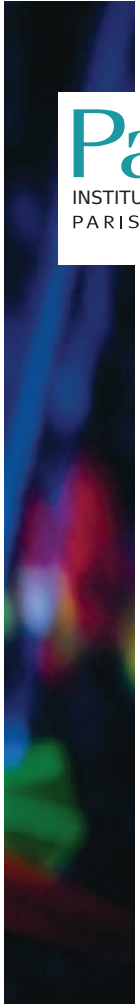
Modélisation probabiliste et inférence par l'algorithme Belief Propagation

Victorin Martin



ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY

MINES ParisTech

École doctorale n°432 : Sciences des Métiers de l'Ingénieur

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

l'École nationale supérieure des mines de Paris

Spécialité « Informatique temps réel, robotique et automatique »

présentée et soutenue publiquement par

Victorin MARTIN

le 23 mai 2013

Modélisation probabiliste et inférence par l'algorithme Belief Propagation

Directeur de thèse : **Arnaud de LA FORTELLE**
Co-encadrement de la thèse : **Cyril FURTLEHNER et Jean-Marc LASGOUTTES**

Jury

M. Nikolas GEROLIMINIS, Professeur associé, École Polytechnique Fédérale de Lausanne	Président
M. Jean-Michel LOUBES, Professeur, Université Paul Sabatier de Toulouse	Rapporteur
M. Didier PIAU, Professeur, Université Joseph Fourier de Grenoble	Rapporteur
M. Jean-Patrick LEBACQUE, Ingénieur général des ponts et chaussées, IFSTTAR	Examinateur
M. Arnaud de LA FORTELLE, Professeur, CAOR, Mines-Paristech	Directeur de thèse
M. Cyril FURTLEHNER, Chargé de Recherche, TAO, Inria	Co-directeur de thèse
M. Jean-Marc LASGOUTTES, Chargé de Recherche, IMARA, Inria	Co-directeur de thèse

MINES ParisTech
Centre de Robotique (CAOR)
60, boulevard Saint-Michel 75006 PARIS

T H È S E

Remerciements

Je tiens tout d'abord à remercier Fawzi Nashashibi et Marc Schoenauer qui m'ont accueilli au sein des équipes Imara et Tao de l'Inria m'offrant un cadre favorable où effectuer ma thèse.

Jean-Michel Loubes et Didier Piau ont accepté de consacrer le temps nécessaire à l'évaluation de ce manuscrit, je les en remercie vivement. Je remercie aussi Nikolas Geroliminis et Jean-Patrick Lebacque pour leur participation à ce jury de thèse. J'adresse tous mes remerciements à Arnaud de La Fortelle qui a dirigé ces travaux.

J'ai eu la chance d'être suivi par deux encadrants, Cyril Furtlehner et Jean-Marc Lasgouttes, qui m'ont consacré plus de temps qu'un thésard ne peut en espérer et qui ont guidé mes pas plus ou moins chancelants au cours de ces trois ans et demi. Au quotidien Jean-Marc [1] a patiemment réussi à me transmettre – entre autres – le virus typographique, les expressions régulières et le principe de choix aléatoire en absence d'information. Il n'aura cependant pas tout à fait réussi à me faire passer le goût des notes de bas de page. Cyril a enduré sans complaintes (en tout cas envers moi) mon scepticisme face à son intuition – généralement valide – et a provoqué – contre mon gré – une certaine admiration de ces physiciens qui font des maths. Je vous adresse à nouveau un immense « Merci ! ».

J'ai apprécié les nombreuses (et longues) discussions à propos du travail entamé (et à venir) avec Yufei Han. Pour cela, ainsi que pour avoir supporter mes ronchonnements lors des périodes de recherche de bugs, je lui dis merci. Arnaud Lewden a consacré du temps à tenter de rationaliser ma manière de coder ; je l'en remercie. Je remercie aussi toutes les personnes qui ont contribué, de près ou de loin, à ces travaux ; je pense notamment à la joyeuse bande de Travesti.

Guy Fayolle a diminué la probabilité d'inondation de mon bureau en refermant derrière moi ma fenêtre lors de mes nombreux oublis. Il fût de plus le guide d'une randonnée forestière inoubliable, pour tout cela merci ! Je remercie les autres personnes que j'ai côtoyées pour des périodes diverses au sein de ce fameux bâtiment 20, notamment Philippe.

1. dont le prénom ne peut s'abréger autrement qu'en « J.-M. ».

Je souhaite aussi adresser mes remerciements aux personnes qui m'ont permis de prendre plaisir à enseigner (Arnaud de La Fortelle aux Mines, Gilles Faÿ et Nicolas Vayatis à Centrale).

Une thèse, c'est long par moment et il vaut mieux être bien entouré hors du cadre académique. Je souhaite maintenant remercier les autres personnes qui m'ont accompagné toutes ces années et pour encore de nombreuses années j'espère. La grande communauté des sudistes de Paris : Colin zi Mousse au Chocolat master [2], Claire pour qui je place Les Sables dans le sud, Benjamin qui fait tout son possible pour valoriser mon revers au tennis, Fred qui ferait bien de prendre exemple sur Benjamin, Frank et Benoît qui se laissent humilier au badminton, Paul qui les venge et tous les autres que je vois moins souvent. Merci à tous d'avoir servi de cobayes à nos tests culinaires ; le jour où l'on ouvre une maison d'hôtes vous serez conviés (avec une réduction de 20% !). La fabuleuse colloc' de Supélec - mes p'tits Ju et Charles, Yan [ian] et le disparu Didi - qui s'est un peu éparpillée. Je remercie aussi mes parents qui m'ont toujours poussé, peut-être trop fort vu le résultat, à faire des études. Sylvain je te vois moins souvent mais je ne t'oublie pas pour autant, et d'ailleurs comment oublier les nuits tortues ninja ?

Je pense très fort à mon oncle Luc dont j'aurais énormément apprécié la présence et qui, je pense, aurait lui aussi aimé être présent pour la conclusion de ma thèse. La vie en a malheureusement décidé autrement.

Pour finir, la plus importante, celle sans qui je ne serais encore qu'un mangeur de pâtes. Julie tu crois en moi pour deux, tu m'as soutenu dans tous les moments difficiles et tu as enduré les moments euphoriques (ce fût peut-être le plus dur ?). Pour cela et pour toute la joie et la tendresse que tu apportes dans ma vie je te remercie.

2. ouais bon j'ai pas trouvé de surnom qui soit à la hauteur pour une vengeance mais tu ne perds rien pour attendre !

Table des matières

Notations

Graphes	
$\mathcal{G} = (\mathbb{V}, \mathbb{E})$	graphe non orienté.
$\mathbb{V}$	ensemble de sommets.
$\mathbb{E}$	ensemble d'arcs non orientés.
∂i	ensemble des voisins du sommet i.
d_i	degré (nombre de voisins) d'un sommet i.
$\mathcal{H} = (\mathbb{V}, \mathbb{F})$	hypergraphe.
$\mathbb{F}$	ensemble d'arcs d'un hypergraphe ou de sous-parties de $\mathbb{V}$.
X_i	variable associée au sommet i.
$\mathbf{X}_A$	vecteur des variables associée à l'ensemble de sommets A.
$\mathbf{X}$	$\mathbf{X}_{\mathbb{V}}$.

Divers	
$\sum_{\mathbf{x}\setminus x_i}, \int_{\mathbf{x}\setminus x_i}$	somme, intégrale sur les valeurs de $\mathbf{x}$ exceptées celles de x_i.
$\perp\!\!\!\perp$	indépendance au sens probabiliste.
$\lvert A \rvert$	cardinal de l'ensemble A.
$X \sim dF$	« La variable aléatoire X est de loi dF. »
$X \succeq Y$	Ordre stochastique entre les variables X et Y.

Chapitre 1

Introduction

1.1 Contexte de l'étude

Le sujet étudié dans cette thèse étant inspiré d'un problème pratique lié au trafic routier, nous allons en faire une brève description avant de l'abstraire comme un problème d'inférence dans la partie suivante. Les chapitres suivants de la thèse seront ensuite consacrés à l'étude de différents aspects de ce problème.

Le problème de régulation des réseaux de transport a vu son importance croître avec les phénomènes d'urbanisation et d'exode rural ; il reste critique malgré la mise en place progressive de politiques de transports publics. Un premier pas vers cette régulation consiste à être capable de fournir aux utilisateurs, et aux autorités, une information en temps réel sur l'état de congestion du réseau ainsi que sur son évolution. Plusieurs types d'informations caractérisent l'état de congestion d'un réseau. De manière non exhaustive, on peut citer la densité de véhicules, leur vitesse moyenne ou *les temps de trajet sur les différents axes du réseau*. On considérera dans ces travaux que c'est cette dernière information qui caractérise l'état du réseau ; c'est en général l'information qui intéresse le plus les utilisateurs de ce réseau.

La méthode classique pour recueillir des données de trafic routier est l'installation d'une boucle magnétique sous la chaussée d'un axe de circulation qui compte le nombre de véhicules y circulant par unité de temps. Il existe d'autres méthodes mais elles impliquent toutes l'installation de capteurs fixes dénombrant les véhicules circulant à un point du réseau. Dans le cas d'un réseau de transport urbain ou péri-urbain, l'installation de ce type de capteurs sur l'ensemble des axes de circulation est techniquement et financièrement très difficile à mettre en œuvre du fait du nombre d'axes. Le coût d'installation d'une boucle magnétique est estimé à 12 500€, ceci ne tenant pas compte des frais de maintenance. En outre, une fois mis en place, un tel système n'offre aucune souplesse, les points de mesures étant figés.

L'émergence et la diffusion massive de systèmes de positionnement dans les véhicules remettent en cause cette approche classique de collecte de données. On propose ici d'étudier l'utilisation qui peut être faite de données provenant d'une flotte de véhicules utilisateurs du réseau. Ces véhicules fournissent, en temps réel, une information sur l'état du trafic rencontré. La nature des données est différente de celles obtenues par comptage. On passe en effet d'une vision eulérienne du trafic, où l'on observe le trafic à partir d'un point de mesure fixe, à une vision lagrangienne, où l'on suit les véhicules au cours du temps. Le point de vue lagrangien est plus adapté à la mesure de temps de trajet alors que le point de vue eulérien est plus adapté à la mesure de débit. Une collecte de données basée sur les véhicules et non plus sur l'infrastructure a déjà été mise en place sur la région de San Francisco dans le cadre du projet Mobile Millenium [75]. Les données collectées sont dans ce cas des traces GPS provenant de téléphones mobiles [45]. Notons cependant qu'obtenir des temps de trajet à partir de traces GPS nécessite de retrouver les chemins empruntés par les véhicules, ce qui peut être difficile, en particulier en zone urbaine où les axes peuvent être courts et proches les uns des autres (voir les travaux de Hunter *et al.* [49] à ce sujet).

L'origine des méthodes étudiées ici remonte au projet React [24, 31, 35, 36] qui a fait l'objet d'un dépôt de brevet [60]. Pour poursuivre cette étude, les équipes de LaRA [1] se sont impliqués dans deux projets Pumas [83] et Travesti [94], le second finançant cette thèse.

Pumas est un projet expérimental dont l'objectif était la mise en place d'une flotte de 1 000 véhicules spécifiquement équipés pour fournir une information sur l'état du trafic rencontré sur l'agglomération de Rouen (CREA). Ce projet a été une source d'inspiration du sujet étudié ici mais, pour diverses raisons, le nombre de véhicules obtenus a finalement été beaucoup plus faible que prévu et nous n'avons pu obtenir suffisamment de données exploitables. La taille de la flotte peut paraître assez faible par rapport à ce que l'on peut espérer en utilisant des téléphones mobiles équipés d'un système GPS comme source de données. Cependant, il est courant d'estimer que la connaissance de 5 à 10% du trafic permet une bonne estimation de l'état global. L'approche Pumas ne se situe donc pas dans la mouvance « Big Data » qui consiste à traiter une masse de données disponibles à peu de frais. En contrepartie de la difficulté du déploiement de la flotte, celle-ci nous fournit des données plus riches. Les véhicules sont par exemple équipés d'une cartographie embarquée et sont donc en mesure de nous fournir les temps de parcours le long de leur itinéraire. Les données obtenues sont donc les temps de parcours des axes successifs de l'itinéraire des véhicules sondes. On se contentera ici de mesurer les temps de trajet par paires de lien, afin de procéder à une approche de type markovienne. Les seules dépendances que l'on pourra observer directement

1. LaRA est un groupe de recherche formé de l'équipe projet Imara de l'Inria et du Caor des Mines-ParisTech.

sont donc celles entre voisins du graphe physique.

Le projet TRAVESTI[2] est le pendant théorique de PUMAS qui s'intéresse au problème de modélisation de réseaux de grande taille dans le but de fournir des prédictions de leur comportement. En particulier, la partie du projet qui nous intéresse dans ces travaux concerne la prédiction en temps réel du comportement d'un réseau routier à partir d'informations distribuées.

Dans la suite, on s'appliquera à trouver des méthodes qui ne sont pas spécifiques au trafic routier. On supposera seulement que les observations dont on dispose correspondent aux hypothèses suivantes :

- *Les observations sont (très) incomplètes.* On n'observera jamais simultanément l'état intégral du réseau mais seulement un faible pourcentage des variables de celui-ci.
- *L'identité des variables observées n'est pas contrôlée.* La flotte de véhicules sondes se déplace selon ses objectifs propres qui ne coïncident pas avec l'observation que l'on souhaite réaliser du réseau.

Les deux grands types d'approche pour la prédiction de trafic routier sont les méthodes dirigées par les données (*i.e.* les méthodes de type régression) et celles basées sur un modèle de trafic. La nature des observations rendent le problème non traitable par les méthodes classiques de régression qui supposent le nombre et l'identité des observations fixés. D'une manière générale, il est difficile d'utiliser des méthodes purement dirigées par les données dans un contexte où l'on ne possède aucune observation complète du réseau.

Les modèles de trafic routier sont nombreux et diffèrent selon l'échelle considérée ; pour un panorama des modèles standards voir [20, 57]. Les modèles macroscopiques, qui nous intéressent ici, sont en général basés sur de la dynamique des fluides *à l'équilibre*, et particulièrement adaptés au cas du trafic sur autoroute. Cette hypothèse n'est cependant pas adaptée au cas d'un réseau urbain. En effet la topologie du réseau est alors plus complexe et l'état du trafic sur un segment entre deux intersections est par nature stochastique et hors d'équilibre, du fait de la présence de feux de circulation ou d'un faible nombre de véhicules par segment.

L'approche proposée ici est celle d'une modélisation probabiliste de l'état du trafic, et les deux problèmes conjoints que l'on va chercher à résoudre sont :

- *Comment définir un modèle probabiliste à partir d'observations (très) incomplètes du réseau ?*
- *Partant d'observations distribuées aléatoirement, quelle méthode de prédiction temps-réel du comportement du réseau peut-on proposer ?*

Ces deux questions sont très fortement liées, on a en effet tout intérêt à définir un modèle compatible avec la méthode de prédiction choisie.

2. Acronyme de « TRAffic Volume Estimation by Spatio-Temporal Inference ».

1.2 Modélisation probabiliste

On commence ici par formaliser le problème décrit dans la section précédente. Soit $\mathbf{X} \stackrel{\text{def}}{=} \{X_i, i \in \mathbb{V}\} \in \mathbb{R}^{|\mathbb{V}|}$ un vecteur aléatoire. On se propose de répondre aux deux questions suivantes :

(i) Quel modèle probabiliste de $\mathbf{X}$ peut-on proposer à partir d'observations de paires ? Plus précisément, on suppose que pour toute paire (i, j) d'un sous-ensemble $\mathbb{E}$ de $\mathbb{V}^2$ on a $N_{i,j}$ réalisations indépendantes du couple (X_i, X_j)

$$\left\{(X_i = x_i^k, X_j = x_j^k)\right\}_{k \in \{1,\dots,N_{i,j}\}}. \tag{1.1}$$

(ii) A partir d'informations partielles $\mathbf{X}_{\mathbb{V}^*} = \mathbf{x}_{\mathbb{V}^*}$, dont l'identité $\mathbb{V}^*$ n'est pas contrôlée et change d'une instance à l'autre, quelle information peut-on obtenir sur les variables aléatoires non observées $\mathbf{X}_{\mathbb{V}\setminus\mathbb{V}^*}$?

Comme on l'a déjà suggéré dans l'introduction générale, les réponses apportées à ces deux questions sont très fortement liées et l'on prend ici le parti de les traiter conjointement. Ce problème diffère des approches classiques, à la fois du fait de la nature des observations (très incomplètes) permettant de construire le modèle mais aussi du fait de la variabilité des observations $\mathbb{V}^*$ dont l'identité n'est pas connue à l'avance. On est contraint de passer par une étape de choix et d'estimation d'un modèle du vecteur aléatoire $\mathbf{X}$. En effet, les méthodes sans modèles, telle que les algorithmes du type « k plus proches voisins », ne peuvent fonctionner avec des observations aussi partielles du vecteur $\mathbf{X}$. La question (ii), lorsque $\mathbb{V}^*$ est fixé, correspond exactement à un problème de régression, c'est à dire que l'on cherche à déterminer tout ou partie des lois conditionnelles des variables X_i non observées. L'ensemble des variables observées $\mathbb{V}^*$ n'étant pas figé, il faudrait cependant calibrer un modèle de régression par valeur de $\mathbb{V}^*$ ce qui conduirait à un nombre exponentiel de modèles et n'est donc pas envisageable. Intéressons nous à présent au choix du modèle probabiliste du vecteur $\mathbf{X}$.

La structure des observations fait apparaître de manière sous-jacente un modèle de variables aléatoires sur un graphe $G = (\mathbb{V}, \mathbb{E})$; les graphes étant la représentation la plus commune des structures de dépendance, comme l'explique Pearl [82, p. 81-82]. Pour mettre cela en évidence d'une autre manière, supposons que les variables sont à valeurs discrètes et que les observations jointes (1.1) sont résumées sous la forme de moments empiriques $\widehat{\boldsymbol{\alpha}}$ correspondant à des fonctions test $\mathbf{r}$ choisies ou subies. Dans ce cas, on recherche une distribution $p_{\boldsymbol{\sigma}}$ du vecteur $\boldsymbol{\sigma}$ vérifiant les contraintes suivantes

$$\begin{cases} \mathbb{E}_{p_{\boldsymbol{\sigma}}}[r_i(\sigma_i)] = \sum_{\mathbf{s}} p_{\boldsymbol{\sigma}}(\mathbf{s}) r_i(s_i) = \widehat{\alpha}_i, \forall i \in \mathbb{V}, \\ \mathbb{E}_{p_{\boldsymbol{\sigma}}}[r_{ij}(\sigma_i, \sigma_j)] = \sum_{\mathbf{s}} p_{\boldsymbol{\sigma}}(\mathbf{s}) r_{ij}(\sigma_i, \sigma_j) = \widehat{\alpha}_{ij}, \forall (i,j) \in \mathbb{E}, \end{cases} \tag{1.2}$$

pour des fonctions $\mathbf{r}$ données et $\widehat{\alpha}_i, \widehat{\alpha}_{ij} \in \mathbb{R}$. Le choix le moins discriminant pour $p_{\boldsymbol{\sigma}}$, vérifiant ces contraintes, est obtenu grâce au principe de maximisation

de l'entropie de Jaynes [51]. Cette maximisation sous contraintes conduit à considérer le lagrangien suivant :

$$\mathcal{L}(p_{\boldsymbol{\sigma}}, \gamma, \boldsymbol{\lambda}) = -\sum_{\mathbf{s}} p_{\boldsymbol{\sigma}}(\mathbf{s}) \ln p_{\boldsymbol{\sigma}}(\mathbf{s}) + \sum_{i \in \mathbb{V}} \lambda_i \left(\sum_{\mathbf{s}} p_{\boldsymbol{\sigma}}(\mathbf{s}) r_i(s_i) - \widehat{\alpha}_i \right) + \sum_{(i,j) \in \mathbb{E}} \lambda_{ij} \left(\sum_{\mathbf{s}} p_{\boldsymbol{\sigma}}(\mathbf{s}) r_{ij}(s_i, s_j) - \widehat{\alpha}_{ij} \right) + \gamma \left(\sum_{\mathbf{s}} p_{\boldsymbol{\sigma}}(\mathbf{s}) - 1 \right),$$

dont les points stationnaires ($\frac{\partial \mathcal{L}(p_{\boldsymbol{\sigma}}, \gamma, \boldsymbol{\lambda})}{\partial p_{\boldsymbol{\sigma}}(\mathbf{s})} = 0$) fournissent la forme de la distribution :

$$p_{\boldsymbol{\sigma}}(\mathbf{s}) = \exp\left(\sum_{i \in \mathbb{V}} \lambda_i r_i(s_i) + \sum_{(i,j) \in \mathbb{E}} \lambda_{ij} r_{ij}(s_i, s_j) + \gamma - 1 \right). \tag{1.3}$$

Les multiplicateurs de Lagrange $\boldsymbol{\lambda}$ et γ sont respectivement obtenus en imposant les contraintes (1.2) et $\sum_{\mathbf{s}} p_{\boldsymbol{\sigma}}(\mathbf{s}) = 1$. Une distribution de la forme (1.3) correspond au cas d'un champ markovien aléatoire avec interactions de paires, comme on le verra dans la section suivante (théorème 1.2).

Regardons maintenant le cas plus simple de la forme (1.3). Il s'agit du cas de variables binaires $\boldsymbol{\sigma} = \{\sigma_i, i \in \mathbb{V}\} \in \{-1, 1\}^{|\mathbb{V}|}$ avec les fonctions $\mathbf{r}$ qui sont tout simplement $r_i(s_i) = s_i$ et $r_{ij}(s_i, s_j) = s_i s_j$. La distribution de $\boldsymbol{\sigma}$ est alors de la forme suivante :

$$\mathbb{P}_{\boldsymbol{\sigma}}(\boldsymbol{\sigma} = \mathbf{s}) = \frac{1}{Z} \exp\left(- \sum_{(i,j) \in \mathbb{E}} J_{ij} s_i s_j - \sum_{i \in \mathbb{V}} h_i s_i \right),$$

qui correspond à un modèle d'Ising en physique statistique. On trouve une description de ces modèles d'Ising dans l'ouvrage de Baxter [8]. Le cas des modèles d'Ising ferromagnétiques uniformes ($J_{ij} = J > 0$) a été étudié de manière extensive. Il admet une transition de phase, pour des structures de graphes suffisamment connectés, vis-à-vis de la valeur de la constante de couplage J. Pour des faibles valeurs de J, seul un état désordonné apparaît, correspondant aux fluctuations des variables σ_i autour de leur valeur moyenne. Au contraire, pour des fortes valeurs de J, les variables tendent toutes à se trouver dans le même état et deux états macroscopiques équiprobables existent : $\boldsymbol{\sigma} = \mathbf{1}$ et $\boldsymbol{\sigma} = -\mathbf{1}$. Revenons un instant au point de vue trafic. Ces deux états correspondent au cas où presque tous les liens sont fluides et à celui où presque tous les liens sont congestionnés. Les configurations réelles de congestion sont bien sûr plus complexes et les modèles d'Ising inhomogènes [3], appelés verres de spin [72] en physique statistique, permettent d'obtenir des superpositions d'états du type modèle de Hopfield [48, 90, 4]. D'un point de vue physique,

3. les coefficients J_{ij} sont alors différents et peuvent être négatifs.

ce type de modèle est satisfaisant. On voit en effet apparaître naturellement la propagation de la congestion de proche en proche dans le réseau, via les interactions de paires.

Les modèles de la forme (1.3) s'adaptent particulièrement bien à l'estimation à partir d'observations de paires (1.1), puisque les paramètres λ_{ij} correspondent aux interactions entre ces mêmes paires de variables.

La section suivante est dédiée à la définition formelle de la notion de champ markovien aléatoire qui généralise les distributions de la forme (1.3). Pour une description plus extensive de ces modèles, on réfère à l'ouvrage de Kindermann *et al.* [56]. En particulier, le théorème 1.2 (Hammersley-Clifford) nous permettra de simplifier leur propos en donnant une forme générique à la distribution jointe du vecteur aléatoire $\mathbf{X}$.

1.3 Champ markovien aléatoire

Commençons tout d'abord par énoncer trois propriétés de Markov qui formalisent l'idée intuitive de dépendance entre variables basée sur un graphe non orienté $\mathcal{G}$.

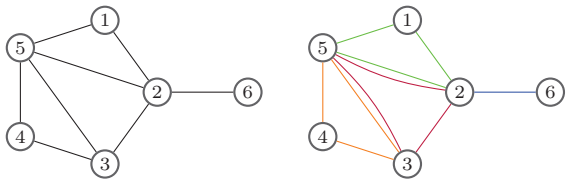
Définition 1.1. *Soit* $\mathbf{X} \stackrel{def}{=} \{X_i, i \in \mathbb{V}\}$ *un vecteur aléatoire et* $\mathcal{G} = (\mathbb{V}, E)$ *un graphe non orienté.* $\mathbf{X}$ *vérifie une propriété de Markov relativement au graphe* $\mathcal{G}$ *s'il vérifie l'une des trois propriétés suivantes :*

- (M1) $\forall (i,j) \notin E,\ X_i \perp\!\!\!\perp X_j \mid \mathbf{X}_{\mathbb{V}\setminus\{i,j\}}$ *: dès lors que deux variables ne sont pas voisines dans* $\mathcal{G}$ *elles sont indépendantes conditionnellement à toutes les autres variables.*
- (M2) $\forall i \in \mathbb{V},\ X_i \perp\!\!\!\perp \mathbf{X}_{\mathbb{V}\setminus\partial i} \mid \mathbf{X}_{\partial i}$ *: conditionnellement, à son voisinage une variable est indépendante de toutes les autres.*
- (M3) $\forall A, B \subset \mathbb{V}$ *et* $\forall C \subset \mathbb{V}$ *tel que* C *sépare* A *et* B *dans* $\mathcal{G}$, $\mathbf{X}_A \perp\!\!\!\perp \mathbf{X}_B \mid \mathbf{X}_C$ *: deux sous-ensembles de variables sont indépendants conditionnellement à un sous-ensemble les séparant.*

On parlera de champs markoviens aléatoires relativement à $\mathcal{G}$ lorsque l'on considère des distributions du vecteur $\mathbf{X}$ pour lesquelles ces trois propriétés de Markov sont équivalentes. C'est l'objet de la proposition suivante :

Proposition 1.1. *Pour toute loi de probabilité* $\mathbb{P}$ *admettant une densité continue et strictement positive par rapport à une mesure, les propriétés de Markov* (M1)*,* (M2) *et* (M3) *sont équivalentes ; un vecteur aléatoire* $\mathbf{X}$ *vérifiant une de ces trois propositions sera alors appelé* champ markovien aléatoire relativement à $\mathcal{G}$*. Dans le cas de variables aléatoires discrètes, la stricte positivité de la loi suffit donc.*

Démonstration. Voir Lauritzen [61, pages 32-35]. L'hypothèse sur les distributions peut même être partiellement relâchée (Lauritzen [61, page 29]). □



$$\mathbb{P}_{\mathbf{X}}(\mathbf{X} = \mathbf{x}) = \Psi_a(x_1, x_2, x_5)\Psi_b(x_2, x_3, x_5)\Psi_c(x_3, x_4, x_5)\Psi_d(x_2, x_6)$$

FIGURE 1.1 : Un exemple d'application du théorème d'Hammerlsey-Clifford. Les cliques maximales sont $a = \{1, 2, 5\}, b = \{2, 3, 5\}, c = \{3, 4, 5\}$ et $d = \{2, 6\}$, d'où la factorisation.

Dans la suite de la thèse, on supposera se situer dans ce cadre, ce qui n'est pas limitant mais permet de s'affranchir de difficultés techniques. On présente maintenant le théorème d'Hammerlsey-Clifford [42] qui est central dans l'étude des champs markoviens aléatoires.

Théorème 1.2 (Hammersley-Clifford)**.** $\mathbf{X}$ *est un champ markovien aléatoire, relativement au graphe* $\mathcal{G} = (\mathbb{V}, E)$*, si et seulement si il existe des fonctions* Ψ *positives telles que*

$$\mathbb{P}_{\mathbf{X}}(\mathbf{X} = \mathbf{x}) = \prod_{a \in C_{\mathcal{G}}} \Psi_a(\mathbf{x}_a), \tag{1.4}$$

où $C_{\mathcal{G}}$ *est l'ensemble des cliques maximales de* $\mathcal{G}$*.*

Démonstration. Voir Lauritzen [61, pages 36-37] □

La factorisation (1.4) n'est évidemment pas unique et les fonctions Ψ n'ont d'autres contraintes que d'être strictement positives ; en particulier, Ψ_a n'est pas une mesure de probabilité de $\mathbf{X}_a$. La figure 1.1 fournit une illustration pour un exemple de graphe de dépendance.

Une conséquence importante de ce théorème est qu'il n'est pas nécessaire de connaître le graphe de dépendance $\mathcal{G}$ pour savoir si $\mathbf{X}$ est un champ markovien. Il suffit d'être capable de factoriser sa loi de probabilité. Dans la suite, on s'intéressera en général au graphe de dépendance $\mathcal{G}$ à travers la forme de la factorisation.

Avant de conclure ce chapitre en annonçant le plan du mémoire, il est nécessaire de dire quelques mots sur l'autre grande classe de modèles de variables aléatoires sur un graphe. Il s'agit des réseaux bayésiens que l'on a décidé de ne pas considéré ici. On trouve une description extensive de ces modèles par Jensen [53] ou Pearl [82]. Les réseaux bayésiens diffèrent des champs markoviens aléatoires en cela que leur structure de dépendance est basée sur un graphe orienté. Du fait de l'aspect non orienté, Pearl montre que les champs markoviens sont moins à même de modéliser les indépendances conditionnelles entre

variables [82, chapitre 3]. D'une manière générale, les réseaux bayésiens permettent de modéliser des structures de dépendance plus riches que les champs markoviens aléatoires mais ceci a évidemment un coût : l'orientation du graphe nécessite d'introduire une expertise sur la nature des dépendances entre variables. Le principe de maximisation de l'entropie, pour le choix de la loi (1.3) développé dans la section 1.2, indique que choisir la loi de $\boldsymbol{\sigma}$ comme étant celle d'un réseau bayésien pour ces mêmes contraintes (1.2) implique l'ajout d'une information *a priori*, *i.e.* non contenue dans (1.2). On fait ici le choix de considérer qu'aucun *a priori* de ce type n'est disponible, même si dans le cas du trafic routier, celui-ci peut être fourni par l'ordre de parcours des axes. Notons cependant que la nature des réseaux bayésiens permet de se ramener aussi à une loi jointe de forme produit

$$\mathbb{P}_{\mathbf{X}}(\mathbf{X} = \mathbf{x}) = \prod_{i \in \mathbb{V}} \mathbb{P}(X_i = x_i | \mathbf{X}_{\mathrm{Pa}(i)} = \mathbf{x}_{\mathrm{Pa}(i)}),$$

où $\mathrm{Pa}(i)$ correspond à l'ensemble des parents de la variable X_i dans le réseau bayésien. L'étude conduite ici pourrait donc assez naturellement être adaptée au cas de réseaux bayésiens, pour peu que l'on puisse et souhaite introduire une expertise sur la nature des dépendances entre variables dans la définition du modèle probabiliste.

1.4 Structure du mémoire

Le choix de modélisation ayant été fait dans la section précédente, on considère maintenant que la loi du vecteur $\mathbf{X}$ s'exprime sous la forme produit suivante

$$\mathbb{P}_{\mathbf{X}}(\mathbf{X} = \mathbf{x}) = \prod_{a \in \mathbb{F}} \Psi_a(\mathbf{x}_a),$$

où $\mathbb{F}$ est un ensemble quelconque de parties de $\mathbb{V}$. En particulier, on n'impose pas que ces parties soient maximales, ce qui implique que l'on peut avoir $a, a' \in \mathbb{F}$ et $a \subsetneq a'$. D'après le théorème 1.2, cette factorisation de $\mathbb{P}_{\mathbf{X}}$ est équivalente à considérer que $\mathbf{X}$ est un champ markovien aléatoire. En effet il est toujours possible d'absorber les fonctions $\Psi_{a'}$ dans Ψ_a pour $a' \subset a$ pour obtenir une factorisation de la forme (1.4).

Répondre à la question (i) de la page 8 revient alors à estimer les fonctions Ψ_a. On explicitera plusieurs manières de procéder dans le cas de variables binaires au chapitre 5 et dans le cas de variables gaussiennes au chapitre 6.

Supposons maintenant que cette estimation ait été réalisée et intéressons nous à la question (ii). Lorsque l'on observe certaines variables $\mathbf{X}_{\mathbb{V}^*} = \mathbf{x}_{\mathbb{V}^*}$, on peut vouloir extraire 2 types d'information :

(MAP) [4] quel est l'état le plus probable étant donné les observations, celui qui maximise $\mathbb{P}_{\mathbf{X}}(\mathbf{X}_{\mathbb{V}\setminus\mathbb{V}^*} | \mathbf{X}_{\mathbb{V}^*} = \mathbf{x}_{\mathbb{V}^*})$?

4. pour « Maximum *A Posteriori* ».

(MAR) [5] quelles sont les distributions marginales $\mathbb{P}_{X_i}(\cdot|\mathbf{X}_{\mathbb{V}^*} = \mathbf{x}_{\mathbb{V}^*})$ des variables non observées $X_i, i \in \mathbb{V} \setminus \mathbb{V}^*$ sachant les observations ?

Dans ce travail, on cherchera à extraire l'information de type (MAR) ce qui consiste tout simplement à résoudre le problème de marginalisation suivant : pour chaque variable $i \notin \mathbb{V}^*$, on doit alors calculer une quantité du type

$$\mathbb{P}_{X_i}(X_i = x_i) = \int_{\mathbf{x} \setminus x_i} \mathbb{P}_{\mathbf{X}}(\mathbf{X} = \mathbf{x}). \tag{1.5}$$

Ce problème peut paraître trivial mais cette approche naïve est rapidement vouée à l'échec. En effet, supposons que le vecteur $\mathbf{X}$ soit formé de N variables binaires ; alors le terme de droite de (1.5) est une somme comportant 2^{N-1} termes et calculer toutes les lois marginales va conduire à considérer $N2^N$ termes. Cette approche n'est donc quasiment jamais utilisée en pratique. Il existe de nombreuses méthodes pour résoudre le problème (MAR), que ce soit de manière exacte ou de manière approchée. Les méthodes exactes les plus populaires sont certainement les algorithmes basés sur les arbres de jonction (Junction Tree ou Joint Tree en anglais), développés par Lauritzen et Spiegelhalter [62]. L'idée consiste à utiliser la distributivité entre la multiplication et l'addition et les indépendances induites par le graphe pour réaliser la sommation de manière optimisée. On trouvera une description extensive de ces méthodes et notamment de la construction de l'ordre de sommation dans l'ouvrage de Cowell *et al.* [21]. Pour ce qui est des méthodes approchées, on peut notamment citer les filtres particulaires décrits par Doucet et al. [27], les algorithmes de type « Markov chain Monte Carlo » (voir [39]) et enfin, l'algorithme Belief Propagation de Pearl [82] qui sera la méthode utilisée dans ces travaux et qui est l'objet du chapitre 2. Cet algorithme étant central dans ces travaux, on présente une étude extensive du rôle joué par la normalisation ainsi qu'un nouveau résultat sur la stabilité de ses points fixes au chapitre 3. Notons qu'une variante de Belief Propagation appelée algorithme « max-product » [105] permet d'extraire l'information (MAP) si on le souhaite.

La raison principale de ce choix est la rapidité de cet algorithme qui nous permet de réaliser la tâche (MAR) en temps réel, pour des réseaux de grandes tailles. À titre d'exemple, le réseau routier de l'agglomération de Rouen contient entre 10 000 et 15 000 variables. De plus, cet algorithme est, par nature, adapté à la parallélisation. On reviendra plus en détails sur cet algorithme et ses propriétés au chapitre 2.

L'algorithme Belief Propagation est classiquement défini pour deux classes de variables aléatoires : des variables à états discrets ou des variables gaussiennes. Les variables aléatoires $\mathbf{X}$ qui nous intéressent ne sont ni à états discrets, ni, *a priori*, gaussiennes ; on suppose seulement qu'elles sont à valeurs réelles. On verra au chapitre 5 qu'il est cependant possible de modéliser

5. pour « Marginalisation ».

ce vecteur $\mathbf{X}$ à travers un modèle d'Ising latent. Pour cela, on associe une variable binaire σ_i à chaque variable réelle X_i et les dépendances sont encodées à travers ces variables binaires. Les observations de variables X_i induisent une information plus faible que la connaissance de l'état de la variable σ_i. Pour accomplir la tâche (MAR), il est alors nécessaire de modifier l'algorithme Belief Propagation. C'est l'objet du chapitre 4, où l'on définit un algorithme qui permet d'imposer les marginales $\mathbb{P}_{\sigma_i}$ de certaines variables.

Le chapitre 6 propose une alternative à la méthode décrite dans les chapitres 4 et 5 en modélisant les données par un champ de Gauss-Markov. On donne alors une méthode d'approximation gaussienne du vecteur $\mathbf{X}$ permettant d'obtenir une meilleure compatibilité avec Belief Propagation.

L'organisation des différents chapitres est résumée sur la figure 1.2.

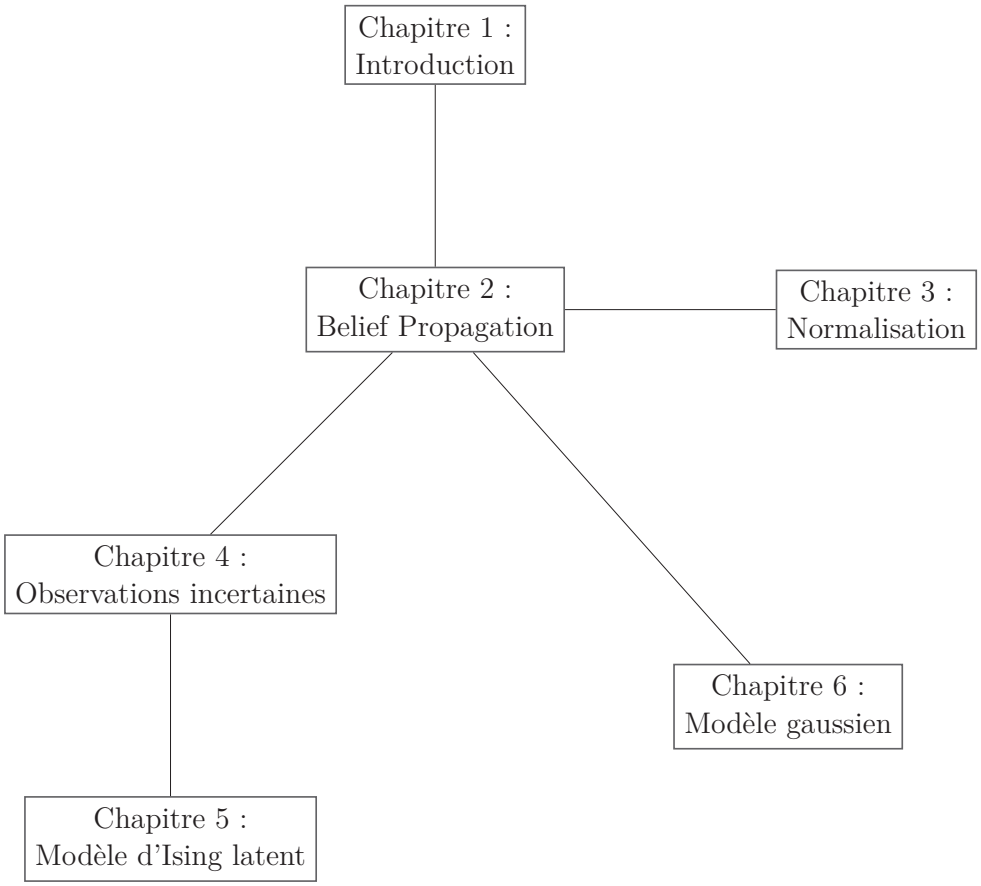


FIGURE 1.2 : Organisation des différents chapitres de la thèse.

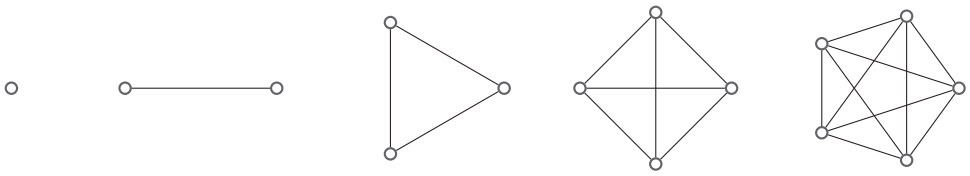


FIGURE 1.3 : Représentation des graphes complets $\mathcal{K}_n$ pour $n \in \{1, 2, 3, 4, 5\}$.

Annexe

1.A Éléments de théorie des graphes

Nous rappelons ici quelques concepts classiques de théorie des graphes, qui nous sont utiles, en particulier pour ce chapitre et le suivant.

Définition 1.2. *$\mathcal{G} = (\mathbb{V}, \mathbb{E})$ est un graphe complet si $\forall i \in \mathbb{V}, \partial i = \mathbb{V} \setminus \{i\}$. C'est-à-dire que chaque sommet de $\mathbb{V}$ est voisin de tous les autres dans $\mathcal{G}$.*

Les graphes complets à n sommets sont en général notés $\mathcal{K}_n$. On en a représentés quelques exemples sur la figure 1.3.

Définition 1.3. *$\mathcal{G}' = (\mathbb{V}', \mathbb{E}')$ est un sous-graphe de $\mathcal{G} = (\mathbb{V}, \mathbb{E})$ si $\mathbb{V}' \subset \mathbb{V}$, $\mathbb{E}' \subset \mathbb{E}$ et $\mathbb{E}' \subset \mathbb{V}' \times \mathbb{V}'$.*

La dernière inclusion consiste à ne pas sélectionner un arc reliant un sommet qui n'appartient pas au sous-graphe.

Définition 1.4. *Une clique d'un graphe $\mathcal{G}$ est un sous-graphe complet de $\mathcal{G}$. Une clique est dite maximale si elle n'est pas incluse dans une clique strictement plus grande.*

La figure 1.1 présente un exemple de détermination des cliques maximales d'un graphe comme illustration du théorème d'Hammerlsey-Clifford (théorème 1.2).

Définition 1.5. *Soit un graphe $\mathcal{G} = (\mathbb{V}, \mathbb{E})$ et deux sous-ensembles de sommets $A, B \subset \mathbb{V}$. On dit que $S \subset \mathbb{V}$ sépare A et B, si tout chemin d'un sommet de A vers un sommet de B passe par un sommet de S.*

Le concept sous-entendu de chemin correspond à l'intuition. Un chemin d'un sommet i vers un sommet j est une suite (éventuellement vide) d'arcs adjacents du graphe telle que l'origine du premier arc soit i et l'arrivée du dernier soit j. En particulier, il y a toujours un chemin (vide) de i à i.

Définition 1.6. *Un graphe $\mathcal{G} = (\mathbb{V}, \mathbb{E})$ est dit bi-partite si l'ensemble de ses sommets $\mathbb{V}$ s'écrit comme l'union ($A \cup B$) de deux sous-ensembles disjoints ($A \cap B = \emptyset$) tels que les arcs ne relient jamais deux éléments de A ou de B ($\mathbb{E} \subset A \times B$).*

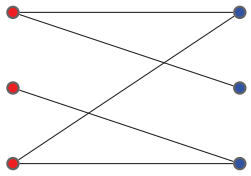


FIGURE 1.4 : Exemple de graphe bi-partite non orienté. Les arcs relient uniquement des sommets rouges à des sommets bleus.

Proposition 1.3. *Soit $\mathcal{G} = (\mathbb{V}, \mathbb{E})$ un graphe non orienté. Le nombre de cycles indépendants de $\mathcal{G}$, noté $\mathcal{C}$, est alors tel que*

$$\mathcal{C} - |\mathbb{E}| + |\mathbb{V}| - \kappa_{\mathcal{G}} = 0, \tag{1.A.1}$$

avec $\kappa_{\mathcal{G}}$ le nombre de composante connexe du graphe $\mathcal{G}$. Lorsque le graphe $\mathcal{G}$ est connexe, cette formule devient

$$\mathcal{C} = |\mathbb{E}| - |\mathbb{V}| + 1.$$

Démonstration. On présente ici l'idée intuitive de la preuve. Pour une preuve plus complète on réfère à l'ouvrage de Berge [9, pages 15-16].

Soit un graphe non orienté $\mathcal{G}$ à $|\mathbb{V}| = N$ sommets, la formule peut se démontrer par récurrence sur le nombre d'arcs $|\mathbb{E}|$. Lorsque $|\mathbb{E}| = 0$, on a alors $\kappa_{\mathcal{G}} = N$ et $\mathcal{C} = 0$ et la formule est donc bien vérifiée. Supposons maintenant que la formule (1.A.1) soit vraie pour tout graphe à $|\mathbb{E}| = n$ arcs. Lorsque l'on cherche à ajouter un arc entre deux sommets i et j, deux cas apparaissent :

- il existe déjà un chemin de i à j dans le graphe à n arcs. Dans ce cas, le nombre de cycles indépendants augmente d'une unité ;
- il n'existe pas de chemin entre i et j dans le graphe à n arcs. Dans ce cas, on a relié deux composantes connexes de ce graphe, diminuant leur nombre d'une unité.

Dans ces deux cas, la formule reste vraie après l'ajout d'un nouvel arc. □

Chapitre 2

L'algorithme Belief Propagation

Comme on l'a déjà évoqué dans le chapitre 1, l'algorithme Belief Propagation de Pearl [82] est un outil qui permet de calculer, en général de manière approchée, les distributions marginales d'une loi de probabilité de forme produit (2.1). Cette forme produit permet d'englober le cas des champs markoviens aléatoires ainsi que celui des réseaux bayésiens.

Après avoir introduit les graphes de facteurs, qui sont l'objet naturel sur lequel définir BP, on présente l'algorithme dans le cas de variables discrètes (section 2.2). La section 2.3 est consacrée à l'interprétation variationnelle de BP. Après une rapide revue des diverses approches généralisant BP, on conclut le chapitre par la présentation de l'algorithme dans le cas de variables gaussiennes (section 2.6).

2.1 Représentation par un graphe de facteurs

On oublie donc ici l'existence du graphe de dépendance et l'on considère un vecteur aléatoire $\mathbf{X}$ dont la loi de probabilité admet une factorisation de la forme

$$\mathbb{P}_{\mathbf{X}}(\mathbf{X} = \mathbf{x}) = \prod_{a \in \mathbb{F}} \Psi_a(\mathbf{x}_a), \tag{2.1}$$

où $\mathbb{F}$ est un ensemble quelconque de parties de $\mathbb{V}$. Contrairement au cas du théorème 1.2, on n'impose pas que ces parties soient maximales, c'est à dire que l'on peut avoir $a, a' \in \mathbb{F}$ et $a \subsetneq a'$.

L'objet naturel qui apparaît dans la factorisation (2.1) est l'hypergraphe $\mathcal{H} = (\mathbb{V}, \mathbb{F})$. On rappelle qu'un hypergraphe est la généralisation de la notion de graphe au cas où les arcs peuvent contenir un nombre quelconque de sommets ; on pourra se référer au livre de Berge [9, chapitre 18] pour plus de détails. La factorisation (2.1) consiste donc à associer une fonction Ψ_a à chaque

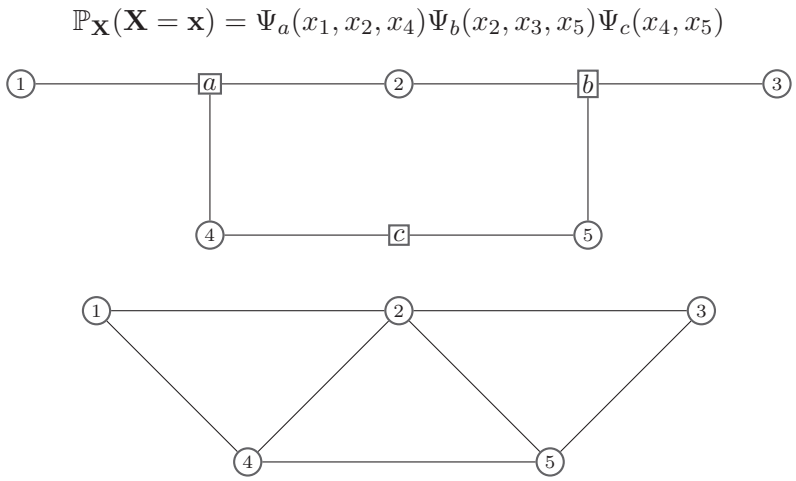


FIGURE 2.1 : De haut en bas : la structure de factorisation, le graphe de facteurs associé et le graphe de dépendance correspondant. Les sommets facteurs sont représentés par des carrés et les sommets variables par des ronds. On remarque que si l'on extrait les cliques maximales du graphe de dépendance, on obtient $\{\{1, 2, 4\}, \{2, 4, 5\}, \{2, 3, 5\}\}$ qui ne correspondent pas aux facteurs a, b et c. En effet, la structure de factorisation donne plus d'information sur la structure de dépendance que ne le fait le graphe de dépendance. La factorisation de départ implique en effet celle du théorème 1.2 appliqué au graphe de dépendance, l'inverse n'étant pas nécessairement vrai.

arc a de l'hypergraphe $\mathcal{H}$. Cette fonction Ψ_a dépend de l'état des variables appartenant à cet arc.

On introduit maintenant une représentation graphique classique et intuitive des hypergraphes : les graphes de facteurs, décrits par Loeliger [65].

Définition 2.1. *Le graphe de facteurs $\mathcal{G}$ basé sur l'hypergraphe $\mathcal{H} = (\mathbb{V}, \mathbb{F})$ est un graphe bi-partite non orienté $\mathcal{G} = (\mathbb{V} \cup \mathbb{F}, \mathbb{E})$. Les 2 sous-ensembles disjoints de sommets du graphe de facteurs $\mathcal{G}$ sont respectivement les sommets $\mathbb{V}$ de l'hypergraphe, appelés* variables*, et ses arcs $\mathbb{F}$, appelés* facteurs*. Il existe un arc entre un facteur $a \in \mathbb{F}$ et une variable $i \in \mathbb{V}$ dans le graphe de facteurs $\mathcal{G}$ dès lors que $i \in a$.*

On utilisera en général la notation $i \in a$ ou $a \ni i$ en lieu et place de $i \in \partial a$ et $a \in \partial i$ pour exprimer le fait que i et a sont voisins dans le graphe de facteurs $\mathcal{G}$. Cette notation doit être comprise au sens de l'hypergraphe sur lequel est basé le graphe de facteurs $\mathcal{G}$. La figure 2.1 présente un exemple de graphe de facteurs correspondant à une factorisation donnée.

On ne s'intéressera pas en général à la structure sous-jacente du graphe de dépendance mais on supposera que l'on connaît la structure de factorisation de $\mathbb{P}_{\mathbf{X}}$ ou, de manière équivalente, que le graphe de facteurs basé sur $\mathcal{H} = (\mathbb{V}, \mathbb{F})$ est connu.

2.2 Définition de l'algorithme

On présente ici l'algorithme Belief Propagation dans le cas de variables discrètes. Soit $\boldsymbol{\sigma} = \{\sigma_i, i \in \mathbb{V}\}$ un champ markovien de loi strictement positive, avec $\forall i \in \mathbb{V}, \sigma_i \in \{0, \dots, q_i - 1\}$. Le théorème 1.2 implique que la distribution de $\boldsymbol{\sigma}$ admet une factorisation de la forme

$$\mathbb{P}_{\boldsymbol{\sigma}}(\boldsymbol{\sigma} = \mathbf{s}) = \prod_{a \in \mathbb{F}} \Psi_a(\mathbf{s}_a) \prod_{i \in \mathbb{V}} \Phi_i(s_i). \tag{2.2}$$

La factorisation (2.2) n'est pas plus générale que (2.1) puisque l'on peut évidemment réabsorber les fonctions Φ_i dans les fonctions Ψ_a, pour $a \ni i$. On utilise (2.2) pour des raisons qui apparaîtront au chapitre 5.

Cet algorithme a été introduit par Pearl [82, chapitre 4] dans un contexte un peu différent : celui des réseaux bayésiens. La présentation s'appuyant ici sur le graphe de facteurs, elle diffère donc assez fortement de celle de Pearl. Lorsque le graphe de facteurs est arborescent, c'est à dire qu'il ne contient pas de cycles, Belief Propagation (BP) permet de calculer efficacement et de manière exacte les marginales $\mathbb{P}_{\sigma_i}$ et $\mathbb{P}_{\boldsymbol{\sigma}_a}$; BP agit alors comme une marginalisation optimale de (2.2).

BP est un algorithme de passage de messages, c'est à dire que les sommets voisins dans le graphe de facteurs s'échangent des messages pour propager, de proche en proche, l'information à travers le graphe. On définit deux types de messages, les messages $n_{i \to a}(s_i)$ envoyés par les variables aux facteurs et les messages $m_{a \to i}(s_i)$ envoyés par les facteurs aux variables. Les règles de calcul de ces messages sont les suivantes

$$n_{i \to a}(s_i) \stackrel{\text{def}}{=} \Phi_i(s_i) \prod_{b \ni i, b \neq a} m_{b \to i}(s_i), \tag{2.3}$$

$$m_{a \to i}(s_i) \propto \sum_{\mathbf{s}_{a \setminus i}} \Psi_a(\mathbf{s}_a) \prod_{j \in a, j \neq i} n_{j \to a}(s_j). \tag{2.4}$$

Le symbole $\propto$ signifie qu'il existe une constante Z_{ai}, indépendante de la valeur de s_i, telle que $m_{a \to i}(s_i)$ soit proportionnel au terme de droite. On reviendra rapidement sur la détermination de cette constante Z_{ai} dans la section suivante et plus en détails dans le chapitre 3. Le message (2.3) $n_{i \to a}$ envoyé par une variable i vers un facteur a est tout simplement le produit des contributions venant des facteurs voisins de i excepté celle venant de a. Le message (2.4) $m_{a \to i}$ envoyé par un facteur a vers une variable i consiste à sommer les produits

des contributions venant des variables voisines de a exceptée la contribution venant de i. Lorsque la convergence est atteinte, on peut alors calculer les différents « beliefs » qui sont une approximation des marginales $\mathbb{P}_{\sigma_i}$ et $\mathbb{P}_{\boldsymbol{\sigma}_a}$.

$$b_i(s_i) = \frac{1}{Z_i(\mathbf{m})} \Phi_i(s_i) \prod_{a \ni i} m_{a \to i}(s_i), \tag{2.5}$$

$$b_a(\mathbf{s}_a) = \frac{1}{Z_a(\mathbf{m})} \Psi_a(\mathbf{s}_a) \prod_{j \in a} n_{j \to a}(s_j), \tag{2.6}$$

avec $Z_i(\mathbf{m})$ et $Z_a(\mathbf{m})$ les constantes de normalisation telles que

$$\sum_{s_i} b_i(s_i) = 1, \qquad \sum_{\mathbf{s}_a} b_a(\mathbf{s}_a) = 1. \tag{2.7}$$

En particulier, ces beliefs sont compatibles

$$\sum_{\mathbf{s}_a \setminus s_i} b_a(\mathbf{s}_a) = b_i(s_i). \tag{2.8}$$

On peut montrer que lorsque le graphe de facteurs est arborescent, l'algorithme converge en un nombre fini d'itérations vers les marginales exactes [53, 82]. En pratique, dans ce cas, l'ordre optimal des mises à jour consiste à partir des feuilles de l'arbre et à remonter le long de celui-ci. Lorsque l'on atteint à nouveau les feuilles de l'arbre, l'algorithme a convergé. L'algorithme 1 page suivante est une implémentation formelle de BP. Cette implémentation est naïve et peut facilement être optimisée ; son unique but est d'illustrer le fonctionnement de l'algorithme.

BP est utilisé dans différents domaines et sous différentes formes depuis assez longtemps. Aji et McEliece [2] ont notamment montré que certains algorithmes de calcul rapide de transformée de Fourier (FFT), l'algorithme de Viterbi [96], BCJR [7] et d'autres, ne sont que des versions différentes d'un même principe utilisé dans BP. Dans tous les cas, ce principe consiste en une sommation optimisée et exacte sur un arbre. Le filtre de Kalman rentre aussi dans cette même catégorie [59]. Pour plus de détails sur ce principe sous-jacent, on réfère le lecteur à [2].

BP comme méthode d'approximation des marginales. Pearl [82] a suggéré d'utiliser son algorithme sur des graphes contenant des cycles. En effet, les formules de mise à jour (2.3)–(2.4) ne supposent pas une structure particulière de graphe. On peut donc décider de répéter ces mises à jour jusqu'à une éventuelle convergence. On reviendra sur le concept de convergence de l'algorithme BP dans le chapitre 3. L'algorithme ne converge pas toujours et il est même possible d'observer des phénomènes oscillatoires dans certains cas [79]. De plus, lorsque l'algorithme converge, les beliefs, calculés par (2.5) et (2.6), ne sont pas les vraies marginales de la loi (2.2). Il a cependant été

Algorithme 1 : *Belief Propagation* (BP)

Données : $\mathcal{H} = (\mathbb{V}, \mathbb{F})$, N_{max} et ε.
initialisation des messages $\mathbf{m}^{(0)}$ et $\mathbf{n}^{(0)}$;
$\delta \leftarrow +\infty$ et $n \leftarrow 0$;
tant que $\delta > \varepsilon$ *et* $n < N_{max}$ **faire**
  $\delta \leftarrow 0$;
  pour $i \in \mathbb{V}$ **faire**
    pour $a \ni i, s_i \in \{0, \dots, q_i - 1\}$ **faire**
      $n_{i\to a}^{(n+1)}(s_i) \leftarrow \Phi_i(s_i) \prod_{\substack{c \ni i \\ c \neq a}} m_{c\to i}(s_i)$;
  pour $a \in \mathbb{F}$ **faire**
    pour $i \in a$ **faire**
      pour $s_i \in \{0, \dots, q_i - 1\}$ **faire**
        $m_{a\to i}^{(n+1)}(s_i) \leftarrow \sum_{\mathbf{s}_{a\setminus i}} \Psi_a(\mathbf{s}_a) \prod_{j \in a} n_{j\to a}(s_j)$;
      $\mathbf{m}_{a\to i}^{(n+1)} \leftarrow \mathbf{m}_{a\to i}^{(n+1)} / \|\mathbf{m}_{a\to i}^{(n+1)}\|$;
      $\delta \leftarrow \max(\delta, \|\mathbf{m}_{a\to i}^{(n+1)} - \mathbf{m}_{a\to i}^{(n)}\|_\infty)$;
  $n \leftarrow n + 1$;
si $\delta < \varepsilon$ **alors**
  pour $i \in \mathbb{V}, s_i$ **faire**
    $b_i(s_i) = \frac{1}{Z_i(\mathbf{m})} \Phi(s_i) \prod_{a \ni i} m_{a\to i}(s_i)$;
  pour $a \in \mathbb{F}, \mathbf{s}_a$ **faire**
    $b_a(\mathbf{s}_a) = \frac{1}{Z_a(\mathbf{m})} \Psi_a(\mathbf{s}_a) \prod_{j \in a} n_{j\to a}(s_j)$;

observé expérimentalement [79] que ces beliefs $\mathbf{b}_i$ et $\mathbf{b}_a$ sont souvent de bonnes approximations des vraies marginales. Il est possible de calculer les corrections induites par les boucles du graphe. Celles-ci ont été introduites par Chertkov et Chernyak [18, 17] et permettent d'obtenir les vraies marginales à partir de ces approximations. Le calcul exact de ces corrections est cependant d'une complexité équivalente à la marginalisation exacte et en pratique, on envisagera donc seulement leur calcul approché comme le proposent Gómez *et al.* [40].

Quel ordre de mises à jour ? Le cas de réseaux arborescents ne pose pas de problème ; en partant des feuilles de l'arbre, un nombre fini (égal au diamètre du graphe) d'itérations suffit pour obtenir la convergence. Cependant les équations de mise à jour (2.3) et (2.4) ne définissent pas d'ordre. En pratique, deux méthodes sont le plus souvent utilisées : la mise à jour séquentielle (on met à jour tous les messages puis on recommence) ou la mise à jour impatiente (on met à jour les messages envoyés par un sommet dès que celui-ci

reçoit de nouvelles informations). Peu de travaux abordent les différences de convergence de l'algorithme pour ces deux types de mises à jour. D'une manière générale, il est plus aisé d'étudier théoriquement la convergence lorsque les mises à jour se font de manière séquentielles. Cependant Taga et Mase [89] montrent que la règle impatiente converge en général plus rapidement.

Mises à jour amorties Pour limiter les phénomènes d'oscillations, il est possible d'amortir les mises à jour. Le principe, valide pour toute suite de la forme $u_{n+1} = f(u_n)$, consiste à remplacer la mise à jour $u_{n+1} \leftarrow f(u_n)$ par $u_{n+1} \leftarrow (1-\varepsilon)f(u_n) + \varepsilon u_n \stackrel{\text{def}}{=} f_\varepsilon(u_n)$. On dit que $\varepsilon \in [0,1[$ est le niveau d'amortissement. L'effet stabilisant de ces mises à jour amorties est détaillé dans l'annexe 2.A.

2.3 L'approche variationnelle

C'est dans les travaux de Yedidia *et al.* [111] qu'apparaissent les arguments justifiant théoriquement l'approximation des marginales de (2.2) par les beliefs (2.5). Les points fixes de BP sont en effet des minima locaux d'un problème de minimisation que nous allons définir.

2.3.1 Définition de la fonction objectif

En physique statistique, l'énergie libre variationnelle F d'un système est définie comme la différence de l'énergie libre moyenne U et de l'entropie S. Plus précisément, on a :

$$F(\mathbf{b}) \stackrel{\text{def}}{=} U(\mathbf{b}) - S(\mathbf{b}),$$

$$U(\mathbf{b}) \stackrel{\text{def}}{=} \mathbb{E}_{\mathbf{b}}(E(\mathbf{s})) = \sum_{\mathbf{s}} b(\mathbf{s}) E(\mathbf{s}),$$

$$S(\mathbf{b}) \stackrel{\text{def}}{=} -\sum_{\mathbf{s}} b(\mathbf{s}) \log b(\mathbf{s}).$$

Dans le cas des mesures de Gibbs, l'énergie $E(\mathbf{s})$ d'un état $\boldsymbol{\sigma} = \mathbf{s}$ du système est reliée à sa loi de probabilité selon

$$\mathbb{P}_{\boldsymbol{\sigma}}(\boldsymbol{\sigma} = \mathbf{s}) = \frac{1}{Z} \exp\left(-E(\mathbf{s})\right).$$

Il est donc possible d'exprimer l'énergie libre variationnelle d'une distribution test $\mathbf{b}$ sous la forme suivante

$$F(\mathbf{b}) = F_H + D_{\mathrm{KL}}(\mathbf{b} || \mathbb{P}_{\boldsymbol{\sigma}}), \tag{2.9}$$

avec

$$F_H \stackrel{\text{def}}{=} -\log Z,$$

l'énergie libre de Helmholtz et

$$D_{\mathrm{KL}}(\mathbf{b}||\mathbb{P}_{\boldsymbol{\sigma}}) \stackrel{\text{def}}{=} \sum_{\mathbf{s}} b(\mathbf{s}) \log \frac{b(\mathbf{s})}{\mathbb{P}_{\boldsymbol{\sigma}}(\boldsymbol{\sigma} = \mathbf{s})}. \tag{2.10}$$

La fonction D_{KL} est connue sous le nom de divergence de Kullback-Leibler ou d'entropie relative. Cette divergence est toujours positive et ne s'annule que pour $\mathbf{b} = \mathbb{P}_{\boldsymbol{\sigma}}$. Il ne s'agit pas à proprement parler d'une distance car elle n'est pas symétrique et ne vérifie pas l'inégalité triangulaire. Cependant, on a $F(\mathbf{b}) \geq F_H$ et la minimisation de F sur l'ensemble des lois de probabilité nous conduit à récupérer exactement la loi $\mathbb{P}_{\boldsymbol{\sigma}}$. Dans le cas de N variables binaires, le nombre de paramètres de la distribution $\mathbf{b}$ est $2^N - 1$. Le problème d'optimisation est donc tout aussi difficile à résoudre que la marginalisation. On cherchera plutôt une famille de mesures pour $\mathbf{b}$ permettant :

1. de limiter le nombre de paramètres à estimer,
2. d'obtenir aisément les marginales $\mathbf{b}_i$, approximations de $\mathbb{P}_{\sigma_i}$.

Avant de continuer, on peut se poser la question du choix de D_{KL} comme fonction objectif, en particulier puisqu'il ne s'agit pas d'une distance. Son intérêt principal est le fait qu'elle est induite par la métrique associée à l'information de Fisher, comme le montre Amari [3]. C'est donc un objet très naturel en théorie de l'information. On a présenté ici la version variationnelle « physique » ; le point de vue de la théorie de l'information consiste en général à minimiser directement la divergence de Kullback-Leibler du fait de son lien avec la métrique de Fisher. Les deux approches sont strictement équivalentes dans notre cas.

Revenons au problème de minimisation et donnons un premier exemple de famille de loi de probabilité pour laquelle le problème de minimisation vérifie les contraintes 1 et 2. Le cas le plus simple est l'approche dite de champ moyen. Il s'agit de chercher à minimiser $F(\mathbf{b})$ (ou D_{KL}) avec $\mathbf{b} \in \mathcal{M}$ de la forme suivante :

$$\mathcal{M} \stackrel{\text{def}}{=} \left\{ b(\mathbf{s}) = \prod_{i \in \mathbb{V}} b_i(s_i) \,\middle|\, \forall i \in \mathbb{V}, \sum_{s_i} b_i(s_i) = 1 \text{ et } \forall s_i, b_i(s_i) \geq 0 \right\}.$$

La minimisation de $F(\mathbf{b})$, pour $\mathbf{b} \in \mathcal{M}$, devient réalisable car le nombre de paramètres à estimer est de l'ordre du nombre de variables. De plus, on obtient directement les marginales de $\mathbf{b}_i$ comme paramètres de $\mathbf{b}$. Cette approche naïve est cependant assez efficace, en particulier dans le cas d'interactions faibles entre variables. Une description complète de cette approche et de l'algorithme qui en découle est faite par Weiss [103].

2.3.2 L'approximation de Bethe

Les lois $\mathbf{b} \in \mathcal{M}$ correspondant à l'approximation des champs moyens supposent l'indépendance des variables, mais d'autres approximations plus fines

existent. L'approche qui va nous conduire à l'algorithme BP consiste à rechercher une loi $\mathbf{b}$ dans les familles suivantes :

$$\mathcal{B}_{\mathrm{D}}(\mathcal{H}) \stackrel{\text{def}}{=} \left\{ b(\mathbf{x}) = \prod_{a\in\mathbb{F}} \frac{b_a(\mathbf{s}_a)}{\prod_{j\in a} b_j(s_j)} \prod_{i\in\mathbb{V}} b_i(s_i) = \prod_{a\in\mathbb{F}} b_a(\mathbf{s}_a) \prod_{i\in\mathbb{V}} b_i(s_i)^{1-d_i} \right\},$$

$$\mathcal{B}_{\mathrm{C}}(\mathcal{H}) \stackrel{\text{def}}{=} \left\{ \mathbf{b} \;\middle|\; \begin{array}{l} \forall i \in \mathbb{V}, \sum_{s_i} b_i(s_i) = 1 \\ \forall a \in \mathbb{F}, \sum_{\mathbf{s}_a} b_a(x_a) = 1, \forall i \in a, \forall s_i, \sum_{\mathbf{s}_{a\setminus i}} b_a(\mathbf{s}_a) = b_i(s_i) \end{array} \right\}.$$

On parle d'approximation de Bethe [10] lorsque l'on choisit une distribution test $\mathbf{b} \in \mathcal{B}(\mathcal{H}) \stackrel{\text{def}}{=} \mathcal{B}_{\mathrm{D}}(\mathcal{H}) \cap \mathcal{B}_{\mathrm{C}}(\mathcal{H})$ et que l'on suppose l'égalité suivante :

$$\sum_{\mathbf{s}_{\mathbb{V}\setminus a}} b(\mathbf{s}) = b_a(\mathbf{s}_a), \forall a \in \mathbb{F}, \forall \mathbf{s}_a \tag{2.11}$$

Comme le montrent Yedidia et al. [111], l'hypothèse (2.11) n'est en général pas vérifiée pour $\mathbf{b} \in \mathcal{B}(\mathcal{H})$. Les contraintes de $\mathcal{B}_{\mathrm{C}}(\mathcal{H})$ n'imposent en effet pas que $\mathbf{b}_a$ et $\mathbf{b}_i$ soient les marginales d'une loi jointe, encore moins celles de $\mathbf{b}$. Pour plus de clarté, construisons un contre-exemple. Considérons 3 variables binaires $(\sigma_1, \sigma_2, \sigma_3)$ et les beliefs suivants

$$\mathbf{b}_{12} = \begin{bmatrix} 0{,}4 & 0{,}1 \\ 0{,}1 & 0{,}4 \end{bmatrix} \qquad \mathbf{b}_{23} = \begin{bmatrix} 0{,}4 & 0{,}1 \\ 0{,}1 & 0{,}4 \end{bmatrix} \qquad \mathbf{b}_{13} = \begin{bmatrix} 0{,}1 & 0{,}4 \\ 0{,}4 & 0{,}1 \end{bmatrix}.$$

MacKay a prouvé [67] qu'il n'existe pas de loi jointe de $(\sigma_1, \sigma_2, \sigma_3)$ qui ait les marginales de paires ci-dessus. Supposons qu'il existe une telle distribution $\{p_{000}, p_{001}, p_{010}, p_{011}, p_{100}, p_{101}, p_{110}, p_{111}\}$. On peut alors montrer que chacun de ces termes est inférieur à 0,1. De la marginale b_{12}, on déduit que $p_{010} \leq 0{,}1, p_{011} \leq 0{,}1, p_{100} \leq 0{,}1$ et $p_{101} \leq 0{,}1$. La marginale b_{23} nous donne de plus que $p_{110} \leq 0{,}1$ et $p_{001} \leq 0{,}1$ et enfin la marginale b_{13} fournit les deux inégalités manquantes $p_{111} \leq 0{,}1$ et $p_{000} \leq 0{,}1$. Donc, la somme de tous ces termes est inférieure à 0,8 et ne correspond pas à une distribution.

Cependant lorsque le graphe de facteurs basé sur $\mathcal{H}$ est acyclique, $\mathcal{B}(\mathcal{H})$ représente l'ensemble des distributions de champs markoviens admettant $\mathcal{H}$ comme factorisation, et l'hypothèse (2.11) est vérifiée.

Pour une loi $\mathbb{P}_{\boldsymbol{\sigma}}$ selon (2.2), avec une mesure $\mathbf{b} \in \mathcal{B}(\mathcal{H})$ la divergence $D_{\mathrm{KL}}(\mathbf{b}||\mathbb{P}_{\boldsymbol{\sigma}})$ se décompose sous la forme suivante :

$$\begin{aligned} D_{\mathrm{KL}}(\mathbf{b}||\mathbb{P}_{\boldsymbol{\sigma}}) &= \sum_{\mathbf{s}} b(\mathbf{s}) \log \frac{b(\mathbf{s})}{\mathbb{P}_{\boldsymbol{\sigma}}(\boldsymbol{\sigma} = \mathbf{s})} \\ &= \sum_{\mathbf{s}} b(\mathbf{s}) \left(\sum_{a\in\mathbb{F}} \log \frac{b_a(\mathbf{s}_a)}{\Psi_a(\mathbf{s}_a)} + \sum_{i\in\mathbb{V}} \log \frac{b_i(s_i)^{1-d_i}}{\Phi_i(s_i)} \right) \\ &= \sum_{a\in\mathbb{F}} \sum_{\mathbf{s}_a} \log \frac{b_a(\mathbf{s}_a)}{\Psi_a(\mathbf{s}_a)} \sum_{\mathbf{s}_{\mathbb{V}\setminus a}} b(\mathbf{s}) + \sum_{i\in\mathbb{V}} \sum_{s_i} \log \frac{b_i(s_i)^{1-d_i}}{\Phi_i(s_i)} \sum_{\mathbf{s}_{\mathbb{V}\setminus i}} b(\mathbf{s}). \end{aligned}$$

De plus, sous l'hypothèse (2.11) on obtient

$$D_{\mathrm{KL}}(\mathbf{b}||\mathbb{P}_{\boldsymbol{\sigma}}) = \sum_{a\in\mathbb{F}}\sum_{\mathbf{s}_a} b_a(\mathbf{s}_a)\log\frac{b_a(\mathbf{s}_a)}{\Psi_a(\mathbf{s}_a)} + \sum_{i\in\mathbb{V}}\sum_{s_i} b_i(s_i)\log\frac{b_i(s_i)^{1-d_i}}{\Phi_i(s_i)}$$
$$\stackrel{\text{def}}{=} F_{\mathrm{Bethe}}(\mathbf{b}). \tag{2.12}$$

On va donc chercher à résoudre le problème de minimisation

$$\min_{\mathbf{b}\in\mathcal{B}(\mathcal{H})} F_{\mathrm{Bethe}}(\mathbf{b}), \tag{2.13}$$

associé au lagrangien

$$\mathcal{L}(\mathbf{b},\boldsymbol{\lambda},\boldsymbol{\gamma}) = F_{\mathrm{Bethe}}(\mathbf{b}) + \sum_{\substack{i\in\mathbb{V},a\ni i\\ s_i}} \lambda_{ai}(s_i)\left(b_i(s_i) - \sum_{\mathbf{s}_a\setminus s_i} b_a(\mathbf{s}_a)\right)$$
$$-\sum_{i\in\mathbb{V}}\gamma_i\left(\sum_{s_i} b_i(s_i) - 1\right) - \sum_{a\in\mathbb{F}}\gamma_a\left(\sum_{\mathbf{s}_a} b_a(\mathbf{s}_a) - 1\right). \tag{2.14}$$

Les multiplicateurs de Lagrange $\boldsymbol{\lambda}$ et $\boldsymbol{\gamma}$ expriment la contrainte $\mathbf{b}\in\mathcal{B}_{\mathrm{C}}(\mathcal{H})$. L'appartenance de $\mathbf{b}$ à $\mathcal{B}_{\mathrm{D}}(\mathcal{H})$ ainsi que l'hypothèse (2.11) sont incluses dans l'expression (2.12) de $F_{\mathrm{Bethe}}(\mathbf{b})$.

2.3.3 Minima de F_{Bethe} et points fixes de BP

Les minima du problème (2.13) sont parmi les points stationnaires du lagrangien (2.14). Les conditions ${\partial\mathcal{L}(\mathbf{b},\boldsymbol{\lambda},\boldsymbol{\gamma})}/{\partial b_a(\mathbf{s}_a)} = 0$ et ${\partial\mathcal{L}(\mathbf{b},\boldsymbol{\lambda},\boldsymbol{\gamma})}/{\partial b_i(s_i)} = 0$ conduisent aux équations suivantes

$$\begin{cases} b_a(\mathbf{s}_a) = \Psi_a(\mathbf{s}_a)\exp\left(\sum_{i\in a}\lambda_{ai}(s_i) - \gamma_a - 1\right), \forall a\in\mathbb{F} \\ b_i(s_i) = \Phi_i(s_i)^{\frac{1}{1-d_i}}\exp\left(\frac{1}{d_i-1}\left(\sum_{a\ni i}\lambda_{ai}(s_i) - \gamma_i\right) - 1\right), \forall i\in\mathbb{V} \mid d_i\neq 1. \end{cases} \tag{2.15}$$

La condition $d_i \neq 1$ est due au fait que pour les sommets $i\in\mathbb{V}$ n'ayant qu'un seul voisin, on a ${\partial F_{\mathrm{Bethe}}(\mathbf{b})}/{\partial b_i(s_i)} = 0$ et $\mathbf{b}_i$ n'intervient donc pas dans la minimisation. $\mathbf{b}_i$ doit alors être obtenu via les contraintes $\sum_{\mathbf{s}_{a\setminus i}} b_a(\mathbf{s}_a) = b_i(s_i)$. Si l'on pose la paramétrisation (inversible) suivante

$$\lambda_{ai}(s_i) \stackrel{\text{def}}{=} \log n_{i\to a}(s_i) \stackrel{\text{def}}{=} \log\left(\Phi_i(s_i)\prod_{c\ni i, c\neq a} m_{c\to i}(s_i)\right), \tag{2.16}$$

on obtient les équations (2.5) et (2.6). Les beliefs $\mathbf{b}_a$ et $\mathbf{b}_i$ d'un point stationnaire de F_{Bethe} se factorisent donc exactement comme les beliefs d'un point

fixe de BP. Les conditions de compatibilité entre beliefs correspondent à la formule (2.4) de mise à jour. Pour constater cela, on exprime la compatibilité (2.8) à un point stationnaire de (2.14) :

$$\begin{aligned}\sum_{\mathbf{s}_{a\setminus i}} \Psi_a(\mathbf{s}_a) \prod_{j\in a} n_{j\to a}(s_j) e^{-\gamma_a} &= \Phi_i(s_i) e^{\frac{\gamma_i}{1-d_i}} \prod_{c\ni i} m_{c\to i}(s_i) \\ &= m_{a\to i}(s_i) n_{i\to a}(s_i) e^{\frac{\gamma_i}{1-d_i}}.\end{aligned}$$

En divisant par $n_{i\to a}(s_i)$, on obtient

$$m_{a\to i}(s_i) = \sum_{\mathbf{s}_{a\setminus i}} \Psi_a(\mathbf{s}_a) \prod_{j\in a\setminus i} n_{j\to a}(s_j) e^{-\gamma_a - \frac{\gamma_i}{1-d_i}},$$

qui est en général résumé par

$$m_{a\to i}(s_i) \propto \sum_{\mathbf{s}_{a\setminus i}} \Psi_a(\mathbf{s}_a) \prod_{j\in a\setminus i} n_{j\to a}(s_j).$$

L'ensemble des points fixes de (2.4), c'est-à-dire des points fixes de BP, correspond donc à l'ensemble des points stationnaires du lagrangien (2.14). De plus, Heskes a montré que les points fixes stables de BP sont des minima de F_{Bethe} [46]. Dans le cas de variables binaires, la réponse définitive à la question du lien entre les points fixes de BP et les extrema locaux de F_{Bethe} a été fourni dans par Watanabe. Elle est résumée dans la proposition suivante.

Proposition 2.1 (Watanabe [100])**.** *Tous les points fixes stables de BP, pour un certain niveau d'amortissement $\varepsilon < 1$, sont des minima de l'énergie libre de Bethe. Cependant, il existe des minima qui ne sont pas des points fixes stables (même avec un niveau d'amortissement $\varepsilon \to 1$).*

Pour conclure, on remarque que le point de vue variationnel fournit même les constantes de proportionnalité sous-entendues par le symbole « $\propto$ ». En particulier, à un point fixe m, on doit avoir

$$m_{a\to i}(s_i) = \frac{Z_a(\mathbf{m})}{Z_i(\mathbf{m})} \sum_{\mathbf{s}_{a\setminus i}} \Psi_a(\mathbf{s}_a) \prod_{j\in a\setminus i} n_{j\to a}(s_j), \tag{2.17}$$

en identifiant $Z_a(\mathbf{m})$ et $Z_i(\mathbf{m})$ par (2.5)–(2.6). Ce sont en effet les mises à jour qui sont proposées par Heskes [46]. Notons cependant que d'autres constantes sont utilisées [1, 101] et qu'il n'y a pas de consensus à ce sujet. On verra dans le chapitre 3 qu'en fait le choix de cette constante n'importe guère.

2.3.4 Généralisations

L'algorithme BP permet de calculer les marginales $\mathbb{P}_{\sigma_i}$ de manière exacte dans le cas de graphes de facteurs arborescents car la forme de la loi jointe $\mathbb{P}_{\boldsymbol{\sigma}}$

sous l'approximation de Bethe est alors exacte. Ceci a inspiré de nombreuses généralisations de l'algorithme BP qui sont obtenues en considérant d'autres formes pour la loi jointe $\mathbf{b}$. Citons notamment le « Generalized Belief Propagation (GBP) » de Yedidia *et al.* [110, 111] qui consiste à considérer des beliefs joints de régions plus grandes que les facteurs. En choisissant convenablement ces régions, qui doivent vérifier certaines contraintes, il est alors possible de réaliser l'inférence exacte sur des graphes de facteurs de structure plus complexe. Bien sûr, cela a un coût et l'algorithme GBP nécessite l'échange de messages de taille supérieure et entraîne une augmentation de la complexité des mises à jour.

Une autre approche consiste à considérer une famille de loi légèrement différente pour $\mathbf{b}$:

$$b(\mathbf{s}) = \prod_{a \in \mathbb{F}} b_a(\mathbf{s}_a)^{c_a} \prod_{i \in \mathbb{V}} b_i(s_i)^{1-c_i},$$

avec $c_i = \sum_{a \ni i} c_a$. L'algorithme ainsi obtenu est appelé « Fractional Belief Propagation (FBP) » et a été développé Wiegerinck et Heskes [107]. La forme des beliefs $\mathbf{b}_a$ à un point fixe de FBP est

$$b_a(\mathbf{s}_a) \propto \Psi_a(\mathbf{s}_a)^{1/c_a} \prod_{j \in a} n_{j \to a}(s_j).$$

L'intuition derrière ces coefficients c est de tenir compte des cycles du graphe de facteurs ; le coefficient c_a sert donc à définir une fonction Ψ_a corrigée pour tenir compte de cela. Minka [74] propose une autre interprétation ; FBP résout en fait un problème de minimisation mais pour une fonction objectif différente : une α-divergence.

Malgré leurs mérites, ces généralisations ont un défaut qui est l'absence de règle générale pour déterminer les régions (pour GBP) ou les coefficients c_a (pour FBP), ce qui réduit leur intérêt pratique.

D'autres approches consistent à développer des méthodes de minimisation de l'énergie libre de Bethe dont la convergence est garantie. C'est notamment le cas de l'algorithme à double boucle de Yuille [112] et des travaux de Meltzer *et al.* [70] qui permettent d'assurer la convergence par le choix de l'ordre des mises à jour. Ces garanties de convergence se payent, là aussi, en terme de temps de calcul et/ou de simplicité de mise en œuvre.

Dans le cas de variables binaires et d'interaction de paires, un algorithme a été proposé par Welling et Teh [106]. Il permet de diminuer l'énergie libre à chaque itération, ce qui assure sa convergence. Sa complexité est équivalente à celle de BP.

Malgré les mérites respectifs de ces généralisations, l'utilisation de la version classique de BP reste très répandue. Une première raison est *sa simplicité de mise en œuvre*, comparé à celle de ses généralisations. Seule la connaissance

de la loi (2.1) est nécessaire pour utiliser BP, alors que pour GBP ou FBP, il est nécessaire d'effectuer une calibration des régions ou des paramètres qui est dépendante du modèle. Cette étape de préparation n'est pas nécessaire pour BP. De plus, comparé aux méthodes de minimisation directe de l'énergie libre, l'algorithme BP est, par nature, *aisément parallélisable*. Cela permet d'envisager son utilisation sur des graphes de taille importante. Enfin, cet algorithme a prouvé son efficacité empirique [79]. Notons qu'en général, et dans ce travail en particulier, il est aussi nécessaire de construire la loi jointe (2.1) de manière approchée. On est donc souvent moins intéressé par l'estimation exacte des marginales d'une loi (2.1), construite de manière approchée, que par une estimation qui soit compatible avec la construction de la loi jointe. Pour plus de précision sur ce principe on réfère le lecteur au papier de Wainwright [98]. On reviendra sur ce sujet aux chapitres 5 et 6.

2.4 BP comme re-paramétrisation de la distribution

Une autre interprétation de l'algorithme BP est obtenue en remarquant que la règle de mise à jour (2.4) peut se ré-exprimer comme

$$m_{a\to i}^{(n+1)}(s_i) \propto \frac{b_{i|a}(s_i)}{b_i(s_i)} m_{a\to i}^{(n)}(s_i),$$

où $b_{i|a}(s_i) \stackrel{\text{def}}{=} \sum_{\mathbf{s}_{a\setminus i}} b_a(\mathbf{s}_a)$ est la marginale de la variable i vue du facteur a. En utilisant les expressions (2.5) et (2.6) des beliefs, il est possible d'éliminer les messages de la formule de mise à jour. On obtient alors les mises à jour suivantes :

$$b_i^{(n+1)}(s_i) \leftarrow b_i^{(n)}(s_i) \prod_{a\ni i} \frac{b_{i|a}^{(n)}(s_i)}{b_i^{(n)}(s_i)},$$

$$b_a^{(n+1)}(\mathbf{s}_a) \leftarrow b_a^{(n)}(\mathbf{s}_a) \prod_{i\in a} \prod_{c\ni i, c\neq a} \frac{b_{i|c}^{(n)}(s_i)}{b_i^{(n)}(s_i)}.$$

Les fonctions Ψ et Φ n'apparaissent plus dans ces formules de mise à jour. L'algorithme dépend cependant bien de la loi jointe (2.2) puisque les beliefs initiaux sont calculés selon (2.5) et (2.6) qui y font référence. Il est possible de prouver que l'algorithme défini ainsi est bien équivalent à BP (voir Wainwright [97, p.120]). En fait, on peut même remarquer qu'à chaque itération la loi jointe $b(\mathbf{s})$ n'évolue pas. On rappelle que $b(\mathbf{s})$ s'exprime comme

$$b(\mathbf{s}) = \prod_{a\in\mathbb{F}} \left(\frac{b_a(\mathbf{s}_a)}{\prod_{j\in a} b_j(s_j)} \right) \prod_{i\in\mathbb{V}} b_i(s_i).$$

En utilisant les définitions (2.6) et (2.5) des beliefs on obtient

$$b(\mathbf{s}) = \prod_{a\in\mathbb{F}} \left(\frac{\Psi_a(\mathbf{s}_a) \prod_{j\in a} n_{j\to a}(s_j)}{\prod_{j\in a} \Phi_j(s_j) \prod_{d\ni j} m_{d\to j}(s_j)} \right) \prod_{i\in\mathbb{V}} \left(\Phi_i(s_i) \prod_{d\ni i} m_{d\to i}(s_i) \right),$$

$$= \left(\prod_{a\in\mathbb{F}} \Psi_a(\mathbf{s}_a) \prod_{i\in\mathbb{V}} \Phi_i(s_i) \right) \frac{\prod_{i\in\mathbb{V}} \prod_{d\ni i} m_{d\to i}(s_i)}{\prod_{a\in\mathbb{F}} \prod_{j\in a} m_{a\to j}(s_j)} = \mathbb{P}_{\boldsymbol{\sigma}}(\boldsymbol{\sigma} = \mathbf{s}).$$

À chaque itération BP ne fait donc rien d'autre que ré-écrire la loi jointe $b(\mathbf{s})$ sous une autre forme. Cet interprétation de BP en tant que reparamétrisation de la loi jointe a été fournie par Wainwright [97, chapitre 5] ; les points fixes de BP correspondent alors aux paramétrisations $\mathbf{b}$ de la loi jointe $\mathbb{P}_{\boldsymbol{\sigma}}$ vérifiant les conditions de compatibilité locale. Cette vision conduit naturellement à considérer des reparamétrisations plus évoluées, BP considérant seulement des reparamétrisations sur les arbres constitués d'un facteur et des variables qu'il contient. Ces généralisations correspondent aux algorithmes « Tree Re-Weighted Belief Propagation » (TRWBP) [58, 97, 99]. Minka a montré [73] que les algorithmes TRWBP sont en fait un cas particulier des algorithmes FBP. L'intuition sous-jacente est cependant très différente et l'étude de la convergence des algorithmes TRWBP est mieux comprise [58].

2.5 Conditions suffisantes de convergence

L'algorithme BP, lorsqu'il converge, permet de trouver un minimum local d'une certaine « distance » aux vraies marginales comme on a vu dans la section 2.3 (proposition 2.1). Cependant, il existe des cas où l'algorithme ne converge pas, comme l'ont observé Murphy et al. [79]. Dans ces cas, BP ne nous fournit aucun résultat d'approximation des marginales ; en effet, à chaque itération l'énergie libre peut arbitrairement croître ou décroître. On comprend donc qu'à l'évidence, la convergence de l'algorithme BP est un aspect important de son étude.

Cas des graphes à un unique cycle. On sait que, par construction, l'algorithme BP converge en un nombre fini d'itérations lorsque le graphe de facteur est acyclique [53]. Le cas le plus simple à considérer est alors celui où le graphe contient un unique cycle. Ce cas a été étudié par Aji *et al.* [1] ainsi que par Weiss [102]. Les itérations de BP s'expriment alors comme des produits de matrices et il est possible de montrer que l'algorithme converge vers le vecteur de Perron de la matrice en question [102]. Les beliefs obtenus ne sont cependant pas les marginales exactes. De plus, par d'autres moyens, Watanabe et Fukumizu [101] ont montré l'unicité du point fixe de BP dans ce cas particulier.

Ce type de graphes, comportant un unique cycle, est assez courant dans le domaine des codes correcteurs d'erreurs ; ils sont associés aux codes dits « tailbiting » dont on peut trouver une description par Ma et Wolf [66].

Étude de BP en tant qu'itérations de Picard. L'approche classique pour exprimer des conditions suffisantes de convergence de l'algorithme BP consiste à étudier la suite des messages $\mathbf{m}^{(n)}$, définis par récurrence selon $\mathbf{m}^{(n+1)} = \Theta(\mathbf{m}^{(n)})$, à partir d'un ensemble de messages initiaux $\mathbf{m}^{(0)}$. Les messages $\mathbf{m}^{\infty}$ obtenus à la convergence sont des points fixes de l'application Θ définie ci-dessous

$$m_{a\to i}^{(n+1)}(s_i) = \Theta_{ai,s_i}(\mathbf{m}^{(n)}) \propto \sum_{\mathbf{s}_{a\setminus i}} \Psi_a(\mathbf{s}_a) \prod_{j\in a} \Phi_j(s_j) \prod_{b\ni j, b\neq a} m_{b\to j}^{(n)}(s_j).$$

Il est donc possible d'étudier la convergence de BP à l'aide d'outils d'analyse appliqués à la fonction Θ. On peut obtenir des conditions suffisantes de convergence en imposant que l'application Θ soit contractante. C'est l'approche employée par Mooij et Kappen [76]. On cherche donc à exprimer des conditions suffisantes pour que Θ soit une contraction vis-à-vis d'une certaine norme, c'est-à-dire des conditions impliquant que :

$$\sup_{\mathbf{m}} ||J_\Theta(\mathbf{m})|| < 1,$$

où $J_\Theta(\mathbf{m})$ est la matrice jacobienne de Θ au point $\mathbf{m}$. Le choix de la norme ne détermine pas si Θ est ou non une contraction, toutes les normes étant équivalentes en dimension finie, mais il influence les conditions suffisantes obtenues. Considérons le cas d'un modèle d'Ising, c'est-à-dire le cas de variables binaires $\sigma_i \in \{-1, +1\}$ avec une loi de probabilité de la forme

$$\mathbb{P}_{\boldsymbol{\sigma}}(\boldsymbol{\sigma} = \mathbf{s}) = \frac{1}{Z} \exp\left(\sum_{(i,j)\in\mathbb{E}} J_{ij} s_i s_j + \sum_{i\in\mathbb{V}} h_i s_i \right). \tag{2.18}$$

En considérant la norme L^1, Mooij et Kappen [76] dérivent la condition suivante :

$$\max_{i\in\mathbb{V}} \max_{k\in\partial i} \sum_{j\in\partial i\setminus k} \tanh|J_{ij}| < 1. \tag{2.19}$$

Un critère plus fort de convergence est obtenu en considérant la norme spectrale de la matrice Θ' :

Proposition 2.2 (Mooij et Kappen [76])**.** *Considérons le cas de variables binaires et d'une loi de probabilité de la forme* (2.18)*. Si le rayon spectral de la matrice* $\mathbf{A}$*, d'indices correspondant au liens du graphe et décrite ci-dessous, est strictement inférieur à* 1 *alors l'algorithme BP converge vers un unique point fixe quelle que soit son initialisation.*

$$A_{i\to j}^{k\to \ell} = \tanh|J_{ij}| \mathbb{1}_{\{i=\ell\}} \mathbb{1}_{\{k\in\partial i\setminus j\}}. \tag{2.20}$$

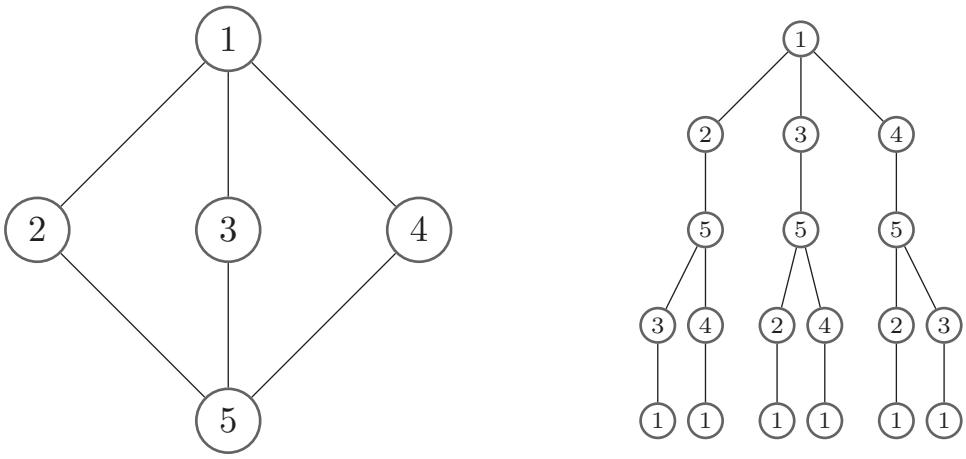


FIGURE 2.2 : Exemple de graphe (à gauche), et de son revêtement universel à partir du sommet 1 (à droite). L'arbre à droite est en réalité infini ; on a représenté ici ses 5 premières couches.

Ces deux résultats semblent indiquer que la convergence vers un unique point fixe correspond au cas d'interactions faibles. La même approche est étendue au cas d'une loi de probabilité de la forme (2.2) pour des variables σ_i discrètes quelconques [76].

Lien avec les mesures de Gibbs. Les premiers travaux faisant le lien entre l'étude des propriétés de BP et les mesures de Gibbs sont l'œuvre de Tatikonda et Jordan [91]. L'intérêt des mesures de Gibbs est de généraliser les distributions de Boltzmann à des graphes infinis. Pour plus de détails, on se référera à la description extensive qui en est faite par Georgii [38]. Le lien entre BP et ces mesures de Gibbs est obtenu en remarquant qu'à chaque itération BP résout, de manière exacte le problème de marginalisation sur un arbre dont la taille croît. Cet arbre a été en premier lieu introduit par Weiss et Freeman [105] sous le nom de d'arbre de calcul [1]. Il s'agit en fait tout simplement d'un arbre infini conservant la connectivité locale du graphe originel. En terme topologique cet arbre de calcul est le revêtement universel, tel que décrit par Angluin [5], du graphe de facteurs $\mathcal{G}$. La figure 2.2 fournit un exemple du concept de revêtement universel d'un graphe. Tatikonda et Jordan fournissent alors une condition suffisante de convergence.

Proposition 2.3 (Tatikonda et Jordan [91])**.** *Dès lors que l'on a unicité de la mesure de Gibbs définie sur l'arbre de calcul, l'algorithme BP converge presque sûrement vers un unique point fixe.*

1. « computation tree » en anglais.

En particulier, il est possible d'expliciter une condition suffisante de convergence de l'algorithme, en s'assurant de l'unicité de la mesure de Gibbs ; en physique statistique, on parle d'absence de transition de phase, chaque mesure étant appelée phase.

Proposition 2.4 (Georgii [38] p. 143)**.** *On a unicité de la mesure de Gibbs, et donc convergence presque sûre de BP, si la condition suivante est vérifiée*

$$\sup_{i\in\mathbb{V}} \sum_{a\ni i} (d_a - 1)\delta(\log \Psi_a) < 2, \tag{2.21}$$

avec $\delta(f) \stackrel{def}{=} \sup_x f(x) - \inf_x f(x)$.

Des conditions d'unicité plus fines utilisant la théorie des mesures de Gibbs sont détaillées dans la thèse de Winkler [108] et dans le livre de Georgii [38].

Notons que, dans le cas d'interactions de paires, la condition suffisante de convergence vers un unique point fixe obtenue par Ihler *et al.* [50] est légèrement plus forte que celle énoncée ici. Leur résultat est obtenu en étudiant les effets d'erreurs multiplicatives sur les messages autour d'un point d'équilibre.

Toutes ces conditions de convergence reviennent à imposer l'existence d'un unique point fixe. On présente au chapitre 3 une condition suffisante de stabilité des points fixes dans un contexte où plusieurs points fixes stables peuvent coexister.

2.6 Cas d'un vecteur gaussien : Gaussian Belief Propagation

Dans ce chapitre, nous avons uniquement considéré des variables à états discrets. Que se passe-t-il lorsque l'on considère des variables $\mathbf{X}$ continues ? Il est toujours possible de définir les messages (2.3) et (2.4) en remplaçant les sommes par des intégrales. En particulier, l'équation de mise à jour (2.4) devient

$$m_{a\to i}(x_i) \leftarrow \int_{\mathbf{x}_{a\setminus i}} \Psi_a(x_a) \prod_{j\in a, j\neq i} n_{j\to a}(x_j).$$

Cependant les messages, dépendants des valeurs des variables, ne sont plus des vecteurs mais des fonctions. Dans le cas de fonctions paramétriques, pour que les formules de mises à jour (2.3)–(2.4) puissent admettre des points fixes, il est nécessaire que les messages obtenus après une itération soient dans la même famille paramétrique que les messages dont ils sont issus. Le cas d'une loi normale multivariée avec des messages initiaux de forme gaussienne est le seul cas pratique, avec le cas de variables discrètes, pour lequel cette propriété est vérifiée.

Considérons un vecteur gaussien $\mathbf{X}$ de dimension n. La densité de $\mathbf{X}$ est alors

$$\mathbb{P}_{\mathbf{X}}(\mathbf{X}=\mathbf{x}) = \sqrt{\frac{\det \mathbf{A}}{(2\pi)^n}} \exp(-\frac{1}{2}\langle \mathbf{x}, \mathbf{A}\mathbf{x}\rangle + \langle \mathbf{h}, \mathbf{x}\rangle), \tag{2.22}$$

où $\mathbf{A}$ est la matrice de précision, c'est-à-dire l'inverse de la matrice de covariance de $\mathbf{X}$. On peut ré-exprimer la distribution de $\mathbf{X}$ sous la forme

$$\mathbb{P}_{\mathbf{X}}(\mathbf{X}=\mathbf{x}) \propto \prod_{i\in\mathbb{V}} \Phi_i(x_i) \prod_{\{i,j\}\in \mathrm{Supp}(\mathbf{A})} \Psi_{ij}(x_i, x_j),$$

avec $\Psi_{ij}(x_i, x_j) \stackrel{\text{def}}{=} \exp(-A_{ij}x_j x_i)$, $\Phi_i(x_i) \stackrel{\text{def}}{=} \exp(-\frac{1}{2}A_{ii}x_i^2 + h_i x_i)$ et $\mathrm{Supp}(\mathbf{A})$ l'ensemble des entrées non nulles hors de la diagonale de la matrice de précision $\mathbf{A}$. Dans ce cas, les facteurs sont donc uniquement constitués d'une paire de variables. La « facilité » à appliquer l'algorithme BP dans le cas gaussien provient des propriétés que l'on va maintenant présenter. Le proposition suivante implique que, si les messages initiaux sont gaussiens, alors ils le restent sous les formules de mise à jour.

Proposition 2.5 (Bickson [11] chapitre 2)**.** *Soit $f_1(x)$ et $f_2(x)$ deux densités gaussiennes. Alors, à une renormalisation près, la fonction produit $g(x) = f_1(x)f_2(x)$ est une densité de loi gaussienne. De plus si $f_1 \sim \mathcal{N}(\mu_1, \sigma_1)$ et $f_2 \sim \mathcal{N}(\mu_2, \sigma_2)$ on a $g \sim \mathcal{N}(\mu, \sigma)$ avec*

$$\mu = \sigma(\frac{\mu_1}{\sigma_1} + \frac{\mu_2}{\sigma_2}), \tag{2.23}$$

$$\sigma = \frac{\sigma_1\sigma_2}{\sigma_1 + \sigma_2}. \tag{2.24}$$

Dans l'équation (2.3), tous les messages $\mathbf{m}_{a\to i}$ correspondent à la même variable i. Les messages $\mathbf{n}$ suivent donc, à une normalisation près, une loi normale dont la moyenne et la variance peuvent être exprimées. L'autre équation de mise à jour (2.4) devient

$$m_{\{i,j\}\to i}(x_i) \leftarrow \int_{x_j} \Psi_{ij}(x_i, x_j) n_{j\to\{i,j\}}(x_j) dx_j, \tag{2.25}$$

et va aussi permettre d'exprimer les messages sous forme de lois normales de moyenne et variance calculables. Dans le cas gaussien, les passages de messages consistent donc à transmettre deux paramètres entre les sommets : des moyennes et variances. Les formules de mise à jour s'expriment alors sous la forme suivante :

$$\sigma_{i\to j} \leftarrow -\frac{A_{ij}^2}{A_{ii} + \sum_{k\in\partial i\setminus\{j\}} \sigma_{k\to i}}, \tag{2.26}$$

$$\mu_{i\to j} \leftarrow \frac{h_i + \sum_{k\in\partial i\setminus\{j\}} \sigma_{k\to i}\mu_{k\to i}}{A_{ij}}. \tag{2.27}$$

Les messages sont échangés entre sommets voisins du graphe défini par le support de la matrice de précision $\mathbf{A}$. Lorsque ces messages ont convergé, on calcule les approximations $\mathcal{N}(\mu_i, \sigma_i)$ des lois marginales $\mathbb{P}_{X_i}$ selon

$$\mu_i = \frac{h_i + \sum_{k \in \partial i} A_{ki} \mu_{k \to i}}{A_{ii} + \sum_{k \in \partial i} \sigma_{k \to i}}, \tag{2.28}$$

$$\sigma_i = \frac{1}{A_{ii} + \sum_{k \in \partial i} \sigma_{k \to i}}. \tag{2.29}$$

Il est prouvé [104] que, lorsqu'il converge, l'algorithme Gaussian Belief Propagation (GaBP) retourne les moyennes exactes des lois marginales $\mathbb{P}_{X_i}$ et ce quelle que soit la structure du graphe. Les variances sont quant à elles fausses [68]. La convergence de l'algorithme est de plus assurée dès lors que la matrice $\mathbf{A}$ est à diagonale dominante [104]. Une étude détaillée des propriétés de convergence de l'algorithme GaBP a été effectuée par Malioutov *et al.* [68].

Une application classique, proposée par Bickson [11], de l'algorithme GaBP consiste à remarquer que résoudre le système linéaire $\mathbf{Ax} = \mathbf{h}$ est équivalent à trouver le maximum de vraisemblance de la loi (2.22). GaBP est alors une méthode efficace pour résoudre ce problème sur certaines matrices creuses.

Annexe

2.A Suites $u_{n+1} = f(u_n)$: quelques notions

On rappelle ici quelques définitions et résultats élémentaires sur les suites de $\mathbb{R}^n$ définies par une relation de récurrence de la forme $u_{n+1} = f(u_n)$, avec f une fonction $\mathcal{C}^1$ (continue et à différentielle continue). Ces résultats sont en particulier utilisés dans les premiers chapitres traitant de l'algorithme BP.

2.A.1 Points fixes et stabilité

Définition 2.2. *Un point fixe u^* de f est un point fixe stable lorsqu'il existe un voisinage $\mathcal{O}$ de u^* tel que, $\forall u_0 \in \mathcal{O}$, la suite $u_{n+1} = f(u_n)$ converge vers u^*.*

Cette définition ne permet pas en général de conclure aisément sur la stabilité d'un point fixe donné. Pour cela on peut faire appel à la proposition suivante.

Proposition 2.6. *Soit u^* un point fixe de f $\mathcal{C}^1$ et J_{u^*} la jacobienne de f en u^*.*

- *Si le spectre de la matrice J_{u^*} est strictement inclus dans le cercle unité du plan complexe, alors u^* est un point fixe stable.*
- *Si la matrice J_{u^*} admet une valeur propre de module strictement supérieur à 1, alors le point fixe u^* est instable.*

Démonstration. Soit u_n dans un voisinage de u^* point fixe de f. On a alors

$$\begin{aligned} u_{n+1} - u^* &= f(u_n) - u^* \\ &= f(u^*) + J_{u^*}(u_n - u^*) - u^* + o(\|u_n - u^*\|^2) \\ &= J_{u^*}(u_n - u^*) + o(\|u_n - u^*\|^2), \end{aligned}$$

c'est-à-dire que localement, autour de u^*, la suite $(u_n - u^*)_{n\in\mathcal{N}}$ se comporte comme le système linéaire définit par $v_{n+1} = J_{u^*}\, v_n$. Et l'on a donc en particulier

$$\|v_{n+1}\| \leq \rho(J_{u^*})\|v_n\|,$$

où $\rho(J_{u^*}) \stackrel{\text{def}}{=} \max\{|\lambda|, \lambda \in \mathrm{Sp}(J_{u^*})\}$ est la plus grande valeur propre, en module, de la matrice J_{u^*}. Lorsque $\rho(J_{u^*}) < 1$, il existe donc un voisinage $\mathcal{O}$ de u^* tel que

$$\forall u_n \in \mathcal{O}, \|u_{n+1} - u^*\| < \|u_n - u^*\|,$$

donc $u_n \to u^*$, ce qui prouve que u^* est un point fixe stable. Réciproquement si $\rho(J_{u^*}) > 1$, quel que soit $\mathcal{O}$ le voisinage de u^*, il existe u_n dans $\mathcal{O}$ tel que $\|u_{n+1} - u^*\| > \|u_n - u^*\|$ et donc u^* est instable. Pour cela, il suffit de considérer u_n tel que $u_n - u^*$ soit un vecteur propre associé à la valeur propre $\rho(J_{u^*})$, que l'on peut rendre de norme aussi faible que voulue. □

On remarque que la proposition ne permet pas de conclure dans le cas où la valeur de plus grand module est sur le cercle unité. Dans ce cas il est parfois possible de conclure en étudiant les dérivées d'ordre supérieures, lorsqu'elles existent [78].

2.A.2 Effets de mises à jour amorties

Le principe, valide pour toute suite de la forme $u_{n+1} = f(u_n)$, consiste à remplacer la mise à jour $u_{n+1} \leftarrow f(u_n)$ par $u_{n+1} \leftarrow (1-\varepsilon)f(u_n) + \varepsilon u_n \stackrel{\text{def}}{=} f_\varepsilon(u_n)$. Ce type de mises à jour permet de limiter les phénomènes oscillatoires. On dit que $\varepsilon \in [0,1[$ est le niveau d'amortissement. Vérifions tout d'abord que ces mises à jours amorties ne modifient pas l'ensemble des points fixes :

Proposition 2.7. *Pour tout $\varepsilon \in [0,1[$, les points fixes de f_ε sont identiques aux points fixes de f.*

Démonstration. Soit x un point de fixe de f, alors $f_\varepsilon(x) = (1-\varepsilon)x + \varepsilon x = x$. Réciproquement si x est un point fixe de f_ε, avec $\varepsilon < 1$, alors $(1-\varepsilon)f(x) = (1-\varepsilon)x$ et donc $x = f(x)$. □

Ces mises à jours amorties permettent de plus de rendre stables certains points fixes jusque là instables ; plus précisément :

Proposition 2.8. *Tout point fixe pour lequel les valeurs propres de la jacobienne sont dans le demi plan $P = \{z \in \mathbb{C}, \Re(z) < 1\}$ peut être rendu stable par un amortissement adapté.*

Démonstration. Pour prouver cela, on exprime la matrice jacobienne J_ϵ correspondant aux mises à jour amorties en fonction de la jacobienne de la version sans amortissement J

$$J_\varepsilon = (1-\varepsilon)J + \varepsilon I.$$

Regardons alors comment évoluent les valeurs propres en calculant le polynôme caractéristique de J_ε.

$$\det(J_\varepsilon - \lambda I) = \det((1-\varepsilon)J - (\lambda - \varepsilon)I).$$

On voit donc que si λ est une valeur propre de J_ε, alors $\frac{\lambda-\varepsilon}{1-\varepsilon}$ est une valeur propre de J. Réciproquement, un amortissement de ε a pour effet de transformer les valeurs propres de J par l'application

$$h_\varepsilon : \lambda \to (1-\varepsilon)\lambda + \varepsilon. \tag{2.30}$$

h_ε est une application affine et tend vers l'application constante égale à 1 lorsque ε tend vers 1. Pour tout $\lambda \in \mathbb{C} \mid \Re(\lambda) < 1$, il existe donc une valeur de ε telle que $|h_\varepsilon(\lambda)| < 1$. Graphiquement, il est évident que dans ce cas seulement, le segment $[\lambda, 1]$ coupe le cercle unité (voir figure 2.3 page suivante). □

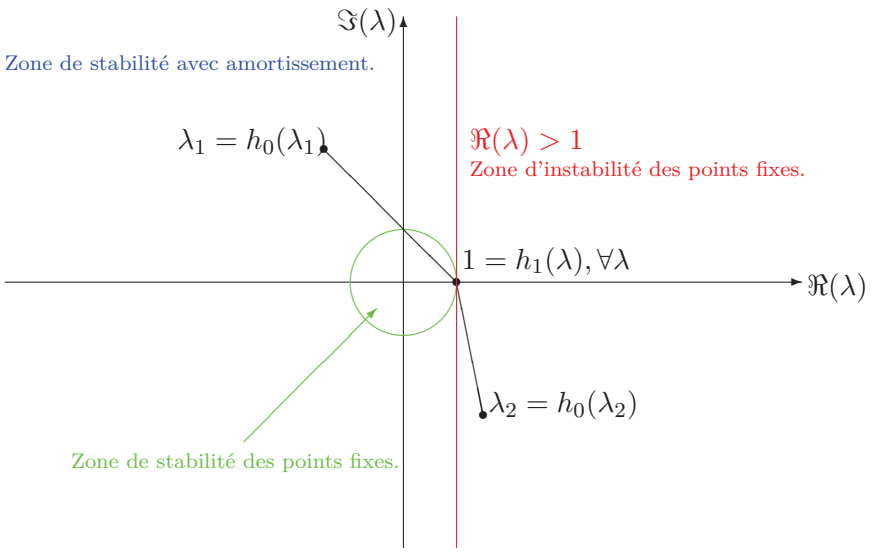


FIGURE 2.3 : Illustration de l'effet stabilisant des mises à jour amorties. Le cercle unité $\{\lambda \in \mathbb{C} \mid |\lambda| < 1\}$ étant inclus dans $\{\lambda \in \mathbb{C} \mid \Re(\lambda) < 1\}$ on voit clairement le caractère stabilisant des amortissements. Cependant ce surcroît de stabilité se fait au détriment de la rapidité de convergence.

Chapitre 3

Normalisation et convergence de BP

Il est utile de remarquer que BP peut être exprimé en éliminant les messages $\mathbf{n}_{i\to a}$ dans (2.4), on obtient alors

$$m_{a\to i}(s_i) \propto \Theta_{ai,s_i}(\mathbf{m}),$$

avec

$$\Theta_{ai,s_i}(\mathbf{m}) \stackrel{\text{def}}{=} \sum_{\mathbf{x}_{a\setminus i}} \Psi_a(\mathbf{s}_a) \prod_{j\in a\setminus i} \Big[\Phi_j(s_j) \prod_{a'\ni j, a'\neq a} m_{a'\to j}(s_j)\Big]. \tag{3.1}$$

On souhaite ici lever le relatif flou qu'implique le signe $\propto$ et on considère par la suite les schémas BP de la forme :

$$m_{a\to i}(s_i) \leftarrow \frac{\Theta_{ai,s_i}(\mathbf{m})}{Z_{ai}(\mathbf{m})}, \tag{3.2}$$

où la fonction Z_{ai} sera appelée « normalisation ». On s'intéresse dans ce chapitre à un aspect de l'algorithme qui n'a pas été étudié en détail dans la littérature : le rôle de cette normalisation Z_{ai} et plus particulièrement son influence sur la convergence de l'algorithme. Sauf mention explicite contraire, on suppose dans ce chapitre que le graphe de facteurs $\mathcal{G}$ est connexe.

On s'intéresse tout d'abord à définir plus précisément la notion de convergence de l'algorithme dans la section 3.1, où l'on remarque que la dynamique des beliefs normalisés (2.5–2.6) est la même quelle que soit la normalisation considérée (proposition 3.2). On introduit ensuite une classe de normalisation permettant de résumer la convergence de BP à la convergence des messages $\mathbf{m}$.

Dans la section 3.2, on donne un sens variationnel au schéma BP sans normalisation ($Z_{ai} \equiv 1$) en prouvant que ses points fixes sont équivalents à ceux du problème variationnel introduit au chapitre 2 (théorème 3.5). Après une comparaison des points fixes des schémas avec ou sans normalisation (partie 3.2.2), on montre (théorème 3.9) que le schéma sans normalisation est

instable et les raisons de cette instabilité sont explicitées (propositions 3.10 et 3.11).

On conclut enfin ce chapitre par une nouvelle condition suffisante de stabilité locale des points fixes de BP (théorème 3.12).

3.1 Types de convergence

Les variables naturelles pour exprimer les itérations de l'algorithme BP sont les messages, comme nous l'avons vu au chapitre précédent. Cependant, la valeur des messages en elle-même ne nous importe guère, c'est la valeur des beliefs qui nous intéresse. Nous allons ici préciser ce fait en décrivant deux types de convergences et leurs relations.

3.1.1 Dynamique des beliefs

A chaque étape de l'algorithme, en utilisant les équations (2.5) et (2.6), il est possible de calculer les beliefs courants $\mathbf{b}_i^{(n)}$ et $\mathbf{b}_a^{(n)}$ associés aux messages $\mathbf{m}^{(n)}$. La suite $\mathbf{m}^{(n)}$ est dite « b-convergente » lorsque les suites $\mathbf{b}_i^{(n)}$ et $\mathbf{b}_a^{(n)}$ convergent. C'est ce type de convergence qui nous intéresse en pratique. Le terme « m-convergence » fera référence à la convergence de la suite $\mathbf{m}^{(n)}$. Les beliefs s'exprimant comme des fonctions continues des messages (2.5)–(2.6), si la suite $\mathbf{m}^{(n)}$ est m-convergente alors elle est évidemment b-convergente, la réciproque n'étant pas vraie en général.

Présentons tout d'abord un lemme qui sera central dans ce chapitre et qui précise le fait, énoncé par Mooij et Kappen [77], que de nombreux messages correspondent aux mêmes beliefs.

Lemme 3.1. *Deux ensembles de messages* $\mathbf{m}$ *et* $\mathbf{m}'$ *correspondent aux mêmes beliefs si et seulement si il existe un ensemble de constantes strictement positives* c_{ai} *telles que*

$$m'_{a\to i}(s_i) = c_{ai} m_{a\to i}(s_i).$$

Démonstration. Il est trivial que des messages vérifiant la relation correspondent aux même beliefs. Pour l'autre implication, (2.5) et (2.6) impliquent que

$$\frac{b_a(\mathbf{s}_a) Z_a(\mathbf{m})}{\Psi_a(\mathbf{s}_a)} = \prod_{j\in a} \prod_{b\ni j, b\neq a} m_{b\to j}(s_j),$$

$$\frac{b_i(s_i) Z_i(\mathbf{m})}{\Phi_i(s_i)} = \prod_{a\ni i} m_{a\to i}(s_i).$$

Supposons que $\mathbf{m}$ et $\mathbf{m}'$ correspondent au même ensemble de beliefs $\mathbf{b}$ et posons $m_{a\to i}(s_i) = c_{ai,s_i}\, m'_{a\to i}(s_i)$. Alors, d'après la relation sur $b_i(s_i)$, le vecteur $\mathbf{c}$ doit vérifier

$$\prod_{a\ni i} c_{ai,s_i} = \prod_{a\ni i} \frac{m_{a\to i}(s_i)}{m'_{a\to i}(s_i)} = \frac{Z_i(\mathbf{m})}{Z_i(\mathbf{m}')} \stackrel{\text{def}}{=} v_i. \tag{3.3}$$

De plus les beliefs $b_a(\mathbf{s}_a)$ doivent aussi être préservés. En utilisant (3.3), on obtient

$$\prod_{j\in a}\frac{m_{a\to j}(s_j)}{m'_{a\to j}(s_j)}=\prod_{j\in a}c_{aj,s_j}=\frac{Z_a(\mathbf{m}')}{Z_a(\mathbf{m})}\prod_{i\in a}v_i\stackrel{\text{def}}{=}v_a. \tag{3.4}$$

Puisque v_i (resp. v_a) ne dépend pas du choix de s_i (resp. $\mathbf{s}_a$), (3.4) implique l'indépendance de c_{ai,s_i} par rapport à s_i. En effet, si l'on compare deux vecteurs $\mathbf{s}_a$ et $\mathbf{s}'_a$ tels que, $\forall i\in a\setminus j$, $s'_i=s_i$, mais $s'_j\neq s_j$, on a alors $c_{aj,s_j}=c_{aj,s'_j}$, ce qui conclut la preuve. □

Le fait qu'un ensemble de messages correspondent aux mêmes beliefs permet de se rendre compte de l'utilité de la notion de b-convergence. En effet certains systèmes dynamiques peuvent converger vers des sous-espaces [43] ; la distinction entre b et m-convergence est donc bien réelle.

Comme suggéré dans [77], il est naturel d'étudier le comportement de BP dans un espace quotient correspondant à cette invariance des beliefs. Introduisons d'abord une paramétrisation qui va faire apparaître cet espace quotient comme un simple espace vectoriel. On montrera alors que, en terme de b-convergence, la normalisation n'a pas d'effet. En considérant le changement de variables suivant

$$\mu_{a\to i}(s_i)\stackrel{\text{def}}{=}\log m_{a\to i}(s_i),$$

l'équation de mise à jour sans normalisation (3.1) devient, pour $\boldsymbol{\mu}\in\mathcal{N}\stackrel{\text{def}}{=}\mathbb{R}^{|\mathbb{E}|\times q}$,

$$\mu_{a\to i}(s_i)\leftarrow\Lambda_{ai,s_i}(\boldsymbol{\mu})\stackrel{\text{def}}{=}\log\left[\sum_{\mathbf{s}_{a\setminus i}}\Psi_a(\mathbf{s}_a)\exp\Big(\sum_{j\in a\setminus i}\sum_{\substack{b\ni j\\ b\neq a}}\mu_{b\to j}(s_j)\Big)\right].$$

Si l'on définit l'espace vectoriel $\mathcal{W}$ qui est engendré par les vecteurs $\{\mathbf{e}_{ai}\in\mathcal{N}\}_{(ai)\in\mathbb{E}}$

$$(\mathbf{e}_{ai})_{cj,s_j}\stackrel{\text{def}}{=}\mathbb{1}_{\{a=c,i=j\}},$$

alors l'ensemble des messages correspondant aux mêmes beliefs que $\boldsymbol{\mu}$ est simplement l'espace affine $\boldsymbol{\mu}+\mathcal{W}$ (lemme 3.1). Donc $\boldsymbol{\mu}^{(n)}$ est b-convergent si et seulement si $\boldsymbol{\mu}^{(n)}$ converge dans l'espace quotient $\mathcal{N}\setminus\mathcal{W}$, qui est un simple espace vectoriel [41]. On utilisera la notation $[x]$ pour la projection canonique de x sur $\mathcal{N}\setminus\mathcal{W}$.

La normalisation du vecteur $\boldsymbol{\mu}$ conduit à un vecteur $\boldsymbol{\mu}+\mathbf{w}$ avec $\mathbf{w}\in\mathcal{W}$. On a en effet

$$\Lambda_{ai,s_i}(\boldsymbol{\mu}+\mathbf{w})=\log\Big(\sum_{j\in a\setminus i}\sum_{\substack{b\ni j\\ b\neq a}}w_{bj}\Big)+\Lambda_{ai,s_i}(\boldsymbol{\mu})\stackrel{\text{def}}{=}l_{ai}+\Lambda_{ai,s_i}(\boldsymbol{\mu}),$$

que l'on peut résumer par $[\Lambda(\boldsymbol{\mu}+\mathcal{W})]=[\Lambda(\boldsymbol{\mu})]$, puisque $\mathbf{l}\in\mathcal{W}$. Cela signifie que la normalisation ne joue pas de rôle dans $\mathcal{N}\setminus\mathcal{W}$ et conduit à la proposition suivante.

Proposition 3.2. *La dynamique,* i.e. *la valeur des beliefs* (2.5)–(2.6) *à chaque étape de l'algorithme, avec ou sans normalisation, est identique.*

3.1.2 Normalisations homogènes positives

On introduit ici une famille de normalisations pour lesquelles les notions de m- et b-convergences sont équivalentes. L'intérêt des ces normalisations est avant tout de permettre l'étude de la convergence de BP en se focalisant uniquement sur la convergence des messages, plus naturelle à étudier. De plus, l'utilisation de ces normalisations permet numériquement de tester la convergence de BP sur les messages m directement.

Définition 3.1. *Une normalisation Z_{ai} est dite* positive homogène *lorsqu'elle est de la forme $Z_{ai} = N_{ai} \circ \Theta_{ai}$, avec $N_{ai} : \mathbb{R}_+^q \mapsto \mathbb{R}_+$ une fonction positive homogène d'ordre* 1 *vérifiant*

$$N_{ai}(\lambda \mathbf{m}_{a\to i}) = \lambda N_{ai}(\mathbf{m}_{a\to i}), \forall \lambda \geq 0, \tag{3.5}$$

$$N_{ai}(\mathbf{m}_{a\to i}) = 0 \iff \mathbf{m}_{a\to i} = 0. \tag{3.6}$$

Une famille de normalisations positives homogènes est obtenue lorsque que l'on considère pour N_{ai} les normes sur $\mathbb{R}^q$. C'est par exemple le cas de la normalisation Z_{ai}^{mess} :

$$Z_{ai}^{\text{mess}}(\mathbf{m}) = \sum_{s_i=1}^{q} \Theta_{ai,s_i}(\mathbf{m}), \tag{3.7}$$

qui correspond au cas où N_{ai} est la norme L_1 est qui souvent utilisée [50, 102]. Il n'est en fait même pas nécessaire de considérer une norme, la normalisation utilisée par Watanabe et Fukumizu [101] correspond à $Z_{ai}^1(\mathbf{m}) \stackrel{\text{def}}{=} \Theta_{ai,1}(\mathbf{m})$.

Notons cependant que la normalisation Z_{ai}^{bel} utilisée par Heskes [46] n'entre pas dans cette catégorie. Les résultats de cette section ne s'appliquent donc pas.

Proposition 3.3. *Pour toute normalisation positive homogène Z_{ai} telle que les fonctions N_{ai} soient* continues*, les notions de m-convergence et de b-convergence sont équivalentes.*

Démonstration. Soit $\mathbf{m}^{(n)}$ une suite b-convergente, c'est à dire que $\mathbf{b}_a^{(n)} \to \mathbf{b}_a$ et $\mathbf{b}_i^{(n)} \to \mathbf{b}_i$ lorsque $n \to \infty$. L'idée de cette preuve est, tout d'abord, d'exprimer les messages normalisés $\tilde{\mathbf{m}}_{a\to i}^{(n)}$ à chaque itération en fonction de beliefs $\mathbf{b}^{(n)}$, puis de conclure par un argument de continuité. Commençons par ré-exprimer (2.5) et (2.6) :

$$b_i^{(n)}(s_i) = \frac{\Phi_i(s_i)}{Z_i(\tilde{\mathbf{m}}^{(n)})} \prod_{a\ni i} \tilde{m}_{a\to i}^{(n)}(s_i),$$

$$b_a^{(n)}(\mathbf{s}_a) = \frac{\Psi_a(\mathbf{s}_a)}{Z_a(\tilde{\mathbf{m}}^{(n)})} \prod_{j\in a} \Phi_j(s_j) \prod_{b\ni j, b\neq a} \tilde{m}_{b\to j}^{(n)}(s_j).$$

En combinant ces deux équations, on obtient

$$\prod_{j\in a} \tilde{m}_{a\to j}^{(n)}(s_j) = \frac{K_{ai}^{(n)}(\mathbf{x}_{a\setminus i}; s_i)}{\tilde{Z}_{ai}(\tilde{\mathbf{m}})},$$

où une variable arbitraire $i \in a$ a été mise à part

$$\frac{1}{\tilde{Z}_{ai}(\tilde{\mathbf{m}})} \stackrel{\text{def}}{=} \frac{\prod_{j\in a} Z_j(\tilde{\mathbf{m}}^{(n)})}{Z_a(\tilde{\mathbf{m}}^{(n)})}, \quad K_{ai}^{(n)}(\mathbf{x}_{a\setminus i}; s_i) \stackrel{\text{def}}{=} \Psi_a(\mathbf{s}_a)\frac{\prod_{j\in a} b_j^{(n)}(s_j)}{b_a^{(n)}(\mathbf{s}_a)}.$$

Supposons maintenant que $\mathbf{x}_{a\setminus i}$ est fixé et considérons le vecteur de $\mathbb{R}^q$ suivant $\mathbf{K}_{ai}^{(n)}(\mathbf{x}_{a\setminus i}) \stackrel{\text{def}}{=} K_{ai}^{(n)}(\mathbf{x}_{a\setminus i}; \cdot)$. En normalisant cette équation avec une fonction positive homogène N_{ai}, on obtient

$$\frac{\tilde{m}_{a\to i}^{(n)}(s_i)}{N_{ai}\big[\tilde{\mathbf{m}}_{a\to i}^{(n)}\big]} = \frac{K_{ai}^{(n)}(\mathbf{x}_{a\setminus i}; s_i)}{N_{ai}\big[\mathbf{K}_{ai}^{(n)}(\mathbf{x}_{a\setminus i})\big]}.$$

En fait, on a $N_{ai}\big[\tilde{\mathbf{m}}_{a\to i}^{(n)}\big] = 1$, puisque $\tilde{\mathbf{m}}_{a\to i}^{(n)}$ a été normalisé par N_{ai}, et on obtient donc

$$\tilde{m}_{a\to i}^{(n)}(s_i) = \frac{K_{ai}^{(n)}(\mathbf{x}_{a\setminus i}; s_i)}{N_{ai}\big[\mathbf{K}_{ai}^{(n)}(\mathbf{x}_{a\setminus i})\big]}.$$

Ce qui conclut la preuve : $\tilde{\mathbf{m}}_{a\to i}^{(n)}$, étant exprimé comme une fonction continue de $\mathbf{b}_i^{(n)}$ et $\mathbf{b}_a^{(n)}$, converge dès lors que les beliefs convergent. □

3.2 Un algorithme BP sans normalisation ?

La proposition 3.2 montre que le choix de la normalisation n'a pas d'effet sur la dynamique de BP et ne joue donc pas de rôle lorsqu'il s'agit de b-convergence. Dans cette partie, on va chercher à comprendre pourquoi cette normalisation, qui trouve sa justification dans les contraintes (2.7) du problème variationnel, semble être un artefact. Dans un premier temps, on s'intéresse au rôle de la normalisation dans le problème variationnel sous-jacent.

3.2.1 Point de vue variationnel

On suppose ici que les beliefs $\mathbf{b}_i$ et $\mathbf{b}_a$ sont normalisés (2.7) et compatibles (2.8). Si seule la compatibilité est vérifiée, on utilisera les notations $\boldsymbol{\beta}_i$ et $\boldsymbol{\beta}_a$. Il est simple de se rendre compte qu'imposer uniquement les contraintes de compatibilité implique l'existence d'une masse totale unique $Z(\boldsymbol{\beta})$ pour les mesures $\boldsymbol{\beta}_a$ et $\boldsymbol{\beta}_i$ dès lors que le graphe de facteurs est connexe.

$$Z(\boldsymbol{\beta}) \stackrel{\text{def}}{=} \sum_{s_i} \beta_i(s_i) = \sum_{\mathbf{s}_a} \beta_a(\mathbf{s}_a).$$

Cette constante $Z(\boldsymbol{\beta})$ n'a pas *a priori* de relation avec les constantes $Z_a(\mathbf{m})$ et $Z_i(\mathbf{m})$ du chapitre précédent. Les quantités $\beta_i(s_i)/Z(\boldsymbol{\beta})$ et $\beta_a(\mathbf{s}_a)/Z(\boldsymbol{\beta})$ peuvent être notées $b_i(s_i)$ et $b_a(\mathbf{s}_a)$ puisque l'équation (2.7) est vérifiée.

Le but de cette section est d'expliciter les relations entre le problème variationnel (2.13) et sa version où l'on relâche les contraintes de normalisation (2.7). On exprime les deux problèmes variationnels de la manière suivante :

$$\mathcal{P}(E) \quad : \quad \operatorname*{argmin}_{\boldsymbol{\beta}\in E} F_{\mathrm{Bethe}}(\boldsymbol{\beta}), \tag{3.8}$$

avec E choisi entre :

– cas non normalisé : $E = E_1$ est l'ensemble de mesures positives telles que (2.8) est vérifiée,

$$E_1 \stackrel{\text{def}}{=} \left\{ \boldsymbol{\beta} > 0 \,\middle|\, \forall a \in \mathbb{F}, \forall i \in a, \forall s_i \sum_{\mathbf{s}_a \setminus s_i} \beta_a(\mathbf{s}_a) = \beta_i(s_i) \right\}; \tag{3.9}$$

– cas normalisé : $E = E_2 \subsetneq E_1$, avec de plus les contraintes (2.7),

$$\begin{aligned} E_2 &\stackrel{\text{def}}{=} E_1 \bigcap \left\{ \boldsymbol{\beta} \,\middle|\, \forall i \in \mathbb{V}, \sum_{s_i} \beta_i(s_i) = 1, \forall a \in \mathbb{F}, \sum_{\mathbf{s}_a} \beta_a(\mathbf{s}_a) = 1 \right\} \\ &= \mathcal{B}(\mathcal{G}). \end{aligned} \tag{3.10}$$

On peut dériver un algorithme BP pour le problème non normalisé de la manière décrite dans la partie 2.3. Les formules de mise à jour obtenues correspondent exactement au schéma de BP sans normalisation ($Z_{ai} \equiv 1$).

Pour comparer les solutions des problèmes de minimisation avec ($\mathcal{P}(E_2)$) ou sans ($\mathcal{P}(E_1)$) normalisation, nous introduisons la bijection φ entre E_1 et $E_2 \times \mathbb{R}_+^*$,

$$\begin{aligned} \varphi : E_2 \times \mathbb{R}_+^* &\longrightarrow E_1 \\ (\ \mathbf{b}\ ,\ Z\ ) &\longrightarrow \mathbf{b}Z. \end{aligned}$$

Le problème variationnel $\mathcal{P}(E_1)$ est équivalent à :

$$(\hat{\mathbf{b}}, \hat{Z}) = \operatorname*{argmin}_{(\mathbf{b},Z)\in E_2} F_{\mathrm{Bethe}}(\varphi(\mathbf{b}, Z)),$$

avec $\varphi(\hat{\mathbf{b}}, \hat{Z}) = \hat{\mathbf{b}}\hat{Z} = \hat{\boldsymbol{\beta}} \stackrel{\text{def}}{=} \operatorname{argmin}_{\boldsymbol{\beta}\in E_1} F_{\mathrm{Bethe}}(\boldsymbol{\beta})$.

On souhaite comparer cette valeur $\hat{\mathbf{b}}$ avec $\mathbf{b} = \operatorname{argmin}_{\boldsymbol{\beta}\in E_2} F_{\mathrm{Bethe}}(\boldsymbol{\beta})$. Pour cela, on exprime l'énergie libre de Bethe $F_{\mathrm{Bethe}}(\boldsymbol{\beta})$ d'une mesure non normalisée $\boldsymbol{\beta} \in E_1$ en fonction de l'énergie libre de Bethe de la mesure normalisée correspondante.

Lemme 3.4. *Dès lors que le graphe de facteurs est connexe, pour tout* $\boldsymbol{\beta} = Z\mathbf{b} \in E_1$*, on a :*

$$F_{\text{Bethe}}(Z\mathbf{b}) = Z\Big(F_{\text{Bethe}}(\mathbf{b}) + (1-\mathcal{C})\log Z\Big), \tag{3.11}$$

avec $\mathcal{C}$ *le nombre de cycles indépendants du graphe de facteurs.*

Démonstration.

$$\begin{aligned} F_{\text{Bethe}}(\boldsymbol{\beta}) &= F_{\text{Bethe}}(Z\mathbf{b}) \\ &= Z\left[\sum_{a,\mathbf{s}_a} b_a(\mathbf{s}_a)\log\left(\frac{Zb_a(\mathbf{s}_a)}{\Psi_a(\mathbf{s}_a)}\right) + \sum_{i,s_i} b_i(s_i)\log\left(\frac{(Zb_i(s_i))^{1-d_i}}{\Phi_i(s_i)}\right)\right] \\ &= Z\Big(F_{\text{Bethe}}(\mathbf{b}) + (|\mathbb{F}| + |\mathbb{V}| - |\mathbb{E}|)\log Z\Big) \\ &= Z\Big(F_{\text{Bethe}}(\mathbf{b}) + (1-\mathcal{C})\log Z\Big), \end{aligned}$$

la dernière égalité étant un résultat classique de théorie des graphes (voir [9] par exemple) valide pour tout graphe connexe. □

La quantité $1-\mathcal{C}$ est négative dans les cas non triviaux (au moins 2 cycles). Puisque tous les $Z\mathbf{b}$ sont équivalents de notre point de vue, on s'intéresse à la dérivée de $F_{\text{Bethe}}(Z\mathbf{b})$ par rapport à Z pour pouvoir comparer les quantités $\hat{\mathbf{b}}$ et $\mathbf{b}$.

Théorème 3.5. *Les beliefs normalisés correspondant aux extremums du problème variationnel non normalisé* $\mathcal{P}(E_1)$ *sont exactement les mêmes que ceux du problème normalisé* $\mathcal{P}(E_2)$ *lorsque* $\mathcal{C} \neq 1$*.*

Démonstration. En utilisant le lemme 3.4, on obtient

$$\frac{\partial F_{\text{Bethe}}(\boldsymbol{\beta})}{\partial Z} = F_{\text{Bethe}}(\mathbf{b}) + (1-\mathcal{C})\left(\log Z + 1\right),$$

et les points stationnaires correspondent à

$$\hat{Z} = \exp\left(\frac{F_{\text{Bethe}}(\hat{\mathbf{b}})}{\mathcal{C}-1} - 1\right). \tag{3.12}$$

À ces points, on peut calculer l'énergie libre de Bethe

$$F_{\text{Bethe}}(\hat{\boldsymbol{\beta}}) = F_{\text{Bethe}}(\hat{Z}\hat{\mathbf{b}}) = (\mathcal{C}-1)\exp\left(\frac{F_{\text{Bethe}}(\hat{\mathbf{b}})}{\mathcal{C}-1} - 1\right).$$

On vérifie aisément que, pour $\mathcal{C} \neq 1$, $x \mapsto (\mathcal{C}-1)e^{\frac{x}{\mathcal{C}-1}-1}$ est une fonction croissante. Les extremums de $F_{\text{Bethe}}(\boldsymbol{\beta})$ sont donc atteints pour les mêmes beliefs normalisés. Plus précisément, si $\mathbf{b}_1$ et $\mathbf{b}_2$ sont des éléments de E_2 tels que $F_{\text{Bethe}}(\mathbf{b}_1) \leq F_{\text{Bethe}}(\mathbf{b}_2)$ alors $F_{\text{Bethe}}(\hat{\boldsymbol{\beta}}_1 = \hat{Z}_1\mathbf{b}_1) \leq F_{\text{Bethe}}(\hat{\boldsymbol{\beta}}_2 = \hat{Z}_2\mathbf{b}_2)$, ce qui nous permet de conclure. □

En d'autres termes, dès lors que $\mathcal{C} \neq 1$, il est équivalent d'imposer la normalisation (2.7) dans le problème variationnel ou de normaliser après avoir obtenu une solution. De plus, dans le cas non normalisé, l'énergie libre de Bethe à un extremum local $\hat{\mathbf{b}}$ s'écrit

$$F_{\text{Bethe}}(\hat{\mathbf{b}}) = (\mathcal{C} - 1)\left(\log \hat{Z} + 1\right). \tag{3.13}$$

On peut donc comparer la « qualité » de différents points fixes en comparant uniquement la constante de normalisation obtenue : plus Z est petit, meilleure est l'approximation. Ceci étant tempéré par le fait que l'on ne minimise pas une vraie distance.

Revenons quelques instants sur le cas $\mathcal{C} = 1$. Dans ce cas, (3.11) s'écrit

$$F_{\text{Bethe}}(\boldsymbol{\beta}) = F_{\text{Bethe}}(Z\mathbf{b}) = ZF_{\text{Bethe}}(\mathbf{b}).$$

Cette relation explicite ce qui se passe : si le minimum du problème variationnel normalisé $\mathcal{P}(E_2)$ est strictement négatif, alors $F_{\text{Bethe}}(\boldsymbol{\beta})$ n'admet pas de borne inférieure finie et Z diverge vers $+\infty$; au contraire, si le minimum est strictement positif, Z tend vers zéro. Dans le cas très particulier où le minimum de $\mathcal{P}(E_2)$ est exactement zéro, toutes les valeurs de $Z \in \mathbb{R}^+$ sont équivalentes pour $\mathcal{P}(E_1)$.

En résumé, dès lors que le problème variationnel non normalisé $\mathcal{P}(E_1)$ est bien défini, il est équivalent au problème normalisé $\mathcal{P}(E_2)$ et la constante de normalisation permet de calculer immédiatement l'énergie libre de Bethe en utilisant (3.13). Lorsque ce n'est plus le cas, les dynamiques des algorithmes restent identiques (proposition 3.2) mais le problème $\mathcal{P}(E_1)$ ne fournira pas de constante Z, et l'on ne pourra obtenir l'énergie libre de Bethe du point fixe par ce biais.

3.2.2 Point du vue itératif.

Revenons ici à une vision purement itérative (3.2) des différents schémas BP et étudions l'effet de la normalisation sur le nombre de points fixes $\mathbf{m}$. Le théorème 3.7 montre que les ensembles de points fixes avec normalisation sont équivalents, en termes de beliefs, à ceux sans normalisation, lorsque $\mathcal{C} \neq 1$. Ce résultat est l'exact analogue du théorème 3.5, qui décrit lui les extremums des problèmes variationnels.

Commençons par énoncer un lemme qui décrit les relations entre les différentes constantes de normalisation d'un point fixe d'un schéma quelconque de BP.

Lemme 3.6. *Soit* $\tilde{\mathbf{m}}$ *tel que*

$$\tilde{m}_{a\to i}(s_i) = \frac{\Theta_{ai,s_i}(\tilde{\mathbf{m}})}{Z_{ai}(\tilde{\mathbf{m}})}.$$

Les constantes de normalisation associées vérifient

$$Z_{ai}(\tilde{\mathbf{m}}) = \frac{Z_a(\tilde{\mathbf{m}})}{Z_i(\tilde{\mathbf{m}})}, \qquad \forall ai \in \mathbb{E}, \tag{3.14}$$

En particulier, lorsque $Z_{ai} \equiv 1$, toutes les constantes Z_a et Z_i sont égales à une unique constante Z dès lors que le graphe est connexe.

Démonstration. L'équation de mise à jour avec normalisation (3.2), et (2.5)–(2.6), induisent que

$$\sum_{\mathbf{s}_a \setminus s_i} b_a(\mathbf{s}_a) = \frac{Z_i(\tilde{\mathbf{m}}) Z_{ai}(\tilde{\mathbf{m}})}{Z_a(\tilde{\mathbf{m}})} b_i(s_i).$$

Par définition de $Z_a(\tilde{\mathbf{m}})$ et $Z_i(\tilde{\mathbf{m}})$, $\mathbf{b}_a$ et $\mathbf{b}_i$ sont normalisés à 1. En sommant cette relation sur les valeurs de s_i, on obtient (3.14). □

Points fixes non normalisés associés à un point fixe normalisé. On montre ici que, dans la plupart des cas, les points fixes d'une version normalisée de BP (peu importe la normalisation utilisée) sont associés à des points fixes de la version non normalisée. On notera $\mathcal{C}$ le nombre de cycles indépendants du graphes de facteurs $\mathcal{G}$.

Théorème 3.7. *Un point fixe $\tilde{\mathbf{m}}$ de l'algorithme BP avec normalisation des messages correspond à un point fixe de l'algorithme non normalisé avec les mêmes beliefs si et seulement si une des deux conditions suivantes est vérifiée :*

(i) le nombre de cycles du graphe $\mathcal{G}$ est soit nul, soit strictement supérieur à 1 ($\mathcal{C} \neq 1$) ;

(ii) $\mathcal{C} = 1$ et les constantes de normalisation des beliefs sont telles que

$$\prod_{a \in \mathbb{F}} Z_a(\tilde{\mathbf{m}}) \prod_{i \in \mathbb{V}} Z_i(\tilde{\mathbf{m}})^{1-d_i} = 1. \tag{3.15}$$

Démonstration. Soit $\tilde{\mathbf{m}}$ un point fixe de (3.2). Trouvons des constantes c_{ai} telles que $m_{a\to i}(s_i) = c_{ai}\, \tilde{m}_{a\to i}(s_i)$ soit un point fixe (non nul) de la version non normalisée de l'algorithme. D'après le lemme 3.1, les messages $\mathbf{m}$ et $\tilde{\mathbf{m}}$ correspondent aux même beliefs. On a

$$\begin{aligned}
\Theta_{ai,s_i}(\mathbf{m}) &= \Big[\prod_{j \in a\setminus i} \prod_{a' \ni j, a' \neq a} c_{a'j} \Big] \Theta_{ai,s_i}(\tilde{\mathbf{m}}) \\
&= \Big[\prod_{j \in a\setminus i} \prod_{a' \ni j, a' \neq a} c_{a'j} \Big] Z_{ai}\, \tilde{m}_{a\to i}(s_i) \\
&= \frac{1}{c_{ai}} \Big[\prod_{j \in a\setminus i} \prod_{a' \ni j, a' \neq a} c_{a'j} \Big] Z_{ai}\, m_{a\to i}(s_i),
\end{aligned}$$

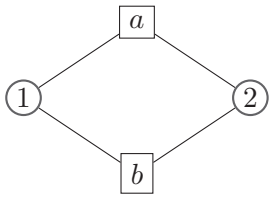


FIGURE 3.1 : Graphe de facteurs comportant un unique cycle.

et donc

$$\log c_{ai} - \sum_{j \in a \setminus i} \sum_{a' \ni j, a' \neq a} \log c_{a'j} = \log Z_{ai}.$$

Cette équation correspond aux conditions du lemme 3.16 page 71, avec $x_{ai} = \log c_{ai}$ et $y_{ai} = \log Z_{ai} = \log Z_a - \log Z_i$. Cette équation admet toujours une solution lorsque $\mathcal{C} \neq 1$; quand $\mathcal{C} = 1$, la condition (3.A.5) est, de plus, nécessaire et (3.15) suit. □

Il existe en général un nombre infini de points fixes $\mathbf{m}$ associés à chaque $\tilde{\mathbf{m}}$. Cependant, ce n'est pas un problème puisque tous ces points fixes correspondent aux même beliefs, d'après le lemme 3.1. En ce sens, normaliser les messages peut avoir l'effet de fusionner des points fixes équivalents.

Arrêtons-nous quelques instants sur ce cas particulier $\mathcal{C} = 1$ qui apparaît à nouveau après le théorème 3.5. Lorsque $\mathcal{C} = 1$, certains schémas normalisés de BP sont toujours convergents [1, 102]. Dans ce cas particulier, le schéma de BP sans normalisation correspond simplement à un système dynamique linéaire de la forme $\mathbf{m}^{(n+1)} = \mathbf{B}\mathbf{m}^{(n)}$ avec $\mathbf{B}$ une matrice dépendant des fonctions Ψ et Φ décrit par Aji *et al.* [1]. Considérons un graphe de facteurs trivial (figure 3.1) avec deux variables et deux facteurs $a = b = \{1, 2\}$ et supposons pour simplifier que $\Phi_1 = \Phi_2 = 1$. L'équation de point fixe devient alors

$$m_{a \to 1}(s_i) = \sum_{x_2} \Psi_a(x_1, x_2) m_{b \to 2}(x_2),$$

où, en notation matricielle,

$$\mathbf{m}_{a \to 1} = \Psi_a \mathbf{m}_{b \to 2} = \Psi_a \Psi_b \mathbf{m}_{a \to 1}.$$

L'existence d'un point fixe non nul pour le schéma de BP non normalisé implique donc que la matrice $\Psi_a \Psi_b$ admette 1 pour valeur propre. Ceci n'étant pas toujours vrai, ce schéma de BP peut n'admettre aucun point fixe. La condition additionnelle (ii) correspond en fait exactement à imposer que l'énergie libre du point fixe, $F_{\text{Bethe}}(\mathbf{b})$, soit nulle (voir proposition 3.17). Les théorèmes 3.5 et 3.7 sont donc analogues : ils donnent un sens au schéma sans normalisation sous les mêmes conditions.

Point fixe normalisé associé à un point fixe non normalisé. Il n'y a pas de résultat général associant tout point fixe de BP sans normalisation à un point fixe de la version Z_{ai}-normalisée. Nous allons donc nous intéresser au cas des normalisations positives homogènes, puis étudier deux exemples hors de cette classe.

La proposition suivante explicite l'effet de ces normalisations positives homogènes.

Proposition 3.8. *Tous les points fixes du schéma de BP non normalisé ($Z_{ai} \equiv 1$) associés au même ensemble de beliefs correspondent à un unique point d'un schéma BP avec une normalisation positive homogène.*

Démonstration. Soit $\mathbf{m}$ un point fixe de BP non normalisé. En utilisant le lemme 3.1, un point fixe $\tilde{\mathbf{m}}$ de BP normalisé est associé aux mêmes beliefs que m si et seulement si

$$\tilde{m}_{a\to i}(s_i) = c_{ai} m_{a\to i}(s_i). \tag{3.16}$$

Puisque Θ est multilinéaire on a,

$$\Theta_{ai,s_i}(\tilde{\mathbf{m}}) = \left(\prod_{j\in a\setminus i} \prod_{d\ni j, d\neq a} c_{dj} \right) \Theta_{ai,s_i}(\mathbf{m}),$$

et, d'après (3.5),

$$Z_{ai}(\tilde{\mathbf{m}}) = \left(\prod_{j\in a\setminus i} \prod_{d\ni j, d\neq a} c_{dj} \right) Z_{ai}(\mathbf{m}),$$
$$\tilde{m}_{a\to i}(s_i) = \frac{\Theta_{ai,s_i}(\tilde{\mathbf{m}})}{Z_{ai}(\tilde{\mathbf{m}})} = \frac{m_{a\to i}(s_i)}{Z_{ai}(\mathbf{m})}.$$

Le point fixe, $\tilde{\mathbf{m}}$, est donc déterminé de manière unique à partir de $\mathbf{m}$. En effet, $\tilde{\mathbf{m}}$ est clairement invariant pour tous les messages $\mathbf{m}$ correspondant aux mêmes beliefs (voir lemme 3.1), ce qui conclut la preuve. □

Pour mieux comprendre ce qu'implique la proposition 3.8, il est intéressant d'étudier par exemple ce qui se passe avec la normalisation Z^{bel} proposée par Heskes [47]. D'après le lemme 3.6, pour toute normalisation, on a à chaque point fixe

$$Z_{ai}(\mathbf{m}) = \frac{Z_a(\mathbf{m})}{Z_i(\mathbf{m})} \stackrel{\text{def}}{=} Z^{\text{bel}}_{ai}(\mathbf{m}).$$

Tout point fixe de tout schéma avec ou sans normalisation est donc un point fixe pour le schéma avec la normalisation Z^{bel}. La différence entre cette normalisation et une normalisation positive homogène apparaît clairement : Ces dernières fusionnent les familles de points fixes équivalents (en termes de beliefs) en un unique point fixe, alors que Z^{bel} au contraire conserve tous les

points fixes de tous les schémas possibles. À l'évidence Z^{bel}, est un schéma pour lequel seule la b-convergence a du sens.

Pour conclure cette partie, nous présentons un exemple de « mauvais choix » de fonction de normalisation pour illustrer un cas pathologique. On considère la normalisation suivante

$$Z_{ai}(\mathbf{m}) = \frac{\sum_x \Theta_{ai,x}(\mathbf{m})}{\sup_x m_{a\to i}(x)}.$$

Cette normalisation, non homogène, définit un schéma de BP n'admettant pas de points fixes. En s'inspirant de la preuve de la proposition 3.8, considérons $\tilde{\mathbf{m}}$ un point fixe de BP avec cette normalisation qui soit associé à un point fixe $\mathbf{m}$ de BP non normalisé selon (3.16). On a alors :

$$\tilde{m}_{a\to i}(s_i) = \frac{\Theta_{ai,s_i}(\tilde{\mathbf{m}})}{Z_{ai}(\tilde{\mathbf{m}})} = \frac{\tilde{m}_{a\to i}(s_i)}{Z_{ai}(\mathbf{m})}.$$

Il est aisé de vérifier que

$$Z_{ai}(\tilde{\mathbf{m}}) = \frac{\prod_{j\in a\setminus i}\prod_{b\ni j, b\neq a} c_{bj}}{c_{ai}} Z_{ai}(\mathbf{m}).$$

Puisque pour tout point fixe $\mathbf{m}$ du schéma non normalisé on a $Z_{ai}(\mathbf{m}) > 1$, aucun message $\tilde{\mathbf{m}}$ ne peut être un point fixe de ce schéma normalisé. Le théorème 3.7 permet de conclure que ce schéma n'admet aucun point fixe.

3.2.3 Un schéma instable

On montre ici que, dès lors que $\mathcal{C} > 1$, le schéma de BP sans normalisation correspondant au problème $\mathcal{P}(E_1)$ n'admet pas de point fixe $\mathbf{m}$ stable.

Soit $\mathbf{m}$ un point fixe de ce schéma sans normalisation, l'étude de sa stabilité s'appuie sur deux objets. Le premier est le graphe orienté $L(\mathcal{G})$ basé sur $\mathcal{G}$, dont les sommets sont les paires $(a, i) = ai$ telles que $i \in a$ et dont les arcs orientés relient ai à $a'j$ si $j \in a \cap a'$, $j \neq i$ et $a' \neq a$. La matrice d'adjacence $\mathbf{A}$ de $L(\mathcal{G})$ est définie par les coefficients

$$A_{ai}^{a'j} \stackrel{\text{def}}{=} \mathbb{1}_{\{j\in a\cap a',\, j\neq i,\, a'\neq a\}}. \tag{3.17}$$

Le second est l'ensemble de matrices stochastiques $\mathbf{B}^{(iaj)}$, attachées aux paires de variables (i, j) faisant partie d'un facteur a et dont les coefficients sont les beliefs conditionnels, pour tout $(k, \ell) \in \{1, \ldots, q\}^2$:

$$b_{k\ell}^{(iaj)} \stackrel{\text{def}}{=} b_a(\sigma_j = \ell | \sigma_i = k) = \sum_{\mathbf{x}_{a\setminus\{i,j\}}} \frac{b_a(\mathbf{s}_a)}{b_i(s_i)}\bigg|_{\substack{\sigma_i=k\\ \sigma_j=\ell}}.$$

On s'intéresse ici à des conditions de stabilité du premier ordre, c'est à dire s'exprimant sur le rayon spectral de la matrice jacobienne de Θ (voir section 2.A.1). Calculons donc la matrice jacobienne du schéma BP sans normalisation à ce point fixe

$$\begin{aligned}\frac{\partial \Theta_{ai,s_i}(m)}{\partial m_{a'\to j}(s_j)} &= \sum_{\mathbf{x}_{a\setminus\{i,j\}}} \frac{b_a(\mathbf{s}_a)}{b_i(s_i)} \frac{m_{a\to i}(s_i)}{m_{a'\to j}(s_j)} \mathbb{1}_{\{j\in a\setminus i\}} \mathbb{1}_{\{a'\ni j, a'\neq a\}} \\ &= \frac{b_{ij|a}(s_i,s_j)}{b_i(s_i)} \frac{m_{a\to i}(s_i)}{m_{a'\to j}(s_j)} A_{ai}^{a'j}.\end{aligned}$$

Cette matrice jacobienne est semblable, par un changement de variables trivial, à la matrice $\mathbf{J}$ définie, pour toute paire (ai,k) et $(a'j,\ell)$ de $\mathbb{E}\times\{0,\dots,q\}$ par les éléments

$$J_{ai,k}^{a'j,\ell} \stackrel{\text{def}}{=} b_{k\ell}^{(iaj)} A_{ai}^{a'j}.$$

Cette expression est analogue à la matrice jacobienne rencontrée par Mooij et Kappen [77]. Notons qu'elle dépend seulement que de la structure du graphe et de la valeur des beliefs normalisés correspondants au point fixe.

Le graphe de facteurs $\mathcal{G}$ étant connexe, la matrice $\mathbf{A}$ est clairement irréductible. Pour simplifier le propos, on suppose dans la suite que la matrice $\mathbf{J}$ est elle aussi irréductible, ceci étant vrai tant que les fonctions Ψ sont strictement positives. On remarque qu'à tout vecteur propre à droite de $\mathbf{A}$ correspond un vecteur propre à droite de $\mathbf{J}$, associé à la même valeur propre : si $\mathbf{v}=(v_{ai}, ai\in\mathbb{E})$ est tel que $\mathbf{A}\mathbf{v}=\lambda\mathbf{v}$, alors le vecteur $\mathbf{v}^+$, défini par ses coordonnées $v^+_{a'j\ell} \stackrel{\text{def}}{=} v_{a'j}$, pour tout $a'j\in\mathbb{E}$ et $\ell\in\{0,\dots,q\}$, vérifie $\mathbf{J}\mathbf{v}=\lambda\mathbf{v}$. *On dira que* $\mathbf{v}^+$ *est un vecteur propre à droite de* $\mathbf{J}$ *basé sur* $\mathbf{A}$. De même, si $\mathbf{u}$ est un vecteur propre à gauche de $\mathbf{A}$, avec des notations évidentes on peut définir un vecteur propre à gauche $\mathbf{u}^+$ de $\mathbf{J}$ basé sur $\mathbf{A}$ comme suit : $u^+_{aik} \stackrel{\text{def}}{=} u_{ai} b_i(k)$.

En utilisant cette correspondance entre les matrices $\mathbf{J}$ et $\mathbf{A}$, on obtient le théorème suivant, qui prouve l'instabilité du schéma BP sans normalisation :

Théorème 3.9. *Si le graphe de facteurs $\mathcal{G}$ comporte plus d'un cycle ($\mathcal{C}>1$), et que la matrice $\mathbf{J}$ est irréductible, alors le schéma BP sans normalisation n'admet aucun point fixe stable.*

Démonstration. Soit $\boldsymbol{\pi}$ le vecteur de Perron à droite de $\mathbf{A}$. Ses entrées sont positives puisque $\mathbf{A}$ est irréductible [85, théorème 1.5]. Le vecteur $\boldsymbol{\pi}^+$ basé sur $\mathbf{A}$ est aussi à entrées positives et il s'agit donc du vecteur de Perron à droite de $\mathbf{J}$ [85, théorème 1.6] ; le rayon spectral de $\mathbf{J}$ est donc égal à celui de $\mathbf{A}$.

Lorsque $\mathcal{C}>1$, le lemme 3.15 implique que 1 est une valeur propre de $\mathbf{A}$ associée à des vecteurs à divergence nulle. De plus, ces vecteurs ne peuvent être à entrées non-négatives, le rayon spectral de $\mathbf{A}$ est donc strictement supérieur à 1. Ceci conclut la preuve du théorème. □

Une conséquence de ce théorème est donc que la normalisation des messages est une condition nécessaire à la m-convergence.

3.2.4 Contraintes linéairement dépendantes.

On explicite ici la raison fondamentale de l'instabilité du schéma sans normalisation : *les ensembles de contraintes E_1 et E_2 des problèmes variationnels sont mal définis*, c'est à dire que les contraintes sont linéairement dépendantes. Commençons par regarder le problème variationnel sans normalisation $\mathcal{P}(E_1)$.

Proposition 3.10. *Les contraintes linéaires du problème variationnel sans normalisation $\mathcal{P}(E_1)$, telles que définies par* (3.9)*, ne sont pas linéairement indépendantes. Ces contraintes peuvent s'écrire sous la forme $\mathbf{D}\boldsymbol{\beta} = 0$ avec $\boldsymbol{\beta}$ le vecteur des $\beta_a(\mathbf{s}_a)$ et $\beta_i(s_i)$ et $\mathbf{D}$ une matrice de taille $q|\mathbb{E}| \times |\boldsymbol{\beta}|$. On a alors* $\text{rang}(\mathbf{D}) = q|\mathbb{E}| - \mathcal{C}$*, c'est-à-dire qu'il y a exactement $\mathcal{C}$ contraintes redondantes dans* (3.9)*.*

Démonstration. Tout d'abord, nommons les différentes contraintes de compatibilité comme suit :

$$C_{ai,s_i}(\boldsymbol{\beta}) \stackrel{\text{def}}{=} \sum_{\mathbf{s}_a \setminus s_i} \beta_a(\mathbf{s}_a) - \beta_i(s_i).$$

On commence par prouver que $\text{rang}(\mathbf{D}) \leq q|\mathbb{E}| - \mathcal{C}$. Pour cela on exhibe $\mathcal{C}$ contraintes qui sont induites par les $q|\mathbb{E}| - \mathcal{C}$ restantes. Soit $\mathcal{T}$ un arbre couvrant du graphe de facteurs $\mathcal{G}$ et considérons un ensemble de mesures $\boldsymbol{\beta}$ vérifiant l'ensemble de contraintes suivant :

$$\begin{cases} \forall ai \in \mathcal{T}, \forall s_i \in \{1 \dots q\}, C_{ai,s_i}(\boldsymbol{\beta}) = 0, \\ \forall ai \in \mathcal{G} \setminus \mathcal{T}, \forall s_i \in \{2 \dots q\}, C_{ai,s_i}(\boldsymbol{\beta}) = 0. \end{cases} \tag{3.18}$$

Par rapport à (3.9), on a donc retiré une contrainte de compatibilité ($C_{ai,1}$) pour tout arc ai de $\mathcal{G}$ n'appartenant pas à l'arbre couvrant $\mathcal{T}$. Les contraintes imposées sur $\mathcal{T}$ imposent l'existence d'une constante $Z(\boldsymbol{\beta}) \stackrel{\text{def}}{=} \sum_{\mathbf{s}_a} \beta_a(\mathbf{s}_a) = \sum_{s_i} \beta_i(s_i)$. Montrons alors que les contraintes « manquantes » sont induites par (3.18) ; soit $ai \in \mathcal{G} \setminus \mathcal{T}$, on a alors :

$$\begin{aligned} C_{ai,1}(\boldsymbol{\beta}) &= \sum_{\mathbf{s}_{a\setminus i}} \beta_a(\mathbf{s}_a, s_i = 1) - \beta_i(1) \\ &= Z(\boldsymbol{\beta}) - \sum_{k=2}^{q} \sum_{\mathbf{s}_{a\setminus i}} \beta_a(\mathbf{s}_a, s_i = k) - Z(\boldsymbol{\beta}) + \sum_{k=2}^{q} \beta_i(k) \\ &= -\sum_{k=2}^{q} C_{ai,k}(\boldsymbol{\beta}) = 0, \end{aligned}$$

où la notation $\beta_a(\mathbf{s}_a, s_i = k)$ indique que la composante s_i de $\mathbf{s}_a$ est égale à k. On a donc prouvé que $\text{rang}(\mathbf{D}) \leq q|\mathbb{E}| - \mathcal{C}$. Pour prouver l'autre inégalité

et conclure, il suffit de prouver qu'il n'est pas possible de faire de même en supprimant $\mathcal{C}+1$ contraintes, quelles qu'elles soient.

On montre tout d'abord qu'en supprimant 2 contraintes sur un même arc $ai \in \mathcal{G}$, on ne peut obtenir l'ensemble de mesures E_1. Pour cela, on considère une mesure $\boldsymbol{\beta}^*$ telle que

$$\forall (cj, s_j) \neq (ai, s_i = 1, 2),\ C_{cj,s_j}(\boldsymbol{\beta}^*) = 0, \tag{3.19}$$

et on suppose que l'on a alors :

$$C_{ai,1}(\boldsymbol{\beta}^*) = C_{ai,2}(\boldsymbol{\beta}^*) = 0.$$

On peut alors construire par permutation une nouvelle distribution $\boldsymbol{\beta}^{\#}$ qui vérifie les mêmes contraintes (3.19) et telle que $C_{ai,1}(\boldsymbol{\beta}^{\#})$ et $C_{ai,2}(\boldsymbol{\beta}^{\#})$ soient non nuls : considérons la permutation η qui laisse invariantes les mesures $\{\boldsymbol{\beta}_j\}_{j\in\mathbb{V}}$ et $\{\boldsymbol{\beta}_c\}_{c\in\mathbb{F}\setminus\{a\}}$ et

$$\begin{cases} (\eta(\boldsymbol{\beta}))_{a,\mathbf{s}_a|s_i=1} = \beta_a(\mathbf{s}_a, s_i = 2) \\ (\eta(\boldsymbol{\beta}))_{a,\mathbf{s}_a|s_i=2} = \beta_a(\mathbf{s}_a, s_i = 1) \\ (\eta(\boldsymbol{\beta}))_{a,\mathbf{s}_a|s_i=k} = \beta_a(\mathbf{s}_a), \text{si } k \notin \{1,2\}. \end{cases}$$

En appliquant η à $\boldsymbol{\beta}$, on permute donc uniquement les éléments $\beta_a(\mathbf{s}_a, s_i = 1)$ et $\beta_a(\mathbf{s}_a, s_i = 2)$. Vérifions maintenant que $\boldsymbol{\beta}^{\#} \overset{\text{def}}{=} \eta(\boldsymbol{\beta}^*)$ vérifie bien les contraintes (3.19) mais est telle que $C_{ai,1}(\boldsymbol{\beta}^{\#}) \neq 0$. On commence par remarquer que, η laissant invariantes les mesures $\{\boldsymbol{\beta}_j\}_{j\in\mathbb{V}}$ et $\{\boldsymbol{\beta}_c\}_{c\in\mathbb{F}\setminus\{a\}}$, on a

$$\forall (cj, s_j) \mid c \neq a,\ C_{cj,s_j}(\boldsymbol{\beta}^{\#}) = C_{cj,s_j}(\boldsymbol{\beta}^*) = 0.$$

De même, du fait que les composantes $(a, \mathbf{s}_a) \mid s_i \neq 1, 2$ sont laissées invariantes par η, on obtient

$$\forall k \notin \{1,2\},\ C_{ai,k}(\boldsymbol{\beta}^{\#}) = C_{ai,k}(\boldsymbol{\beta}^*) = 0.$$

Reste enfin le cas des contraintes $C_{aj,k}$ avec $j \neq i$

$$\begin{aligned} C_{aj,k}(\boldsymbol{\beta}^{\#}) &= \sum_{\mathbf{s}_{a\setminus j}} \beta_a^{\#}(\mathbf{s}_a, s_j = k) - \beta_j^{\#}(k) \\ &= \sum_{y=1}^{q} \sum_{\mathbf{s}_{a\setminus i,j}} \beta_a^{\#}(\mathbf{s}_a, s_j = k, s_i = y) - \beta_j^*(k) \\ &= \sum_{\mathbf{s}_{a\setminus j}} \beta_a^*(\mathbf{s}_a, s_j = k) - \beta_j^*(k) \\ &= C_{ai,k}(\boldsymbol{\beta}^*) = 0, \end{aligned}$$

car la permutation η a juste pour effet de changer l'ordre des termes dans la somme. Calculons maintenant la quantité $C_{ai,1}(\boldsymbol{\beta}^{\#})$ pour conclure :

$$C_{ai,1}(\boldsymbol{\beta}^{\#}) = C_{ai,2}(\boldsymbol{\beta}^*) + \beta_i^*(2) - \beta_i^*(1) = \beta_i^*(2) - \beta_i^*(1).$$

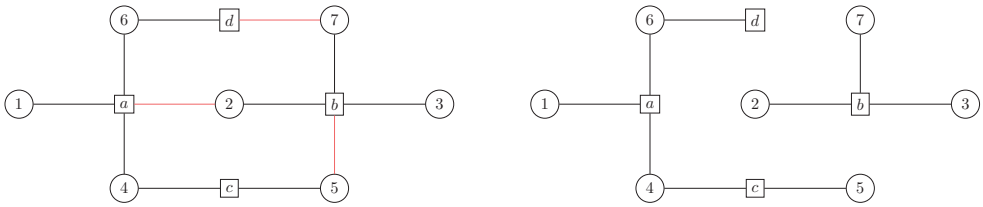


FIGURE 3.2 : Exemple de graphe de facteurs $\mathcal{G}^{\Gamma}$ résultant, basé sur un ensemble de contraintes Γ où $\mathcal{C}+1=3$ contraintes ont été supprimées. Ici les contraintes n'appartenant pas à Γ sont $(a2,1)$, $(d7,1)$ et $(b5,1)$ et les arcs correspondants sont représentés en rouge sur la figure de gauche. Le graphe de facteur $\mathcal{G}^{\Gamma}$ est représenté à droite.

On aura donc $C_{ai,1}(\eta(\boldsymbol{\beta}^*)) \neq 0$ pour toute mesure $\boldsymbol{\beta}^*$ vérifiant (3.19) et telle que $\beta_i^*(1) \neq \beta_i^*(2)$; supprimer 2 contraintes sur un même arc ne peut donc conduire à l'ensemble de mesures E_1. Il ne nous reste alors plus qu'à étudier la suppression de $\mathcal{C}+1$ contraintes sur des arcs différents. On note Γ l'ensemble des contraintes restantes. Considérons le graphe $\mathcal{G}^{\Gamma}$ défini comme suit :

$$ai \in \mathcal{G}^{\Gamma} \text{ si } \forall s_i \in \{1,\dots,q\}, (ai,s_i) \in \Gamma.$$

C'est-à-dire qu'il existe un arc entre a et i si les q contraintes de compatibilité sont imposées sur l'arc $ai \in \mathcal{G}$. Une illustration de $\mathcal{G}^{\Gamma}$ est fournie sur la figure 3.2. Puisque l'on a supprimé $\mathcal{C}+1$ contraintes sur des arcs différents, ce graphe ne peut être connecté ; on notera C^z la composante connexe de $\mathcal{G}^{\Gamma}$ contenant z. On considère une mesure $\boldsymbol{\beta}$ vérifiant toutes les contraintes de (3.9) et l'on va construire une mesure $\boldsymbol{\beta}'$ basée sur $\boldsymbol{\beta}$ qui violera les contraintes n'appartenant pas à Γ. Considérons un sommet quelconque $\alpha \in \mathcal{G}$, la mesure $\boldsymbol{\beta}'$ est alors construite comme suit :

(i) $\forall z \notin C^{\alpha}$, $\boldsymbol{\beta}'_z = \boldsymbol{\beta}_z$, c'est-à-dire que seules les mesures attachées à des sommets de la composante connexe C^{α} seront modifiés.

(ii) $\forall i \in C^{\alpha} \cap \mathbb{V}$, soit $S_i \stackrel{\text{def}}{=} \{s_i | \exists a \in \partial i \setminus C^{\alpha} \text{ et } C_{ai,s_i} \notin \Gamma\}$ l'ensemble des valeurs de s_i pour lesquelles il existe une contrainte C_{ai,s_i} manquantes dans Γ,
- si $S_i = \emptyset$, alors $\forall s_i, \beta'_i(s_i) = \beta_i(s_i) + \frac{1}{q_i}$,
- sinon, $\forall s_i, \beta'_i(s_i) = \beta_i(s_i) + \frac{\mathbb{1}_{\{s_i \in S_i\}}}{|S_i|}$.

(iii) $\forall a \in C^{\alpha} \cap \mathbb{F}$,

$$\beta'_a(\mathbf{s}_a) = Z'^{1-d_a} \prod_{i \in a} \left(\beta'_i(s_i) + \frac{\mathbb{1}_{\{s_i \in S_i\}} \mathbb{1}_{\{i \notin C^{\alpha}\}}}{|S_i|} \right).$$

Du fait de la connexité de C^{α}, toutes les mesures attachées à un sommet de C^{α} ont la même masse totale Z'. De plus (ii) indique que la valeur de cette masse

est $Z' = \sum_{x_\alpha} \beta_\alpha(x_\alpha) + 1 = Z + 1$. De plus les contraintes de compatibilités n'appartenant pas à Γ pour tout $i \in C^\alpha$ dont au moins un voisin n'est pas dans C^α sont violées. Il reste donc à vérifier que la construction (iii) permet de vérifier les contraintes de Γ et viole celles n'appartenant pas à Γ. Notons tout d'abord que la masse totale de $\boldsymbol{\beta}'_a$, pour $a \in C^\alpha$ est bien égale à Z'. En outre, on a trivialement

$$\sum_{\mathbf{s}_{a\setminus i}} \beta'_a(\mathbf{s}_a) = \beta'_i(s_i) + \frac{\mathbb{1}_{\{s_i \in S_i\}} \mathbb{1}_{\{i \notin C^\alpha\}}}{|S_i|},$$

ce qui implique le respect des contraintes de Γ uniquement. Ceci conclut donc la preuve, puisque pour tout système de contraintes Γ où l'on a ôté $\mathcal{C} + 1$, contraintes on a construit une mesure $\boldsymbol{\beta}'$ vérifiant les contraintes Γ et n'appartenant pas à E_1. □

La situation est similaire pour $\mathcal{P}(E_2)$, sauf que cette fois le nombre de contraintes redondantes est plus important.

Proposition 3.11. *Les contraintes affines du problème variationnel avec normalisation $\mathcal{P}(E_2)$ telles que définies par* (3.10) *ne sont pas linéairement indépendantes. Un système minimal de contraintes générant E_2 est obtenu en retirant $|\mathbb{E}|$ contraintes convenablement choisies. Autrement dit, le nombre de contraintes redondantes dans* (3.10) *est exactement le nombre d'arcs $|\mathbb{E}|$ du graphe de facteurs.*

Démonstration. Commençons par définir la notation des contraintes de normalisation

$$N_i(\mathbf{b}) \stackrel{\text{def}}{=} \sum_{s_i} b_i(s_i) - 1,$$
$$N_a(\mathbf{b}) \stackrel{\text{def}}{=} \sum_{\mathbf{s}_a} b_a(\mathbf{s}_a) - 1.$$

On remarque que l'ensemble de mesures E_2 est strictement inclus dans l'ensemble E_1 ($E_2 \subsetneq E_1$). E_1 et E_2 étant les ensembles de solutions d'équations affines, on en déduit que le nombre minimal de contraintes définissant E_2 est au moins $q|\mathbb{E}| - \mathcal{C} + 1$, c'est à dire 1 de plus que pour E_1. Soit une mesure $\mathbf{b}$ vérifiant les contraintes suivantes

$$\begin{cases} \forall ai \in \mathcal{T}, \forall s_i \in \{1, \dots, q\}, C_{ai,s_i}(\mathbf{b}) = 0, \\ \forall ai \in \mathcal{G} \setminus \mathcal{T}, \forall s_i \in \{2, \dots, q\}, C_{ai,s_i}(\mathbf{b}) = 0, \\ N_z(\mathbf{b}) = 0, \text{ pour un sommet quelconque } z \in \mathcal{G}. \end{cases}$$

Les deux premières lignes impliquent que $\mathbf{b} \in E_1$, comme on l'a vu dans la preuve de la proposition 3.10. Cela impose en particulier que l'on ait

$$\forall ai \in \mathcal{G}, \forall s_i, \sum_{\mathbf{s}_{a\setminus i}} b_a(\mathbf{s}_a) = b_i(s_i),$$

et donc

$$Z(\mathbf{b}) \stackrel{\text{def}}{=} \sum_{\mathbf{s}_a} b_a(\mathbf{s}_a) = \sum_{s_i} b_i(s_i).$$

La dernière contrainte, $N_z(\mathbf{b}) = 0$, impose de plus que $Z(\mathbf{b}) = 1$. On a donc bien identifié un système de $q|\mathbb{E}| - \mathcal{C} + 1$ contraintes qui génèrent E_2, ce qui conclut la preuve. □

Pour mieux comprendre cette dernière proposition, considérons une mesure $\mathbf{b}$ vérifiant les contraintes suivantes, où l'on a supprimé une contrainte de compatibilité pour chaque arc de $\mathcal{G}$

$$\begin{cases} \forall ai \in \mathcal{G}, \forall s_i \in \{2 \dots q\}, C_{ai,s_i}(\mathbf{b}) = 0, \\ \forall i \in \mathbb{V}, N_i(\mathbf{b}) = 0, \\ \forall a \in \mathbb{F}, N_a(\mathbf{b}) = 0. \end{cases} \tag{3.20}$$

Il est alors aisé de voir que les contraintes manquantes sont vérifiées :

$$\begin{aligned} C_{ai,1}(\mathbf{b}) &= \sum_{\mathbf{s}_a \setminus s_i} b_a(\mathbf{s}_a | s_i = 1) - b_i(1) \\ &= 1 - \sum_{k=2}^{q} \sum_{\mathbf{s}_a \setminus s_i} b_a(\mathbf{s}_a | s_i = k) - 1 + \sum_{k=2}^{q} b_i(k) \\ &= -\sum_{k=2}^{q} C_{ai,k}(\mathbf{b}) = 0. \end{aligned}$$

Le système (3.20) est aussi un système minimal : le nombre de contraintes est en effet $|\mathbb{V}| + |\mathbb{F}| + (q-1)|\mathbb{E}| = q|\mathbb{E}| - \mathcal{C} + 1$. C'est exactement ce type de contraintes que produit l'utilisation d'une normalisation positive homogène. En effet ces normalisations imposent une contrainte du type $N_{ai}(\mathbf{m}) = 1$ sur chaque arc ai de $\mathcal{G}$, ce qui réduit d'une unité le nombre de degrés de liberté de chaque vecteur $\mathbf{m}_{a \to i}$. Les messages $\mathbf{m}$ étant en bijection avec les multiplicateurs de Lagrange $\boldsymbol{\lambda}$, on a donc réduit de $|\mathbb{E}|$ le nombre de degrés de liberté de $\boldsymbol{\lambda}$ par l'utilisation d'une telle normalisation.

Il ne paraît cependant pas évident qu'un schéma BP avec une normalisation positive homogène corresponde très exactement au problème variationnel avec normalisation $\mathcal{P}(E_2)$. En effet, comme l'indique (2.17), on devrait alors utiliser la normalisation Z_{ai}^{bel}. On peut en fait interpréter toute normalisation positive homogène $Z_{ai} = N_{ai} \circ \Theta_{ai}$ comme imposant à chaque instant une relation du type

$$N_{ai}(\mathbf{m}_{a \to i}) \exp\left(-\gamma_a - \frac{\gamma_i}{1 - d_i}\right) = N_{ai}(\mathbf{m}_{a \to i}) \frac{Z_a(\mathbf{m})}{Z_i(\mathbf{m})} = 1.$$

Ces relations diminuent le nombre de degrés de liberté sur l'ensemble des multiplicateurs $(\boldsymbol{\lambda}, \boldsymbol{\gamma})$ de $|\mathbb{E}|$, et ces schémas résolvent très exactement le problème variationnel avec normalisation $\mathcal{P}(E_2)$.

Pour conclure cette partie, on voit donc qu'étudier le rôle de la normalisation dans le problème variationnel permet de comprendre les raisons de son absence apparente d'effet sur le schéma BP.

3.3 Une condition suffisante de stabilité.

Comme indiqué au chapitre 2, la convergence de l'algorithme BP reste un problème ouvert. Les approches classiques consistent à exprimer des conditions suffisantes sur les fonctions Ψ et Φ qui garantissent la convergence de BP vers un unique point fixe [50, 77, 91]. On change ici de point de vue en se plaçant dans un environnement ou plusieurs points fixes peuvent coexister et en examinant les propriétés de stabilité de chacun d'entre eux. En particulier, on exprime une condition suffisante (théorème 3.12) pour qu'un point fixe donné soit stable en fonction des beliefs correspondant et de propriétés spectrales du graphe.

D'après la proposition 3.3 toutes les normalisations positives homogènes continues rendent la m-convergence équivalente à la b-convergence. Celles-ci jouant toutes le même rôle, on considère ici la normalisation $Z_{ai}^{\text{mess}}(\mathbf{m})$ qui présente l'avantage d'être à la fois simple et différentiable. La matrice jacobienne à un point fixe $\mathbf{m}$, correspondant aux beliefs $\mathbf{b}$, a pour coefficients

$$\frac{\partial}{\partial \tilde{m}_{a'\to j}(\ell)}\left[\frac{\Theta_{ai,k}(\tilde{\mathbf{m}})}{\sum_{x=1}^{q}\Theta_{ai,x}(\tilde{\mathbf{m}})}\right] = J_{ai,k}^{a'j,\ell}\frac{m_{a\to i}(k)}{m_{a'\to j}(\ell)} - m_{a\to i}(k)\sum_{x=1}^{q} J_{ai,x}^{a'j,\ell}\frac{m_{a\to i}(x)}{m_{a'\to j}(\ell)},$$

et est donc semblable à la matrice $\widetilde{\mathbf{J}}$ de terme général

$$\widetilde{J}_{ai,k}^{a'j,\ell} \stackrel{\text{def}}{=} \left[b_{k\ell}^{(iaj)} - \sum_{x=1}^{q} m_{a\to i}(x) b_{x\ell}^{(iaj)}\right] A_{ai}^{a'j} = J_{ai,k}^{a'j,\ell} - \sum_{x=1}^{q} m_{a\to i}(x) J_{ai,x}^{a'j,\ell}. \quad (3.21)$$

Cette matrice peut être exprimée sous la forme $\widetilde{\mathbf{J}} = (\mathbf{I}-\mathbf{M})\mathbf{J}$, avec $\mathbf{I}$ la matrice identité et $\mathbf{M}$ de terme général :

$$M_{ai,k}^{a'j,\ell} \stackrel{\text{def}}{=} m_{a'\to j}(\ell)\mathbb{1}_{\{a=b,i=j\}}.$$

La présence des messages $\mathbf{m}$ dans la jacobienne $\widetilde{\mathbf{J}}$ semble compliquer l'étude, mais le spectre de $\widetilde{\mathbf{J}}$ ne dépend en fait pas des messages eux-mêmes. On sait [98, 34] qu'il est possible de choisir les fonctions $\hat{\Phi}$ et $\hat{\Psi}$ suivantes

$$\hat{\Phi}_i(s_i) \stackrel{\text{def}}{=} \hat{b}_i(s_i), \qquad \hat{\Psi}_a(\mathbf{s}_a) \stackrel{\text{def}}{=} \frac{\hat{b}_a(\mathbf{s}_a)}{\prod_{i\in a}\hat{b}_i(s_i)}, \quad (3.22)$$

pour obtenir un ensemble de beliefs donnés $\hat{\mathbf{b}}$ comme point fixe. Dans ce cas, BP admet alors un point fixe avec $\mathbf{b}_a = \hat{\mathbf{b}}_a$ et $\mathbf{b}_i = \hat{\mathbf{b}}_i$ qui correspond à $m_{a\to i}(s_i) \equiv 1$. Puisque l'on ne s'intéresse ici qu'aux beliefs, sans perte de

généralité, on utilise ici les fonctions (3.22). La définition (3.21) de $\widetilde{\mathbf{J}}$ s'exprime alors indépendamment des messages

$$\widetilde{J}^{a'j,\ell}_{ai,k} \stackrel{\text{def}}{=} \left[b^{(iaj)}_{k\ell} - \frac{1}{q} \sum_{x=1}^{q} b^{(iaj)}_{x\ell} \right] A^{a'j}_{ai} = J^{a'j,\ell}_{ai,k} - \frac{1}{q} \sum_{x=1}^{q} J^{a'j,\ell}_{ai,x}.$$

Pour présenter notre résultat principal, quelques notations sont nécessaires. Pour chaque paire (i,j) de variables appartenant à un facteur a, on définit le noyau stochastique $\mathbf{B}^{(iaj)}$ et le noyau stochastique combiné $\mathbf{K}^{(iaj)} \stackrel{\text{def}}{=} \mathbf{B}^{(iaj)}\mathbf{B}^{(jai)}$, qui a pour coefficients

$$K^{(iaj)}_{k\ell} \stackrel{\text{def}}{=} K^{(iaj)}(\sigma_j = \ell | \sigma_i = k) \stackrel{\text{def}}{=} \sum_{m=1}^{q} b^{(iaj)}_{km} b^{(jai)}_{m\ell}. \tag{3.23}$$

Dans la suite on considérera $\mathbf{b}_i$ en tant que vecteur de $\mathbb{R}^q$. Étant donné que $\mathbf{b}_i\, B^{(iaj)} = \mathbf{b}_j$, $\mathbf{b}_i$ est la mesure invariante associée à $\mathbf{K}^{(iaj)}$:

$$\mathbf{b}_i\, \mathbf{K}^{(iaj)} = \mathbf{b}_i\, \mathbf{B}^{(iaj)}\mathbf{B}^{(jai)} = \mathbf{b}_j\, \mathbf{B}^{(jai)} = \mathbf{b}_i,$$

et de plus $\mathbf{K}^{(iaj)}$ est réversible, puisque

$$b_i(k) K^{(iaj)}_{k\ell} = \sum_{m=1}^{q} b^{(jai)}_{mk} b_j(m) b^{(jai)}_{m\ell} = \sum_{m=1}^{q} b^{(jai)}_{mk} b^{(iaj)}_{\ell m} b_i(\ell) = b_i(\ell)\, K^{(iaj)}_{\ell k}.$$

Soit $\mu_2^{(iaj)}$ la deuxième plus grande valeur propre de $\mathbf{K}^{(iaj)}$ et définissons

$$\mu_2 \stackrel{\text{def}}{=} \max_{(iaj)} \sqrt{|\mu_2^{(iaj)}|}.$$

L'effet combiné de la structure du graphe et des corrélations locales sur la stabilité du point fixe considéré s'exprime de la manière suivante.

Théorème 3.12. *Soit λ_1 la valeur propre de Perron de la matrice $\mathbf{A}$.*

(i) si $\lambda_1 \mu_2 < 1$, le point fixe du schéma BP (3.2, 3.7) associé à $\mathbf{b}$ est stable.

(ii) Si le système est homogène ($\mathbf{B}^{(iaj)} = \mathbf{B}$ indépendant de i, j et a), $\lambda_1 \mu_2 \leq 1$ est aussi une condition nécessaire, avec μ_2 la deuxième plus grande valeur propre de $\mathbf{B}$.

Dans le cas homogène, si $\mathcal{G}$ a des degrés uniformes d_a et d_i, la condition devient

$$\mu_2 (d_a - 1)(d_i - 1) < 1.$$

La preuve du théorème 3.12 s'inspire de l'approche de Brémaud [14, chapitre 6] pour étudier les valeurs propres associées à certaines chaînes de Markov. Pour prouver la partie (i) du théorème, on considère la norme locale sur $\mathbb{R}^q$ attachée à chaque sommet i,

$$\|\mathbf{x}\|_{\mathbf{b}_i} \stackrel{\text{def}}{=} \sqrt{\sum_{k=1}^{q} x_k^2 b_i(k)},$$

ainsi que la moyenne de $\mathbf{x} \in \mathbb{R}^q$ pondérée par $\mathbf{b}_i$

$$\langle \mathbf{x} \rangle_{\mathbf{b}_i} \stackrel{\text{def}}{=} \sum_{k=1}^{q} x_k b_i(k).$$

Pour des raisons pratiques, on considère aussi la norme « hybride » sur $\mathbb{R}^{q \times |\mathbb{E}|}$

$$\|\mathbf{x}\|_{\boldsymbol{\pi},\mathbf{b}} \stackrel{\text{def}}{=} \sum_{(ai)\in\mathbb{E}} \pi_{ai} \|\mathbf{x}_{ai}\|_{\mathbf{b}_i},$$

où $\boldsymbol{\pi}$ est le vecteur de Perron à droite de $\mathbf{A}$, associé à λ_1. On a alors l'inégalité suivante :

Lemme 3.13. *Pour tout* $(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) \in \mathbb{R}^q \times \mathbb{R}^q$, *tel que* $\langle \mathbf{x}^{(i)} \rangle_{\mathbf{b}_i} = 0$ *et* $x_\ell^{(j)}\, \mathbf{b}_j(\ell) = \sum_k x_k^{(i)}\, b_i(k) B_{k\ell}^{(iaj)}$,

$$\langle \mathbf{x}^{(j)} \rangle_{\mathbf{b}_j} = 0 \qquad \textit{et} \qquad \|\mathbf{x}^{(j)}\|_{\mathbf{b}_j}^2 \leq \mu_2^{(iaj)} \|\mathbf{x}^{(i)}\|_{\mathbf{b}_i}^2.$$

Démonstration. Par définition des noyaux $\mathbf{K}^{(iaj)}$, on a

$$\begin{aligned} \|\mathbf{x}^{(j)}\|_{\mathbf{b}_j}^2 &= \sum_{k=1}^{q} \frac{1}{b_j(k)} \Big| \sum_{\ell=1}^{q} b_{\ell k}^{(iaj)} b_i(\ell)\, x_\ell^{(i)} \Big|^2 \\ &= \sum_{\ell,m} x_\ell^{(i)} x_m^{(i)} K_{\ell m}^{(iaj)} b_i(\ell). \end{aligned}$$

Puisque $\mathbf{K}^{(iaj)}$ est réversible, le théorème de Rayleigh [14, p. 217] implique que

$$\mu_2^{(iaj)} \stackrel{\text{def}}{=} \sup_{\mathbf{x}} \Big\{ \frac{\sum_{k\ell} x_k x_\ell K_{k\ell}^{(iaj)} b_i(k)}{\sum_k x_k^2 b_i(k)}, \langle \mathbf{x} \rangle_{\mathbf{b}_i} = 0, \mathbf{x} \neq 0 \Big\},$$

ce qui conclut la preuve. □

On s'intéresse aux itérations de J, que l'on va exprimer comme sommes sur des chemins de graphe,

$$(J^n)_{ai,k}^{a'j,\ell} = (A^n)_{ai}^{a'j} \left(B_{ai,a'j}^{(n)}\right)_{k\ell},$$

avec $\mathbf{B}_{ai,a'j}^{(n)}$ un noyau stochastique moyen,

$$\mathbf{B}_{ai,a'j}^{(n)} \stackrel{\text{def}}{=} \frac{1}{|\Gamma_{ai,a'j}^{(n)}|} \sum_{\gamma \in \Gamma_{ai,a'j}^{(n)}} \prod_{(ck,d\ell)\in\gamma} \mathbf{B}^{(kc\ell)}. \tag{3.24}$$

$\Gamma_{ai,a'j}^{(n)}$ est l'ensemble des chemins orientés de longueur n joignant ai et $a'j$ sur $L(\mathcal{G})$. Il contient exactement $|\Gamma_{ai,a'j}^{(n)}| = (A^n)_{ai}^{a'j}$ éléments.

Lemme 3.14. *Pour tout* $(\mathbf{x}^{(ai)}, \mathbf{x}^{(a'j)}) \in \mathbb{R}^{2q}$*, tel que* $\langle \mathbf{x}^{(ai)} \rangle_{\mathbf{b}_i} = 0$ *et*

$$x_\ell^{(a'j)}\, b_j(\ell) = \sum_k x_k^{(ai)}\, b_i(k) \big(B^{(n)}_{ai,a'j}\big)_{k\ell}\ ,$$

l'inégalité suivante est vérifiée :

$$\|\mathbf{x}^{(a'j)}\|_{\mathbf{b}_j} \leq \mu_2^n \|\mathbf{x}^{(ai)}\|_{\mathbf{b}_i}.$$

Démonstration. Soit $\mathbf{x}^{(a'j)}(\gamma)$ la contribution à $\mathbf{x}^{(a'j)}$ correspondant au chemin $\gamma \in \Gamma^{(n)}_{ai,a'j}$. D'après le lemme 3.13, on obtient pour chaque chemin

$$\|\mathbf{x}^{(a'j)}(\gamma)\|_{\mathbf{b}_j} \leq \mu_2^n \|\mathbf{x}^{(ai)}\|_{\mathbf{b}_i},$$

et l'inégalité triangulaire nous permet de conclure,

$$\|\mathbf{x}^{(a'j)}\|_{\mathbf{b}_j} \leq \frac{1}{|\Gamma^{(n)}_{ai,a'j}|} \sum_{\gamma \in \Gamma^{(n)}_{ai,a'j}} \|\mathbf{x}^{(a'j)}(\gamma)\|_{\mathbf{b}_j} \leq \mu_2^n \|\mathbf{x}^{(ai)}\|_{\mathbf{b}_i}.$$

□

Il est maintenant possible de prouver le théorème 3.12.

Preuve du théorème 3.12. Soient $\mathbf{v}$ et $\mathbf{v}'$ deux vecteurs tels que $\mathbf{v}' = \mathbf{v}\widetilde{\mathbf{J}}^n = \mathbf{v}(\mathbf{I}-\mathbf{M})\mathbf{J}^n$, car $\widetilde{\mathbf{J}}\mathbf{M} = 0$. On rappelle que l'effet de $(\mathbf{I}-\mathbf{M})$ est tout d'abord de projeter sur un vecteur de somme nulle :

$$\sum_k \left(\mathbf{v}\,(\mathbf{I}-\mathbf{M})\right)_{ai,k} = 0,\ \forall i \in \mathbb{V}.$$

On suppose donc directement $\mathbf{v}$ de la forme suivante

$$v_{ai,k} = x_{ai,k}\, b_i(k), \qquad \text{avec} \qquad \langle \mathbf{x}_{ai} \rangle_{\mathbf{b}_i} = 0.$$

Et donc, $\mathbf{v}' = \mathbf{v}\mathbf{J}^n$ est de la même forme. Définissons $x'_{a'j,\ell} \stackrel{\text{def}}{=} v'_{a'j,\ell}/b_j(\ell)$. On a

$$\|\mathbf{x}'\|_{\boldsymbol{\pi},\mathbf{b}} \leq \sum_{(a'j)\in\mathbb{E}} \pi_{a'j} \sum_{(ai)\in\mathbb{E}} \big(A^n\big)_{ai}^{a'j} \|\mathbf{y}^{(ai)}_{a'j}\|_{\mathbf{b}_j},$$

avec $y^{(ai)}_{a'j,\ell}\, b_j(\ell) = \sum_k x_{ai,k}\, b_i(k) \big(B^{(n)}_{ai,a'j}\big)_{k\ell}$. En appliquant le lemme 3.14 à $\mathbf{y}^{(ai)}_{a'j}$ on obtient

$$\|\mathbf{x}'\|_{\boldsymbol{\pi},\mathbf{b}} \leq \sum_{(a'j)\in\mathbb{E}} \pi_{a'j} \sum_{(ai)\in\mathbb{E}} \big(A^n\big)_{ai}^{a'j} \mu_2^n \|\mathbf{x}_{ai}\|_{\mathbf{b}_i} = \lambda_1^n \mu_2^n \|\mathbf{x}\|_{\boldsymbol{\pi},\mathbf{b}},$$

puisque $\boldsymbol{\pi}$ est le vecteur de Perron à droite de $\mathbf{A}$. Nous avons donc prouvé (i). Pour (ii), lorsque le système est homogène, $\widetilde{\mathbf{J}}$ est le produit tensoriel de $\mathbf{A}$ avec $\widetilde{\mathbf{B}}$, son spectre est donc le produit de leurs spectres respectifs. □

La quantité μ_2 est représentative du niveau d'information mutuelle entre les variables. Elle est liée au trou spectral (voir [26] pour des bornes géométriques) de chaque matrice stochastique $\mathbf{B}^{(iaj)}$; λ_1 pour sa part représente les propriétés statistiques de la connectivité du graphe. La borne $\lambda_1\mu_2 < 1$ peut être raffinée en considérant la moyenne des sommes sur les chemins de (3.24) qui permet de définir μ_2 comme

$$\mu_2 = \lim_{n\to\infty} \max_{(ai,a'j)} \Big\{ \frac{1}{|\Gamma^{(n)}_{ai,a'j}|} \sum_{\gamma\in\Gamma^{(n)}_{ai,a'j}} \Big(\prod_{(x,y)\in\gamma} \mu_2^{(xy)} \Big)^{\frac{1}{2n}} \Big\}. \tag{3.25}$$

Pour conclure cette partie on va expliciter la condition suffisante de stabilité du théorème 3.12 dans un cas simple.

Exemple 3.1. *Considérons le cas où toutes les variables sont binaires $\sigma_i \in \{0,1\}, \forall i \in \mathbb{V}$. La seconde valeur propre de $\mathbf{K}^{(iaj)}$ se calcule aisément puisqu'il s'agit d'une matrice 2×2. Soient les matrices suivantes*

$$\mathbf{P}^{(iaj)} \stackrel{def}{=} \begin{bmatrix} b_{ij|a}(0,0) & b_{ij|a}(0,1) \\ b_{ij|a}(1,0) & b_{ij|a}(1,1) \end{bmatrix},$$

et

$$\mathbf{Q}_i \stackrel{def}{=} \begin{bmatrix} b_i(0) & 0 \\ 0 & b_i(1) \end{bmatrix}.$$

On a alors $\mathbf{B}^{(iaj)} = (\mathbf{Q}_i)^{-1}\mathbf{P}^{(iaj)}$, et donc

$$\mathbf{K}^{(iaj)} = \mathbf{B}^{(iaj)}\mathbf{B}^{(jai)} = (\mathbf{Q}_i)^{-1}(\mathbf{P}^{(iaj)})^2(\mathbf{Q}_j)^{-1}.$$

Comme $\mathbf{K}^{(iaj)}$ est une matrice stochastique de taille 2, elle admet 1 comme valeur propre et son déterminant est égal à la seconde valeur propre $\mu_2^{(iaj)}$. On obtient donc

$$\mu_2^{(iaj)} = \det\left(\mathbf{K}^{(iaj)}\right) = \frac{\Big(b_{ij|a}(1,1)b_{ij|a}(0,0) - b_{ij|a}(0,1)b_{ij|a}(1,0)\Big)^2}{b_i(0)b_i(1)b_j(0)b_j(1)},$$

qui s'exprime plus simplement comme

$$\mu_2^{(iaj)} = \frac{\mathrm{cov}_a(\sigma_i,\sigma_j)^2}{\mathrm{var}(\sigma_i)\,\mathrm{var}(\sigma_j)} = \rho_a(\sigma_i,\sigma_j)^2,$$

où cov_a et ρ_a doivent être compris comme la covariance et le coefficient de corrélation linéaire selon la distribution $\mathbf{b}_a$. La condition suffisante (i) devient donc

$$\mu_2 = \max_{(iaj)} |\rho_a(\sigma_i,\sigma_j)| < \frac{1}{\lambda_1}.$$

En d'autres termes, tous les points fixes dont les corrélations sont inférieures à une certaine borne $\frac{1}{\lambda_1}$, fixée par la structure du graphe, sont des points fixes stables.

Annexes

3.A Propriétés spectrales du graphe de facteurs

Cette annexe est consacrée à certaines propriétés de la matrice d'adjacence $\mathbf{A}$ du graphe $L(\mathcal{G})$ définie par (3.17).

On considère deux types de champs associés à $\mathcal{G}$, les champs scalaires et les champs vectoriels. Les champs scalaires sont des quantités attachées aux sommets du graphe de facteurs $\mathcal{G}$, tandis que les champs vectoriels sont attachés aux arcs de $\mathcal{G}$. Un champ vectoriel $\mathbf{w} = \{w_{ai},\ ai \in \mathbb{E}\}$ est dit *à divergence nulle* si

$$\forall a \in \mathbb{F},\ \sum_{i \in a} w_{ai} = 0 \quad \text{et} \quad \forall i \in \mathbb{V},\ \sum_{a \ni i} w_{ai} = 0.$$

Un champ vectoriel $\mathbf{u} = \{u_{ai},\ ai \in \mathbb{E}\}$ est un *gradient* s'il existe un champ scalaire $\{u_a, u_i,\ a \in \mathbb{F},\ i \in \mathbb{V}\}$ tel que

$$\forall ai \in \mathbb{E},\ u_{ai} = u_a - u_i.$$

Tout champ vectoriel s'écrit comme la somme d'un champ à divergence nulle et d'un gradient. De plus, ces deux champs sont orthogonaux. En effet, le produit scalaire

$$\langle \mathbf{w}, \mathbf{u} \rangle = \sum_{ai \in \mathbb{E}} w_{ai} u_{ai} = \sum_{a \in \mathbb{F}} u_a \sum_{i \in a} w_{ai} - \sum_{i \in \mathbb{V}} u_i \sum_{a \ni i} w_{ai},$$

vaut 0 pour tous les champs gradient $\mathbf{u}$ si et seulement si $\mathbf{w}$ est à divergence nulle. Des considérations de dimension montrent que tout champ vectoriel $\mathbf{v}$ se décompose de cette manière.

Définissons l'opérateur de Laplace Δ associé $\mathcal{G}$ qui sera utilisé dans la suite. Pour tout champ scalaire $\mathbf{u}$:

$$(\Delta \mathbf{u})_a \stackrel{\text{def}}{=} d_a u_a - \sum_{i \in a} u_i, \qquad \forall a \in \mathbb{F} \tag{3.A.1}$$

$$(\Delta \mathbf{u})_i \stackrel{\text{def}}{=} d_i u_i - \sum_{a \ni i} u_a, \qquad \forall i \in \mathbb{V}. \tag{3.A.2}$$

Le lemme suivant décrit le spectre de la matrice A sous forme d'équation de Laplace sur le graphe $\mathcal{G}$.

Lemme 3.15. *(i) Les sous-espaces vectoriels correspondant aux vecteurs gradient et à divergence nulle sont stables par* $\mathbf{A}$*. De plus les vecteurs à divergence nulle sont des vecteurs propres de* $\mathbf{A}$ *associés à la valeur propre* 1*. (ii) les vecteurs propres associés à des valeurs propres* $\lambda \neq 1$ *sont les gradients d'un champ scalaire* $\mathbf{u}$ *qui vérifie*

$$(\Delta \mathbf{u})_a = \frac{(\lambda - 1)(d_a - 1)}{\lambda} u_a \text{ et } (\Delta \mathbf{u})_i = (1 - \lambda) u_i. \tag{3.A.3}$$

De plus il existe un vecteur propre gradient associé à la valeur propre 1 *si et seulement si* $\mathcal{G}$ *comporte exactement un cycle (*$\mathcal{C} = 1$*).*

Démonstration. En appliquant $\mathbf{A}$ à un vecteur $\mathbf{x}$ donné, on obtient :

$$\sum_{a'j\in\mathbb{E}} A_{ai}^{a'j} x_{a'j} = \sum_{j\in a}\Big(\sum_{a'\ni j} x_{a'j} - x_{aj}\Big) - \sum_{a'\ni i} x_{a'i} + x_{ai},$$

Les deux premiers termes du second membre s'annulent si $\mathbf{x}$ est à divergence nulle. De plus, le premier terme entre parenthèses est indépendant de i tandis que le second est indépendant de a ce qui justifie la première assertion. Résolvons maintenant l'équation aux valeurs propres $\mathbf{Ax}-\lambda\mathbf{x} = 0$ pour un gradient $\mathbf{x}$, avec $x_{ai} = u_a - u_i$. $\mathbf{Ax} - \lambda\mathbf{x}$ est le gradient d'une constante $K \in \mathbb{R}$, et on obtient par identification

$$\begin{cases} (\Delta\mathbf{u})_a + \sum_{j\in a}(\Delta\mathbf{u})_j = (1-\lambda)u_a + K \\ (\Delta\mathbf{u})_i = (1-\lambda)u_i + K. \end{cases}$$

Le laplacien d'une constante vaut zéro. Donc, pour $\lambda \neq 1$, K peut être re-absorbé dans $\mathbf{u}$ et, en combinant ces deux équations avec les définitions (3.A.1,3.A.2), on obtient l'équation (3.A.3). Pour $\lambda = 1$, on obtient

$$(\Delta\mathbf{u})_a = (1-d_a)K \qquad \text{et} \qquad (\Delta\mathbf{u})_i = K. \tag{3.A.4}$$

Soit $\mathbf{D}$ la matrice diagonale associée au graphe $\mathcal{G}$ dont les entrées diagonales sont les degrés d_a et d_i de chaque sommet. $\mathbf{M} = \mathbf{I} - \mathbf{D}^{-1}\Delta$ est une matrice stochastique irréductible, dont l'unique vecteur de Perron à droite $(1,\dots,1)$ génère le noyau de Δ. En conséquence, pour $K = 0$, la solution de (3.A.4) est $u_a = u_i = cte$ pour que $x_{ai} = 0$.

Pour $K \neq 0$, il y a une solution si le second membre de (3.A.4) est orthogonal au noyau (Δ est un opérateur symétrique). La condition s'exprime comme

$$0 = \sum_a (1-d_a) + \sum_i 1 = |\mathbb{F}| - |\mathbb{E}| + |\mathbb{V}| = 1 - \mathcal{C}.$$

□

Sachant que 1 est valeur propre de la matrice $\mathbf{A}$, on s'intéresse aux équations linéaires impliquant la matrice $\mathbf{I} - \mathbf{A}$. Puisque l'on sait déjà que les vecteurs à divergence nulle sont dans le noyau de cette matrice, on se restreint au cas où le terme constant est de forme gradient.

Lemme 3.16. *Soit* $\mathbf{y}$ *un champ vectoriel de type gradient, l'équation*

$$(\mathbf{I} - \mathbf{A})\mathbf{x} = \mathbf{y},$$

a une unique solution, à un vecteur à divergence nulle près, si et seulement si un des deux conditions suivantes est vérifiée :

1. $\mathcal{C} \neq 1$.
2. $\mathcal{C} = 1$ *et*

$$\sum_{a \in \mathbb{F}} y_a + \sum_{i \in \mathbb{V}} (1 - d_i) y_i = 0. \tag{3.A.5}$$

Démonstration. On recherche ici seulement des solutions de type gradient $x_{ai} = u_a - u_i$ et l'on écrit $y_{ai} = y_a - y_i$. Par les mêmes arguments que ceux du lemme 3.15, il existe une constante K telle que

$$\begin{aligned} \left(\Delta \mathbf{u}\right)_a &= K(d_a - 1) + y_a - \sum_{j \in a} y_j \\ \left(\Delta \mathbf{u}\right)_i &= y_i - K. \end{aligned}$$

En imposant la condition de compatibilité entre ces équations, on obtient

$$\sum_{a \in \mathbb{F}} y_a + \sum_{i \in \mathbb{V}} (1 - d_i) y_i = K(\mathcal{C} - 1).$$

Il est toujours possible de trouver une constante K satisfaisant cette relation lorsque que $\mathcal{C} \neq 1$. Lorsque $\mathcal{C} = 1$, (3.A.5) doit être vérifiée. □

3.B F_{Bethe} à partir des constantes de normalisation

On montre ici que l'énergie libre de Bethe correspondant aux beliefs $\mathbf{b}(\mathbf{m})$ à un point fixe $\mathbf{m}$ de BP s'exprime en fait uniquement en fonction des constantes de normalisation $Z_a(\mathbf{m})$ et $Z_i(\mathbf{m})$.

Proposition 3.17. *L'énergie libre de Bethe des beliefs* $\mathbf{b}(\mathbf{m})$ *associés aux messages* $\mathbf{m}$ *s'exprime comme*

$$F_{\text{Bethe}}(\mathbf{b}(\mathbf{m})) = -\log\left(\prod_{a \in \mathbb{F}} Z_a(\mathbf{m}) \prod_{i \in \mathbb{V}} Z_i(\mathbf{m})^{1-d_i}\right),$$

avec $Z_a(\mathbf{m})$ *et* $Z_i(\mathbf{m})$ *les constantes de normalisation de* (2.6) *et* (2.5)*.*

Démonstration. On rappelle la définition de l'énergie libre de Bethe

$$\begin{aligned} F_{\text{Bethe}}(\mathbf{b}(\mathbf{m})) &= \sum_{a \in \mathbb{F}, \mathbf{s}_a} b_a(\mathbf{s}_a) \log\left(\frac{b_a(\mathbf{s}_a)}{\Psi_a(\mathbf{s}_a)}\right) + \sum_{i \in \mathbb{V}, s_i} b_i(s_i) \log\left(\frac{b_i(s_i)^{(1-d_i)}}{\Phi_i(s_i)}\right). \\ &\stackrel{\text{def}}{=} \sum_{a \in \mathbb{F}} A(\mathbf{b}_a(\mathbf{m})) + \sum_{i \in \mathbb{V}} I(\mathbf{b}_i(\mathbf{m})) \end{aligned}$$

Commençons par exprimer les termes de F_{Bethe} correspondant aux facteurs

$$\begin{aligned} A(\mathbf{b}_a(\mathbf{m})) &= \sum_{\mathbf{s}_a} b_a(\mathbf{s}_a) \sum_{j \in a} \log \left(\Phi_j(s_j) \prod_{\substack{c \ni j \\ c \neq a}} m_{c \to j}(s_j) \right) - \log Z_a(\mathbf{m}), \\ &= \sum_{j \in a, s_j} b_j(s_j) \log \left(\Phi_j(s_j) \prod_{\substack{c \ni j \\ c \neq a}} m_{c \to j}(s_j) \right) - \log Z_a(\mathbf{m}), \end{aligned}$$

en utilisant le fait que $\sum_{\mathbf{s}_{a \setminus j}} b_a(\mathbf{s}_a) = b_j(s_j)$ car $\mathbf{m}$ est un point fixe de BP. Définissons les notations suivantes

$$\begin{cases} \Delta\Phi_i \stackrel{\text{def}}{=} \sum_{s_i} b_i(s_i) \log \Phi_i(s_i), \\ \Delta\mathbf{m}_i \stackrel{\text{def}}{=} \sum_{s_i} b_i(s_i) \log \prod_{a \ni i} m_{a \to i}(s_i). \end{cases}$$

On a alors :

$$A(\mathbf{b}_a(\mathbf{m})) = -\log Z_a(\mathbf{m}) + \sum_{j \in a} (\Delta\Phi_j + \Delta\mathbf{m}_j) - \sum_{j \in a} b_j(s_j) \log m_{a \to j}(x_j).$$

Exprimons maintenant les termes de F_{Bethe} correspondant aux variables

$$\begin{aligned} I(\mathbf{b}_i(\mathbf{m})) &= \sum_{s_i} b_i(s_i) \log \left(\Phi_i(s_i)^{-d_i} \prod_{a \ni i} m_{a \to i}(s_i)^{1-d_i} \right) - (1-d_i) \log Z_i(\mathbf{m}), \\ &= -d_i \Delta\Phi_i + (1-d_i)\Delta\mathbf{m}_i - (1-d_i) \log Z_i(\mathbf{m}). \end{aligned}$$

En sommant tous ces termes, on obtient

$$\begin{aligned} F_{\text{Bethe}}(\mathbf{b}(\mathbf{m})) &= \sum_{a \in \mathbb{F}} A(\mathbf{b}_a(\mathbf{m})) + \sum_{i \in \mathbb{V}} I(\mathbf{b}_i(\mathbf{m})) \\ &= -\sum_{a \in \mathbb{F}} \log Z_a(\mathbf{m}) - \sum_{i \in \mathbb{V}} (1-d_i) \log Z_i(\mathbf{m}) + \sum_{i \in \mathbb{V}} \Delta\mathbf{m}_i \\ &\qquad - \sum_{a \in \mathbb{F}} \sum_{j \in a} b_j(s_j) \log m_{a \to j}(x_j). \end{aligned}$$

Pour obtenir le résultat, il ne reste donc qu'à prouver l'égalité suivante

$$\sum_{i \in \mathbb{V}} \Delta\mathbf{m}_i - \sum_{a \in \mathbb{F}} \sum_{j \in a} b_j(s_j) \log m_{a \to j}(x_j) = 0,$$

qui est une trivialité d'après la définition de $\Delta\mathbf{m}_i$. □

Chapitre 4

Belief Propagation avec observations incertaines

Dans les chapitres 2 et 3, l'algorithme Belief Propagation a été discuté mais pas son application au problème qui nous intéresse et qui est décrite dans le chapitre 1. On suppose dans ce chapitre que le modèle est donné sous la forme du graphe de facteurs $\mathcal{G} = (\mathbb{V} \cup \mathbb{F}, \mathbb{E})$ ainsi que des fonctions Ψ et Φ. On s'intéresse alors à la manière d'introduire des observations *incertaines* de certaines variables et de réaliser l'inférence conditionnellement à celles-ci. Dans la suite, on notera $\mathbb{V}^* \subset \mathbb{V}$ le sous-ensemble de sommets correspondant aux variables observées. On parlera d'*observations certaines* lorsque la valeur des variables σ est observée :

$$\forall i \in \mathbb{V}^* \subset \mathbb{V}, \quad \sigma_i = s_i^* \in \{0, \dots, q-1\}.$$

Dans le cas d'observations certaines, il suffit [34, 28] de remplacer les mises à jour (2.3) des messages $\mathbf{n}_{i\to a}$ pour $i \in \mathbb{V}^*$ par :

$$n_{i\to a}(s_i) = \begin{cases} 0 \text{ si } s_i \neq s_i^* \\ 1 \text{ si } s_i = s_i^*, \end{cases}$$

pour réaliser l'inférence sur la loi $\mathbb{P}_{\boldsymbol{\sigma}}(\boldsymbol{\sigma}_{\mathbb{V}\setminus\mathbb{V}^*}, \boldsymbol{\sigma}_{\mathbb{V}^*} = \mathbf{s}_{\mathbb{V}^*})$.

On cherche ici à construire des algorithmes d'inférence basés sur BP pour lesquels il est possible d'inclure des observations plus générales sur la distribution des variables $\sigma_i, i \in \mathbb{V}^*$. Deux types d'observations des distributions de σ_i sont utilisés :

- les observations dites *virtuelles*, décrites par Pearl [82, chapitre 2] sous le terme de « virtual evidence », qui s'interprètent comme l'ajout de variables aléatoires binaires ζ_i, pour $i \in \mathbb{V}^*$, au modèle. Chacune de ces variables ζ_i est reliée uniquement à σ_i, et sa distribution est telle que

$$\mathbb{P}(\zeta_i = 1 | \sigma_i) = \mathbf{b}_i^*,$$

où $\mathbf{b}_i^*$ est notre observation à propos de la distribution de σ_i. Celle-ci est donc résumée par l'observation certaine $\zeta_i = 1$. Ceci peut être interprété comme une perturbation de la vraisemblance du modèle [81, 84] ou, dans le cadre d'une loi de la forme (2.1), par une modification des champs locaux Φ_i.

- les *observations incertaines* (« soft evidence ») qui consistent à imposer les contraintes suivantes :

$$\forall i \in \mathbb{V}^* \subset \mathbb{V}, \quad \mathbb{P}_{\sigma_i} = \mathbf{b}_i^*, \tag{4.1}$$

c'est-à-dire que l'on impose que la distribution marginale de σ_i soit conforme à notre observation $\mathbf{b}_i^*$.

Dans les deux cas, au lieu d'observer l'état de $\boldsymbol{\sigma}_{\mathbb{V}^*}$, on obtient une information sur la distribution des variables $\sigma_i, i \in \mathbb{V}^*$. Les observations virtuelles ont l'avantage d'avoir un sens probabiliste bien compris puisque l'on définit tout simplement un nouveau modèle où les observations sont inclues dans de nouvelles variables. On ne contrôle cependant pas l'effet exact qu'aura cette perturbation du modèle. En particulier, on ne connaît pas à l'avance la valeur des marginales $\mathbb{P}_{\sigma_i}$ pour $i \in \mathbb{V}^*$ du nouveau modèle [12].

Au contraire, les observations incertaines conduisent à rechercher une nouvelle distribution $\mathbb{P}'_{\boldsymbol{\sigma}}$ comme perturbation de $\mathbb{P}_{\boldsymbol{\sigma}}$, telle que les marginales $\mathbb{P}'_{\sigma_i}$ soit conformes à nos observations. Des travaux préliminaires [33] comparant les deux approches suggèrent que l'utilisation du second type conduit à de meilleurs résultats pour l'application qui nous intéresse. Notons que ce type d'*observation incertaine* est fréquemment utilisé dans un contexte bayésien [12, 15, 55]. Il s'agit d'une vision plus variationnelle : on cherche la distribution la plus proche, en un sens décrit par Chan et Darwiche [15], de $\mathbb{P}_{\boldsymbol{\sigma}}$ vérifiant les contraintes (4.1).

Dans la suite de ce chapitre, on supposera que les distributions $\{\mathbf{b}_i^*, i \in \mathbb{V}^*\}$ sont valides et en particulier normalisées :

$$\sum_{s_i=0}^{q_i-1} b_i^*(s_i) = 1.$$

On construit dans la section 4.1 un algorithme itératif de passage de messages, *mirror BP* (mBP), pour résoudre le problème variationnel (2.13) auquel sont ajoutées les contraintes (4.1). On prouve que, pour toute une classe de graphes et de répartitions des observations, l'algorithme mirror BP converge vers un unique point fixe (proposition 4.1) et que, pour une classe plus grande, l'algorithme admet un unique point fixe (proposition 4.3). Dans la section 4.1.3 on réalise quelques expérimentations numériques dans des cas simples, qui montrent que l'algorithme converge en général rapidement.

La section 4.2 s'intéresse à l'utilisation qui peut être faite d'un degré de confiance associé aux observations (4.1). Ceci aboutit à un algorithme qui

consiste à utiliser comme équations de mise à jour une simple pondération des mises à jour de BP classique et de mBP.

Une autre approche (section 4.3) est de construire un algorithme résolvant le problème variationnel de manière approchée en minimisant une énergie libre de Bethe moyenne. L'algorithme obtenu a l'avantage d'avoir les mêmes propriétés de convergence que BP avec observations certaines. Malheureusement, la gestion des dépendances entre variables observées doit être faite manuellement, sous peine de dégrader fortement les résultats (figure 4.8 page 96).

4.1 Contraintes fortes : mirror BP

Ici, notre objectif est de construire une variante de BP, à partir de la minimisation de l'énergie libre de Bethe associée à $\mathbb{P}_{\boldsymbol{\sigma}}$ (2.2), en imposant des contraintes additionnelles (4.1) correspondant à des observations incertaines. Pour cela, on va dérouler un raisonnement analogue à celui de la section 2.3.

4.1.1 Construction des règles de mises à jour

Commençons par exprimer le problème de minimisation que notre algorithme va résoudre :

$$\min_{\mathbf{b}\in\mathcal{B}(\mathcal{G},\mathbb{V}^*)} F_{\text{Bethe}}(\mathbf{b}), \tag{4.2}$$

avec F_{Bethe} l'énergie libre de Bethe telle que définie en (2.12), dont on rappelle la définition :

$$F_{\text{Bethe}}(\mathbf{b}) \stackrel{\text{def}}{=} \sum_{a\in\mathbb{F}}\sum_{\mathbf{s}_a} b_a(\mathbf{s}_a)\log\frac{b_a(\mathbf{s}_a)}{\Psi_a(\mathbf{s}_a)} + \sum_{i\in\mathbb{V}}\sum_{s_i} b_i(s_i)\log\frac{b_i(s_i)^{1-d_i}}{\Phi_i(s_i)}.$$

L'approximation de Bethe est incluse dans la définition de F_{Bethe}, comme on l'a déjà vu dans la section 2.3.2. L'ensemble des mesures $\mathcal{B}(\mathcal{G},\mathbb{V}^*)$ considéré est le suivant :

$$\mathcal{B}(\mathcal{G},\mathbb{V}^*) \stackrel{\text{def}}{=} \left\{ \mathbf{b} \;\middle|\; \begin{array}{l} \forall ai\in\mathbb{E}|i\notin\mathbb{V}^*, \sum_{\mathbf{s}_a\setminus s_i} b_a(\mathbf{s}_a) = b_i(s_i) \\ \forall ai\in\mathbb{E}|i\in\mathbb{V}^*, \sum_{\mathbf{s}_a\setminus s_i} b_a(\mathbf{s}_a) = b_i^*(s_i) \\ \forall a\in\mathbb{F}, \sum_{\mathbf{s}_a} b_a(\mathbf{s}_a) = 1 \\ \forall i\in\mathbb{V}, \sum_{s_i} b_i(s_i) = 1 \end{array} \right\}.$$

Il s'agit tout simplement du même ensemble de contraintes que celui utilisé pour l'algorithme BP classique auquel on a ajouté les contraintes (4.1). Écri-

vons le lagrangien associé au problème de minimisation (4.2) :

$$\begin{aligned}\mathcal{L}(\mathbf{b},\boldsymbol{\lambda},\boldsymbol{\gamma}) = F_{\text{Bethe}}(\mathbf{b}) &+ \sum_{\substack{i\in\mathbb{V}\setminus\mathbb{V}^*\\ a\ni i,s_i}} \lambda_{ai}(s_i)\left(b_i(s_i) - \sum_{\mathbf{s}_{a\setminus i}} b_a(\mathbf{s}_a)\right)\\ &+ \sum_{\substack{i\in\mathbb{V}^*\\ a\ni i,s_i}} \lambda_{ai}(s_i)\left(b^*_i(s_i) - \sum_{\mathbf{s}_{a\setminus i}} b_a(\mathbf{s}_a)\right)\\ &+ \sum_{a\in\mathbb{F}} \gamma_a\left(\sum_{\mathbf{s}_a} b_a(\mathbf{s}_a) - 1\right) + \sum_{i\in\mathbb{V}} \gamma_i\left(\sum_{s_i} b_i(s_i) - 1\right).\end{aligned}$$

Les points stationnaires ($\partial\mathcal{L}(\mathbf{b},\boldsymbol{\lambda},\boldsymbol{\gamma})/\partial b_a(\mathbf{s}_a) = 0$ et $\partial\mathcal{L}(\mathbf{b},\boldsymbol{\lambda},\boldsymbol{\gamma})/\partial b_i(s_i) = 0$) du lagrangien correspondent à

$$\begin{cases} b_a(\mathbf{s}_a) = \Psi_a(\mathbf{s}_a)\exp\left(\sum_{i\in a}\lambda_{ai}(s_i) - 1\right),\ \forall a\in\mathbb{F},\\ b_i(s_i) = \Phi_i(s_i)\exp\left(\frac{\sum_{a\ni i}\lambda_{ai}(s_i)}{d_i - 1} + 1 - \gamma_i\right), \forall i\in\mathbb{V}\setminus\mathbb{V}^* \mid d_i \neq 1.\end{cases}$$

La condition $d_i \neq 1$ est due au fait que, pour $i \in \mathbb{V}\setminus\mathbb{V}^*$ tel que $d_i = 1$, on a $\partial F_{\text{Bethe}}(\mathbf{b})/\partial b_i(s_i) = 0$ et donc $\mathbf{b}_i$ n'intervient pas dans la fonction à minimiser et doit être obtenu via les contraintes $\sum_{\mathbf{s}_a\setminus s_i} b_a(\mathbf{s}_a) = b_i(s_i)$. Pour $i \in \mathbb{V}^*$, les beliefs $\mathbf{b}_i$ ne font plus partie des variables de l'optimisation mais, par convention, on les définit selon (4.1). Comme Yedidia *et al.* [111] et comme dans le chapitre 2, on introduit la paramétrisation suivante :

$$\lambda_{ai}(s_i) \stackrel{\text{def}}{=} \log n_{i\to a}(s_i),$$

pour tout arc ai de $\mathcal{G}$. On pose de plus, pour tout sommet $i \notin \mathbb{V}^*$,

$$n_{i\to a}(s_i) \stackrel{\text{def}}{=} \Phi_i(s_i)\prod_{b\ni i, b\neq a} m_{b\to i}(s_i),$$

la relation classique entre les messages $\mathbf{m}$ et $\mathbf{n}$. Imposer la compatibilité $\sum_{\mathbf{s}_a\setminus s_i} b_a(\mathbf{s}_a) = b_i(s_i)$ pour un sommet $i \notin \mathbb{V}^*$ conduit tout naturellement à la formule de mises à jour (2.4) de l'algorithme BP

$$m_{a\to i}(s_i) \propto \sum_{\mathbf{s}_{a\setminus i}} \Psi_a(\mathbf{s}_a)\prod_{j\in a, j\neq i} n_{j\to a}(s_j) \tag{4.3}$$

qui reste donc valide pour tout sommet $i \notin \mathbb{V}^*$. Intéressons nous maintenant aux contraintes de compatibilité à un sommet de $i \in \mathbb{V}^*$:

$$\begin{aligned} b^*_i(s_i) &= \sum_{\mathbf{s}_{a\setminus i}} b_a(\mathbf{s}_a) = \sum_{\mathbf{s}_{a\setminus i}} \Psi_a(\mathbf{s}_a)\prod_{j\in a} n_{j\to a}(s_j)\\ &= n_{i\to a}(s_i)\sum_{\mathbf{s}_{a\setminus i}} \Psi_a(\mathbf{s}_a)\prod_{j\in a, j\neq i} n_{j\to a}(s_j),\end{aligned}$$

ce qui nous fournit l'équation de mise à jour pour $n_{i\to a}(x_i)$. Jusqu'à présent, nous n'avons pas défini de messages $\mathbf{m}_{a\to i}$ envoyés par un facteur a à une variable $i \in \mathbb{V}^*$. Par commodité, on définit ces messages selon (4.3) ; la même formule de mise à jour est donc utilisée pour tous les messages $\mathbf{m}_{a\to i}$. L'équation de compatibilité précédente devient alors

$$b_i^*(s_i) = n_{i\to a}(s_i) m_{a\to i}(s_i),$$

comme pour l'algorithme BP classique. Cette identité n'est plus une conséquence mais la formule de mise à jour qui remplace (2.3) lorsque $i \in \mathbb{V}^*$

$$n_{i\to a}(s_i) \leftarrow \frac{b_i^*(s_i)}{m_{a\to i}(s_i)}. \tag{4.4}$$

Il est en fait possible de ré-interpréter la formule de mise à jour (4.4), en remarquant que

$$\frac{b_i^*(s_i)}{m_{a\to i}(s_i)} = \frac{b_i^*(s_i)}{b_i(s_i)} \Phi_i(s_i) \prod_{b\ni i, b\neq a} m_{b\to i}(s_i),$$

c'est-à-dire que le message (4.4) qui est envoyé par une variable observée, est en fait égal à celui qui serait envoyé dans le cas sans observation (2.3), multiplié par le rapport du belief observé $b_i^*(s_i)$ sur le « belief courant $b_i(s_i)$ » calculé via (2.5). Ce type de mise à jour correspond à l'algorithme « iterative proportional fitting » (IPF, [23]). On reviendra sur ce point dans la section 4.1.4.

Résumons rapidement les propriétés de la variante de BP que l'on vient de construire :

- tous les facteurs et les variables dont la distribution n'a pas été fixée envoient des messages calculés selon les formules (2.3)–(2.4) de l'algorithme BP classique ;
- les variables dont la distribution est fixée envoient des messages calculés selon (4.4) ;
- les beliefs $\mathbf{b}_a$ et $\mathbf{b}_i$ pour $i \notin \mathbb{V}^*$, c'est à dire non fixés, sont calculés en utilisant (2.6) et (2.5) ;
- la formule (2.5) n'a plus de sens pour les sommets $i \in \mathbb{V}^*$, la valeur fixée $\mathbf{b}_i^*$ doit être utilisée à la place.

Dans la version classique de BP, l'information envoyée par un sommet du graphe ne peut lui revenir que le long d'un cycle du graphe. Cependant, lorsque la formule de mise à jour (4.4) est utilisée, la variable dont le belief est imposé se comporte comme un miroir et renvoie un message dépendant uniquement du message reçu au lieu de propager l'information dans le graphe. C'est ce comportement qui a inspiré le nom *mirror BP* (mBP) que l'on donne à cet algorithme. Une représentation formelle, non optimisée, de l'algorithme est donnée (algorithme 2).

Algorithme 2 : *mirror BP* (mBP)

Données : $\mathcal{H} = (\mathbb{V}, \mathbb{F})$, N_{max} et ε.
initialisation des messages $\mathbf{m}^{(0)}$ et $\mathbf{n}^{(0)}$;
$\delta = +\infty$ et $n = 0$;
tant que $\delta > \varepsilon$ *et* $n < N_{max}$ **faire**
  $\delta \leftarrow 0$;
  pour $i \in \mathbb{V}$ **faire**
    pour $a \ni i, s_i \in \{0, \ldots, q_i - 1\}$ **faire**
      si $i \in \mathbb{V}^*$ **alors**
        $n_{i\to a}^{(n+1)}(s_i) \leftarrow {}^{b_i^*(s_i)}/_{m_{a\to i}(s_i)}$;
      sinon
        $n_{i\to a}^{(n+1)}(s_i) \leftarrow \Phi_i(s_i) \prod_{\substack{c\ni i \\ c\neq a}} m_{c\to i}(s_i)$;
  pour $a \in \mathbb{F}$ **faire**
    pour $i \in a$ **faire**
      pour $s_i \in \{0, \ldots, q_i - 1\}$ **faire**
        $m_{a\to i}^{(n+1)}(s_i) \leftarrow \sum_{\mathbf{s}_a \setminus s_i} \Psi_a(\mathbf{s}_a) \prod_{j\in a} n_{j\to a}(s_j)$;
      $\mathbf{m}_{a\to i}^{(n+1)} \leftarrow {}^{\mathbf{m}_{a\to i}^{(n+1)}}/_{\|\mathbf{m}_{a\to i}^{(n+1)}\|}$;
      $\delta = \max(\delta, \|\mathbf{m}_{a\to i}^{(n+1)} - \mathbf{m}_{a\to i}^{(n)}\|_\infty)$;
  $n \leftarrow n + 1$;
si $\delta < \varepsilon$ **alors**
  pour $i \in \mathbb{V}, s_i$ **faire**
    si $i \in \mathbb{V}^*$ **alors**
      $b_i(s_i) = b_i^*(s_i)$;
    sinon
      $b_i(s_i) = \frac{1}{Z_i(\mathbf{m})} \Phi(s_i) \prod_{a\ni i} m_{a\to i}(s_i)$;
  pour $a \in \mathbb{F}, \mathbf{s}_a$ **faire**
    $b_a(\mathbf{s}_a) = \frac{1}{Z_a(\mathbf{m})} \Psi_a(\mathbf{s}_a) \prod_{j\in a} n_{j\to a}(s_j)$;

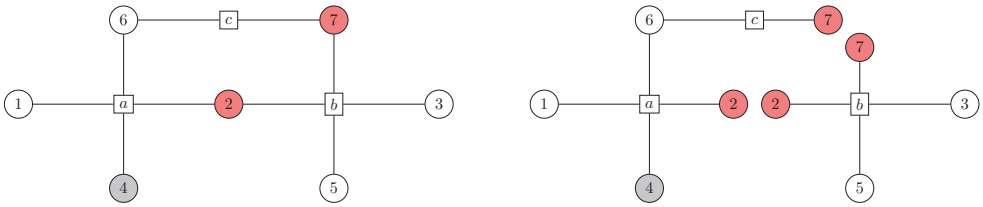


FIGURE 4.1 : Illustration de la $\mathbb{V}^*$-segmentation du graphe, où $\mathbb{V}^*$ correspond à l'ensemble des sommets de couleur rouge ou grise. Le graphe de facteurs $\mathcal{G}$ d'origine est représenté à gauche tandis que le graphe résultant est représenté à droite.

Un algorithme semblable à mBP, bien que construit de manière différente, a déjà été proposé par Teh et Welling [92]. On décrira les différences entre ces deux algorithmes dans la section 4.1.4.

4.1.2 Convergence de l'algorithme

On sait que BP peut ne pas converger sur des graphes cycliques [79]. Même si certaines conditions suffisantes de convergence de l'algorithme sont connues [77, 91, 50], la convergence de BP dans les graphes cycliques reste un problème ouvert. Étant donné que le comportement de notre mBP est assez différent de celui de BP, il est important d'obtenir des garanties de convergence dans des cas particuliers.

Introduisons tout d'abord une transformation du graphe induite par $\mathbb{V}^*$ qui nous permettra de simplifier l'étude.

Définition 4.1. *Soit $\mathcal{T}(\mathcal{G}, \mathbb{V}^*)$ le graphe de facteurs dans lequel chaque sommet $i \in \mathbb{V}^*$ a été répliqué d_i fois, chaque réplique étant reliée à un (et un seul) voisin de i. On appelle « $\mathbb{V}^*$-segmentation du graphe » la transformation $\mathcal{T}(\cdot, \mathbb{V}^*)$ appliquée au graphe de facteurs $\mathcal{G}$ pour un ensemble de sommets variables $\mathbb{V}^*$.*

Un exemple d'une telle segmentation $\mathcal{T}(\mathcal{G}, \mathbb{V}^*)$, avec $\mathbb{V}^*$ l'ensemble des sommets rouges ou gris, est présenté dans la Figure 4.1. La proposition suivante décrit certains cas où la convergence de mBP est garantie.

Proposition 4.1. *Dans le cas de variables σ_i binaires ($q = 2$), si le graphe $\mathcal{T}(\mathcal{G}, \mathbb{V}^*)$ est formé d'arbres déconnectés contenant chacun au plus deux feuilles issues de $\mathbb{V}^*$, l'algorithme mBP défini précédemment est stable et converge vers un point fixe unique.*

Avant de passer à la preuve de la proposition 4.1, considérons l'exemple de la figure 4.1. Lorsque les sommets en rouge correspondent à $\mathbb{V}^*$ la proposition 4.1 nous permet de conclure que mBP converge puisque le graphe

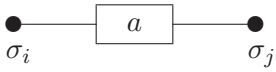


FIGURE 4.2 : Un facteur a contenant deux variables σ_i et σ_j qui sont toutes deux observées.

résultant $\mathcal{T}(\mathcal{G}, \mathbb{V}^*)$ (à droite) est constitué de deux arbres déconnectés contenant chacun deux sommets de $\mathbb{V}^*$. Si l'on ajoute le sommet en gris à $\mathbb{V}^*$, alors la proposition 4.1 ne s'applique plus ; l'arbre de gauche contenant trois sommets de $\mathbb{V}^*$, on ne peut alors pas conclure au sujet de la convergence. Cependant mBP converge sur l'autre partie du graphe par application de la proposition 4.1.

Regardons tout d'abord le cas d'un facteur a contenant deux variables binaires σ_i et σ_j, toutes deux observées (figure 4.2). On suppose que l'on normalise les messages $\mathbf{m}_{a\to i}$ de sorte que

$$m_{a\to i}(0) + m_{a\to i}(1) = 1.$$

Pour étudier la convergence des messages, on introduit la notation suivante

$$u_n \stackrel{\text{def}}{=} m_{a\to i}(0), \quad v_n \stackrel{\text{def}}{=} m_{a\to j}(0),$$

et on a alors $1 - u_n = m_{a\to i}(1)$ et $1 - v_n = m_{a\to j}(1)$. Les règles de mise à jour (2.4) et (4.4) induisent les formules de récurrence suivantes :

$$\begin{bmatrix} u_{n+1} \\ v_{n+1} \end{bmatrix} = \begin{bmatrix} \dfrac{\Psi_{00}\alpha_j\bar{v}_n + \Psi_{01}\bar{\alpha}_j v_n}{(\Psi_{00}+\Psi_{10})\alpha_j\bar{v}_n + (\Psi_{01}+\Psi_{11})\bar{\alpha}_j v_n} \\ \dfrac{\Psi_{00}\alpha_i\bar{u}_n + \Psi_{10}\bar{\alpha}_i u_n}{(\Psi_{00}+\Psi_{01})\alpha_i\bar{u}_n + (\psi_{10}+\Psi_{11})\bar{\alpha}_i u_n} \end{bmatrix} \tag{4.5}$$

avec $\alpha_i \stackrel{\text{def}}{=} b_i^*(0)$, $\Psi_{xy} \stackrel{\text{def}}{=} \Psi_{ij}(\sigma_i = x, \sigma_j = y)$ et en utilisant la convention $\bar{z} \stackrel{\text{def}}{=} 1 - z$.

Lemme 4.2. *Les suites $(u_n)_{n\in\mathbb{N}}$ et $(v_n)_{n\in\mathbb{N}}$, définies par l'équation de récurrence* (4.5)*, convergent vers un unique point fixe $\forall (u_0, v_0) \in]0,1[^2$.*

Démonstration. Les rôles de u_n et v_n étant symétriques, on prouvera seulement la convergence de u_n. En utilisant (4.5), on obtient une équation de récurrence de la forme $u_{n+2} = f(u_n)$ avec

$$f(x) = \frac{h_0 x + K_0}{(h_0 + h_1)x + (K_0 + K_1)},$$

où

$$\begin{aligned} h_0 &\stackrel{\text{def}}{=} \Psi_{00}\alpha_j(\bar{\alpha}_i\Psi_{11} - \alpha_i\Psi_{01}) + \Psi_{01}\bar{\alpha}_j(\Psi_{10}\bar{\alpha}_i - \Psi_{00}\alpha_i), \\ h_1 &\stackrel{\text{def}}{=} \Psi_{10}\alpha_j(\bar{\alpha}_i\Psi_{11} - \alpha_i\Psi_{01}) + \Psi_{11}\bar{\alpha}_j(\Psi_{10}\bar{\alpha}_i - \Psi_{00}\alpha_i), \\ K_0 &\stackrel{\text{def}}{=} \Psi_{00}\Psi_{01}\alpha_i, \quad K_1 \stackrel{\text{def}}{=} \alpha_i(\Psi_{10}\Psi_{01}\alpha_j + \Psi_{11}\Psi_{00}\bar{\alpha}_j). \end{aligned}$$

La dérivée de f est donc

$$f'(x) = \frac{h_0 K_1 - h_1 K_0}{\left((h_0 + h_1)x + (K_0 + K_1)\right)^2},$$

qui est de signe constant. Si $f'(x) \geq 0$ les suites $(u_{2n})_{n\in\mathbb{N}}$ et $(u_{2n+1})_{n\in\mathbb{N}}$ sont monotones. La suite u_n étant bornée, les deux suites convergent. Prouvons maintenant qu'il existe un unique point fixe de f dans l'intervalle $[0, 1]$, ce qui prouvera que les limites des suites $(u_{2n})_{n\in\mathbb{N}}$ et $(u_{2n+1})_{n\in\mathbb{N}}$ sont égales et donc que la suite $(u_n)_{n\in\mathbb{N}}$ converge.

Commençons par étudier certains cas triviaux. Lorsque $f(1) = 1$, on a alors $\alpha_i = 1$ et

$$\begin{aligned} h_0 &= -\psi_{00}\psi_{01} = -K_0, \\ h_1 &= -\psi_{10}\psi_{01}\alpha_j - \psi_{11}\psi_{00}\bar{\alpha}_j = -K_1, \end{aligned}$$

f est donc une fonction constante égale à $\frac{K_0}{K_0+K_1}$. Si $f(0) = 0$, on a alors $\alpha_i = K_0 = K_1 = 0$, et f est encore constante. Les cas $f(1) = 0$ et $f(0) = 1$ sont traités de la même manière et f est encore une fonction constante, ce qui implique trivialement que la suite u_n converge.

Cas 1 : f est croissante. On sait qu'il existe au moins un point fixe dans $[0, 1]$ puisque $f([0, 1]) \subset [0, 1]$. L'étude des racines de $f(x) - x$ montre que le nombre de points fixes est au plus deux car il s'agit des racines d'un polynôme de degré deux.

De plus, sachant que $f(0) > 0$, f croissante et $f(1) < 1$ le nombre de points fixes doit être impair. En effet, la courbe de f doit croiser un nombre impair de fois la première bissectrice. On peut donc conclure qu'il existe un unique point fixe sur $[0, 1]$, et que les suites $(u_{2n})_{n\in\mathbb{N}}$ et $(u_{2n+1})_{n\in\mathbb{N}}$ convergent vers ce point fixe.

Cas 2 : f est décroissante. On considère alors la suite $(1-u_n)_{n\in\mathbb{N}}$, qui est similaire, mais est définie par une récurrence de la forme $1 - u_{n+2} = g(1 - u_n)$ avec une fonction g telle que g' est positive. On applique alors le résultat du cas 1 pour conclure. □

Le cas que l'on vient d'étudier est en fait plus général qu'il n'y paraît. En effet, dès qu'un arbre comprend exactement deux sommets où les beliefs sont fixés, la situation est équivalente à celle du lemme 4.2 comme l'indique la proposition 4.1.

Preuve de la proposition 4.1. Tout d'abord, on se rend facilement compte que fixer les beliefs d'un ensemble de sommets $\mathbb{V}^* \subset \mathbb{V}$ a exactement le même effet que la $\mathbb{V}^*$-segmentation $\mathcal{T}(\cdot, \mathbb{V}^*)$ du point de vue de la propagation des messages. En effet, la forme des mises à jour (4.4) indique que tout se passe comme

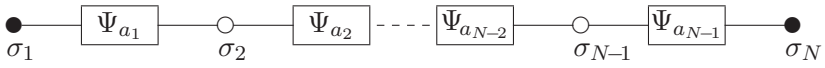


FIGURE 4.3 : Chaîne de N facteurs de paires ; les variables correspondant aux sommets extrêmes σ_1 et σ_N sont observées.

si le graphe était segmenté à chaque sommet de $\mathbb{V}^*$. Pour conclure la preuve, il suffit alors d'étudier la convergence de mBP sur un arbre contenant deux feuilles issues de $\mathbb{V}^*$. Considérons la chaîne de la figure 4.3. On peut montrer qu'elle est équivalente à la figure 4.2 page 82 pour une fonction $\widetilde{\Psi}(s_1, s_N)$ bien choisie. En propageant les règles de mises à jour, on obtient en effet

$$m_{a_1 \to 1}(s_1) \propto \sum_{s_N} \left(\sum_{s_1 \dots s_{N-2}} \prod_{i=1}^{N-2} \Psi_{a_i}(\mathbf{s}_{a_i}) \Phi_i(s_i) \right) \frac{\Psi_{a_{N-1}}(\mathbf{s}_{a_{N-1}}) b_r^*(s_N)}{m_{a_{N-1} \to N}(s_N)},$$

de telle sorte que

$$\widetilde{\Psi}(s_1, s_N) \stackrel{\text{def}}{=} \left(\sum_{s_1 \dots s_{N-2}} \prod_{i=1}^{N-2} \Psi_{a_i}(\mathbf{s}_{a_i}) \Phi_i(s_i) \right) \Psi_{a_{N-1}}(\mathbf{s}_{a_{N-1}}),$$

et l'on utilise alors le résultat du lemme 4.2 pour obtenir la convergence des messages sur cette chaîne de facteurs. Dans le cas général d'un arbre avec deux feuilles dans $\mathbb{V}^*$, on remarque que les sous-arbres reliés aux variables $\sigma_i, i \in \{2 \dots N-1\}$ ou aux facteurs a_i envoient des messages indépendants de ce qui se passe sur la chaîne et peuvent donc être absorbés par une redéfinition des Φ_i ou Ψ_{a_i}. En fait, le graphe étant un arbre, on sait que l'information envoyée par σ_1 et σ_N à ces parties ne reviendra pas en σ_1 où σ_N. À une redéfinition près des fonctions Φ et Ψ près, le cas d'un arbre dont deux feuilles sont issues de $\mathbb{V}^*$ est donc équivalent à celui du lemme 4.2. □

On considère maintenant le cas plus général des arbres où le nombre de feuilles appartenant à $\mathbb{V}^*$ est quelconque. La proposition suivante indique que, dans le cas où $\mathcal{T}(\mathcal{G}, \mathbb{V}^*)$ se réduit à des arbres, si l'algorithme mBP converge alors il converge vers le minimum global de (4.2). Teh et Welling [92] montrent que dans ce cas il existe un ordre de mise à jour des messages tel que l'algorithme converge, mais n'explicitent pas cet ordre.

Proposition 4.3. *Lorsque le graphe résultant $\mathcal{T}(\mathcal{G}, \mathbb{V}^*)$ est constitué d'un ensemble d'arbres déconnectés, le problème variationnel* (4.2) *admet une unique solution.*

Si l'on revient au cas de la figure 4.1 page 81 lorsque $\mathbb{V}^*$ est constitué des sommets rouges ou gris, la proposition 4.3 indique qu'il existe un unique point fixe de mBP.

Démonstration. La propagation des messages sur chaque arbre est indépendante ; on peut donc regarder indépendamment les problèmes variationnels sur chaque arbre. Par le même argument que dans la preuve de la proposition 4.1 l'étude d'un arbre dont n feuilles appartiennent à $\mathbb{V}^*$ se ramène à l'étude d'un facteur relié à n sommets de $\mathbb{V}^*$. L'énergie libre de Bethe s'exprime alors comme

$$F_{\text{Bethe}}(\mathbf{b}) = \sum_{\mathbf{s}_a} b_a(\mathbf{s}_a) \log \frac{b_a(\mathbf{s}_a)}{\Psi_a(\mathbf{s}_a)},$$

avec une fonction Ψ_a bien choisie. La matrice hessienne $\nabla^2 F_{\text{Bethe}}(\mathbf{b})$ est donc la matrice diagonale des $\frac{1}{b_a(\mathbf{s}_a)}$, pour $\mathbf{s}_a \in \{0, \dots, q-1\}^{d_a}$. Les beliefs vérifiant la relation $b_a(\mathbf{s}_a) \leq 1$, la matrice $\nabla^2 F_{\text{Bethe}}(\mathbf{b}) - \mathbf{I}$ est toujours définie positive et F_{Bethe} est strictement convexe. Les contraintes de (4.2) étant linéaires, un résultat classique d'optimisation convexe [13, p.137] indique que (4.2) admet au plus un minimum local qui est donc global. F_{Bethe} étant strictement positive et à valeur dans un compact, elle atteint sa borne inférieure et l'on obtient le résultat. □

4.1.3 Expérimentations numériques

Les résultats de convergence obtenus dans la partie précédente sont très partiels, et en particulier, ils ne donnent pas d'indication sur la vitesse de convergence. On propose quelques expérimentations numériques pour vérifier l'applicabilité de l'algorithme mBP.

On considère différents cas qui se ramènent tous à celui d'un unique facteur a relié à d_a sommets dont toutes les variables ont leurs beliefs fixés selon (4.1). Notons que dans ce cas précis, la fonction Ψ_a représente exactement la loi du vecteur $\boldsymbol{\sigma}_a$ en l'absence d'information. Pour pouvoir représenter une grande variété de cas possibles, on répète 10 000 fois l'expérience suivante :

- on génère aléatoirement une fonction de compatibilité Ψ_a constituée de $(d_a)^q$ nombres positifs de somme égale à 1. Ceci peut être fait à l'aide d'une loi de Dirichlet $\mathcal{D}((d_a)^q)$;
- on génère aléatoirement d_a beliefs $\mathbf{b}_i^*$, chacun étant constitué de q nombres positifs dont la somme vaut 1. Pour cela, on utilise une loi de Dirichlet $\mathcal{D}(q)$.

On rappelle qu'une loi de Dirichlet d'ordre K et de paramètre $\boldsymbol{\theta}$ a pour densité

$$f_{\boldsymbol{\theta},K}(\mathbf{x}) = \frac{1}{B(\boldsymbol{\theta})} \prod_{i=1}^{K} x_i^{\theta_i - 1} \mathbb{1}_{[0,1]}(x_i), \quad \text{si } \sum_{i=1}^{K} x_i = 1,$$

et 0 sinon. On s'intéresse ici uniquement au cas où tous les paramètres θ_i sont égaux à 1, ce qui correspond à une distribution uniforme sur le simplexe de dimension $K-1$, et on note $\mathcal{D}(K)$ une telle distribution. Les figures des trois cas étudiés ici (figures 4.4 page 87, 4.5 page 88 et 4.6 page 89) sont bâties sur le même principe. On étudiera pour chaque cas deux types de mises à jour :

- *parallèles*, pour lesquelles $\mathbf{m}_{a\to i}^{(n+1)}$ ne dépend que des messages $\mathbf{m}_{a\to j}^{(n)}$,
- *asynchrones*, pour lesquelles lors du calcul de $\mathbf{m}_{a\to i}^{(n+1)}$, on utilise $\mathbf{m}_{a\to j}^{(n+1)}$ lorsqu'il a déjà été calculé et $\mathbf{m}_{a\to j}^{(n)}$ sinon.

On note N_p et N_a le nombre d'itérations nécessaires à la convergence dans le cas de mises à jour parallèles et asynchrones. La force de l'interaction induite par la fonction Ψ_a sera quantifiée par l'information mutuelle $\mathrm{I}(\Psi_a)$ entre les variables en l'absence d'observation ; elle est définie comme suit

$$\mathrm{I}(\Psi_a) \stackrel{\text{def}}{=} \sum_{\mathbf{s}_a} \Psi_a(\mathbf{s}_a) \log_q \left(\frac{\Psi_a(\mathbf{s}_a)}{\prod_{j\in a} \left(\sum_{\mathbf{s}_{a\setminus j}} \Psi_a(\mathbf{s}_a) \right)} \right),$$

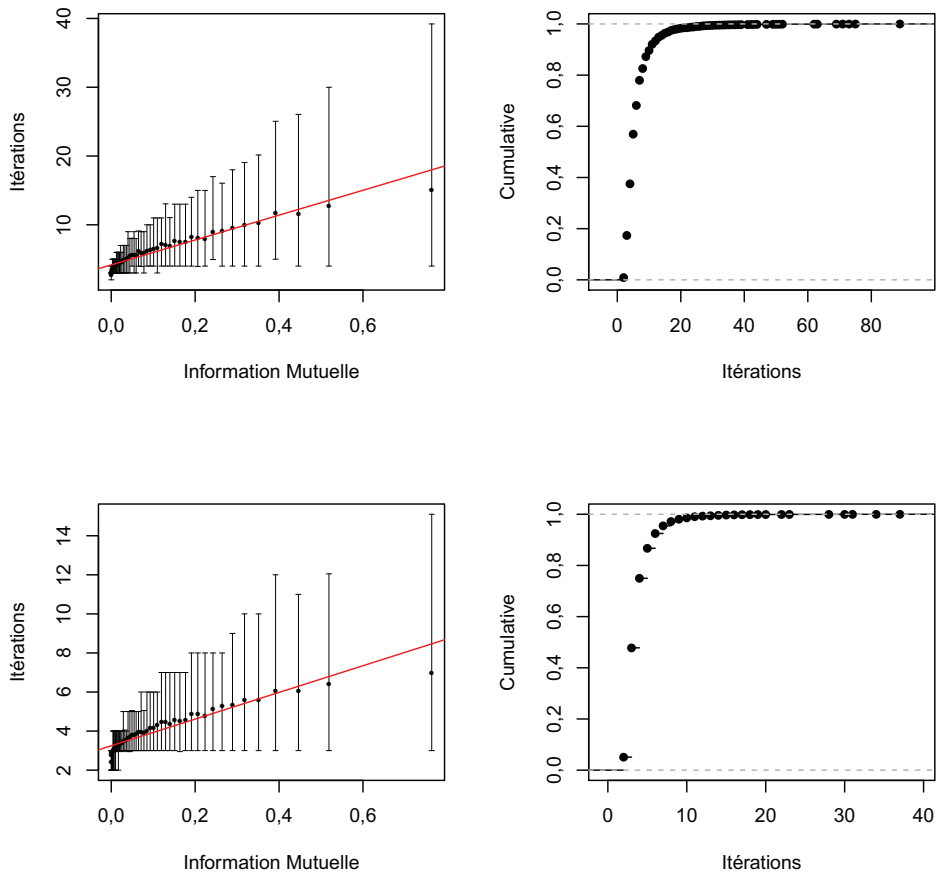
où l'on suppose que Ψ_a est normalisée, *i.e.* $\sum_{\mathbf{s}_a} \Psi_a(\mathbf{s}_a) = 1$.

La figure 4.4 page suivante correspond au cas du lemme 4.2, pour lequel la convergence est prouvée. Ce cas est équivalent à tout cas où $\mathcal{T}(\mathcal{G}, \mathbb{V}^*)$ est constitué d'un arbre dont deux feuilles sont issues de $\mathbb{V}^*$. Commençons par des constations générales qui resteront vraies pour les 2 autres expériences :

- *dans la majorité des cas, la convergence est rapide, comme le montrent les distributions empiriques du nombre d'itérations avant convergence ;*
- *les cas où la convergence se révèle lente correspondent aux cas d'interactions* $\mathrm{I}(\Psi_a)$ *fortes, comme le suggèrent les coefficients de corrélation ;*
- *enfin, les mises à jour asynchrones sont en général plus rapides à converger.*

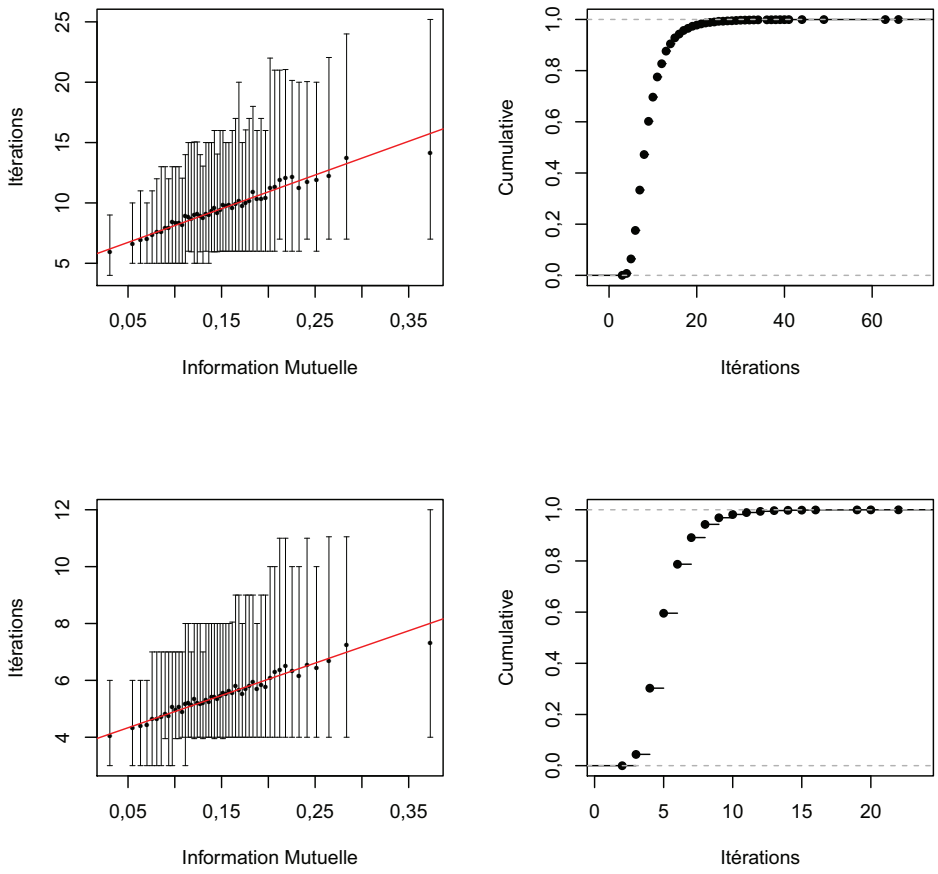
Le troisième point s'explique par le fait que les mises à jour asynchrones permettent de réduire les phénomènes oscillatoires particulièrement visibles dans les suites (u_n) et (v_n) du lemme 4.2.

Considérons maintenant le cas où les variables σ_i et σ_j de la figure 4.2 page 82 ne sont plus binaires mais à valeurs dans un alphabet à $q = 4$ éléments. Les résultats correspondants sont présentés sur la figure 4.5 page 88. On remarque tout d'abord que, même si l'on a pas prouvé la convergence de l'algorithme dans ce cas, on n'observe pas de cas non convergents et les statistiques sont comparables à celle du cas binaire. Lorsque l'on augmente encore q la convergence semble se produire de plus en plus facilement, cela étant certainement dû au fait que, la dimension de l'espace augmentant, il devient de plus en plus difficile de détecter les cas pathologiques où la convergence est lente. Pour finir cette section, on réalise à nouveau la même expérience, mais cette fois dans le cas d'un facteur relié à cinq variables binaires dont les beliefs sont fixés (figure 4.6 page 89). À nouveau, on n'observe pas de cas non convergent.



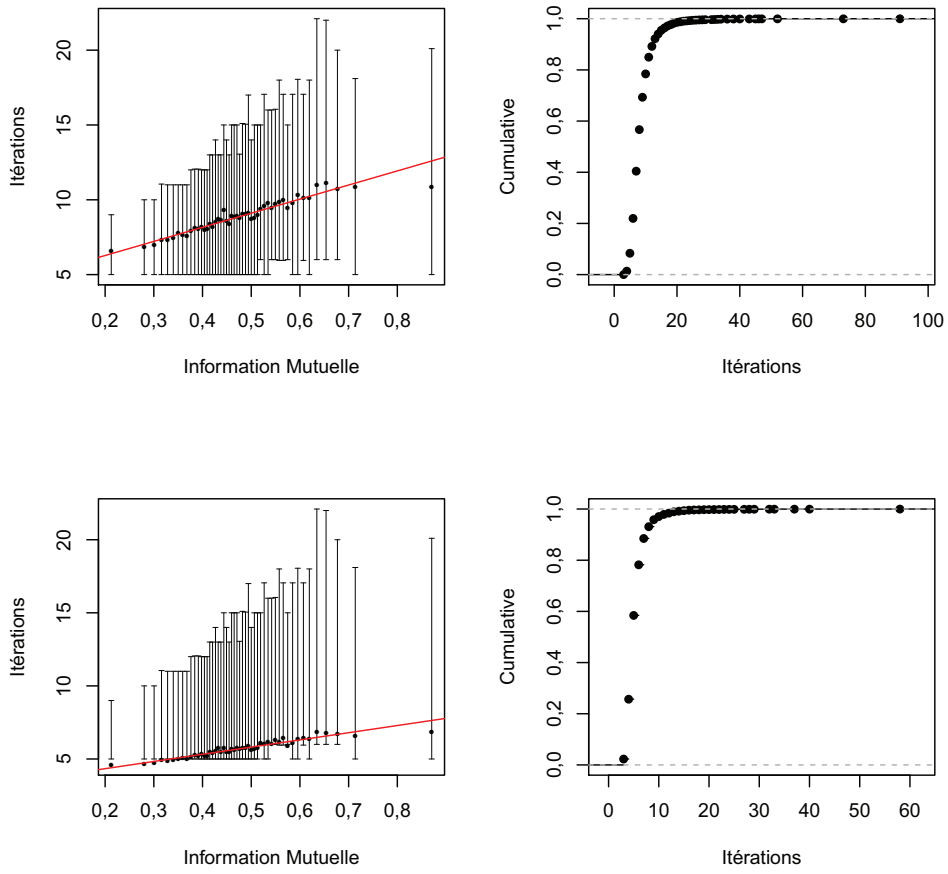
	$q_X^{0,5}$	$q_X^{0,75}$	$q_X^{0,9}$	$q_X^{0,99}$	$\langle X \rangle$	$\hat{\sigma}_X$	ρ	r_s
N_p	5	7	11	25	6,37	4,64	0,58	0,74
N_a	4	5	6	11	4,08	1,89	0,53	0,63

FIGURE 4.4 : On considère ici le cas du lemme 4.2. À gauche, on a subdivisé les points en 50 classes de même poids selon l'information mutuelle $\mathrm{I}(\Psi_a)$. Pour chaque classe, on trace le nombre moyen d'itérations nécessaires à la convergence, à $\varepsilon = 10^{-3}$ près, ainsi que ses quantiles d'ordre 0,05 et 0,95. À droite, les fonctions de répartition empirique du nombre d'itérations. La partie haute correspond aux mises à jour parallèles (N_p) et la partie basse à des mises à jour asynchrones (N_a). Le tableau récapitule quelques statistiques des vitesses de convergence. q_X^p, $\langle X \rangle$ et $\hat{\sigma}_X$ sont, respectivement, les estimations empiriques des quantiles d'ordre p, de la moyenne et de l'écart-type de la variable X. Les quantités ρ et r_s sont les coefficients de corrélation linéaire de Pearson et celui sur les rangs, dit de Spearman.



	$q_X^{0,5}$	$q_X^{0,75}$	$q_X^{0,9}$	$q_X^{0,99}$	$\langle X \rangle$	$\hat{\sigma}_X$	ρ	r_s
N_p	9	11	14	24	9,62	4,02	0,43	0,33
N_a	5	6	8	12	5,51	1,74	0,40	0,31

FIGURE 4.5 : Même chose que la figure 4.4 page précédente, dans le cas d'un facteur contenant deux variables dans $\mathbb{V}^*$, les variables sont ici à valeurs dans un alphabet à $q = 4$ éléments.



	$q_X^{0,5}$	$q_X^{0,75}$	$q_X^{0,9}$	$q_X^{0,99}$	$\langle X \rangle$	$\hat{\sigma}_X$	ρ	r_s
N_p	8	10	13	22	8,88	3,74	0,29	0,31
N_a	5	6	8	14	5,70	2,18	0,26	0,27

FIGURE 4.6 : Même chose que la figure 4.4 page 87, dans le cas d'un facteur contenant cinq variables binaires ($q = 2$) appartenant toutes à $\mathbb{V}^*$.

4.1.4 Comparaison avec « IPF-BP »

L'algorithme mBP décrit dans la Section 4.1 est proche de l'algorithme « IPF-BP » décrit par Teh et Welling [92]. Cet algorithme cherche à résoudre, de manière approchée, un problème de minimisation équivalent à (4.2), dans le cas de variables σ_i binaires et pour des interactions de paire, c'est-à-dire lorsque tous les facteurs sont de degré $d_a = 2$. Bien qu'étant le résultat d'un cheminement différent, notre algorithme mBP est donc une généralisation de l'algorithme IPF-BP, assez évidente après coup. Les équations de mises à jour des deux algorithmes sont identiques et donc leurs ensembles de point fixes le sont aussi. Il y a toutefois une différence importante, peu évidente à première vue : IPF-BP est décrit comme une procédure à 2 étapes.

(i) BP : on exécute l'algorithme BP sur les sommets de $\mathbb{V} \setminus \mathbb{V}^*$ jusqu'à convergence.

(ii) IPF : on effectue les mises à jour correspondant à l'équation (4.4) pour les messages envoyés par les sommets de $\mathbb{V}^*$.

Les étapes (i) et (ii) sont alternées jusqu'à obtenir la convergence globale. Dans la construction de l'algorithme IPF-BP, les mises à jour (4.4) sont obtenues par utilisation directe de l'algorithme IPF.

Le principal avantage de mBP par rapport à IPF-BP consiste à relâcher la contrainte d'un ordre particulier pour les mises à jour, contrainte qui est particulièrement handicapante dans le cadre d'une parallélisation de ces algorithmes. Cet ordre est dû au fait que Teh et Welling considèrent que les mises à jour (4.4) et (2.3) sont de nature différente ce qui n'est pas nécessaire.

4.2 Degré de confiance des observations : une règle de mise à jour robuste

On a supposé jusqu'ici que les observations incertaines (4.1) mènent à une connaissance certaine des distributions $\mathbb{P}_{\sigma_i} = \mathbf{b}_i^*$. En pratique, on n'a généralement pas une confiance absolue dans ces observations. On propose ici d'adapter l'algorithme pour utiliser une information supplémentaire du type suivant : *on associe un « degré de confiance »* $\alpha_i \in [0, 1]$ *à chaque observation* $\mathbb{P}_{\sigma_i} = \mathbf{b}_i^*$. On ne considère pas ici la manière dont ces degrés de confiance sont obtenus. L'algorithme que l'on souhaite construire doit vérifier les contraintes suivantes :

- lorsque $\alpha_i = 0$, l'algorithme est équivalent à la version classique de BP, l'information fournie n'étant pas prise en compte car sans valeur ;
- lorsque $\alpha_i = 1$, l'algorithme est équivalent à mBP puisque l'information est considérée comme certaine.

Un algorithme vérifiant trivialement ces deux propriétés est obtenu en réalisant tout simplement la combinaison convexe, avec poids α_i et $1 - \alpha_i$, des mises à jour des messages $\mathbf{n}_{i \to a}$ correspondant à mBP et BP (les messages

$\mathbf{m}_{a\to i}$ étant identiques pour les deux algorithmes, la combinaison convexe est triviale). On va montrer ici que cet algorithme résout en fait un problème variationnel qui a du sens.

Pour cela, on introduit un vecteur de variables aléatoires binaires $\boldsymbol{\omega} \stackrel{\text{def}}{=} \{\omega_i\}_{i\in\mathbb{V}^*}$, associées aux sommets pour lesquels une observation du type (4.1) est fournie, avec un degré de confiance α_i. L'événement $\{\omega_i = 1\}$ correspond à la validité de l'information $\mathbb{P}_{\sigma_i} = \mathbf{b}_i^*$. La loi de ω_i est alors donnée par $\mathbb{P}(\omega_i = 1) = \alpha_i$. Plus précisément, on va ici considérer F_{Bethe} comme une fonction du vecteur aléatoire $\boldsymbol{\omega}$ telle que

$$F_{\text{Bethe}}(\mathbf{b}, \mathbf{b}_{\mathbb{V}^*})(\boldsymbol{\omega}) = F_{\text{Bethe}}\left(\mathbf{b}, \mathbf{b}^{\boldsymbol{\omega}}_{\mathbb{V}^*}\right),$$

où

$$\mathbf{b}^{\boldsymbol{\omega}}_{\mathbb{V}^*} = \left(\mathbf{b}_i^*\omega_i + \mathbf{b}_i(1-\omega_i)\right)_{i\in\mathbb{V}^*}.$$

On cherche alors à résoudre le problème de minimisation de l'espérance, en $\boldsymbol{\omega}$, de F_{Bethe} :

$$\min_{(\mathbf{b},\mathbf{b}_{\mathbb{V}^*})\in\mathcal{B}(\mathbb{P}_{\boldsymbol{\omega}})} \quad \mathbb{E}_{\boldsymbol{\omega}}\left[F_{\text{Bethe}}(\mathbf{b}, \mathbf{b}^{\boldsymbol{\omega}}_{\mathbb{V}^*})\right]. \tag{4.6}$$

Avec les beliefs $(\mathbf{b}, \mathbf{b}_{\mathbb{V}^*})$ soumis aux contraintes suivantes

$$\mathcal{B}\left(\mathbb{P}_{\boldsymbol{\omega}}\right) \stackrel{\text{def}}{=} \left\{ (\mathbf{b}, \mathbf{b}_{\mathbb{V}^*}) \;\middle|\; \begin{array}{l} \forall ai \in \mathcal{G}, \forall s_i, \sum_{\mathbf{s}_{a\setminus i}} b_a(\mathbf{s}_a) = \mathbb{E}_{\omega_i}\left[b_i^{\omega_i}(s_i)\right] \\ \forall a \in \mathbb{F}, \sum_{\mathbf{s}_a} b_a(\mathbf{s}_a) = 1 \\ \forall i \in \mathbb{V}, \sum_{s_i} \mathbb{E}_{\omega_i}\left[\mathbf{b}_i^{\omega_i}(s_i)\right] = 1 \end{array} \right\}.$$

On se rend facilement compte que si tous les α_i sont nuls, on obtient exactement le problème variationnel (2.13) conduisant à BP ; lorsque tous les α_i valent 1, on obtient alors le problème variationnel (4.2) conduisant à mBP. La minimisation de (4.6) possède donc bien les propriétés que l'on recherche. Dès lors que les mesures $\mathbf{b}_i^*$ sont normalisées, les contraintes $\sum_{s_i} \mathbb{E}_{\omega_i}\left[b_i^{\omega_i}(s_i)\right] = 1$ deviennent tout simplement $\sum_{s_i} b_i(s_i) = 1$, quelle que soit la valeur de $\alpha_i \neq 1$. Comme d'habitude, on écrit le lagrangien correspondant à (4.6)

$$\begin{aligned}\mathcal{L}(\mathbf{b}, \boldsymbol{\lambda}, \boldsymbol{\gamma}) = \mathbb{E}_{\boldsymbol{\omega}}\left[F_{\text{Bethe}}(\mathbf{b}, \mathbf{b}^{\boldsymbol{\omega}}_{\mathbb{V}^*})\right] &+ \sum_{\substack{ai\in\mathcal{G} \\ s_i}} \lambda_{ai}(s_i)\left(\mathbb{E}_{\omega_i}\left[b_i^{\omega_i}(s_i)\right] - \sum_{\mathbf{s}_{a\setminus i}} b_a(\mathbf{s}_a)\right) \\ &- \sum_{i\in\mathbb{V}} \gamma_i\left(\sum_{s_i} b_i(s_i) - 1\right) - \sum_{a\in\mathbb{F}} \gamma_a\left(\sum_{\mathbf{s}_a} b_a(\mathbf{s}_a) - 1\right).\end{aligned}$$

On a bien sûr la relation

$$\mathbb{E}_{\omega_i}\left[b_i^{\omega_i}(s_i)\right] = \alpha_i b_i^*(s_i) + (1-\alpha_i) b_i(s_i),$$

par définition de ω_i. Il ne nous reste alors plus qu'à expliciter la valeur de $\mathbb{E}_{\boldsymbol{\omega}}[F_{\text{Bethe}}(\mathbf{b}, \mathbf{b}^{\boldsymbol{\omega}}_{\mathbb{V}^*})]$. On remarque tout d'abord que

$$\mathbb{E}_{\boldsymbol{\omega}}[F_{\text{Bethe}}(\mathbf{b}, \mathbf{b}^{\boldsymbol{\omega}}_{\mathbb{V}^*})] = \sum_{a\in\mathbb{F},\mathbf{s}_a} b_a(\mathbf{s}_a)\log\frac{b_a(\mathbf{s}_a)}{\Psi_a(\mathbf{s}_a)} + \sum_{\substack{i\in\mathbb{V}\setminus\mathbb{V}^*\\ s_i}} b_i(s_i)\log\frac{b_i(s_i)^{1-d_i}}{\Phi_i(s_i)} + \mathbb{E}_{\boldsymbol{\omega}}\left[\sum_{i\in\mathbb{V}^*,s_i} b_i^{\omega_i}(s_i)\log\frac{b_i^{\omega_i}(s_i)^{1-d_i}}{\Phi_i(s_i)}\right],$$

par linéarité de l'espérance et puisque les $\mathbf{b}_a$ et $\mathbf{b}_i, i \in \mathbb{V}\setminus\mathbb{V}^*$ sont indépendants de $\boldsymbol{\omega}$. Au final, on obtient donc

$$\mathbb{E}_{\boldsymbol{\omega}}[F_{\text{Bethe}}(\mathbf{b}, \mathbf{b}^{\boldsymbol{\omega}}_{\mathbb{V}^*})] = \sum_{a\in\mathbb{F},\mathbf{s}_a} b_a(\mathbf{s}_a)\log\frac{b_a(\mathbf{s}_a)}{\Psi_a(\mathbf{s}_a)} + \sum_{\substack{i\in\mathbb{V}\setminus\mathbb{V}^*\\ s_i}} b_i(s_i)\log\frac{b_i(s_i)^{1-d_i}}{\Phi_i(s_i)} + \sum_{\substack{i\in\mathbb{V}^*\\ s_i}}\left((1-\alpha_i)\,b_i(s_i)\log\frac{b_i(s_i)^{1-d_i}}{\Phi_i(s_i)} + \alpha_i\, b_i^*(s_i)\log\frac{b_i^*(s_i)^{1-d_i}}{\Phi_i(s_i)}\right),$$

quelle que soit la loi jointe $\mathbb{P}_{\boldsymbol{\omega}}$ considérée. Les points stationnaires du lagrangien s'expriment alors comme précédemment

$$\begin{cases} b_a(\mathbf{s}_a) = \Psi_a(\mathbf{s}_a)\exp\left(\sum_{j\in a}\lambda_{aj}(s_j) - 1 - \gamma_a\right) \\ b_i(s_i) = \Phi_i(s_i)^{\frac{1}{1-d_i}}\exp\left(\frac{1}{d_i-1}\left(\sum_{c\ni i}\lambda_{ci}(s_i) - \gamma_i\right) - 1\right), \end{cases} \tag{4.7}$$

puisque le terme $(1-\alpha_i)$ apparaît à la fois devant $\mathbf{b}_i$ dans l'énergie libre et dans la contrainte de compatibilité. On utilise à nouveau le changement de variables (2.16) et on obtient les formules de mises à jour en imposant les équations de compatibilités suivantes

$$\sum_{\mathbf{s}_{a\setminus i}} b_a(\mathbf{s}_a) = \mathbb{E}_{\omega_i}\left[b_i^{\omega_i}(s_i)\right]. \tag{4.8}$$

Pour les variables non observées, $i \in \mathbb{V}\setminus\mathbb{V}^*$, on obtient la formule de mise à jour classique de BP (2.4), et pour les variables observées, $i \in \mathbb{V}^*$, (4.8) implique alors

$$\sum_{\mathbf{s}_{a\setminus i}}\Psi_a(\mathbf{s}_a)\prod_{j\in a} n_{j\to a}(s_j) \propto \alpha_i b_i^*(s_i) + (1-\alpha_i)\Phi_i(s_i)\prod_{c\ni i} m_{c\to i}(s_i).$$

Si l'on décide de garder la forme de mise à jour (2.4) pour $\mathbf{m}_{a\to i}, i \in \mathbb{V}^*$:

$$m_{a\to i}(s_i) \propto \sum_{\mathbf{s}_{a\setminus i}}\Psi_a(\mathbf{s}_a)\prod_{j\in a, j\neq i} n_{j\to a}(s_j),$$

on obtient alors

$$\alpha_i b_i^*(s_i) + (1-\alpha_i)\Phi_i(s_i)\prod_{c\ni i} m_{c\to i}(s_i) = n_{i\to a}(s_i)m_{a\to i}(s_i),$$

ce qui conduit aux règles de mises à jour manquantes pour $\mathbf{n}_{i\to a}, i \in \mathbb{V}^*$

$$n_{i\to a}(s_i) \leftarrow \alpha_i \frac{b_i^*(s_i)}{m_{a\to i}(s_i)} + (1-\alpha_i)\Phi_i(s_i)\prod_{b\ni i, b\neq a} m_{b\to i}(s_i). \tag{4.9}$$

La formule de mise à jour (4.9) montre que l'algorithme obtenu est bien celui attendu correspondant à la combinaison convexe des mises à jour,

$$\text{« } (1-\alpha_i)\text{BP} + \alpha_i \text{ mBP »},$$

à laquelle on a donné un sens variationnel.

4.3 Minimisation d'une énergie libre moyenne.

On s'intéresse dans cette partie à une approche permettant de conserver les mises à jour de BP intactes et qui va simplement conduire à modifier localement le modèle sous-jacent. Supposons dans un premier temps qu'une seule variable σ_k soit observée de manière incertaine selon (4.1). L'idée est de construire une nouvelle règle de mise à jour telle que cette variable n'intervienne plus dans les mises à jour, au contraire de ce qui se passe pour mBP.

Commençons par définir les notations utiles dans cette partie. On va considérer le graphe de facteurs $\mathcal{G}$ privé de certains sommets : $\mathcal{G}\setminus\mathbb{V}^*$, pour $\mathbb{V}^* \subset \mathbb{V}$. On notera a' le facteur de $\mathcal{G}\setminus\mathbb{V}^*$ correspondant au facteur a de $\mathcal{G}$; on a alors tout simplement $a' \stackrel{\text{def}}{=} a\setminus\mathbb{V}^*$, c'est à dire que s'il existe des sommets voisins de a dans $\mathbb{V}^*$ alors ces sommets n'appartiennent pas à a'.

4.3.1 Cas d'une unique observation

On construit la fonction à minimiser selon l'idée suivante : sans certitude sur la valeur prise par la variable σ_k, on cherche à obtenir une distribution $\mathbf{b}$ robuste vis à vis de la distribution de la variable σ_k. C'est à dire que l'on va minimiser l'espérance, relative à σ_k, de l'énergie libre de Bethe :

$$\min_{\mathbf{b}\in\mathcal{B}(\mathcal{G}\setminus\{k\})} \mathbb{E}_{\sigma_k}\left[F^{\sigma_k}_{\text{Bethe}}(\mathbf{b})\right], \tag{4.10}$$

avec la fonction à minimiser définie ci-dessous

$$\mathbb{E}_{\sigma_k}\left[F^{\sigma_k}_{\text{Bethe}}(\mathbf{b})\right] \stackrel{\text{def}}{=} \sum_{s_k} b_k^*(s_k) F^{\sigma_k=s_k}_{\text{Bethe}}(\mathbf{b}),$$

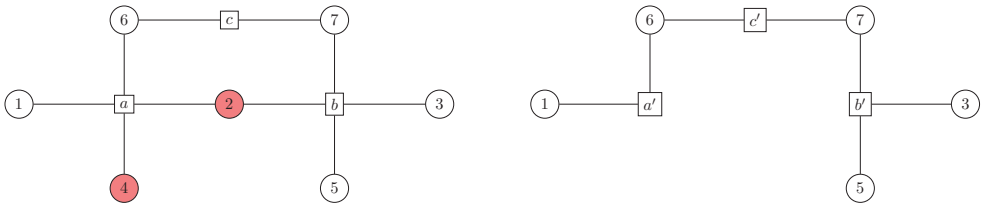


FIGURE 4.7 : Exemple de transformation du graphe de facteurs induites par l'utilisation des formules de mise à jour robustes. Les variables observées correspondent aux sommets 2 et 4 (en rouge). Le graphe $\mathcal{G} \setminus \{2,4\}$ est représenté à droite. Seul le facteur c (et donc sa fonction de compatibilité Ψ_c), n'est pas concerné par la transformation. En particulier, on a par exemple $\mathbf{s}_a = (s_1, s_2, s_3, s_4)$ et $\mathbf{s}_{a'} = (s_1, s_6)$.

où

$$F_{\text{Bethe}}^{\sigma_k = s_k}(\mathbf{b}) = \sum_{\substack{a \in \mathbb{F} \\ \mathbf{s}_{a'}}} b_{a'}(\mathbf{s}_{a'}) \log \frac{b_{a'}(\mathbf{s}_{a'})}{\Psi_a(\mathbf{s}_{a'}, \sigma_k = s_k)} + \sum_{\substack{i \in \mathbb{V} \setminus \{k\} \\ s_i}} b_i(s_i) \log \frac{b_i(s_i)^{1-d_i}}{\Phi_i(s_i)}.$$

La notation $\Psi_a(\mathbf{s}_{a'}, \sigma_k = s_k)$ correspond à la valeur $\Psi_a(\mathbf{s}_{a'})$ dès lors que $k \notin a$; sinon la valeur manquante pour évaluer Ψ_a est tout simplement s_k. On peut alors exprimer la fonction objectif comme

$$\mathbb{E}_{\sigma_k}\left[F_{\text{Bethe}}^{\sigma_k}(\mathbf{b})\right] = \sum_{\substack{a \in \mathbb{F} \\ \mathbf{s}_{a'}}} b_{a'}(\mathbf{s}_{a'}) \log \frac{b_{a'}(\mathbf{s}_{a'})}{\widetilde{\Psi}_{a'}(\mathbf{s}_{a'})} + \sum_{\substack{i \in \mathbb{V} \setminus \{k\} \\ s_i}} b_i(s_i) \log \frac{b_i(s_i)^{1-d_i}}{\Phi_i(s_i)}, \quad (4.11)$$

avec $\log \widetilde{\Psi}_{a'}$ l'espérance, en σ_k, de $\log \Psi_a$:

$$\widetilde{\Psi}_{a'}(\mathbf{s}_{a'}) = \prod_{s_k=0}^{q-1} \Psi_a(\mathbf{s}_a)^{b_k^*(s_k)}. \quad (4.12)$$

On remarque que pour tout facteur a ne contenant pas k, la fonction de compatibilité $\widetilde{\Psi}_{a'}$ n'est rien d'autre que Ψ_a, $\log \Psi_a(\mathbf{s}_a)$ étant alors une fonction constante de s_k. L'information (4.1) sur σ_k est donc entièrement et uniquement utilisée pour construire les nouvelles fonctions $\widetilde{\Psi}_{a'}$. La fonction à minimiser s'exprime donc exactement comme une énergie libre de Bethe sur le graphe $\mathcal{G} \setminus \{4\}$ avec les fonctions de compatibilité (4.12). L'algorithme obtenu est donc l'algorithme BP classique avec modification des fonctions de compatibilité Ψ_a.

4.3.2 Avec plusieurs observations

Supposons maintenant que l'on connaisse les beliefs d'un sous-ensemble $\mathbb{V}^* \subset \mathbb{V}$ de variables. Pour étendre le résultat précédent, la fonction à minimi-

ser serait alors l'espérance, en $\sigma_{\mathbb{V}^*}$, de l'énergie libre de Bethe :

$$\mathbb{E}_{\boldsymbol{\sigma}_{\mathbb{V}^*}}\left[F_{\text{Bethe}}^{\boldsymbol{\sigma}_{\mathbb{V}^*}}(\mathbf{b})\right] \stackrel{\text{def}}{=} \sum_{\mathbf{s}_{\mathbb{V}^*}} b^*_{\mathbb{V}^*}(\mathbf{s}_{\mathbb{V}^*}) F_{\text{Bethe}}^{\boldsymbol{\sigma}_{\mathbb{V}^*}=\mathbf{s}_{\mathbb{V}^*}}(\mathbf{b}). \tag{4.13}$$

De la même manière que précédemment, on dérive alors un algorithme BP classique avec les fonctions de compatibilité $\tilde{\Psi}_{a'}$ qui sont tous simplement telles que

$$\widetilde{\Psi}_{a'}(\mathbf{s}'_a) = \prod_{\mathbf{s}_{\mathbb{V}^*}} \Psi_a(\mathbf{s}_a)^{b^*_{\mathbb{V}^*}(\mathbf{s}_{\mathbb{V}^*})}.$$

Le problème qui se pose est que l'objet naturel qui apparaît ici est la loi jointe des observations $\mathbf{b}^*_{\mathbb{V}^*} = \mathbb{P}_{\boldsymbol{\sigma}_{\mathbb{V}^*}}$. Cette loi jointe n'est pas connue, *a priori*, puisque l'on a supposé uniquement la connaissance des lois marginales $\mathbf{b}^*_k, k \in \mathbb{V}^*$. Sans information supplémentaire, on est contraint de faire l'approximation suivante :

$$b^*_{\mathbb{V}^*}(\mathbf{s}_{\mathbb{V}^*}) = \prod_{k\in\mathbb{V}^*} b^*_k(s_k),$$

ce qui revient à supposer l'indépendance des variables σ_k et ne paraît, pour le moins, pas très pertinent.

L'avantage évident de mBP par rapport à ce qui est proposé ici est que la structure de dépendance entre les variables observées est déterminée comme compatible avec la loi $\mathbb{P}_{\boldsymbol{\sigma}}$ (2.2) par l'algorithme alors qu'elle doit être présupposée ici. On verra dans la section suivante que ce défaut peut être très pénalisant.

4.3.3 Comparaison avec mBP

Dans cette section, on cherche à comparer l'approche *minimisation d'une énergie libre de Bethe moyenne* et la minimisation (4.2). Pour cela, on considère deux cas simples : le cas d'un facteur a contenant trois variables binaires, avec une ou deux de ces trois variables appartenant à $\mathbb{V}^*$. Dans ces cas, mBP nous fournit la solution exacte de (4.2) (propositions 4.1 et 4.3). On réalise une expérience similaire à celle décrite dans la section 4.1.3 pour ce qui est de la génération de la fonction Ψ_a et des beliefs $\mathbf{b}^*_i$ qui sont fixés. Pour chacune des ces expériences, on récupère alors les solutions de (4.2) et (4.10). On répète ces expériences 10 000 fois et les résultats sont présentés à la figure 4.8 page suivante.

Remarquons tout d'abord que la qualité de l'approximation (4.10) se dégrade lorsque l'information mutuelle $\mathrm{I}(\Psi_a)$ augmente, comme l'indiquent les droites de régression clairement croissantes (partie droite). En effet les coefficients de corrélation de Pearson et Spearman $(\rho; r_s)$ valent respectivement (0,40 ; 0,40) pour la partie haute (une variable fixée) et (0,32 ; 0,31) pour la partie basse (deux variables fixées).

Dans le cas où une seule variable est fixée (partie haute), l'approximation est globalement assez bonne : l'erreur L^1 moyenne est de l'ordre de 0,03 et 95%

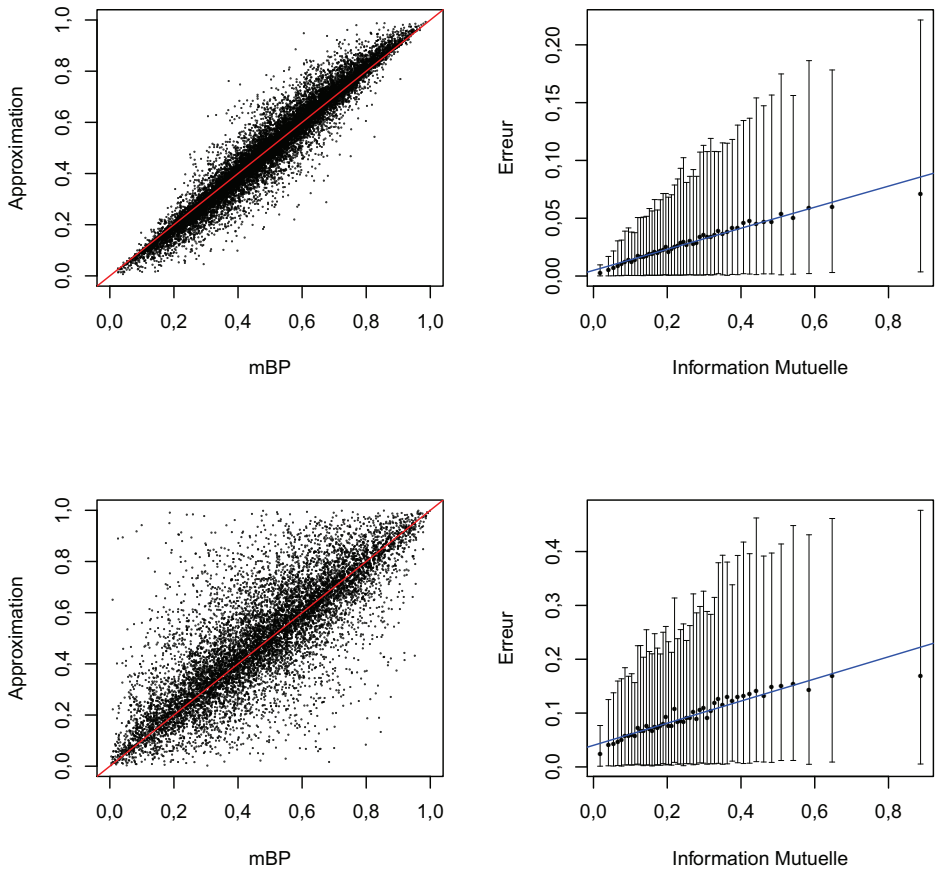


FIGURE 4.8 : Comparaison des solutions de (4.2) et de (4.10) pour un facteur contenant trois variables binaires lorsqu'une (partie haute) ou deux (partie basse) variables sont observées selon (4.1). Les figures de gauche correspondent aux nuages de points avec la solution de (4.2) en abscisse ; la droite rouge est la première bissectrice, où tous les points seraient concentrés si les solutions des deux problèmes variationnels étaient les mêmes. À droite, on a subdivisé les points en 50 classes de même poids selon l'information mutuelle et on représente la distance L^1 moyenne entre les solutions de (4.10) et celles de (4.2) pour chaque classe, ainsi que ses quantiles d'ordre 0,05 et 0,95. La droite bleue est la régression linéaire du nuage de points.

des erreurs sont inférieures à 0,1. On remarque cependant que les solutions de (4.10) sont légèrement plus polarisées vers les états $0-1$ que celles de (4.2).

Dans le cas où 2 variables sont fixées (partie basse) et comme on s'y attendait, la qualité de l'approximation se dégrade fortement : l'erreur L_1 moyenne est de l'ordre de 0,09 et surtout 10% de ces erreurs sont supérieures à 0,2. En effet, l'approche (4.10) suppose l'indépendance des variables observées ce qui conduit à des résultats qui peuvent être totalement aberrants, notamment lorsque l'information mutuelle est relativement élevée. Malgré ses mérites, en particulier en termes de convergence, l'approche basée sur la minimisation d'une énergie libre moyenne ne semble pas être utilisable sauf à supposer une connaissance *a priori* sur les dépendances entre variables observées.

Chapitre 5

Inférence sur des variables réelles par un modèle d'Ising

Dans les chapitres 2 et 4, nous avons décrit et développé des outils qui vont maintenant nous être utiles. On aborde ici une manière de traiter le problème décrit au chapitre 1. Considérons un champ markovien aléatoire $\mathbf{X}$ à valeurs réelles. On suppose que les données disponibles sont de la forme suivante : pour un ensemble de paires de variables $(i,j) \in \mathbb{E}$, on a N_{ij} réalisations indépendantes du couple $(X_i, X_j) = \mathbf{X}_{\{i,j\}}$ contenues dans les vecteurs $\mathbf{x}^k_{\{i,j\}}$. C'est-à-dire que les observations sont du type

$$\mathbf{X}_{\{i,j\}} = \mathbf{x}^k_{i,j} \text{ pour } k \in \{1, \dots, N_{ij}\}.$$

On notera $\mathbf{x}$ le vecteur contenant tous les $\mathbf{x}^k_{i,j}$. On considère donc dans ce chapitre que le graphe de facteurs $\mathcal{G} = (\mathbb{V} \cup \mathbb{F}, \mathbb{E})$ est connu.

Il arrive que les variables réelles $\mathbf{X}$ soient dotées d'une description sous-jacente de type binaire, fluide/congestionné dans le cas de trafic routier par exemple. Il est alors tentant de vouloir abstraire cette perception simplifiée des variables dans des variables latentes. On construit ici une modélisation utilisant cette possibilité, en encodant les dépendances entre variables réelles au niveau des variables latentes. Ceci peut être vu comme une alternative à l'algorithme « Expectation Propagation » [73] et, comparé aux méthodes traditionnelles telle que les filtres à particules [27], on s'attend à une complexité plus favorable.

On formalise le modèle de la manière suivante : l'état du système est représenté par un vecteur $\mathbf{X} = (X_i)_{i \in \mathbb{V}}$ de N variables aléatoires réelles, qui sont à valeurs dans les ensembles $\mathcal{X}_i \subset \mathbb{R}$. À chaque X_i, on associe une variable latente binaire σ_i, qui reste à définir. On insiste sur ce point précis : on suppose uniquement l'existence de variables latentes sous-jacentes, leur identité est inconnue. Par exemple, dans le cas de données de trafic routier, la définition de l'état fluide/congestionné à partir d'un temps de trajet est non triviale. Il sera donc ici nécessaire de *définir* les variables latentes et plusieurs choix seront

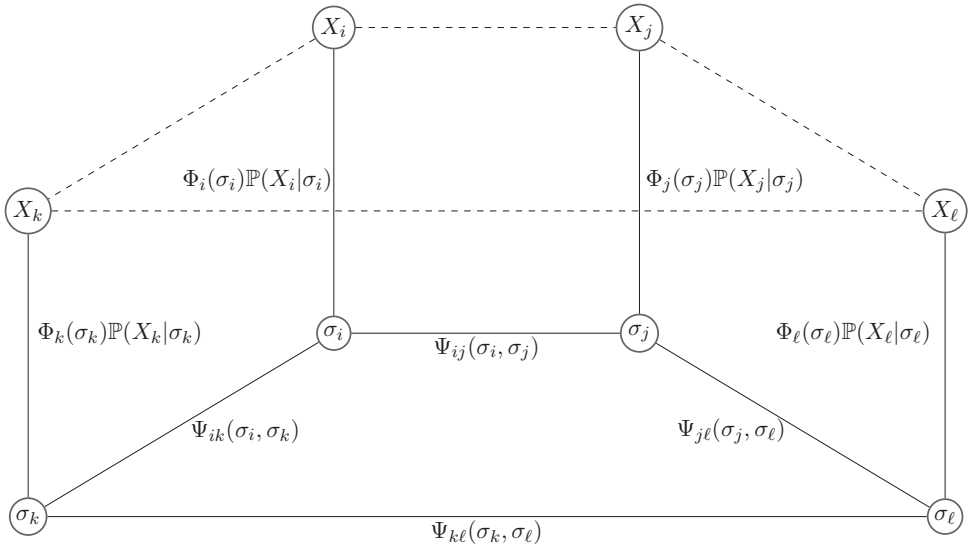


FIGURE 5.1 : Champ markovien aléatoire correspondant au vecteur $(\mathbf{X}, \boldsymbol{\sigma})$ pour $\mathbb{V} = \{i, j, k, \ell\}$. Le « vrai » modèle sur le vecteur $\mathbf{X}$ (lignes pointillées) est approché à travers les variables latentes binaires $\boldsymbol{\sigma}$ (lignes continues).

admissibles. On ne se situe donc pas dans le cadre des modèles markoviens à états cachés où, à partir d'observations indirectes, on cherche à estimer les variables latentes. Les variables latentes n'ont ici que peu d'intérêt en elles-mêmes. Elles sont seulement des auxiliaires utilisés pour prédire les variables réelles X_i.

On fait l'hypothèse que les variables X_i sont indépendantes, conditionnellement à l'état du vecteur des variables latentes $\boldsymbol{\sigma}$. Dans le but de pouvoir inférer le comportement des variables X_i à partir d'observations partielles du système, on modélise la loi de $\boldsymbol{\sigma}$ par un champ markovien aléatoire avec interactions de paires, *i.e.* un modèle d'Ising. La mesure jointe des variables $\mathbf{X}$ et $\boldsymbol{\sigma}$ se factorise comme (Figure 5.1) :

$$\mathbb{P}(\mathbf{X} \leq \mathbf{x}, \boldsymbol{\sigma} = \mathbf{s}) = \mathbb{P}(\boldsymbol{\sigma} = \mathbf{s}) \prod_{i \in \mathbb{V}} \mathbb{P}(X_i \leq x_i | \sigma_i = s_i),$$

avec

$$\mathbb{P}(\boldsymbol{\sigma} = \mathbf{s}) = \frac{1}{Z} \prod_{(i,j) \in \mathbb{E}} \Psi_{ij}(s_i, s_j) \prod_{i \in \mathbb{V}} \Phi_i(s_i),$$

et Z une constante telle que $\sum_{\mathbf{s}} \mathbb{P}(\boldsymbol{\sigma} = \mathbf{s}) = 1$. Sous ces hypothèses, on tente de répondre aux trois questions suivantes :

(i) Comment associer une observation de la variable réelle X_i à l'état de la variable latente σ_i ?

(ii) Comment construire la structure de dépendance entre variables latentes σ_i de manière efficace pour l'inférence sur les X_i ?

(iii) Quelle méthode d'inférence employer pour prédire les variables non observées ?

Ces trois questions sont évidemment très interdépendantes. Notre méthode approchée s'appuie sur l'algorithme BP, et on utilisera la variante mBP décrite au chapitre 4 en tant qu'outil d'inférence répondant à la question (iii). Ceci influencera évidemment la structure du modèle d'Ising. Notons qu'il existe d'autres méthodes basées sur BP permettant de traiter ce problème directement dans l'espace du vecteur aléatoire réel $\mathbf{X}$. Citons notamment l'algorithme Belief Propagation non paramétrique de Sudderth *et al.* [88] et l'algorithme « Non-Paranormal Belief Propagation » d'Elidan et Cario [29]. Pour ces deux algorithmes la complexité de l'étape d'inférence est cependant supérieure à celle de l'algorithme BP classique. On propose ici une nouvelle méthode basée sur BP pour laquelle l'étape d'inférence est peu coûteuse. En particulier, on souhaite que celle-ci soit utilisable en temps réel sur des graphes de grande taille comme le suggère l'application au trafic routier décrite au chapitre 1.

Le chapitre est organisé de la manière suivante : la section 5.1 tente de répondre à la question (i), en exhibant des fonctions qui associent le paramètre d'une variable de Bernoulli σ à une observation de X. Dans la section 5.2, on s'intéresse à la question (ii) qui concerne l'encodage optimal des dépendances du vecteur réel $\mathbf{X}$ dans l'espace latent. Enfin la section 5.3 présente quelques résultats numériques.

5.1 Définition des variables latentes

Soit X une variable aléatoire réelle dont la fonction de répartition est notée $F(x) \stackrel{\text{def}}{=} \mathbb{P}(X \leq x)$. On s'intéresse ici à la manière d'associer une variable latente binaire σ à une observation $X = x$. Par la suite, on appellera « paramètre de σ » la valeur de $\mathbb{P}(\sigma = 1)$.

5.1.1 Un seuil de décision aléatoire

Un moyen simple d'associer une observation $X = x$ à la variable latente σ est de considérer une application Λ telle que $\Lambda(x)$ soit le paramètre de la variable aléatoire binaire σ. L'application Λ sera appelée « fonction d'encodage » et peut dépendre de F. σ étant une variable latente, elle ne pourra être observée directement, mais, conditionnellement à une observation $X = x$, sa distribution est :

$$\mathbb{P}(\sigma = 1 | X = x) = \Lambda(x). \tag{5.1}$$

Pour simplifier le propos, on considère des fonctions Λ *càdlàg*[1] et croissantes. Considérer $1 - \Lambda$ revient simplement à permuter les états 0 et 1 de σ et la condition de croissance est en fait équivalente au choix d'une fonction Λ

1. *continue à droite, limitée à gauche.*

monotone. On souhaite de plus que Λ croisse de 0 à 1, mais pas nécessairement que $\Lambda(\mathcal{X}) = [0, 1]$. On résume cette contrainte par l'équation suivante :

$$\int_{\mathcal{X}} d\Lambda(X) = 1. \tag{5.2}$$

On insiste à nouveau sur le fait que σ n'est pas seulement une variable aléatoire latente non-observable dont l'estimation est imposée. Il s'agit d'un objet intermédiaire, que nous définissons nous-même, dans le but de résoudre le problème d'inférence sur $\mathbf{X}$. Cet encodage fait partie du schéma suivant

$$\begin{array}{ccc} X_i = x_i \in \mathcal{X}_i & \xrightarrow{\Lambda_i} & \mathbb{P}(\sigma_i = 1 | X_i = x_i) \in \Lambda_i(\mathcal{X}_i) \\ & & \big\downarrow \text{mBP} \\ X_j = x_j \in \mathcal{X}_j & \xleftarrow{\Gamma_j} & b(\sigma_j = 1) \in [0, 1] \end{array} \tag{5.3}$$

où les observations des variables X_i sont encodées par l'état d'une variable aléatoire latente binaire. On réalise alors l'inférence sur ces variables latentes par l'une des techniques précédemment décrites au chapitre 4 et on peut finalement estimer le comportement des autres variables réelles X_j qui n'ont pas été observées selon le schéma suivant

Ce schéma nécessite de définir une fonction « réciproque » de Λ que l'on note $\Gamma : [0, 1] \mapsto \mathcal{X}$. Λ pouvant être non inversible, la fonction de décodage Γ n'est pas nécessairement la fonction inverse Λ^{-1}. On reviendra sur le choix de la fonction Γ à la section 5.1.3.

Pour bien comprendre l'interaction entre σ et X, définissons les fonctions de répartition conditionnelles suivantes :

$$F^0(x) \overset{\text{def}}{=} \mathbb{P}(X \leq x | \sigma = 0),$$
$$F^1(x) \overset{\text{def}}{=} \mathbb{P}(X \leq x | \sigma = 1).$$

Le théorème de Bayes nous permet d'écrire que

$$\mathbb{P}(\sigma = 1 | X = x) = \mathbb{P}(\sigma = 1) \frac{dF^1}{dF}(x),$$

et donc

$$dF^1(x) = \frac{\Lambda(x)}{\mathbb{P}(\sigma = 1)} dF(x). \tag{5.4}$$

En déconditionnant par rapport à σ, on obtient

$$F(x) = \mathbb{P}(\sigma = 1) F^1(x) + \mathbb{P}(\sigma = 0) F^0(x), \tag{5.5}$$

ce qui permet d'obtenir l'autre distribution conditionnelle

$$dF^0(x) = \frac{1 - \Lambda(x)}{\mathbb{P}(\sigma = 0)} dF(x). \tag{5.6}$$

Proposition 5.1. *Le choix d'une fonction d'encodage Λ croissante induit une décomposition de la variable aléatoire X comme mixture des deux variables aléatoires stochastiquement ordonnées $X^1 \sim dF^1$ et $X^0 \sim dF^0$. En effet, on a alors*

$$X \sim \mathbb{1}_{\{\sigma=1\}} X^1 + \mathbb{1}_{\{\sigma=0\}} X^0,$$

où le symbole $\sim$ indique l'égalité en loi. En terme d'ordre stochastique on a les inégalités suivantes

$$X^0 \preceq X \preceq X^1.$$

Démonstration. Pour prouver le résultat il faut, (et il suffit de) montrer que

$$\forall x \in \mathcal{X}, \quad F^1(x) \leq F(x) \leq F^0(x).$$

Prouvons tout d'abord la première inégalité ($F^1 \leq F$) ; soit $x \in \mathcal{X}$ tel que $\Lambda(x) \leq \mathbb{P}(\sigma = 1)$, on a alors :

$$\begin{aligned} F^1(x) &= \int_{-\infty}^{x} dF^1(y) = \int_{-\infty}^{x} \frac{\Lambda(y)}{\mathbb{P}(\sigma=1)} dF(y), \\ &\leq \int_{-\infty}^{x} dF(y) = F(x), \end{aligned}$$

car, Λ étant croissante, on a $\Lambda(y) \leq \mathbb{P}(\sigma = 1)$ pour tout $y \in]-\infty, x]$. Considérons maintenant une valeur x telle que $\Lambda(x) \geq \mathbb{P}(\sigma = 1)$; on a de même

$$\begin{aligned} F^1(x) &= 1 - \int_{x}^{+\infty} dF^1(y) = 1 - \int_{x}^{+\infty} \frac{\Lambda(y)}{\mathbb{P}(\sigma=1)} dF(x), \\ &\leq 1 - \int_{x}^{+\infty} dF(y) = F(x), \end{aligned}$$

toujours en utilisant la croissance de Λ. L'autre inégalité ($F \leq F^0$) est tout simplement obtenue en utilisant (5.5). □

La fonction d'encodage Λ étant croissante et vérifiant (5.2), elle peut être considérée comme la fonction de répartition d'une variable aléatoire Y,

$$\mathbb{P}(Y \leq x) \stackrel{\text{def}}{=} \Lambda(x).$$

Cela permet d'interpréter les quantités précédentes en terme de cette variable aléatoire Y :

$$\begin{aligned} \mathbb{P}(\sigma = 1) &= \int_{\mathcal{X}} \mathbb{P}(\sigma = 1 | X = x) dF(x) \\ &= \int_{\mathcal{X}} \Lambda(x) dF(x) = \int_{\mathcal{X}} \mathbb{P}(Y \leq x) dF(x) \\ &= \mathbb{P}(Y \leq X), \end{aligned}$$

sous l'hypothèse d'indépendance entre Y et X. On peut donc définir la variable aléatoire σ de la manière suivante

$$\sigma \overset{\text{def}}{=} \mathbb{1}_{\{Y \leq X\}},$$

qui indique que la variable Y agit comme un seuil aléatoire permettant de décider si l'observation de X correspond à l'état $\sigma = 0$ ou $\sigma = 1$. L'ordre stochastique entre $(X|\sigma = 0)$ et $(X|\sigma = 1)$ apparaît alors naturellement.

Remarque 5.1. *Lorsque $\Lambda = F$, on obtient les distributions conditionnelles de X suivantes :*

$$(X|\sigma = 1) \sim \max(X_1, X_2),$$
$$(X|\sigma = 0) \sim \min(X_1, X_2),$$

avec X_1 et X_2 deux variables aléatoires indépendantes de même loi que X.

5.1.2 Choix de la fonction d'encodage

Maintenant que l'on a décrit la structure de relation entre les variables X et σ, il reste à choisir la fonction d'encodage Λ optimale en un certain sens. Il s'avère qu'il est difficile de proposer un unique critère pour cette tâche. Dans cette section, on proposera différentes approches basées sur des critères différents. Il n'est pas possible d'effectuer un choix *a priori* parmi ces différentes fonctions d'encodages. On verra dans la section 5.3 que chacune a ses mérites suivant les données considérées.

Information mutuelle. L'idée est de choisir Λ ou, de manière équivalente, σ (comme on l'a vu dans la section 5.1.1), de sorte que l'information mutuelle $I(X, \sigma)$ entre les variables X et σ soit maximale. En d'autres termes, une quantité d'information sur une variable doit en fournir autant que possible sur l'autre variable.

Proposition 5.2. *Soit $q_X^{0,5}$ la médiane de X. La fonction d'encodage Λ_{MI} maximisant l'information mutuelle $I(X, \sigma)$ entre les variables X et σ est la fonction de seuil*

$$\Lambda_{\text{MI}}(x) \overset{def}{=} \mathbb{1}_{\{x \geq q_X^{0,5}\}}.$$

Démonstration. Explicitons la fonction à maximiser

$$\begin{aligned} I(X, \sigma) &= \sum_s \int_{\mathcal{X}} \mathbb{P}(\sigma = s) \log \left(\frac{dF^s(x)}{dF(x)} \right) dF^s(x) \\ &= H(\mathbb{P}(\sigma = 1)) - \int_{\mathcal{X}} H(\Lambda(x)) dF(x), \end{aligned}$$

avec $H(p) \overset{\text{def}}{=} -p \log p - (1-p) \log(1-p)$, la fonction d'entropie d'une variable binaire. Parmi toutes les variables aléatoires σ de même entropie $H(\mathbb{P}(\sigma = 1))$,

celles qui maximisent $I(X, \sigma)$ sont évidemment les fonctions déterministes de X, ou de manière équivalente les fonctions indicatrices. Puisque l'on s'est limité aux fonctions *càdlàg* croissantes, on obtient alors $\Lambda = \mathbb{1}_{[a,+\infty[}$ pour $a \in \mathcal{X}$. Il ne reste alors plus qu'à maximiser l'entropie de la variable σ, ce qui conduit à $P(\sigma = 1) = \frac{1}{2}$ et $a = q_X^{0,5}$. □

Principe du maximum d'entropie. Un autre choix naturel, pour rendre maximale l'information contenue dans la variable latente σ, consiste à maximiser l'entropie de la variable $U = \Lambda(X)$. Cette variable U correspond en effet aux données que l'on utilisera pour construire le modèle d'Ising des variables latentes (voir section 5.2). On suppose ici que la variable X admet une densité notée f. On ajoute aux contraintes explicitées dans la section précédente que Λ est une bijection entre $\mathcal{X}$ et $[0, 1]$. Puisque l'on considère des variables continues, l'entropie n'a de sens que relativement à une mesure [52, pp 374-375]. En suivant l'argumentation de Jaynes [51] – U étant le paramètre d'une variable de Bernoulli dont les deux issues sont de probabilité non nulles et en supposant une ignorance complète – on obtient la mesure uniforme comme référence.

Proposition 5.3. *Soit X une variable aléatoire à densité. Alors, la fonction (croissante) inversible qui maximise l'entropie relative à la mesure uniforme de $U = \Lambda(X)$ est F, la fonction de répartition de X.*

Démonstration. La variable U d'entropie maximale est connue et correspond à la densité uniforme $h_\Lambda(u) = \mathbb{1}_{[0,1]}(u)$, où h_Λ est la densité de U. La fonction Λ telle que $\Lambda(X)$ soit une variable uniforme sur $[0, 1]$ est la fonction de répartition de X, ce qui conclut la preuve. □

Erreur moyenne de décodage. Un autre critère possible pour le choix de la fonction d'encodage Λ consiste à minimiser l'erreur moyenne de prédiction sur X causée par une erreur sur l'estimation du paramètre b de la variable σ, c'est à dire l'espérance $\mathbb{E}\left[\frac{dx}{d\Lambda(X)}\right]$ selon la distribution de X. En un certain sens, cela fournit la fonction d'encodage la plus robuste aux erreurs d'inférence sur les variables binaires. La fonction d'encodage obtenue est cependant un objet moins naturel que celles obtenues précédemment.

Proposition 5.4. *Supposons que X admette une densité f, et que $f \in \mathcal{L}^{1/2}$. La fonction Λ_{DE} qui minimise l'erreur moyenne de décodage $\mathbb{E}\left[\frac{1}{\Lambda'(X)}\right]$ est*

$$\Lambda_{\mathrm{DE}}(x) \stackrel{def}{=} \kappa \int_{-\infty}^{x} \sqrt{f(x)}dx,$$

avec $\kappa = \int_{\mathcal{X}} \sqrt{f(x)}dx.$

Démonstration. L'erreur moyenne de décodage sur les variables réelles s'exprime comme :

$$\mathbb{E}\left[\frac{1}{\Lambda'(X)}\right] = \int_{\mathcal{X}} \frac{f(x)}{\Lambda'(x)} dx,$$

avec la contrainte $\int_{\mathcal{X}} \Lambda'(x)dx = 1$. Le lagrangien correspondant est donc

$$\mathcal{L}(\Lambda', \gamma) = \int_{\mathcal{X}} \frac{f(x)}{\Lambda'(x)} dx + \gamma \left(\int_{\mathcal{X}} \Lambda'(x)dx - 1 \right),$$

dont les conditions de stationnarité ($\frac{\delta \mathcal{L}}{\delta \Lambda'(x)} = 0$) sont

$$-\frac{f(x)}{\Lambda'(x)^2} + \gamma = 0.$$

Ceci conduit au résultat annoncé, sous réserve de la condition $f \in \mathcal{L}^{1/2}$. □

Résumons rapidement les différents choix disponibles pour la fonction d'encodage Λ. Tout d'abord on a proposé deux méthodes intuitives :

- Λ_{MI} qui correspond à un encodage déterministe, c'est-à-dire que σ indique la position de X vis-à-vis de sa médiane ;
- F la fonction de répartition de X, qui permet de faire le choix le moins discriminant sur la distribution des données encodées.

On verra à la section 5.1.3 que ces deux encodages ont des propriétés très différentes : Λ_{MI} est un choix beaucoup plus conservatif que F. Le choix Λ_{MI} est par contre plus apte à modéliser les distributions jointes (section 5.2.1). De plus, ce choix permet d'utiliser l'algorithme BP à la place de mBP. Les observations fournissent tout simplement la valeur de σ. On a vu au chapitre 4 que la convergence est alors plus rapide en comparaison au cas de mBP.

On a enfin proposer un troisième choix Λ_{DE} basé sur un critère moins intuitif. On verra que si le critère menant à ce choix est adapté au problème étudié la difficulté à estimer ses paramètres le rende plus difficilement utilisable.

5.1.3 Choix de la fonction de décodage

Avant de se tourner vers la définition de la fonction de décodage Γ, considérons tout d'abord la question suivante :

> *Quel est le meilleur prédicteur d'une variable aléatoire réelle X, connaissant sa distribution ?*

La réponse dépend bien évidemment du critère considéré et cela influencera les choix de la fonction de décodage Γ dont le rôle est justement de prédire la variable X. Supposons que le critère soit la minimisation de la norme L^r, le meilleur prédicteur $\hat{\theta}_r(X)$ est alors défini comme

$$\hat{\theta}_r(X) \stackrel{\text{def}}{=} \operatorname*{argmin}_{c \in \mathbb{R}} \mathbb{E}[|X - c|^r].$$

Dans le cas $r = 1$, ce prédicteur optimal $\hat{\theta}_1(X)$ est tout simplement la médiane de X ; $r = 2$ correspond à $\hat{\theta}_2(X) = \mathbb{E}[X]$, la moyenne de X. Même si ces deux cas sont parmi les plus intéressants, notons que tout autre choix de statistique en tant que meilleur prédicteur est admissible. Dans la suite on appellera *prédiction de base* cette prédiction de X sans autre information disponible que la loi de X.

Revenons maintenant à la détermination de la fonction de décodage Γ, pour laquelle deux définitions naturelles se présentent. Lorsque Λ est une application bijective l'idée la plus simple pour construire un prédicteur de X connaissant la distribution $b = \mathbb{P}(\sigma = 1)$ est $\Lambda^{-1}(b)$. D'après la définition (5.1) de Λ il s'agit de l'unique valeur x telle que $\mathbb{P}(\sigma = 1|X = x) = b$. On notera

$$\Gamma^{\mathcal{D}} \stackrel{\text{def}}{=} \Lambda^{-1},$$

ce premier choix pour la fonction de décodage. $\Gamma^{\mathcal{D}}$ correspond, en un certain sens, à un prédicteur basé sur le maximum de vraisemblance. En effet, supposons que l'on remplace la connaissance $b = \mathbb{P}(\sigma = 1)$ par un échantillon de M réalisations indépendantes s^k d'une variable de loi $\mathbb{P}(\sigma|X = x)$. L'estimateur du maximum de vraisemblance pour x est alors $\Lambda^{-1}(\sum_k s^k/M)$. Le choix de Λ^{-1} correspond donc au maximum de vraisemblance d'un échantillon dont la probabilité empirique est b.

Dans le cas plus général d'une fonction d'encodage Λ croissante et *càd-làg*, la connaissance de la loi de σ permet de mettre à jour la loi de X en tenant compte de cette information. En appliquant la règle de mise à jour de Jeffrey [15] on obtient la fonction de répartition $F^{\mathcal{P}}$ mise à jour suivante

$$F^{\mathcal{P}}(x) = bF^1(x) + (1 - b)F^0(x). \tag{5.7}$$

On notera $X^{\mathcal{P}}$ la variable aléatoire de loi $F^{\mathcal{P}}$. On peut alors utiliser le prédicteur $\hat{\theta}(X^{\mathcal{P}})$ défini précédemment comme prédicteur de X. Cette méthode peut bien entendu être utilisée que Λ soit bijective ou non. La statistique voulue de $X^{\mathcal{P}}$ ne s'exprime pas toujours comme une fonction explicite de b, et peut nécessiter une estimation numérique. Certains choix de fonctions d'encodage Λ conduisent cependant à des formules explicites comme nous allons le voir. Pour faire référence à ce second choix, on utilisera la notation

$$\Gamma^{\mathcal{P}} \stackrel{\text{def}}{=} \hat{\theta}(X^{\mathcal{P}}).$$

Information mutuelle. On considère ici le cas de la fonction seuil Λ_{MI} comme fonction d'encodage. Cette fonction d'encodage n'est pas inversible et la seule possibilité de décodage est celle correspondant au décodage probabiliste $\Gamma^{\mathcal{P}}$. D'après (5.4)–(5.7), la distribution de $X^{\mathcal{P}}$ s'exprime comme

$$F^{\mathcal{P}}(x) = \begin{cases} 2(1 - b)F(x), & \text{si } x \leq q_X^{0,5}, \\ F^{\mathcal{P}}(q_X^{0,5}) + 2b\left(F(x) - F(q_X^{0,5})\right), & \text{si } x > q_X^{0,5}. \end{cases} \tag{5.8}$$

Pour obtenir $\hat{\theta}_1(X^{\mathcal{P}})$, il suffit de résoudre l'équation $F^{\mathcal{P}}(x) = 1/2$, qui conduit à la fonction de décodage

$$\Gamma^{\mathcal{P}}(b) = \begin{cases} F^{-1}\left(\frac{1}{4(1-b)}\right), & \text{si } b \leq \frac{1}{2}, \\ F^{-1}\left(\frac{4b-1}{4b}\right), & \text{si } b > \frac{1}{2}. \end{cases}$$

Lorsque F n'est pas inversible, la médiane peut ne pas être unique et l'application F^{-1} doit être interprétée comme le pseudo-inverse de F, généralement utilisé pour définir les quantiles d'une loi, $F^{-1}(b) \stackrel{\text{def}}{=} \inf_x\{x|F(x) \geq b\}$.

Enfin, si l'on choisit le prédicteur $\hat{\theta}_2$ basé sur la norme L^2, on obtient par linéarité de l'espérance :

$$\Gamma^{\mathcal{P}}(b) = \mathbb{E}\left[X^{\mathcal{P}}\right] = b\,\mathbb{E}\left[X|\sigma = 1\right] + (1-b)\mathbb{E}\left[X|\sigma = 0\right].$$

Principe du maximum d'entropie. Si l'on choisit d'utiliser $\Gamma^{\mathcal{D}} = F^{-1}$ comme fonction de décodage, la prédiction de base, *i.e.* en l'absence d'observations, est tout simplement $F^{-1}(\mathbb{P}(\sigma = 1))$. Or l'on a $\mathbb{P}(\sigma = 1) = \mathbb{E}[F(X)] = 1/2$, la prédiction de base est donc la médiane de X. Le choix de F comme fonction d'encodage et de F^{-1} comme fonction de décodage est donc optimal vis-à-vis d'une mesure L^1 pour l'erreur de prédiction.

L'autre possibilité consiste à utiliser la fonction de décodage probabiliste $\Gamma^{\mathcal{P}}$ en calculant, par exemple, le prédicteur $\hat{\theta}_1(X^{\mathcal{P}})$ qu'est la médiane. En utilisant (5.4)–(5.7), on obtient la fonction de répartition de $X^{\mathcal{P}}$

$$F^{\mathcal{P}}(x) = \big((2b-1)F(x) - 2(b-1)\big)F(x),$$

et la fonction recherchée est donc solution de l'équation quadratique suivante

$$\left((2b-1)F(x) - 2(b-1)\right)F(x) = \frac{1}{2},$$

dont une seule racine est atteignable. On obtient ainsi la fonction de décodage probabiliste

$$\Gamma^{\mathcal{P}}(b) = F^{-1}\left(\frac{2(b-1) + \sqrt{(2b-1)^2+1}}{4b-2}\right). \tag{5.9}$$

À ce stade il est possible d'observer la différence de nature entre les deux types de décodages. Les fonctions de décodage probabiliste $\Gamma^{\mathcal{P}}$ sont beaucoup plus conservatrices que $\Gamma^{\mathcal{D}}$. En effet, l'utilisation de F^{-1} nous permet de faire des prédictions qui couvrent tout l'ensemble de valeurs de X alors que le décodage probabiliste ne le permet pas. La figure 5.2 illustre cela. Supposons que deux variables aléatoires X_i et X_j soient p.s. égales. Même en construisant un modèle latent tel que $\mathbb{P}(\sigma_i = \sigma_j) = 1$, le choix de $\Gamma^{\mathcal{P}}$ comme fonction décodage ne nous permettra jamais de faire des prédictions telles que $X_j = X_i$. Ce choix s'appuie en fait sur la distribution approchée du couple (X_i, X_j) qui

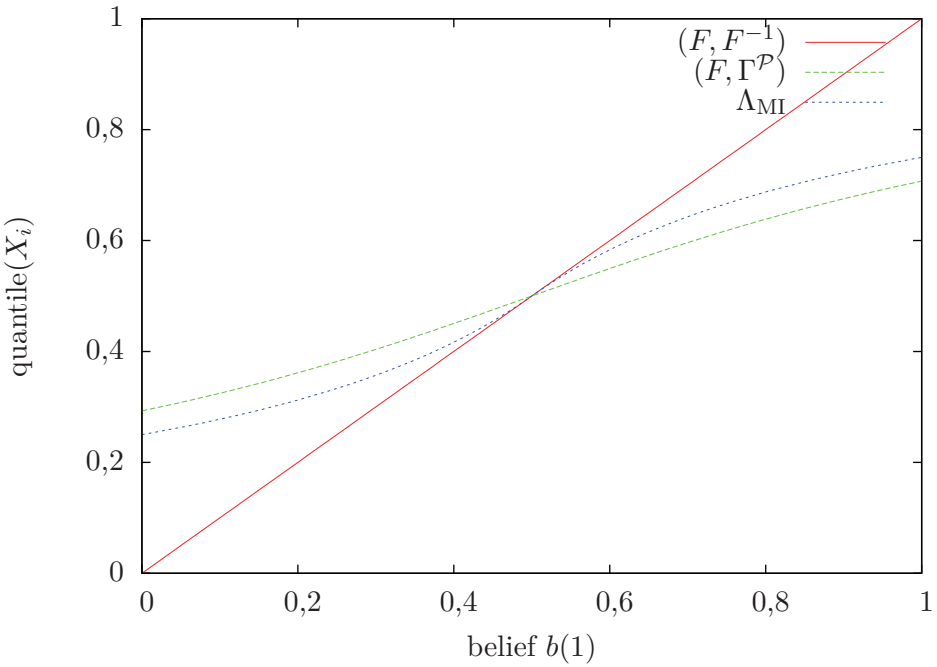


FIGURE 5.2 : Prédiction, exprimée en quantiles, de la variable X_i correspondant à un belief $b_i(1)$. Les valeurs limites pour Λ_{MI} sont $1/4$ et $3/4$. Pour F et $\Gamma^{\mathcal{P}}$ il s'agit alors de $1 - \sqrt{2}/2$ et $\sqrt{2}/2$.

ne peut qu'être sous-optimale pour des corrélations fortes (proposition 5.5). On constatera expérimentalement dans la section 5.3 que le choix $\Gamma^{\mathcal{P}}$ est en effet sous-optimal lorsque les variables sont fortement corrélées. Au contraire le choix (F, F^{-1}) correspond à effectuer une régression entre les quantiles des variables X_i.

Erreur moyenne de décodage. Le critère conduisant à la fonction d'encodage Λ_{DE} n'a de sens que si l'on utilise le décodage par l'application inverse $\Gamma^{\mathcal{P}}$. En effet le critère correspond à minimiser l'erreur sur $\Lambda^{-1}(b)$ pour une erreur sur l'estimation de b. Cependant, la prédiction de base correspondante, c'est-à-dire $\Lambda_{\mathrm{DE}}^{-1}(\mathbb{E}[\Lambda_{\mathrm{DE}}(X)])$, n'a aucune propriété notable. En particulier, elle n'est pas optimale vis-à-vis des mesures L^1 ou L^2 de l'erreur.

Dans le cas général où l'on choisit le décodage au sens du maximum de vraisemblance $\Gamma^{\mathcal{P}} = \Lambda^{-1}$ il est naturel de vouloir généraliser les critères de choix de la fonction d'encodage Λ pour obtenir une prédiction de base optimale vis-à-vis du critère choisi. Le cas de la maximisation de l'entropie mène à une fonction facilement calculable. Le critère de minimisation de l'erreur de décodage pose lui plus de difficultés. Ces deux cas sont détaillés dans les

annexes 5.A et 5.B.

Dans la suite on considérera une mesure des erreurs basée sur la norme L^1. Par rapport à une erreur L^2 celle-ci permet de donner moins d'importance aux valeurs extrêmes.

5.2 Distribution jointe des variables latentes

On vient d'étudier dans la section 5.1 la manière dont la variable réelle X_i et sa variable latente σ_i sont reliées, par l'intermédiaire de la fonction d'encodage Λ_i. On s'intéresse maintenant à la construction des dépendances entre variables latentes et, plus précisément, à l'estimation des paramètres du modèle d'Ising sur $\boldsymbol{\sigma}$. Précisons que l'on considère ici les variables σ_i à valeur dans $\{0, 1\}$. Soient deux variables aléatoires réelles X_i et X_j, dont les fonctions de répartition sont respectivement F_i et F_j, ainsi que deux variables binaires σ_i et σ_j. On souhaite construire le modèle d'interaction par paire décrit par la figure 5.1 page 100. La distribution du vecteur $(X_i, X_j, \sigma_i, \sigma_j)$ selon ce modèle est

$$\mathbb{P}(X_i \leq x_i, X_j \leq x_j, \sigma_i = s_i, \sigma_j = s_j) = p_{ij}(s_i, s_j) F_i^{s_i}(x_i) F_j^{s_j}(x_j). \tag{5.10}$$

Les variables σ_i et σ_j étant binaires, $p_{ij}(s_i, s_j)$ s'exprime au moyen de 3 paramètres indépendants,

$$\begin{aligned} p_{ij}(s_i, s_j) = p_{ij}^{11} s_i s_j + \left(p_j^1 - p_{ij}^{11}\right) \bar{s}_i s_j + \left(p_i^1 - p_{ij}^{11}\right) s_i \bar{s}_j \\ + \left(1 - p_i^1 - p_j^1 + p_{ij}^{11}\right) \bar{s}_i \bar{s}_j. \end{aligned}$$

en utilisant la notation $\bar{s} \stackrel{\text{def}}{=} 1 - s$ et avec

$$\begin{aligned} p_i^1 &\stackrel{\text{def}}{=} \mathbb{P}(\sigma_i = 1) = \mathbb{E}(\sigma_i), \\ p_{ij}^{11} &\stackrel{\text{def}}{=} \mathbb{P}(\sigma_i = 1, \sigma_j = 1) = \mathbb{E}(\sigma_i \sigma_j). \end{aligned}$$

Pour obtenir une loi de probabilité valide, on doit avoir p_i^1 et $p_j^1 \in [0, 1]$ et

$$p_{ij}^{11} \in \mathbb{D}(p_i^1, p_j^1) \stackrel{\text{def}}{=} \left[\max(0, p_i^1 + p_j^1 - 1), \min(p_i^1, p_j^1)\right].$$

Jusqu'ici, nous avons été en mesure de faire des choix optimaux – en un certain sens – pour la calibration de la loi du vecteur $\boldsymbol{\sigma}$. Le nombre de paramètres ne sera cependant pas suffisant pour encoder de manière exacte toutes les dépendances comme on va le voir dans la section 5.2.1.

On se concentre ensuite sur la manière de déterminer les lois p_{ij} sans chercher à savoir comment construire à partir de celles-ci la loi jointe de $\boldsymbol{\sigma}$. On reviendra sur ce point en fin de section. On va présenter deux possibilités d'estimation des lois p_{ij} : une méthode conduisant à des formules explicites basée sur la méthode des moments (section 5.2.2), puis une méthode de maximisation itérative de la vraisemblance (section 5.2.3).

5.2.1 Capacité des fonctions d'encodage

Avant de passer à l'estimation de la loi jointe des variables latentes on cherche ici à quantifier la perte d'information induite par notre modélisation. On s'intéresse en particulier au cas de la modélisation de l'interaction entre une paire de variables (X_1, X_2) via les variables binaires latentes (σ_1, σ_2). Pour quantifier cette perte d'information, on commence par exprimer la divergence de Kullback-Leibler entre la distribution empirique jointe $\mathcal{P}$ de (X_1, X_2) et celle correspondant à notre modèle que l'on note $\mathbb{P}$.

$$\begin{aligned} D_{\mathrm{KL}}(\mathcal{P}||\mathbb{P}) &= \int \mathcal{P}(x_1, x_2) \log \frac{\mathcal{P}(x_1, x_2)}{\mathbb{P}(x_1, x_2)} dx_1 dx_2 \\ &= I_{\mathcal{P}}(X_1, X_2) + \int \mathcal{P}(x_1, x_2) \log \frac{\mathcal{P}(x_1)\mathcal{P}(x_2)}{\mathbb{P}(x_1, x_2)} dx_1 dx_2, \end{aligned}$$

avec $I_{\mathcal{P}}(X_1, X_2)$ l'information mutuelle entre les variables X_i et X_j selon $\mathcal{P}$. Le terme de droite s'interprète comme l'opposée d'une information mutuelle entre X_1 et X_2 dans notre modèle mais vis à vis de la mesure empirique $\mathcal{P}$. En effet il s'exprime comme

$$\mathbb{I}(X_1, X_2) \overset{\text{def}}{=} \int \mathcal{P}(x_1, x_2) \log \left(\frac{\sum_{s_1, s_2} \mathbb{P}(s_1, s_2)\mathbb{P}(x_1|s_1)\mathbb{P}(x_2|s_2)}{\mathbb{P}(x_1)\mathbb{P}(x_2)} \right),$$

en utilisant le fait que $\mathbb{P}(x_i) = \mathcal{P}(x_i)$ et en développant $\mathbb{P}(x_1, x_2)$ vis-à-vis des variables σ_1 et σ_2. Au final la divergence de Kullback-Leibler s'exprime donc comme une différence d'information mutuelles

$$D_{\mathrm{KL}}(\mathcal{P}||\mathbb{P}) = I_{\mathcal{P}}(X_1, X_2) - \mathbb{I}(X_1, X_2).$$

Dans le cas d'une fonction d'encodage quelconque on ne va pas obtenir l'expression exacte de $\mathbb{I}(X_1, X_2)$ en terme des variables binaires. On va donc construire une borne supérieure de $\mathbb{I}(X_1, X_2)$. Rappelons tout d'abord la forme des distributions conditionnelles

$$\mathbb{P}(X_i|\sigma_i) \overset{\text{def}}{=} \frac{\Lambda^{\sigma_i}(X_i)\mathcal{P}(X_i)}{\mathbb{P}(\sigma_i)},$$

avec $\Lambda^1 \overset{\text{def}}{=} \Lambda$ et $\Lambda^0 \overset{\text{def}}{=} 1 - \Lambda$. On a alors

$$\begin{aligned} \mathbb{I}(X_1, X_2) &= \int \mathcal{P}(x_1, x_2) \log \left(\sum_{\sigma_1, \sigma_2} \frac{\mathbb{P}(\sigma_1, \sigma_2)}{\mathbb{P}(\sigma_1)\mathbb{P}(\sigma_2)} \Lambda^{\sigma_1}(x_1)\Lambda^{\sigma_2}(x_2) \right) dx_1 dx_2, \\ &\leq \log \left(\int \mathcal{P}(x_1, x_2) \sum_{\sigma_1, \sigma_2} \frac{\mathbb{P}(\sigma_1, \sigma_2)}{\mathbb{P}(\sigma_1)\mathbb{P}(\sigma_2)} \Lambda^{\sigma_1}(x_1)\Lambda^{\sigma_2}(x_2) dx_1 dx_2 \right), \\ &= \log \left(\sum_{\sigma_1, \sigma_2} \frac{\mathbb{P}(\sigma_1, \sigma_2)}{\mathbb{P}(\sigma_1)\mathbb{P}(\sigma_2)} \mathbb{E}_{\mathcal{P}}[\Lambda^{\sigma_1}(X_1)\Lambda^{\sigma_2}(X_2)] \right). \end{aligned}$$

En posant $\mathcal{P}_{\sigma_1\sigma_2} \stackrel{\text{def}}{=} \mathbb{E}_{\mathcal{P}}[\Lambda^{\sigma_1}(X_1)\Lambda^{\sigma_2}(X_2)]$, on obtient finalement

$$\begin{aligned}\mathbb{I}(X_1, X_2) &\leq \log\left(\sum_{\sigma_1,\sigma_2} \frac{\mathbb{P}(\sigma_1,\sigma_2)}{\mathbb{P}(\sigma_1)\mathbb{P}(\sigma_2)}\mathcal{P}_{\sigma_1\sigma_2}\right), \\ &\leq \log\left(\sum_{\sigma_1,\sigma_2} \frac{1}{\mathbb{P}(\sigma_1)}\mathcal{P}_{\sigma_1\sigma_2}\right) = \log\left(\sum_{\sigma_1} 1\right) = \log(2),\end{aligned}$$

car on a $\mathbb{P}(\sigma_1,\sigma_2) \leq \mathbb{P}(\sigma_2)$ et $\sum_{\sigma_2} \mathcal{P}_{\sigma_1\sigma_2} = \mathbb{P}(\sigma_1)$. Cette borne nous donne une information sur la quantité maximale d'information que l'on peut espérer obtenir par notre modélisation. Ceci est résumé par la proposition suivante.

Proposition 5.5. *Lorsque l'information mutuelle $I_{\mathcal{P}}(X_1, X_2)$ entre les variables réelles est strictement supérieure à* $\log(2)$*, notre modèle n'est pas capable d'encoder parfaitement la distribution jointe de X_1 et X_2 et ce quelle que soit la fonction d'encodage choisie.*

Ce résultat était attendu. Quelle que soit la définition des variables binaires on ne peut transférer plus d'un bit d'information entre deux variables binaires. Il est donc illusoire d'espérer modéliser de manière précise la distribution jointe du vecteur réel $\mathbf{X}$ à travers notre modèle d'Ising latent. La tâche que l'on s'est fixée ici est cependant plus modeste : on souhaite effectuer des prédictions sur les variables réelles. On verra dans la section suivante que l'on cherche en fait à résoudre un problème de régression sur les variables X_i qui est par nature très différent de la modélisation de la distribution $\mathcal{P}$. En particulier, comme on le verra à la section 5.3, il est possible d'obtenir des performances quasi-optimales pour le problème de prédiction lorsque l'information mutuelle entre variables est strictement supérieur à $\log(2)$.

5.2.2 Estimation par la méthode des moments

Les résultats de la section 5.1.1 nous fournissent de manière naturelle une estimation des distributions marginales p_i^1 ; en effet, on a vu que

$$p_i^1 = \mathbb{E}[\Lambda_i(X_i)] = \int_{\mathcal{X}_i} \Lambda_i(x) dF_i(x).$$

On peut donc estimer ces paramètres par les moments empiriques et il reste alors à fixer les paramètres de corrélation p_{ij}^{11}. La forme de la dépendance entre X_i et X_j, dans le modèle de la figure 5.1 page 100, est obtenue en déconditionnant (5.10) par rapport aux variables σ_i et σ_j. La distribution jointe de (X_i, X_j) s'exprime alors comme

$$\mathbb{P}_{p_{\boldsymbol{\sigma}}}(X_i \leq x_i, X_j \leq x_j) = \sum_{s_i,s_j} p_{ij}(s_i, s_j) F_i^{s_i}(x_i) F_j^{s_j}(x_j).$$

On propose donc d'estimer les marginales de paires p_{ij} par la méthode des moments correspondant aux contraintes suivantes :

$$\mathbb{E}[\Lambda_i(X_i)] = \langle \Lambda_i(X_i) \rangle, \forall i \in \mathbb{V}, \tag{5.11}$$
$$\mathbb{E}[\Lambda_i(X_i)\Lambda_j(X_j)] = \langle \Lambda_i(X_i)\Lambda_j(X_j) \rangle, \forall (i,j) \in \mathbb{E}. \tag{5.12}$$

où $\mathbb{E}$ correspond à l'espérance sous la loi de $\boldsymbol{\sigma}$ et $\langle \cdot \rangle$ est la moyenne empirique. La dernière contrainte (5.12) s'inspire des deux premières qui apparaissent naturellement via la définition des variables $\boldsymbol{\sigma}$. L'espérance $\mathbb{E}\left[\Lambda_i(X_i)\Lambda_j(X_j)\right]$ s'exprime comme

$$\mathbb{E}[\Lambda_i(X_i)\Lambda_j(X_j)] = \sum_{s_i, s_j} p_{ij}(s_i, s_j) \lambda_i^{s_i} \lambda_j^{s_j}, \tag{5.13}$$

avec

$$\lambda_i^s \overset{\text{def}}{=} \mathbb{E}\left[\Lambda_i(X) | \sigma_i = s\right] = \int_{\mathcal{X}_i} \Lambda_i(x)\, dF_i^s(x).$$

Considérons M réalisations indépendantes $(x_i^k)_{k \leq M}$ de la variable X_i. Ces espérances conditionnelles peuvent être estimées par les quantités empiriques suivantes

$$\hat{\lambda}_i^1 \overset{\text{def}}{=} \frac{1}{M \cdot \mathbb{P}(\sigma_i = 1)} \sum_{k=1}^{M} \Lambda_i(x_i^k)^2,$$
$$\hat{\lambda}_i^0 \overset{\text{def}}{=} \frac{1}{M \cdot \mathbb{P}(\sigma_i = 0)} \sum_{k=1}^{M} (1 - \Lambda_i(x_i^k))\Lambda_i(x_i^k),$$

en remarquant tout simplement que

$$\lambda_i^1 = \frac{\mathbb{E}[\Lambda_i(X_i)^2]}{\mathbb{P}(\sigma_i = 1)} \quad \text{et} \quad \lambda_i^0 = \frac{\mathbb{E}[(1 - \Lambda_i(X_i))\Lambda_i(X_i)]}{\mathbb{P}(\sigma_i = 0)}.$$

p_{ij}^{11} est alors choisi de sorte que (5.12) soit vérifiée, avec $\hat{\lambda}^0$ et $\hat{\lambda}^1$ comme estimateurs des espérances conditionnelles de (5.13)

$$p_{ij}^{11} = p_i^1 p_j^1 + \frac{\widehat{\text{cov}}(\Lambda_i(X_i), \Lambda_j(X_j))}{(\hat{\lambda}_i^1 - \hat{\lambda}_i^0)(\hat{\lambda}_j^1 - \hat{\lambda}_j^0)},$$

où $\widehat{\text{cov}}(\Lambda_i(X_i), \Lambda_j(X_j))$ est un estimateur de la covariance de $\Lambda_i(X_i)$ et $\Lambda_j(X_j)$. Cette relation peut aussi s'exprimer en termes de variances et covariances,

$$\text{cov}(\sigma_i, \sigma_j) = \widehat{\text{cov}}(\Lambda_i(X_i), \Lambda_j(X_j)) \frac{\text{var}(\sigma_i)\,\text{var}(\sigma_j)}{\text{var}(\Lambda_i(X_i))\,\text{var}(\Lambda_j(X_j))},$$

et l'on voit donc que lorsque le ratio des variances devient trop élevé, l'équation d'égalité des moments (5.12) peut ne plus être satisfaite puisque l'on doit toujours avoir $p_{ij}^{11} \in \mathbb{D}(p_i^1, p_j^1)$. Ce coefficient sature donc à une valeur limite pour laquelle la différence entre les moments est la plus faible possible.

Exemple 5.1. *Dans le cas particulier de $\Lambda_i^S = F_i$, qui correspond à considérer les corrélations de rang, de nombreuses quantités sont calculables explicitement. On a notamment*

$$\mathbb{P}(\sigma_i = 1) = \mathbb{E}[F_i(X_i)] = \frac{1}{2},$$

dès que X_i n'a pas de point d'accumulation. De plus

$$\lambda_i^1 = \frac{2}{3} \quad et \quad \lambda_i^1 = \frac{1}{3}.$$

La méthode des moments conduit donc à l'expression suivante

$$p_{ij}^{11} = \frac{1}{4} + 9\ \widehat{\mathrm{cov}}(F_i(X_i), F_j(X_j)),$$

et seule la covariance des rangs doit être estimée empiriquement.

Le cas de la fonction seuil Λ_{MI} est encore plus simple. On a alors

$$\mathbb{P}(\sigma_i = 1) = \mathbb{E}\left[X_i \geq q_X^{0,5}\right] = \frac{1}{2}$$

et

$$p_{ij}^{11} = \langle \Lambda_i(X_i)\Lambda_j(X_j) \rangle,$$

qui est donc la seule quantité qui doit être empiriquement estimée. Cette quantité est la probabilité que X_i et X_j soient simultanément supérieurs à leurs médianes respectives.

5.2.3 Estimation par maximisation de la vraisemblance

On suppose de plus ici que les variables aléatoires admettent des densités. On notera la densité jointe de (X_i, X_j) associée à une distribution $p_{\boldsymbol{\sigma}}$.

$$f_{p_{\boldsymbol{\sigma}}}^{ij}(x_i, x_j) \stackrel{\text{def}}{=} \sum_{s_i, s_j} p_{ij}(s_i, s_j) f_i^{s_i}(x_i) f_j^{s_j}(x_j),$$

où $f_i^{s_i}$ est la densité associée à $dF_i^{s_i}$. Exprimons tout d'abord le logarithme de la vraisemblance des observations $\mathbf{x}$ correspondant à une distribution $\mathbb{P}_{\boldsymbol{\sigma}}$ du vecteur $\boldsymbol{\sigma}$. On rappelle que l'on suppose que les observations de paires sont indépendantes.

$$\begin{aligned}
L(\mathbf{x}, p_{\boldsymbol{\sigma}}) &= \sum_{(ij)\in\mathbb{E}} \sum_{k=1}^{N_{ij}} \log\left(f_{p_{\boldsymbol{\sigma}}}^{ij}(X_i = x_{i,j}^k, X_j = x_{j,i}^k)\right) \\
&= \sum_{(ij)\in\mathbb{E}} \sum_{k=1}^{N_{ij}} \log\left(\sum_{s_i, s_j} p_{\boldsymbol{\sigma}}(\sigma_i = s_i, \sigma_j = s_j) f_i^{s_i}(x_{i,j}^k) f_j^{s_j}(x_{j,i}^k)\right) \\
&= \sum_{(ij)\in\mathbb{E}} \sum_{k=1}^{N_{ij}} \log\left(\sum_{s_i, s_j} p_{ij}(s_i, s_j) f_i^{s_i}(x_{i,j}^k) f_j^{s_j}(x_{j,i}^k)\right)
\end{aligned}$$

La présence des variables « cachées » σ_i et σ_j introduit une somme dans le logarithme. Il ne sera donc pas possible d'exprimer analytiquement la forme des lois p_{ij} qui maximise $L(\mathbf{x}, p_{\boldsymbol{\sigma}})$. L'approche classique consiste alors à utiliser l'algorithme « Expectation Maximization »(EM) introduit par Dempster *et al.* [25]. L'idée consiste à maximiser itérativement une borne inférieure de la vraisemblance. Pour des raisons pratiques, on pose la notation suivante :

$$\begin{aligned} p_{ij}(s_i, s_j|x_i, x_j) &\stackrel{\text{def}}{=} \mathbb{P}_{p_{\boldsymbol{\sigma}}}(\sigma_i = s_i, \sigma_j = s_j|X_i = x_i, X_j = x_j) \\ &= \frac{p_{ij}(s_i, s_j) f_i^{s_i}(x_i) f_j^{s_j}(x_j)}{\sum_{s'_i, s'_j} p_{ij}(s'_i, s'_j) f_i^{s'_i}(x_i) f_j^{s'_j}(x_j)}. \end{aligned}$$

On considère une distribution de référence $p_{\boldsymbol{\sigma}}^{(n)}$ et on écrit la vraisemblance comme

$$L(\mathbf{x}, p_{\boldsymbol{\sigma}}) = \sum_{ij,k} \log \left(\sum_{s_i, s_j} p_{ij}(s_i, s_j) f_i^{s_i}(x_{i,j}^k) f_j^{s_j}(x_{j,i}^k) \frac{p_{ij}^{(n)}(s_i, s_j|x_{i,j}^k, x_{j,i}^k)}{p_{ij}^{(n)}(s_i, s_j|x_{i,j}^k, x_{j,i}^k)} \right),$$

ce qui nous permet de construire la borne inférieure suivante

$$\begin{aligned} L(\mathbf{x}, p_{\boldsymbol{\sigma}}) &\geq \sum_{ij,k} \sum_{s_i, s_j} p_{ij}^{(n)}(s_i, s_j|x_{i,j}^k, x_{j,i}^k) \log \left(\frac{p_{ij}(s_i, s_j) f_i^{s_i}(x_{i,j}^k) f_j^{s_j}(x_{j,i}^k)}{p_{ij}^{(n)}(s_i, s_j|x_{i,j}^k, x_{j,i}^k)} \right), \\ &\stackrel{\text{def}}{=} \Delta(p_{\boldsymbol{\sigma}}||p_{\boldsymbol{\sigma}}^{(n)}). \end{aligned}$$

Cette borne $\Delta(p_{\boldsymbol{\sigma}}||p_{\boldsymbol{\sigma}}^{(n)})$ est obtenue grâce à l'inégalité de Jensen – la fonction logarithme est concave – et en utilisant le fait que

$$\sum_{s_i, s_j} p_{ij}^{(n)}(s_i, s_j|x_{i,j}^k, x_{j,i}^k) = 1.$$

On va donc maximiser cette borne $\Delta(p_{\boldsymbol{\sigma}}||p_{\boldsymbol{\sigma}}^{(n)})$ pour obtenir une nouvelle distribution $p_{\boldsymbol{\sigma}}^{(n+1)}$. On simplifie la fonction à maximiser en ôtant tous les termes qui ne dépendent pas de $p_{\boldsymbol{\sigma}}$. On obtient finalement le problème de maximisation suivant

$$p_{\boldsymbol{\sigma}}^{(n+1)} = \underset{p_{\boldsymbol{\sigma}}}{\operatorname{argmax}} \quad \ell(p_{\boldsymbol{\sigma}}||p_{\boldsymbol{\sigma}}^{(n)}),$$

avec

$$\ell(p_{\boldsymbol{\sigma}}||p_{\boldsymbol{\sigma}}^{(n)}) \stackrel{\text{def}}{=} \sum_{(ij),k} \sum_{s_i, s_j} p_{ij}^{(n)}(s_i, s_j|x_{i,j}^k, x_{j,i}^k) \log \left(p_{ij}(s_i, s_j) \right).$$

Le nom l'algorithme EM vient du fait que l'on va successivement calculer une espérance selon la loi $p_{\boldsymbol{\sigma}}^{(n)}$, qui apparaît dans la forme de $\ell(p_{\boldsymbol{\sigma}}||p_{\boldsymbol{\sigma}}^{(n)})$, puis chercher à maximiser cette espérance comme fonction de $p_{\boldsymbol{\sigma}}$ pour obtenir

$p_{\boldsymbol{\sigma}}^{(n+1)}$. Calculons donc les dérivées de $\ell(p_{\boldsymbol{\sigma}}||p_{\boldsymbol{\sigma}}^{(n)})$ par rapport aux paramètres

$$\frac{\partial \ell(p_{\boldsymbol{\sigma}}||p_{\boldsymbol{\sigma}}^{(n)})}{\partial p_{ij}^{11}} = \sum_{k=1}^{N_{ij}} \sum_{s_i,s_j} \left(2\mathbb{1}_{\{s_i=s_j\}} - 1\right) \frac{p_{ij}^{(n)}(s_i,s_j|x_i^k,x_j^k)}{p_{ij}(s_i,s_j)}$$
$$\frac{\partial \ell(p_{\boldsymbol{\sigma}}||p_{\boldsymbol{\sigma}}^{(n)})}{\partial p_{i}^{1}} = \sum_{j\in\partial i} \sum_{k=1}^{N_{ij}} \sum_{s_i} (2s_i - 1) \frac{p_{ij}^{(n)}(s_i,0|x_i^k,x_j^k)}{p_{ij}(s_i,0)}$$

En cherchant à annuler les dérivées ℓ par rapport à p_{ij}^{11}, on obtient une solution triviale qui correspond à

$$p_{ij}(s_i,s_j) = \frac{1}{N_{ij}} \sum_{k=1}^{N_{ij}} p_{ij}^{(n)}(s_i,s_j|x_i^k,x_j^k).$$

Avec ce choix, on obtient de plus

$$\frac{\partial \ell(p_{\boldsymbol{\sigma}}||p_{\boldsymbol{\sigma}}^{(n)})}{\partial p_{i}^{1}} = \sum_{j\in\partial i} \sum_{s_i} (2s_i - 1) N_{ij} = 0.$$

La fonction que l'on maximise étant concave par construction – puisque somme de logarithmes – cette solution correspond à l'unique point stationnaire de $\ell(p_{\boldsymbol{\sigma}}||p_{\boldsymbol{\sigma}}^{(n)})$. On déduit donc l'algorithme EM suivant

$$\begin{aligned}
p_{ij}^{(n+1)}(1,1) &= \frac{1}{N_{ij}} \sum_{k=1}^{N_{ij}} \frac{\psi_{ij}^{(n)}(1,1)\Lambda_i(x_i^k)\Lambda_j(x_j^k)}{Z_{ij}(x_i^k,x_j^k)},\\
p_{ij}^{(n+1)}(1,0) &= \frac{1}{N_{ij}} \sum_{k=1}^{N_{ij}} \frac{\psi_{ij}^{(n)}(1,0)\Lambda_i(x_i^k)\left(1-\Lambda_j(x_j^k)\right)}{Z_{ij}(x_i^k,x_j^k)},\\
p_{ij}^{(n+1)}(0,1) &= \frac{1}{N_{ij}} \sum_{k=1}^{N_{ij}} \frac{\psi_{ij}^{(n)}(0,1)\left(1-\Lambda_i(x_i^k)\right)\Lambda_j(x_j^k)}{Z_{ij}(x_i^k,x_j^k)},\\
p_{ij}^{(n+1)}(0,0) &= \frac{1}{N_{ij}} \sum_{k=1}^{N_{ij}} \frac{\psi_{ij}^{(n)}(0,0)\left(1-\Lambda_i(x_i^k)\right)\left(1-\Lambda_j(x_j^k)\right)}{Z_{ij}(x_i^k,x_j^k)},
\end{aligned} \tag{5.14}$$

avec

$$\psi_{ij}^{(n)}(s_i,s_j) \stackrel{\text{def}}{=} \frac{p_{ij}^{(n)}(s_i,s_j)}{p_i^{(n)}(s_i)p_j^{(n)}(s_j)},$$

et

$$\begin{aligned}
Z_{ij}(x_i,x_j) &= \psi_{ij}^{(n)}(1,1)\Lambda_i(x_i)\Lambda_j(x_j) + \psi_{ij}^{(n)}(1,0)\Lambda_i(x_i)(1-\Lambda_j(x_j))\\
&+ \psi_{ij}^{(n)}(0,1)(1-\Lambda_i(x_i))\Lambda_j(x_j) + \psi_{ij}^{(n)}(0,0)(1-\Lambda_i(x_i))(1-\Lambda_j(x_j)).
\end{aligned} \tag{5.15}$$

Les règles de mises à jour obtenues (5.14) sont donc relativement simples. Malheureusement le point stationnaire de $\ell(p_{\boldsymbol{\sigma}}||p_{\boldsymbol{\sigma}}^{(n)})$ auquel elles correspondent

n'est pas nécessairement dans le domaine de $p_{\boldsymbol{\sigma}}$. En effet, les lois p_{ij} ne sont pas indépendantes et doivent être compatibles. En particulier, on doit avoir

$$\sum_{s_j} p_{ij}(1, s_j) = \sum_{s_k} p_{ik}(1, s_k) = p_i^1, \quad \forall i, s_i.$$

Si ce n'est pas le cas, cela signifie qu'il n'existe pas de point stationnaire à l'intérieur du domaine de $p_{\boldsymbol{\sigma}}$. Le maximum de ℓ est alors atteint à un point du bord, ce qui complique sa recherche.

Pour résoudre ce problème, on propose tout d'abord une méthode mixte, entre maximisation de la vraisemblance et méthode des moments, qui va consister à fixer les lois marginales p_i^1 via les méthodes des moments et (5.11) puis à optimiser la vraisemblance comme fonction des paramètres p_{ij}^{11}. On est alors amené à résoudre $|\mathbb{E}|$ problèmes de maximisation distincts, et l'on peut mettre à jour p_{ij}^{11} selon la première ligne de (5.14). On vérifie alors que les paramètres obtenus sont bien valides, c'est à dire $p_{ij}^{11} \in \mathbb{S}(p_i^1, p_j^1)$. Si ce n'est pas le cas, cela signifie que le paramètre sature une des deux bornes. On dénotera EM-MM cette approche dans la suite.

Une autre approche possible est l'utilisation d'un algorithme du type EM généralisé. Cet algorithme, décrit par Neal et Hinton [80], relâche le fait de trouver le maximum global ℓ à chaque itération. On peut alors proposer de mettre à jour les coefficients p_i^1, p_j^1 et p_{ij}^{11} paire par paire, suivant les équations (5.14). A chaque itération, les coefficients p_i^1 sont donc bien définis. Il reste juste à vérifier que l'on a bien pour toutes les paires $(ij) \in \mathbb{E}$, $p_{ij}^{11} \in \mathbb{S}(p_i^1, p_j^1)$, ce qui est facile à faire à chaque itération. Cette méthode est cependant plus complexe à mettre en œuvre. Dans la suite on privilégiera l'approche par la méthode des moments (MM) ou l'approche mixte (EM-MM).

Pour conclure remarquons que dans le cas de la fonction d'encodage déterministe Λ_{MI} l'estimation par l'algorithme EM est en fait strictement équivalente à l'approche par la méthode des moments. Ceci apparaît lorsque l'on regarde les équations de mises à jour (5.14).

5.2.4 Estimation compatible avec BP

On s'intéresse ici au problème qui consiste à fixer la distribution jointe $p_{\boldsymbol{\sigma}}$ du vecteur $\boldsymbol{\sigma}$ à partir de ses marginales $\{p_{ij}\}_{(i,j)\in\mathbb{E}}$. Notons tout d'abord que, comme on l'a vu au chapitre 2, le fait d'avoir des marginales compatibles ne garantit pas l'existence d'une loi jointe $p_{\boldsymbol{\sigma}}$ telle que

$$\forall (i,j) \in \mathbb{E}, \forall s_i, \sum_{\mathbf{s}_{\mathbb{V}\setminus i,j}} p_{\boldsymbol{\sigma}}(\boldsymbol{\sigma} = \mathbf{s}) = p_{ij}(s_i, s_j).$$

Dans le cas ou le graphe $\mathcal{G} = (\mathbb{V}, \mathbb{E})$ ne contient pas de cycles, la situation est cependant plus simple. La distribution jointe est alors déterminée de

manière unique par la donnée de ses marginales de paires. Cette loi jointe s'exprime sous la forme

$$p_{\boldsymbol{\sigma}}(\boldsymbol{\sigma} = \mathbf{s}) = \prod_{(i,j)\in\mathbb{E}} \frac{p_{ij}(s_i, s_j)}{p_i(s_i)p_j(s_j)} \prod_{i\in\mathbb{V}} p_i(s_i), \tag{5.16}$$

avec p_i la loi marginale de p_{ij} – qui ne dépend pas de j par hypothèse.

Dans le cas plus général d'un graphe cyclique, la situation est plus complexe. Ce problème inverse sur le modèle d'Ising est très étudié en physique statistique. Il est potentiellement NP-difficile – voire peut ne pas avoir de solution – seules des méthodes approchées peuvent donc être envisagées pour des réseaux de grande taille. Wainwright [98] propose une approche particulièrement intéressante qui consiste à tenir compte du fait qu'une fois la loi $p_{\boldsymbol{\sigma}}$ fixée de manière approchée l'inférence elle aussi sera réalisée de manière approchée. Il propose donc que ces deux approximations soient compatibles, dans le cas présent cela consiste à choisir la loi $p_{\boldsymbol{\sigma}}$ selon l'approximation de Bethe (5.16). Ce choix permet d'obtenir l'ensemble de beliefs

$$b_{ij}(s_i, s_j) = p_{ij}(s_i, s_j) \qquad b_i(s_i) = p_i(s_i), \tag{5.17}$$

comme point fixe de BP comme on l'a vu au chapitre 3. De plus, on peut montrer que tout autre choix de $p_{\boldsymbol{\sigma}}$ admettant (5.17) comme point fixe de BP est équivalent. Ceci est une conséquence de la propriété de reparamétrisation de BP (section 2.4 page 36). Si ce choix est un bon point de départ, le point fixe obtenu est en général instable et il est nécessaire de l'améliorer. Pour cela il est possible, entre autres, de s'appuyer sur la théorie de la réponse linéaire (Welling et Teh [106] ; Yasuda et Tanaka [109] ; Mézard et Mora [71]). Il est aussi possible de déformer le modèle par quelques paramètres faciles à calibrer (Furtlehner *et al.* [34]).

5.3 Expérimentations numériques

Nous appliquons ici les méthodes décrites dans ce chapitre et le chapitre 4 à des données synthétiques, dans le but de comprendre leurs comportements. Pour pouvoir se concentrer autant que possible sur l'aspect d'encodage/décodage, on considère des structures de graphes simples. On regarde successivement trois cas de complexité croissante

- le cas d'une simple paire (X_1, X_2) de variables réelles,
- le cas d'une chaîne de Markov (figure 5.5 page 121) et enfin
- le cas d'arbres réguliers, c'est-à-dire dont la connectivité est fixée.

On répète alors *l'expérience de décimation* suivante : pour une réalisation jointe du vecteur $\mathbf{X}$ on révèle ses composantes X_i dans un ordre aléatoire et l'on prédit les variables non observées en utilisant l'algorithme mirror BP. Nous allons alors pouvoir comparer les performances des différents choix d'encodage/décodage en fonction de la proportion de variables observées. Cette

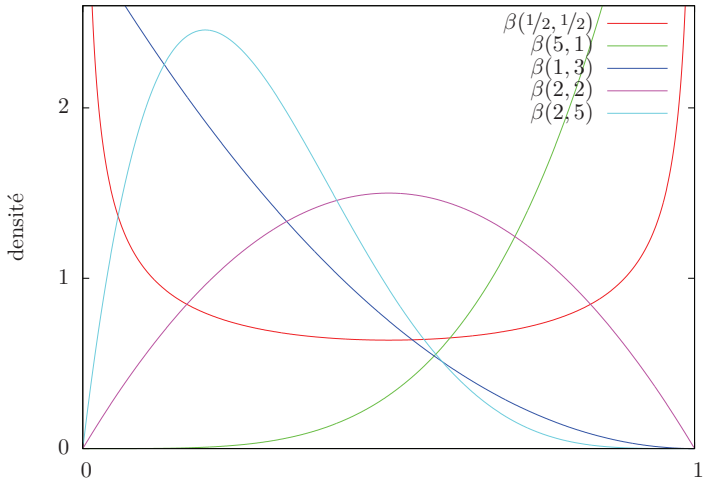


FIGURE 5.3 : Quelques exemples de densités de variables de loi $\beta(a, b)$. On remarque la variété de cas possibles suivant la valeur des paramètres a et b.

expérience correspond à celle décrite dans le chapitre 1 où l'on a supposé que l'identité des variables observées ainsi que leur nombre ne sont pas contrôlés.

Dans la suite on considérera les choix de fonctions d'encodage/décodage suivant

- la fonction de seuil Λ_{MI} avec le décodage probabiliste (5.8),
- la fonction de répartition F avec le décodage direct $\Gamma^{\mathcal{D}} = F^{-1}$,
- la fonction de répartition F avec le décodage probabiliste (5.9),
- la fonction Λ_{DE} avec le décodage direct $\Gamma^{\mathcal{D}}$.

Chacun de ces choix fournit un estimateur θ dont on calculera la performance au sens de la mesure L^1

$$\mathbb{E}_X\Big[|\theta(X) - X|\Big]. \tag{5.18}$$

Génération des modèles. Pour générer ces modèles synthétiques, on définit un vecteur gaussien $\mathbf{Y}$ en générant une matrice de précision, qui est la matrice inverse de la matrice de covariance, dont le support est le graphe correspondant à l'un des trois cas précédents. On obtient cette matrice de précision en générant aléatoirement les corrélations partielles– c'est-à-dire les entrées de la matrice de précision de $\mathbf{Y}$ – via des variables aléatoires uniformes sur $[-1, 1]$. Malheureusement cette procédure ne permet pas de générer à coup sûr une matrice de précision définie positive. On prend donc cette matrice comme point de départ et on réduit progressivement les corrélations de plus grande magnitude jusqu'à obtenir une matrice définie positive.

On génère alors des réalisations de ce vecteur gaussien que l'on transforme, en utilisant l'application qui transforme une variable gaussienne $\mathcal{N}(0, 1)$ en une

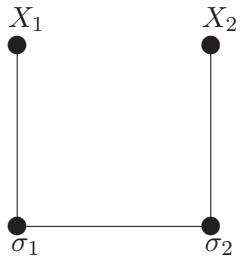


FIGURE 5.4 : Modèle joint pour $(X_1, X_2, \sigma_1, \sigma_2)$. La vraie distribution jointe entre X_1 et X_2 est approchée à travers les variables latentes σ_1 et σ_2.

variable $\beta(a, b)$, pour des valeurs de a et b données. On rappelle que la densité $f^{\beta}_{a,b}$ d'une variable de loi $\beta(a, b)$ est

$$f^{\beta}_{a,b}(x) = \frac{1}{B(a,b)} x^{a-1}(1-x)^{b-1}\mathbb{1}_{[0,1]}(x),$$

où la constante de normalisation $B(a, b)$ est la fonction bêta. Plus précisément chaque composante du vecteur $\mathbf{X}$ est alors définie comme

$$X_i = F^{-1}_{\beta(a,b)}\left(F_{\mathcal{N}(0,1)}(Y_i)\right),$$

avec $F_{\beta(a,b)}$ et $F_{\mathcal{N}(0,1)}$ les fonctions de répartition d'une variable de loi $\beta(a, b)$ et d'une variable de loi $\mathcal{N}(0, 1)$. La procédure utilisée pour générer le modèle nous permet de réaliser l'inférence exacte en raisonnant sur le vecteur gaussien. Le choix des distributions $\beta(a, b)$ est justifié par la variabilité de cas envisageables suivant les paramètres a et b. On peut obtenir des distributions quasi-binaires, unimodales ainsi qu'une distribution uniforme sur le segment $[0, 1]$. Quelques exemples de densités $\beta(a, b)$ sont représentées à la figure 5.3.

Cas d'une paire (X_1, X_2) de variables réelles. On commence par considérer le cas dégénéré où le vecteur $\mathbf{X}$ est constitué de deux variables réelles (figure 5.4). On répète 100 000 fois l'expérience de décimation. Dans le cas présent cette expérience se réduit trivialement à observer successivement X_i ou X_j pour une réalisation du couple (X_i, X_j) et à prédire la variable non observée. En plus de la performance L^1 on calcule ici le biais des différents estimateurs θ

$$\mathbb{E}_X\left[\theta(X) - \hat{\theta}_1(X)\right]. \tag{5.19}$$

On rappelle que $\hat{\theta}_1(X)$ est l'estimateur optimal au sens de l'erreur L^1 *i.e.* la médiane conditionnelle. Les choix envisagés pour l'estimation de la loi p_{12} sont la méthode des moments (MM) (5.12) et l'approche mixte (EM-MM).

Les résultats de l'expérience, pour différentes valeurs de a, b et $\rho \stackrel{\text{def}}{=} \text{cov}(Y_i, Y_j)$, sont donnés dans la table 5.1. La première ligne correspond à la

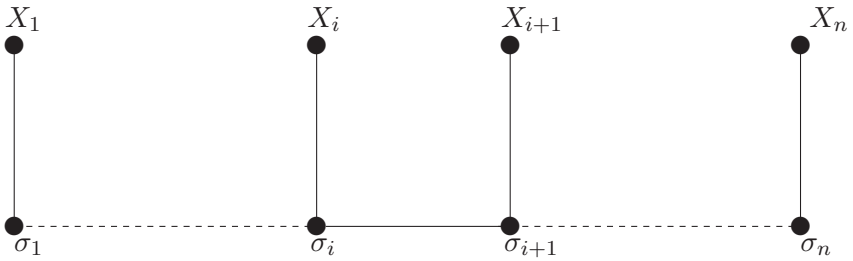


FIGURE 5.5 : Modèle joint pour $(\mathbf{X}, \boldsymbol{\sigma})$ dans le cas où la vraie distribution du vecteur $\mathbf{X}$ correspond à une chaîne de Markov.

performance L^1 (5.18) et la seconde au biais (5.19). D'une manière générale, à corrélation faible tous les estimateurs se comportent de manière satisfaisante. On remarque cependant que le choix d'encodage/décodage qui se comporte le mieux est la fonction de répartition $\Lambda = F$ avec l'inversion $\Gamma^{\mathcal{D}} = F^{-1}$. Malgré le fait qu'il ne constitue pas le choix permettant de minimiser l'écart à la distribution du vecteur $\mathbf{X}$ (voir section 5.2.1) il semble tout à fait à même de résoudre le problème d'inférence. Comme on pouvait s'y attendre l'aspect conservateur du choix de décodage probabiliste $\Gamma^{\mathcal{P}}$ est particulièrement pénalisant dans le cas de corrélations fortes puisqu'il est alors impossible de prédire des valeurs extrêmes (voir figure 5.2). Lorsque l'on considère des variables X_i symétriques, le biais des différents estimateurs est très faible. Même pour le cas de variables $\beta(2, 3)$, le biais reste négligeable. Dans le cas de variables asymétriques, ici des variables $\beta(1/2, 2/10)$, ce biais devient non négligeable, sans empêcher d'obtenir de bonnes performances. Pour finir, les mauvaises performances de Λ_{DE} dans le cas de variables $\beta(1/2, 2/10)$ anti-corrélées ($\rho = -0.7$) sont simplement dues à l'estimation plus instable.

D'une manière générale l'estimation de la loi jointe des variables latentes par l'approche mixte EM-MM semble plus robuste. C'est celle que l'on considère dans les expériences suivantes.

Cas d'une chaîne de Markov. Considérons maintenant le cas d'un vecteur $\mathbf{X}$ de structure de dépendance linéaire, c'est-à-dire que le graphe sous-jacent est une chaîne (figure 5.5). On considère ici des chaînes à 100 variables. On réalise alors 1 000 fois l'expérience de décimation et on calcule les performances L^1 des différents estimateurs. Les résultats obtenus sont représentés sur la figure 5.6. Il apparaît clairement que le choix (F, F^{-1}) est le meilleur dans tous les cas présentés. Ceci n'est que peu surprenant puisque les autres choix étaient déjà sous-optimaux lorsque les corrélations entre variables sont relativement élevées. On écarte à ce stade Λ_{DE} qui apparaît comme le moins bon choix.

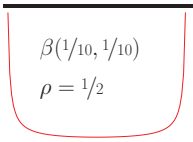
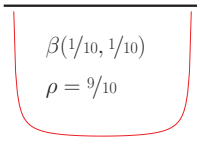
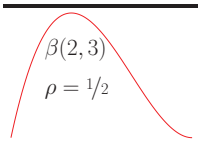
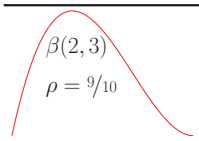
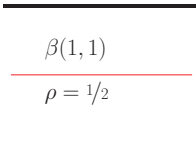
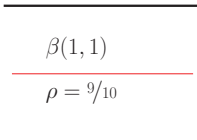
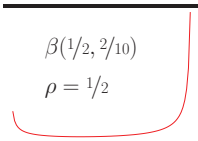
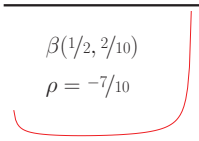
Densités	Estimation	$(F, \Gamma^{\mathcal{D}})$	$(F, \Gamma^{\mathcal{P}})$	Λ_{MI}	Λ_{DE}	Exact
 $\beta(1/10, 1/10)$ $\rho = 1/2$	MM	**0,7%** 0,07	**0,8%** −0,22	3,3% 0,01	**0,7%** 0,05	32,04 0
	EM-MM	**0,2%** −0,34	3,1% −0,79	∼ ∼	**0,7%** 0,05	∼ ∼
 $\beta(1/10, 1/10)$ $\rho = 9/10$	MM	**0,1%** 0	14,8% −0,46	9,5% −0,06	**0,1%** 0,01	13,62 0
	EM-MM	**0,1%** −0,15	22% −0,72	∼ ∼	**0,1%** 0,01	∼ ∼
 $\beta(2, 3)$ $\rho = 1/2$	MM	9,1% −0,9	3% 0,4	6,7% 0,44	10,2% −0,84	14,22 0
	EM-MM	**1,4%** −0,26	4,4% 0,47	∼ ∼	9,5% 0,15	∼ ∼
 $\beta(2, 3)$ $\rho = 9/10$	MM	**0,7%** −0,06	67,1% 1,24	61,7% 0,92	**1%** 0,02	6,89 0
	EM-MM	**1,3%** 0,1	68,8% 1,25	∼ ∼	38,9% 0,74	∼ ∼
 $\beta(1, 1)$ $\rho = 1/2$	MM	7,3% 0,04	3% 0,01	7,3% −0,03	7,7% 0,07	20,96 0
	EM-MM	**0,1%** 0,02	4,6% 0,01	∼ ∼	10,4% 0,07	∼ ∼
 $\beta(1, 1)$ $\rho = 9/10$	MM	**0,6%** 0,03	64,4% −0,09	58,6% 0	**0,8%** 0,07	9,83 0
	EM-MM	**0,4%** 0,02	66,2% −0,09	∼ ∼	24,9% 0,01	∼ ∼
 $\beta(1/2, 2/10)$ $\rho = 1/2$	MM	**2,8%** 9,93	3,3% −4,13	7,5% −4,11	7,5% 10,24	23 0
	EM-MM	**2,7%** 6,77	4,4% −5,14	∼ ∼	12,8% 1,81	∼ ∼
 $\beta(1/2, 2/10)$ $\rho = -7/10$	MM	**3,5%** 5,28	17,4% −8,96	20,8% −5,16	47,2% −3,35	18,21 0
	EM-MM	**4,1%** 5,74	16,9% −8,8	∼ ∼	47,9% −5,16	∼ ∼

TABLE 5.1 : Performances des différents prédicteurs dans le cas de la figure 5.4. La première ligne correspond à la performance L^1 du prédicteur en % d'écart à la performance optimale et la seconde à son biais. Les valeurs en gras correspondent aux choix les plus performants. Le symbole ∼ indique que la valeur ne dépend pas de la méthode d'estimation.

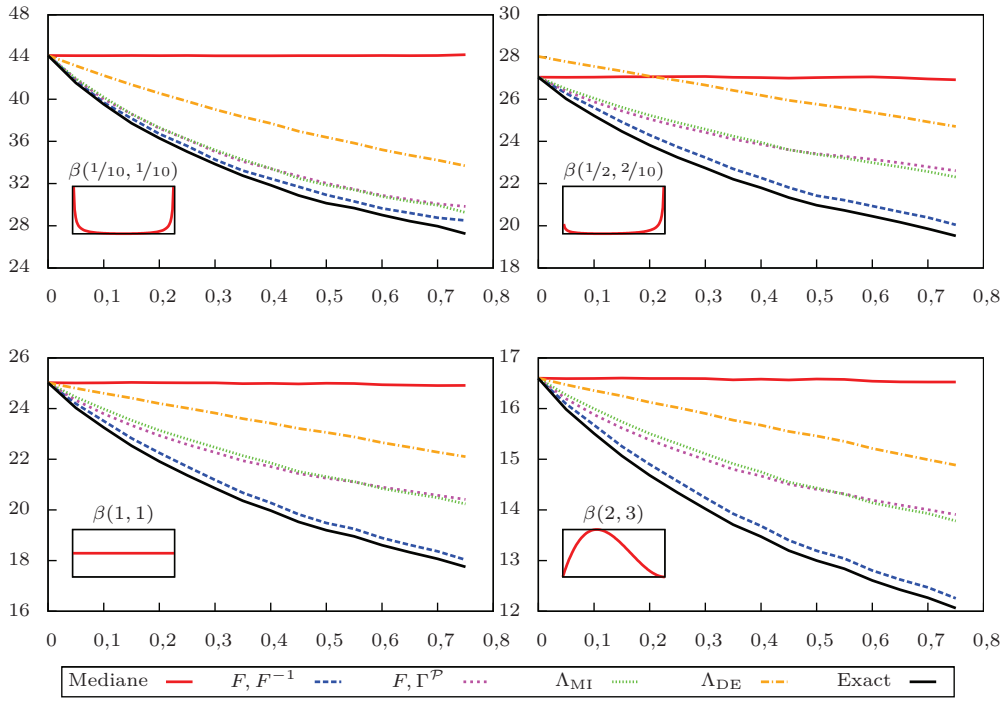


FIGURE 5.6 : Performance L^1 de prédiction des variables non observées ($\times 100$) en fonction de la proportion de variables révélées ; les petites figures incrustées représentent la densité de la variable β correspondante. À titre de comparaison, deux autres prédicteurs sont tracés : la médiane (en rouge) et le prédicteur exact (en noir), qui est obtenu par inférence sur les variables Y_i. Chaque chaîne est constituée de 100 variables.

Cas d'un arbre régulier. Pour conclure cette section, on considère le cas d'arbres n-aires, c'est-à-dire pour lesquels chaque sommet non terminal possède exactement n voisins. Par exemple pour $n = 2$ on obtient les chaînes étudiées précédemment. Pour $n = 3$ on obtient un arbre binaire. Pour pouvoir comparer avec un outil statistique classique, on utilise le prédicteur basé sur les k plus proches voisins (k-NN) [2]. Ce prédicteur est connu pour être efficace pour des données de trafic routier [86], mais sa complexité est un handicap comparé à BP pour des réseaux de grande taille. De plus, il nécessite des informations historiques exhaustives, *i.e.* des observations complètes du réseaux, qui ne sont pas accessibles dans le cas de données provenant de véhicules sondes. On réalise à nouveau l'expérience décrite dans le cas d'une chaîne. Quelques résultats sont présentés à la figure 5.7. D'une manière générale le choix (F, F^{-1}) est encore celui qui se comporte le mieux. Notons cependant que lorsque la connectivité augmente encore la situation change. En effet, à forte connectivité la convergence de mBP peut devenir très lente (voir sec-

2. *k-nearest neighbors.*

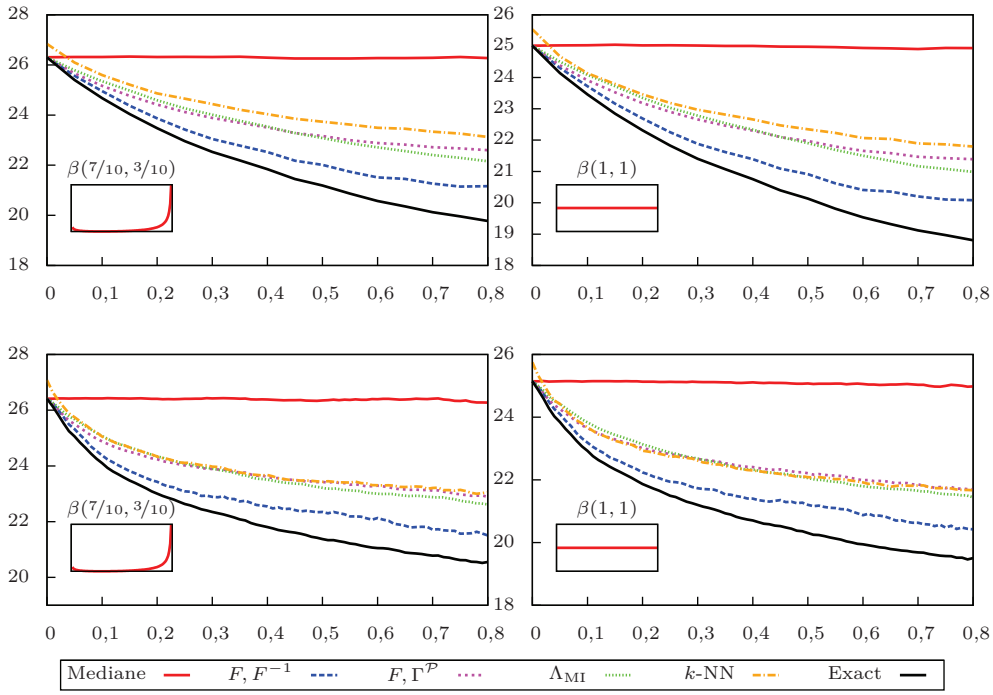


FIGURE 5.7 : Même chose que la figure 5.6. On trace de plus, à titre de comparaison, le prédicteur k-NN (en orange), approximativement optimisé au $k = 50$ plus proches voisins de l'historique utilisé pour construire le modèle. La connectivité est $n = 3$ pour les figures du haut, $n = 5$ pour les figures du bas. Chaque arbre est constitué de 100 variables.

tion 4.1.3 page 85). Les cas non convergents deviennent alors trop nombreux et expliquent les « bosses » apparaissant dans la figure 5.8. Dans ces cas là les performances de Λ_{MI} restent acceptables. En fait, le choix Λ_{MI} rend l'algorithme mBP strictement équivalent à BP avec des observations directes. Dans le cas présent cela correspond à un algorithme plus stable puisque BP converge très rapidement sur un arbre quelle que soit la connectivité de celui-ci.

5.4 Conclusion

Dans ce chapitre nous avons proposé un manière minimale de modéliser les interactions entre variables aléatoires réelles à partir d'une information constituée de

- une fonction de répartition par variable réelle ;
- une matrice de covariance incomplète.

En un certain sens il s'agit d'un équivalent en termes d'indicatrices binaires de la modélisation par un vecteur gaussien. Cette dernière sera abordée dans le chapitre suivant.

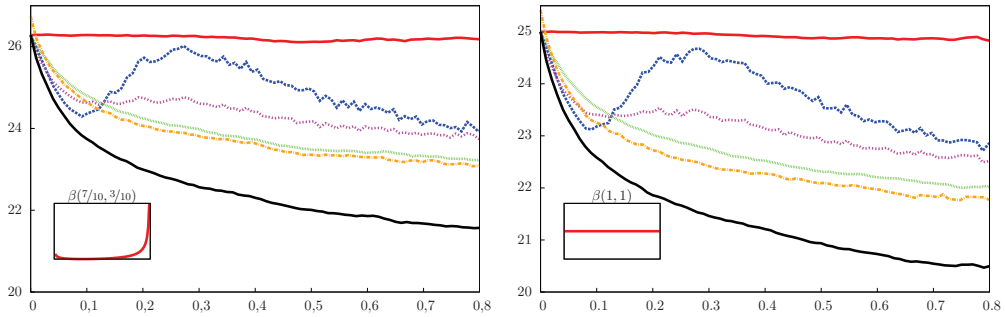


FIGURE 5.8 : Même expérience et même légende que la figure 5.6. Les arbres sont constitués de 100 variables et correspondent à une connectivité $n = 10$.

Le choix de la fonction de répartition comme fonction d'encodage et de son inverse comme fonction de décodage semble le meilleur dès lors que la connectivité du graphe reste raisonnable. Au delà d'une certaine connectivité, qui semble dépendre de la distribution des variables réelles, l'algorithme mBP devient trop peu efficace face à des corrélations fortes et le choix d'un encodage déterministe Λ_{MI} est préférable.

Une fois le choix de la fonction d'encodage arrêtée et les marginales de paires p_{ij} estimées de nombreux choix sont possibles pour définir le modèle d'Ising latent. Le choix de ce dernier dépend de la nature des données considérées et nécessite des tests sur données réelles. Cependant les résultats présentés ici permettent d'être assez optimiste pour l'application de notre méthode à des données de trafic routier, la modélisation binaire sous-jacente y étant assez naturelle.

Annexes

5.A Généralisation du critère de maximisation de l'entropie

Si l'on choisit d'utiliser la fonction de décodage $\Gamma^{\mathcal{D}}$, il paraît intéressant de construire des fonctions d'encodage Λ basées sur un principe de maximum d'entropie retournant autre chose que la médiane comme prédiction de base, *i.e.* en l'absence d'observations. Pour cela, considérons le nouveau problème variationnel suivant : étant donné un « prédicteur optimal » $\hat{\theta}(X) = x^*$ de la variable X, on se restreint aux fonctions d'encodage qui vérifient

$$\mathbb{P}(\sigma = 1) = \int_{\mathcal{X}} \Lambda(x) f(x) dx = \Lambda(x^*), \tag{5.20}$$

ou, en intégrant par parties,

$$\int_{\mathcal{X}} \left(\mathbb{1}_{\{x \leq x^*\}} + F(x) \right) \Lambda'(x) dx = 1,$$

car on a $\Lambda(x^*) = \int_{\mathcal{X}} \mathbb{1}_{\{x \leq x^*\}} \Lambda'(x) dx$. La contrainte (5.20) fournit bien des fonctions d'encodage avec une prédiction de base optimale lorsque l'on utilise $\Gamma^{\mathcal{D}}$. La proposition suivante montre que cela conduit à une fonction d'encodage explicite.

Proposition 5.6. *Supposons que X admette une densité. La fonction d'encodage $\Lambda_{\mathrm{S}}^{x^*}$ qui maximise l'entropie, relativement à la mesure uniforme, sous les contraintes* (5.2) *et* (5.20) *est*

$$\begin{cases} \Lambda_{\mathrm{S}}^{x^*}(x) = \dfrac{1}{\alpha} \log\left(\alpha F(x) + 1\right), \forall x \leq x^* \\ \Lambda_{\mathrm{S}}^{x^*}(x) = 1 + \dfrac{1}{\alpha} \log\left(\alpha(F(x) - 1) + 1\right), \forall x > x^*, \end{cases}$$

où α est l'unique solution de

$$F(x^*) = \frac{1 + e^{\alpha}(\alpha - 1)}{\alpha(e^{\alpha} - 1)}. \tag{5.21}$$

On remarque que, lorsque $\alpha \to 0$, $F(x^*) \to 1/2$ et $\Lambda_{\mathrm{S}}^{x^*} \to F$, ce qui est cohérent avec le résultat de la proposition 5.3.

Démonstration. L'entropie que l'on considère ici est

$$S(\Lambda(X)) = S(U) = -\int_0^1 h_\Lambda(u) \log h_\Lambda(u) du,$$

avec h_Λ la densité de U

$$h_\Lambda(u) = \frac{f(\Lambda^{-1}(u))}{\Lambda'(\Lambda^{-1}(u))}.$$

En effectuant le changement de variable $x = \Lambda^{-1}(u)$, l'entropie s'exprime alors

$$S(U) = -\int_{\mathcal{X}} f(x) \log \frac{f(x)}{\Lambda'(x)} dx.$$

L'ajout de la contrainte (5.20) conduit au lagrangien suivant :

$$\mathcal{L}(\Lambda', \alpha, \beta) = \int_{\mathcal{X}} f(x) \log \frac{f(x)}{\Lambda'(x)} dx + \beta \left(\int_{\mathcal{X}} \Lambda'(x) dx - 1 \right) + \alpha \left(\int_{\mathcal{X}} \left(\mathbb{1}_{\{x \leq x^*\}} + F(x) \right) \Lambda'(x) dx - 1 \right).$$

Les points stationnaires ($\frac{\delta \mathcal{L}}{\delta \Lambda'(x)} = 0$) correspondent à

$$-\frac{f(x)}{\Lambda'(x)} + \alpha \left(F(x) + \mathbb{1}_{\{x \leq x^*\}} \right) + \beta = 0,$$

et on obtient donc

$$\Lambda'(x) = \frac{f(x)}{\alpha \left(F(x) + \mathbb{1}_{\{x \leq x^*\}} \right) + \beta}.$$

La fonction d'encodage optimale Λ est finalement de la forme

$$\begin{cases} \Lambda(x) = \dfrac{1}{\alpha} \log \left(\dfrac{\alpha(F(x)+1)+\beta}{\alpha+\beta} \right), \forall x \leq x^* \\ \Lambda(x) = \Lambda(x^*) + \dfrac{1}{\alpha} \log \left(\dfrac{\alpha F(x)+\beta}{\alpha F(x^*)+\beta} \right), \forall x > x^*. \end{cases}$$

En imposant les contraintes, on obtient $\alpha + \beta = 1$, et (5.21). Quel que soit x^* l'équation (5.21) admet une unique solution. En effet la fonction g qui constitue son terme de droite est une fonction strictement croissante de α dont l'image est le segment $[0, 1]$. Pour le prouver regardons sa dérivée pour $x \neq 0$

$$g'(x) = -\frac{e^x}{e^x - 1} + \frac{1}{x^2} = \left(\frac{1}{x} - \frac{e^{x/2}}{e^x - 1} \right) \left(\frac{1}{x} + \frac{e^{x/2}}{e^x - 1} \right),$$

qui se simplifie sous la forme

$$g'(x) = \left(\frac{1}{x} - \frac{1}{2\sinh(x/2)} \right) \left(\frac{1}{x} + \frac{e^{x/2}}{e^x - 1} \right).$$

Le terme de droite est du signe de x comme somme de termes de même signe. De plus, la fonction sinh est convexe sur $[0, +\infty[$ et concave pour $]-\infty, 0]$. On a donc les inégalités suivantes $2\sinh(x/2) > x$ pour $x > 0$ et $2\sinh(x/2) < x$ pour $x < 0$ ce qui permet de conclure que g' est strictement positive sur $\mathbb{R}^*$ et donc que (5.20) admet une unique solution quelle que soit la valeur de x^*. □

5.B Généralisation du critère de minimisation de l'erreur de décodage

Le critère conduisant à Λ_{DE} n'a de sens que si l'on utilise le décodage $\Gamma^{\mathcal{D}}$. Cependant, la prédiction de base correspondante n'a aucune propriété notable et en particulier, elle n'est pas optimale pour les normes L^1 ou L^2, sauf exception. Il est alors naturel de vouloir ajouter la contrainte (5.20), comme dans le cas du critère entropique. Malheureusement, l'ajout de cette contrainte ne conduit pas à une forme explicite de la solution. Écrivons le lagrangien du nouveau problème que l'on considère

$$\mathcal{L}(\Lambda', \alpha, \gamma) = \int_{\mathcal{X}} \frac{f(x)}{\Lambda'(x)} dx + \alpha \left(\int_{\mathcal{X}} \left(\mathbb{1}_{\{x \leq x^*\}} + F(x) \right) \Lambda'(x) dx - 1 \right) + \gamma \left(\int_{\mathcal{X}} \Lambda'(x) dx - 1 \right).$$

L'équation de point stationnaire ($\frac{\partial \mathcal{L}}{\partial \Lambda'(x)} = 0$) est

$$\frac{f(x)}{\Lambda'(x)^2} = \alpha \left(\mathbb{1}_{\{x \leq x^*\}} + F(x) \right) + \gamma,$$

ce qui permet d'obtenir la forme de Λ'

$$\Lambda'(x) = \sqrt{\frac{f(x)}{\alpha \left(\mathbb{1}_{\{x \leq x^*\}} + F(x) \right) + \gamma}}.$$

À la différence la fonction Λ_{DE} décrite à la proposition 5.4, on ne peut ici exprimer Λ explicitement, avant de la calculer numériquement. En effet, la présence des multiplicateurs de Lagrange α et γ dans Λ' impose d'utiliser un schéma itératif pour leur estimation. On définit le vecteur $\mathbf{y} \stackrel{\text{def}}{=} (\alpha, \gamma)$ et la fonction $h : \mathbb{R}^2 \mapsto \mathbb{R}^2$

$$h(\mathbf{y}) \stackrel{\text{def}}{=} \begin{pmatrix} \int_{\mathcal{X}} \left(\mathbb{1}_{\{x \leq x^*\}} + F(x) \right) \Lambda'(x) dx - 1 \\ \int_{\mathcal{X}} \Lambda'(x) dx - 1 \end{pmatrix}.$$

On peut, par exemple, proposer un schéma basé sur l'algorithme de Newton multidimensionnel qui consiste, à partir de multiplicateurs de Lagrange initiaux, à résoudre le système

$$\nabla h(\mathbf{y}_n)(\mathbf{y}_{n+1} - \mathbf{y}_n) = -h(\mathbf{y}_n),$$

pour déterminer $\mathbf{y}_{n+1}$. A chaque itération, il est donc nécessaire de recalculer $h(\mathbf{y}_n)$ et $\nabla h(\mathbf{y}_n)$. Numériquement il est donc nécessaire à chaque itération de calculer 5 sommes distinctes, 3 pour $\nabla h(\mathbf{y}_n)$ et 2 pour $h(\mathbf{y}_n)$ et de résoudre

un système linéaire d'ordre 2. Le nombre de termes de ces sommes dépend directement de la discrétisation choisie pour la densité f. Ce type de schéma est cependant très instable et sa convergence très dépendante du choix du point initial. Pour finir donnons la forme explicite du gradient $\nabla h(\mathbf{y}_n)$

$$\begin{aligned}
\frac{\partial h_1}{\partial \alpha} &= \int_{\mathcal{X}} \frac{-\left(\mathbb{1}_{\{x\leq x^*\}} + F(x)\right)^2 \Lambda'(x)dx}{2\left(\alpha\left(\mathbb{1}_{\{x\leq x^*\}} + F(x)\right) + \gamma\right)} \\
\frac{\partial h_1}{\partial \gamma} &= \int_{\mathcal{X}} \frac{-\left(\mathbb{1}_{\{x\leq x^*\}} + F(x)\right) \Lambda'(x)dx}{2\left(\alpha\left(\mathbb{1}_{\{x\leq x^*\}} + F(x)\right) + \gamma\right)} \\
\frac{\partial h_2}{\partial \alpha} &= \frac{\partial h_1}{\partial \gamma} \\
\frac{\partial h_2}{\partial \gamma} &= \int_{\mathcal{X}} \frac{-\Lambda'(x)dx}{2\left(\alpha\left(\mathbb{1}_{\{x\leq x^*\}} + F(x)\right) + \gamma\right)}.
\end{aligned}$$

Chapitre 6

Modèle gaussien

On s'intéresse dans ce chapitre à la construction d'un modèle gaussien pour résoudre le problème d'inférence à partir de données incomplètes décrit au chapitre 1. L'objectif est donc de modéliser un champ markovien à valeurs réelles $\mathbf{X}$ par un vecteur gaussien. Cette modélisation est une alternative à l'approche par le modèle d'Ising développée aux chapitres 4 et 5. Les modèles d'Ising sont en effet plus à même de représenter des distributions multimodales alors qu'un modèle gaussien est par nature unimodale. Cependant le modèle gaussien possède l'avantage d'encoder naturellement des variables réelles, ce qui n'est pas le cas du modèle d'Ising. On suppose que le vecteur aléatoire $\mathbf{X}$ suit une loi centrée, *i.e.*

$$\mathbb{E}[\mathbf{X}] = \mathbf{0}.$$

Suivant les cas on supposera que les observations historiques du vecteur $\mathbf{X}$ sont de deux types :

- *complètes* : c'est-à-dire que l'on possède K observations indépendantes du vecteur $\mathbf{X}$.
- *incomplètes* : c'est-à-dire que les K observations ne concernent pas le vecteur $\mathbf{X}$ en entier. Chaque observation nous révèle seulement la valeur d'un sous-vecteur de $\mathbf{X}$.

Dans le cas d'observations complètes on peut calculer une estimation complète $\widehat{\mathbf{\Sigma}}$ de la matrice de covariance de $\mathbf{X}$. Dans le cas d'observations incomplètes cela n'est pas toujours possible. En particulier si aucun des sous-vecteurs des K observations ne contient le vecteur (X_i, X_j) nous n'aurons alors pas d'estimation de la covariance $\mathrm{cov}(X_i, X_j)$. Ce dernier cas nous intéresse particulièrement puisqu'il modélise assez bien la forme des données provenant de véhicules sondes. Dans ce cas là les seules observations disponibles sont celles de paires voisines sur le graphe physique. Ce cas de données incomplètes va donc nous fournir naturellement une estimation incomplète de la matrice de covariance de $\mathbf{X}$. Notons que cette distinction est une propriété caractéristique de la nature des données et n'est pas liée à la quantité de données disponibles.

Dans tous les cas on résumera les données disponibles sous la forme suivante

$$\left\{\mathbf{X}_{\mathbb{V}_k} = \mathbf{x}^k\right\}_{k\in\{1,\dots,\mathrm{K}\}}, \tag{6.1}$$

où $\mathbf{X}_{\mathbb{V}_k}$ représente le sous-vecteur de $\mathbf{X}$ observé.

On rappelle que l'on souhaite utiliser le modèle construit à partir des ces observations pour être capable de prédire en temps réel le comportement du vecteur $\mathbf{X}$. En particulier, on souhaite pour une observation du type $\mathbf{X}_S = \mathbf{x}$ avec S un sous-ensemble non choisi de $\mathbb{V}$ être capable de prédire le comportement des variables non observées. Supposons un instant que l'on ait construit un modèle de Gauss-Markov de $\mathbf{X}$ par une méthode qui reste à déterminer. Une manière simple de résoudre ce problème d'inférence en temps réel est d'utiliser l'algorithme Gaussian Belief Propagation (GaBP, décrit à la section 2.6 page 40). Les observations $\mathbf{X}_S = \mathbf{x}$ sont alors des observations certaines, au sens du chapitre 4. Il n'est donc pas nécessaire d'utiliser un algorithme du type mBP, il suffit d'utiliser l'algorithme GaBP sur la loi conditionnelle aux observations. On rappelle que lorsque l'algorithme GaBP converge il fournit les moyennes exactes des lois marginales du modèle.

On va proposer ici une méthode de construction d'un modèle gaussien du vecteur $\mathbf{X}$ la plus précise possible – en un sens encore à définir – et qui soit compatible avec l'algorithme GaBP. On parlera de *compatibilité avec l'algorithme GaBP* lorsque l'algorithme converge en l'absence d'observations. En pratique, si un modèle est compatible avec GaBP alors ses lois conditionnelles le sont aussi.

Nous avons supposé au chapitre 5 que le graphe $\mathcal{G} = (\mathbb{V}, \mathbb{E})$ était connu. Dans ce chapitre nous allons relâcher cette hypothèse et notre méthode estimera à la fois la structure de $\mathcal{G}$ ainsi que les paramètres du modèle. En particulier, on pourrait envisager d'utiliser les méthodes décrites dans ce chapitre dans le but obtenir une structure de dépendance pour le modèle d'Ising du chapitre précédent.

Dans le cas d'un vecteur gaussien $\mathbf{Y}$, le graphe de dépendance $\mathcal{G}$ est déterminé par la matrice de covariance inverse $\mathbf{\Sigma}^{-1}$. En effet, lorsque la matrice de covariance $\mathbf{\Sigma}$ de $\mathbf{Y}$ est régulière – c'est-à-dire inversible – on a l'équivalence suivante

$$Y_i \perp\!\!\!\perp Y_j \mid \mathbf{Y}_{\mathbb{V}\setminus\{i,j\}} \Leftrightarrow \left(\mathbf{\Sigma}^{-1}\right)_{ij} = 0.$$

Une preuve de cette équivalence est fournie par Lauritzen [61, p. 129]. Le support de la matrice $\mathbf{\Sigma}^{-1}$, qu'on appellera *matrice de précision*, contient donc toute l'information sur la structure de dépendance du champ markovien. Dans la suite, pour une matrice symétrique $\mathbf{M}$ on notera $\mathrm{Supp}(\mathbf{M})$ le graphe définit par son support, *i.e.*

$$(i,j) \in \mathrm{Supp}(\mathbf{M}) \Leftrightarrow M_{ij} \neq 0 \text{ et } i \neq j.$$

Dans le cas général du vecteur aléatoire quelconque, les coefficients de la matrice de précision s'interprètent comme l'opposé des corrélations partielles entre variables [61, p. 130].

C'est cette matrice de précision $\mathbf{\Sigma}^{-1}$ – qui est donc la paramétrisation naturelle du modèle gaussien – que l'on va chercher à estimer dans ce chapitre.

On cherche donc à construire une estimation de la matrice de précision de $\mathbf{X}$ *qui soit compatible avec l'algorithme Gaussian Belief Propagation (GaBP).* L'algorithme GaBP, tout comme BP, peut rencontré des problèmes de convergence lorsque le graphe comporte de cycles. En particulier, dans le cas de GaBP, ce sont les cycles frustrés, c'est-à-dire les cycles le long desquels le produit des corrélations est négatif, qui posent problème [68]. L'algorithme GaBP se comportera donc mieux, en terme de vitesse d'exécution et de propriétés de convergence, lorsque la structure du graphe est creuse. On parlera ici de graphe creux lorsque le nombre d'arc du graphe est proportionnel au nombre de sommets.

De nombreuses méthodes existent pour obtenir une estimation creuse de la matrice de précision $\mathbf{\Sigma}^{-1}$. Elles se basent en général sur une maximisation d'une version pénalisée de la log-vraisemblance $\mathcal{L}$, c'est-à-dire une résolution du problème de maximisation

$$\max_{\mathbf{\Sigma} \succ 0} \quad \mathcal{L}(\mathbf{\Sigma}^{-1}) - \lambda P(\mathbf{\Sigma}^{-1}),$$

où P est la fonction de pénalisation qui permet d'imposer une structure creuse à la matrice de précision. Le coefficient λ représente le compromis entre le degré de parcimonie de la matrice $\mathbf{\Sigma}^{-1}$ et son accord aux données. Les fonctions de pénalisation considérées se basent en général sur la norme L^0 ou L^1 de la matrice $\mathbf{\Sigma}^{-1}$. L'approche utilisant la norme L^1 comme pénalisation a d'abord été introduite par Tibshirani [93] dans le cadre de l'estimation par la méthodes des moindres carrés sous le terme de régression Lasso. Cette approche a ensuite été étendue au problème d'estimation d'une matrice de précision par Friedman *et al.* [30].

La norme L^0 compte tout simplement le nombre d'entrées non nulles de la matrice $\mathbf{\Sigma}^{-1}$. C'est un choix naturel de représentation de la structure de la matrice de précision. Cependant comme la norme L^0 n'est ni continue ni différentiable, la résolution exacte du problème de maximisation est un problème combinatoire NP-difficile. Seules des méthodes approchées peuvent donc être mises en œuvre. Une première classe de méthode approchée consiste à utiliser une approximation continue de la norme L^0 comme décrit par Dicker *et al.* [63].

L'approche que l'on va proposer ici est une heuristique gloutonne de résolution du problème de maximisation. On propose deux procédures itératives, basées sur l'algorithme iterative proportionnal scaling (IPS) [62], pour construire une estimation creuse de la matrice de précision. L'aspect local

de notre procédure va nous permettre d'imposer des contraintes qui garantissent une meilleure compatibilité avec l'algorithme GaBP. On a montré expérimentalement que nos méthodes sont compétitives par rapport à ces autres approches basées sur une pénalisation L_1 ou L_0 approchée [32].

On commence par introduire dans la section 6.1 la fonction objectif que l'on optimise itérativement. Dans le cas d'observations complètes du vecteur $\mathbf{X}$ cette fonction correspond simplement à la log-vraisemblance des observations. La perturbation optimale d'une paire de variables vis-à-vis de cette fonction, ainsi que son effet, sont alors très simples à calculer (section 6.1.3). Ces perturbations correspondent aux itérations de l'algorithme IPS. Dans le cas général, cette fonction objectif est définie à partir d'une mesure empirique dont la nature est décrite à la section 6.2. La section 6.3 décrit quelques méthodes simples permettant de vérifier qu'une perturbation de paire est compatible avec l'algorithme GaBP.

Deux procédures basées sur ces principes sont ensuite présentées dans la section 6.4. Ces dernières fournissent toute une famille d'estimations creuses de la matrice de précision de $\mathbf{X}$. On conclut enfin le chapitre par quelques expérimentations numériques (section 6.5).

6.1 Une fonction objectif basée sur la vraisemblance des observations

Commençons par introduire quelques notations qui nous seront utiles pour ce chapitre.

Notations. Soit i et j deux éléments de $\mathbb{V}$ et $\mathbf{M}$ une matrice carrée associée aux éléments de $\mathbb{V}$. La notation $\mathbf{M} \succ 0$ indique que la matrice $\mathbf{M}$ est définie positive. $\mathbf{M}_{\{i,j\}}$ désigne la sous-matrice suivante :

$$\mathbf{M}_{\{i,j\}} \stackrel{\text{def}}{=} \begin{bmatrix} M_{ii} & M_{ij} \\ M_{ji} & M_{jj} \end{bmatrix}.$$

Lorsque l'on combinera cette notation avec l'inversion de matrice on supposera que l'inversion est effectuée en second, *i.e.*

$$\mathbf{M}^{-1}_{\{i,j\}} = \left(\mathbf{M}_{\{i,j\}}\right)^{-1}.$$

Enfin, si $\mathbf{N}$ est une matrice de taille 2×2 associée aux sommets i et j de $\mathbb{V}$, on note $[\mathbf{N}]$ la matrice de taille $|\mathbb{V}| \times |\mathbb{V}|$ dont tous les éléments valent 0 exceptés ceux correspondant aux entrées i et j

$$\begin{cases} [\mathbf{N}]_{k\ell} = 0, \text{ si } (k,\ell) \notin \{i,j\}^2 \\ [\mathbf{N}]_{k\ell} = N_{k\ell}, \text{ sinon.} \end{cases}$$

On généralise naturellement ces notations au cas d'un sous-ensemble quelconque de $\mathbb{V}$. Par exemple, pour $\mathbb{V}' \subset \mathbb{V}$ on a

$$\begin{cases} [\mathbf{N}_{\mathbb{V}'}]_{k\ell} = 0, \text{ si } (k,\ell) \notin \mathbb{V}' \times \mathbb{V}' \\ [\mathbf{N}_{\mathbb{V}'}]_{k\ell} = N_{k\ell}, \text{ sinon.} \end{cases}$$

6.1.1 Expression exacte de la log-vraisemblance

On rappelle que les observations (6.1) de notre champ markovien gaussien $\mathbf{X}$ sont incomplètes. Comme nous allons le voir, il est de ce fait difficile de maximiser directement leur vraisemblance.

Pour exprimer la vraisemblance associée aux observations (6.1) il est tout d'abord nécessaire de connaître la densité de probabilité marginale associée à un sous-vecteur $\mathbf{X}_{\mathbb{V}_k}$. Si le vecteur $\mathbf{X}$ suit une loi normale $\mathcal{N}(\mathbf{0}, \mathbf{A}^{-1})$, avec $\mathbf{A} = \mathbf{\Sigma}^{-1}$ sa matrice de précision, la densité $f_{\mathbf{A}}$ du vecteur $\mathbf{X}$ est alors

$$f_{\mathbf{A}}(\mathbf{x}) \stackrel{\text{def}}{=} \sqrt{\frac{\det(\mathbf{A})}{(2\pi)^n}} \exp\left(-\frac{1}{2}\langle \mathbf{x}, \mathbf{A}\mathbf{x}\rangle\right). \tag{6.2}$$

Pour déterminer une loi marginale de $\mathbf{X}$ il suffit de considérer la sous-matrice de covariance correspondante. On a donc

$$\mathbf{X}_{\mathbb{V}_k} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_{\mathbb{V}_k}),$$

et la densité marginale du sous-vecteur $\mathbf{X}_{\mathbb{V}_k}$ est donc $f_{\mathbf{\Sigma}^{-1}_{\mathbb{V}_k}}$. Puisque l'on suppose que les K observations sont indépendantes on peut maintenant donner l'expression de la log-vraisemblance $\mathcal{L}_{\mathbf{A}}$ des observations (6.1).

$$\mathcal{L}_{\mathbf{A}}(\{\mathbf{x}^k\}_{k\in\{1\ldots \mathrm{K}\}}) = \frac{1}{2}\sum_{k=1}^{\mathrm{K}} \left(\log\det \mathbf{\Sigma}^{-1}_{\mathbb{V}_k} - \left\langle \mathbf{x}^k, \mathbf{\Sigma}^{-1}_{\mathbb{V}_k}\mathbf{x}^k\right\rangle\right) + \kappa, \tag{6.3}$$

où κ est une constante qui ne dépend pas de la matrice $\mathbf{A}$. La forme de la vraisemblance (6.3) est difficile à utiliser du fait de la présence des inverses de sous-matrices de $\mathbf{\Sigma}$. En particulier les termes des produits scalaires ne peuvent être regroupés par paires de variables car ils correspondent à des matrices $\mathbf{\Sigma}_{\mathbb{V}_k}$ différentes.

On rappelle que notre but est de construire un algorithme glouton considérant les perturbations de $\mathbf{A}$ associées à des paires de variables. Ceci va être trop délicat à réaliser en regardant directement la vraisemblance. Nous allons cependant nous inspirer de l'expression de la vraisemblance dans le cas d'observations complètes pour définir notre fonction objectif.

Cas d'observations complètes. Supposons pour quelques instants que pour tout k, $\mathbb{V}_k = \mathbb{V}$; c'est-à-dire que chaque observation est complète. La

log-vraisemblance de ces observations $\mathcal{L}_{\mathbf{A}}$ s'exprime alors comme

$$\mathcal{L}_{\mathbf{A}}(\{\mathbf{x}^k\}_{k\in\{1\ldots \mathrm{K}\}}) = \sum_{k=1}^{\mathrm{K}} \log f_{\mathbf{A}}(\mathbf{x}^k),$$

avec $f_{\mathbf{A}}$ la densité d'une loi normale $\mathcal{N}(\mathbf{0}, \mathbf{A}^{-1})$ donnée en (6.2). Dans ce cas, il peut être commode de ré-exprimer la log-vraisemblance $\mathcal{L}_{\mathbf{A}}$ sous la forme

$$\mathcal{L}_{\mathbf{A}}(\{\mathbf{x}^k\}_{k\in\{1\ldots \mathrm{K}\}}) = \int \log(f_{\mathbf{A}}(\mathbf{x}))\hat{\mu}^c(\mathbf{x})d\mathbf{x}, \tag{6.4}$$

où $\hat{\mu}^c$ est la mesure empirique suivante :

$$\hat{\mu}^c(\mathbf{x}) \stackrel{\text{def}}{=} \frac{1}{\mathrm{K}} \sum_{k=1}^{\mathrm{K}} \prod_{i\in\mathbb{V}} \mathbb{1}_{\{x_i = x_i^k\}}. \tag{6.5}$$

On utilisera aussi la notation $\mathcal{L}(\hat{\mu}^c, \mathbf{A})$ pour désigner la log-vraisemblance $\mathcal{L}_{\mathbf{A}}(\{\mathbf{x}^k\}_{k\in\{1\ldots \mathrm{K}\}})$. Cette notation sous-entend que toute l'information des $\mathbf{x}^k$ est incluse dans la mesure $\hat{\mu}^c$.

L'expression (6.4) de la log-vraisemblance est plus compacte mais elle est surtout plus facile à manipuler comme l'on s'en rendra compte dans la section 6.1.3. C'est cette forme qui nous donne l'intuition de la fonction définie dans la section suivante et que l'on utilise pour traiter le cas de données incomplètes.

6.1.2 Log-vraisemblance associée à une mesure empirique

On revient ici au cas de données incomplètes (6.1). L'expression exacte de la log-vraisemblance (6.3) étant trop difficile à utiliser on va définir ici une nouvelle fonction pour jouer son rôle. En s'inspirant de la forme de la log-vraisemblance (6.4) dans le cas de données complètes on propose ici de maximiser une quantité du même type. On définit celle-ci vis-à-vis d'une mesure empirique $\hat{\mu}$

$$\mathcal{L}(\hat{\mu}, \mathbf{A}) \stackrel{\text{def}}{=} \int \log(f_{\mathbf{A}}(\mathbf{x}))\hat{\mu}(\mathbf{x})d\mathbf{x}. \tag{6.6}$$

Avant de revenir sur la définition exacte de la mesure empirique $\hat{\mu}$ associée aux observations (6.1), précisons la forme de la fonction $\mathcal{L}(\hat{\mu}, \mathbf{A})$:

$$\begin{aligned}
\mathcal{L}(\hat{\mu}, \mathbf{A}) &= \int \log\left(f_{\mathbf{A}}(\mathbf{x})\right) \hat{\mu}(\mathbf{x})d\mathbf{x} \\
&= \frac{1}{2}\log\det(\mathbf{A}) - \frac{1}{2}\int \langle \mathbf{x}, \mathbf{A}\mathbf{x}\rangle \, \hat{\mu}(\mathbf{x})d\mathbf{x} + \kappa \\
&= \frac{1}{2}\log\det(\mathbf{A}) - \frac{1}{2}\mathbb{E}_{\hat{\mu}}\left[\langle \mathbf{x}, \mathbf{A}\mathbf{x}\rangle\right] + \kappa,
\end{aligned}$$

où κ est une constante qui ne dépend pas de la matrice de précision $\mathbf{A}$. Si l'on développe le produit scalaire dans l'espérance on obtient une somme de

termes de la forme « $A_{ij}x_ix_j$ ». Par linéarité de l'espérance on voit donc que la quantité $\mathbb{E}_{\hat{\mu}}\left[\langle \mathbf{x}, \mathbf{A}\mathbf{x}\rangle\right]$ et donc $\mathcal{L}(\hat{\mu}, \mathbf{A})$ ne dépendent de $\hat{\mu}$ qu'à travers sa matrice de covariance. On obtient au final

$$\mathcal{L}(\hat{\mu}, \mathbf{A}) = \frac{1}{2}\log\det(\mathbf{A}) - \frac{1}{2}\sum_{(i,j)\in\mathbb{V}^2} A_{ij}\mathbb{E}_{\hat{\mu}^{ij}}\left[X_iX_j\right] + \kappa.$$

Dans le cas où la mesure $\hat{\mu}$ est gaussienne ce résultat est évident, puisque la mesure jointe est déterminée de manière unique par sa matrice de covariance. On note dans la suite $\hat{\mathbf{\Sigma}}$ la matrice de covariance associée à la mesure $\hat{\mu}$.

Si dans le cas d'observations complètes la définition d'une mesure empirique était triviale ce n'est pas le cas ici. En effet, l'analogue de (6.5) s'écrit alors

$$\hat{\mu}^p(\mathbf{x}) = \frac{1}{\mathrm{K}}\sum_{k=1}^{\mathrm{K}}\prod_{i\in\mathbb{V}_k}\mathbb{1}_{\{x_i=x_i^k\}},$$

qui n'est pas normalisée ($\int \hat{\mu}^p(\mathbf{x})d\mathbf{x} = +\infty$) dès lors qu'il existe des observations incomplètes. Pour définir (6.6) en toute rigueur, il est donc nécessaire de faire d'avoir une version normalisée de $\hat{\mu}^p$. Pour cela on peut faire une supposition sur la distribution des variables non observées ou de manière équivalente introduire des termes de régularisation. On obtient alors

$$\hat{\mu}(\mathbf{x}) = \frac{1}{\mathrm{K}}\sum_{k=1}^{\mathrm{K}}\prod_{i\in\mathbb{V}_k}\mathbb{1}_{\{x_i=x_i^k\}}\nu(\mathbf{x}_{\mathbb{V}\setminus\mathbb{V}_k}|\mathbf{x}_{\mathbb{V}_k}).$$

La détermination exacte de cette mesure empirique jointe est une question difficile. D'un point de vue théorique la perte induite par la substitution de la version (6.6) à la place de la log-vraisemblance exacte (6.3) n'est pas quantifiée. On verra cependant à la section 6.2 qu'asymptotiquement, lorsque les données manquantes sont déterminées d'une manière indépendante du vecteur aléatoire $\mathbf{X}$, cette perte devient nulle.

Insistons encore sur le fait que l'on a pas fait d'hypothèses sur la distribution de $\mathbf{X}$. La fonction objectif que l'on va maximiser ne s'appuie que sur la matrice de covariance de $\mathbf{X}$.

6.1.3 Modification optimale d'une paire (i, j)

Supposons ici que l'on ait une estimation de la matrice de précision $\mathbf{A}$ et que l'on cherche à optimiser, au sens de $\mathcal{L}(\hat{\mu}, \cdot)$, l'interaction entre une paire de variables (X_i, X_j). On souhaite maximiser la variation de log-vraisemblance $\mathcal{L}$ en restant dans la même famille de modèle. Cela revient à chercher une nouvelle matrice $\mathbf{A}'$ dont la densité $f_{\mathbf{A}'}$ correspond à une perturbation de $f_{\mathbf{A}}$ par une fonction Ψ_{ij}. Plus précisément la nouvelle densité jointe s'exprime comme

$$f_{\mathbf{A}'}(\mathbf{x}) = f_{\mathbf{A}}(\mathbf{x})\Psi_{ij}(x_i, x_j). \tag{6.7}$$

On utilise dans la suite la notation

$$\hat{\mu}^{ij}(x_i, x_j) \stackrel{\text{def}}{=} \int \hat{\mu}(\mathbf{x}) d\mathbf{x}_{\mathbb{V}\setminus\{i,j\}},$$

qui correspond à la marginale de $\hat{\mu}$ associée à la paire (i, j). De même, on notera $f^{ij}_{\mathbf{A}}$ la marginale de $f_{\mathbf{A}}$ associée à la paire (i, j). La proposition suivante explicite la fonction Ψ_{ij} recherchée ainsi que la variation de vraisemblance associée.

Proposition 6.1. *La perturbation* (6.7) *optimale au sens de* $\mathcal{L}(\hat{\mu}, \cdot)$ *correspond à la fonction* $\hat{\Psi}_{ij}(x_i, x_j)$ *suivante,*

$$\hat{\Psi}_{ij}(x_i, x_j) = \frac{f_{\widehat{\mathbf{\Sigma}}^{-1}_{\{i,j\}}}(x_i, x_j)}{f_{\mathbf{\Sigma}^{-1}_{\{i,j\}}}}, \tag{6.8}$$

où $\mathbf{\Sigma}(\hat{\mu}^{ij})$ *est la matrice de covariance associée à la mesure empirique* $\hat{\mu}^{ij}$*. De plus, la variation* $\Delta\mathcal{L}$ *de la log-vraisemblance s'exprime comme*

$$\begin{aligned} \Delta\mathcal{L} &\stackrel{def}{=} \mathcal{L}(\hat{\mu}, \mathbf{A}') - \mathcal{L}(\hat{\mu}, \mathbf{A}) \\ &= D_{KL}(f_{\widehat{\mathbf{\Sigma}}^{-1}_{\{i,j\}}} || f_{\mathbf{\Sigma}^{-1}_{\{i,j\}}}). \end{aligned} \tag{6.9}$$

Démonstration. Calculons la log-vraisemblance associée à la densité $f_{\mathbf{A}'}$ (6.7)

$$\begin{aligned} \mathcal{L}(\hat{\mu}, \mathbf{A}') &= \int \log(f_{\mathbf{A}'}(\mathbf{x}))\hat{\mu}(\mathbf{x})d\mathbf{x} \\ &= \int \log\left(f_{\mathbf{A}}(\mathbf{x})\Psi_{ij}(x_i, x_j)\right)\hat{\mu}(\mathbf{x})d\mathbf{x} \\ &= \mathcal{L}(\hat{\mu}, \mathbf{A}) + \int \log \Psi_{ij}(x_i, x_j)\hat{\mu}^{ij}(x_i, x_j)dx_i dx_j. \end{aligned}$$

On en déduit donc la variation $\Delta\mathcal{L}$ de la log-vraisemblance

$$\begin{aligned} \Delta\mathcal{L} &= \int \log\left(\Psi_{ij}(x_i, x_j)\right)\hat{\mu}^{ij}(x_i, x_j)dx_i dx_j, \\ &= \int \log\left(\frac{\Psi_{ij}(x_i, x_j)f^{ij}_{\mathbf{A}}(x_i, x_j)\hat{\mu}^{ij}(x_i, x_j)}{f^{ij}_{\mathbf{A}}(x_i, x_j)\hat{\mu}^{ij}(x_i, x_j)}\right)\hat{\mu}^{ij}(x_i, x_j)dx_i dx_j, \\ &= \int \log\left(\frac{f^{ij}_{\mathbf{A}'}(x_i, x_j)\hat{\mu}^{ij}(x_i, x_j)}{f^{ij}_{\mathbf{A}}(x_i, x_j)\hat{\mu}^{ij}(x_i, x_j)}\right)\hat{\mu}^{ij}(x_i, x_j)dx_i dx_j, \\ &= D_{\mathrm{KL}}(\hat{\mu}^{ij} \| f^{ij}_{\mathbf{A}}) - D_{\mathrm{KL}}(\hat{\mu}^{ij} \| f^{ij}_{\mathbf{A}'}). \end{aligned}$$

Maximiser la variation $\Delta\mathcal{L}$ revient donc à trouver la meilleure approximation gaussienne de $\hat{\mu}^{ij}$ au sens de la divergence de Kullback-Leibler. Ceci est un résultat classique et on a

$$f^{ij}_{\mathbf{A}'} \sim f_{\widehat{\mathbf{\Sigma}}^{-1}_{\{i,j\}}}.$$

Pour une preuve on renvoie à Herbrich [44]. On obtient donc la forme de la perturbation optimale (6.8). Il ne reste alors plus qu'à expliciter la variation $\Delta\mathcal{L}$ correspondante

$$\Delta\mathcal{L} = \int \log\left(\frac{f_{\widehat{\mathbf{\Sigma}}^{-1}_{\{i,j\}}}(x_i, x_j)}{f^{ij}_{\mathbf{A}}(x_i, x_j)}\right)\hat{\mu}^{ij}(x_i, x_j)dx_i dx_j.$$

On rappelle que $\mathcal{L}$ est invariante pour toutes les mesures de même matrice de covariance. Pour calculer $\Delta\mathcal{L}$ on peut donc substituer $\hat{\mu}^{ij}$ par la loi $f_{\widehat{\mathbf{\Sigma}}^{-1}_{\{i,j\}}}$, on obtient alors l'expression (6.9). □

C'est cette entropie relative $D_{\mathrm{KL}}(f_{\widehat{\mathbf{\Sigma}}^{-1}_{\{i,j\}}} || f^{ij}_{\mathbf{A}})$ qui sera le critère principal qui nous permettra de trier les différentes paires de variables candidates.

Avant de continuer il est nécessaire de s'assurer que la nouvelle matrice de précision $\mathbf{A}'$ ainsi obtenue définie bien un modèle valide, c'est-à-dire qu'elle est bien définie positive. La perturbation $\hat{\Psi}_{ij}$ correspond à remplacer la matrice de précision $\mathbf{A}$ par la matrice $\mathbf{A}'$ avec

$$\mathbf{A}' = \mathbf{A} + \left[\widehat{\mathbf{\Sigma}}^{-1}_{\{i,j\}} - \mathbf{\Sigma}^{-1}_{\{i,j\}}\right] \stackrel{\text{def}}{=} \mathbf{A} + \left[\mathbf{V}^{ij}\right]. \tag{6.10}$$

La proposition suivante montre que, lorsque $\widehat{\mathbf{\Sigma}}_{\{i,j\}}$ est régulière, la matrice $\mathbf{A}'$ est en fait toujours définie positive.

Proposition 6.2. *Soit* $\mathbf{A}$ *une matrice précision et* $\mathbf{\Sigma}$ *la matrice de covariance associée. Alors la matrice* $\mathbf{A}'$*, définie selon* (6.10)*, est définie positive dès lors que* $\mathbf{\Sigma}(\hat{\mu}^{ij})$ *est une matrice symétrique définie positive.*

Les propositions 6.1 et 6.2 ne sont en fait que des cas particuliers d'un principe plus général. Elles se généralisent à des perturbations d'ordre supérieur et on obtient alors l'algorithme IPS décrit par Speed et Kiiveri dans le cas gaussien [87].

Démonstration. Considérons les matrices $\mathbf{A}'(\alpha)$ pour $\alpha \in [0, 1]$ telles que

$$\mathbf{A}'(\alpha) \stackrel{\text{def}}{=} \mathbf{A} + \alpha\left[\mathbf{V}^{ij}\right].$$

On rappelle que le spectre d'une matrice est une fonction continue de ses coefficients. Ce résultat découle de la continuité des racines d'un polynôme en tant que fonction de ses coefficients (voir par exemple Uherka et Sergott [95]). On va prouver que la fonction $\alpha \to \det(\mathbf{A}'(\alpha))$ ne s'annule pas sur $[0, 1]$ et donc que $\mathbf{A}' = \mathbf{A}'(1)$ est bien définie positive.

La matrice $\mathbf{A}$ est inversible, car définie positive, et on a donc

$$\begin{aligned}\det\left(\mathbf{A}'(\alpha)\right) &= \det\left(\mathbf{A}\right)\det\left(\mathbf{I} + \alpha\mathbf{A}^{-1}\left[\mathbf{V}^{ij}\right]\right)\\ &\stackrel{\text{def}}{=} \det\left(\mathbf{A}\right)Q(\alpha),\end{aligned}$$

avec Q un polynôme de degré deux. L'expression de Q se simplifie en utilisant le fait que $[\mathbf{V}^{ij}]$ est non nulle uniquement sur le bloc i, j. On obtient donc le déterminant d'une matrice 2×2 suivant

$$\begin{aligned} Q(\alpha) &= 1 + \alpha \operatorname{Tr}\left(\mathbf{V}^{ij}\boldsymbol{\Sigma}_{\{i,j\}}\right) + \alpha^2 \det\left(\mathbf{V}^{ij}\boldsymbol{\Sigma}_{\{i,j\}}\right) \\ &= \alpha^2 \det\left(\mathbf{V}^{ij}\boldsymbol{\Sigma}_{\{i,j\}} + \frac{1}{\alpha}\mathbf{I}\right) \\ &= \alpha^2 \det\left(\widehat{\boldsymbol{\Sigma}}^{-1}_{\{i,j\}}\boldsymbol{\Sigma}_{\{i,j\}} - \frac{\alpha-1}{\alpha}\mathbf{I}\right). \end{aligned}$$

Le polynôme Q (et donc $\det(\mathbf{A}'(\alpha))$) ne s'annule donc pas sur $[0, 1]$ si et seulement si la matrice produit $\widehat{\boldsymbol{\Sigma}}^{-1}_{\{i,j\}}\boldsymbol{\Sigma}_{\{i,j\}}$ est définie positive. On sait déjà que le produit de ses valeurs propres, c'est-à-dire son déterminant, est positif car $\widehat{\boldsymbol{\Sigma}}_{\{i,j\}}$ et $\boldsymbol{\Sigma}_{\{i,j\}}$ sont toutes deux définies positives. Regardons la somme de ses valeurs propres, c'est-à-dire sa trace :

$$\operatorname{Tr}\left(\widehat{\boldsymbol{\Sigma}}^{-1}_{\{i,j\}}\boldsymbol{\Sigma}_{\{i,j\}}\right) = \Sigma_{ii}U_{ii} + \Sigma_{jj}U_{jj} + 2\Sigma_{ij}U_{ij},$$

en notant U_{ij} les termes de $\widehat{\boldsymbol{\Sigma}}^{-1}_{\{i,j\}}$. Pour conclure remarquons que

$$\begin{aligned} \left(\frac{\Sigma_{ii}U_{ii} + \Sigma_{jj}U_{jj}}{2}\right)^2 &= \frac{U_{ii}^2\Sigma_{ii}^2 + U_{jj}^2\Sigma_{jj}^2}{4} + \Sigma_{ii}\Sigma_{jj}U_{ii}U_{jj} \\ &\geq \Sigma_{ii}\Sigma_{jj}U_{ii}U_{jj} \\ &\geq \Sigma_{ij}^2U_{ij}^2, \end{aligned}$$

car les matrices $\widehat{\boldsymbol{\Sigma}}^{-1}_{\{i,j\}}$ et $\boldsymbol{\Sigma}_{\{i,j\}}$ sont définies positives ce qui implique que

$$\Sigma_{ii}\Sigma_{jj} \geq \Sigma_{ij}^2 \qquad \text{et} \qquad U_{ii}U_{jj} \geq U_{ij}^2.$$

En outre, $\widehat{\boldsymbol{\Sigma}}_{\{i,j\}}$ et $\boldsymbol{\Sigma}_{\{i,j\}}$ sont des matrices de covariances, leurs termes diagonaux sont donc positifs. On a donc $\operatorname{Tr}(\boldsymbol{\Sigma}_{\{i,j\}}\widehat{\boldsymbol{\Sigma}}^{-1}_{\{i,j\}}) > 0$ dès lors que ni $\widehat{\boldsymbol{\Sigma}}_{\{i,j\}}$ ni $\boldsymbol{\Sigma}_{\{i,j\}}$ ne sont dégénérées. La matrice $\mathbf{A}'$ est donc définie positive. □

6.2 Spécification de la mesure empirique

Afin de pouvoir mettre en œuvre l'approche proposée ici il va être nécessaire de spécifier plus avant cette mesure empirique $\hat{\mu}$. On a vu à la section précédente que la fonction $\mathcal{L}$ que l'on optimise est invariante sur la classe des mesures empirique $\hat{\mu}$ possédant la même matrice de covariance $\widehat{\boldsymbol{\Sigma}}$.

On ne va donc pas chercher à spécifier plus précisément cette mesure empirique $\hat{\mu}$ que par sa matrice de covariance $\widehat{\boldsymbol{\Sigma}}$. On va maintenant spécifier la manière dont on construit les sous-matrices de $\widehat{\boldsymbol{\Sigma}}$ à partir d'observations incomplètes (6.1).

Pour $S \subset \mathbb{V}$ on introduit l'ensemble $\mathcal{O}_S$ des observations pour lesquelles tout le vecteur $\mathbf{X}_S$ est observé :

$$\mathcal{O}_S \stackrel{\text{def}}{=} \{k \in \{1, \dots, \mathrm{K}\} \mid S \subset \mathbb{V}_k\}.$$

Supposons que l'ensemble $\mathbb{V}_k$ soit déterminé d'une manière aléatoire indépendante des valeurs prises par le vecteur $\mathbf{X}$, hypothèse dite MCAR [1] qui est décrite, entre autres, par Little et Rubin [64]. Sous réserve que $\lim_{\mathrm{K}\to+\infty} |\mathcal{O}_S| = +\infty$, la loi forte des grands nombres nous fournit le résultat suivant

$$\widehat{\mathbf{\Sigma}}_S \stackrel{\text{def}}{=} \frac{1}{|\mathcal{O}_S|} \sum_{k \in \mathcal{O}_S} {}^t\mathbf{x}_S^k \, \mathbf{x}_S^k \xrightarrow{p.s.} \mathbf{\Sigma}_S. \tag{6.11}$$

Définissons les estimateurs de variance covariance basés sur les ensembles $\mathcal{O}_S$.

$$\hat{c}_{i,j}^S \stackrel{\text{def}}{=} \frac{1}{|\mathcal{O}_S|} \sum_{k \in \mathcal{O}_S} x_i^k x_j^k.$$

Pour les estimateurs des variances on utilise la notation $\hat{c}_i^S \stackrel{\text{def}}{=} \hat{c}_{i,i}^S$. Puisque l'on s'intéresse ici aux interactions de paires, la loi des grands nombres (6.11) implique les convergences suivantes, dès que $\lim_{\mathrm{K}\to+\infty} |\mathcal{O}_{\{i,j\}}| = +\infty$

$$\begin{cases} c_i^{\{i\}} & \xrightarrow{p.s.} \operatorname{var}(X_i), \\ c_{i,j}^{\{i,j\}} & \xrightarrow{p.s.} \operatorname{cov}(X_i, X_j), \\ c_i^{\{i,j\}} & \xrightarrow{p.s.} \operatorname{var}(X_i). \end{cases} \tag{6.12}$$

Supposons que l'on souhaite définir les marginales $\hat{\mu}^{ij}$ de la mesure empirique $\hat{\mu}$ à partir des estimateurs $\widehat{\mathbf{\Sigma}}_{\{i,j\}}$. La question naturelle qui se pose alors est : *existe-t-il une mesure empirique $\hat{\mu}$ dont les marginales vérifient les contraintes imposées ?*

Asymptotiquement la réponse est oui, du fait de la convergence (6.11). En effet tous ces estimateurs tendent vers les sous-matrices de covariances de $\mathbf{X}$, il existe donc une mesure empirique $\hat{\mu}$ dont les marginales sont $\widehat{\mathbf{\Sigma}}_{\{i,j\}}$: la loi de $\mathbf{X}$.

Cependant, lorsque le nombre d'observations K est fini la réponse est non. Le choix $\operatorname{cov}(\hat{\mu}^{ij}) = \widehat{\mathbf{\Sigma}}_{\{i,j\}}$ correspond en effet à des marginales non compatibles. On a alors

$$\mathbb{P}_{\mathbf{X}}\left(c_i^{\{i,j\}} \neq c_i^{\{i,k\}}\right) = 1,$$

dès que $\mathcal{O}_{\{i,j\}} \neq \mathcal{O}_{\{i,k\}}$. C'est-à-dire que les variances selon les différentes paires sont presque sûrement différentes. Un premier pas nécessaire à l'existence d'une mesure jointe $\hat{\mu}$ est de construire des estimateurs $\mathbf{\Sigma}_{\{i,j\}}$ compatibles. On peut alors proposer de calculer chaque quantité, variance ou covariance, avec l'intégralité des données disponibles

$$\widehat{\mathbf{\Sigma}}_{\{i,j\}}^1 \stackrel{\text{def}}{=} \begin{bmatrix} c_i^{\{i\}} & c_{i,j}^{\{i,j\}} \\ c_{i,j}^{\{i,j\}} & c_j^{\{j\}} \end{bmatrix}.$$

1. « Missing Completly At Random ».

Cet estimateur fournit bien des variances compatibles puisque celles-ci ne sont calculées qu'une seule fois. Cependant, l'estimateur $\widehat{\mathbf{\Sigma}}^1_{\{i,j\}}$ peut ne pas correspondre à une matrice définie positive comme cela a été observé par Matthai [69]. Le problème est que la corrélation $c_{i,j}^{\{i,j\}}/\sqrt{c_i^{\{i\}}c_j^{\{j\}}}$ peut prendre des valeurs hors de l'intervalle $[-1,1]$. Un estimateur classique pour remédier à ce problème consiste à calculer le coefficient de corrélation entre X_i et X_j en utilisant toutes les données disponibles puis à multiplier le résultat par la moyenne géométrique des variances. On obtient alors l'estimateur suivant

$$\widehat{\mathbf{\Sigma}}^m_{\{i,j\}} \stackrel{\text{def}}{=} \begin{bmatrix} c_i^{\{i\}} & c_{i,j}^{\{i,j\}} \dfrac{\sqrt{c_i^{\{i\}}c_j^{\{j\}}}}{\sqrt{c_i^{\{i,j\}}c_j^{\{i,j\}}}} \\ c_{i,j}^{\{i,j\}} \dfrac{\sqrt{c_i^{\{i\}}c_j^{\{j\}}}}{\sqrt{c_i^{\{i,j\}}c_j^{\{i,j\}}}} & c_j^{\{j\}} \end{bmatrix}, \tag{6.13}$$

qui correspond à une matrice presque sûrement définie positive. De plus les convergences (6.12) impliquent que

$$\frac{\sqrt{c_i^{\{i\}}c_j^{\{j\}}}}{\sqrt{c_i^{\{i,j\}}c_j^{\{i,j\}}}} \xrightarrow{p.s.} 1,$$

ce qui fournit la consistance forte $\widehat{\mathbf{\Sigma}}^m_{\{i,j\}} \xrightarrow{p.s.} \mathbf{\Sigma}_{\{i,j\}}$.

L'astuce de calcul menant à cet estimateur ne résout cependant pas tous les problèmes. Notons $\widehat{\mathbf{\Sigma}}^m$ l'estimateur de la matrice de covariance complète agrégeant les estimations $\widehat{\mathbf{\Sigma}}^m_{\{i,j\}}$ de ses sous-matrices. Cet estimateur ne fournit pas toujours une matrice définie positive comme cela est décrit par Little et Rubin [64, p. 54-55].

La question générale de l'existence d'une mesure jointe $\hat{\mu}$, gaussienne ou non, telle que

$$\forall (i,j) \in \mathbb{E}, \operatorname{cov}(\hat{\mu}^{ij}) = \widehat{\mathbf{\Sigma}}^m_{\{i,j\}}, \tag{6.14}$$

est une question difficile. Lorsque la matrice $\widehat{\mathbf{\Sigma}}^m$ est définie positive la réponse est triviale et il existe une mesure $\hat{\mu}$ correspondant à la loi $\mathcal{N}(\mathbf{0}, \widehat{\mathbf{\Sigma}}^m)$. Lorsque ce n'est pas le cas cela signifie que, si une telle mesure $\hat{\mu}$ existe, elle n'est pas gaussienne. Ce problème d'existence d'une mesure jointe est assez peu étudié dans la littérature. Citons notamment les travaux de Arnold *et al.* [6] ou de Gelman et Speed [37] mais qui s'intéressent plus à l'existence d'une mesure jointe compatible avec des lois conditionnelles.

Revenons à notre problème d'estimation itérative d'une approximation creuse de $\mathbf{\Sigma}^{-1}$, la matrice de précision de $\mathbf{X}$. Pour cela il n'est pas nécessaire de définir la mesure plus précisément que par sa matrice de covariance. Il est même possible de ne pas posséder d'estimation complète de cette matrice de covariance, certaines valeurs peuvent être manquantes. On ne pourra alors

pas envisager l'ajout d'un arc pour l'entrée manquante de la matrice de covariance. On suppose seulement dans la suite que pour les paires (i, j) telles que $|\mathcal{O}_{\{i,j\}}| \neq 0$ on a $\mathrm{cov}(\hat{\mu}^{ij}) = \widehat{\mathbf{\Sigma}}^m_{\{i,j\}}$.

Il est en fait maintenant possible de ré-exprimer le problème que l'on cherche à résoudre sous la forme suivante : *est-il possible de trouver une loi jointe de* $\mathbf{X}$ *vérifiant les contraintes* (6.14) *et qui soit compatible avec l'algorithme GaBP ?*

Si l'on oublie la compatibilité avec l'algorithme GaBP, c'est exactement le problème que cherche à résoudre l'algorithme IPS. En effet, Speed et Kiiveri [87] ont remarqué que trouver une loi jointe dont les marginales sont spécifiées est équivalent maximiser la vraisemblance lorsque le support de la matrice de précision est connu. Pour plus de détails sur cette vision on renvoie au papier de Cramer [22] et aux références qui y sont incluses. D'une certaine manière l'algorithme IPS sur lequel se base notre procédure est une manière standard de répondre à la question de l'existence d'une mesure gaussienne vérifiant les contraintes (6.14).

On ne cherche pas ici à obtenir une mesure jointe vérifiant précisément les contraintes (6.14) mais plutôt à obtenir une approximation qui soit compatible avec l'algorithme GaBP. On garde cependant à l'esprit qu'il existe des cas où aucune mesure jointe $\hat{\mu}$ sur $\mathbf{X}$ ne correspond aux contraintes (6.14).

6.3 Compatibilité avec l'algorithme GaBP

Dans cette section on va développer des méthodes qui nous permettent de vérifier qu'une perturbation associée à une paire (i, j) conduit à un modèle compatible avec l'algorithme GaBP que l'on a introduit dans la section 2.6 page 40. On rappelle tout d'abord (section 6.3.1) les résultats connus sur la convergence de cet algorithme. On montre ensuite comment il est possible d'imposer certaines contraintes spectrales à la matrice $\mathbf{A}$ dans le but d'obtenir une meilleure compatibilité avec l'algorithme GaBP.

6.3.1 Convergence de l'algorithme GaBP

Weiss et Freeman [104] ont prouvé que lorsque l'algorithme GaBP converge les lois marginales obtenues correspondent aux moyennes exactes. Les variances quant à elles sont en général fausses [68].

On sait déjà que l'algorithme converge lorsque le graphe défini par le support de la matrice de précision $\mathbf{A}$ est un arbre. Exprimons une première condition suffisante de convergence de l'algorithme.

Proposition 6.3 (Weiss et Freeman [104])**.** *Lorsque la matrice de précision* $\mathbf{A}$ *est strictement à diagonale dominante,* i.e.

$$\forall i \in \mathbb{V}, |A_{ii}| > \sum_{j \in \mathbb{V}, j \neq i} |A_{ij}|, \tag{6.15}$$

l'algorithme GaBP converge.

Cette condition est très restrictive : il est aisé de construire des arbres pour lesquels elle n'est pas vérifiée. Par exemple la matrice de précision suivante

$$\begin{bmatrix} 1 & 0{,}5 & 0 \\ 0{,}5 & 1 & 0{,}6 \\ 0 & 0{,}6 & 1 \end{bmatrix} \tag{6.16}$$

est bien définie positive (sa plus petite valeur propre vaut environ 0,22) mais sa diagonale n'est pas dominante.

La condition suffisante de convergence de GaBP la plus précise connue à ce jour a consiste à imposer que la matrice $\mathbf{A}$ soit *walk-summable* (WS). Cette condition a été introduite et développée par Malioutov *et al.* [68]. Avant de l'énoncer introduisons quelques notations commodes. On notera $\mathrm{Diag}(\mathbf{A})$ la matrice diagonale construite à partir de $\mathbf{A}$

$$(\mathrm{Diag}(\mathbf{A}))_{ij} = A_{ii}\mathbb{1}_{\{i=j\}},$$

$R(\mathbf{A}) \stackrel{\text{def}}{=} \mathbf{A} - \mathrm{Diag}(\mathbf{A})$, la matrice des coefficients hors diagonaux de $\mathbf{A}$ et $R'(\mathbf{A})$ sa version normalisée de terme général

$$R'(\mathbf{A})_{ij} = \frac{R(\mathbf{A})_{ij}}{\sqrt{R(\mathbf{A})_{ii}R(\mathbf{A})_{jj}}}.$$

Enfin, pour une matrice $\mathbf{B}$ on note $|\mathbf{B}|$ la matrice dont les coefficients sont les valeurs absolues de ceux de $\mathbf{B}$. On utilisera les deux conditions équivalentes à la WS de la proposition suivante

Proposition 6.4 (Malioutov *et al.* [68])**.** *Lorsque la matrice de précision* $\mathbf{A}$ *vérifie une des deux conditions équivalentes suivantes*

(i) la matrice $W(\mathbf{A}) \stackrel{def}{=} \mathrm{Diag}(\mathbf{A}) - |R(\mathbf{A})|$ *est définie positive ;*

(ii) le rayon spectral $\rho(|R'(\mathbf{A})|)$ *est strictement inférieur à* 1*,*

le modèle est WS et l'algorithme GaBP converge.

La notion de WS vient du fait que l'algorithme GaBP calcule à chaque itération une somme basée sur la série $\sum_k R'(\mathbf{A})^k$. La condition (ii) impose que cette série soit absolument convergente, ce qui implique la converge des sommes et de l'algorithme. La condition (i) quant à elle correspond à imposer que le modèle dont la matrice de précision est $W(\mathbf{A})$ soit valide. $W(\mathbf{A})$ correspond au même modèle de dépendance que $\mathbf{A}$ mais où l'on a rendu toutes les corrélations partielles positives. Les termes hors diagonaux d'une matrice de précision s'interprètent en effet comme l'opposé des corrélations partielles [61, p. 130].

6.3.2 Contraintes de compatibilité avec GaBP

On étudie ici l'effet d'une perturbation (6.10) de la matrice de précision $\mathbf{A}$ pour une paire (i, j) donnée. Regardons tout d'abord le cas de la walk-summability.

Supposons que la matrice $\mathbf{A}$ soit WS. D'après la proposition 6.4, en utilisant la définition (i), $\mathbf{A}'$ sera WS si et seulement si $W(\mathbf{A}')$ reste définie positive. La proposition suivante indique une condition suffisante pour que $\mathbf{A}'$ soit WS.

Proposition 6.5. *Soit* $\mathbf{A}$ *une matrice de précision WS. La matrice* $\mathbf{A}'$ *définie selon* (6.10) *est WS si*

$$\Theta(\alpha) > 0, \forall \alpha \in [0, 1], \tag{6.17}$$

avec Θ *le polynôme de degré deux par morceaux suivant*

$$\begin{aligned}\Theta(\alpha) \stackrel{def}{=} \det(W^{-1}_{\{ij\}}) \left(\alpha^2 V_{ii} V_{jj} - B(\alpha)^2\right) \\ + \alpha \left(W^{-1}_{ii} V_{ii} + W^{-1}_{jj} V_{jj}\right) + 2B(\alpha) W_{ij},\end{aligned} \tag{6.18}$$

où $B(\alpha) \stackrel{def}{=} |A_{ij}| - |\alpha V_{ij} + A_{ij}|$.

Pour vérifier la condition (6.17) il est nécessaire, dans le pire cas, de calculer les racines de deux polynômes de degré deux. Notons cependant qu'il est de plus nécessaire de connaître la matrice $W(\mathbf{A})^{-1}$. On reviendra sur ce point dans la description de l'algorithme.

Démonstration. On obtient la condition suffisante en déroulant le même raisonnement que celui de la preuve de la proposition 6.2. Pour $\alpha \in [0, 1]$ on a

$$W(\mathbf{A} + \alpha \mathbf{V}) = W(\mathbf{A}) + [\phi(\alpha \mathbf{V}, \mathbf{A})],$$

avec

$$\phi(\mathbf{V}, \mathbf{A}) \stackrel{\text{def}}{=} \begin{bmatrix} V_{ii} & |A_{ij}| - |V_{ij} + A_{ij}| \\ |A_{ij}| - |V_{ji} + A_{ji}| & V_{jj} \end{bmatrix}.$$

$W(\mathbf{A})$ étant inversible par hypothèse, le déterminant de $W(\mathbf{A}+\alpha\mathbf{V})$ s'exprime comme

$$\begin{aligned}\det(W(\mathbf{A} + \alpha \mathbf{V})) &= \det\left(W(\mathbf{A})\right) \det\left(\mathbf{I} + W(\mathbf{A})^{-1} [\phi(\alpha \mathbf{V}, \mathbf{A})]\right) \\ &= \det\left(W(\mathbf{A})\right) \Theta(\alpha),\end{aligned}$$

avec Θ définie en (6.18). Le spectre de $W(\mathbf{A}+\alpha\mathbf{V})$ étant une fonction continue de α la condition (6.17) suit. □

Comme nous le verrons dans les expérimentations numériques, imposer au modèle d'être WS est en général une contrainte trop forte. Il est en effet

aisé de trouver des modèles non WS compatibles avec l'algorithme GaBP. Le principe présenté ici nous permet en fait d'imposer des contraintes spectrales plus faibles.

On propose maintenant d'imposer que la matrice $\mathrm{diag}(\mathbf{A}) - R(\mathbf{A})$ reste définie positive. De manière équivalente cela revient à contraindre le rayon spectral $\rho(R'(\mathbf{A}))$ à être strictement inférieur à 1. D'après les travaux de Malioutov *et al.* [68] il s'agit d'une condition nécessaire à la convergence des sommes calculées par l'algorithme GaBP. On dérive alors la condition suivante

Proposition 6.6. *Soit* $\mathbf{A}$ *une matrice de précision telle que* $\rho(R'(\mathbf{A})) < 1$ *et* $\mathbf{S} \stackrel{def}{=} (\mathrm{Diag}(\mathbf{A}) - R(\mathbf{A}))^{-1}$. *La matrice* $\mathbf{A}'$ *définie selon* (6.10) *vérifie* $\rho(R'(\mathbf{A}')) < 1$ *si*

$$\Pi(\alpha) > 0, \forall \alpha \in [0,1], \tag{6.19}$$

avec Π *le polynôme de degré deux suivant*

$$\Pi(\alpha) \stackrel{def}{=} \alpha^2 \det(\mathbf{V}\mathbf{S}_{\{i,j\}}) + \alpha(V_{ii}S_{ii} + V_{jj}S_{jj} - 2V_{ij}S_{ij}) + 1. \tag{6.20}$$

Vérifier la condition (6.19) nécessite seulement de calculer les racines d'un polynôme de degré deux. Comme pour la proposition 6.5, il est cependant nécessaire de connaître une matrice inverse, ici $\mathbf{S}$.

Démonstration. Toujours selon le même raisonnement que celui de la section 6.1.3 on définit les matrices $\mathbf{M}(\alpha)$ pour $\alpha \in [0,1]$

$$\begin{aligned}\mathbf{M}(\alpha) &\stackrel{\mathrm{def}}{=} \mathrm{Diag}(\mathbf{A} + \alpha[\mathbf{V}]) - R(\mathbf{A} + \alpha[\mathbf{V}]) \\ &= \mathbf{S}^{-1} + \alpha\,\mathrm{Diag}([\mathbf{V}]) - R([\mathbf{V}]).\end{aligned}$$

On a donc

$$\begin{aligned}\det(\mathbf{M}(\alpha)) &= \det(\mathbf{S}^{-1})\det\left(\mathbf{I} + \alpha\mathbf{S}\left[\mathrm{Diag}(\mathbf{V}) - R(\mathbf{V})\right]\right) \\ &= \det(\mathbf{S}^{-1})\Pi(\alpha),\end{aligned}$$

avec Π définie en (6.20). Le spectre de $\mathbf{M}(\alpha)$ étant une fonction continue de α, la condition (6.19) suit. □

Les conditions exprimées dans les propositions 6.5 et 6.6 constituent des conditions suffisantes pour que la propriété spectrale soit conservée après la perturbation (6.10). Cependant les cas où les critères conduisent à rejeter une perturbation admissible sont peu nombreux. Il est en effet nécessaire qu'une, ou plusieurs, valeurs propres s'annulent pour $\alpha \in]0,1[$ et redeviennent positives pour $\alpha = 1$.

Pour faire référence aux paires (i,j) qui vérifient les différentes conditions que l'on vient d'énoncer on introduit les ensembles $\mathcal{E}(\eta, \mathbf{A})$ suivants

$$\begin{aligned}
&\mathcal{E}(0,\mathbf{A}) = \{(i,j) \in \mathbb{E}\}\,,\\
&\mathcal{E}(1,\mathbf{A}) = \{(i,j) \in \mathbb{E} \mid (6.17) \text{ est vérifiée}\}\,,\\
&\mathcal{E}(2,\mathbf{A}) = \{(i,j) \in \mathbb{E} \mid (6.19) \text{ est vérifiée}\}\,.
\end{aligned} \tag{6.21}$$

$\mathcal{E}(0,\mathbf{A})$ correspond à l'ensemble des paires (i,j) qui peuvent être considérées dans le cas non contraint, $\mathcal{E}(1,\mathbf{A})$ dans le cas où l'on impose la WS et $\mathcal{E}(2,\mathbf{A})$ dans le cas où l'on impose $\rho(\mathbf{R}') < 1$.

6.4 Algorithme glouton de sélection et d'estimation

Après avoir mis en place les différents outils nécessaires dans les sections précédentes on va décrire ici la procédure qui va nous permettre de construire une estimation creuse de la matrice de précision du vecteur $\mathbf{X}$. On suppose donc ici que l'on a résumé les observations incomplètes (6.1) sous la forme des marginales empiriques de paires $\hat{\mu}^{ij}$. Notons que pour certaines paires (i,j) la marginale $\hat{\mu}^{ij}$ peut ne pas avoir été observée, c'est-à-dire que $|\mathcal{O}_{\{i,j\}}| = 0$. D'une manière compatible avec le principe du maximum d'entropie de Jaynes [51], on ne modélisera pas une interaction directe pour une paire (i,j) sans observation conjointe de cette paire. On suppose donc que l'on a une estimation, *a priori* incomplète, de la matrice de covariance de $\mathbf{X}$ que l'on note $\widehat{\mathbf{\Sigma}}$ et dont les sous-matrices $\widehat{\mathbf{\Sigma}}_{\{i,j\}}$ sont celles décrites à la section 6.2.

On propose tout d'abord à la section 6.4.1 une heuristique de résolution du problème de maximisation suivant

$$\max_{\substack{\mathbf{A}\succ 0\\ |\mathrm{Supp}(\mathbf{A})|\leq K_{\mathrm{obj}}}} \mathcal{L}(\hat{\mu},\mathbf{A}). \tag{6.22}$$

Cela correspond à rechercher le meilleur modèle, au sens de $\mathcal{L}(\hat{\mu},\cdot)$, parmi les modèle de connectivité moyenne inférieure à une valeur donnée. L'algorithme 3 permet même d'obtenir une solution approchée des problèmes de maximisation (6.22) pour toutes les valeurs de $K \leq K_{\mathrm{obj}}$.

Dans la section 6.4.2 on présente une heuristique de résolution du problème de maximisation de la vraisemblance pénalisée

$$\max_{\mathbf{A}\succ 0} \quad \mathcal{L}(\hat{\mu},\mathbf{A}) - \frac{\nu}{2}|\mathrm{Supp}(\mathbf{A})|.$$

Enfin, dans la section 6.4.3 on s'intéresse brièvement au choix du point initial de ces algorithmes.

Algorithme 3 : Sélection incrémentale de K_{obj} arcs.

Données : Matrice de précision $\mathbf{A}$, matrice de covariance $\mathbf{\Sigma} = \mathbf{A}^{-1}$;
Options : $\eta \leftarrow 1$ et $W(\mathbf{A})^{-1}$ si l'on souhaite imposer la WS, $\eta \leftarrow 2$ et $(\text{Diag}(\mathbf{A}) - R(\mathbf{A}))^{-1}$ si l'on souhaite imposer $\rho(\mathbf{R}') < 1$ et $\eta \leftarrow 0$ sinon;

1 **tant que** $\Delta\mathcal{L}_{\max} > \varepsilon$ *ou* $K < K_{\text{obj}}$ **faire**
2 $\Delta\mathcal{L}_{\max} \leftarrow 0$;
3 **pour** $(i,j) \in \mathcal{E}(\eta, \mathbf{A})$ **faire**
4 calculer $\Delta\mathcal{L}^{ij}$ selon (6.25);
5 **si** $\Delta\mathcal{L}^{ij} > \Delta\mathcal{L}_{\max}$ **alors**
6 $\Delta\mathcal{L}_{\max} \leftarrow \Delta\mathcal{L}^{ij}$;
7 $(y,z) \leftarrow (i,j)$;
8 $\mathbf{V} \leftarrow [\mathbf{V}^{ij}]$ défini selon (6.24);
9 **si** $A_{yz} = 0$ **alors**
10 $K \leftarrow K + 1$;
11 $\mathbf{A} \leftarrow \mathbf{A} + \mathbf{V}$;
12 calculer $\mathbf{\Sigma} = \mathbf{A}^{-1}$ selon (6.27);
13 **si** $\eta = 1$ **alors**
14 calculer $W(\mathbf{A})^{-1}$ selon (6.26);
15 **si** $\eta = 2$ **alors**
16 calculer $(\text{Diag}(\mathbf{A}) - R(\mathbf{A}))^{-1}$ selon (6.26);
17 **si** $K = K_{\text{obj}}$ **alors**
18 $\mathbb{E} \leftarrow \text{Supp}(\mathbf{A})$ (on fige la structure du graphe);
19 $N \leftarrow 0$;
20 **tant que** $\Delta\mathcal{L}_{\max} < \varepsilon$ *et* $N < N_{\max}$ **faire**
21 répéter les étapes **2** à **16**.

6.4.1 Algorithme incrémental

Supposons que l'on ait une estimation initiale de la matrice de précision $\mathbf{A}$ de $\mathbf{X}$. L'algorithme que l'on propose ici consiste à choisir une paire (i,j) pour laquelle la mesure $\hat{\mu}^{ij}$ est disponible et à modifier le modèle selon (6.10), c'est-à-dire :

$$\mathbf{A}' = \mathbf{A} + [\mathbf{V}^{ij}], \tag{6.23}$$

avec

$$\mathbf{V}^{ij} = \widehat{\mathbf{\Sigma}}^{-1}_{\{i,j\}} - \mathbf{\Sigma}^{-1}_{\{i,j\}}, \tag{6.24}$$

où $\mathbf{\Sigma} = \mathbf{A}^{-1}$ est la matrice de covariance associée à $\mathbf{A}$. Cet algorithme est classiquement utilisée pour l'estimation d'un modèle gaussien lorsque le support de $\mathbf{A}$ est connu. Il s'agit d'un cas particulier de l'algorithme « Iterative Pro-

portional Scaling » (IPS) décrit par Lauritzen [61]. L'originalité de l'approche proposée ici est d'estimer le support de $\mathbf{A}$ grâce à cet algorithme.

La première question qui se pose est celle du choix de la paire (i, j) devant être considérée. D'après la section 6.1.3, pour une paire (i, j) fixée, le gain $\Delta\mathcal{L}^{ij}$ obtenu par la perturbation (6.23) est exactement $D_{\mathrm{KL}}(\hat{\mu}^{ij}||f^{ij}_{\mathbf{A}})$, qui s'exprime ici comme

$$\Delta\mathcal{L}^{ij} = \frac{1}{2}\left(\mathrm{Tr}\left(\widehat{\mathbf{\Sigma}}_{\{i,j\}}\mathbf{\Sigma}^{-1}_{\{i,j\}}\right) - 2 - \log\frac{\det\left(\widehat{\mathbf{\Sigma}}_{\{i,j\}}\right)}{\det\left(\mathbf{\Sigma}_{\{i,j\}}\right)}\right). \tag{6.25}$$

C'est donc tout naturellement le critère que l'on considère ici. À chaque itération on modifie la paire (i, j) pour laquelle la variation $\Delta\mathcal{L}^{ij}$ est la plus grande.

Après cette modification il est nécessaire de calculer la nouvelle matrice de covariance $\mathbf{\Sigma}' = (\mathbf{A}')^{-1}$ pour pouvoir itérer la procédure. La perturbation de la matrice $\mathbf{A}$ correspondant à une matrice de taille 2×2, son inverse peut donc être facilement calculé. On a en effet

$$\begin{aligned}(\mathbf{A} + [\mathbf{V}])^{-1} &= \mathbf{A}^{-1} - \mathbf{A}^{-1}[\mathbf{V}](\mathbf{I} + \mathbf{A}^{-1}[\mathbf{V}])^{-1}\mathbf{A}^{-1} \\ &= \mathbf{A}^{-1} - \mathbf{A}^{-1}[\mathbf{V}](\mathbf{I} + [\mathbf{\Sigma}_{\{i,j\}}][\mathbf{V}])^{-1}\mathbf{A}^{-1},\end{aligned} \tag{6.26}$$

valide pour une matrice $\mathbf{V} = \mathbf{V}_{\{i,j\}}$de taille 2×2. On obtient donc la nouvelle matrice de covariance comme

$$\mathbf{\Sigma} = (\mathbf{A}')^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\left[\mathbf{\Sigma}^{-1}_{\{i,j\}}\right]\left(\mathbf{I} - \left[\widehat{\mathbf{\Sigma}}_{\{i,j\}}\right]\left[\mathbf{\Sigma}^{-1}_{\{i,j\}}\right]\right)\mathbf{A}^{-1}. \tag{6.27}$$

La complexité de la mise à jour de la matrice de covariance selon (6.27) est donc en $\mathcal{O}(N^2)$. Notons que ce type de mise à jour de la matrice inverse se généralise pour des perturbations d'ordres plus élevées (Chang [16]).

Pour obtenir un modèle compatible avec l'algorithme GaBP on peut souhaiter imposer les contraintes spectrales décrites aux propositions 6.5 et 6.6. On rappelle que la contrainte (6.17) est une condition suffisante de convergence de l'algorithme GaBP, alors que la contrainte (6.19) est nécessaire à la convergence mais n'offre pas de garantie. Dans ces deux cas il est nécessaire de connaître à chaque itération de l'algorithme une des deux matrices inverses suivantes : $W(\mathbf{A})^{-1}$ ou $(\mathrm{Diag}(\mathbf{A}) - R(\mathbf{A}))^{-1}$.

Lorsque l'on perturbe le modèle selon (6.23) les deux matrices $W(\mathbf{A})$ et $\mathrm{Diag}(\mathbf{A}) - R(\mathbf{A})$ sont elles aussi perturbées par l'ajout d'une matrice 2×2. Il est donc là aussi possible de mettre à jour leurs inverses en $\mathcal{O}(N^2)$. Pour cela on applique la formule (6.26) à la matrice concernée.

L'algorithme 3 page ci-contre est une implémentation formelle de notre procédure.

6.4.2 Algorithme avec ajouts et suppressions

On présente ici une variante de l'algorithme 3 pour lequel il est possible de choisir de supprimer certains liens. Supprimer un lien (i, j) consiste à rendre nuls les coefficients A_{ij} et A_{ji} de la matrice de précision $\mathbf{A}$. On sait, d'après la section 6.1.3, qu'il ne s'agit pas de la perturbation optimale du modèle. Pour que l'algorithme sélectionne parfois des liens à supprimer on va modifier la fonction objectif. Au lieu de cherche à maximiser itérativement $\mathcal{L}(\hat{\mu}, \mathbf{A})$ on va considérer une version pénalisée $\mathcal{L}^{\nu}(\hat{\mu}, \mathbf{A})$. On cherche donc à résoudre le problème d'optimisation suivant :

$$\max_{\mathbf{A} \succ 0} \mathcal{L}^{\nu}(\hat{\mu}, \mathbf{A}), \tag{6.28}$$

avec

$$\mathcal{L}^{\nu}(\hat{\mu}, \mathbf{A}) \stackrel{\text{def}}{=} \mathcal{L}(\hat{\mu}, \mathbf{A}) - \frac{\nu}{2}|\text{Supp}(\mathbf{A})|. \tag{6.29}$$

Le coefficient ν représente la pénalisation imposée à l'ajout d'un coefficient non nul hors de la diagonale de $\mathbf{A}$. La résolution exacte de ce problème est NP-difficile comme le montrent Karger et Srebro [54]. On va là aussi proposer une heuristique gloutonne de maximisation de cette fonction. Notons que ce principe de suppression d'arc permet de mettre en place une compensation à la non commutativité de l'ajout des liens. En effet, après un certain nombre d'itération un lien ajouté peut voir son importance décroître. Il peut alors être intéressant de le voir céder sa place à un autre lien.

Commençons par décrire précisément ce que l'on entend par « supprimer l'arc (i, j) ». Il s'agit de perturber la matrice de précision $\mathbf{A}$ par la matrice $[\mathbf{V}]$ définie ci-dessous

$$\mathbf{V} = \begin{bmatrix} 0 & -A_{ij} \\ -A_{ji} & 0 \end{bmatrix}. \tag{6.30}$$

La nouvelle matrice de précision est alors $\mathbf{A}' = \mathbf{A} + [\mathbf{V}]$.

Avant de passer à la description de l'algorithme on va exprimer des conditions sous lesquelles la suppression d'un arc conserve les propriétés du modèle. On commence par regarder sous quelles conditions la matrice $\mathbf{A}'$ reste définie positive. C'est l'objet de la proposition suivante

Proposition 6.7. *Une condition suffisante pour que la suppression d'un arc $(i, j) \in \mathbb{V}^2$ conduise à un modèle valide est que*

$$\frac{1}{\Sigma_{ij} - \sqrt{\Sigma_{ii}\Sigma_{jj}}} < A_{ij} < \frac{1}{\Sigma_{ij} + \sqrt{\Sigma_{ii}\Sigma_{jj}}}. \tag{6.31}$$

Démonstration. Pour $\alpha \in [0, 1]$ on a

$$\begin{aligned} \det\left(\mathbf{A} + \alpha[\mathbf{V}]\right) &= \det(\mathbf{A}) \det(\mathbf{I} + \alpha\mathbf{A}^{-1}[\mathbf{V}]) \\ &\stackrel{\text{def}}{=} \det(\mathbf{A}) P(\alpha), \end{aligned}$$

avec P le polynôme de degré deux suivant

$$\begin{aligned} P(\alpha) &= \det\left(\mathbf{I} + \alpha \mathbf{\Sigma}_{\{i,j\}} \mathbf{V}\right), \\ &= (1 - \alpha A_{ij}\Sigma_{ij})^2 - \alpha^2 A_{ij}^2 \Sigma_{ii}\Sigma_{jj}, \\ &= \left(1 - \alpha A_{ij}\left(\Sigma_{ij} - \sqrt{\Sigma_{ii}\Sigma_{jj}}\right)\right)\left(1 - \alpha A_{ij}\left(\Sigma_{ij} + \sqrt{\Sigma_{ii}\Sigma_{jj}}\right)\right). \end{aligned}$$

Lorsque P ne s'annule pas sur $[0,1]$, la matrice $\mathbf{A}' = \mathbf{A} + [\mathbf{V}]$ est définie positive. Ceci nous conduit à la condition (6.31). □

Supposons maintenant que la matrice $\mathbf{A}$ soit WS, la situation est alors encore plus simple puisque la suppression d'un arc (i,j) conduit à un modèle lui aussi WS

Proposition 6.8. *Soit* $\mathbf{A}$ *une matrice de précision WS. La suppression d'une paire* $(i,j) \in \mathbb{V}^2$ *conduit à une matrice de précision elle aussi WS.*

Démonstration. La suppression d'un arc (i,j) conduit à une matrice $\mathbf{A}'$ telle que $|R'(\mathbf{A}')| \leq |R'(\mathbf{A})|$. L'opérateur $\leq$ correspond à la comparaison terme à terme. On a alors $\rho(|R'(\mathbf{A}')|) \leq \rho(|R'(\mathbf{A})|) < 1$ (voir Seneta [85, p. 22]). Le modèle définit par $\mathbf{A}'$ est donc WS. □

Le cas de la conservation de la condition $\rho(R'(\mathbf{A})) < 1$ a déjà été traité. Il suffit de vérifier la condition (6.19) de la proposition 6.6 pour la matrice $\mathbf{V}$ définie en (6.30).

Pour faire référence aux paires (i,j) qui respectent les différentes conditions que l'on vient d'énoncer et que l'on peut donc envisager supprimer on introduit les ensembles $\mathcal{E}_S(\eta, \mathbf{A})$ suivants

$$\begin{aligned} \mathcal{E}_S(0,\mathbf{A}) &= \{(i,j) \in \mathrm{Supp}(\mathbf{A}) \mid (6.31) \text{ est vérifiée}\}, \\ \mathcal{E}_S(1,\mathbf{A}) &= \{(i,j) \in \mathrm{Supp}(\mathbf{A})\}, \\ \mathcal{E}_S(2,\mathbf{A}) &= \{(i,j) \in \mathrm{Supp}(\mathbf{A}) \mid (6.19) \text{ est vérifiée}\}. \end{aligned} \tag{6.32}$$

$\mathcal{E}_S(0,\mathbf{A})$ correspond à l'ensemble des paires (i,j) qui peuvent être supprimées dans le cas non contraint, $\mathcal{E}_S(1,\mathbf{A})$ dans le cas où l'on impose la WS et $\mathcal{E}_S(2,\mathbf{A})$ dans le cas où l'on impose $\rho(\mathbf{R}') < 1$.

Revenons maintenant à notre procédure d'ajout/suppression de liens. Pour une matrice de précision $\mathbf{A}$ donnée on étudie la perturbation d'une paire qui maximise la variation de $\mathcal{L}^\nu(\hat{\mu}, \mathbf{A})$. Deux cas de figure apparaissent suivant qu'il existe un arc (i,j), c'est-à-dire $A_{ij} \neq 0$, où non.

Arc (i,j) déjà existant, *i.e.* $A_{ij} \neq 0$. On est alors confronté à deux choix possibles :

(i) supprimer l'arc (i,j) ;

(ii) optimiser la paire (i, j) selon (6.23).

Le choix (i) correspond à définir une nouvelle matrice $\mathbf{A}' = \mathbf{A} + [\mathbf{V}]$ avec $\mathbf{V}$ définie selon (6.30). On envisagera ce choix seulement lorsque la condition (6.31) est vérifiée (voir proposition 6.7). Notons que si $\mathbf{A}$ est WS aucune condition n'est imposée. Calculons maintenant la variation $\Delta\mathcal{L}$ associée à cette perturbation de $\mathbf{A}$. On a

$$\begin{aligned}\Delta\mathcal{L} &= \mathcal{L}(\hat{\mu}, \mathbf{A}') - \mathcal{L}(\hat{\mu}, \mathbf{A}) \\ &= \int \hat{\mu}^{ij}(\mathbf{x}) \log \chi_{ij}(x_i, x_j) dx_i dx_j - \log Z_\chi,\end{aligned}$$

avec $\chi_{ij}(x_i, x_j) \stackrel{\text{def}}{=} \exp(A_{ij} x_i x_j)$ et $Z_\chi \stackrel{\text{def}}{=} \int f^{ij}_{\mathbf{A}}(x_i, x_j)\chi_{ij}(x_i, x_j) dx_i dx_j$. On obtient donc

$$\Delta\mathcal{L} = A_{ij}\widehat{\Sigma}_{ij} - \frac{1}{2}\log\left(\det\left(\mathbf{I} + \mathbf{\Sigma}_{\{i,j\}}\mathbf{V}\right)\right). \tag{6.33}$$

La variation de la fonction objectif est tout simplement $\Delta\mathcal{L}^\nu = \Delta\mathcal{L} + \nu$.

Le choix (ii) mène à une variation $\Delta\mathcal{L}^\nu = \Delta\mathcal{L}$ car la connectivité du graphe ne change pas. On a donc $\Delta\mathcal{L}^\nu$ qui est donnée en (6.25).

Arc (i, j) non existant, *i.e.* $A_{ij} = 0$. On est alors confronté à deux choix possibles :

(i) ne rien faire ;

(ii) optimiser la paire (i, j) selon (6.23).

Le choix (ii) a en fait déjà été traité. La variation $\Delta\mathcal{L}$ de la log-vraisemblance associée est donnée par (6.25). On obtient donc $\Delta\mathcal{L}^\nu = \Delta\mathcal{L} - \nu$ puisque la connectivité du graphe augmente d'une unité. En fait on privilégiera le choix (i) dès lors que pour (ii) on a $\Delta\mathcal{L}^\nu < 0$.

L'algorithme 4 page suivante est une implémentation formelle de notre procédure de sélection et d'estimation par ajout et suppression.

6.4.3 Choix de la matrice initiale

Les algorithmes 3 et 4 supposent tous deux que l'on choisisse une matrice de précision $\mathbf{A}$ initiale vérifiant les contraintes que l'on souhaite imposer. On propose ici deux choix qui conviennent :

(i) le cas indépendant, *i.e.* $\mathbf{A} = \left(\text{Diag}\left(\widehat{\mathbf{\Sigma}}\right)\right)^{-1}$;

(ii) le cas où le support de la matrice $\mathbf{A}$ forme la matrice d'adjacence d'un arbre.

En effet le cas (i) correspond trivialement à une matrice définie positive, WS et telle que $\rho(R'(\mathbf{A})) = 0 < 1$. Pour le cas (ii) Malioutov *et al.* [68] ont prouvé que les conditions WS et $\rho(R'(\mathbf{A})) < 1$ sont équivalentes au fait que $\mathbf{A}$ soit définie positive.

Le cas (i) possède de nombreux avantages. Il est notamment aisé de calculer les matrices inverses nécessaires, puisqu'elles sont toutes des matrices diagonales.

Algorithme 4 : Heuristique de maximisation de la vraisemblance avec pénalisation L^0.

Données : Matrice de précision $\mathbf{A}$, matrice de covariance $\mathbf{\Sigma} = \mathbf{A}^{-1}$;
Options : $\eta \leftarrow 1$ et $W(\mathbf{A})^{-1}$ si l'on souhaite imposer (6.17), $\eta \leftarrow 2$ et $(\mathrm{Diag}(\mathbf{A}) - R(\mathbf{A}))^{-1}$ si l'on souhaite imposer (6.19) ou $\eta \leftarrow 0$ sinon;

tant que $\Delta\mathcal{L}^{\nu}_{\max} > \varepsilon$ **faire**
  $\Delta\mathcal{L}_{\max} \leftarrow 0$;
  pour $(i,j) \in \mathbb{E}$ **faire**
    si $A_{ij} = 0$ *et* $(i,j) \in \mathcal{E}(\eta, \mathbf{A})$ **alors**
      calculer $\Delta\mathcal{L}^{ij}$ selon (6.25);
      si $\Delta\mathcal{L}^{ij} - \nu > \Delta\mathcal{L}^{\nu}_{\max}$ **alors**
        $\Delta\mathcal{L}^{\nu}_{\max} \leftarrow \Delta\mathcal{L}^{ij} - \nu$;
        $(y,z) \leftarrow (i,j)$;
        $\mathbf{V} \leftarrow [\mathbf{V}^{ij}]$ défini selon (6.24);
    sinon
      si $(i,j) \in \mathcal{E}_S(\eta, \mathbf{A})$ **alors**
        calculer $\Delta\mathcal{L}^{ij}_{\mathrm{suppr}}$ selon (6.33);
        si $\Delta\mathcal{L}^{ij}_{\mathrm{suppr}} + \nu > \Delta\mathcal{L}^{\nu}_{\max}$ **alors**
          $\Delta\mathcal{L}^{\nu}_{\max} \leftarrow \Delta\mathcal{L}^{ij}_{\mathrm{suppr}} + \nu$;
          $(y,z) \leftarrow (i,j)$;
          $\mathbf{V} \leftarrow [\mathbf{V}^{ij}]$ défini selon (6.30);
      si $(i,j) \in \mathcal{E}(\eta, \mathbf{A})$ **alors**
        calculer $\Delta\mathcal{L}^{ij}_{\mathrm{modif}}$ selon (6.25);
        si $\Delta\mathcal{L}^{ij}_{\mathrm{modif}} > \Delta\mathcal{L}^{\nu}_{\max}$ **alors**
          $\Delta\mathcal{L}^{\nu}_{\max} \leftarrow \Delta\mathcal{L}^{ij}_{\mathrm{modif}}$;
          $(y,z) \leftarrow (i,j)$;
          $\mathbf{V} \leftarrow [\mathbf{V}^{ij}]$ défini selon (6.30);
  $\mathbf{A} \leftarrow \mathbf{A} + \mathbf{V}$;
  calculer $\mathbf{\Sigma} = \mathbf{A}^{-1}$ selon (6.26);
  si $\eta = 1$ **alors**
    calculer $W(\mathbf{A})^{-1}$ selon (6.26);
  si $\eta = 2$ **alors**
    calculer $(\mathrm{Diag}(\mathbf{A}) - R(\mathbf{A}))^{-1}$ selon (6.26);

Dans le cas (ii) le choix optimal est l'arbre couvrant maximal au sens de l'information mutuelle entre variables. C'est un résultat classique dû à Chow *et al.* [19]. Il est nécessaire dans ce cas de réaliser l'inversion de la matrice de précision $\mathbf{A}$, ainsi qu'éventuellement d'une autre matrice si l'on souhaite imposer la WS ou $\rho(R'(\mathbf{A})) < 1$. Dans tous les cas il s'agit d'inverser une matrice dont le support définit un arbre. Ceci peut être fait efficacement en $\mathcal{O}(N)$ où N est le nombre de variables de l'arbre.

L'algorithme 4 permet d'envisager de partir d'un graphe quelconque. On peut par exemple partir d'un graphe très connecté et supprimer de nombreux arcs. Il est cependant nécessaire d'obtenir les matrices inverses de matrices qui ne sont alors plus des matrices creuses. Ceci peut être très coûteux en temps de calcul.

Dans les expérimentations numériques on utilisera le choix (i).

6.5 Expérimentations numériques

On propose, comme au chapitre 5, de mener des expériences sur des données synthétiques dans le but de pouvoir évaluer simplement les performances de notre approche.

On suppose que nos observations historiques du vecteur $\mathbf{X}$ sont résumées dans une matrice de covariance $\widehat{\mathbf{\Sigma}}$ calculée de la manière indiquée à la section 6.2. Cette matrice sera complète ou incomplète suivant les cas. On teste ici notre algorithme incrémental, décrit dans la section 6.4.1, pour construire une estimation creuse de la matrice de précision du vecteur $\mathbf{X}$. Du fait de son aspect incrémental cet algorithme va en fait nous permettre de générer toute une famille $\{\mathbf{A}(\kappa)\}_\kappa$ de matrices de précision basée sur des graphes de connectivité moyenne κ croissante. On représentera le comportement de l'algorithme par la qualité des estimations, mesurée par la log-vraisemblance $\mathcal{L}(\mathbf{A}(\kappa))$, en fonction de la connectivité moyenne κ du support de la matrice de précision. On s'intéresse particulièrement à trouver la matrice de précision compatible avec l'algorithme GaBP qui soit de meilleure qualité. On rappelle qu'on parle de compatibilité lorsque l'algorithme GaBP converge sur la loi (6.2) définie par la matrice de précision. En pratique il existe une valeur limite κ_{compat} de connectivité moyenne telle que, pour que $\kappa \leq \kappa_{\text{compat}}$ les matrices $\mathbf{A}(\kappa)$ sont compatibles avec l'algorithme GaBP et ne le sont plus pour $\kappa > \kappa_{\text{compat}}$. Une fois la famille de matrices de précision $\{\mathbf{A}(\kappa)\}_\kappa$ générée on peut donc calculer cette valeur limite par simple dichotomie en testant la convergence de GaBP.

Cet algorithme nous permet de plus d'imposer certaines contraintes spectrales à la famille de matrices de précision comme on l'a décrit à la section 6.3. L'objectif de ces contraintes est d'augmenter la compatibilité avec l'algorithme GaBP, c'est-à-dire d'accroître la valeur limite κ_{compat}. On générera ici différentes familles $\{\mathbf{A}(\kappa)\}_\kappa$ de matrices de précision à l'aide de l'algorithme 3 correspondant à

- $\eta = 0$: le cas sans contraintes,
- $\eta = 1$: le cas où $\mathbf{A}(\kappa)$ doit être walk-summable (WS),
- $\eta = 2$: le cas où $\mathbf{A}(\kappa)$ doit être telle que $\rho(\mathbf{R}'(\mathbf{A}(\kappa))) < 1$.

On rappelle que la contrainte WS impose que toute la famille soit compatible avec l'algorithme GaBP. La contrainte $\rho(\mathbf{R}'(\mathbf{A}(\kappa))) < 1$ est nécessaire à la compatibilité avec l'algorithme GaBP mais n'est pas une condition suffisante à celle-ci. Dans le but de vérifier la pertinence de ces contraintes on considère deux familles supplémentaires de matrices de précision, elles aussi basées sur l'algorithme 3, obtenues en imposant des contraintes plus simples

- le graphe définit par le support de $\mathbf{A}(\kappa)$ ne comporte pas de boucles de taille trois ($|\mathcal{C}| > 3$).
- le graphe définit par le support de $\mathbf{A}(\kappa)$ ne comporte pas de boucles frustrées de taille trois ($|\mathcal{C}_f| > 3$).

On rappelle que une boucle est dite frustrée lorsque le produit des corrélations le long de celle-ci est négatif. Ces deux contraintes sont basées sur les constations empiriques indiquant que GaBP, et plus généralement BP, rencontre des problèmes à gérer les boucles courtes et frustrées.

6.5.1 Cas d'observations complètes

On étudie tout d'abord les performances de nos procédures dans le cas d'observations complètes. On possède alors une estimation de toutes les corrélations entre paires de variables. Notre estimation de la matrice de covariance de $\mathbf{X}$ est donc complète. On génère ici un modèle selon la méthode décrite ci-dessous avec un vecteur $\mathbf{X}$ de taille 100 et dont le graphe de dépendance a une connectivité moyenne $\kappa_0 = 10$.

Génération du modèle. Pour une connectivité moyenne κ_0, on propose de générer le modèle synthétique d'une manière assez similaire à celle du chapitre 5, c'est-à-dire que

1. On génère le support du graphe de manière aléatoire en sélectionnant le nombre d'arcs adéquat.
2. Pour les arcs choisis on tire une corrélation partielle uniforme sur l'intervalle $[-1, 1]$. Celles-ci sont résumées dans la matrice $\mathbf{R}_0$.
3. On augmente α tant que la matrice $\alpha\mathbf{I} - \mathbf{R}_0$ n'est pas définie positive. Cela revient à décroître toutes les corrélations jusqu'à obtenir un modèle valide.

On possède alors un vecteur gaussien $\mathbf{X}$ de loi $\mathcal{N}(\mathbf{0}, \alpha\mathbf{I} - \mathbf{R}_0)$.

Résultats. Les résultats de l'algorithme 3 sont représentés à la figure 6.1. On voit clairement qu'imposer la contrainte WS est une condition beaucoup trop forte. En particulier la courbe WS ne fournit pas de meilleur modèle que le meilleur modèle compatible avec GaBP obtenu sans imposer de contraintes.

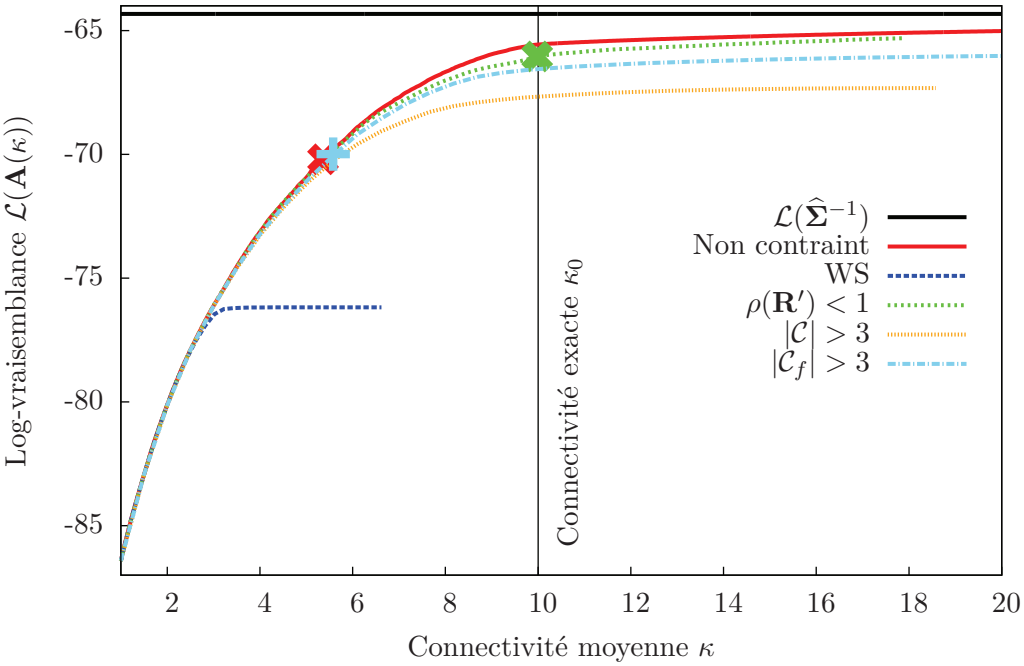


FIGURE 6.1 : Log-vraisemblance en fonction de la connectivité du graphe défini par le support de $\mathbf{A}$ pour différents algorithmes incrémentaux. Les croix représentent la limite de compatibilité avec l'algorithme GaBP des différents estimateurs. Le vrai graphe comporte 100 variables et a une connectivité moyenne de 10. $\mathcal{L}(\widehat{\mathbf{\Sigma}}^{-1})$ est la valeur maximale de la vraisemblance.

Dans le cas où il n'est pas possible de tester expérimentalement la compatibilité avec GaBP, cela permet quand même d'obtenir un modèle compatible mais à un coût important en terme de qualité d'estimation. avec l'algorithme GaBP. Ici il est possible d'obtenir un modèle compatible d'une connectivité comparable à la vraie connectivité du modèle sous-jacent.

Regardons maintenant le cas où l'on ne crée jamais de boucles de taille trois ($|\mathcal{C}| < 3$). Dans cet exemple cela fournit des modèles compatibles avec l'algorithme GaBP quelle que soit leur connectivité moyenne. Cependant on obtient des modèles de qualité moindre que ceux obtenus par la condition $\rho(\mathbf{R}'(\mathbf{A})) < 1$. Enfin, la contrainte imposant de ne pas avoir de boucles frustrées de taille trois ($|\mathcal{C}_f| > 3$) est trop lâche. Elle ne permet pas d'éliminer certains choix incompatibles avec GaBP et sa compatibilité est ici équivalente à celle du cas non contraint ($\eta = 0$). De plus la vraisemblance maximale associée à cette famille d'estimations est inférieure à celle du modèle compatible limite correspondant à $\rho(\mathbf{R}'(\mathbf{A})) < 1$.

Pour plus d'informations sur le comportement de nos procédures dans le cas d'observations complètes on réfère le lecteur à [32].

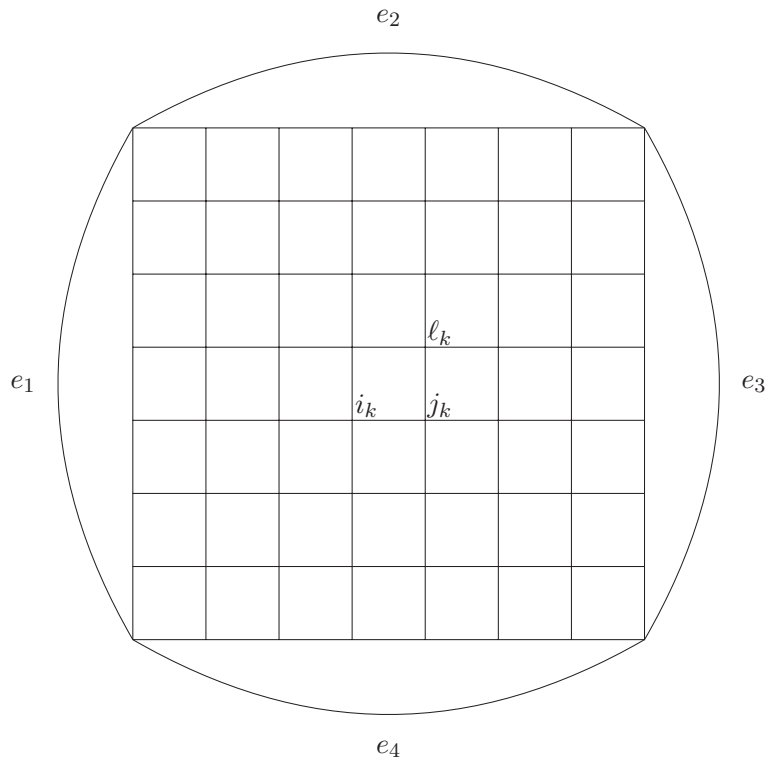


FIGURE 6.2 : Modèle simple de réseau routier urbain. Le quadrillage intérieur représente les axes du centre-ville et les quatre axes extérieurs les branches d'un boulevard périphérique.

6.5.2 Cas d'observations incomplètes : un réseau urbain

On considère maintenant un autre modèle synthétique pour lequel on observe uniquement des données incomplètes. La structure de dépendance du vecteur $\mathbf{X}$ est représenté sur le graphe $\mathcal{G}$ de la figure 6.2. Il s'agit d'une modélisation extrêmement grossière d'un réseau routier. La grille régulière centrale représente un centre-ville (de style américain) et les quatre arcs extérieurs, notés (e_1, e_2, e_3, e_4), peuvent être vus comme les branches d'un boulevard périphérique.

Nature des données manquantes. On commence par expliciter le type de d'observations à notre disposition. Les périphériques d'une ville sont en général équipé de capteurs fixes. On supposera ici que lors de chaque observation les quatre branches du périphériques sont observées, c'est-à-dire que pour chaque observation est de la forme $\mathbf{X}_{\mathbb{V}_k} = \mathbf{x}^k$ avec $\{e_1, e_2, e_3, e_4\} \subset \mathbb{V}_k$. Pour ce qui est de l'observation des variables du centre-ville, on suppose que celles-ci sont observées par paire de variables voisines dans le graphe $\mathcal{G}$. Chaque sous-

vecteur observée $\mathbf{X}_{\mathbb{V}_k}$ est donc de la forme $\mathbb{V}_k = \{a_k, b_k, e_1, e_2, e_3, e_4\}$ avec a_k et b_k deux arcs adjacents – $a_k = (i_k, j_k)$ et $b_k = (j_k, \ell_k)$ avec $i_k \neq \ell_k$ – de $\mathcal{G} \setminus \{e_1, e_2, e_3, e_4\}$. Ceci représente grossièrement les données que l'on obtiendrait de véhicules circulant sur le réseau.

Dans le but d'étudier différents cas on propose de modifier légèrement la manière dont on génère le modèle.

Génération du modèle. On va considérer ici des matrices de précision dont le support correspond au graphe adjoint du graphe de la figure 6.2. On rappelle que le graphe adjoint représente la relation d'adjacence des arcs d'un graphe. On suppose donc que le graphe de dépendance est fixé par les possibilités de routage (on interdit le demi-tour à un carrefour).

On considère deux modèles définis sur ce graphe que l'on notera (m1) et (m2) et qui sont générés de la manière décrite ci-dessous.

(m1) qui est définit selon la procédure décrite dans l'expérience sur données complètes. La procédure consiste à réduire toutes les corrélations de manière uniforme jusqu'à obtenir un modèle valide. Les corrélations obtenues sont en moyenne assez faibles. Les résultats le concernant sont représentés sur la figure 6.3.

(m2) qui est généré selon la procédure suivante :

1. pour chaque arc on génère une corrélation partielle de signe aléatoire et de magnitude aléatoire uniforme sur $[2/10, 1]$;
2. tant que la matrice n'est pas définie positive on diminue les corrélations de plus grande magnitude.

Cette second procédure à l'effet de générer un modèle avec des corrélations plus élevées puisqu'on ne réduit plus celles-ci de manière uniforme. Les résultats le concernant sont représentés sur la figure 6.4.

Résultats. Notons tout d'abord que dans le cas de données incomplètes la log-vraisemblance $\mathcal{L}$ associée à la mesure empirique $\hat{\mu}$, définie à la section 6.2 page 140, ne peut être calculée. En effet, cette mesure $\hat{\mu}$ n'est pas entièrement spécifiée puisque l'on impose seulement certaines marginales selon (6.14). On peut cependant calculer les variations $\Delta\mathcal{L}$ de la log-vraisemblance associée à $\hat{\mu}$ au cours de notre procédure selon (6.25). Ceci n'est pas gênant car la valeur de la log-vraisemblance n'a pas d'interprétation particulière, seules ses variations en ont une.

On propose ici de vérifier la qualité de l'approximation de la vraisemblance par celle basée sur notre mesure empirique $\hat{\mu}$. Pour cela on tracera sur nos courbes, pour chaque famille $\{\mathbf{A}(\kappa)\}_\kappa$ de matrices de précision générée, les deux fonctions suivantes

– la log-vraisemblance $\mathcal{L}(\mu_{\mathbf{X}}, \mathbf{A}(\kappa))$ vis-à-vis à la vraie distribution $\mu_{\mathbf{X}}$ du vecteur $\mathbf{X}$. Celle-ci sera représentée par des lignes continues.

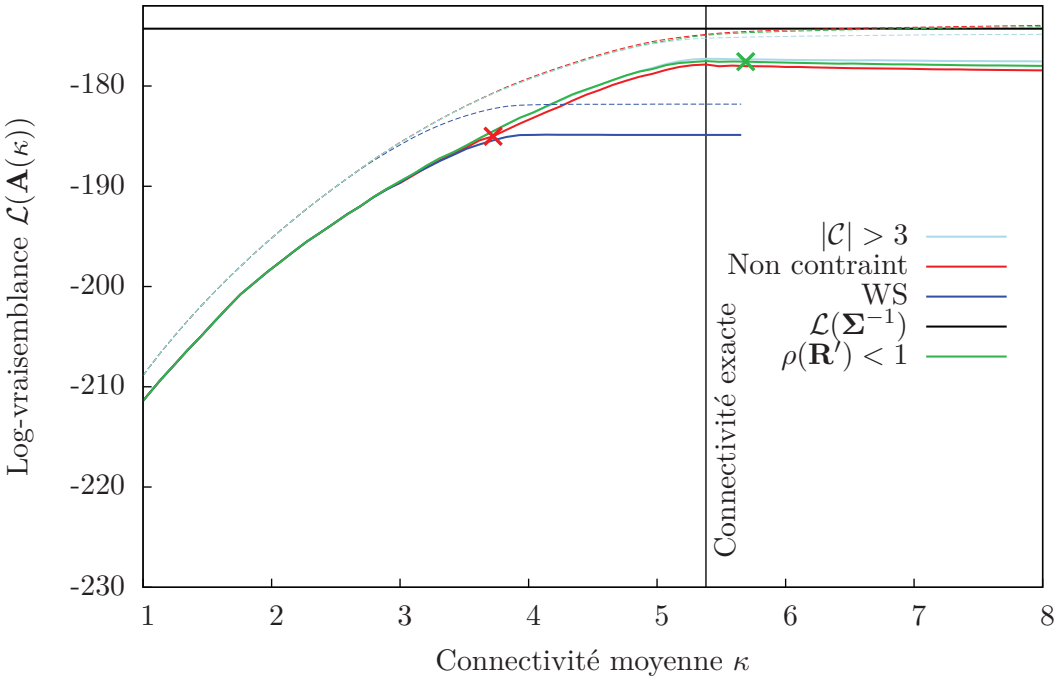


FIGURE 6.3 : Log-vraisemblance vis-à-vis de la distribution de $\mathbf{X}$ (en traits pleins) et vis-à-vis de la mesure empirique $\hat{\mu}$ (en traits pointillés) en fonction de la connectivité du graphe défini par le support de $\mathbf{A}$. La mesure empirique est définie à partir de 1 000 observations pour chaque paire de variables voisines du graphe de la figure 6.2. Les croix représentent la limite de compatibilité avec l'algorithme GaBP des différents estimateurs. L'absence de croix indique que toute la famille d'estimations est compatible avec GaBP. Le modèle sous-jacent est ici le modèle (m1).

– la log-vraisemblance $\mathcal{L}(\hat{\mu}, \mathbf{A}(\kappa))$ vis-à-vis de la distribution empirique $\hat{\mu}$ du vecteur $\mathbf{X}$. $\mathcal{L}(\hat{\mu}, \mathbf{A}(\kappa))$ ne pouvant être déterminée qu'à une constante additive près, on fixe arbitrairement celle-ci de manière à rendre la représentation graphique claire. On la représentera par une ligne pointillée.

On considère ici que les observations de paires sont assez nombreuses, *i.e.* $\mathrm{K} = 1\ 000$. Les résultats du modèle à faibles corrélations (m1) sont représentées à la figure 6.3 et ceux du modèle à fortes corrélations (m2) à la figure 6.4.

Dans ces deux cas l'approximation de la log-vraisemblance par $\mathcal{L}(\hat{\mu}, \cdot)$ est globalement satisfaisante. L'écart entre ces deux mesures s'accroît lorsque la complexité du modèle – c'est-à-dire la connectivité moyenne κ – augmente. Ceci reste vrai dans le cas d'observations complètes, il s'agit tout simplement du phénomène connu de surapprentissage qui conduit à générer des modèles de complexité trop élevée. Les résultats sont alors assez similaires à ceux de la figure 6.1.

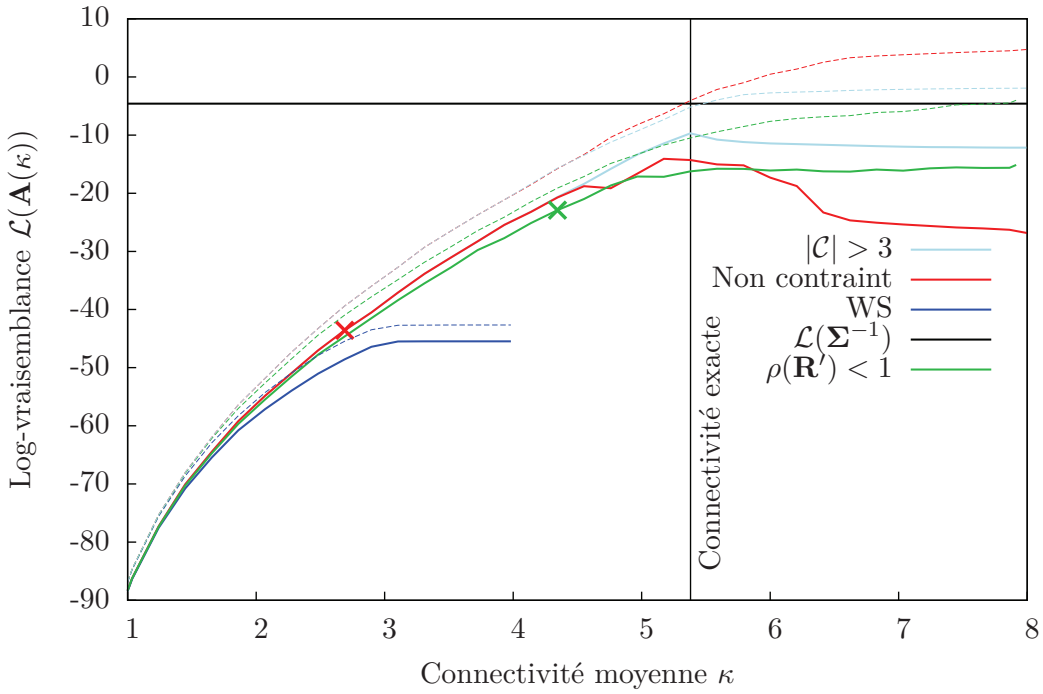


FIGURE 6.4 : Même chose que la figure 6.3 pour le modèle (m2).

La famille de matrices de précision obtenue en imposant la WS reste sous-optimale mais est nettement plus compétitive que dans le cas de la figure 6.1.

Regardons la famille $\{\mathbf{A}(\kappa)\}$ obtenue lorsqu'on impose de ne jamais créer de boucles de taille trois. Dans le cas de corrélations en moyenne faibles – modèle (m1), figure 6.3 – on obtient une famille $\{\mathbf{A}(\kappa)\}$ entièrement GaBP-compatible. Lorsque les corrélations sont plus grandes – modèle (m2), figure 6.4 – ce critère perd cependant son utilité et ne permet pas d'obtenir une meilleure compatibilité avec l'algorithme GaBP. En effet le critère principal indiquant si un cycle est problématique pour GaBP n'est pas sa taille mais le produit des corrélations le long de celui-ci. Dans le cas de corrélations faibles, la probabilité de voir un cycle de longueur strictement supérieure à trois posant problème est suffisamment faible pour pouvoir être négligée (figure 6.3). Ceci n'est plus le cas lorsque les corrélations augmentent (figure 6.4).

Comme dans le cas d'observations complètes, la contrainte $\rho(\mathbf{R}'(\mathbf{A})) < 1$ est celle qui permet d'obtenir les meilleurs modèles compatibles avec GaBP. Dans le cas de corrélations faibles (figure 6.3) elle fournit un modèle de qualité équivalente à la contrainte imposant de ne pas créer de boucle de taille trois. Elle est cependant plus robuste que celle-ci et continue d'être efficace lorsque les corrélations augmentent (figure 6.4).

6.6 Conclusion

Dans ce chapitre nous avons proposé une procédure d'approximation gaussienne d'un champ markovien $\mathbf{X}$ à valeur réelle. Cette approche est complémentaire de celle proposée au chapitre 5. En particulier, le caractère gaussien du modèle construit le rend plus adapté au cas de données unimodales. Pour construire cette approximation on utilise seulement une *matrice de covariance incomplète*. Il est donc possible de raffiner l'estimation proposée à la section 6.2 pour tenir compte de la manière dont le vecteur $\mathbf{X}$ est observé.

La principale nouveauté de l'approche consiste à imposer des contraintes assurant une meilleure compatibilité avec l'algorithme GaBP pour les approximations gaussiennes obtenues. Dans les cas favorables, c'est-à-dire lorsque le nombre d'observations est suffisant ou que les corrélations sur le réseaux ne sont pas trop élevées, la contrainte $\rho(\mathbf{R}') < 1$ semble pertinente. Elle permet d'obtenir des modèles de meilleure qualité GaBP-compatible. Cependant, puisqu'elle n'offre pas de garantie de convergence de GaBP, son utilisation nécessite de déterminer *a posteriori* la limite de compatibilité avec GaBP. Ceci peut être fait par simple dichotomie. Lorsque l'on souhaite avoir une garantie de compatibilité sans effectuer de test, il est possible d'imposer au modèle d'être walk-summable. Cependant la qualité de l'estimation obtenue peut alors se dégrader fortement.

Nous n'avons pas ici abordé la question du nombre de données nécessaires à l'estimation de la matrice de covariance incomplète. Nous avons remarqué que lorsque cette estimation est de mauvaise qualité les résultats se dégradent fortement. Comme les performances de notre approche sont très liées à celles de l'algorithme IPS, une conjecture est que l'existence d'une mesure gaussienne compatible avec les spécifications (6.14) est une condition nécessaire à l'efficacité de notre approche. En effet, l'algorithme IPS converge lorsqu'une telle mesure existe. Des résultats préliminaires suggèrent qu'il est possible de construire des critères empiriques sur les variations de $\mathcal{L}(\hat{\mu}, \cdot)$ permettant de déterminer si l'on se trouve dans une situation favorable ou non. Leur validation nécessite cependant de plus amples expérimentations.

Chapitre 7

Conclusion générale

Dans ces travaux on s'est intéressé à la construction de modèles de variables aléatoires sur un graphe, à partir d'observations incomplètes, adaptés à un problème de régression non standard. L'identité des variables observées (et donc celle des variables à prédire) varie d'une instance à l'autre. Le choix d'un champ markovien aléatoire comme modèle graphique découle naturellement du principe de maximisation de l'entropie de Jaynes. Pour réaliser l'inférence on s'est tourné vers l'algorithme Belief Propagation (BP) dont la simplicité et l'efficacité permettent l'utilisation sur des réseaux de grande taille.

Le chapitre 3 comble un vide relatif dans la littérature en détaillant le rôle exacte de la normalisation dans l'algorithme BP. La liberté des pratiques à ce sujet est justifiée par des propriétés de l'énergie de libre de Bethe ainsi que par la définition des contraintes du problème de minimisation résolu par BP. Ceci valide la pratique répandue consistant à considérer la convergence de beliefs comme le critère robuste. On présente de plus dans ce chapitre une condition suffisante de stabilité d'un point fixe de BP qui s'exprime en fonction des beliefs du point fixe et de propriétés spectrale du graphe de facteurs (théorème 3.12). À notre connaissance il s'agit du premier résultat sur la stabilité des points fixes de BP dans un contexte où plusieurs points fixes peuvent coexister. Cette condition suffisante peut être raffinée (3.25) en une condition faisant intervenir des sommes de matrices stochastiques sur les chemins d'un graphe qui rappellent les « walk-sum » du cas gaussien.

Le chapitre 4 est consacré à l'étude d'un problème d'inférence généralisé sur les champs markoviens à états discrets. Il s'agit de réaliser l'inférence sous des contraintes – assez naturelles dans un cadre bayésien – imposant la distribution marginales de certaines variables. Pour résoudre ce problème on propose l'algorithme mBP (algorithme 2) inspiré de BP mais dont le comportement est assez différent. Le graphe est segmenté au niveau des variables dont les marginales sont imposées et celles-ci ne propagent plus l'information dans le graphe. Le prix à payer en échange est que la convergence de mBP peut être lente sur un arbre même si son comportement global est satisfaisant.

Notons que mBP est une généralisation, plus simple à mettre en œuvre, d'un algorithme proposé par Teh et Welling [92].

Le chapitre 5 fournit la dernière partie de notre approche, qui permet de passer de variables réelles à des variables binaires. On donne une manière simple de modéliser les interactions entre variables aléatoires réelles, grâce à un modèle d'Ising latent, à partir d'une information constituée de

– une fonction de répartition par variable réelle ;
– une matrice de covariance incomplète.

Deux choix naturels de fonctions associant une variable binaire latente à une variable réelle sont présentés. Le premier se base sur une approche déterministe et correspond à une fonction seuil tandis que le second s'appuie sur une approche stochastique et correspond à la fonction de répartition des variables réelles. Le choix de la fonction de répartition comme fonction d'encodage et de son inverse comme fonction de décodage semble le meilleur dès lors que la connectivité du graphe reste raisonnable. Au delà d'une certaine connectivité, qui semble dépendre de la distribution des variables réelles, ce choix perd en efficacité face à des corrélations fortes et le choix d'un encodage déterministe est préférable. On propose une méthode simple et efficace d'estimation des distributions marginales des variables latentes. La question de la définition exacte du modèle d'Ising à partir des ces marginales dépend de la nature des données considérées et nécessite des tests sur données réelles. Cependant les résultats présentés ici permettent d'être assez optimiste pour l'application de notre méthode à des données de trafic routier, la modélisation binaire sous-jacente y étant assez naturelle. La question de la généralisation de cette approche à des modèles de variables latentes à plus d'états reste ouverte. Si le cas d'un encodage déterministe est naturel, l'équivalent du critère entropique reste flou.

Enfin, dans le chapitre 6 s'intéresse à une approche complémentaire de celle proposée aux chapitres 4 et 5. On y présente une heuristique d'approximation gaussienne d'un champ markovien $\mathbf{X}$ à valeurs réelles. La seule information nécessaire est là aussi une matrice de covariance incomplète. Cette heuristique s'appuie sur l'algorithme Iterative Proportionnal Scaling (IPS). On propose des contraintes assurant une meilleure compatibilité avec l'algorithme GaBP pour l'approximation gaussienne (section 6.3). Lorsque l'on possède une bonne estimation de la matrice de covariance, la contrainte imposant une condition nécessaire de convergence de GaBP semble pertinente. Quand ce n'est pas le cas, la qualité des modèles se dégrade fortement, en particulier lorsque aucune mesure jointe gaussienne n'est compatible avec l'estimation de la matrice de covariance. Certains critères empiriques – dont la validation nécessite de plus amples expérimentations – semblent émerger et pourraient permettre de déterminer si l'on se trouve dans une situation favorable ou non à l'algorithme IPS et donc à notre heuristique.

D'une manière générale une originalité de ces travaux est de prendre en

compte une contrainte inhabituelle dans la sélection de modèles : on se restreint à des modèles qui permettent de réaliser l'inférence en temps réel. Ceci nécessite de trouver un bon compromis entre la qualité du modèle et sa simplicité.

Les modèles proposés dans ces travaux s'appuient uniquement sur les corrélations locales au sein d'un réseau de transport. Si cette approche permet de modéliser le phénomène de propagation de la congestion à travers le réseau elle est par contre impuissante à modéliser l'influence de variables extérieures. Certaines variables macroscopiques, telles que les conditions météorologiques par exemple, ont pourtant une forte influence sur le niveau de congestion. La définition même de ces variables macroscopiques est elle aussi non triviale. Pour cela une analyse de données macroscopiques des données trafic est nécessaire. Une approche de ce type a été conduite dans le cadre du projet TRAVESTI [94]. Plusieurs approches sont envisageables pour introduire ces variables macroscopiques dans nos modèles. Déterminer la forme la plus pertinente reste un problème ouvert.

Table des figures

Liste des Algorithmes

Bibliographie

[1] S. Aji, G. Horn, and R. McEliece. On the convergence of iterative decoding on graphs with a single cycle. In *Proceedings of the IEEE International Symposium on Information Theory*, 1998.

[2] S. Aji and R. McEliece. The generalized distributive law. *Information Theory, IEEE Transactions on*, 46(2):325–343, 2000.

[3] S. Amari, K. Kurata, and H. Nagaoka. Information geometry of Boltzmann machines. *Neural Networks, IEEE Transactions on*, 3(2):260–271, 1992.

[4] D. Amit, H. Gutfreund, and H. Sompolinsky. Spin-glass models of neural networks. *Physical Review A*, 32(2):1007, 1985.

[5] D. Angluin. Local and global properties in networks of processors. In *Proceedings of the twelfth annual ACM symposium on Theory of computing*, STOC '80, pages 82–93. ACM, 1980.

[6] B. C. Arnold, E. Castillo, and J.-M. Sarabia. *Conditionally specified distributions*. Springer-Verlag New York, 1992.

[7] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv. Optimal decoding of linear codes for minimizing symbol error rate (corresp.). *Information Theory, IEEE Transactions on*, 20(2):284–287, 1974.

[8] R. Baxter. *Exactly solved models in statistical mechanics*. Dover Publications, 2008.

[9] C. Berge. *Théorie des graphes et ses applications*, volume II of *Collection Universitaire des Mathématiques*. Dunod, 2ème edition, 1967.

[10] H. A. Bethe. Statistical theory of superlattices. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 150(871):552–575, 1935.

[11] D. Bickson. *Gaussian Belief Propagation: Theory and Application*. PhD thesis, Hebrew University of Jerusalem, 2008.

[12] J. Bilmes. On soft evidence in bayesian networks. Technical report, University of Washington, 2004.

[13] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[14] P. Brémaud. *Markov chains: Gibbs fields, Monte Carlo simulation and queues.* Springer-Verlag, 1999.

[15] H. Chan and A. Darwiche. On the revision of probabilistic beliefs using uncertain evidence. *Artificial Intelligence*, 163(1):67–90, 2005.

[16] F. C. Chang. Inversion of a perturbed matrix. *Applied mathematics letters*, 19(2):169–173, 2006.

[17] V. Chernyak and M. Chertkov. Loop calculus and Belief Propagation for q-ary alphabet: Loop tower. In *Information Theory, IEEE International Symposium on*, pages 316–320, 2007.

[18] M. Chertkov and V. Y. Chernyak. Loop series for discrete statistical models on graphs. *Journal of Statistical Mechanics: Theory and Experiment*, 2006.

[19] C. Chow and C. Liu. Approximating discrete probability distributions with dependence trees. *Information Theory, IEEE Transactions on*, 14(3):462–467, 1968.

[20] D. Chowdhury, L. Santen, and A. Schadschneider. Statistical physics of vehicular traffic and some related systems. *Physics Reports*, 329(4):199–329, 2000.

[21] R. Cowell, P. Dawid, S. Lauritzen, and D. Spiegelhalter. *Probabilistic networks and expert systems: Exact computational methods for Bayesian networks*. Springer, 2007.

[22] E. Cramer. Conditional iterative proportional fitting for Gaussian distributions. *Journal of multivariate analysis*, 65(2):261–276, 1998.

[23] J. Darroch and D. Ratcliff. Generalized iterative scaling for log-linear models. *The annals of mathematical statistics*, 43(5):1470–1480, 1972.

[24] A. de La Fortelle, J.-M. Lasgouttes, and C. Furtlehner. Statistical physics algorithms for traffic reconstruction. *ERCIM News*, 67:34–35, 2007.

[25] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 1–38, 1977.

[26] P. Diaconis and D. Strook. Geometric bounds for eigenvalues of Markov chains. *The Annals of Applied Probability*, 1(1):36–61, 1991.

[27] A. Doucet, N. de Freitas, and N. Gordon. An introduction to sequential Monte Carlo methods. *Sequential Monte Carlo methods in practice*, 2001.

[28] F. Eaton and Z. Ghahramani. Choosing a variable to clamp: Approximate inference using conditioned Belief Propagation. In *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, volume 5, pages 145–152, 2009.

[29] G. Elidan and C. Cario. Nonparanormal Belief Propagation (NPNBP). In *Advances in Neural Information Processing Systems 25*, pages 908–916, 2012.

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

[31] C. Furtlehner, A. de La Fortelle, and J.-M. Lasgouttes. Belief Propagation algorithm for a traffic prediction system based on probe vehicles. Research Report 5807, INRIA, 2006.

[32] C. Furtlehner, Y. Han, J.-M. Lasgouttes, and V. Martin. Pairwise MRF Calibration by Perturbation of the Bethe Reference Point. Research Report 8059, INRIA, 2012.

[33] C. Furtlehner, Y. Han, J.-M. Lasgouttes, V. Martin, F. Marchal, and F. Moutarde. Spatial and temporal analysis of traffic states on large scale networks. In *Intelligent Transportation Systems (ITSC), 13th International IEEE Conference on*, pages 1215–1220, 2010.

[34] C. Furtlehner, J.-M. Lasgouttes, and A. Auger. Learning multiple Belief Propagation fixed points for real time inference. *Physica A: Statistical Mechanics and its Applications*, 389(1):149–163, 2010.

[35] C. Furtlehner, J.-M. Lasgouttes, and A. de La Fortelle. Belief Propagation and Bethe approximation for traffic prediction. Research report 6144, INRIA, 2007.

[36] C. Furtlehner, J.-M. Lasgouttes, and A. De La Fortelle. A Belief Propagation approach to traffic prediction using probe vehicles. In *Proceedings of the IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 1022–1027, 2007.

[37] A. Gelman and T. Speed. Characterizing a joint probability distribution by conditionals. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 185–188, 1993.

[38] H. Georgii. *Gibbs Measures and Phase Transitions*. de Gruyter, 1988.

[39] W. Gilks. Markov chain Monte Carlo. *Encyclopedia of Biostatistics*, 2005.

[40] V. Gómez, J. Mooij, and H. J. Kappen. Truncating the loop series expansion for Belief Propagation. *The Journal of Machine Learning Research*, 8:1987–2016, 2007.

[41] P. R. Halmos. *Finite-Dimensional Vector Space*. Springer-Velag, 1974.

[42] J. M. Hammersley and P. Clifford. Markov field on finite graphs and lattices. Preprint, 1971.

[43] D. J. Hartfiel. System behavior in quotient systems. *Applied Mathematics and Computation*, 81(1):31–48, 1997.

[44] R. Herbrich. Minimising the Kullback–Leibler divergence. Technical report, Microsoft Research, 2005.

[45] J. Herrera, D. Work, R. Herring, X. Ban, Q. Jacobson, and A. Bayen. Evaluation of traffic data obtained via GPS-enabled mobile phones: The mobile century field experiment. *Transportation Research Part C: Emerging Technologies*, 18(4):568–583, 2010.

[46] T. Heskes. Stable fixed points of loopy Belief Propagation are minima of the Bethe free energy. *Advances in Neural Information Processing Systems*, 15, 2003.

[47] T. Heskes. On the uniqueness of loopy Belief Propagation fixed points. *Neural Computation*, 16:2379–2413, 2004.

[48] J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558, 1982.

[49] T. Hunter, R. Herring, P. Abbeel, and A. Bayen. Path and travel time inference from GPS probe vehicle data. *NIPS Analyzing Networks and Learning with Graphs*, 2009.

[50] A. Ihler, J. I. Fischer, and A. Willsky. Loopy Belief Propagation: Convergence and effects of message errors. *The Journal of Machine Learning Research*, 6:905–936, 2005.

[51] E. T. Jaynes. Prior probabilities. *Systems Science and Cybernetics, IEEE Transactions on*, 4(3):227–241, 1968.

[52] E. T. Jaynes. *Probability Theory: The Logic of Science (Vol 1)*. Cambridge University Press, 2003.

[53] F. V. Jensen. *An Introduction to Bayesian Networks*. UCL Press, London, 1996.

[54] D. Karger and N. Srebro. Learning markov networks: Maximum bounded tree-width graphs. In *Proceedings of the twelfth annual ACM-SIAM symposium on discrete algorithms*, pages 392–401, 2001.

[55] Y. Kim, M. Valtorta, and J. Vomlel. A prototypical system for soft evidential update. *Applied Intelligence*, 21(1):81–97, 2004.

[56] R. Kindermann, J. Snell, et al. *Markov random fields and their applications*. American Mathematical Society Providence, 1980.

[57] A. Klar, R. Kühne, and R. Wegener. *Mathematical models for vehicular traffic*. Arbeitsgruppe Technomathematik, Univ., 1995.

[58] V. Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(10):1568–1583, 2006.

[59] F. R. Kschischang, B. J. Frey, and H. Loeliger. Factor graphs and the sum-product algorithm. *Information Theory, IEEE Transactions on*, 47(2):498–519, 2001.

[60] C. Laurgeau, A. de La Fortelle, and B. Steux. Brevet 0851809: Système et procédé d'information sur le trafic dans un réseau routier, 2009.

[61] S. Lauritzen. *Graphical models*, volume 17. Oxford University Press, USA, 1996.

[62] S. Lauritzen and D. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 157–224, 1988.

[63] B. H. Lee Dicker and X. Lin. Variable selection and estimation with the seamless-l_0 penalty. *Statistica Sinica*, 23(2):929–962, 2012.

[64] R. J. A. Little and D. B. Rubin. *Statistical Analysis with missing data*. Wiley-Interscience, 2nd edition, 2002.

[65] H. A. Loeliger. An introduction to factor graphs. *Signal Processing Magazine, IEEE*, 21(1):28–41, 2004.

[66] H. Ma and J. Wolf. On tail biting convolutional codes. *Communications, IEEE Transactions on*, 34(2):104–111, 1986.

[67] D. J. Mackay, J. S. Yedidia, W. T. Freeman, Y. Weiss, et al. A conversation about the Bethe free energy and sum-product. Available at http://www.merl.com/publications/TR2001-018/, 2001.

[68] D. Malioutov, J. Johnson, and A. Willsky. Walk-sums and Belief Propagation in Gaussian graphical models. *The Journal of Machine Learning Research*, 7:2031–2064, 2006.

[69] A. Matthai. Estimation of parameters from incomplete data with application to design of sample surveys. *Sankhyā: The Indian Journal of Statistics (1933-1960)*, 11(2):145–152, 1951.

[70] T. Meltzer, A. Globerson, and Y. Weiss. Convergent message passing algorithms: a unifying view. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 393–401, 2009.

[71] M. Mézard and T. Mora. Constraint satisfaction problems and neural networks: A statistical physics perspective. *Journal of Physiology-Paris*, 103(1-2):107–113, 2009.

[72] M. Mézard, G. Parisi, and M. Virasoro. *Spin Glass Theory and Beyond*. World Scientific, Singapore, 1987.

[73] T. Minka. Expectation Propagation for approximate bayesian inference. In *Proceedings of the Seventeenth conference on Uncertainty in Artificial Intelligence*, pages 362–369, 2001.

[74] T. Minka. Divergence measures and message passing. Technical report, Microsoft Research, 2005.

[75] The Mobile Millenium project: `http://traffic.berkeley.edu/`.

[76] J. M. Mooij and H. J. Kappen. On the properties of the Bethe approximation and loopy Belief Propagation on binary networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2005.

[77] J. M. Mooij and H. J. Kappen. Sufficient conditions for convergence of the sum-product algorithm. *Information Theory, IEEE Transactions on*, 53(12):4422–4437, 2007.

[78] K. Murakami. Stability for non-hyperbolic fixed points of scalar difference equations. *Journal of mathematical analysis and applications*, 310(2):492–505, 2005.

[79] K. Murphy, Y. Weiss, and M. Jordan. Loopy Belief Propagation for approximate inference: an empirical study. In *Proceedings of the Fifteenth conference on Uncertainty in Artificial Intelligence*, pages 467–475, 1999.

[80] R. Neal and G. Hinton. A view of the EM algorithm that justifies incremental, sparse, and other variants. *NATO ASI Series D, Behavioural and social sciences*, 89:355–370, 1998.

[81] R. Pan, Y. Peng, and Z. Ding. Belief update in bayesian networks using uncertain evidence. In *Tools with Artificial Intelligence, 18th IEEE International Conference on*, pages 441–444, 2006.

[82] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Network of Plausible Inference*. Morgan Kaufmann, 1988.

[83] The Pumas project: `http://team.inria.fr/pumas/`.

[84] T. Raiko. Partially observed values. In *Neural Networks, Proceedings IEEE International Joint Conference on*, volume 4, pages 2825–2830, 2004.

[85] E. Seneta. *Non-negative matrices and Markov chains*. Springer, 2006.

[86] B. L. Smith, B. M. Williams, and R. Keith Oswald. Comparison of parametric and nonparametric models for traffic flow forecasting. *Transportation Research Part C: Emerging Technologies*, 10(4):303–321, 2002.

[87] T. Speed and H. Kiiveri. Gaussian Markov distributions over finite graphs. *The Annals of Statistics*, 14(1):138–150, 1986.

[88] E. Sudderth, A. Ihler, M. Isard, W. Freeman, and A. Willsky. Nonparametric Belief Propagation. *Communications of the ACM*, 53(10):95–103, Oct. 2010.

[89] N. Taga and S. Mase. On the convergence of loopy Belief Propagation algorithm for different updates rules. *Fundamentals of Electronics, Communications and Computer Sciences, IEICE transactions on*, 89(2):575–582, 2006.

[90] M. Talagrand. Rigorous results for the Hopfield model with many patterns. *Probability theory and related fields*, 110(2):177–275, 1998.

[91] S. Tatikonda and M. Jordan. Loopy Belief Propagation and Gibbs measures. In *Proceedings of the Eighteenth Conference on Uncertainty in Artificial Intelligence*, pages 493–50, 2002.

[92] Y. W. Teh and M. Welling. Passing and bouncing messages for generalized inference. Technical report, UCL, 2001.

[93] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society,Series B*, 58:267–288, 1996.

[94] The Travesti Project: `http://travesti.gforge.inria.fr/`.

[95] D. Uherka and A. M. Sergott. On the continuous dependence of the roots of a polynomial on its coefficients. *American Mathematical Monthly*, pages 368–370, 1977.

[96] A. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *Information Theory, IEEE Transactions on*, 13(2):260–269, 1967.

[97] M. J. Wainwright. *Stochastic processes on graphs with cycles: geometric and variational approaches*. PhD thesis, MIT, 2002.

[98] M. J. Wainwright. Estimating the "wrong" graphical model: benefits in the computation-limited setting. *The Journal of Machine Learning Research*, 7:1829–1859, 2006.

[99] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky. Tree-reweighted Belief Propagation algorithms and approximate ML estimation by pseudomoment matching. *Workshop on Artificial Intelligence and Statistics*, 2003.

[100] Y. Watanabe. *Discrete geometric analysis of message passing algorithm on graphs*. PhD thesis, The Graduate University for Advanced Studies (SOKENDAI), 2010.

[101] Y. Watanabe and K. Fukumizu. Graph zeta function in the Bethe free energy and loopy Belief Propagation. In *Advances in Neural Information Processing Systems 22*, pages 2017–2025, 2009.

[102] Y. Weiss. Correctness of local probability propagation in graphical models with loops. *Neural Computation*, 12(1):1–41, 2000.

[103] Y. Weiss. *Comparing the mean field method and Belief Propagation for approximate inference in MRFs*. The MIT Press, 2001.

[104] Y. Weiss and W. Freeman. Correctness of Belief Propagation in Gaussian graphical models of arbitrary topology. *Neural Computation*, 13(10):2173–2200, 2001.

[105] Y. Weiss and W. Freeman. On the optimality of solutions of the max-product Belief Propagation algorithm in arbitrary graphs. *Information Theory, IEEE Transactions on*, 47(2):723–735, 2001.

[106] M. Welling and Y. W. Teh. Approximate inference in Boltzmann machines. *Artificial Intelligence*, 143(1):19–50, 2003.

[107] W. Wiegerinck and T. Heskes. Fractional Belief Propagation. *Advances in Neural Information Processing Systems*, pages 455–462, 2003.

[108] S. N. Winkler. *Uniqueness of Gibbs Measures with Applications to Gibbs Sampling and the Sum-Product Algorithm*. PhD thesis, Yale, 2007.

[109] M. Yasuda and K. Tanaka. Approximate learning algorithm in boltzmann machines. *Neural computation*, 21(11):3130–3178, 2009.

[110] J. S. Yedidia, W. T. Freeman, and Y. Weiss. Bethe free energies, Kikuchi approximations, and Belief Propagation algorithms. Technical Report 2001-16, MERL, 2001.

[111] J. S. Yedidia, W. T. Freeman, and Y. Weiss. Constructing free-energy approximations and generalized Belief Propagation algorithms. *Information Theory, IEEE Transactions on*, 51(7):2282–2312, 2005.

[112] A. Yuille. CCCP algorithms to minimize the Bethe and Kikuchi free energies: Convergent alternatives to Belief Propagation. *Neural computation*, 14(7):1691–1722, 2002.

Modélisation probabiliste et inférence par l'algorithme Belief Propagation

Résumé : On s'intéresse à la construction et l'estimation – à partir d'observations incomplètes – de modèles de variables aléatoires à valeurs réelles sur un graphe. Ces modèles doivent être adaptés à un problème de régression non standard où l'identité des variables observées (et donc celle des variables à prédire) varie d'une instance à l'autre. La nature du problème et des données disponibles nous conduit à modéliser le réseau sous la forme d'un champ markovien aléatoire, choix justifié par le principe de maximisation d'entropie de Jaynes. L'outil de prédiction choisi dans ces travaux est l'algorithme Belief Propagation – dans sa version classique ou gaussienne – dont la simplicité et l'efficacité permettent son utilisation sur des réseaux de grande taille. Après avoir fourni un nouveau résultat sur la stabilité locale des points fixes de l'algorithme, on étudie une approche fondée sur un modèle d'Ising latent où les dépendances entre variables réelles sont encodées à travers un réseau de variables binaires. Pour cela, on propose une définition de ces variables basée sur les fonctions de répartition des variables réelles associées. Pour l'étape de prédiction, il est nécessaire de modifier l'algorithme Belief Propagation pour imposer des contraintes de type bayésiennes sur les distributions marginales des variables binaires. L'estimation des paramètres du modèle peut aisément se faire à partir d'observations de paires. Cette approche est en fait une manière de résoudre le problème de régression en travaillant sur les quantiles.

D'autre part, on propose un algorithme glouton d'estimation de la structure et des paramètres d'un champ markovien gaussien, basé sur l'algorithme Iterative Proportional Scaling. Cet algorithme produit à chaque itération un nouveau modèle dont la vraisemblance, ou une approximation de celle-ci dans le cas d'observations incomplètes, est supérieure à celle du modèle précédent. Cet algorithme fonctionnant par perturbation locale, il est possible d'imposer des contraintes spectrales assurant une meilleure compatibilité des modèles obtenus avec la version gaussienne de Belief Propagation. Les performances des différentes approches sont illustrées par des expérimentations numériques sur des données synthétiques.

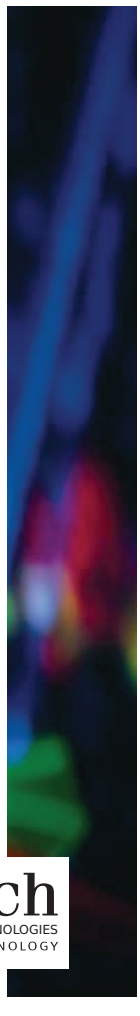
Mots clés : Belief Propagation, Inférence, Champ markovien aléatoire, Modèle gaussien, Iterative Proportional Scaling.

Probabilistic Modelling and Inference using the Belief Propagation Algorithm

Abstract: In this work, we focus on the design and estimation – from partial observations – of graphical models of real-valued random variables. These models should be suited for a non-standard regression problem where the identity of the observed variables (and therefore of the variables to predict) changes from an instance to the other. The nature of the problem and of the available data lead us to model the network as a Markov random field, a choice consistent with Jaynes' maximum entropy principle. For the prediction task, we turn to the Belief Propagation algorithm – in its classical or Gaussian flavor – which simplicity and efficiency make it usable on large scale networks. After providing a new result on the local stability of the algorithm's fixed points, we propose an approach based on a latent Ising model, where dependencies between real-valued variables are encoded through a network of binary variables. To this end, we propose a definition of these variables using the cumulative distribution functions of the real-valued variables. For the prediction task, it is necessary to modify the Belief Propagation algorithm in order to impose Bayesian-like constraints on marginal distributions of the binary variables. Estimation of the model parameters can easily be performed using only pairwise observations. In fact, this approach is a way to solve the regression problem by working on quantiles.

Furthermore, we propose a greedy algorithm for estimating both the structure and the parameters of a Gauss-Markov random field based on the Iterative Proportional Scaling procedure. At each iteration, the algorithm yields a new model which likelihood, or an approximation of it in the case of partial observations, is higher than the one of the previous model. Because of its local perturbation principle, this algorithm allows us to impose spectral constraints, increasing the compatibility with the Gaussian Belief Propagation algorithm. The performances of all approaches are empirically illustrated on synthetic data.

Keywords: Belief Propagation, Inference, Markov Random Field, Gaussian model, Iterative Proportional Scaling.



MINES ParisTech

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY