, il est dans la majorité des cas impossible de distinguer les deux. Cependant on peut tout de même calculer toutes les gammes possibles de l'intensité du potentiel lien causal entre deux variables [62]. Cela suppose en premier lieu que nos données respectent les diverses hypothèses du modèle : fidélité, acyclicité, linéarité et gaussianité. Dans un tel cas, en utilisant un algorithme d'inférence au choix, nous obtenons un CPDAG, une classe d'équivalence pour les DAGs. Pour les graphes assez parcimonieux, il est possible de lister l'ensemble des DAGs appartenant au CPDAG estimé, et d'en calculer les effets causaux. Cependant ce n'est pas possible pour des graphes plus denses. On peut prouver néanmoins qu'il suffit de calculer l'ensemble des valeurs possibles pour les effets causaux sans énumérer les DAGs : parmi ces multitudes de DAGs, seules une poignée de valeurs sont possibles pour les effets causaux. En utilisant la propriété de Markov et la caractérisation d'un CPDAG, il suffit de regarder localement autour des noeuds d'intérêt pour les déterminer.

Dans la suite de la thèse, nous travaillerons sauf mention contraire dans le cadre de données gaussiennes multivariées et sans variables cachées. En fonction du problème posé, ces hypothèses peuvent ne pas être réalistes. Dans le cas des données de génomique, l'hypothèse gaussienne est couramment utilisée et semble pertinente. Cependant, comme dans de nombreux problèmes concrets, et en particulier en biologie, il est impossible d'écarter les variables cachées et/ou non mesurées. Dans notre cas, cela peut être

Algorithme 9 : IDA

Entrées : CPDAG $\mathcal{G}$, Variable expliquée j, variable explicative i
Sorties : Effets causaux possibles Θ
pour chaque ensemble $S \subset adj_j$ **faire**
  Construire le graphe $\mathcal{G}_S$ en orientant dans $\mathcal{G}$ les arêtes adjacentes à j de la façon suivante :
  — si $X_k \in S$ alors $X_k \rightarrow X_i$
  — sinon $X_i \rightarrow X_k$
  si $\mathcal{G}_S$ a exactement les mêmes v-structures que $\mathcal{G}$ **alors**
    Calculer l'effet causal de i sur j dans $\mathcal{G}_S$ et l'ajouter à Θ
retourner Θ

l'état du milieu intracellulaire, des métabolites ou des protéines sur l'activité d'un gène donné.

1.4 Organisation du manuscrit

Cette thèse se découpe en 5 chapitres. Dans le chapitre 2, on proposera une discussion autour de l'hypothèse d'acyclicité. Dans le chapitre 3, on propose une approche marginale qui utilise les données interventionnelles pour estimer les effets causaux. Dans le chapitre 4, je reprends une approche d'inférence de réseaux par MCMC que j'améliore via une pénalisation ridge pour permettre un passage à l'échelle. La chapitre 5 propose d'utiliser le *parallel tempering* pour inférer les effets causaux. Enfin dans le chapitre 6, on détaillera une méthode d'inférence de réseaux basée sur l'approximation de Laplace.

Chapitre 2

De l’hypothèse d’acyclicité

2.1 Introduction

L'implication des méthodes statistiques dans de nombreux domaines, de l'écologie à la biologie, implique de nombreux échanges avec les experts concernés. Une façon courante de résumer ces informations est l'utilisation de graphes. Cela permet d'interpréter facilement les résultats. De nombreux travaux ont été publiés, mais principalement sur les graphes non-orientés ou orientés acycliques [56]. Lorsque l'on essaye d'inférer des relations causales, il est pratique d'utiliser les graphes orientés. Dans ce cas, les arêtes représentent les liens de cause à effet. Lorsqu'on essaye de traiter du caractère causal du modèle, on peut de plus associer des équations structurantes (ou SEM pour *Structural Equation Modelling*) pour signifier les liens entre les différentes variables. Ces équations peuvent être dites récursives, lorsqu'elles sont associées à un DAG, ou non-récursives lorsqu'elles le sont avec un graphe orienté cyclique. La question de la différence de natures et de l'interprétation de ces deux modèles, cyclique ou acyclique, est ancienne [100]. Les problèmes d'identification et les résultats théoriques déjà acquis pour les DAGs sont des arguments opérant en leur faveur, et ce même si l'hypothèse d'acyclicité n'est en générale pas réaliste. Par exemple en génétique, un réseau de régulation de gènes peut permettre des boucles [51]. Dans d'autres domaines, comme en écologie [29] ou en psychologie [72], des boucles sont présentes et il est courant d'utiliser des équations non-recursives. L'hypothèse d'acyclicité est malgré tout majoritairement utilisée, et le problème de la modélisation des données réelles est adressé de deux manières. Si nous disposons de données temporelles, alors il n'y a plus aucun problème à l'utiliser, puisque le temps impose un ordre objectif sur les données. Si cela n'est pas le cas, on peut toujours émettre l'hypothèse d'être arrivé à un point d'équilibre stationnaire, pour lequel le graphe sous-jacent est acyclique. Néanmoins cet argument n'est pas pertinent s'il existe un doute non négligeable que l'on se trouve dans un régime cyclique. Ici, nous allons discuter de l'état de l'art sur les cas cycliques, et étudier la possibilité de relâcher l'hypothèse d'acyclicité.

De nombreuses études ont été faites sur les graphes orientés cycliques. Nous savons que les graphes cycliques préservent certaines propriétés, comme le critère de d-séparation [96]. Nous savons aussi qu'il y a, comme dans le modèle acyclique, un problème d'équivalence. Cependant les classes d'équivalences induites sont beaucoup plus grandes [78].

Se pose également la question de l'inférence des paramètres. Une proposition pour effectuer une telle inférence est d'utiliser l'équivalence entre les modèles, et de se ramener lorsque cela est possible à un modèle acyclique [18]. Cependant le graphe inféré doit être Markov par rapport à la loi sous-jacente, ce qui est alors rarement le cas. Dans le cas où nous avons des bruits non-gaussiens, une procédure appelée Ling D et basée sur l'analyse en composantes indépendantes à été proposée [55].

La classe d'équivalence étant un espace trop vague et difficilement interprétable, certaines méthodes ont tenté de spécifier un modèle précis plutôt qu'une collection. Cela est possible en présence de données interventionnelles. Ces interventions dans le cas cyclique ont été modélisées pour des variables discrètes [90]. Dans les cas où les interactions sont non-linéaires et/ou les bruits sont non gaussiens, la présence de données interventionnelles pour chaque variable a amené à l'inférence précise d'un graphe cyclique, avec de plus potentiellement des variables latentes [48]. Si seules des interventions aléatoires sont disponibles, l'estimation de certains liens de cause à effet est toujours possible [70]. Si la structure est connue, seule l'estimation des paramètres peut être faite de manière précise grâce aux données d'interventions [70].

Dans ce chapitre, nous allons rappeler la définition et les propriétés des graphes orien-

tés cycliques. Nous allons ensuite montrer que sous les hypothèses de linéarité et de gaussianité, nous pouvons écrire analytiquement la loi jointe. Nous montrerons dans quelles conditions le modèle est dégénéré, et nous parlerons aussi de l'interprétation de ce modèle. En particulier nous parlerons des paramètres causaux. Nous simulerons enfin quelques exemples décrivant les difficultés relatives aux modèles orientés cycliques.

2.2 Méthode

2.2.1 Graphe orienté cyclique

Je vais ici introduire le concept de graphe orienté cyclique, dans l'optique de modéliser des liens causaux. Il faut comprendre que la méthode est utilisée sur des données en régime permanent, où les cycles représentent un équilibre d'un processus temporel. On appelle un graphe $\mathcal{G} = (E, V)$ un ensemble de noeuds E et d'arêtes orientées V. Une loi de probabilité $\mathbb{P}$ est adjointe à ce graphe, donnant lieu à un modèle graphique probabiliste $D = (\mathcal{G}, \mathbb{P})$. Le lien entre la loi de probabilité et la représentation graphique tient aux indépendances conditionnelles. A partir de celles-ci, on peut montrer la propriété de factorisation suivante, très utile en pratique :

Definition 1 *Soit* G *un graphe orienté. Une loi de probabilité* $\mathbb{P}$ *d'une variable aléatoire* $\mathbf{X} = (X_1, \ldots, X_N)$ *est dite factorisable dans* G *si et seulement si on peut écrire :*

$$\mathbb{P}(X_1, \ldots, X_n) = \frac{1}{Z} \prod_{i=1,n} \mathbb{P}(X_i | pa(X_i))$$

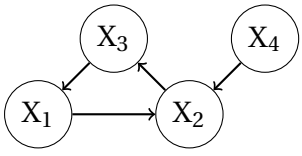
où $pa(X_i)$ *est l'ensemble des parents de* X_i*, et* Z *une constante de normalisation.*

En termes graphiques, une arête du noeud X_i au noeud X_j est tracée si et seulement si X_i est un parent de X_j. On peut remarquer que dans le cas où l'hypothèse d'acyclicité est retenue, la constante de normalisation est égale à 1. A l'opposé, pour le cas cyclique, la valeur de cette constante n'est pas connue, ce qui conduit éventuellement à des cas dégénérés ($Z = \infty$).

Nous allons étudier ce cas particulier de modèle graphique en utilisant des équations structurelles non-récursives. Cela nous permet de donner un sens et une interprétation causale en dehors de données d'interventions. Cela signifie juste que l'on autorise le graphe correspondant à avoir des cycles orientés, même entre deux variables, mais sans laisser la possibilité d'une boucle de noeuds sur lui-même. De plus, nous allons supposer que les fonctions au sein des équations sont linéaires, et que les bruits sont gaussiens. Un exemple d'un tel système est présenté ci-dessous.

Example 1 *Soit* $\boldsymbol{\epsilon} = (\epsilon_1, \epsilon_2, \epsilon_3, \epsilon_4)$ *un bruit gaussien décorrélé. Soient* X_1, X_2, X_3 *et* X_4 *quatre variables aléatoires, suivant le système d'équations structurelles suivant.*

$$\begin{aligned} X_1 &= 2 \times X_3 + \epsilon_1 \\ X_2 &= -X_1 + X_4 + \epsilon_2 \\ X_3 &= 0.3 \times X_2 + \epsilon_3 \\ X_4 &= \epsilon_4 \end{aligned}$$



Le système peut être associé au graphe orienté cyclique à droite.

Le lien avec la représentation graphique est fait grâce à l'hypothèse de fidélité. On peut l'écrire de la façon suivante : toutes les indépendances conditionnelles sont lues directement du modèle structurel, et non à cause d'éventuelles compensations entre les effets causaux. Dans le cas de relations linéaires, nous avons l'équivalence entre ces indépendances conditionnelles et le critère de d-séparation [97].

Permettre la présence de cycles rend ce modèle beaucoup plus réaliste que ceux étant acycliques. En particulier, de nombreuses boucles de rétroaction sont présentes en biologie et en génomique, rendant les systèmes plus stables.

Pour un ensemble fixé d'indépendances conditionnelles ou de d-séparations, il y a souvent plus d'un système d'équations structurelles ou d'un graphe compatible. Cela crée une relation d'équivalence entre modèles, qui pose problème dans le cadre d'une inférence causale. Ce problème peut être facilement résolu via des données interventionnelles. Il existe une façon agréable de résumer cette classe d'équivalence pour les DAGs. Celle-ci est beaucoup plus complexe pour les graphes orientés cycliques, et une caractérisation de celle-ci est indiquée en Annexe de ce chapitre.

Il est possible de tester si deux graphes cycliques sont équivalents, en se basant sur un algorithme de complexité $o\left(n^9\right)$ [79]. En raison de sa complexité, il n'existe pas dans la littérature une représentation graphique claire de la classe d'équivalence. Il est cependant possible de représenter un graphe englobant le squelette de l'ensemble des modèles équivalents : dans ce cas, il suffit de moraliser un représentant de la classe d'équivalence, d'ajouter une arête entre chaque nœud d'un *unshielded conductor* (comme défini en annexe) et de retirer toute orientation de chaque arête.

2.2.2 Innovation

Dans cette section, nous allons montrer comment calculer la constante de normalisation d'un modèle d'équations structurelles non-récursif linéaire gaussien, ainsi que la valeur des paramètres causaux dans de tels modèles.

Loi jointe

Nous allons décrire la loi jointe associée à un graphe gaussien cyclique. Nous caractérisons la loi à partir du modèle suivant :

$$\mathbf{X} = \boldsymbol{\mu} + \mathbf{X}\mathbf{W} + \boldsymbol{\varepsilon} \quad (2.1)$$

où $\mathbf{X}$ est un vecteur ligne, $\boldsymbol{\mu}$ est un vecteur de moyenne, $\mathbf{W}$ est une matrice d'adjacence et $\boldsymbol{\varepsilon}$ est un vecteur de bruits gaussiens indépendants, de variance $\boldsymbol{\sigma^2}$ et de moyenne nulle. De plus, nous pouvons écrire la densité de la loi : $\mathbb{P}(\mathrm{X}_i|\mathrm{pa}(\mathrm{X}_i))$:

$$f(\mathrm{X}_i = x|\,\mathrm{pa}(\mathrm{X}_i)) = \frac{1}{\sqrt{2\pi}\sigma_i}\exp\left(-\frac{(x-\mu_i-\mathbf{X}\mathbf{W}\boldsymbol{e}_i)^2}{2\sigma_i^2}\right),$$

et la loi jointe :

$$f(\mathrm{X}_1,\ldots,\mathrm{X}_n) = \frac{1}{e^{-\mathrm{Z}}}\prod_{i=1}^{n} f(\mathrm{X}_i = x|\,\mathrm{pa}(\mathrm{X}_i)).$$

Soit $\mathbf{P}$ la matrice de précision associée à notre loi jointe, alors la constante de normalisation s'écrit :

$$\mathrm{Z} = \frac{1}{2}\log(|\mathbf{P}|) + \frac{1}{2}\sum_{i=1}^{n}\log(\sigma_i^2).$$

Proposition 1 *Soit* X *un vecteur aléatoire suivant* (2.1)*, et si* $(\mathbf{I}-\mathbf{W})$ *est inversible, alors :*

$$\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}(\mathbf{I}-\mathbf{W})^{-1}, ((\mathbf{I}-\mathbf{W})^{-1})^{\mathrm{T}}\mathrm{diag}(\boldsymbol{\sigma}^{\mathbf{2}})(\mathbf{I}-\mathbf{W})^{-1}).$$

Cette proposition est très utile car elle nous permet de simuler n'importe quelle loi associée à un graphe orienté cyclique non dégénéré, et ce de manière très rapide. En effet, considérons l'exemple naïf suivant. Nous voulons simuler un échantillon d'un graphe cyclique par MCMC. A partir d'un point d'initialisation arbitraire, on utilise un algorithme de Métropolis-Hastings avec une loi de proposition qui consiste en une perturbation gaussienne. La trajectoire est montré 2.1.

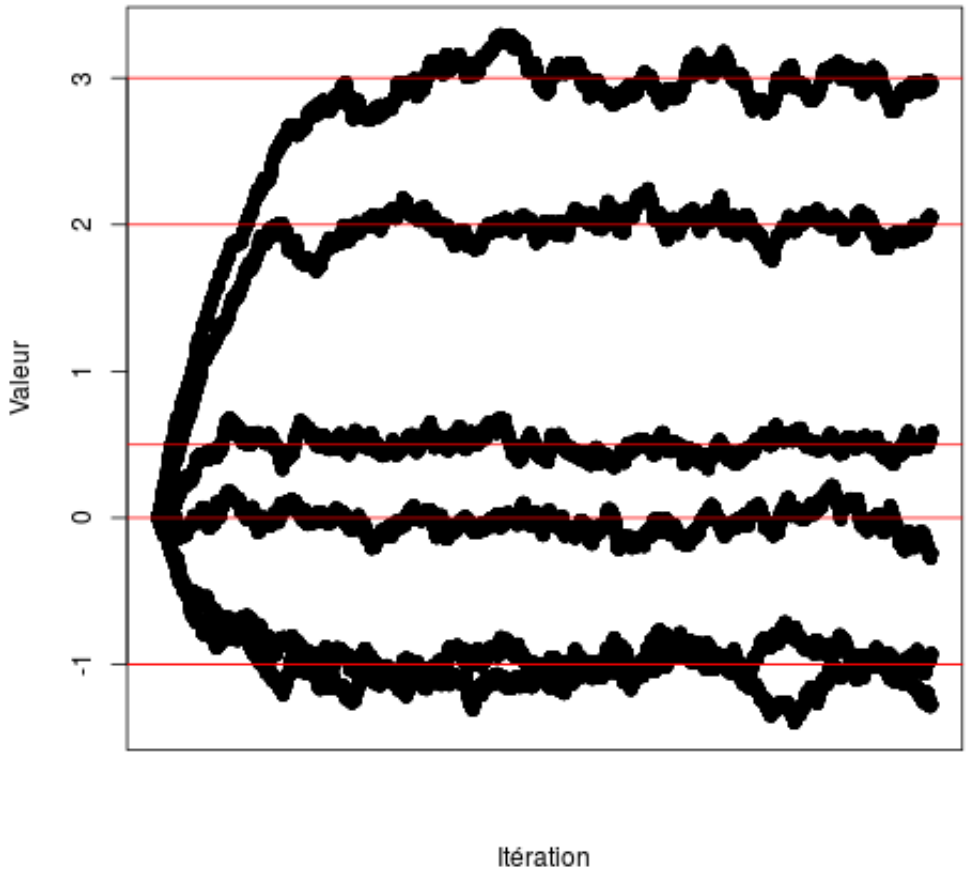


FIGURE 2.1 – Evolution de la trajectoire des différentes variables pour un graphe de 6 noeuds. Les lignes rouges indiquent la valeur moyenne attendue pour chaque variable.

On peut comparer alors la vitesse d'exécution entre cette procédure, et celle dérivée du calcul analytique précédent. En moyenne, pour cette simulation, la procédure MCMC a pris 626232 microsecondes, tandis que notre proposition n'en prend que 423. On va donc plus de 1000 fois plus vite, tout en simulant des échantillons de meilleur qualité.

Paramètres causaux

Être capable de simuler un modèle est une bonne chose, mais on souhaite également pouvoir l'inférer. Dans de nombreux cas, nous voulons trouver un sens causal à nos

graphes. Ici nous allons réutiliser quelques opérateurs standards de l'analyse causale pour décrire les paramètres dans la SEM gaussienne. Cet opérateur est la traduction mathématique des actions dans notre monde réel. Cela est nécessaire pour spécifier le modèle, en particulier du fait de la relation d'équivalence qu'il existe entre certains modèles. Dans ces cas là, causes et conséquences peuvent être inversées, sauf en présence de données interventionnelles. Nous introduisons l'opérateur do : la variable aléatoire $\mathbf{X}|\,\mathtt{do}\,(\mathrm{X}_i = x_i)$ définie par l'équation (1) est modifiée de telle sorte que l'équation i est supprimée, et la variable aléatoire X_i est remplacée dans toutes les autres équations par la valeur x_i. D'un point de vue graphique, c'est comme si l'on fixait X_i tout en supprimant tous les liens avec ses parents.

Nous proposons pour formaliser la notion de causalité d'utiliser la définition d'un effet causal défini par [74] :

Définition 4 *L'effet causal direct de* X_i *sur* X_j *est défini par :*

$$\frac{\partial}{\partial x_i}\mathbb{E}(\mathrm{X}_j|\,\mathit{do}(\mathrm{X}_i = x_i, \mathrm{X}_{-j} = x_{-j})).$$

L'effet causal total de X_i *sur* X_j *est défini par :*

$$\frac{\partial}{\partial x}\mathbb{E}(\mathrm{X}_j|\,\mathit{do}(\mathrm{X}_i = x_i)).$$

Nous pouvons écrire la loi de $\mathrm{X}_j|\mathbf{X}_{-j} = \boldsymbol{y}$:

$$\begin{aligned}\mathbb{E}(\mathrm{X}_j|\mathrm{X}_{-j}) &= \mu_j + \Sigma_{j,-j}\Sigma_{-j,-j}^{-1}(\boldsymbol{y} - \boldsymbol{\mu}_{-j})\\ \mathrm{Var}(\mathrm{X}_j|\mathrm{X}_{-j}) &= \Sigma_{j,j} - \Sigma_{j,-j}\Sigma_{-j,-j}^{-1}\Sigma_{-j,j},\end{aligned}$$

où $\Sigma_{j,-j}$ est la j-ieme ligne de la matrice de covariance, sans la colonne j. Vouloir calculer la loi de $\left(\mathrm{X}_j|\,\mathtt{do}(\mathrm{X}_i = x), \mathbf{X}_{-j} = \boldsymbol{y}\right)$ revient à calculer $\left(\tilde{\mathrm{X}}_j|\tilde{\mathbf{X}}_{-j} = \boldsymbol{y}\right)$, où

$$\tilde{\mathbf{X}} = \tilde{\boldsymbol{\mu}} + \tilde{\mathbf{X}}\tilde{\mathbf{W}} + \tilde{\boldsymbol{\varepsilon}},$$

et où les différentes quantités sont définies par : $\tilde{\mathbf{X}}$ est de dimension $\mathrm{N}-1$, $\tilde{\mathbf{W}}$ est la matrice $\mathbf{W}$ sans la ligne et la colonne i, $\tilde{\boldsymbol{\mu}}$ est $\boldsymbol{\mu} + x\boldsymbol{e}_i^{\mathrm{T}}\mathbf{W}$, et $\tilde{\boldsymbol{\varepsilon}}$ est un vecteur gaussien de moyenne nulle et de longueur j.

Avec cette formulation, nous avons :

$$\begin{aligned}\mathbb{E}(\mathrm{X}_j|\,\mathtt{do}(\mathrm{X}_i = x), \mathrm{X}_{-j}) &= \mu_j + x\boldsymbol{e}_i^{\mathrm{T}}\mathbf{W}\boldsymbol{e}_j + \tilde{\Sigma}_{j,-j}\tilde{\Sigma}_{-j,-j}^{-1}(\boldsymbol{y} - \tilde{\boldsymbol{\mu}}_{-j})\\ \mathrm{Var}(\mathrm{X}_j|\,\mathtt{do}(\mathrm{X}_i = x), \mathrm{X}_{-j}) &= \tilde{\Sigma}_{j,j} - \tilde{\Sigma}_{j,-j}\tilde{\Sigma}_{-j,-j}^{-1}\tilde{\Sigma}_{-j,j}.\end{aligned}$$

Nous voyons que la moyenne est linéaire en x, en conséquence l'effet causal direct l'est aussi :

$$\frac{\partial}{\partial x}\mathbb{E}(\mathrm{X}_j|\,\mathtt{do}(\mathrm{X}_i = x), \mathbf{X}_{-j}) = \mathrm{W}_{i,j}.$$

De la même façon, si on commence avec l'expression d'un vecteur gaussien $\tilde{\mathbf{X}}$:

$$\tilde{\mathbf{X}} \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}(\mathbf{I} - \tilde{\mathbf{W}})^{-1}, ((\mathbf{I} - \tilde{\mathbf{W}})^{-1})^{\mathrm{T}}\mathrm{diag}(\tilde{\boldsymbol{\sigma}}^{\mathbf{2}})(\mathbf{I} - \tilde{\mathbf{W}})^{-1}).$$

Nous avons en particulier la moyenne de la variable X_j, qui est $\left[\mu_j + x\boldsymbol{e}_i^{\mathrm{T}}\mathbf{W}\boldsymbol{e}_j\right](\mathbf{I} - \tilde{\mathbf{W}})^{-1}$. C'est encore linéaire en x. L'effet total est :

$$\frac{\partial}{\partial x}\mathbb{E}(\mathrm{X}_j|\,\mathtt{do}(\mathrm{X}_i = x)) = \mathrm{W}_{i,j}(\mathbf{I} - \tilde{\mathbf{W}})^{-1}.$$

Ces quantités, tout en nous permettant de décrire les phénomènes réels avec une plus grande acuité, permettent aussi de choisir entre plusieurs modèles équivalents. Ces modèles peuvent être inférés, pour les plus petits d'entre eux, en utilisant des tests classiques sur les indépendances conditionnelles.

2.3 Applications et exemples

Nous proposons ici une étude numérique qui décrit les problèmes de l'interprétation d'un graphe cyclique avec seulement des données d'observation.

Example 2 *Soit le modèle gaussien causal suivant défini avec ces paramètres :*

$$\boldsymbol{\mu_A} = (1 \quad 1 \quad 1 \quad -1)$$

$$\mathbf{W_A} = \begin{pmatrix} 0 & 0 & 0 & 0.3 \\ 2 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

$$\boldsymbol{\sigma_A^2} = \begin{pmatrix} 10^{-10} & 10^{-3} & 10^{-6} & 10^{-8} \end{pmatrix}$$

Nous pouvons jouer avec les paramètres, et trouver un équivalent :

$$\boldsymbol{\mu_B} = (1.875 \quad -0.5 \quad 0.001008981 \quad -1)$$

$$\mathbf{W_B} = \begin{pmatrix} 0 & 0.5 & 0.299697306 & 2.970297 \times 10^{-1} \\ 0 & 0 & -0.001008981 & -1.009059 \times 10^{-9} \\ 0 & 0 & 0 & 9.900990 \times 10^{-3} \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\boldsymbol{\sigma_B^2} = \begin{pmatrix} 1.564078 \times 10^{-3} & 2.500000 \times 10^{-11} & 1.008981 \times 10^{-6} & 9.900990 \times 10^{-9} \end{pmatrix}.$$

On peut montrer que les deux graphes G_A *et* G_B *sont associés à ces paramètres. (Figure* 2.2*).*

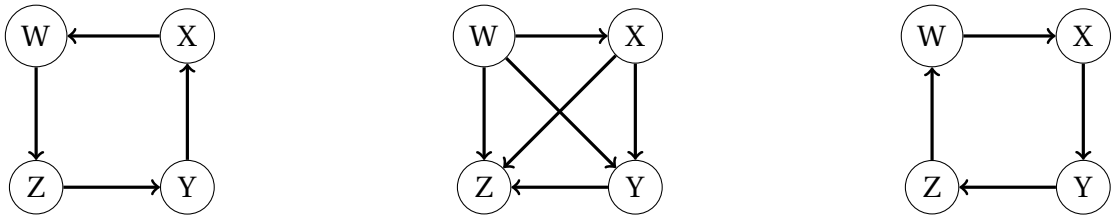


FIGURE 2.2 – A gauche, modèle A, au centre, modèle B, à droite modèle C

Cependant, dans la seconde paramétrisation, nous n'avons pas $X \perp\!\!\!\perp Z | (W,Y)$ *et* $W \perp\!\!\!\perp Y | (X,Z)$ *: l'hypothèse de fidélité est ici non remplie, et en conséquence le graphe perd sa signification, en particulier pour l'inférence causale. En fait, aucun graphe acyclique ne correspond aux relations d'indépendances conditionnelles du modèle* A*. Même si nous avons une équivalence paramétrique entre les deux modèles. Un modèle équivalent cyclique est le modèle C :*

$$\boldsymbol{\mu_C} = (10/3 \quad -0.5 \quad 1 \quad 1)$$

$$\mathbf{W_C} = \begin{pmatrix} 0 & 0.5 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \\ 10/3 & 0 & 0 & 0 \end{pmatrix}$$

$$\boldsymbol{\sigma_C^2} = \begin{pmatrix} 10/3 \times 10^{-8} & 1/4 \times 10^{-10} & 10^{-3} & 10^{-6} \end{pmatrix}.$$

Voici un autre exemple où il y a une équivalence entre deux graphes :

Example 3 *Soit les modèles gaussiens causaux suivants :*

$$\boldsymbol{\mu_A} = (1 \quad 1 \quad 1 \quad -1 \quad -1)$$

$$\mathbf{W_A} = \begin{pmatrix} 0 & 0 & 0 & 0.3 & 0 \\ 0 & 0 & 1.3 & 0 & 0 \\ 0 & 0 & 0 & -0.8 & 0 \\ 0 & 0 & 0 & 0 & 0.7 \\ 0 & -0.5 & 0 & 0 & 0 \end{pmatrix}$$

$$\boldsymbol{\sigma_A^2} = \begin{pmatrix} 10 & 10^{-2} & 10^{-1} & 10^{-2} & 10^{-1} \end{pmatrix}$$

$$\boldsymbol{\mu_B} = (1 \quad -0.7692308 \quad -1.25 \quad 1.428571 \quad 2)$$

$$\mathbf{W_B} = \begin{pmatrix} 0 & 0 & 0.375 & 0 & 0 \\ 0 & 0 & 0 & 0 & -2 \\ 0 & 0.7692308 & 0 & 0 & 0 \\ 0 & 0 & -1.25 & 0 & 0 \\ 0 & 0 & 0 & 1.428571 & 0 \end{pmatrix}$$

$$\boldsymbol{\sigma_B^2} = (10 \quad 0.0591716 \quad 0.0156250 \quad 0.2040816 \quad 0.04)$$

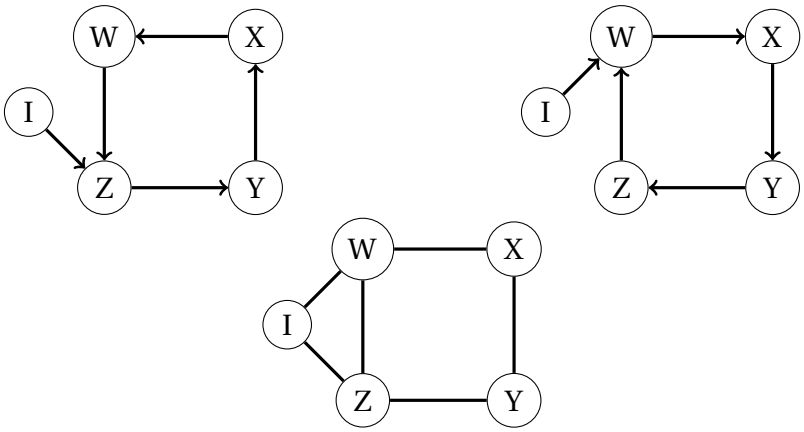


FIGURE 2.3 – A gauche, modèle A, à droite, modèle B, en dessous, graphe englobant les squelettes de l'ensemble des modèles équivalents

Nous voyons ici d'où provient la complexité pour les modèles orientés : le squelette, et en particulier la propriété de localité ne sont pas préservés parmi les modèles équivalents. En conséquence, lorsque nous relâchons l'hypothèse d'acyclicité, la classe d'équivalence associée :

— Ne préserve pas le squelette,

— Et ne préserve pas les immoralités.

Inférence

Nous avons vu qu'il était impossible de résumer simplement les classes d'équivalence dans le cadre de graphes orientés cycliques. Tout algorithme d'inférence développé ne peut alors simplement restituer son résultat, en tout cas sans une présence massive de données interventionnelles. Cependant, nous avons vu que si nous essayons d'inférer un DAG à partir d'un jeu de données provenant d'un modèle cyclique, nous pouvons casser l'hypothèse de fidélité et en conséquence arriver à de très mauvais résultats. Essayer donc d'inférer un graphe acyclique à partir de données manifestement cycliques est une idée arbitrairement mauvaise. La question du développement d'algorithme spécifique au cas cyclique est donc d'importance. A ce jour, il n'existe qu'un seul algorithme qui peut effectuer cette inférence pour des bruits gaussiens et sans interventions.

L'algorithme est appelé "*cyclic causal discovery*"[80]. Le but de cet algorithme est de trouver directement les informations derrière le modèle structurel ou le graphe : les indépendances conditionnelles. Das le cas gaussien, des tests sur les corrélations partielles sont utilisés. Ces tests souffrent pourtant de deux problèmes. En premier lieu, chaque paire de variables a besoin d'être testée conditionnellement pour chaque sous-ensemble possible. Cela conduit à un nombre exponentiel de tests à effectuer en fonction du nombre de nœuds. En second lieu, à cause de cette procédure de tests multiples, une correction doit être appliquée. Malheureusement, du fait du nombre de tests, la puissance statistique tend rapidement vers 0. Cette méthode ne fonctionne donc que pour des réseaux de petite dimension.

2.4 Discussion

Nous venons de voir que si nous relâchons l'hypothèse d'acyclicité, il est toujours possible de bien définir le modèle, dans le cas linéaire et gaussien. Nous pouvons aussi calculer la constante de normalisation, et ainsi simuler simplement le modèle cyclique. Nous pouvons de plus définir des quantités causales au sein de ce modèle, ce qui est particulièrement judicieux lorsqu'il s'agit de traiter de l'espace des modèles équivalents. Cependant, contrairement au modèle acyclique, nous n'avons pas une façon simple de visualiser ou de traiter ces modèles équivalents.

Ces modèles équivalents ne peuvent être exprimés aussi simplement que ceux du modèle acyclique. Dans ce dernier cas, trouver le squelette et les structures en v sont des conditions suffisantes pour connaître la classe d'équivalence, ce qui réduit drastiquement la complexité des algorithmes d'inférence. Les tests sur les indépendances conditionnelles peuvent alors n'être utilisés que pour trouver le squelette dans un premier temps, puis les structures en v dans un second temps. Dans le cas cyclique, la procédure est plus difficile, du fait d'une notion d'adjacence élargie, permettant à deux nœuds d'être adjacents dans un sens plus faible sans l'être concrètement dans la représentation graphique.

Lorsque l'hypothèse d'acyclicité n'est pas vraie, essayer d'inférer un graphe acyclique à partir d'un jeu de données issu d'un phénomène cyclique peut être très mauvais, et cassent la signification du graphe et de l'interprétation causale. Essayer d'inférer un graphe cyclique directement semble être une piste bien plus prometteuse, mais est hélas limité par la capacité de représentation de la classe d'équivalence. De plus, le seul algorithme capable d'effectuer une telle inférence est limité à des graphes très petits.

2.5 Annexe

L'annexe présente est rédigée en anglais. Ceci est dû au fait qu'il s'agit d'une discussion technique dont de nombreux termes sont difficilement traduisibles.

Here we discuss the characterisation of equivalent models for cyclic graphs. This is mainly based on the *unshielded triple*, also called a v-structure.

Definition 2 *An unshielded triple is a triple* (A, B, C) *such that* $A \to B \leftarrow C$ *but where* A *and* C *are not adjacent.*

Although we do not have a graphical criterion to summarize equivalent models for cyclic graphs (and therefore be able to list all equivalent models), we have conditions to decide if two cyclic graphs are equivalent [78; 79]. These conditions are based on *unshielded perfect non-conductors.*

Definition 3 *In a cyclic graph, two nodes* A *and* B *are said to be adjacent if and only if at least one of these two propositions holds :*

— *there exists an edge between* A *and* B,

— A *and* C *have a common child* B *such that* B *is an ancestor of* A *or* C.

Definition 4 *We say that* (A, B, C) *form an unshielded conductor if :*

1. A *and* B, *and* B *and* C *are adjacent, but* A *and* C *are not adjacent,*
2. B *is an ancestor of* A *or* C.

Definition 5 *We say that a triple of vertices* A, B *and* C *is an* unshielded perfect non-conductor *if :*

1. A *and* B, B *and* C *are adjacent, but* A *and* C *are not adjacent,*
2. B *is not an ancestor of* A *or* C,
3. B *is a descendant of a common child of* A *and* C.

If (1) *and* (2) *holds, but* (3) *do not holds, then* A,B *and* C *form an* unshielded imperfect non-conductor.

An itinerary is a sequence of vertices $X_0, \ldots, X_n$ such that $\forall i, \quad 0 \leq i < n$, X_i is adjacent of X_{i+1}. Moreover, if $\forall j \notin \{i-1, i, i+1\}$, X_i is not adjacent of X_j, then it is an uncovered itinerary.

Definition 6 *Let* $X_0, \ldots, X_{n+1}$ *be a sequence of vertices such that :*

1. $\forall t \quad 1 \leq t \leq n$, (X_{t-1}, X_t, X_{t+1}) *is a conductor,*
2. $\forall k \quad 1 \geq k \geq n$, X_{k-1} *and* X_{k+1} *are ancestors of* X_k,
3. X_0 *is not a descendant of* X_1, *and* X_n *is not an ancestor of* X_{n+1}.

then (X_0, X_1, X_2) *and* (X_{n-1}, X_n, X_{n+1}) *are mutually exclusive conductors on the itinerary* $X_0, \ldots, X_{n+1}$.

Here an example illustrating these various definitions.

Example 4 *In the directed cyclic graph below, we can see that :*

— X_1 *and* X_4 *are adjacent,*

— *The triple* (X_1, X_2, X_4) *is an* unshielded conductor,

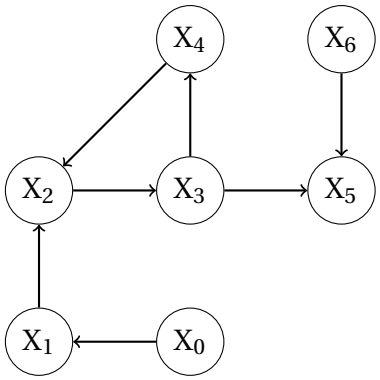


FIGURE 2.4 – Example of a directed cyclic graph

— (X_0, X_1, X_4) *form an* unshielded imperfect non-conductor,

— (X_3, X_5, X_6) *form an* unshielded perfect non-conductor

Cyclic graphs $\mathcal{G}$ and $\mathcal{G}'$ are equivalent if and only if the following conditions hold :

1. $\mathcal{G}$ and $\mathcal{G}'$ have the same adjacencies.
2. $\mathcal{G}$ and $\mathcal{G}'$ have the same unshielded conductors.
3. $\mathcal{G}$ and $\mathcal{G}'$ have the same unshielded perfect non-conductors.
4. (A, B, C) and (X, Y, Z) are mutually exclusive conductors on some uncovered itinerary in $\mathcal{G}$ if and only if (A, B, C) and (X, Y, Z) are mutually exclusive conductors on some uncovered itinerary in $\mathcal{G}'$.
5. If (A, X, C), and (A, Y, C) are unshielded imperfect non-conductors, then X is an ancestor of Y in $\mathcal{G}$ iff X is an ancestor of Y in $\mathcal{G}'$.
6. If (A, B, C) and (X, Y, Z) are mutually exclusive conductors on some uncovered itinerary and (A, M, Z) is an imperfect non-conductor, then M is a descendant of B in $\mathcal{G}$ if and only if M is a descendant of B in $\mathcal{G}'$.

Chapitre 3

Identification of marginal causal relationships in gene networks form observational and interventional expression data

Ce chapitre a fait l'objet d'une publication dans le journal *PLOS One*. Il concerne le traitement de données interventionnelles dans le but d'en déterminer des liens causaux. Il existe dans de nombreuses études en transcriptomique différents types de données, relevant de conditions expérimentales différentes. Ces différentes conditions peuvent être très complexes. Par exemple on peut considérer les différences entre deux types de lignées animales. Dans ce cas on peut s'attendre à ce que de nombreux éléments de la machinerie biologique diffèrent, et il est alors difficile de mesurer l'intégralité de ces différences. A contrario, on peut aussi se concentrer sur une lignée et essayer de comprendre l'influence d'une mutation soudaine.

C'est dans ce cadre que la présente méthode à été développée. La question est de savoir si l'on peut inférer des relations causales dans ce cas. La méthodologie actuelle repose en vérité sur l'analyse différentielle, qui est schématiquement une version spécifique du test de Student. Rappelons que ce test cherche à déterminer si les espérances de deux variables aléatoires sont égales. Dans notre cas, on cherche donc à savoir si un gène s'exprime en moyenne de manière différente en présence d'une mutation. Il n'y a pas à proprement parlé de langage causal derrière ce test, même si les conditions expérimentales font en sorte que le résultat de ce test est considéré comme un résultat causal : si la moyenne est significativement différente entre le groupe muté et non-muté, c'est qu'il y a une influence causale de la mutation sur le gène.

Nous avons choisi d'utiliser formellement un langage causal pour développer une méthode bayésienne, à l'opposé du test fréquentiste précédent. Nous avons deux types de données, dont seule une mutation change leur génération. Nous écrivons ici plusieurs modèles : l'un décrivant une influence causale du gène muté, l'autre l'absence d'une telle influence. Nous en décrivons leurs vraisemblances, puis par un jeu de réécriture, nous les utilisons pour en dériver un facteur de Bayes. C'est ce facteur qui détermine dans quel cas on se place. L'avantage d'une telle approche est sa flexibilité : il est aisé de modifier légèrement le modèle pour se poser des questions plus précises. On montre dans cet article un résultat sur des données simulées ainsi que sur des données réelles. Dans le cadre de ces expériences, on confirme aussi la pertinence du test de Student pour effectuer une inférence causale. Un package R, `MarginalCausality`, a été développé pour cette méthode et est disponible sur le site Github.

MONNERET, Gilles, JAFFRÉZIC, Florence, RAU, Andrea, et al. Identification of marginal causal relationships in gene networks from observational and interventional expression data. *PLOS One*, 2017, vol. 12, no 3, p. e0171142.

Identification of Marginal Causal Relationships in Gene Networks from Observational and Interventional Expression Data

Gilles Monneret[1,2,*], Florence Jaffrézic[1], Andrea Rau[1], Tatiana Zerjal[1], Grégory Nuel[2]

1 UMR GABI, AgroParisTech, INRA, Université Paris-Saclay, 78350 Jouy-en-Josas, France
2 LPMA, UMR CNRS 7599, UPMC, Sorbonne Universités, 4 place Jussieu, 75005 Paris, France

* gilles.monneret@upmc.fr

Abstract

Causal network inference is an important methodological challenge in biology as well as other areas of application. Although several causal network inference methods have been proposed in recent years, they are typically applicable for only a small number of genes, due to the large number of parameters to be estimated and the limited number of biological replicates available. In this work, we consider the specific case of transcriptomic studies made up of both observational and interventional data in which a single gene of biological interest is knocked out. We focus on a marginal causal estimation approach, based on the framework of Gaussian directed acyclic graphs, to infer causal relationships between the knocked-out gene and a large set of other genes. In a simulation study, we found that our proposed method accurately differentiates between downstream causal relationships and those that are upstream or simply associative. It also enables an estimation of the total causal effects between the gene of interest and the remaining genes. Our method performed very similarly to a classical differential analysis for experiments with a relatively large number of biological replicates, but has the advantage of providing a formal causal interpretation. Our proposed marginal causal approach is computationally efficient and may be applied to several thousands of genes simultaneously. In addition, it may help highlight subsets of genes of interest for a more thorough subsequent causal network inference. The method is implemented in an R package called `MarginalCausality` (available on GitHub).

3.1 Introduction

Causal network inference is of great interest in systems biology, particularly for transcriptomic studies that aim to identify regulatory relationships among genes, i.e., gene regulatory networks. In the context of probabilistic graphical models, several algorithms have been proposed to infer the skeleton of directed, undirected, or partially-directed graphs using conditional independence tests [49; 75], score-based procedures [9; 15; 17; 76] or mutual information [20; 28; 60; 103]. These skeletons correspond to an equivalence class, i.e. an indistinguishable subset of graphs. Undirected graphs can be used to obtain a supergraph of the skeleton of a directed graph, which is a good starting point to

infer causality when the underlying graph is unknown. Several undirected network inference methods, based on the parsimonious estimation of the inverse covariance matrix, have also been proposed for Gaussian graphical models [30; 33]. Although methods based on mutual information can also be used to infer the full graph of undirected networks [7; 21], estimating causal networks with these algorithms tends to be very computationally demanding and applicable only for low-dimensional networks. In addition, such approaches require a significant amount of interventional data to reduce the space of equivalent networks [102]. However, even with a sufficient amount of interventional data, i.e. roughly one knock-out for each gene, a directed acyclic graph (DAG) cannot generally be accurately estimated [66], perhaps due to the heterogeneous coverage of the gene network space [2]. As such, in this work we focus on estimating a few causal effects rather than attempting to infer the full network [62].

In order to reduce the complexity of the parameter search space, a topological ordering of nodes in the graph can be estimated instead of an exact network. As shown by Rau *et al.* [77], a rich set of interventional data allows the node ordering associated with a DAG to be identified. In many transcriptomic experiments, however, only a small number of interventions are available ; in this work, we consider the specific case of a knock-out intervention being performed on a single gene of interest. In such a case, only a restricted equivalence class is identifiable [99], and it is reasonable to instead consider a marginal approach to estimate only the causal effects of the knocked-out gene of interest on another set of genes.

To this end, we propose a method to identify downstream causal relationships between a knocked-out gene and all other genes from replicated observational (steady state) transcriptomic data arising from an unknown graph. We first present a brief introduction to graphical models, which we use to define our model and hypothesis. The use of a mathematical operator to describe the intervention process, as defined by Pearl [74], allows the idea of causality to be formally defined in the model. This enables a closed-form expression of the likelihood to be obtained. We then illustrate the interest of our method on a set of simulated data, and we apply it to a set of microarray data from chickens carrying a functional knock-out of the growth hormone receptor gene [25]. The main advantages of the proposed marginal causal approach are that 1) it enables the accurate differentiation of downstream causal relationships from those that are upstream or simply associative ; 2) it is computationally efficient, and thus simultaneously applicable to several thousands of genes ; and 3) it provides a formal framework for causal interpetation. The proposed method is implemented in an R package called `MarginalCausality`, freely available on GitHub.

3.2 Materials and methods

Gaussian causal models

A directed graph $\mathscr{G}$ is a set of nodes $\mathbf{V}$ and edges $\mathscr{E}$. For $(X,Y) \in \mathscr{E}$, X is said to be a parent of Y ($X \in \mathtt{pa}(Y)$), or Y a child of X, if an edge starts at X and points to Y. A directed path is a succession of nodes such that each element is a parent of the following node. A graph is said to be acyclic if there is no directed path from a node back to itself. It is then called a DAG.

A probability density P can be associated with a DAG. Assuming that all variables are Gaussian, such that the joint probability is a multivariate Gaussian distribution, the follo-

wing factorization holds for the joint density of the graph :

$$P(\mathbf{V}) = \prod_{V \in \mathbf{V}} f(V|\mathtt{pa}(V)).$$

Two graphs are said to be Markov equivalent if they have the same joint probability distribution ; in this case, they belong to the same equivalence class. The mathematical $\mathtt{do}(X = x)$ operator [74] can be graphically defined by deleting all edges pointing from $\mathtt{pa}(X)$ to X. For the associated probability distribution, this corresponds to replacing the conditional distribution $f(X|\mathtt{pa}(X))$ by $\mathbf{1}_{X=x}$.

Model selection

Consider a graph $\mathcal{G}$ where an intervention was performed on a single node of interest G. Two kinds of causality can be defined : 1) upstream causality, which refers to edges pointing to G ; and 2) downstream causality, which refers to edges pointing away from G. With the do operator defined above, when an intervention is performed on a single node G, it is possible to identify genes with a downstream causal relationship to G, i.e. the causal effects of G on all of the other nodes in the network (see Fig 3.1 for an illustration of upstream and downstream causal relationships).

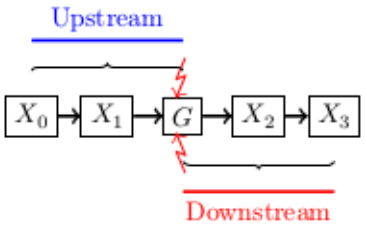


FIGURE 3.1 – **Illustration of upstream and downstream causality.** Nodes X_0 and X_1 are both upstream causally related to knocked-out gene G, while nodes X_2 and X_3 are both downstream causally related to G.

Our goal is to identify genes with downstream causal relationships to a knocked-out gene of interest when the underlying graph is unknown. It is well known that when observational data alone are available, only an equivalence class for the DAG is identifiable[74]. With the addition of interventions, it is possible to reduce this set of equivalence classes, but it is still often not possible to identify a unique DAG. For this reason, we consider a marginal approach to estimate only the causal downstream relationships from a single node of interest.

Using the framework of Gaussian structural equations, three possible cases may be defined for each node X of the graph. First, if X is a child of the node of interest G, the following equation holds :

$$M_1 : \quad X = \mu_X + \alpha G + \varepsilon_X,$$

where μ_X is the residual mean of X, α is the total causal effect from G to X, and ε_X is centered Gaussian noise with variance σ^2_X. On the other hand, if X is a parent of G, the following equation can similarly be written :

$$M_0 : \quad G = \mu_G + \alpha X + \varepsilon_G,$$

where μ_G is the residual mean of G, α is the total causal effect from X to G, and σ_G the residual standard deviation of G. Finally, if X is neither a child (descendant) nor a parent (ancestor) of G, the model can no longer be expressed in terms of a Gaussian structural equation. However, as the pair of variables X and G may still be correlated, the pair (X, G) can be considered to be a random variable following a bivariate Gaussian distribution.

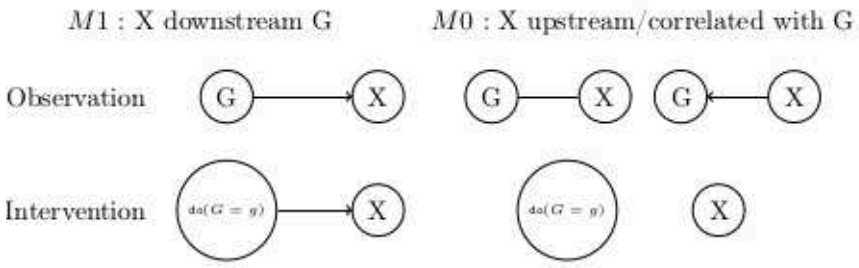


FIGURE 3.2 – **Models given observational or interventional data.** Graphical representation of the M_1 (downstream) and M_0 (upstream or correlated) models under observational and interventional data.

We next must compute the likelihood functions associated with each of these three cases when both observational and interventional data are available. However, as illustrated in Fig 3.2, even with the availability of interventional data, the causal downstream model and the correlated model cannot be distinguished from one another, as their likelihoods are identical. In Model M_1 (the downstream case), the distribution of X under the do operator is needed. In Model M_0 (the upstream or correlation cases), the marginal distribution of X must be used. Using the Markov equivalence for observational data, all models can be reparameterized as a downstream model. Our models may thus written as follows :

$$\begin{aligned}
M_1 : Z_1 \sim \mathcal{N}(\mu_1, \sigma_1^2), \quad Z_2 \sim \mathcal{N}(\mu_2, \sigma_2^2),\\
G = Z_1, \quad X = \alpha Z_1 + Z_2,\\
M_0 : \tilde{Z_1} \sim \mathcal{N}(\tilde{\mu_1}, \tilde{\sigma_1}^2), \quad \tilde{Z_2} \sim \mathcal{N}(\tilde{\mu_2}, \tilde{\sigma_2}^2),\\
G = \beta \tilde{Z_1} + \tilde{Z_2}, \quad X = \tilde{Z_1},
\end{aligned}$$

$$M_0 : \begin{pmatrix} G \\ X \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} m_1 \\ m_2 \end{pmatrix}, \begin{pmatrix} s_1^2 & \rho s_1 s_2 \\ \rho s_1 s_2 & s_2^2 \end{pmatrix} \right).$$

We can now explicitly write the following equalities :

$$\begin{aligned}
\mu_1 = m_1, \quad \mu_2 = m_2 - \alpha m_1, \quad \sigma_1 = s_1,\\
\alpha = \rho s_2 / s_1, \quad \sigma_2 = \sqrt{s_2^2 - \alpha^2 s_1^2},\\
\tilde{\mu_1} = m_2, \quad \tilde{\mu_2} = m_1 - \beta m_2, \quad \tilde{\sigma_1} = s_2,\\
\beta = \rho s_1 / s_2, \quad \tilde{\sigma_2} = \sqrt{s_1^2 - \beta^2 s_2^2}.
\end{aligned}$$

We have thus obtained all of the necessary distributions to compute the respective likelihoods for each model : the joint distribution, the conditional distribution of X given G, and the marginal distribution of X, all with the same set of parameters. For simplification, we define $\{\mu_X, \sigma_X\}$ and $\{\mu_G, \sigma_G\}$ to be the set of residual parameters associated with X and

G, respectively, and θ the full set of parameters. We obtain two likelihood functions, where WT (wild type) represents the observational data, and KO (knock-out) the interventional data :

$$\begin{aligned}
\ell_{\mathrm{WT}}(\theta) &= \sum_{k\in \mathrm{WT}} \log\Phi\left(\mathrm{X}_k|\mu_{\mathrm{X}} + \alpha \mathrm{G}_k, \sigma^2_{\mathrm{X}}\right) + \log\Phi\left(\mathrm{G}_k|\mu_{\mathrm{G}}, \sigma^2_{\mathrm{G}}\right), \\
\ell^1_{\mathrm{KO}}(\theta) &= \sum_{k\in \mathrm{KO}} \log\Phi\left(\mathrm{X}_k|\mu_{\mathrm{X}} + \alpha \mathrm{G}_k, \sigma^2_{\mathrm{X}}\right), \\
\ell^2_{\mathrm{KO}}(\theta) &= \sum_{k\in \mathrm{KO}} \log\Phi\left(\mathrm{X}_k|\mu_{\mathrm{X}} + \alpha \mu_{\mathrm{G}}, \alpha^2\sigma^2_{\mathrm{G}} + \sigma^2_{\mathrm{X}}\right), \\
\ell_{\mathrm{M}_1}(\theta) &= \ell_{\mathrm{WT}}(\theta) + \ell^1_{\mathrm{KO}}(\theta), \qquad (3.1) \\
\ell_{\mathrm{M}_0}(\theta) &= \ell_{\mathrm{WT}}(\theta) + \ell^2_{\mathrm{KO}}(\theta). \qquad (3.2)
\end{aligned}$$

Once the likelihoods in Equations (3.1) and (3.2) have been maximized, a Bayes factor can be calculated for each gene to choose the most probable model between M_0 and M_1 :

$$\mathrm{B} = \frac{\mathbb{P}(\mathrm{data}|\mathrm{M}_0)}{\mathbb{P}(\mathrm{data}|\mathrm{M}_1)} = \frac{\mathbb{P}(\mathrm{M}_0|\mathrm{data})}{\mathbb{P}(\mathrm{M}_1|\mathrm{data})} \times \frac{\mathbb{P}(\mathrm{M}_1)}{\mathbb{P}(\mathrm{M}_0)}.$$

The Bayes factor may then be used to order the nodes according to the strength of the downstream causal relationship with node G. If it is greater than 1, model M_0 is preferred, whereas if it is less than 1, model M_1 is preferred.

3.3 Results

Simulation study

In order to assess the performance of our proposed method, we performed a simulation study to ensure that it correctly distinguishes downstream causality from correlation. We considered a simulation setting similar to the experimental design of the transcriptomic data presented below. For 100 independent genes, we simulated 24 replicates in both the observational and the single-KO interventional data, using a Gaussian framework as presented in the Methods section with either a downstream causal or correlated relationship with the KO gene. For each of the 100 simulations, the Bayes factor was then calculated. Two sets of simulations were performed for each model with the same means and causal effects, but with different residual variances.

Results for the four simulation settings are presented in Fig 3.3. On the left, data were simulated with $\sigma_{\mathrm{G}} = 0.09, \sigma_{\mathrm{X}} = 0.15$, and on the right, $\sigma_{\mathrm{G}} = 0.3, \sigma_{\mathrm{X}} = 0.5$. This range of values was chosen based on the transcriptomic data presented in the following section, representing small variances (within the lower quantile) and large variances (within the upper quantile), respectively. We note that for small variances, the logarithm of the Bayes factor is strongly negative for the downstream causal model, while it is around zero for the correlation model. A similar pattern is obtained for larger variances, with smaller differences between the correlation and downstream causality cases.

In a second simulation, we investigated whether the proposed method is able to identify marginal downstream causal partners from a simulated graph. Data were simulated under a Gaussian structural equation according to the DAG structure presented in Fig 3.4. We simulated a knock-out intervention on Gene 6 alone ; as before, 24 replicates were simulated for both the observational and interventional data for each of the other genes in

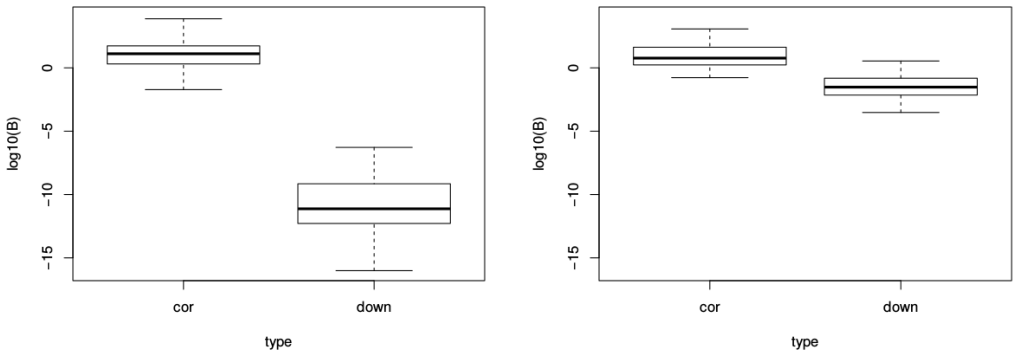


FIGURE 3.3 – **Bayes factor for the correlation (M_0) and downstream (M_1) models.** $\log_{10}$ Bayes factor for simulated data under the downstream model ("down") and correlation model ("cor"), with low (left ; $\sigma_G = 0.09, \sigma_X = 0.15$) and high (right ; $\sigma_G = 0.3, \sigma_X = 0.5$) variance.

the network in 100 independent runs. We note that the intervention node, represented in yellow in Fig 3.4, has several types of relationships with the other nodes : causal upstream ancestors (in blue), downstream causal genes (in red), and simply correlated genes (in green).

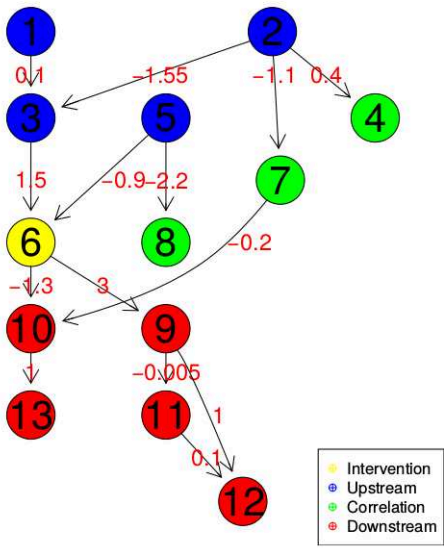


FIGURE 3.4 – **Simulated graph structure.** KO interventions were simulated for Gene 6 (yellow) alone. This DAG encompasses various types of relationships with respect to the yellow node : causal upstream ancestors (blue), downstream causal genes (red), and simple correlations (green). Numbers along edges indicate the strength of direct causal effects.

The values of the Bayes factor for each gene in the network are plotted in Fig 3.5. As before, the more strongly negative the logarithm of the Bayes factor, the more evidence there is for a downstream causal model. We note that for the genes with a truly downstream causal relationship with the KO gene (in red), the Bayes factor indeed tends to be strongly negative. Only node 11 is not detected as a descendent ; this can be explained by

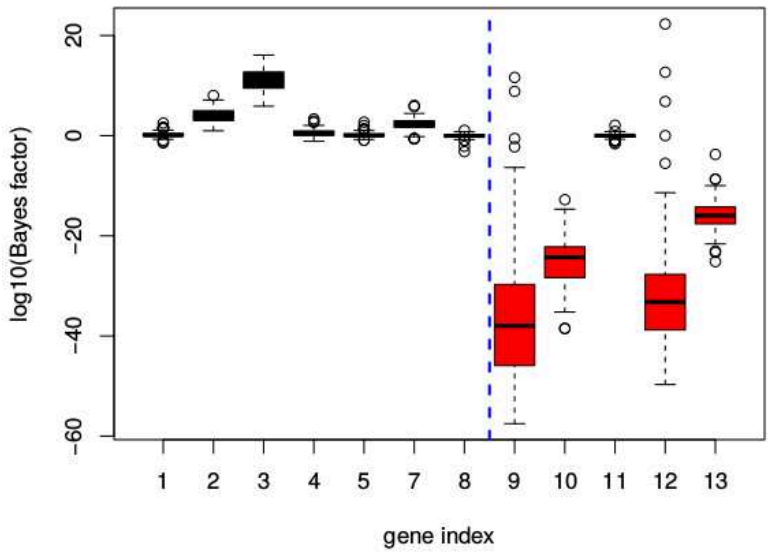


FIGURE 3.5 – **Bayes factor for the simulated graph structure.** Results from 100 simulations based on the graph in Fig 3.4. Nodes simulated under the upstream/correlation model (M_0) appear to the left in black, and those simulated under the downstream model (M_1) appear to the right in red.

the weak total causal effect for this node, equal to −0.015 from node 6 (the knocked-out gene) to node 11. These simulations thus confirm the ability of the proposed method to differentiate downstream causal relationships from upstream or simple correlation ones in a directed acyclic graph. An additional advantage of the marginal causal approach is that it provides an estimation of the total causal effect between the KO gene of interest and each of the others. The simulated values for these effects, as well as the estimations obtained for various numbers of replicates are provided in Table 3.1. The accuracy of the estimation is very robust even with a low number of replicates, with very low variability across simulations. These effects thus appear to be generally well-estimated with the proposed method.

In order to compare the results of our marginal causal approach with another marginal (but non-causal) approach routinely used in practice for comparisons between two groups (here, observational and interventional), we also performed a classical differential analysis on the same set of simulated data using the R/Bioconductor limma package [94]. Briefly, limma makes use of a robust moderated two-sample t-test between the observational and interventional samples for each gene, where an empirical Bayes method is used to shrink per-gene sample variances towards a common value. We calculated the area under the receiver operating characteristic (ROC) curve (AUC) over 100 simulations to compare the sensitivity and specificity of the marginal causal approach and the differential analysis to detect downstream causal relationships. Results are presented in Fig 3.6. We note that for a relatively large number of replicates (10 WT / 10 KO or 25 WT / 25 KO), both methods perform very similarly, with slightly better results for the proposed causal approach. In the simulation setting with a small number of replicates (5 WT / 5 WO), the performance of the differential analysis tends to deteriorate more strongly than that of the causal approach ; in particular, the AUC values much lower than in the other settings,

Gene	True value	6 WT / 6 KO	12 WT / 12 KO	24 WT / 24 KO	48 WT / 48 KO
1	0.00	0.01 (0.21)	0.04 (0.15)	0.04 (0.10)	0.02 (0.07)
2	0.00	-0.03 (0.06)	-0.04 (0.04)	-0.04 (0.02)	-0.04 (0.02)
3	0.00	0.08 (0.12)	0.10 (0.07)	0.09 (0.05)	0.09 (0.03)
4	0.00	0.00 (0.06)	-0.02 (0.04)	-0.01 (0.02)	-0.02 (0.02)
5	0.00	-0.01 (0.04)	0.00 (0.03)	-0.01 (0.02)	-0.01 (0.01)
7	0.00	0.03 (0.08)	0.05 (0.05)	0.04 (0.04)	0.04 (0.03)
8	0.00	0.04 (0.27)	-0.01 (0.18)	0.01 (0.16)	0.01 (0.10)
9	3.00	2.99 (0.17)	2.98 (0.08)	2.99 (0.23)	2.99 (0.10)
10	-1.30	-1.31 (0.08)	-1.30 (0.05)	-1.30 (0.04)	-1.31 (0.03)
11	-0.02	-0.02 (0.06)	-0.02 (0.04)	-0.01 (0.03)	-0.02 (0.02)
12	3.00	3.00 (0.11)	2.97 (0.24)	3.00 (0.14)	3.00 (0.04)
13	-1.30	-1.32 (0.16)	-1.29 (0.09)	-1.30 (0.10)	-1.31 (0.05)

TABLEAU 3.1 – **Estimated total causal effects for simulated graph structure.** Mean (standard deviation) of the true and estimated total causal effects over 100 simulations for various numbers of replicates.

with a large variability across the 100 simulations.

Finally, although our approach focuses on marginal causal effects and not the full network, it is of interest to compare it to a more global network-wide approach. As an illustration, we make use of the Greedy Interventional Equivalence Search (GIES) algorithm, a score-based method to infer the full directed acyclic graph based on observational and interventional data [43]. For this comparison, we focus only on the downstream causal relationships from the KO gene of interest. We use the graph structure in Fig 3.4 to simulate data as above (100 datasets, with 24 replicates in each of the WT and KO groups), and we define an F-score to assess the performance of each algorithm :

$$\mathrm{R} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}},$$

$$\mathrm{P} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}, \qquad ,$$

$$\text{F-score} = 2\frac{\mathrm{R}+\mathrm{P}}{\mathrm{RP}},$$

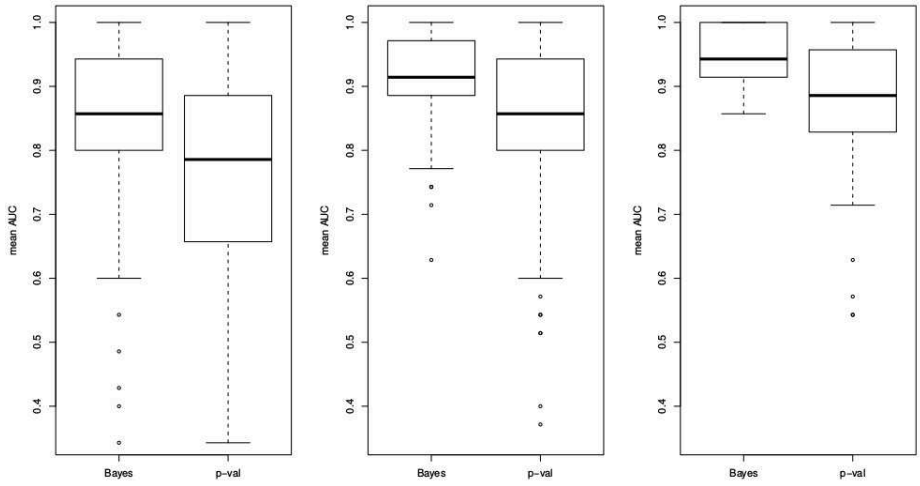
where “TP" corresponds to nodes that were simulated to be downstream of the KO gene and were correctly identified by a given method, “FP" corresponds to nodes that were not simulated to be downstream of the KO gene but were incorrectly identified by a given method, and “FN" corresponds to nodes that were simulated to be correlated/upstream of the KO gene but were incorrectly missed by a given method.

The boxplot of F-scores for each method is shown in Fig 3.7, as well as the same simulation for upstream/correlation links. As our marginal approach focuses on the downstream links and does not try to infer the topology of a full network, it obtains more consistent results than the GIES algorithm. This suggests that in cases where interest is on the downstream causal links with a single KO gene, attempting to infer a complete network topology may lead to more inaccurate results than focusing on marginal causal effects.

Real data analysis

We applied our marginal causal method to a set of transcriptomic data in the context of gene regulatory networks [27]. These data were produced at the French National Institute for Agricultural Research (INRA), in a study investigating gene expression differences

FIGURE 3.6 – **AUC for the simulated graph structure.** Results from 100 simulations under three settings (5 WT / 5 KO, 10 WT / 10 KO, 25 WT / 25 KO) based on the graph in Fig 3.4. “Bayes" (left) corresponds to the causal marginal method, and “p-val" (right) to the p-values obtained using limma.



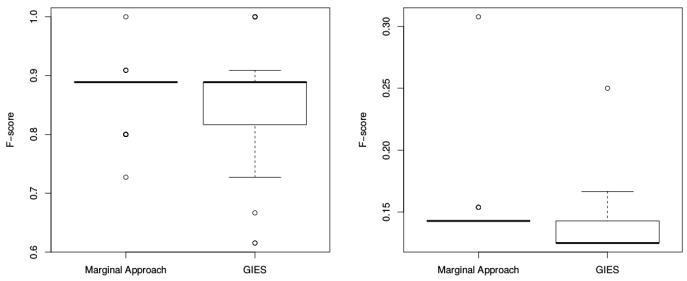


FIGURE 3.7 – **F-score of the marginal causal approach and GIES algorithm.** The F-score is based on downstream (left) or not-downstream (right) links. For the marginal approach, a hard threshold of -0.5 is used for the $\log_{10}$ Bayes factor to select between models. For the GIES algorithm, the inferred topology is used to classify nodes as downstream or not.

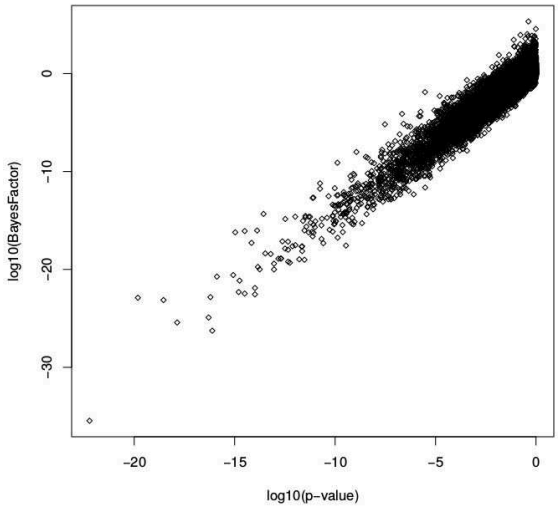


FIGURE 3.8 – **Comparison between the differential analysis and the marginal causal approach on chicken microarray data.** Each point corresponds to a gene for which the differential and marginal causal analyses have been applied.

between wild-type chickens and their siblings carrying a functionally inactive growth hormone receptor (GHR) gene, leading to a dwarf phenotype ; in this case, a functional KO refers to a mutation for which the associated protein is generated but can no longer fulfill its role [25]. We considered the mutation of the GHR gene to be an experimental knockout, and the expression level of the GHR gene was set to a value close to zero for the dwarf chickens. Customized Agilent microarrays were used to measure gene expression from liver samples taken from 24 wild type and 24 knock-out chickens for 18,855 genes. These data are available at GEO under the GEO accession number GSE91084.

After standard preprocessing and normalization steps, we aimed to identify genes that are downstream causally related to the GHR gene using the marginal causal method presented in this work. A classical per-gene differential analysis was also performed between wild-type and dwarf chickens using limma [94]. Fig 3.8 compares the Bayes factors obtained with the marginal causal method and the p-values of the differential analysis. In this case, the results are very similar for the two analyses, with a clear linear trend in the scatter plot. This follows the results obtained in the simulation study for more than 10 biological replicates ; the added value provided by the marginal causal method, however, is that it provides a formal interpretation of the differential analysis in terms of downstream causal relationships. A similar result may be seen in Fig 3.9, which shows a clear correspondence between the fifty most highly ranked genes according to the Bayes Factor and p-values of the differential analysis. Interestingly, it also illustrates the similarity in using the estimated total causal effects and the log fold-change values (both in absolute value) to rank genes that are downstream causally related to the GHR gene.

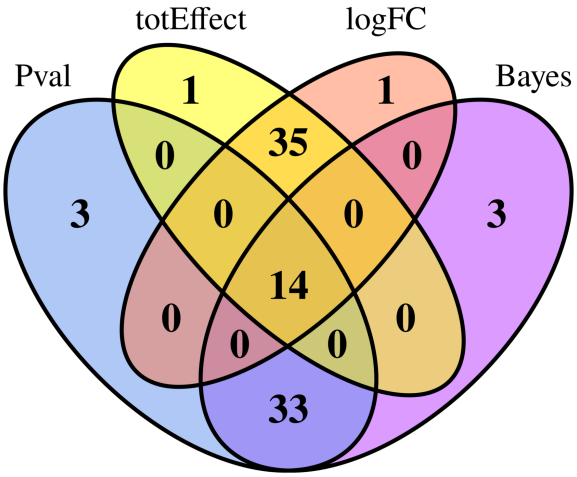


FIGURE 3.9 – **Venn diagram for the top 50 genes ranked according to the differential analysis (p-value or log-fold change) and marginal causal approach (Bayes factor or total causal effects.** Ranking was performed from lowest to highest for p-values and highest to lowest for absolute total effect, absolute log fold-change, and Bayes factor.

3.4 Discussion

We have proposed a novel approach to detect marginal causal relationships in high dimensional data when interventions are available for a single node of interest. This method was developed in the context of transcriptomic data, and can be particularly useful to perform a pre-selection of genes prior to a more thorough causal network inference. It is computationally efficient and can be simultaneously applied thousands of genes. In addition, our simulation study illustrated that the proposed method was able to accurately classify between downstream causal relationships and upstream or simple correlation relationships when the underlying DAG is unknown.

We showed that the results of differential analyses comparing KO to WT samples can indeed be interpreted as causal, given their similarity to the causal Gaussian Bayesian network. It is true that the new approach described here provides little or no improvement over classical differential analysis hypothesis tests. However, it is precisely through the new causal interpretation of these classical tests that our approach shows promise. For example, with the development of CRISPR/Cas9 genome editing [47], it is clear that the number of intervention experiments in molecular biology will increase dramatically in the coming years. In this context, although differential analysis is clearly a natural way to deal with fully controlled experiments (like randomized clinical trials), it is not particularly well-adapted for analyzing multi-factorial experiments and/or partially complete intervention designs. For these more complex types of studies, we believe that a test based on causal Gaussian Bayesian networks will be an innovative and efficient way to test and infer causality.

The proposed method relies on structural Gaussian equations, which assume linear relationships and graph acyclicity. Though not always biologically relevant, these assumptions are often made in causal gene network inference as they allow closed-form formulae of the likelihood functions to be obtained, which makes the proposed model very compu-

tationally efficient. It would be interesting in the future to evaluate whether results hold under less restrictive assumptions. The method presented here is defined in the context of interventions on one of the nodes of the network. It could similarly be applied to several interventions, if they are assumed independent of one another. However, if the interventions are causally linked, adjustments to the model would have to be considered. Finally, the proposed method was derived within an empirical Bayesian framework, where the maximum likelihood estimators were used. It would be interesting to investigate a fully Bayesian approach, using priors on the parameters that could include informative biological knowledge.

3.5 Références

[1] Pinna A, Heise S, Flassig RJ, de la Fuente A, Klamt S. Reconstruction of large-scale regulatory networks based on perturbation graphs and transitive reduction : improved methods and their evaluation. BMC Systems Biology. 2013 ;7(1). 43, 58

[2] Kalisch M, Bühlmann P. Estimating high-dimensional directed acyclic graphs with the PC-algorithm. The Journal of Machine Learning Research. 2007 ;8 :613–636. 43, 58

[3] Pinna A, Soranzo N, De La Fuente A. From knockouts to networks : establishing direct cause-effect relationships through graph analysis. PLoS One. 2010 ;5(10) :e12912. 24, 43, 58, 76, 91

[4] Chickering DM. Optimal structure identification with greedy search. The Journal of Machine Learning Research. 2003 ;3 :507–554. 21, 43, 58, 76

[5] Bühlmann P, Kalisch M, Meier L. High-Dimensional Statistics with a View Toward Applications in Biology. Annual Review of Statistics and Its Application. 2014 ;1 :255–278. 43, 58

[6] Chickering DM, Heckerman D, Meek C. A Bayesian approach to learning Bayesian networks with local structure. In : Proceedings of the Thirteenth conference on Uncertainty in artificial intelligence. Morgan Kaufmann Publishers Inc. ; 1997. p. 80–89. 43

[7] Luo W, Hankenson KD, Woolf PJ. Learning transcriptional regulatory networks from high throughput gene expression data using continuous three-way mutual information. BMC bioinformatics. 2008 ;9(1) :1. 43

[8] Watkinson J, Liang Kc, Wang X, Zheng T, Anastassiou D. Inference of Regulatory Gene Interactions from Expression Data Using Three-Way Mutual Information. Annals of the New York Academy of Sciences. 2009 ;1158(1) :302–313. 43

[9] Emmert-Streib F, Glazko G, De Matos Simoes R, et al. Statistical inference and reverse engineering of gene regulatory networks from observational expression data. Frontiers in genetics. 2012 ;3 :8. 43

[10] de Matos Simoes R, Dehmer M, Emmert-Streib F. Interfacing cellular networks of S. cerevisiae and E. coli : Connecting dynamic and genetic information. BMC genomics. 2013 ;14(1) :1. 43

[11] Friedman J, Hastie T, Tibshirani R. Sparse inverse covariance estimation with the graphical lasso. Biostatistics. 2008 ;9(3) :432–441. 10, 17, 44, 58

[12] Fu F, Zhou Q. Learning sparse causal Gaussian networks with experimental intervention : regularization and coordinate descent. Journal of the American Statistical Association. 2013 ;108(501) :288–300. 44, 64

[13] de Matos Simoes R, Emmert-Streib F. Bagging statistical network inference from large-scale gene expression data. PLoS One. 2012 ;7(3) :e33624 44, 76

[14] Basso K, Margolin AA, Stolovitzky G, Klein U, Dalla-Favera R, Califano A. Reverse engineering of regulatory networks in human B cells. Nature genetics. 2005 ;37(4) :382–390. 44

[15] Judea Pearl, TS Verma Equivalence and synthesis of causal models. In : Proceedings of Sixth Conference on Uncertainty in Artijicial Intelligence ; 1991. p. 220–227. 13, 44

[16] Marbach D, Prill RJ, Schaffter T, Mattiussi C, Floreano D, Stolovitzky G. Revealing strengths and weaknesses of methods for gene network inference. Proceedings of the National Academy of Sciences. 2010 ;107(14) :6286–6291. 24, 44, 67, 76, 91

[17] Altay G, Emmert-Streib F. Inferring the conservative causal core of gene regulatory networks. BMC Systems Biology. 2010 ;4(1) :132. 44

[18] Maathuis MH, Kalisch M, Bühlmann P, et al. Estimating high-dimensional intervention effects from observational data. The Annals of Statistics. 2009 ;37(6A) :3133–3164. 26, 44, 58

[19] Rau A, Jaffrézic F, Nuel G. Joint estimation of causal effects from observational and intervention gene expression data. BMC Systems Biology. 2013 ;7(1) :111. 26, 44, 58, 59, 76, 77, 86, 92

[20] Spirtes P, Glymour CN, Scheines R. Causation, prediction, and search. vol. 81. MIT press ; 2000. 10, 44, 76

[21] Pearl J. Causality : models, reasoning and inference. vol. 29. Cambridge Univ Press ; 2000. 14, 15, 34, 44, 45, 60

[22] Duriez B, Sobrier M, Duquesnoy P, Tixier-Boichard M, Decuypere E, Coquerelle G, et al. A naturally occurring growth hormone receptor mutation : in vivo and in vitro evidence for the functional importance of the WS motif common to all members of the cytokine receptor superfamily. Molecular endocrinology. 1993 ;7(6) :806–814. 44, 52

[23] Smyth GK. Linear Models and Empirical Bayes Methods for Assessing Differential Expression in Microarray Experiments. In : Statistical Applications in Genetics and Molecular Biology ; 2004. p. 1–25. 9, 49, 52

[24] Hauser A, Bühlmann P. Characterization and greedy learning of interventional Markov equivalence classes of directed acyclic graphs. Journal of Machine Learning Research. 2012 ;13(Aug) :2409–2464. 14, 50, 92

[25] Emmert-Streib F, Dehmer M, Haibe-Kains B. Gene regulatory networks and their applications : understanding biological and medical problems in terms of networks. Frontiers in Cell and Developmental Biology. 2014 ;2 :38. 9, 50

[26] Hsu PD, Lander ES, Zhang F. Development and applications of CRISPR-Cas9 for genome engineering. Cell. 2014 ;157(6) :1262–1278.
53

Chapitre 4

Estimation d'effets causaux par pénalisation L2

Ce chapitre a fait l'objet d'une publication en français dans le journal "Revue d'Intelligence Artificielle". Il s'agit d'une amélioration de l'algorithme MCMC-Ordre, dont le but premier est la passage à l'échelle. Il s'agit de l'implémentation d'une pénalisation L2, ce qui permet dans le même temps de permettre l'utilisation de la méthode même pour de petits jeux de données. L'utilisation d'un squelette est aussi envisagé, exemples à l'appui.

MONNERET, Gilles, JAFFREZIC, Florence, RAU, Andrea, et al. Estimation d'effets causaux dans les réseaux de régulation génique : vers la grande dimension. Revue des Sciences et Technologies de l'Information-Série RIA : Revue d'Intelligence Artificielle, 2015, vol. 29, no 2, p. 205-227.

4.1 Introduction

Depuis ces quinze dernières années, l'introduction des mathématiques au sein du domaine de la biologie se fait sentir. On a notamment une foule de données à traiter depuis l'avènement du séquençage ADN. Un des enjeux majeurs est de comprendre le rôle et les interactions de chacun des gènes codants, et pour cela une façon naturelle de modéliser ces informations est le graphe. Appliqué à la régulation génique, on parle de réseaux géniques : l'objectif est de trouver un lien causal entre différents gènes, en terme d'influence sur l'expression de ces derniers. Des applications bien concrètes peuvent bénéficier de ce genre de modèle comme la compréhension des mécanismes de certaines maladies.

Plusieurs méthodes ont déjà été proposées pour retrouver des liens entre différents gènes. Une revue générale de ces méthodes a été proposée par [9]. Une première méthode consiste à modéliser un graphe non orienté et utiliser la technique du *graphical lasso* pour sélectionner les arêtes [30]. Cela consiste à effectuer un calcul de vraisemblance pénalisée, la pénalisation contrôlant le nombre d'arêtes effectivement présentes dans le graphe obtenu. Cette méthode est cependant extrêmement sensible et de longues procédures de validation sont nécessaires en pratique pour la calibrer. Une autre approche proposée par Kalisch and Bühlmann [49] consiste à modéliser les données sous la forme d'un réseau bayésien gaussien, puis d'utiliser des tests de corrélations partielles pour déterminer la présence ou non de lien entre des variables [61; 62]. Elle n'utilise que des données d'observations et permet de retrouver le squelette du graphe ainsi que quelques orientations, mais elle ne peut retrouver l'intégralité des relations de causales. La maximisation de la vraisemblance sous une pénalité ℓ_0 permet d'arriver à un résultat similaire [15]. Une autre approche utilisant la théorie des graphes a été proposée par Pinna [75; 76] consiste à utiliser des matrices de déviation : en intervenant sur un noeud, on peut mesurer directement l'effet sur les autres noeuds. Si on voit un effet fort, on peut considérer qu'il y a une relation causale. Cette méthode nécessite cependant des interventions sur chacun des noeuds d'intérêt. Enfin récemment, il a été proposé une méthode utilisant le maximum de vraisemblance combiné aux méthodes de Monte Carlo pour trouver un ordre partiel sur les données [77]. Elle apporte l'avantage de pouvoir intégrer des données d'observation et d'intervention sur un ou plusieurs noeuds.

Bien sûr, modéliser toute la complexité du vivant par un simple graphe est une grossière approximation mais celle-ci est justifiée par la qualité des résultats obtenus en terme de fidélité aux données. Celles-ci sont obtenues grâce à des techniques récentes, notamment les puces à ADN, le séquençage et l'invalidation génétique. Ces techniques sont nécessaires pour non seulement obtenir des données transcriptomiques d'observations, mais aussi d'interventions. Ces dernières données sont absolument nécessaires car ce sont elles qui nous permettent de passer du monde de l'association à celui de la causalité. Malheureusement du fait de limitations financières, le nombre de réplicats disponibles par population est limité, réduisant l'applicabilité des méthodes déjà existantes dans la littérature. Par ailleurs, bien que les techniques usant de réseaux bayésiens sont maintenant bien connues, l'intégration des données d'interventions est récente.

L'objectif de cet article est le suivant : en se basant sur la méthode proposée dans Rau et al. [77], nous décrivons une méthode s'appliquant à des réseaux de grande dimension, ne nécessitant qu'un faible nombre de réplicats et dont la qualité des résultats est satisfaisante.

La première section sera centrée sur la notion de causalité, nous rappellerons la différence entre causalité et corrélation et nous introduirons l'opérateur `do` de Pearl. Dans la deuxième section, nous introduirons la méthode MCMC-Mallows développée par Rau

et al. [77]. Dans une troisième partie, nous proposerons des modifications à la précédente technique permettant un calcul avec un temps et un nombre de réplicats raisonnable. Enfin, nous proposerons une série de simulations permettant d'étudier l'apport de ces différentes modifications.

4.2 Causalité dans l'expression des gènes

4.2.1 Fondement biologique

Un des objectifs des biologistes et généticiens est de mettre en lumière les liens causaux entre les gènes. Biologiquement, les différents gènes peuvent par exemple coder une protéine (facteur de transcription) qui modulera l'activité d'un autre gène. Cela se fait par l'intermédiaire de l'ARNm, molécule qui fait le lien entre les informations du gène et sa traduction en protéine. Les données utilisées sont des données transcriptomiques, c'est-à-dire des quantités d'expressions d'ARNm, mesurées par une puce à ADN. Celle-ci consiste en une plaque sur laquelle est fixée des brins d'ADN, sur lesquels se fixeront des morceaux complémentaires obtenus par transcription inverse de l'ARNm. Nous obtenons ainsi une mesure en continue de l'intensité de l'hybridation entre la molécule d'ARNm cible et le morceau d'ADN fixé sur la plaque : c'est ce que l'on appelle des données d'expression. Nous supposons que l'analyse se place à l'équilibre, après un temps "long" : cela justifie l'approximation que le système ne dispose ni de boucle, ni de rétroaction. Une autre façon d'obtenir des données est d'utiliser un séquenceur ADN. La technique consiste à bloquer spécifiquement la polymérisation sur chacune des quatre bases azotées, via des bases modifiées et marquées. On obtient des brins d'ADN de taille croissante se terminant tous par une base marquée. Nous obtenons des données de comptage. Il est nécessaire d'obtenir des données d'interventions : ces données sont construites à partir de la technique de l'invalidation génétique, aussi appelée "knock-out" qui consiste à éteindre l'expression d'un certain gène soit en l'inhibant, soit en l'excluant. On peut aussi utiliser des techniques de réduction d'expression, le "knock-down".

Notre travail de statisticien consiste à retrouver de l'information à partir des différentes données collectées : on cherche les liens causaux entre les différents noeuds à partir des différentes quantités d'expressions obtenues. Savoir comment sont récoltées les différentes données est primordial puisque cela influe fortement sur la modélisation mathématique que l'on propose.

4.2.2 Causalité et équivalence de Markov

On se place dans le cadre d'une chaîne de Markov à 3 noeuds (X,Y,Z) et on essaye de retrouver la structure du DAG à partir de la loi de probabilité. On s'aperçoit alors que deux cas possibles sont strictement identiques du point de vue des probabilités :

$$\begin{aligned}\mathbb{P}(X)\mathbb{P}(Y|X)\mathbb{P}(Z|Y) &= \frac{\mathbb{P}(X)\mathbb{P}(Y,X)\mathbb{P}(Y,Z)\mathbb{P}(Z)}{\mathbb{P}(X)\mathbb{P}(Y)\mathbb{P}(Z)} \\ &= \mathbb{P}(Z)\mathbb{P}(Y|Z)\mathbb{P}(X|Y).\end{aligned}$$

On parle de DAG équivalents aux sens de Markov (FIGURE 4.1).

Il devient naturel de vouloir représenter la classe d'équivalence associée à cette relation d'équivalence : ceci est fait avec les graphes essentiels (ou CPDAG, *Completed Partially Directed Acyclic Graph*, voir FIGURE 4.2). Pour construire cet objet, on passe par la notion de *Partially Directed Acyclic Graph*. Il s'agit d'un graphe possédant des arêtes

Graphe équivalent au sens de Markov

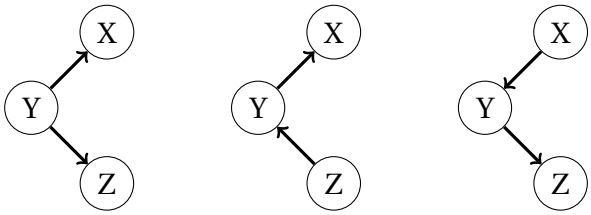


FIGURE 4.1

Exemple de CPDAG : a gauche un DAG et à droite le CPDAG associé

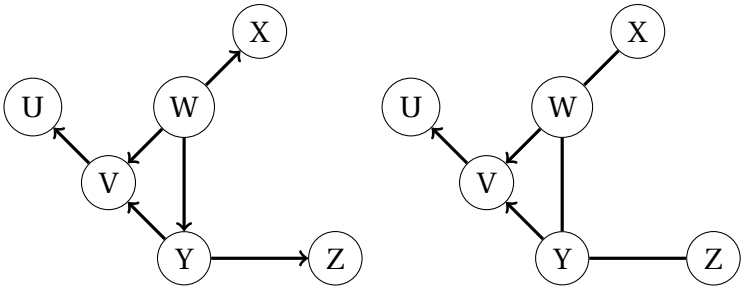


FIGURE 4.2

orientées et des arêtes non orientées tel qu'aucun cycle orienté ne soit présent. Un DAG est équivalent à un PDAG s'il dispose du même squelette et des mêmes structures en v. Un CPDAG est un PDAG tel que :

— Si une arête orientée est présente dans l'ensemble des DAG de la classe d'équivalence, elle est présente dans le CPDAG ;

— Pour toute arête non orientée i— j du CPDAG, il existe un DAG appartenant à la classe d'équivalence tel que $i \rightarrow j$ et un DAG tel que $i \leftarrow j$.

On le voit, du fait de l'identifiabilité de la distribution sous-jacente, des observations ne peuvent nous donner que la classe d'équivalence associée à un DAG et non le DAG lui-même. On ne peut donc retrouver l'ensemble des relations causales à partir de simples observations. Pour contourner ce problème, on va ajouter des interventions : en fixant la valeur d'un noeud, on crée un nouveau modèle informatif. La traduction mathématique de cette opération est l'opérateur `do`, qui consiste à briser les arêtes des ascendants [74]. Dans un tel modèle, nous pouvons orienter les arêtes autour de la perturbation. Les informations que l'on récolte de ce modèle perturbé rend possible la description des liens

Description graphique de l'opérateur `do`

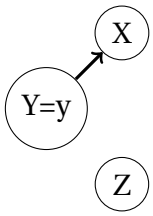


FIGURE 4.3

causaux dans le modèle initial. Soit X et Y deux noeuds d'un graphe G, on dit que X est parent de Y si l'arrête X → Y est présente dans G. On dénote l'ensemble des parents de X_i par pa_i ou $\mathtt{pa}(X_i)$.

Definition de l'opérateur do : Soit $(x_i)_{i\in 1,\dots,n}$ des variables à valeur dans $\mathbb{R}$ et $(f_i)_{i=1,\dots,n}$ des fonctions. On définit un modèle causal par l'ensemble des équations du type $x_i = f_i(pa_i), \quad i = 1,\dots,n$. Soit deux ensembles de variables disjoints X et Y, l'effet causal de X sur Y dénoté par $\mathbb{P}(y|\,\mathtt{do}(X = x))$ est une fonction de X dans l'ensemble des distributions sur Y. Pour chaque réalisation x de X, $\mathbb{P}(y|\,\mathtt{do}(X = x))$ donne la probabilité de $Y = y$ induite en supprimant du modèle toutes les équations définissant les éléments de X et en remplaçant $X = x$ dans les équations restantes.

On insiste sur la différence entre les quantités $\mathbb{P}(X|\,\mathtt{do}(Y = y))$ et $\mathbb{P}(X|Y = y)$. Tandis que la première formulation sous-entend un bouleversement du modèle, la seconde notation introduit une connaissance toute chose égale par ailleurs.

En posant X_{-j} le vecteur X privé de sa j-iéme composante, on définit les effets causaux directs $\left(\alpha_{i,j}\right)_{i,j}$ de la façon suivante :

$$\alpha_{i,j} = \frac{\partial}{\partial x}\mathbb{E}(X_j|X_{-j}, \mathtt{do}(X_i = x)).$$

Cela correspond à l'effet direct d'une variable sur une autre : c'est-à-dire qu'on ne prend en compte que le lien direct entre ces deux variables, les autres étant fixées. C'est souvent cette relation que l'on veut mettre en évidence. On définit les effets causaux totaux $\left(\beta_{i,j}\right)_{i,j}$ par :

$$\beta_{i,j} = \frac{\partial}{\partial x}\mathbb{E}(X_j|\,\mathtt{do}(X_i = x)).$$

Cette quantité décrit l'ensemble des effets d'une variable sur une autre, y compris par l'intermédiaire d'autres variables.

4.3 Navigation dans l'espace des ordres : MCMC-Mallows

4.3.1 Estimateur du maximum de vraisemblance

Nous allons présenter dans cette partie la stratégie de Rau et al. (2013) qui contrairement à la direction prise par Bülhmann, consiste en la simulation d'un DAG complet à partir de données d'observations et d'interventions. On se place dans le cadre d'un réseau bayésien gaussien : soit $\mathbf{X} = (X_1,\dots,X_p)$ les p variables gaussiennes attachées aux p noeuds. On inclut les liens causaux de la façon suivante :

$$\mathbf{X} = \boldsymbol{m} + \mathbf{X}\mathbf{W} + \boldsymbol{\epsilon}$$

où $\boldsymbol{m}$ est le vecteur des moyennes, $\mathbf{W}$ la matrice des effets causaux directs et $\boldsymbol{\epsilon}$ le vecteur de bruit de variance $\boldsymbol{\sigma}^2$. L'objectif est de réussir à inférer la matrice $\mathbf{W}$ à partir des données disponibles. On suppose que l'on dispose de l'ordre topologique des données, c'est-à-dire que $i < j$ seulement si $i \in \mathtt{pa}(j)$. Nous souhaitons un DAG, nous cherchons donc une matrice $\mathbf{W}$ triangulaire supérieure. En identifiant la loi $\mathbb{P}(X_j|\,\mathtt{do}(X_i = x), \mathtt{pa}(X_j))$, on s'aperçoit rapidement que cette matrice correspond à la matrice des effets causaux directs. Plutôt que de viser la matrice des effets causaux directs $\mathbf{W}$, on peut se fixer comme objectif l'estimation de la matrice des effets causaux totaux $\mathbf{L} = (\mathbf{I} - \mathbf{W})^{-1} = \mathbf{I} + \mathbf{W} + \mathbf{W}^2 + \ldots + \mathbf{W}^{p-1}$. Cette seconde expression n'est possible que grâce à la nilpotence de $\mathbf{W}$, induite par le caractère DAG de notre modèle. Les valeurs de cette matrice $\mathbf{L}$ désignent l'ensemble des

effets d'une variable sur une autre, directs ou indirects. On introduit quelques notations : soit $\boldsymbol{x}$ l'ensemble des données à disposition, on note $\mathscr{J}_k$ l'ensemble des noeuds pour lesquels on a effectué une intervention au réplicat k. On note $\mathscr{K}_j$ l'ensemble des réplicats tel qu'il n'y ait pas d'intervention sur le noeud j et $\mathrm{N}_j = \mathrm{card}(\mathscr{K}_j)$. Nous pouvons écrire alors la log-vraisemblance [71], le détail du calcul est laissé en annexe :

$$\ell(\boldsymbol{m}, \mathbf{W}, \boldsymbol{\sigma^2}) = -\frac{\log(2\pi)}{2}\sum_j \mathrm{N}_j - \sum_j \mathrm{N}_j \log(\sigma_j) - \frac{1}{2}\sum_k \sum_{j \notin \mathscr{J}_k} \frac{1}{\sigma_2^2}(x_j^k - \boldsymbol{x}^k \mathbf{W} \boldsymbol{e}_j^{\mathrm{T}} - m_j)^2.$$

On maximise cette vraisemblance en utilisant le gradient par rapport aux différentes quantités. On obtient :

$$m_j = \frac{1}{\mathrm{N}_j}\sum_{k \in \mathscr{K}_j}\left(x_j^k - \boldsymbol{x}^k \mathbf{W} \boldsymbol{e}_j^{\mathrm{T}}\right).$$

On peut réecrire la log-vraisemblance en recentrant les données, c'est-à-dire en posant :

$$\boldsymbol{y}^{k,j} = \boldsymbol{x}^k - \frac{1}{\mathrm{N}_j}\sum_{k' \in \mathscr{K}_j} \boldsymbol{x}^{k'}.$$

Nous obtenons alors :

$$\tilde{\ell}(\boldsymbol{\sigma}, \mathbf{W}) = -\frac{\log(2\pi)}{2}\sum_j \mathrm{N}_j - \sum_j \mathrm{N}_j \log(\sigma_j) - \frac{1}{2}\sum_k \sum_{j \notin \mathscr{J}_k} \frac{1}{\sigma_j^2}\left(y_j^{k,j} - \boldsymbol{y}^{k,j} \mathbf{W} \boldsymbol{e}_j^{\mathrm{T}}\right)^2.$$

De cette façon, on peut annuler le gradient et obtenir $\mathbf{W}$ comme solution d'un système linéaire :

$$\sum_{i',(i',j)\in\mathscr{E}} w_{(i',j)} \sum_{k \in \mathscr{K}_j} y_i^{k,j} y_{i'}^{k,j} = \sum_{k \in \mathscr{K}_j} y_i^{k,j} y_j^{k,j} \qquad \forall (i,j) \in \mathscr{E}.$$

Ce système peut toutefois être dégénéré si nous ne disposons pas de données suffisantes ou si les interventions n'apportent pas assez d'informations. On peut enfin obtenir $\boldsymbol{\sigma}$ de la manière suivante :

$$\sigma_j^2 = \frac{1}{\mathrm{N}_j}\sum_{k \notin \mathscr{K}_j}\left(y_j^{k,j} - \boldsymbol{y}^{k,j} \mathbf{W} \boldsymbol{e}_j^{\mathrm{T}}\right)^2.$$

Nous disposons ainsi, à ordre connu, de l'estimateur de maximum de vraisemblance. Cependant, dans les cas concrets, nous n'avons pas de structure prédéfinie sur les données et nous devons ainsi découvrir un ordre topologique acceptable.

4.3.2 Architecture Métropolis-Hastings

On travaille avec une approche bayésienne empirique : on se donne une loi *a priori* uniforme et on tente de trouver une loi *a posteriori* des ordres :

$$\begin{aligned} \mathbb{P}(o|\text{données}) &\propto \int_\theta \mathbb{P}(\text{données}|\theta, o)\mathbb{P}(\theta|o)\mathbb{P}(o)\,d\theta \\ &\propto \mathbb{P}(\text{données}|\hat{\theta}_o)\mathbb{P}(o). \end{aligned}$$

La seconde ligne est une approximation qui caractérise le bayésien empirique : nous supposons que la loi en θ est proche d'une masse entièrement concentrée en son maximum de vraisemblance. C'est ainsi qu'on voit apparaître l'expression de la vraisemblance et on a accès à cette loi *a posteriori*. On va se placer dans un cadre Hastings-Métropolis pour réussir à simuler cette loi [42]. On utilise pour cela une loi de proposition de Mallows [65]. C'est une loi symétrique, qui admet un paramètre de mode et un paramètre de dispersion. La symétrie est très intéressante car elle permet de simplifier le test d'acceptation.

Son paramètre de dispersion qu'on appelle sa température, nous permet de calibrer la loi de proposition pour explorer l'espace au mieux. Cette température est choisie en fonction du taux d'acceptation : on choisit une température qui donnera un taux d'acceptation autour de 25% ce qui nous semble être un bon compromis par analogie avec la loi multinormale où il a été prouvé que le taux d'acceptation optimal est aux alentours de 25% [81]. La loi de Mallows permet de simuler des permutations aléatoires d'un vecteur, on va donc l'utiliser pour tirer des ordres aléatoirement. L'algorithme que l'on utilise est le suivant :

Algorithme 10 : MCMC-Order

Entrées : Ordre initial o, Données $\boldsymbol{x}$, Température β, nombre d'itérations n
Sorties : Echantillons $o_1, \ldots, o_n$
pour *i=1 :n* **faire**
  Simuler $y_n \sim \texttt{rmallow}(o_{n-1}, \beta)$;
  $o_n = y_n$ avec probabilité $\min\left(1, \frac{\mathbb{P}(\text{données}|\hat{\theta}_{y_n})}{\mathbb{P}(\text{données}|\hat{\theta}_{o_{n-1}})}\right)$;
  Sinon $o_n = o_{n-1}$

En sortie de l'algorithme, nous obtenons une trajectoire d'ordres et de paramètres associés. En effectuant les moyennes empiriques sur les matrices des effets causaux directs et totaux, nous nous approchons de l'estimation des mécanismes sous-jacents. Nous obtenons cependant un graphe complet, avec des effets variables. Nous devons choisir un seuil en valeur absolue qui nous permet de choisir quelles arêtes ont un effet suffisant pour être estimées présentes. Ce seuil est choisi en pratique avec une procédure de validation croisée.

4.3.3 Limites de la méthode

La méthode proposée a été testée sur des petits réseaux de gènes, de l'ordre d'une dizaine de noeuds, et elle montre déjà quelques limites. Tout d'abord le nombre de réplicats nécessaires pour l'estimation est de l'ordre du nombre de noeuds, et en pratique nous ne les avons pas toujours. En effet, la résolution du système linéaire donnant $\mathbf{W}$ est de rang maximum du nombre de réplicats.

Même si nous avons un nombre de réplicats suffisant, un effet de surajustement apparaît, du fait de l'espace des paramètres. Celui-ci est de l'ordre du nombre de nœuds au carré, il faut réussir à traiter ce problème avant de pouvoir espérer passer à des problèmes de plus grande dimension. Enfin la navigation par MCMC pose deux problèmes : le premier est le temps de calcul. En petite dimension, l'estimation est immédiate, le temps de calcul pour une itération est donc très faible. Lorsque l'on augmente la dimension, les calculs peuvent être plus lents, rendant alors la procédure MCMC extrêmement lente. D'autre part, l'espace des ordres croît de manière factorielle en fonction du nombre de nœuds, il est nécessaire alors d'avoir beaucoup plus d'itérations pour pouvoir convenablement couvrir l'espace et valider notre procédure.

4.4 Vers la grande dimension : Pénalisation ridge & squelette

4.4.1 Vraisemblance pénalisée

Estimateur du maximum de vraisemblance pénalisée

Une méthode reconnue pour solutionner le problème de surajustement est d'introduire une pénalité dans l'opération de maximisation de la vraisemblance. Cette méthode a déjà été proposée dans le cadre de réseaux bayésiens [33]. Nous choisissons une pénalité ridge pour éviter de sélectionner les arêtes et permettre ainsi même les petites contributions. L'objectif sous-jacent est de ne pas pénaliser l'estimation de la matrice des effets causaux totaux $\mathbf{L}$. Soit $\lambda \in \mathbb{R}$, la vraisemblance pénalisée s'écrit :

$$\ell(\boldsymbol{m}, \mathbf{W}, \boldsymbol{\sigma^2}) = -\frac{\log(2\pi)}{2}\sum_j \mathrm{N}_j - \sum_j \mathrm{N}_j \log(\sigma_j) \\ -\frac{1}{2}\sum_k \sum_{j \notin \mathscr{J}_k} \frac{1}{\sigma_2^2}(x_j^k - \boldsymbol{x}^k \mathbf{W} \boldsymbol{e}_j^{\mathrm{T}} - m_j)^2 - \frac{1}{2}\lambda \sum_{(i,j)\in\mathscr{E}} \left(\mathbf{W}_{ij}\right)^2.$$

En maximisant en $\boldsymbol{m}$, on obtient le même estimateur que précédemment.

$$\mathbf{m} = \frac{1}{\mathrm{N}_j}\sum_{k\in\mathscr{K}_j}(x_j^k - \boldsymbol{x}^k \mathbf{W} \boldsymbol{e}_j^{\mathrm{T}}).$$

On réinjecte cette solution et on recentre les données en posant $\boldsymbol{y}^{k,j} = x^k - \frac{1}{\mathrm{N}_j}\sum_{k'\in\mathscr{K}_j}$, on obtient cette fonctionnelle à maximiser en $(\mathbf{W}, \boldsymbol{\sigma}^2)$:

$$\tilde{\ell}(\boldsymbol{\sigma}, \mathbf{W}) = \mathrm{Cst} - \sum_j \mathrm{N}_j \log(\sigma_j) - \frac{1}{2}\sum_k \sum_{j\notin\mathscr{J}_k} \frac{1}{\sigma_j^2}\left(y_j^{k,j} - \boldsymbol{y}^{k,j}\mathbf{W}\boldsymbol{e}_j^{\mathrm{T}}\right)^2 - \frac{1}{2}\lambda \sum_{(i,j)\in\mathscr{E}} \left(\mathbf{W}_{ij}\right)^2.$$

La solution à ce problème se dérive facilement, $\mathbf{W}$ s'obtient comme résolution du système linéaire suivant :

$$\sum_{i',(i',j)\in\mathscr{E}} w_{(i',j)} \sum_{k\in\mathscr{K}_j} y_i^{k,j} y_{i'}^{k,j} = \sum_{k\in\mathscr{K}_j} y_i^{k,j} y_j^{k,j} - \lambda\sigma_j^2 \mathbf{W}_{(i,j)} \qquad \forall\,(i,j)\in\mathscr{E}.$$

Et $\boldsymbol{\sigma}^2$ de la façon suivante :

$$\sigma_j^2 = \frac{1}{\mathrm{N}_j}\sum_{k\in\mathscr{K}_j}(y_j^{k,j} - \boldsymbol{y}^{k,j}\mathbf{W}\boldsymbol{e}_j^{\mathrm{T}})^2.$$

Nous obtenons donc un estimateur sensiblement différent, notamment dans l'estimation de $\mathbf{W}$. En effet, reformulons le système linéaire à résoudre dans le cas non pénalisé. On voit pour cela $\mathbf{W}$ non plus comme une matrice, mais comme un vecteur dont les différents éléments correspondent aux différents effets causaux non nuls. On considère que ce vecteur est construit en lisant la matrice colonne par colonne. Alors le système à résoudre

est $\mathbf{AW} = \boldsymbol{b}$, où $\mathbf{A}$ est une matrice diagonale par bloc :

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_2 & 0 & \dots & \dots & 0 \\ 0 & \mathbf{A}_3 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \dots & \dots & 0 & \mathbf{A}_p \end{pmatrix}$$

$$\mathbf{A}_j = \left(\sum_{k \in \mathcal{K}_j} y_i^{k,j} y_{i'}^{k,j}\right)_{i,i'} \qquad (i,j) \in \mathcal{E}, (i',j) \in \mathcal{E}.$$

Et $\boldsymbol{b}$ s'écrit simplement :

$$\boldsymbol{b} = \left(\sum_{k \in \mathcal{K}_j} y_i^{k,j} y_j^{k,j}\right)_{(i,j) \in \mathcal{E}}.$$

Nous voyons que le rang de chacun des blocs $\mathbf{A}_j$ est borné par le nombre de réplicats. En effet le rang d'une somme de matrices est inférieur ou égal à la somme des rangs, et on a $\mathrm{rg}(xx^t) = 1$. Enfin la matrice $\mathbf{A}$ étant diagonale par bloc, son rang est égal à la somme des rangs de chacun des blocs. Donc la matrice $\mathbf{A}$ n'est pas de rang plein si le nombre de réplicats est inférieur au nombre de noeuds. Dans la formule suivante les vecteurs $(\tilde{y}^{k,j})_{k,j}$ correspondent aux vecteurs $(y^{k,j})_{k,j}$ tels que seul les indices i tel que $(i,j) \in \mathcal{E}$ sont retenus.

$$\mathrm{rg}(\mathbf{A}) = \sum_{j=2}^{p} \mathrm{rg}(\mathbf{A}_j) \leq \sum_{j=2}^{p} \sum_{k \in \mathcal{K}_j} \mathrm{rg}((\tilde{y}^{k,j})^{\mathrm{T}} \tilde{y}^{k,j}).$$

Dans le cas de la vraisemblance pénalisée, les blocs diagonaux de la matrice A sont légèrement différents (δ est ici le symbole de Kronecker) :

$$\mathbf{A}_j = \left(\sum_{k \in \mathcal{K}_j} y_i^{k,j} y_{i'}^{k,j}\right)_{i,i'} + \delta_{(i,i')} \lambda \sigma_j \quad (i,j) \in \mathcal{E}, (i',j) \in \mathcal{E}.$$

Nous voyons ici clairement un résultat connu de la pénalisation Ridge : par addition d'une constante positive sur la diagonale, le conditionnement du système linéaire est amélioré.

Avec cet ajout nous ajoutons un lien entre les différents paramètres : l'espace des paramètres est réduit en conséquence. Le modèle ainsi défini gagne en robustesse par rapport à l'estimateur du maximum de vraisemblance, mais contrairement à ce dernier il est biaisé. Nous devons calibrer la pénalité choisie : nous nous retrouvons dans le cadre de la sélection de modèle, avec comme objectif le choix du meilleur compromis entre biais et variance. En effet, en augmentant la pénalité, nous diminuons le nombre de paramètres effectifs du modèle, nous amenant dans le cas limite à des estimateurs nuls. A l'inverse, si la pénalité est nulle, nous nous ramenons à l'estimateur du maximum de vraisemblance.

Le problème de σ

Dans les expressions précédentes, nous utilisons $\boldsymbol{\sigma}$ pour minimiser $\mathbf{W}$, et $\mathbf{W}$ pour minimiser $\boldsymbol{\sigma}$. Nous avons un problème quant à la façon de procéder à cette minimisation. Nous avons choisi une méthode simple qui a pour avantage de ne pas alourdir la procédure. Elle consiste à estimer à chaque étape la matrice $\mathbf{W}$ à partir du paramètre de dispersion de l'itération précédente. C'est-à-dire :

$$\begin{aligned} \mathbf{W}_k &= \mathrm{argmax}_{\mathbf{W}} \ell(\boldsymbol{m}_k, \mathbf{W}, \boldsymbol{\sigma}_{k-1}) \\ \boldsymbol{\sigma}_k &= \mathrm{argmax}_{\boldsymbol{\sigma}} \ell(\boldsymbol{m}_k, \mathbf{W}_k, \boldsymbol{\sigma}). \end{aligned}$$

Ceci est justifié par la stabilité de $\boldsymbol{\sigma}$ au cours des itérations successives. On donne un exemple d'une évolution possible dans la FIGURE 4.4.

Evolution d'un paramètre de dispersion. 10 noeuds, 10000 itérations.

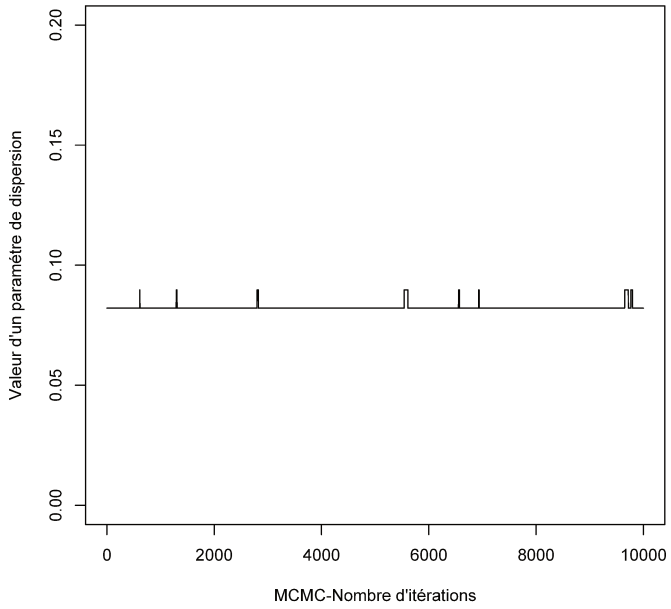


FIGURE 4.4

Nous voyons dans cet exemple que le paramètre tout en effectuant certains sauts correspondant à des changements de configuration reste dans une fenêtre de valeur restreinte, entre 0.082 et 0.090, c'est-à-dire 8% de variation. C'est pourquoi nous avons adopté cette méthode.

4.4.2 Squelette

Une idée intuitive pour accélérer l'algorithme et réduire l'erreur de prédiction est d'introduire un squelette déjà connu. En effet, en supposant que l'on dispose d'un tel squelette, par exemple grâce à des modèles biologiques externes, nous pouvons améliorer fortement la rapidité de notre algorithme. D'une part, l'estimation de la matrice des effets causaux directs nécessite la résolution d'un système linéaire de la taille du nombre d'arêtes. Cela est crucial, étant donné que la plupart des réseaux rencontrés sont creux. D'autre part, l'étape MCMC en est fortement améliorée en terme de couverture d'espace, d'exploration et de loi *a posteriori*. Dans la FIGURE 4.5, on a un exemple décrivant deux DAGs à partir d'un même squelette. Nous pouvons voir le squelette à gauche, au centre un DAG déterminé par l'ordre (D, C, B, A, E, F) ou (D, C, B, E, A, F) et tout à droite un DAG déterminé par l'ordre (F, B, C, E, A, D).

Nous voyons que le DAG du milieu peut être construit à partir de deux ordres : cela illustre le fait que dans le cas de l'introduction d'un squelette, l'espace des ordres est plus grand que l'espace des DAG.

DAGs issus d'un même squelette (à gauche)

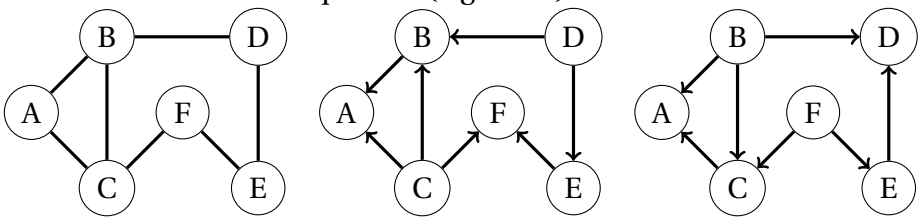


FIGURE 4.5

4.5 Simulations et résultats

Nous présentons ici quelques résultats obtenus par simulation. Nous expliquons tout d'abord le procédé qui nous donne notre modèle numérique, préliminaire à nos expérimentations. Le modèle numérique proposé est le suivant : on se donne un squelette de graphe (issu dans nos simulations des données DREAM 4 [66]) et nous choisissons aléatoirement un ordre topologique sur ce squelette. Ceci nous assure que le graphe ainsi formé est un DAG. Nous simulons ensuite aléatoirement les poids des différentes arêtes. Nous obtenons donc un modèle de référence qui nous sert à tirer des variables aléatoires.

4.5.1 Estimation ridge

Nous voulons vérifier que l'estimation ridge améliore de façon sensible la qualité des résultats. Nous effectuons pour cela l'expérience suivante : on se donne une observation et des interventions (simulant une invalidation génétique) sur chacun des noeuds, ainsi que l'ordre topologique. On utilise sur ces données l'estimateur du maximum de vraisemblance et sa version pénalisée pour calculer la matrice des effets causaux directs $\mathbf{W}$ et la matrice des effets causaux totaux $\mathbf{L}$. On calcule ensuite l'erreur quadratique que l'on obtient vis-à-vis du véritable modèle en fonction de la pénalisation utilisée. Les résultats sont présentés FIGURE 4.6. Nous voyons très nettement qu'une pénalisation permet d'améliorer grandement les résultats. Le problème est le choix de cette pénalisation : nous voyons ici qu'une pénalisation égale à 1.0 apporte de bons résultats, mais en pratique nous ne connaissons pas le véritable modèle et l'on doit donc utiliser une procédure de validation croisée.

Nous voulons aussi vérifier que le nombre d'estimations nécessaire est bien réduit par cette méthode. Pour cela on simule un jeu de données d'observations croissant (et uniquement des observations). Nous estimons le meilleur paramètre de régularisation pour p observations. Nous l'utilisons ensuite pour comparer les deux estimateurs en fonction du nombre d'observations à disposition. Le procédé est répété une centaine de fois et une moyenne est présentée pour le cas 10 noeuds. Les résultats sont présentés FIGURE 4.7. Les résultats sont clairs : comme attendu, plus le nombre d'observations augmente et plus notre erreur se réduit. Mais c'est surtout la capacité de calcul qui est surprenante : avec la régularisation on est capable de calculer notre estimateur avec un très faible nombre d'observations, avec une erreur raisonnable. Cependant pour 100 noeuds, on voit apparaître un effet étonnant avec peu d'observations : le résultat semble meilleur qu'avec une centaine d'observations. On explique cette singularité par notre façon d'estimer la variance.

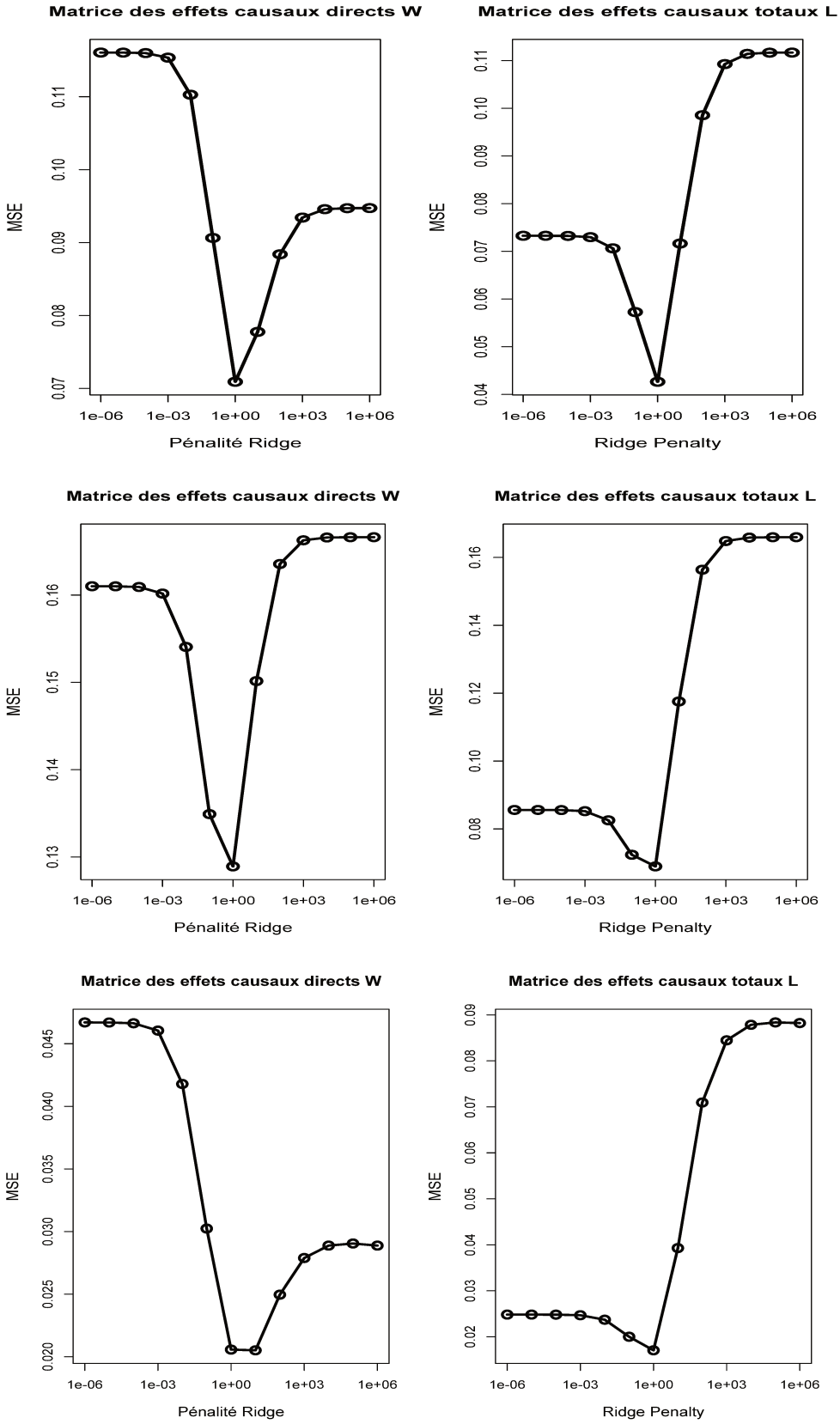


FIGURE 4.6 – Quelques résultats pour différentes simulations. Les résultats en haut et au centre sont issus d'un graphe à 10 noeuds, le dernier d'un graphe à 100 noeuds.

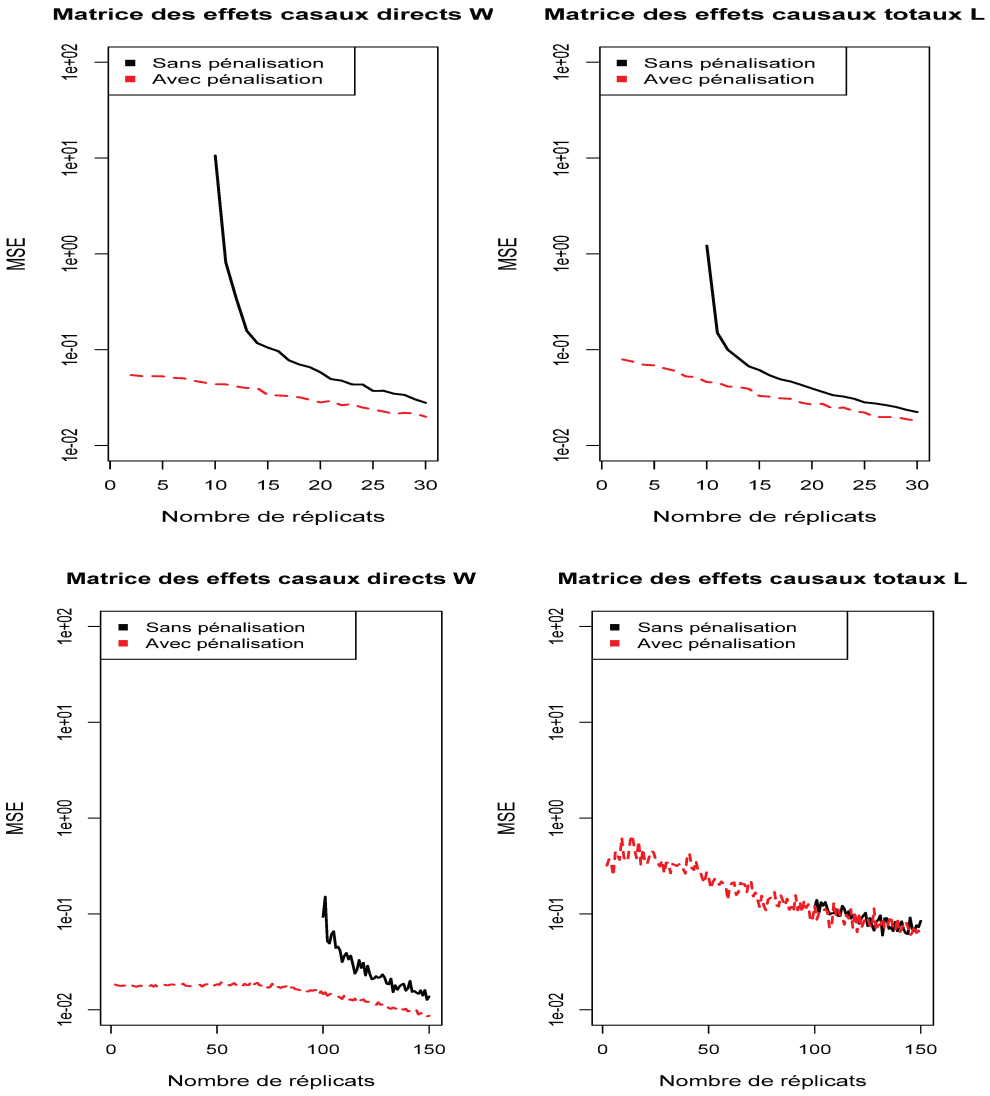


FIGURE 4.7 – Evolution de l'erreur quadratique en fonction du nombre d'observations, avec ou sans régularisation. L'absence de résultat indique une impossibilité de calcul. En haut 10 noeuds, en bas 100 noeuds.

Temperature	Sans squelette	Avec squelette
0.1	0.9995	1.0000
0.3	0.8285	0.8915
0.5	0.5170	0.6430
0.7	0.2910	0.3940
0.9	0.1520	0.2325
1	0.1240	0.2000
10	0.0040	0.0050

TABLEAU 4.1 – Taux d'acceptation pour différentes températures

	Nombre N d'arêtes	Temps de calcul pour l'EMV	Temps de calcul pour 50.000 itérations
Sans squelette	5050	30s	17 jours
Avec squelette	250	<1s	2 heures

TABLEAU 4.2 – Temps de calcul en fonction du nombre d'arêtes pour un exemple de 100 noeuds, avec ou sans squelette(EMV : Estimateur du maximum de vraisemblance)

4.5.2 Squelette

On tente de quantifier l'apport du squelette à notre méthode. Une première expérience a pour but de mesurer l'apport en terme de qualité d'estimation. On se place dans le cas simple de l'utilisation de l'estimateur du maximum de vraisemblance, et nous comparons l'erreur quadratique entre les cas graphe "plein" et avec le "vrai" squelette. Les données que l'on considère sont une observation et une intervention sur chacun des noeuds. Les résultats présentés dans le tableau suivant sont une moyenne de 100 simulations.

L'apport de squelette est indiscutable, nous obtenons des résultats de 20 à 2000 fois plus précis. Par notre approche MCMC, l'apport du squelette peut aussi aider à naviguer dans l'espace des ordres. Pour nous en convaincre, on vérifie l'ordre *a posteriori* obtenu et le taux d'acceptation en fonction de la température. On note que le taux d'acceptation est plus élevé, permettant une plus grande température de la loi de proposition et ainsi une plus grande couverture de l'espace. L'ordre *a posteriori* est aussi bien plus fidèle à ce que l'on espérait, nous pouvons le voir FIGURE 4.8. Cette figure est à comprendre dans ce sens : sur l'ordonnée sont classés les noeuds selon un ordre topologique acceptable vis-à-vis du DAG de référence. Sur l'abscisse est indiqué l'ordre dans lequel le noeud est classé, de tel sorte qu'une diagonale indique le meilleur résultat possible. Plus une case est de forte intensité et plus la fréquence de cet événement est grande dans notre simulation. On peut donc le voir, comme attendu le squelette apporte un énorme gain de vitesse et de précision.

	MSE **W**	MSE **L**
Sans squelette	271.61	21.18
Avec squelette	0.17	0.02

TABLEAU 4.3 – Erreur quadratique moyenne avec ou sans squelette

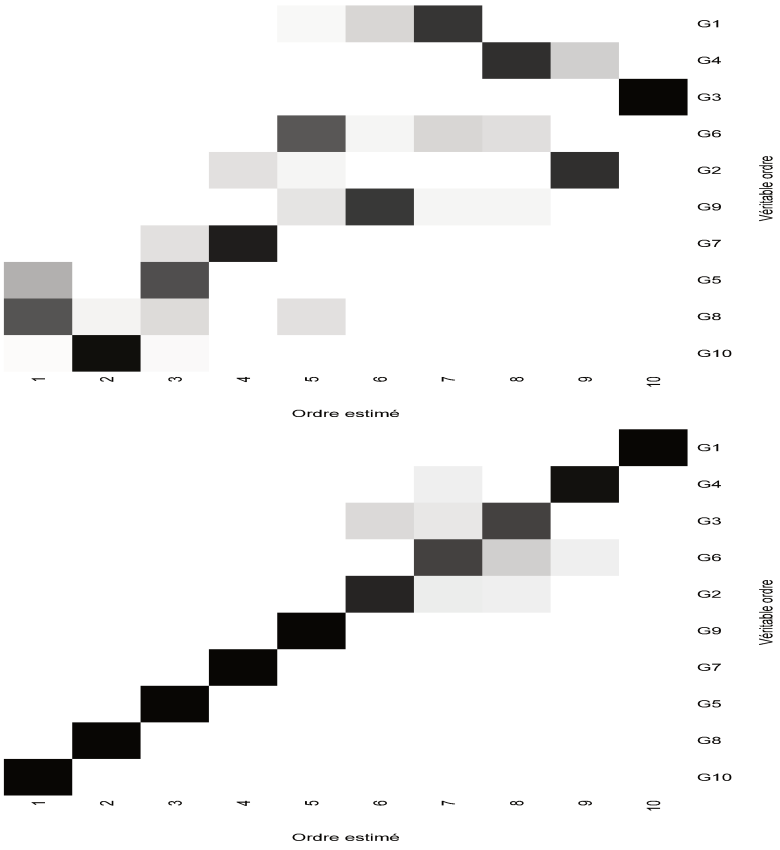


FIGURE 4.8 – Ordre *a posteriori* pour un réseau de 10 noeuds. En bas le squelette est utilisé, en haut il ne l'est pas.

4.5.3 Approximation DAG

Nous avons supposé que le modèle sous-jacent était un DAG durant toutes nos simulations. L'objet de cette sous-section est de vérifier à quel point cette approximation peut fausser les résultats. Nous construisons un modèle gaussien cyclique de la façon suivante : nous prenons les équations récursives précédentes, on suppose que l'on atteint un point fixe, et on inverse le système obtenu. Nous travaillons sur le graphe de la FIGURE 4.9. Nous avons notament inclus dans ce graphe le cycle $v2 \to v6 \to v5 \to v2$. C'est dans un soucis de simplicité que nous travaillons sur un petit modèle de 6 noeuds, sans le squelette et avec assez de données pour se passer de la régulation ridge (FIGURE 4.10). Une fois ce modèle construit, nous simulons des données via ce modèle et nous tentons d'appliquer notre procédure MCMC avec 30.000 itérations. Nous appliquons un seuillage au résultat obtenu afin d'obtenir des graphes creux et proches du résultat escompté.

On note (N1,...,N4) les quatre premiers graphes en terme de fréquence dans notre résultat, et $\bar{N}$ le graphe moyen. Nous faisons tout d'abord une description topologique. On considère dans le tableau suivant la présence ou non d'une arête, ainsi que le nombre

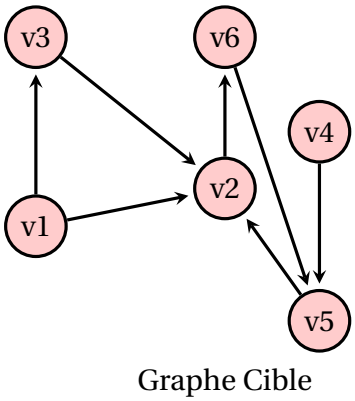


FIGURE 4.9 – Graphe sur lequel se base les simulations de cette partie

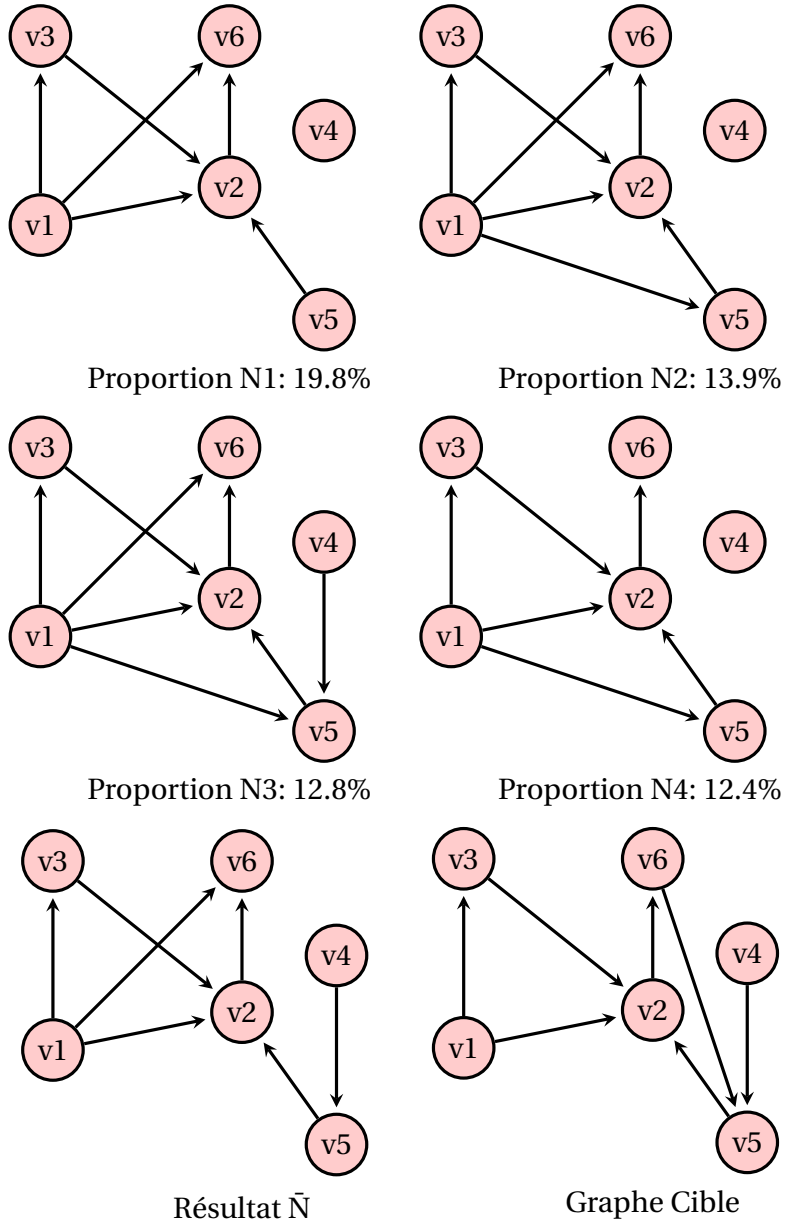


FIGURE 4.10 – Principaux graphes obtenus par la méthode. Nous notons la proportion obtenue dans l'échantillon final des principaux graphes. La référence est donnée par le dernier graphe. Le graphe "Résultat" est le graphe moyen, puis seuillé.

	G1	G2	G3	G4	G5	G6
G1	0	10	10	0	0	0
G2	0	0	0	0	0	10
G3	0	10	0	0	0	0
G4	0	0	0	0	10	0
G5	0	10	0	0	0	0
G6	0	0	0	0	10	0

	G1	G2	G3	G4	G5	G6
G1	0	11.88	9.13	-0.68	0.15	-1.53
G2	0	0	0	0.01	0	10.00
G3	0	11.63	0	0.08	-0.74	0.30
G4	-0.02	-0.56	0.46	0	6.47	-0.41
G5	0	11.50	-0.01	0.01	0	0.10
G6	0	0	0.00	0	0	0

TABLEAU 4.4 – Comparaison des effets causaux directs du modèle en haut, et en sortie de l'algorithme en bas.

de faux-positifs.

	N1	N2	N3	N4	$\bar{\text{N}}$
$v1 \to v2$	✓	✓	✓	✓	✓
$v1 \to v3$	✓	✓	✓	✓	✓
$v3 \to v2$	✓	✓	✓	✓	✓
$v4 \to v5$	×	×	✓	×	✓
$v2 \to v6$	✓	✓	✓	✓	✓
$v5 \to v2$	✓	✓	✓	✓	✓
$v6 \to v5$	×	×	×	×	×
Faux positifs	1	2	2	1	1

Nous voyons que globalement la structure est retrouvée. Logiquement, le cycle est perdu pour chacun des échantillons, et un effet de compensation apparaît via les arêtes $v1 \to v5$ et $v1 \to v6$. Bien que les composantes cycliques du graphe ne soient pas retrouvées, nous retrouvons tout de même les arêtes non présentes dans un cycle.

De la même façon, les effets causaux sont correctement estimés, mis à part ceux correspondant aux arêtes présentes dans le cycle.

4.6 Conclusion

Nous avons présenté ici une méthode permettant de décrire les liens causaux dans un réseau bayésien. Elle a l'avantage d'intégrer les données d'intervention et de donner des résultats fiables. Avec la pénalisation ridge, elle ne nécessite que quelques réplicats, et sous réserve d'avoir à disposition un squelette, elle est applicable à des réseaux de plus grandes dimensions, le facteur limitant étant pour l'instant la résolution d'un système linéaire de l'ordre du nombre de noeuds au cube. Nous avons vu pour finir que notre méthode pouvait retrouver un certain nombre d'effets causaux au sein d'un graphe orienté cyclique. Dans le cadre d'un squelette connu, nous nous efforçons aussi de parcourir un espace qui admet des ordres partiels redondants, et nous devrions pouvoir directement parcourir l'espace des DAGs (ou des graphes orientés) plutôt que l'espace des ordres. En-

fin une perspective enthousiasmante est la possibilité de maximiser l'information découverte sur un graphe en proposant des "knock-out" multiples, dans le cadre d'un plan expérimental optimal.

Chapitre 5

Parallel tempering dans les réseaux bayésiens

Ce chapitre est une ébauche d'article non publié, issue d'un travail en cours en collaboration avec Pascal Fieth (université d'Oldenburg, Allemagne). Il s'agit d'une amélioration de l'algorithme MCMC-Ordre dont le principal intérêt est de permettre la sortie des maxima locaux. Cela permet d'explorer un plus grand espace, et favorise la convergence vers un maximum global. Nous avons écrit le modèle ainsi qu'imaginé les simulations ensemble, Pascal Fieth ayant implémenté le parallel tempering en C.

5.1 Introduction

Les réseaux de régulation de gènes sont des outils importants en bio-informatique pour permettre la découverte et la visualisation des relations causales entre les gènes. Ils sont utilisés pour de nombreuses tâches, comme l'analyse des virus ou l'étude des signaux intercellulaires. Plus généralement, ils peuvent être utilisés dès que l'on souhaite comprendre l'implication d'un gène ou d'un groupe de gènes dans une voie métabolique. Les réseaux de régulation de gènes sont surtout utilisés dans le cadre de petits réseaux, car les données expérimentales, nécessaires lorsque l'on souhaite faire une interprétation causale, sont chères et difficiles à obtenir. Les données d'interventions sur les niveaux d'expression d'un nombre conséquent de gènes deviennent de plus en plus disponibles, grâce à la réduction des coûts des puces à ADN, de séquençage et de l'évolution des techniques associées.

Il existe de nombreux algorithmes d'inférence basés sur des données d'observations seules. Spirtes et Glymour furent des précurseurs en proposant d'inférer un réseau en utilisant des indépendances conditionnelles [99]. Dans le cas gaussien, des corrélations partielles peuvent être testées successivement pour trouver ces indépendances conditionnelles. [101] a proposé d'inférer ces réseaux en deux étapes : d'abord par l'inférence d'un squelette, grâce aux corrélations partielles, puis en estimant les orientations grâce à un algorithme d'optimisation basé sur un score. L'utilisation des algorithmes de Monte Carlo est aussi un piste active [45]. D'autres approches utilisent l'information conditionnelle [21], ou encore des algorithmes gloutons [15]. Plutôt que d'inférer un réseau, la recherche des effets des différents gènes entre eux, et en particulier des effets causaux, est aussi une piste de recherche. Dans ce contexte, [61] utilise seulement des données d'observation pour estimer des bornes sur les potentiels effets causaux sur les gènes. Bien d'autres algorithmes et d'outils en ligne existent déjà pour de tels jeux de données [58].

Il existe cependant un grand intérêt pour l'inférence de réseaux de tailles conséquentes à partir de données expérimentales, comme l'atteste le grand nombre de participants au challenge DREAM [66]. Plusieurs algorithmes ont été proposés pour de tels algorithmes. [44] a modifié un algorithme glouton préexistant afin de pouvoir l'utiliser sur des données d'interventions. [68] a proposé d'utiliser ces données expérimentales pour trouver et valider les liens causaux. Enfin, [76], les gagnants du challenge DREAM 4, ont proposé de travailler directement sur ces nombreuses interactions entre les gènes, en utilisant les données interventionnelles. Il faut cependant pour cela pouvoir effectuer des interventions sur chaque gène, ce qui en pratique n'est rarement le cas.

Une difficulté qui apparait pour ces différentes méthodes porte sur la combinaison des données d'observation et d'intervention. Et en particulier lorsqu'il existe une abondance de données expérimentales avec plus d'une intervention par expérience, même si, en moyenne [89] a montré qu'il fallait un *knock-out* par gène pour pouvoir retrouver l'intégralité de la structure. [77] a introduit une méthode qui, comme celle de [31], est basée sur les réseaux bayésiens gaussiens causaux. Un ordre peut être alors utilisé sur les variables, et un estimateur du maximum de vraisemblance peut être associé à chaque ordre en fonction des interventions utilisées pour générer les données Nuel et al. [71]. En implémentant un algorithme de Métropolis Hastings pour explorer l'espace des ordres, on peut alors chercher le maximum de vraisemblance. Cet algorithme a prouvé son utilité pour des petits réseaux, de l'ordre de la dizaine de gènes seulement.

Les approches utilisant les réseaux bayésiens gaussiens causaux et les ordres causaux introduisent des simplifications. La plus importante est probablement la nécessité d'utiliser un graphe orienté acyclique, prévenant de ce fait toute boucle de rétroaction. En

fonction des données utilisées, les boucles de rétroaction peuvent pourtant jouer un rôle important. Les données de puce nous donnant des données en régime permanent, on fera l'hypothèse que nous regardons ici un régime à l'équilibre sans cycles.

Le but de cet article est de montrer que l'introduction d'un *parallel tempering* permet aux simulations MCMC d'étendre ce travail sur les ordres causaux à des réseaux de l'ordre de 100 gènes. Nous montrerons que la vraisemblance associée à ces ordres possède des maxima locaux. Les simulations effectuées démontrent qu'il est difficile de sortir de ces maxima. Le *parallel tempering* aide à trouver les ordres avec la plus haute valeur de maximum de vraisemblance, tout en permettant d'explorer les ordres alternatifs possédant aussi de grandes valeurs de vraisemblance. Dans le cas où seules quelques interventions sont disponibles, cela permet d'explorer un plus grand nombre d'ordres. Il s'agit d'une amélioration importante, puisque l'ordre disposant de la plus haute valeur de score n'est pas garanti de représenter le réseau sous-jacent. En effet, les simplifications introduites dans le modèle ainsi que dans de nombreux cas la faible quantité de données ne permet que de trouver une collection d'ordres. En conséquence, les chances d'obtenir un résultat crédible ou même le véritable réseau sous-jacent sont ainsi augmentées grâce à cette innovation.

Méthode

Pour inférer les réseaux de régulation de gènes à partir de données d'expression, nous utilisons les réseaux bayésiens gaussiens causaux, ou GBN pour *Gaussian Bayesian Networks*. L'utilisation des GBNs pour l'inférence de réseaux a été introduite par Friedman [32]. Un GBN est un graphe $G = (V, E)$ avec un ensemble de p noeuds (ou sommets) V et arêtes $E \subset (V \times V)$. En utilisant un graphe orienté acyclique, nous pouvons introduire pour un noeud X_i un ensemble de parents $\mathrm{pa}(X_i)$. En résumé, en suivant les travaux de [77] et [71], nous introduisons un ordre causal pour les noeuds. Pour une matrice W, contenant les valeurs des arêtes de G, on classe les noeuds de façon à suivre l'ordre parental, c'est-à-dire $X_j \notin \mathrm{pa}(X_i)$ si $i < j$.

5.1.1 Estimateur du Maximum de Vraisemblance

On considère un ensemble de variables aléatoires gaussiennes $X_{\mathscr{I}}$ avec $\mathscr{I} = \{1, \ldots, p\}$:

$$X_j = m_j + \sum_{i<j} w_{i,j} X_i + \epsilon_j \text{ avec } \epsilon_j \sim \mathscr{N}(0, \sigma^2).$$

Ou en écriture matricielle :

$$\mathbf{X} = \mathbf{m} + \mathbf{X}\mathbf{W} + \epsilon.$$

La log-vraisemblance du modèle utilisant N observations est :

$$\ell(\boldsymbol{m}, \mathbf{W}, \boldsymbol{\sigma^2}) = K - N \sum_j \log(\sigma_j) - \frac{1}{2} \sum_k \sum_j \frac{1}{\sigma_2^2} (x_j^k - \boldsymbol{x}^k \mathbf{W} \boldsymbol{e}_j^T - m_j)^2.$$

Soit N expériences pour lesquelles nous avons les données $x^k = x_1^k, \ldots, x_p^k$, $1 \le k \le N$ avec des interventions sur le sous-ensemble $\mathscr{J}_k \subset \mathscr{I} = \{1, \ldots, p\}$. Nous utilisons les notations suivantes : $\mathscr{K}_j = \{k, j \notin \mathscr{J}_k\}$ and $N_j = |\mathscr{K}_j|$. La log-vraisemblance du modèle pour un mélange d'observations et d'interventions données peut être écrite comme :

$$\ell(\boldsymbol{m}, \mathbf{W}, \boldsymbol{\sigma^2}) = K - \sum_j N_j \log(\sigma_j) - \frac{1}{2} \sum_k \sum_{j \notin \mathscr{J}_k} \frac{1}{\sigma_2^2} (x_j^k - \boldsymbol{x}^k \mathbf{W} \boldsymbol{e}_j^T - m_j)^2.$$

Nous pouvons ainsi trouver analytiquement $\mathbf{m}$, $\mathbf{W}$, et $\boldsymbol{\sigma}$ qui maximisent la vraisemblance $\ell(\mathbf{m}, \mathbf{W}, \boldsymbol{\sigma})$.

5.1.2 Algorithme Métropolis Hastings

Bien que le calcul du maximum de vraisemblance pour un ordre et des données expérimentales données soit relativement facile à obtenir, trouver l'ordre avec la plus grande vraisemblance ne l'est pas. L'ordre dit optimal, c'est-à-dire prédisant le mieux les données est atteint au maximum de vraisemblance. Pour un nombre conséquent de données, et en supposant que le système sous-jacent soit acyclique, cet ordre optimal représente ce système. Cependant, nous devons le trouver dans un ensemble de $p!$ ordres pour un réseau constitué de p noeuds. Calculer la vraisemblance pour chacun de ces ordres n'est en conséquence pas une possibilité, même pour des réseaux relativement petits.

L'algorithme de Métropolis-Hastings Hastings [42] est un algorithme MCMC (pour *Markov Chain Monte Carlo*) dans lequel des chaînes d'états diffèrent sont générées. Il est souvent utilisé dans des problèmes dans lesquels échantillonner d'une distribution est difficile. Ici, nous essayons d'échantillonner l'espace des ordres. Dans le cas d'ordres causaux, nous commençons avec quelques ordres aléatoires $\mathbf{o}$ et on génère à chaque étape du Monte Carlo un nouvel ordre qui est accepté avec probabilité :

$$\mathrm{P}(\mathbf{o}_{t+1} = \mathbf{o} | \mathbf{o}_t) = \min \left\{ \frac{\pi(\mathbf{o}, \mathrm{T}) \mathrm{Q}(\mathbf{o}_t, \mathbf{o})}{\pi(\mathbf{o}_t, \mathrm{T}) \mathrm{Q}(\mathbf{o}, \mathbf{o}_t)}, 1 \right\}. \tag{5.1}$$

La fonction $\pi(\mathbf{o}, \mathrm{T}) = \exp(\ell(\mathbf{o})/\mathrm{T})$ introduit une température T utilisée comme paramètre d'échelle, où $\ell(\mathbf{o})$ correspond à la vraisemblance maximisé pour un ordre donnée. Les hautes températures facilitent l'acceptation des mouvements des petites vraisemblances. Pour la distribution Q, le modèle de Mallows est utilisé Mallows [65]. La densité s'écrit comme :

$$\mathrm{P}(\mathbf{o}) \propto 1 \tag{5.2}$$

$$\mathrm{P}(\mathbf{o}|\mathbf{data}) \propto \mathrm{P}(\mathbf{data}|\mathbf{o}) \mathrm{P}(\mathbf{o}) \tag{5.3}$$

On introduit ici une autre température utilisée par la loi de proposition, Φ, qui peut être ajustée pour optimiser le taux d'acceptation du Metropolis Hastings. Z est une constante de normalisation, $d(;)$ est une mesure de dissimilarité sur l'ordre basée sur le désaccord pour chaque paire dans la séquence. On peut échantillonner Q en utilisant le modèle d'insertion répété (RIM pour *Repeated Insertion Model*) Doignon et al. [23]. En utilisant les notations précédentes, la distribution de cette loi s'écrit :

$$\mathbb{P}(\boldsymbol{o}; \phi, \boldsymbol{r}) = \phi^{d(\boldsymbol{o}, \boldsymbol{r})}.$$

Algorithme 11 : Repeated Insertion Model

Input : $\mathbf{o}_{\texttt{ref}}, \beta$
Output : $\mathbf{o}_{\texttt{new}}$
Soit p la dimension de $\mathbf{o}_{\texttt{ref}}$;
Soit $\phi = \exp^{-1/\beta}$;
Soit $\pi_1 = 1$ la permutation de l'ancien ordre;
for $i = 2$ *à* p **do**
    ;
    if $\phi < 1$ **then**
        Soit $P = (1-\phi) \times \frac{\phi^{(i-(1:i))}}{(1-\phi^i)}$;
        Choisir j dans $[1, i]$ avec probabilité P ;
    else
        Choisir j dans $[1, i]$ uniformément ;
    Construire π_i de longueur i comme π_{i-1}, ;
    et où i est inséré à la place de j;
$\mathbf{o}_{\texttt{new}} = \mathbf{o}_{\texttt{ref}}[\pi_p]$;

5.1.3 *Parallel tempering*

L'algorithme Metropolis Hastings peut rapidement se retrouver coincé dans un maximum local, dû au caractère hautement multi-dimensionnel du problème. Pour de faibles températures, le système accepte plutôt les ordres augmentant la température. A un ordre donné, si tous les ordres voisins ont une faible vraisemblance, il est difficile d'atteindre des états avec une meilleure vraisemblance dans un temps raisonnable. Pour des températures élevées, il est possible de partir de ces maxima locaux, mais alors l'algorithme se comporte comme une simple marche aléatoire, où la probabilité de trouver un ordre avec une forte vraisemblance est extrêmement faible.

Une approche commune pour de tels problèmes est l'application du *parallel tempering*. Les avantages des basses et hautes températures sont alors utilisés conjointement en simulant de multiples chaînes en parallèle, à N_T températures différentes T_i. Les températures sont ordonnées, de telle sorte que $T_i < T_{i+1}, \forall T_i$. Pour chaque système l'algorithme Metropolis Hastings est utilisé avec une température spécifique T_i pour un certain nombre d'itérations (typiquement un balayage, c'est-à-dire p itérations pour les réseaux de p noeuds). Puis N_t échanges aléatoires sont tentés pour des chaînes adjacentes. Deux chaînes sont échangées, c'est-à-dire qu'elles sont simulées à partir de l'échange avec la température de sa partenaire, avec probabilité :

$$P_{sw} = \min\left\{ e^{\left(\ell_i - \ell_j\right)\Delta}, 1 \right\}, \tag{5.4}$$

avec $\Delta = T_i^{-1} - T_j^{-1}$ et ℓ_i la vraisemblance de chaque système.

Les conditions comme la balance détaillée sont remplies et permettent l'échantillonnage durant les simulations. Nous utilisons les chaînes à haute température pour être capable de partir des maxima locaux et "sauter" les barrières de basses vraisemblances en "chauffant et refroidissant" les différents systèmes. Les températures sont choisies manuellement dans les tests, de telle sorte que les tentatives d'échanges entre des chaînes voisines se réalisent une fois sur deux. En pratique, cela signifie que le taux d'échange réussi est entre 0.3 et 0.7.

5.2 Résultats

Pour évaluer la performance de l'algorithme de *parallel tempering* par rapport au simple Metropolis Hastings, nous avons travaillé sur un réseau artificiel. Bien que les réseaux générés ne rendent pas compte de la réalité des mécanismes en génomique, ils permettent d'étudier plus objectivement la réussite d'un algorithme. L'utilisation d'un grand nombre d'expériences rend probable l'apparition de ce réseau artificiel parmi ceux qui ont une vraisemblance élevée.

5.2.1 Evolution de la vraisemblance

Les réseaux artificiels ont été générés en utilisant un modèle de réseau bayésien gaussien. Le modèle est centré et les différents paramètres sont choisis aléatoirement. Les interactions sont fixées à $w_{i,j} = 0$ pour $i \leq j$ en suivant l'ordre original et simulés uniformément dans $(-1,-0.25) \cup (0.25,1)$ pour les autres interactions. Le nombre d'expériences E varie d'un jeu de données à l'autre.

Comme stratégie naïve, nous avons décidé d'effectuer les interventions aléatoirement. Pour chaque expérience, et chaque nœud, une intervention est simulée par une loi de Bernouilli de probabilité $p_{int} = (2\mathrm{N})^{-1}$, où N est ici fixé à 10. Pour chaque nombre d'expériences E deux configurations différentes ont été tentées, l'une avec E = N et l'autre avec E = 2N expériences. En conséquence, une configuration a, en moyenne, N interventions, ce qui en théorie est sur la transition de phase. La deuxième configuration n'a plus que N/2 interventions en moyenne, rendant la probabilité d'inférer le réseau initial très basse.

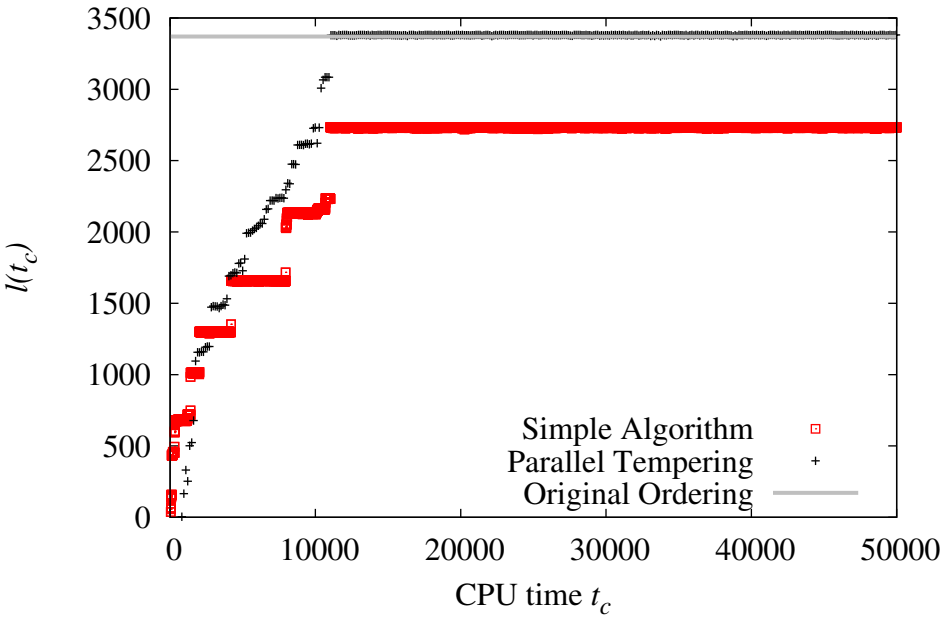


FIGURE 5.1 – Log-vraisemblance avec ou sans *parallel tempering*, en fonction du temps de calcul en secondes. La chaîne affichée pour le *parallel tempering* est calibrée à la température 1, selon la première configuration cité dans le texte.

Une façon d'étudier l'amélioration que procure le *parallel tempering* est de regarder l'évolution de la valeur de la vraisemblance. Dans la figure 5.1, on voit apparaître clairement l'amélioration. Alors que l'algorithme Métropolis Hastings simple reste bloqué dans un maximum local et ne retrouve jamais l'ordre optimal, avec la procédure parallèle on

converge vers celui-ci. De plus, on voit apparaître des transitions plus douces, ce qui traduit le passage à des configurations plus ou moins équivalentes mais éventuellement loin en terme de distance, là où pour la procédure simple les transitions sont très abruptes.

5.2.2 Distribution a posteriori

L'évaluation sur le jeu de données de référence DREAM est aussi une bonne façon de s'assurer de la fiabilité de notre algorithme. Il s'agit d'une base de données issue de simulations complexes, essayant le plus possible de se rapprocher de données réelles. En conséquences, l'hypothèse d'acyclicité est violée. Il s'agit néanmoins d'une bonne base pour évaluer notre algorithme. Nous avons ainsi comparé la distribution a posteriori retrouvée en lançant MCMC simple et MCMC PT (pour MCMC parallel tempering) sur ces données. Pour quatres graphes sur cinq, la distribution a posteriori est systématiquement constante. En revanche, le résultat de MCMC simple change drastiquement en fonction du point d'initialisation, relevant le fait que nous n'arrivons pas à sortir des différents maxima locaux en rapport avec les points d'initialisation, comme le montre la figure 5.2.

5.2.3 Entropie

Pour confirmer ce résultat, nous avons calculé l'entropie de chaque chaîne dans les deux cas. Cette entropie est calculé à partir de la loi a posteriori, qui donne une probabilité d'être à un rang i pour chaque noeuds j, que l'on peut noter $p_{i,j}$. Celle-ci est définie par la formule suivante.

$$\text{Entropie} = -\sum_{i,j} p_{i,j}\, log\left(p_{i,j}\right)$$

Après une moyenne sur l'ensemble des 130 initialisations par jeu de données, il en résulte la table suivante. Il en ressort que l'entropie est bien plus grande, en moyenne, avec

#	Simple	PT
1	4.60204	7.83782
2	4.26482	10.163
3	4.74911	8.93702
4	4.14197	7.86007
5	5.19716	8.50191

TABLEAU 5.1 – Entropie moyenne sur 130 initialisations différentes, pour chaque jeu de données, pour l'algorithme MCMC simple et MCMC PT.

l'algorithme parallel tempering qu'avec l'algorithme simple. Ceci signifie qu'en moyenne, MCMC PT explore bien plus d'états que sa version sans *parallel tempering*. Cela confirme l'interprétation précédente, à savoir qu'avec notre amélioration, on explore bien plus largement l'espace et on augmente drastiquement nos chances de trouver un maximum global.

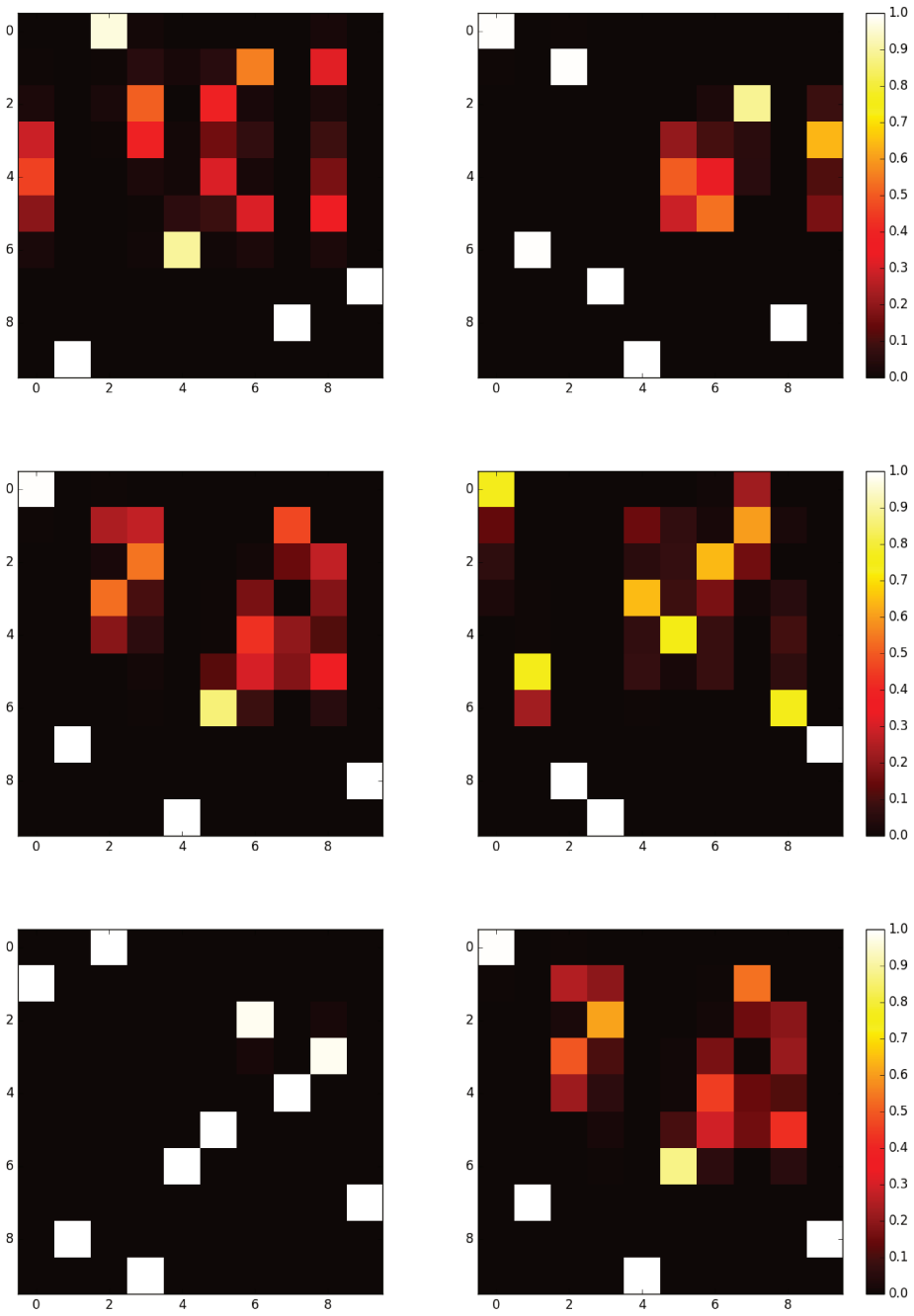


FIGURE 5.2 – Distribution a posteriori retrouvée par MCMC parallel tempering, en haut à gauche, et MCMC simple pour diverse initialisations pour le reste. La configuration MCMC PT est stable peut importe le point initial, tandis que pour MCMC simple la configuration est fonction de l'initialisation.

5.3 Conclusion

L'inférence de réseaux est un problème hautement multi-dimensionnel, ce qui justifie d'utiliser une procédure détournée comme le calcul des ordres. Même sous des hypothèses gaussiennes, linéaires et d'acyclicité, la vraisemblance d'un tel modèle est aussi multi-modale. Nous avons vu ici comment répondre à ce problème, en ajustant notre algorithme avec le *parallel tempering*. Cela permet d'explorer plus largement l'espace des différents ordres possibles, et en corollaire des différents réseaux possibles. De plus, cela exploite particulièrement bien l'architecture des processeurs actuels, construits pour les processus parallèles. De ce fait, cela améliore grandement la qualité de l'inférence effectuée.

Chapitre 6

Approximation de Laplace

6.1 Introduction

On présente ici un nouvel algorithme d'estimations pour les réseaux bayésiens gaussiens, dans le cas où nous disposons de données d'observations et d'interventions. Cette méthode à été développé à la suite du travail de [77], qui proposait un procédé d'inférence basé sur les ordres topologiques plutôt que sur les structures. L'algorithme en lui-même supposait de fixer préalablement une structure, généralement pleine, qui ne bougeait pas tout le long de l'inférence : seul l'ordre entre les différentes variables pouvait évoluer. Cela pose évidemment un problème lorsque l'on cherche à quantifier les interactions, *a fortiori* causales. Pour répondre à ce problème nous avons décidé de suivre une autre voie, déjà empruntée dans la littérature, celle de l'inférence de la structure du DAG. C'est un problème ambitieux de sélection de modèles, avec la particularité de composer avec un espace très complexe.

Après un bref rappel de l'état de l'art, et une illustration de l'intérêt de l'inférence de la structure du DAG, nous introduirons notre innovation. Il s'agit de la dérivation et de l'implémentation d'un score issu d'une approximation de Laplace. Celle-ci ne nécessite qu'une statistique exhaustive qui peut être calculée simplement. De plus, notre paramétrisation nous permet d'être dans le cadre d'une optimisation par blocs, ce qui rend le calcul d'un maximum extrêmement rapide. Nous utiliserons ensuite ce score dans le cadre d'un algorithme de chaîne de Markov, dit MC3, pour inférer ces structures.

En préliminaire on indiquera que lorsqu'on effectue une inférence d'un DAG par MCMC, les résultats empiriques semblent montrer que le mélange se passe bien mieux dans l'espace des ordres que dans l'espace des structures [31]. Cependant cela se paie en temps de calcul, qui est malheureusement la donnée essentielle, puisque dans l'immense majorité des cas les jeux de données se rapportent à des réseaux de tailles conséquentes. De nombreuses tentatives d'alternatives ou d'améliorations de l'algorithme MC3 ont été proposées pour améliorer ce problème de mélange. Parmi elles, citons notamment la proposition de [40] qui consiste à rajouter un mouvement dans l'algorithme MC3, et celle de [37] qui propose de faire de l'échantillonnage de Gibbs sur les parents des noeuds plutôt que de travailler sur les arêtes.

Nous avons ici une approche bayésienne. Nous allons fixer un *a priori* impropre uniforme sur les différents paramètres, pour lequel nous dériverons ensuite notre score. La littérature propose plusieurs lois a priori, généralement choisies du fait de leurs caractères conjugués [34; 39; 50; 84]. L'approximation de Laplace a déjà été proposée pour les modèles graphiques gaussiens par Banerjee and Ghosal [5]. De manière générale, ces différents algorithmes utilisent tous une version plus ou moins amendée de l'algorithme MC3, que nous décrirons un peu plus loin. On peut noter la spécificité de l'algorithme proposé par [8], dénommé SSS pour *Stochastic Short-gun Search*. Celui-ci consiste à sélectionner au hasard N graphes éloignés d'une arête du graphe courant, puis de tirer aléatoirement parmis ces graphes en fonction de leurs scores, ou plus exactement en fonction de leurs probabilités *a posteriori*.

Nous allons présenter maintenant en détail notre modèle, la dérivation de notre approximation de Laplace puis nous commenterons quelques résultats.

6.2 Méthode

Dans un premier temps je vais illustrer un lien entre les modèles d'équations structurelles linéaires et la régression linéaire, et je vais l'utiliser pour expliquer en quoi la structure d'un DAG est importante. Ensuite, je développerai en détail le calcul de la vraisemblance, de notre formulation bayésienne et enfin de l'algorithme MC3 utilisé.

6.2.1 Équations structurelles linéaires et régressions linéaires

Soit $X_1,\ldots,X_n$ un vecteur aléatoire engendré par le jeu d'équations structurelles linéaires suivant, où ϵ est gaussien :

$$\begin{array}{rcl} X_1 & = & \sum_{i\in \mathrm{pa}(X_1)} \alpha_{i,1} X_i + \epsilon_1 \\ , & \vdots & \\ X_n & = & \sum_{i\in \mathrm{pa}(X_n)} \alpha_{i,n} X_i + \epsilon_n. \end{array} \tag{6.1}$$

On peut retrouver les différents coefficients d'effets causaux directs $\alpha_{i,j}$ à partir d'une régression linéaire. En effet, lorsque les variables explicatives sont exactement les parents de la variable expliquée, les coefficients de régression correspondent à ces effets causaux. Lorsque ce n'est pas le cas, il n'y a plus cette correspondance.

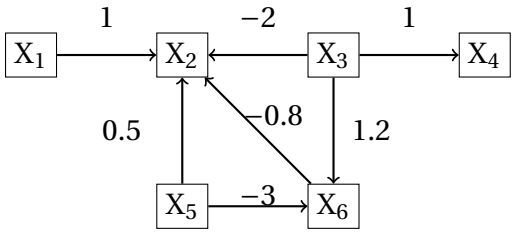


FIGURE 6.1 – La régression de la variable X_6 par les autres covariables ne donne pas le coefficient causal attendu.

Exemple 8 *Une régression de la variable* X_6 *dans le modèle présenté figure 6.1 illustre ce problème d'ajustement des covariables. Ici, la moyenne résiduelle est nulle et on se place dans un modèle homoscédastique de variance 0,1. En effet, lorsqu'on prend les variables* X_3 *et* X_5 *comme variables explicatives, on retrouve bien les coefficients d'effets causaux.*

```
## Call:
## lm(formula = V6 ~ 0 + V1 + V3 + V5, data = data)
##
## Coefficients:
## V1 V3 V5
## -0.01578 1.20735 -3.00789
```

Par contre, lorsqu'on introduit la variable X_2*, on s'éloigne sensiblement d'une estimation de l'effet causal.*

```
## Call:
## lm(formula = V6 ~ 0 + V2 + V3 + V5, data = data)
##
## Coefficients:
##  V2  V3  V5
## -0.3007 0.3327 -2.1512
```

Le même problème peut bien sûr apparaitre dans le cas où on n'inclut pas assez de covariables. Le choix des différentes covariables est donc à faire judicieusement.

En terme d'estimation d'effets causaux, le modèle plein peut donc s'éloigner de manière arbitraire de la réalité du processus génératif. Cette différence est quantifiable : elle correspond à la corrélation partielle entre la variable explicative et la variable expliquée si on enlevait le lien direct qui les lie. C'est pourquoi il est nécessaire de passer dans le monde de la sélection de modèle.

6.2.2 Vraisemblance

Pour chaque modèle de la forme 6.1 il est possible de le réécrire sous forme matricielle. Soit $\mathcal{J} = \{1, ..., p\}$, on peut définir $\mathbf{W} = (w_{i,j})_{i,j} = (\alpha_{i,j})_{i,j}$ où les éléments $\alpha_{i,j}$ sont nuls s'ils ne sont pas présents dans 6.1. Cette réécriture rappelle le lien avec la représentation graphique, où $\mathbf{W}$ est la matrice d'adjacence du graphe orienté associée, là où les α sont des coefficients de régression. On obtient alors l'équation suivante, pour un modèle non centré :

$$\mathbf{X} = \boldsymbol{m} + \mathbf{X}\mathbf{W} + \boldsymbol{\varepsilon} \quad \text{avec} \quad \boldsymbol{\varepsilon} \sim \mathcal{N}(0, \mathbf{I} \times \boldsymbol{\sigma}^{\mathbf{2}}).$$

Il est alors aisé de montrer que notre modèle est équivalent à $\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, où :

$$\boldsymbol{\mu} = m\mathbf{L} \quad \text{et} \quad \boldsymbol{\Sigma} = \mathbf{L}^{\mathrm{T}} \mathrm{diag}(\sigma^2)\mathbf{L} = \sum_j \sigma_j^2 \mathbf{L}^{\mathrm{T}} e_j^{\mathrm{T}} e_j \mathbf{L},$$

où e_j est un vecteur ligne nul sauf pour la j^{ieme} composante, égale à 1, et où $\mathbf{L} = (\mathbf{I} - \mathbf{W})^{-1} = \mathbf{I} + \mathbf{W} + \ldots + \mathbf{W}^{p-1}$ avec $\mathbf{W} = (w_{i,j})_{i,j \in \mathcal{J}}$. On peut noter que la nilpotence de $\mathbf{W}$ est garantie par l'acyclicité du modèle. En effet dans ce cas, il est possible de permuter $\mathbf{W}$ de telle sorte qu'elle soit triangulaire supérieure.

Soit N réalisations générées sous $x^k = (x_1^k, \ldots, x_p^k)$ $(1 \leqslant k \leqslant \mathrm{N})$ avec des interventions sur $\mathcal{J}_k$ ($\mathcal{J}_k = \emptyset$ signifie dans ce cas pas d'interventions). On note $\mathcal{K}_j = \{k, j \notin \mathcal{J}_k\}$, et $\mathrm{N}_j = |\mathcal{K}_j|$ son cardinal. La log-vraisemblance du modèle peut s'écrire comme :

$$\ell(m, \sigma, w) = -\frac{\log(2\pi)}{2} \sum_j \mathrm{N}_j - \sum_j \mathrm{N}_j \log(\sigma_j) - \frac{1}{2} \sum_j \frac{1}{\sigma_j^2} \sum_{k \in \mathcal{K}_j} (x_j^k - x^k \mathbf{W} e_j^{\mathrm{T}} - m_j)^2.$$

Cela est principalement dû au fait que pour chaque ensemble d'interventions $\mathcal{J}$, on a $\mathbf{W}_{\mathcal{J}} e_j^{\mathrm{T}} = \mathbf{W} e_j^{\mathrm{T}}$ pour tout $j \notin \mathcal{J}$.

En considérant la dérivée par rapport à m_j on a pour tout j tel que $\mathrm{N}_j > 0$:

$$\hat{m}_j = \frac{1}{\mathrm{N}_j} \sum_{k \in \mathcal{K}_j} (x_j^k - x^k \mathbf{W} e_j^{\mathrm{T}}),$$

qui peut être inséré dans la vraisemblance pour avoir :

$$\tilde{\ell}(\sigma, w) = -\frac{\log(2\pi)}{2} \sum_j \mathrm{N}_j - \sum_j \mathrm{N}_j \log(\sigma_j) - \frac{1}{2} \sum_j \frac{1}{\sigma_j^2} \sum_{k \in \mathcal{K}_j} (y_j^{k,j} - y^{k,j} \mathbf{W} e_j^{\mathrm{T}})^2,$$

où (k, j) tel que pour $k \in \mathcal{K}_j$ on ait :

$$y^{k,j} = x^k - \frac{1}{\mathrm{N}_j} \sum_{k' \in \mathcal{K}_j} x^{k'}.$$

Chose intéressante, il suffit pour calculer cette vraisemblance d'une statistique exhaustive, nous évitant ainsi d'avoir à jouer avec l'éventuelle complexité des données. Pour tout $i, i', j \in \{1, \dots, p\}$ on définit la statistique exhaustive suivante :

$$z_{i,i'}^{j} = \sum_{k \in \mathcal{K}_j} y_j^{k,i} y_j^{k,i'}$$

et avec la reparamétrisation $\sigma_j = \exp(s_j)$ on obtient finalement :

$$\ell(s, w) = -\sum_j \left(\frac{\log(2\pi)}{2} \mathrm{N}_j + \mathrm{N}_j s_j + \frac{1}{2} \exp(-2s_j) \left[z_{j,j}^{j} - 2 \sum_{i \in \mathrm{pa}_j} w_{i,j} z_{i,j}^{j} + \sum_{i \in \mathrm{pa}_j} \sum_{i' \in \mathrm{pa}_j} w_{i,j} z_{i,j}^{j} w_{i',j} z_{i',j}^{j} \right] \right)$$

où $(s, w) \in \mathbb{R}^{p+m}$, m est le nombre de termes différents de zéro dans W (c'est-à-dire le nombre d'arêtes dans le DAG).

Chaque bloc $(s_j, w_{\cdot,j})$ peut être maximisé indépendamment en résolvant :

$$\sum_{i' \in \mathrm{pa}_j} w_{i',j} z_{i,i'}^{j} = z_{i,j}^{j} \quad \text{pour tout } i \in \mathrm{pa}_j,$$

et

$$\mathrm{N}_j \exp(2s_j) = z_{j,j}^{j} - 2 \sum_{i \in \mathrm{pa}_j} w_{i,j} z_{i,j}^{j} + \sum_{i \in \mathrm{pa}_j} \sum_{i' \in \mathrm{pa}_j} w_{i,j} z_{i,j}^{j} w_{i',j} z_{i',j}^{j}.$$

La complexité pour résoudre le bloc j est alors $\mathcal{O}(q_j^3)$ où $q_j = |\mathrm{pa}_j|$. On verra alors $\mathbf{A}_j \in \mathbb{R}^{q_j \times q_j}$ comme la matrice symétrique et définie positive du système linéaire en $w_{\cdot,j}$ définie au dessus. En insérant l'estimateur du maximum de vraisemblance, on obtient :

$$\ell(\hat{s}, \hat{w}) = -\sum_j \left(\frac{\log(2\pi)}{2} \mathrm{N}_j + \mathrm{N}_j \hat{s}_j + \frac{\mathrm{N}_j}{2} \right).$$

6.2.3 Approximation de Laplace

Le calcul de l'estimateur du maximum de vraisemblance est une bonne chose, mais il ne peut pas être utilisé tel quel pour faire de la sélection de modèle. On souhaiterait pouvoir discriminer différents graphes **G** et éventuellement choisir celui qui correspond le mieux au modèle. En dénotant θ les différents paramètres en fonction de la structure associée **G**, et en plaçant un a priori sur ceux-ci, on peut écrire la probabilité a posteriori d'un graphe donné :

$$\mathbb{P}(\mathbf{G}|\text{data}) \propto \int \mathbb{P}(\text{data}|\boldsymbol{\theta}, \mathbf{G}) \times \mathbb{P}(\boldsymbol{\theta}|\mathbf{G}) \times \mathbb{P}(\mathbf{G}) d\boldsymbol{\theta}.$$

Le calcul d'une telle quantité est complexe. Il est possible de l'effectuer en choisissant les bonnes quantités pour les *a priori*, par exemple avec des lois conjuguées. C'est un choix qui restreint néanmoins les possibilités. Il est aussi possible de calculer cette intégrale par évaluation numérique, via MCMC. C'est un choix coûteux qui peut fortement ralentir l'inférence. On préférera ici effectuer une approximation dite de Laplace, permettant de choisir si besoin l'*a priori* des différentes quantités. Il s'agit d'une approximation du second ordre de l'intégrale, nous donnant le résultat voulu. On la décrit ici.

On cherche à calculer l'intégrale suivante :

$$\int \mathrm{Q}(\boldsymbol{\theta}) d\theta = \int \mathbb{P}(\text{data}|\boldsymbol{\theta}, \mathbf{G}) \times \mathbb{P}(\boldsymbol{\theta}|\mathbf{G}) d\boldsymbol{\theta}.$$

L'idée est d'utiliser un développement de Taylor du second ordre de logQ pour reconnaitre une densité gaussienne. On effectue donc ce développement autour du maximum $\hat{\boldsymbol{\theta}}$:

$$\log \mathrm{Q}(\boldsymbol{\theta}) \approx \log \mathrm{Q}\left(\hat{\boldsymbol{\theta}}\right) + \nabla \log \mathrm{Q}\left(\hat{\boldsymbol{\theta}}\right)(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}) - \frac{1}{2}(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^{\mathrm{T}} \mathbf{A} (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}),$$

où $\mathbf{A} = -\mathrm{Hess}(\log \mathrm{Q})\left(\hat{\boldsymbol{\theta}}\right)$. Le gradient est nul du fait de l'évaluation au maximum. On a alors :

$$\log \mathrm{Q}(\boldsymbol{\theta}) \approx \log \mathrm{Q}\left(\hat{\boldsymbol{\theta}}\right) - \frac{1}{2}(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^{\mathrm{T}} \mathbf{A} (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}).$$

En revenant à Q on a :

$$\mathrm{Q}(\boldsymbol{\theta}) \approx \mathrm{Q}\left(\hat{\boldsymbol{\theta}}\right) \exp\left(-\frac{1}{2}(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^{\mathrm{T}} \mathbf{A} (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})\right).$$

Nous voyons alors la densité d'une distribution gaussienne multivariée, que l'on sait intégrer :

$$\int \mathrm{Q}(\boldsymbol{\theta}) d\theta \approx \mathbb{P}(\mathrm{data}|\hat{\boldsymbol{\theta}}, \boldsymbol{o}) \mathbb{P}\left(\hat{\boldsymbol{\theta}}|\boldsymbol{o}\right) \sqrt{\frac{(2\pi)^{\mathrm{K}}}{\det(\mathbf{A})}}.$$

On a donc ici approché notre loi a posteriori par une loi gaussienne multivariée, c'est-à-dire : $\mathbb{P}(\mathrm{X}|\boldsymbol{\theta}) \approx \mathcal{N}\left(\hat{\boldsymbol{\theta}}, \mathbf{A}^{-1}\right)$. On peut vérifier que log(det(A)) croît avec $p\log(\mathrm{N})$, nous donnant le critère BIC si l'on dispose d'assez de données [16]. Pour pouvoir calculer cette approximation, il est nécessaire d'obtenir la dérivée seconde du log-prior $\log(\mathbb{P}(\boldsymbol{\theta}|\boldsymbol{o}))$ et le maximum a posteriori $\boldsymbol{\theta}^{\mathbf{MAP}}$ par rapport à ce prior. L'*a priori* uniforme nous donne $\boldsymbol{\theta}^{\mathbf{MAP}} = \boldsymbol{\theta}^{\mathbf{MLE}}$, que nous avons calculé préalablement. En résumé, on obtient l'approximation de Laplace de notre probabilité a posteriori en posant #E le nombre d'arêtes et p le nombre de nœuds :

$$\mathbb{P}(\mathbf{G}|\mathrm{data}) \approx \mathbb{P}(\mathrm{data}|\widehat{\boldsymbol{\theta}}, \mathbf{G}) \mathbb{P}\left(\widehat{\boldsymbol{\theta}}|\mathbf{G}\right) \mathbb{P}(\mathbf{G}) \sqrt{\frac{(2\pi)^{(\#\mathrm{E}+2\times p)}}{\det(\mathbf{A})}}.$$

Ce score peut être utilisé pour discriminer les différents modèles. Nous allons vérifier dans la section suivante la validité de notre score, puis son application au sein d'un algorithme d'inférence pour les DAGs.

6.3 Résultats

Nous allons décrire l'algorithme d'inférence que nous utiliserons pour exploiter le score que l'on a développé dans la section précédente. Il s'agit d'un algorithme de type chaîne de Markov travaillant dans l'espace des structures [63; 64]. L'idée est la suivante : partant d'un graphe donné G_i, on définit $\mathcal{N}_i$ comme l'ensemble des graphes identiques G_i modulo une arête. Cela peut être une arête enlevée ou une arête ajoutée par rapport à G_i. On définit ensuite une loi de proposition π_i sur ces graphes, dont le support repose sur $\mathcal{N}_i$. Concrètement on prendra une loi uniforme sur l'ensemble de ces graphes. On simulera ensuite un nouveau graphe G_{i+1} à partir de π_i. Enfin on acceptera le mouvement avec probabilité :

$$\min\left\{\frac{|\mathcal{N}_i|\,\mathbb{P}(G_{i+1}|\text{data})}{|\mathcal{N}_{i+1}|\,\mathbb{P}(G_i|\text{data})},1\right\}.$$

On obtient, après simulation, des échantillons de la loi $\mathbb{P}(\mathbf{G}|\text{data})$. A partir de ces échantillons on pourra ensuite choisir quels modèles semblent être les plus probables.

6.3.1 Jeu de données de référence DREAM 4

Les données DREAM 4 ont été développées pour évaluer les forces et les faiblesses de l'inférence de réseaux de gènes [66]. Ces jeux de données sont divisés en deux sous-groupes de tailles différentes : l'un comporte 10 gènes et l'autre 100 gènes. De nombreuses informations sont données, comme par exemple des séries temporelles ou des expériences de *knock-out*. Pour tester notre algorithme, nous n'utiliserons que les données en régime permanent. Celles-ci se séparent en deux états : des données sont observationnelles, et d'autres interventionnelles, c'est-à-dire qu'une invalidation génique à été simulée. A chaque fois, une seule donnée observationnelle est fournie, et de même une seule intervention sur chacun des gènes, indépendamment, est fournie. Enfin tout ceci est simulé sous cinq graphes orientés différents, qui ne sont pas nécessairement acycliques puisque le but est d'être au plus près d'un réseau biologique.

6.3.2 Algorithmes concurrents

De nombreux algorithmes sont disponibles pour résoudre le problème de la sélection dans l'espace des DAGs. Cependant, nous avons choisi ici trois concurrents, l'algorithme GIES, un algorithme à base Z-score et MCMC-Order, qui sont tous particulièrement appropriés vis à vis de notre jeu de données de référence. En effet, chacun des algorithmes que nous allons présenter succinctement utilisent les données interventionnelles au sein de leur modèle. Bien sûr, dans le cas de données réelles, nous avons généralement bien plus d'observations, mais aussi bien moins de données expérimentales. Cela peut cependant évoluer dans le futur, et être capable de gérer de nombreuses expériences différentes est un réel enjeu.

A partir d'ici, p est le nombre de nœuds, $\boldsymbol{x}^{wt}$ correspond aux échantillons issus d'une observation et $\boldsymbol{x}^i$ est un échantillon où le nœud i est invalidé.

Le gagnant du challenge DREAM 4 a utilisé un Z-score pour déterminer la présence d'une arête [76]. L'algorithme commence par calculer la moyenne et la variance "interventionnelle" pour chaque nœud :

$$\begin{aligned}\mu_j &= \frac{1}{p}\sum_{i\in\{1,\dots,p\}} x_j^i,\\ \sigma_j &= \frac{1}{p}\sum_{i\in\{1,\dots,p\}}\left(x_j^i-\mu_j\right)^2.\end{aligned}$$

On pourrait noter qu'ici nous n'utilisons pas de réplicat observationnel. On calcule ensuite le Z-score suivant :

$$W_{i,j} = \frac{x_j^i - \mu_j}{\sigma_j}.$$

Une fois ces scores obtenus, la seconde partie de l'algorithme consiste à construire le graphe en deux phases. Un seuil λ doit être choisi, éventuellement en utilisant une procédure de validation croisée. Avec ce seuil, et la matrice de Z-scores, on peut alors construire un premier graphe : seules les arêtes dont la valeur absolue sera au dessus du seuil sont conservées. Cependant, certaines arêtes appelées *feed-forward edges* peuvent être le résultat d'un effet indirect, c'est-à-dire un effet causal à travers un chemin de longueur supérieure à 2. Ainsi pour une arête entre un nœud A et un nœud B, il est nécessaire de vérifier qu'il n'existe pas un chemin indirect entre A et B. Si tel est le cas, on retire l'arête entre les deux nœuds. D'autre part, des cycles peuvent apparaître. Dans ce cas, on considérera que l'ensemble des nœuds du cycle sont fortement connectés et ils se comporteront comme un seul nœud. Après le premier seuillage, on cherchera dans le graphe construit ces composantes fortement connectées et ces *feed-forward edges*, et on élaguera le graphe en accord avec ces éléments.

Une deuxième méthode est basée sur les équations structurelles linéaires. Chaque graphe, ici des DAGs, sont modélisés par les équations suivantes :

$$\forall i \quad x_i = \beta_i + \sum_{\mathrm{pa}_i} \alpha_{j,i} x_j + \epsilon_i,$$

en posant $\epsilon_i \sim \mathcal{N}\left(0, \sigma_i^2\right)$. On peut alors écrire la vraisemblance du modèle et en calculer l'estimateur du maximum de vraisemblance par rapport à ce modèle. Les interventions sont aussi incluses dans la vraisemblance. Pour comparer des modèles différents, une pénalité BIC est ajoutée, donnant le score suivant :

$$\mathrm{Score} = \ell - \frac{1}{2} n \#\mathrm{E},$$

où n est le nombre d'échantillons, #E est le nombre d'arêtes et ℓ la log-vraisemblance. Hauser and Bühlmann [43] proposent d'utiliser ce score au sein d'un algorithme glouton. Le principe est tiré de l'algorithme GES : il consiste à itérer deux phases, l'une expansive consistant à ajouter successivement l'arête maximisant le score, puis une phase d'élagage avec le même principe mais cette fois-ci en retirant une arête. Lorsque le score ne bouge plus, l'algorithme aura convergé. De nombreuses heuristiques sont proposées pour améliorer la rapidité de convergence de l'algorithme.

Enfin, la dernière méthode, plutôt que d'essayer de trouver directement un DAG, consiste à essayer de trouver l'ordre sur les noeuds. Il ne s'agit pas exactement d'une méthode de sélection de modèle. Cependant à partir des ordres inférés, il est possible de seuiller la matrice d'adjacence afin de trouver un graphe parcimonieux. En supposant un graphe plein, on peut écrire la vraisemblance pour un ordre donné. Puis, en utilisant une stratégie du type *maximum a posteriori*, on peut écrire une approximation de la probabilité a posteriori d'un ordre donné :

$$\mathbb{P}(\boldsymbol{o}|\mathrm{data}) \propto \mathbb{P}\left(\mathrm{data}|\boldsymbol{o}, \hat{\theta}\right) \mathbb{P}(\boldsymbol{o}).$$

On peut ensuite tirer des échantillons à partir de cette loi à traver un algorithme de Metropolis-Hastings. En utilisant ces échantillons, une matrice d'adjacence empirique est dérivée, qui peut être seuillée pour obtenir un seul graphe [77].

6.3.3 Qualité de l'approximation de Laplace

Des simulations ont été effectuées sur un graphe de 10 nœuds et possédant 13 arêtes. On a généré ainsi 20 observations et 25 interventions sur différents nœuds. Puis on a calculé l'approximation de Laplace, pour laquelle on montre graphiquement le caractère gaussien de notre loi a posteriori pour divers paramètres, comme montré dans la figure 6.2.

Dans un second temps, on a calculé la valeur du score pénalisé pour quelques graphes, ainsi que pour notre graphe de référence. Cela montre que l'on pénalise bien les graphes dans le sens que l'on souhaite. Enfin, on a refait cette expérience, mais à partir d'un graphe de 7 nœuds, et on a ensuite calculé le score de l'ensemble des graphes possibles. On retrouve bien le maximum attendu. Le résultat est montré figure 6.4.

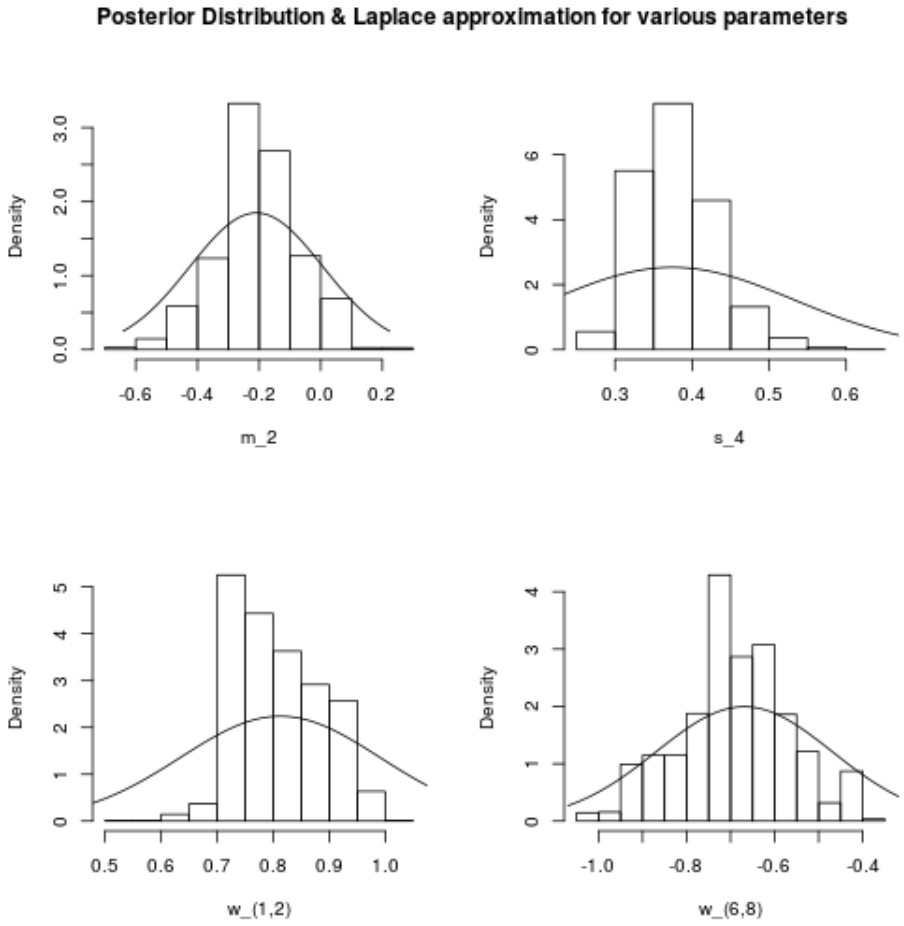


FIGURE 6.2 – Superposition des échantillons tirés de l'approximation de Laplace avec la densité d'une loi gaussienne.

La rapidité de calcul en utilisant notre algorithme MC3 a aussi été étudiée figure 6.5. On regarde qu'on reste bien plus rapide que la méthode MCMC-Ordre, mais que l'on perd cet avantage comparé aux méthodes à scores gloutonnes. Cela s'explique principalement du fait de l'utilisation d'une procédure MCMC, nous donnant des échantillons d'une loi plutôt qu'un seul graphe en sortie.

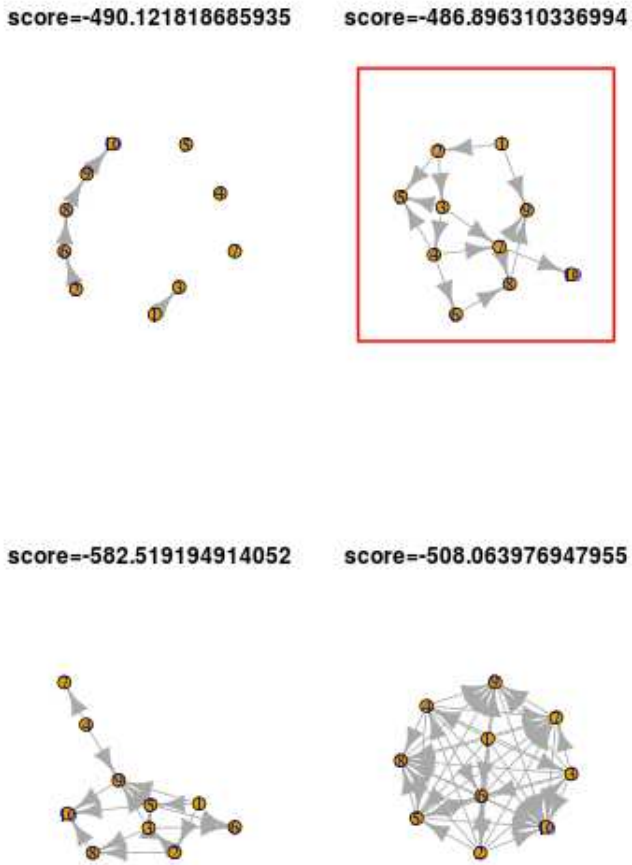


FIGURE 6.3 – Valeur du score pour quelques graphes. Le meilleur modèle est entouré en rouge.

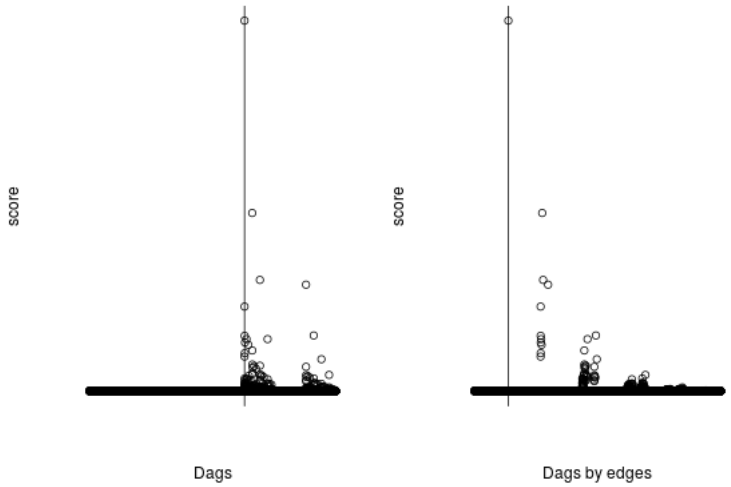


FIGURE 6.4 – Valeur du score pour l'ensemble des DAGs de 7 noeuds, à partir d'un jeu de données simulé. A gauche, les DAGs ayant une structure proche ont un classement proche, c'est-à-dire que deux graphes successifs ne diffères que d'une arête. A droite, les DAGs sont classés en fonction du nombre d'arêtes. La barre verticale indique le graphe de référence.

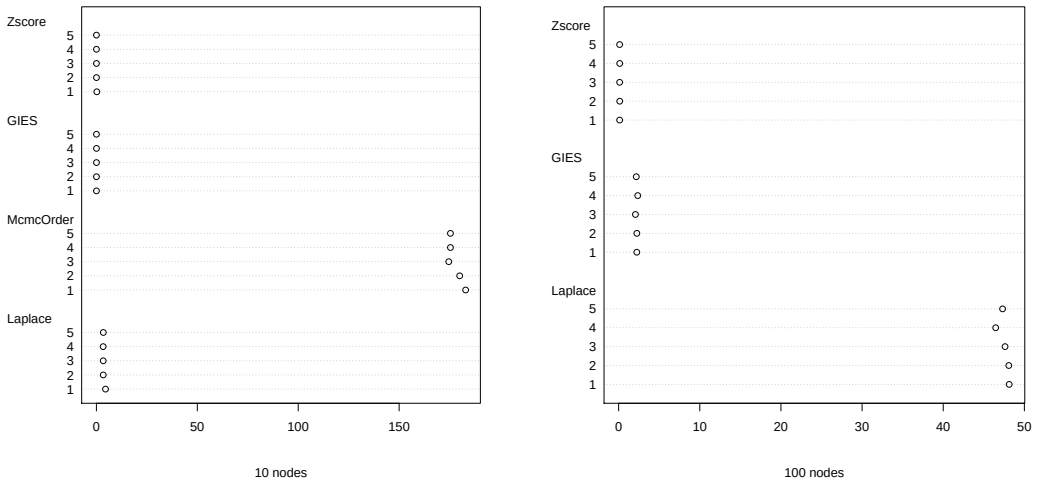


FIGURE 6.5 – Vitesse de calcul en microsecondes pour la méthode Zscore, GIES, MCMC-Order et notre approximation de Laplace sur le benchmark DREAM 4. A gauche, pour 10 noeuds, et à droite pour 100 noeuds.

6.4 Discussion-Ouverture

Les résultats de ce chapitre montrent l'intérêt computationnel de la navigation dans l'espace des structures. En évitant d'avoir à estimer l'intégralité des paramètres, mais seulement ceux d'intérêt à chaque itération, on réduit énormément la durée de calcul. De ce fait, notre algorithme est applicable à un plus grand nombre de gènes. Reste la question du mélange et de la convergence de la chaine de Markov : il faudra vraisemblablement inventer des lois de propositions permettant de converger plus vite vers la loi à posteriori. En effet, outre les différents graphes statistiquement équivalents, bien que relativement éloignés selon la procédure MC3, la loi de proposition de cette dernière doit couvrir un espace qui grandit de manière super exponentielle en fonction du nombre de nœuds. De manière générale, l'avantage de notre méthode comparée aux méthodes relativement efficaces actuelles est l'obtention d'une distribution a posteriori, ce qui permet d'analyser plus en profondeur les résultats. En particulier, on peut tenter de quantifier l'incertitude via un intervalle de crédibilité, chose impossible avec des méthodes gloutonnes ou basées sur des tests fréquentistes. Cela laisse la possibilité à diverses améliorations, comme l'introduction du *parallel tempering* pour mieux gérer les optimums locaux [6].

Chapitre 7

Discussion

Contributions

Nous avons dans le cadre de cette thèse proposé plusieurs orientations pour améliorer l'inférence causale de réseaux de gènes à partir de données d'observations et d'interventions.

Nous nous sommes tout d'abord dans le chapitre 2 interrogé sur l'hypothèse d'acyclicité qui est classiquement faite dans l'inférence de réseaux. Celle-ci est en effet peu crédible dans de nombreuses applications, avec la présence de boucles de rétroaction. Nous avons effectué une synthèse des connaissances actuelles sur les réseaux orientés cycliques. Nous avons démontré qu'il était possible de calculer la constante de normalisation dans ce cas, ce qui, à notre connaissance, n'avait encore jamais été fait, et qui offre la possibilité de simuler des données facilement dans le cadre d'un graphe cyclique. Nous avons cependant observé que l'hypothèse d'acyclicité ne peut être relâchée dans le cadre de l'inférence de réseaux et l'hypothèse DAG a donc été maintenue dans les différents chapitres de cette thèse.

Dans le cas d'une expérience avec un knock-out, nous avons proposé dans le chapitre 3 une approche marginale causale permettant d'inférer les liens causaux entre ce gène et tous les autres gènes inclus dans l'expérience de transcriptomique. Cette approche est très rapide et peut être appliquée sur plusieurs milliers de gènes simultanément. Elle permet de faire une pré-sélection de gènes d'intérêt au préalable d'une inférence de réseaux causaux. Cette étude a également permis de mettre en lumière le caractère causal de l'analyse différentielle dans le cas d'une expérience avec un knock-out. Elle a donné lieu à un article publié dans *PLOS One*, ainsi qu'à plusieurs présentations dans des congrès nationaux et internationaux.

Dans le cadre du recueil de données en génomique, il est courant de n'obtenir qu'un très faible nombre de réplicats, notamment du fait du coût de l'expérience. En raison de l'aspect grandement multidimensionnel du problème de l'inférence de réseaux, ceci donne lieu à du sur-ajustement, ce qui signifie que le pouvoir prédictif du modèle est extrêmement faible, l'estimation étant calquée sur les valeurs de l'échantillon. Pour résoudre ce problème, nous avons proposé dans le chapitre 4 d'incorporer une pénalité de type L2 dans le modèle. Cette méthode a été valorisée par une publication dans le journal RIA. Nous nous sommes aussi posé la question de la rapidité de l'inférence de réseaux. Lorsqu'une approche de type MCMC est utilisée, l'inférence peut être extrêmement longue, d'autant plus si le nombre de noeuds est conséquent. En collaboration avec une équipe allemande, nous avons proposé dans le chapitre 5 de paralléliser l'algorithme, utilisant ainsi les propriétés des architectures modernes. Cette parallélisation permet également d'utiliser une technique dite de *parallel tempering*, qui améliore la convergence

d'une chaîne de Markov. Un article est en cours de préparation sur ce travail. Nous avons également changé le langage de programmation en passant de R à C.

L'inférence de réseaux pose le problème de la sélection de modèle sans être dans le cadre de modèles emboîtés. Avoir un score qui tient compte de la dimension du modèle, et l'utiliser pour effectuer une inférence, est une question méthodologique en soit. J'ai ainsi travaillé dans le chapitre 6 sur l'élaboration d'un score basé sur une approximation de Laplace, celle-ci tenant compte de la dimension du modèle. Elle permet en outre de ne pas se limiter aux a priori conjugués. Cette approche est prometteuse et beaucoup plus rapide en temps de calcul que l'algorithme MCMC précédemment proposé.

Limites de l'inférence causale

Si l'inférence causale est aujourd'hui en partie possible, c'est grâce à notre capacité à capter et traiter un grand nombre de données. En effet, les réseaux orientés sont d'une complexité super-exponentielle, si bien qu'un réseau de 5 noeuds n'est déjà plus traitable "à la main". L'idée même d'essayer de traiter ce genre de problème sans ordinateur est incongrue.

Plutôt que de se concentrer sur l'inférence de réseaux, c'est en cherchant à spécifier un modèle précis, décrivant déjà les relations de cause à effet, que Wright a posé les bases des modèles graphiques au début du siècle précédent. Cette idée, qui cherche à caractériser un modèle que l'on pense vrai plutôt qu'à l'inférer, a ainsi poursuivi son chemin jusqu'à aujourd'hui. L'algorithme IDA, qui cherche seulement à caractériser les effets causaux dans une classe d'équivalence donnée, en est la dernière caractérisation.

Si la tentation de détourner le problème, en passant de l'inférence, à la caractérisation des effets causaux dans un modèle déjà spécifié est si grande, c'est parce que la question de l'inférence est extrêmement difficile. Une double contrainte s'applique. D'une part, on est poussé à inclure le minimum de variables possible pour lutter contre la dimension du problème. D'autre part, on est aussi tenté d'en inclure un maximum pour limiter le risque qu'induisent des variables cachées. La capacité d'obtenir ces données étant aussi une limite dans ce dilemme.

Cela explique l'émergence des réseaux de régulation génique. Ceux-ci n'ont pas de réalité physique directe, et ne modélisent donc pas une cascade d'interactions classique comme peuvent avoir l'habitude les biologistes. Cependant, le coût pour obtenir des données d'expressions de gènes est faible, contrairement à d'autres types de données, comme les métabolites. C'est pourquoi il est plus facile de travailler sur des modèles n'incluant que celles-ci, la charge de l'interprétation revenant ensuite au biologiste. Il reste donc encore du chemin à parcourir en terme d'acquisition de données pour pouvoir créer une théorie causale cohérente en génomique.

La méthodologie actuelle a aussi ses limites. L'analyse différentielle, à la suite d'une intervention, permet de saisir un ensemble de gènes candidats pour établir une chaîne de causalité. Cependant, elle est limitée par la possibilité de l'effectuer, certains gènes éteints pouvant mener directement à la mort ou au mauvais développement de l'organisme. De plus, elle ne permet que d'analyser la chaîne causale d'un gène précis et peut passer à coté d'effets de compensation.

L'inférence de réseaux, qu'elle soit faite par tests ou par scores, est limitée par la complexité du problème. La méthode la plus fiable semble être l'approche gloutonne, qui n'offre aucune garantie globale de convergence. De plus, certaines structures sont particulièrement difficiles à estimer.

Plan optimal d'expériences interventionnelles

Les travaux en génomique incluent de plus des interventions, et on peut s'attendre à ce que cela se généralise grâce aux techniques de type CRISPR-Cas9. Pour autant, les considérations éthiques continueront de limiter celles-ci. Cela pose la question du choix de ces interventions.

L'avis des experts est bien sûr crucial, mais si des données sont déjà acquises, elles peuvent être utilisées pour déterminer une prochaine intervention optimale. Cela peut se faire de plusieurs manières. On peut par exemple en premier lieu effectuer une inférence du DAG, puis en second lieu utiliser l'information de Fisher. Ceci permet de signifier le degré de confiance de l'estimation de chaque paramètre. Le choix des interventions peut être fait ensuite dans le but de maximiser cette information. Une autre façon de faire est d'estimer une classe d'équivalence et les différentes valeurs d'effets causaux possibles associées. On peut ensuite choisir d'effectuer les interventions permettant de réduire au maximum la classe d'équivalence en terme d'effets causaux potentiels.

Les interventions effectuées peuvent être de natures diverses. Dans le cadre des réseaux de gènes, ces interventions peuvent être produites via une intervention chimique extérieure, comme l'usage d'un ARN interférent ou du complexe CRISPR-Cas9, mais aussi produit à l'issue d'une mutation naturelle sur un gène. Celle-ci produira alors une protéine dont la configuration tridimensionnelle aura été modifiée, et en conséquence dont l'action aura été annulée. Souvent modélisées comme une inhibition ou une activation totale, elles peuvent aussi être de petites perturbations de la variable concernée, à la hausse comme à la baisse.

Conclusions

L'inférence de réseaux, et en particulier de réseaux de régulation de gènes, est un sujet compliqué pour de multiples raisons. L'étape de modélisation est cruciale, car elle permet d'expliciter les nombreuses hypothèses possibles pour ces réseaux. Gaussienne ou non, linéaire ou non, acyclicité ou cyclicité, présence d'interventions, variables cachées, données temporelles, sont tout autant d'hypothèses qui permettent de faciliter l'établissement d'un résultat. Mais cela peut souvent être au prix d'un recueil de données plus difficile. Il faut donc garder en tête qu'avec les hypothèses les plus éloignées de la réalité, l'inférence effectuée ne sera qu'un indice du déroulement réel des processus en cours. Si cela est possible, le recours aux données temporelles, ou l'exhaustivité des variables est préférable.

J'ai effectué cette thèse en programmant majoritairement en R. C'est un langage de programmation interprété très utile du fait des nombreux paquets disponibles, et notamment dans l'inférence de réseaux. Cependant certains algorithmes d'inférence peuvent être très gourmands en calculs, notamment ceux basés sur les chaînes de Marvov. J'ai alors dû utiliser un langage compilé comme C, permettant de bien mieux gérer la mémoire et ainsi diminuer le temps de calcul d'un facteur 100. Les progrès en terme de recueil des données étant constants, l'avenir de l'inférence de réseaux en biologie et plus particulièrement en génomique semble se porter sur la capacité de modéliser et d'assembler les différentes catégories de données accessibles. Cela permettra aux méthodes d'inférence de consolider leur interprétation et leur caractère causal.

Annexe A

Annexes

A.1 Approche bayésienne

Bayesien et fréquentiste

J'ai essentiellement dans le cadre de cette thèse travaillé sous une approche bayésienne, en m'appuyant sur les livres de références de Gelman et al. [35] et Marin and Robert [67]. Tout d'abord, celle-ci n'a été possible que par l'émergence d'une puissance de calcul suffisante. En effet, les bases théoriques ont été posées depuis Thomas Bayes et Pierre Simon de Laplace au cours du 18eme siècle. Il aura fallu cependant attendre jusque dans les années 1980 pour que, à travers notamment les méthodes de Monte-Carlo par chaîne de Markov, cette vision devienne numériquement applicable. En effet, une partie essentielle de l'approche bayésienne est le calcul de la distribution a posteriori.

Rappelons ici à travers un tableau les avantages et les inconvénients du bayésien en comparaison du fréquentiste.

	Bayésien	Fréquentiste
Vision	subjective	objective
Théorème	Théorème de Bayes	Théorème central limite
Paramètres	Aléatoires	Fixés
Résultat	Distribution	Test

Expliquons ce tableau. La vision fréquentiste consiste a modéliser le problème sous la forme de variables aléatoires. Une fois cette modélisation faites, les données obtenues sont supposées réalisées par ces variables aléatoires, généralement assez simples. L'approche fréquentiste est objective, dans le sens où les données ne peuvent pas avoir une autre interprétation que celle suggérée dans le modèle. En utilisant alors le théorème central limite, on peut extrapoler les données en présence, et utiliser des résultats en imaginant avoir une infinité de données. On peut alors ainsi calculer les différents paramètres du modèle via des estimateurs, et éventuellement tester des hypothèses ou donner un intervalle de confiance basé sur ces résultats asymptotiques.

En revanche l'approche bayésienne, toujours après une étape de modélisation, nécessite de choisir une probabilité a priori $\mathbb{P}(\theta)$ pour les données. Ce choix est subjectif et dépend des connaissances vis-à-vis des données. Suite à ce choix, et en utilisant le théorème de bayes, on en déduit à une constante près la loi de probabilité a posteriori des paramètres du modèle $\mathbb{P}(\theta|D)$, en fonction de la vraisemblance et de l'a priori. Cette loi

est généralement complexe. En conséquence pour en calculer un moment d'ordre quelconque (généralement d'ordre 1), il est nécessaire de procéder à une intégration. Celle-ci est généralement impossible analytiquement, d'où la nécessité d'une intégration numérique. En utilisant par exemple les MCMC, on peut simuler cette loi en calculant les moments empiriques. On peut si nécessaire utiliser des intervalles de crédibilités pour quantifier l'incertitude sur les paramètres, ou des facteurs de bayes pour choisir entre différents modèles.

A noter que sous certaines hypothèses, il existe un théorème central limite sous l'approche bayésienne. Dans de tels cas, les deux approches se confondent en présence d'un grand nombre de données.

Formulation bayésienne et MCMC

Écrivons ici plus précisément la manière dont se déroule une étude sous le prisme du bayésien. Tout d'abord, en fonction du problème, le modèle est spécifié. Celui-ci inclus un certain nombre de variables aléatoires toutes incluses dans le vecteur $\mathbf{X}$, dont chacune dispose d'un ou de plusieurs paramètres inclus dans le vecteur $\boldsymbol{\theta}$. Ces paramètres sont aléatoires, et une loi a priori $\mathbb{P}(\boldsymbol{\theta})$ est choisie. En fonction du modèle et en posant D les données a disposition, la vraisemblance $\mathbb{P}(\mathrm{D}|\boldsymbol{\theta})$ est calculable en fixant θ. En utilisant ensuite le théorème de Bayes, on obtient :

$$\mathbb{P}(\boldsymbol{\theta}|\mathrm{D}) = \frac{1}{\mathbb{P}(\mathrm{D})}\mathbb{P}(\mathbf{D}|\boldsymbol{\theta})\,\mathbb{P}(\boldsymbol{\theta}).$$

La quantité $\mathbb{P}(\mathrm{D})$ est indépendante du paramètre, il s'agit d'une constante entièrement dépendante des données. Pour pouvoir visualiser cette loi on peut la simuler, en utilisant une procédure MCMC. Un autre choix est d'utiliser à escient le choix de l'a priori pour obtenir un a posteriori simple : il s'agit des lois conjuguées. Cependant pour des données de faibles volumes ou traduisant un espace hautement multidimensionnel, l'a priori à une grande influence sur le résultat, et il est souhaitable de le choisir avec plus de minutie.

De nombreux algorithmes MCMC existent, mais on peut en ressortir deux majeurs : l'algorithme de Métropolis-Hastings et l'échantillonnage de Gibbs.

L'algorithme de Métropolis-Hastings (Algorithme 12) consiste à créer une chaîne de Markov de la façon suivante. On cherche à créer des échantillons d'une loi π complexe. On sait par ailleurs proposer des échantillons à partir d'une loi q symétrique, de telle sorte que $q(x|y) = q(y|x)$. Cette loi va servir à effectuer des propositions à partir d'un point initial. Cependant, pour que la chaîne tende vers la loi π, il est nécessaire qu'elle vérifie la balance détaillée :

$$\mathbb{P}\left(y|x\right)\mathbb{P}\left(x\right) = \mathbb{P}\left(x|y\right)\mathbb{P}\left(y\right).$$

De telle sorte qu'une phase d'acception est nécessaire pour s'assurer de cela. Ce taux d'acceptation est alors de :

$$\tau = \min\left\{1, \frac{\mathbb{P}\left(x_{new}\right)q\left(x_{old}|x_{new}\right)}{\mathbb{P}\left(x_{old}\right)q\left(x_{new}|x_{old}\right)}\right\}.$$

En sortie nous obtenons des échantillons de la loi cible. En pratique, on compensera la corrélation entre les différents échantillons par un seuillage, par exemple en ne prenant qu'un échantillon sur 100. On s'assurera de plus la convergence de l'algorithme graphiquement, en élaguant la première moitié ou le premier quart de la chaîne.

L'échantillonnage de Gibbs (Algorithme 13) quant à lui est utilisé dans les systèmes de très haute dimension. Dans de tels cas, il est éventuellement plus facile de simuler une

Algorithme 12 : Metropolis-Hastings

Entrées : Données $\mathscr{D}$, Loi de proposition q,
Initialisation $x^{(0)}$, Iteration maximale $i_{\max}$
Sorties : Echantillons issus de π
pour $i \in \{1, ..., i_{\max}\}$ **faire**
  $x^{(i)} \sim q(x|x^{(i-1)})$
  $\mathscr{T}_{x^{(i-1)},x^{(i)}} = \min\left(\frac{\pi(x^{(i)})q(x^{(i)}|x^{(i-1)})}{\pi(x^{(i-1)})q(x^{(i-1)}|x^{(i)})}, 1\right)$
  Avec probabilité $1 - \mathscr{T}_{x^{(i-1)},x^{(i)}}$, refuser le mouvement et fixer $x^{(i)} = x^{(i-1)}$
retourner $\boldsymbol{x} = \left(x^{(0)}, ..., x^{(i_{\max})}\right)$

variable conditionnellement à toutes les autres : c'est ce que propose l'échantillonnage de Gibbs.

Algorithme 13 : Échantillonnage de Gibbs

Entrées : Données $\mathscr{D}$,
Initialisation $x^{(0)}$, Iteration maximale $i_{\max}$, Dimension p
Sorties : Echantillons issus de π
pour $i \in \{1, ..., i_{\max}\}$ **faire**
  pour $k \in \{1, ..., p\}$ **faire**
    Simuler x_k^i avec la loi conditionnelle $\mathbb{P}\left(x_k | x_1^i, ..., x_{k-1}^i, x_{k+1}^{i-1}, ..x_p^{i-1}\right)$
retourner $\boldsymbol{x} = \left(x^{(0)}, ..., x^{(i_{\max})}\right)$

Le principe est d'évaluer les nouveaux échantillons en fonction de l'information transmise par les anciens échantillons. Plutôt que de faire un balayage, d'autre formes de Gibbs, comme le Gibbs aléatoire, ou par blocs.

Pénalités

Ici je cherche a faire le lien entre l'utilisation de pénalités dans des procédures de score et la réalité de l'a priori associées. De nombreuses pénalités ont été dérivées dans la littératures, les plus connues étant peut-être les pénalités BIC, pénalités L1 et L2 :

$$\texttt{score} = -\ell + \frac{1}{2}\text{pen}.$$

Elles cherchent en général à pénaliser la dimension, pour mettre en application le principe philosophique du rasoir d'Ockham : les modèles les plus simples sont les meilleurs. En ce sens la pénalité BIC est la plus adaptée, néanmoins elle n'est systématiquement appliquée, car elle transforme des problèmes convexes en problèmes non-convexe. La pénalité L1 est la transformation convexe du problème résolue par le BIC : c'est cette convexification permet de s'assurer de trouver un maximum global. Si un système linéaire est en jeu dans la résolution du problème, il peut arriver que l'on manque de données pour permettre une telle résolution. Ceci est particulièrement vrai pour des problèmes en grande dimension. L'avantage de ces pénalités est qu'elles créent une dépendance entre les différentes variables, permettant une résolution même pour un faible nombre de données. Cependant les calculs peuvent être extrêmement compliqués à gérer. La pénalité L2 n'offre pas de possibilité de sélection de variable, mais elle permet cependant, avec des

calculs efficients, d'utiliser cette propriété de dépendances pour permettre la résolution de ces problèmes. On rappelle ici comment on dérive ces différentes pénalités d'une façon bayésienne.

La pénalité BIC est en fait un résultat asymptotique issue d'une approximation de Laplace. Plutôt que d'essayer d'estimer un paramètre θ, on essaye de choisir entre différents modèles m. En choisissant un certain a priori sur ces modèles, on obtient la formulation de la loi a posteriori :

$$\mathbb{P}(m|\mathrm{D}) \propto \mathbb{P}(\mathrm{D}|m)\,\mathbb{P}(m).$$

On ne sait pas quantifier directement $\mathbb{P}(\mathrm{D}|m)\,\mathbb{P}(m)$, cependant on peut calculer cette quantité à travers l'intégrale suivante :

$$\int \mathbb{P}(\mathrm{D}|\theta, m)\,\mathbb{P}(\theta|m)\,\mathbb{P}(m)\,d\theta.$$

On a donc un a priori sur les paramètres en fonction du modèle. On peut approximer cette quantité via une approximation de Laplace. Cela nous donne l'expression suivante :

$$\mathbb{P}(m|\mathrm{D}) \approx \mathbb{P}(\mathrm{D}|\widehat{\theta}, m)\mathbb{P}\left(\widehat{\theta}|m\right)\mathbb{P}(m)\sqrt{\frac{(2\pi)^{\mathrm{K}}}{\det(\mathbf{A})}},$$

où $\mathscr{A}$ est la Hessienne de $\log(\mathbb{P}(\mathrm{D}|\theta, m)\,\mathbb{P}(\theta|m))$, et $\widehat{\theta}$ correspond au maximum a posteriori. En passant au logarithme, on peut prouver que $\sqrt{\frac{(2\pi)^{\mathrm{K}}}{\det(\mathbf{A})}}$ tend vers $\frac{\log(n)k}{2}$. Enfin si on choisit un a priori uniforme sur les paramètres et les modèles, on obtient :

$$\texttt{score} = -\ell + \frac{\log(n)k}{2}.$$

La pénalisation de L1, ou lasso, peut être vue comme l'application d'un a priori de Laplace. Rappelons que la densité de probabilité de Laplace s'écrit :

$$f(x;c) = c\frac{\exp(-c|x|)}{2}, \quad c > 0.$$

Lorsque l'on reprend une formulation bayésienne et que l'on applique le logarithme, on obtient :

$$\mathbb{P}(\boldsymbol{\theta}|\mathrm{D}) \propto \log(\mathbb{P}(\mathbf{D}|\boldsymbol{\theta})) + \log(c/2) - c|x|.$$

En occultant la constante et en réécrivant légèrement l'expression, on obtient, :

$$\texttt{score} = -\ell + \frac{\lambda}{2}|x|.$$

On peut effectuer le même raisonnement pour la pénalité l2, ou ridge. L'a priori choisit dans ce cas est gaussienne centré. On obtient en unidimensionnel :

$$\mathbb{P}(\boldsymbol{\theta}|\mathrm{D}) \propto \log(\mathbb{P}(\mathbf{D}|\boldsymbol{\theta})) - \log(\sqrt{2\pi}\sigma) - \frac{1}{2\sigma^2}x^2.$$

En réécrivant, on obtient :

$$\texttt{score} = -\ell + \frac{\lambda}{2}x^2.$$

Annexe B

Bibliographie

B.1 Références

[1] Albert, R. (2005). Scale-free networks in cell biology. *Journal of cell science*, 118(21) :4947–4957. 10

[2] Altay, G. and Emmert-Streib, F. (2010). Inferring the conservative causal core of gene regulatory networks. *BMC Systems Biology*, 4(1) :132. 44

[3] Anders, S. and Huber, W. (2010). Differential expression analysis for sequence count data. *Genome biology*, 11(10) :R106. 9

[4] Andersson, S. A., Madigan, D., Perlman, M. D., et al. (1997). A characterization of markov equivalence classes for acyclic digraphs. *The Annals of Statistics*, 25(2) :505–541. 13

[5] Banerjee, S. and Ghosal, S. (2015). Bayesian structure learning in graphical models. *Journal of Multivariate Analysis*, 136 :147–162. 86

[6] Barker, D., Hill, S., and Mukherjee, S. (2010). Mc 4 : a tempering algorithm for large-sample network inference. *Pattern Recognition in Bioinformatics*, pages 431–442. 96

[7] Basso, K., Margolin, A. A., Stolovitzky, G., Klein, U., Dalla-Favera, R., and Califano, A. (2005). Reverse engineering of regulatory networks in human b cells. *Nature genetics*, 37(4) :382–390. 44

[8] Ben-David, E., Li, T., Massam, H., and Rajaratnam, B. (2011). High dimensional bayesian inference for gaussian directed acyclic graph models. *arXiv preprint arXiv :1109.4371*. 86

[9] Bühlmann, P., Kalisch, M., and Meier, L. (2014). High-dimensional statistics with a view toward applications in biology. *Annual Review of Statistics and Its Application*, 1 :255–278. 43, 58

[10] Bumgarner, R. (2013). Overview of dna microarrays : types, applications, and their future. *Current protocols in molecular biology*, pages 22–1. 8

[11] Butte, A. J. and Kohane, I. S. (2000). Mutual information relevance networks : functional genomic clustering using pairwise entropy measurements. In *Pac Symp Biocomput*, volume 5, page 26. 10

[12] Campos, L. M. d. (2006). A scoring function for learning bayesian networks based on mutual information and conditional independence tests. *Journal of Machine Learning Research*, 7(Oct) :2149–2187. 10

[13] Carter, S. L., Brechbühler, C. M., Griffin, M., and Bond, A. T. (2004). Gene co-expression network topology provides a framework for molecular characterization of cellular state. *Bioinformatics*, 20(14) :2242–2250. 10

[14] Cartwright, N. and McMullin, E. (1984). How the laws of physics lie. 14

[15] Chickering, D. M. (2003). Optimal structure identification with greedy search. *The Journal of Machine Learning Research*, 3 :507–554. 21, 43, 58, 76

[16] Chickering, D. M. and Heckerman, D. (1997). Efficient approximations for the marginal likelihood of bayesian networks with hidden variables. *Machine learning*, 29(2-3) :181–212. 90

[17] Chickering, D. M., Heckerman, D., and Meek, C. (1997). A bayesian approach to learning bayesian networks with local structure. In *Proceedings of the Thirteenth conference on Uncertainty in artificial intelligence*, pages 80–89. Morgan Kaufmann Publishers Inc. 43

[18] Clarke, B., Leuridan, B., and Williamson, J. (2014). Modelling mechanisms with causal cycles. *Synthese*, 191(8) :1651–1681. 30

[19] De La Fuente, A., Bing, N., Hoeschele, I., and Mendes, P. (2004). Discovery of meaningful associations in genomic data using partial correlation coefficients. *Bioinformatics*, 20(18) :3565–3574. 10

[20] de Matos Simoes, R., Dehmer, M., and Emmert-Streib, F. (2013). Interfacing cellular networks of s. cerevisiae and e. coli : Connecting dynamic and genetic information. *BMC genomics*, 14(1) :1. 43

[21] de Matos Simoes, R. and Emmert-Streib, F. (2012). Bagging statistical network inference from large-scale gene expression data. *PLoS One*, 7(3) :e33624. 44, 76

[22] Dobra, A., Eicher, T. S., and Lenkoski, A. (2010). Modeling uncertainty in macroeconomic growth determinants using gaussian graphical models. *Statistical Methodology*, 7(3) :292–306. 17

[23] Doignon, J.-P., Pekeč, A., and Regenwetter, M. (2004). The repeated insertion model for rankings : Missing link between two subset choice models. *Psychometrika*, 69(1) :33–54. 78

[24] Dolinoy, D. C. (2008). The agouti mouse model : an epigenetic biosensor for nutritional and environmental alterations on the fetal epigenome. *Nutrition reviews*, 66(suppl_1) :S7–S11. 6

[25] Duriez, B., Sobrier, M., Duquesnoy, P., Tixier-Boichard, M., Decuypere, E., Coquerelle, G., Zeman, M., Goossens, M., and Amselem, S. (1993). A naturally occurring growth hormone receptor mutation : in vivo and in vitro evidence for the functional importance of the ws motif common to all members of the cytokine receptor superfamily. *Molecular endocrinology*, 7(6) :806–814. 44, 52

[26] D'haeseleer, P., Liang, S., and Somogyi, R. (2000). Genetic network inference : from co-expression clustering to reverse engineering. *Bioinformatics*, 16(8) :707–726. 10

[27] Emmert-Streib, F., Dehmer, M., and Haibe-Kains, B. (2014). Gene regulatory networks and their applications : understanding biological and medical problems in terms of networks. *Frontiers in cell and developmental biology*, 2 :38. 9, 50

[28] Emmert-Streib, F., Glazko, G., De Matos Simoes, R., et al. (2012). Statistical inference and reverse engineering of gene regulatory networks from observational expression data. *Frontiers in genetics*, 3 :8. 43

[29] Folmer, E. O., van der Geest, M., Jansen, E., Olff, H., Anderson, T. M., Piersma, T., and van Gils, J. A. (2012). Seagrass–sediment feedback : an exploration using a non-recursive structural equation model. *Ecosystems*, 15(8) :1380–1393. 30

[30] Friedman, J., Hastie, T., and Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3) :432–441. 10, 17, 44, 58

[31] Friedman, N. and Koller, D. (2003). Being bayesian about network structure. a bayesian approach to structure discovery in bayesian networks. *Machine learning*, 50(1-2) :95–125. 26, 76, 86

[32] Friedman, N., Linial, M., Nachman, I., and Pe'er, D. (2000). Using bayesian networks to analyze expression data. *Journal of computational biology*, 7(3-4) :601–620. 77

[33] Fu, F. and Zhou, Q. (2013). Learning sparse causal gaussian networks with experimental intervention : regularization and coordinate descent. *Journal of the American Statistical Association*, 108(501) :288–300. 44, 64

[34] Geiger, D. and Heckerman, D. (1999). Parameter priors for directed acyclic graphical models and the characterization of several probability distributions. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 216–225. Morgan Kaufmann Publishers Inc. 86

[35] Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2014). *Bayesian data analysis*, volume 2. CRC press Boca Raton, FL. I

[36] Gillispie, S. B. and Perlman, M. D. (2001). Enumerating markov equivalence classes of acyclic digraph dels. In *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*, pages 171–177. Morgan Kaufmann Publishers Inc. 14

[37] Goudie, R. J. and Mukherjee, S. (2016). A gibbs sampler for learning dags. *The Journal of Machine Learning Research*, 17(1) :1032–1070. 86

[38] Granger, C. W. (1969). Investigating causal relations by econometric models and cross-spectral methods. *Econometrica : Journal of the Econometric Society*, pages 424–438. 15

[39] Greenfield, A., Hafemeister, C., and Bonneau, R. (2013). Robust data-driven incorporation of prior knowledge into the inference of dynamic regulatory networks. *Bioinformatics*, 29(8) :1060–1067. 86

[40] Grzegorczyk, M. and Husmeier, D. (2008). Improving the structure mcmc sampler for bayesian networks by introducing a new edge reversal move. *Machine Learning*, 71(2) :265–305. 86

[41] Hansen, K. D., Irizarry, R. A., and Wu, Z. (2012). Removing technical variability in rna-seq data using conditional quantile normalization. *Biostatistics*, 13(2) :204–216. 9

[42] Hastings, W. K. (1970). Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1) :97–109. 62, 78

[43] Hauser, A. and Bühlmann, P. (2012). Characterization and greedy learning of interventional markov equivalence classes of directed acyclic graphs. *Journal of Machine Learning Research*, 13(Aug) :2409–2464. 14, 50, 92

[44] Hauser, A. and Bühlmann, P. (2015). Jointly interventional and observational data : estimation of interventional markov equivalence classes of directed acyclic graphs. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 77(1) :291–318. 76

[45] He, Y., Jia, J., Yu, B., et al. (2013). Reversible mcmc on markov equivalence classes of sparse directed acyclic graphs. *The Annals of Statistics*, 41(4) :1742–1779. 76

[46] Heather, J. M. and Chain, B. (2016). The sequence of sequencers : The history of sequencing dna. *Genomics*, 107(1) :1–8. 8

[47] Hsu, P. D., Lander, E. S., and Zhang, F. (2014). Development and applications of crispr-cas9 for genome engineering. *Cell*, 157(6) :1262–1278. 53

[48] Hyttinen, A., Eberhardt, F., and Hoyer, P. O. (2012). Learning linear cyclic causal models with latent variables. *Journal of Machine Learning Research*, 13(Nov) :3387–3439. 30

[49] Kalisch, M. and Bühlmann, P. (2007). Estimating high-dimensional directed acyclic graphs with the pc-algorithm. *The Journal of Machine Learning Research*, 8 :613–636. 43, 58

[50] Kasza, J. and Solomon, P. (2015). Comparing score-based methods for estimating bayesian networks using the kullback–leibler divergence. *Communications in Statistics-Theory and Methods*, 44(1) :135–152. 86

[51] Kim, S. Y., Imoto, S., and Miyano, S. (2003). Inferring gene networks from time series microarray data using dynamic bayesian networks. *Briefings in bioinformatics*, 4(3) :228–235. 30

[52] Kistler, M. (2004). La causalité dans la philosophie contemporaine. *Intellectica*, 38(1) :139–185. 14

[53] Koller, D. and Friedman, N. (2009). *Probabilistic graphical models : principles and techniques*. MIT press. 10

[54] Krumsiek, J., Suhre, K., Illig, T., Adamski, J., and Theis, F. J. (2011). Gaussian graphical modeling reconstructs pathway reactions from high-throughput metabolomics data. *BMC systems biology*, 5(1) :21. 17

[55] Lacerda, G., Spirtes, P. L., Ramsey, J., and Hoyer, P. O. (2012). Discovering cyclic causal models by independent components analysis. *arXiv preprint arXiv :1206.3273*. 30

[56] Lauritzen, S. L. (1996). *Graphical models*, volume 17. Clarendon Press. 12, 30

[57] Law, C. W., Chen, Y., Shi, W., and Smyth, G. K. (2014). Voom : precision weights unlock linear model analysis tools for rna-seq read counts. *Genome biology*, 15(2) :R29. 9

[58] Lee, W.-P. and Tzou, W.-S. (2009). Computational methods for discovering gene networks from expression data. *Briefings in bioinformatics*, 10(4) :408–423. 76

[59] Lesne, A. (2009). Biologie des systèmes-l'organisation multiéchelle des systèmes vivants. *médecine/sciences*, 25(6-7) :585–587. 7

[60] Luo, W., Hankenson, K. D., and Woolf, P. J. (2008). Learning transcriptional regulatory networks from high throughput gene expression data using continuous three-way mutual information. *BMC bioinformatics*, 9(1) :1. 43

[61] Maathuis, M. H., Colombo, D., Kalisch, M., and Bühlmann, P. (2010). Predicting causal effects in large-scale systems from observational data. *Nature Methods*, 7(4) :247–248. 58, 76

[62] Maathuis, M. H., Kalisch, M., Bühlmann, P., et al. (2009). Estimating high-dimensional intervention effects from observational data. *The Annals of Statistics*, 37(6A) :3133–3164. 26, 44, 58

[63] Madigan, D., York, J., and Allard, D. (1995). Bayesian graphical models for discrete data. *International Statistical Review/Revue Internationale de Statistique*, pages 215–232. 24, 91

[64] Madigan, D. and York, J. C. (1997). Bayesian methods for estimation of the size of a closed population. *Biometrika*, 84(1) :19–31. 91

[65] Mallows, C. L. (1957). Non-null ranking models. i. *Biometrika*, pages 114–130. 62, 78

[66] Marbach, D., Prill, R. J., Schaffter, T., Mattiussi, C., Floreano, D., and Stolovitzky, G. (2010). Revealing strengths and weaknesses of methods for gene network inference. *Proceedings of the National Academy of Sciences*, 107(14) :6286–6291. 24, 44, 67, 76, 91

[67] Marin, J.-M. and Robert, C. (2007). *Bayesian core : a practical approach to computational Bayesian statistics*. Springer Science & Business Media. I

[68] Meinshausen, N., Hauser, A., Mooij, J. M., Peters, J., Versteeg, P., and Bühlmann, P. (2016). Methods for causal inference from gene perturbation experiments and validation. *Proceedings of the National Academy of Sciences*, 113(27) :7361–7368. 76

[69] Miller, J. A., Cai, C., Langfelder, P., Geschwind, D. H., Kurian, S. M., Salomon, D. R., and Horvath, S. (2011). Strategies for aggregating gene expression data : the collapserows r function. *BMC bioinformatics*, 12(1) :322. 9

[70] Mooij, J. and Heskes, T. (2013). Cyclic causal discovery from continuous equilibrium data. *arXiv preprint arXiv :1309.6849*. 30

[71] Nuel, G., Rau, A., and Jaffrézic, F. (2013). Joint likelihood calculation for intervention and observational data from a gaussian bayesian network. *arXiv preprint arXiv :1305.0709*. 62, 76, 77

[72] Nunkoo, R., Ramkissoon, H., and Gursoy, D. (2013). Use of structural equation modeling in tourism research : past, present, and future. *Journal of Travel Research*, 52(6) :759–771. 30

[73] Pearl, J. (1985). *Bayesian networks : A model of self-activated memory for evidential reasoning*. University of California (Los Angeles). Computer Science Department. 10

[74] Pearl, J. (2000). *Causality : models, reasoning and inference*, volume 29. Cambridge Univ Press. 14, 15, 34, 44, 45, 60

[75] Pinna, A., Heise, S., Flassig, R. J., de la Fuente, A., and Klamt, S. (2013). Reconstruction of large-scale regulatory networks based on perturbation graphs and transitive reduction : improved methods and their evaluation. *BMC Systems Biology*, 7(1). 43, 58

[76] Pinna, A., Soranzo, N., and De La Fuente, A. (2010). From knockouts to networks : establishing direct cause-effect relationships through graph analysis. *PLoS One*, 5(10) :e12912. 24, 43, 58, 76, 91

[77] Rau, A., Jaffrézic, F., and Nuel, G. (2013). Joint estimation of causal effects from observational and intervention gene expression data. *BMC Systems Biology*, 7(1) :111. 26, 44, 58, 59, 76, 77, 86, 92

[78] Richardson, T. (1994). Equivalence in non-recursive structural equation models. In *Compstat*, pages 482–487. Springer. 30, 38

[79] Richardson, T. (1995). A polynomial algorithm for deciding equivalence in directed cyclic graphical models. 32, 38

[80] Richardson, T. (1996). A discovery algorithm for directed cyclic graphs. In *Proceedings of the Twelfth international conference on Uncertainty in artificial intelligence*, pages 454–461. Morgan Kaufmann Publishers Inc. 37

[81] Roberts, G. O., Gelman, A., Gilks, W. R., et al. (1997). Weak convergence and optimal scaling of random walk metropolis algorithms. *The annals of applied probability*, 7(1) :110–120. 63

[82] Robinson, R. W. (1977). Counting unlabeled acyclic digraphs. *Combinatorial mathematics V*, 622(1977) :28–43. 14

[83] Rocha, E. (2000). Analyse exploratoire des génomes bactériens. *Génétique cellulaire et moléculaire, Université de Versailles Saint-Quentin-En-Yvelines*. 7

[84] Roverato, A. and Consonni, G. (2004). Compatible prior distributions for directed acyclic graph models. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 66(1) :47–61. 86

[85] Rubin, D. B. (1974). Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of educational Psychology*, 66(5) :688. 15

[86] Russell, B. (1912). On the notion of cause. In *Proceedings of the Aristotelian society*, volume 13, pages 1–26. JSTOR. 14

[87] Sanger, F., Nicklen, S., and Coulson, A. R. (1977). Dna sequencing with chain-terminating inhibitors. *Proceedings of the national academy of sciences*, 74(12) :5463–5467. 6

[88] Sanger, F. and Thompson, E. (1953). The amino-acid sequence in the glycyl chain of insulin. 1. the identification of lower peptides from partial hydrolysates. *Biochemical Journal*, 53(3) :353. 6

[89] Schawe, H., Bleim, R., and Hartmann, A. K. (2017). Phase transitions of the typical algorithmic complexity of the random satisfiability problem studied with linear programming. *arXiv preprint arXiv :1702.02821*. 76

[90] Schmidt, M. and Murphy, K. (2009). Modeling discrete interventional data using directed cyclic graphical models. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 487–495. AUAI Press. 30

[91] Schwarz, G. et al. (1978). Estimating the dimension of a model. *The annals of statistics*, 6(2) :461–464. 21

[92] Sen, G. L. and Blau, H. M. (2006). A brief history of rnai : the silence of the genes. *The FASEB journal*, 20(9) :1293–1299. 7

[93] Shalem, O., Sanjana, N. E., Hartenian, E., Shi, X., Scott, D. A., Mikkelsen, T. S., Heckl, D., Ebert, B. L., Root, D. E., Doench, J. G., et al. (2014). Genome-scale crispr-cas9 knockout screening in human cells. *Science*, 343(6166) :84–87. 7

[94] Smyth, G. K. (2004). Linear models and empirical bayes methods for assessing differential expression in microarray experiments. In *Statistical Applications in Genetics and Molecular Biology*, pages 1–25. 9, 49, 52

[95] Someren, E. v., Wessels, L., Backer, E., and Reinders, M. (2002). Genetic network modeling. *Pharmacogenomics*, 3(4) :507–525. 9

[96] Spirtes, P. (1994). Conditional independence in directed cyclic graphical models for feedback. 30

[97] Spirtes, P. (1995). Directed cyclic graphical representations of feedback models. In *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*, pages 491–498. Morgan Kaufmann Publishers Inc. 32

[98] Spirtes, P., Glymour, C., and Scheines, R. (1991). From probability to causality. *Philosophical Studies*, 64(1) :1–36. 19

[99] Spirtes, P., Glymour, C. N., and Scheines, R. (2000). *Causation, prediction, and search*, volume 81. MIT press. 10, 44, 76

[100] Strotz, R. H. and Wold, H. O. (1960). Recursive vs. nonrecursive systems : An attempt at synthesis (part i of a triptych on causal chain systems). *Econometrica : Journal of the Econometric Society*, pages 417–427. 30

[101] Tsamardinos, I., Brown, L. E., and Aliferis, C. F. (2006). The max-min hill-climbing bayesian network structure learning algorithm. *Machine learning*, 65(1) :31–78. 22, 76

[102] udea Pearl, T. V. J. (1991). Equivalence and synthesis of causal models. In *Proceedings of Sixth Conference on Uncertainty in Artijicial Intelligence*, pages 220–227. 13, 44

[103] Watkinson, J., Liang, K.-c., Wang, X., Zheng, T., and Anastassiou, D. (2009). Inference of regulatory gene interactions from expression data using three-way mutual information. *Annals of the New York Academy of Sciences*, 1158(1) :302–313. 43

[104] Watson, J. and Crick, F. (1953). Molecular structure of nucleic acids : A structure for deoxyribose nucleic acid. *Nature*, 171(737-738) :3–12. 6

[105] Weirauch, M. T. (2011). Gene coexpression networks for the analysis of dna microarray data. *Applied statistics for network biology : methods in systems biology*, pages 215–250. 10

[106] Weiss, Y. and Freeman, W. T. (2000). Correctness of belief propagation in gaussian graphical models of arbitrary topology. In *Advances in neural information processing systems*, pages 673–679. 17

[107] Wright, S. (1921). Correlation and causation. *Journal of agricultural research*, 20(7) :557–585. 14

[108] Wright, S. (1934). The method of path coefficients. *The annals of mathematical statistics*, 5(3) :161–215. 9

[109] Zhang, B., Horvath, S., et al. (2005). A general framework for weighted gene coexpression network analysis. *Statistical applications in genetics and molecular biology*, 4(1) :1128. 10