Graphical models

Laurent Younes

Contents

Introduction

Graphical models provide a convenient framework for building probability distributions over large numbers of interacting variables. They are used in a variety of contexts and applications, including speech processing and language models, image processing, computational biology or neural networks. Their name comes from the facts that their construction relies on several graph theoretic concepts, that they can be easily built and interpreted using graphical representations.

One can measure the complexity of the initial modeling problem simply by counting the number of free parameters that describe a discrete probability distribution over N variables when N is large. In the most favorable case, when each variable can only take two values (binary case), the number of possible configurations is 2^N and specifying a probability of occurrence for each of them requires defining $2^N - 1$ numbers. When N is large (a few tens or more), even enumerating these numbers is unfeasible. One needs to rely on analytical ways in order to define them.

For example, one may focus on small-dimensional marginals (a few variables at a time). More precisely, assume that the variables are $(X_1, \ldots, X_N)$. Specifying all pairwise distributions (i.e., the joint distributions of (X_i, X_j) for $1 \leq i < j \leq N$) requires, in the binary case, to choose $3N(N-1)/2$ parameters, which is manageable. Once this is done, the problem is then to extend these pairwise distributions to a single joint distributions for all variables together. This can be done, as will be presented in these notes, using a general method, called maximum entropy extension, which will induce a special form of graphical model. Extensions do not always exist, however, and making sure that small-dimensional distributions provide a consistent set of constraints is not always easy.

In fact, the most convenient way to build complex distributions over large sets of variables is not via marginal distributions, but via conditional distributions. More precisely, we will see that specific structural constraints on graphical models will arise from *conditional independence* clauses for two variables (or sets of variables) given a set of other variables. Because this approach provides a powerful and generic way for building interesting models, conditional independence and related concepts will be at the heart of graphical model theory. The presentation and discussion of these fundamental concepts will be the subject of the next chapter.

In Chapter 2, these concepts are applied to build graphical models structured on undirected graphs. Chapter 3 tackles the inference problem for such models. In Chapter 4, we will review Bayesian networks, which are graphical models inheriting their structure from directed acyclic graphs. Chapter 5 will review parametric estimation issues.

Disclaimer: *These notes are designed as a tool for students taking the Graphical Models class at Johns Hopkins University. They are work in progress, adapting to the course content*

that changes over time, and do not claim to provide an exhaustive treatise on the subject. Cited references cannot be assumed to be complete either.

This is not published (or even publishable) work. In particular, these notes should not be cited in theses or research papers bibliographies, since their availability over time is not guaranteed. They should not be disseminated in electronic form beyond JHU circles either.

CHAPTER 1

Independence and Conditional Independence

1. Definitions

The statistical behavior of a large set of random variables can be efficiently analyzed from the conditional independence relations they have. In this section, we review this important concept and discuss its most important properties.

We assume that $(\Omega, \mathcal{F}, P)$ is a probability space. This means that $\mathcal{F}$ is a collection of subsets of Ω, and $P : \mathcal{F} \to [0, 1]$ is a probability distribution, both $\mathcal{F}$ and P satisfying the usual probability axioms. A discrete random variable (r.v.) is a mapping $X : \Omega \to R_X$, where $R_X := X(\Omega)$ is the set in which X takes its values and is either finite or countable, and X is such that the set (or "event")

$$[X = x] = \{\omega \in \Omega : X(\omega) = x\}$$

belongs to $\mathcal{F}$, so that $P(X = x)$ is well defined. A real-valued r.v. X is such that the event $X \in I$ belongs to $\mathcal{F}$ for every real interval I, and vector-valued r.v.'s are such that all their coordinates are real-valued r.v.'s. A real-valued r.v. X has a density if there exists an integrable function f_X defined over $\mathbb{R}$ such that

$$P(X \in I) = \int_I f_X(x)dx.$$

We will mostly work with discrete random variables in this text.

If X is a discrete random variable and $f : R_X \to \mathbb{R}$ is a function, the expectation of $f(X)$ is defined by

$$E(f(X)) = \sum_{x \in R_X} f(x)P(X = x).$$

If $R_X \subset \mathbb{R}$, then one can use $f =$ identity, letting

$$E(X) = \sum_{x \in R_X} xP(X = x).$$

Note that this definition is consistent, in the sense that, if $Y = f(X)$, then Y is also a r.v., and one can check that

$$\sum_{y \in R_Y} yP(Y = y) = \sum_{x \in R_X} f(x)P(X = x)$$

so that the two apparently distinct definitions of $E(Y)$ coincide.

The expectation of real-valued r.v.'s in the non-discrete case, is more difficult to define, and requires measure theory in order to handle the general case. However, if X has a density,

then

$$E(X) = \int_{\mathbb{R}} x f_X(x) dx$$

provided the integral is convergent.

DEFINITION 1. *Two discrete random variables $X : \Omega \to R_X$ and $Y : \Omega \to R_Y$ are independent if and only if*

$$\forall x \in R_X, \forall y \in R_Y : P(X = x, Y = y) = P(X = x)P(Y = y).$$

The general definition for arbitrary r.v.'s is that

$$E(f(X)g(Y)) = E(f(X))\, E(g(Y))$$

for any pair of functions $f : R_X \to \mathbb{R}$ and $g : R_Y \to \mathbb{R}$ that satisfy a "measurability" condition ensuring that both $f(X)$ and $g(Y)$ are random variables.

Independence between two variables means that X brings no information on Y and vice versa. This can be rephrased in terms of conditional expectations and distributions. If X and Y are discrete r.v.'s, then

$$P(Y = y \mid X = x) = P(Y = y, X = x)/P(X = x)$$

if $P(X = x) > 0$ and is undefined otherwise. Then, if Y is real-valued and discrete, one defines the conditional expectation of Y given X, denoted $E(Y \mid X)$, by

$$E(Y \mid X)(\omega) = \sum_{y \in R_Y} y P(Y = y \mid X = X(\omega))$$

for all ω such that $P(X = X(\omega)) > 0$. Note that $E(Y|X)$ is a random variable, defined over Ω. It however only depends on the values of X, in the sense that $E(Y \mid X)(\omega) = E(Y \mid X)(\omega')$ if $X(\omega) = X(\omega')$. We will use the notation

$$E(Y \mid X = x) = \sum_{y \in R_Y} y P(Y = y \mid X = x)$$

so that $E(Y \mid X)(\omega) = E(Y \mid X = X(\omega))$.

One can characterize $E(Y \mid X)$ by the properties

$$\begin{cases} E(Y|X) \text{ is a function of } X \\ \forall f : R_X \to \mathbb{R}, E(E(Y \mid X) f(X)) = E(Y f(X)). \end{cases} \tag{1}$$

The proof that our definition of $E(Y \mid X)$ for discrete random variables is the only one satisfying these properties should be carried out by the reader. The interest of reformulating the definition of the conditional expectation via (1) is that this provides a definition that works for general random variables (with the additional assumption that f is measurable), not only for discrete ones.

With these definitions, one can easily check that X and Y are independent if and only if, for any function $g : R_Y \mapsto \mathbb{R}$, one has

$$E(g(Y) \mid X) = E(g(Y)).$$

NOTATION 1. *Independence is a property that involves two variables X and Y and an underlying probability distribution P. Independence of X and Y relative to P will be denoted $(X \perp\!\!\!\perp Y)_P$, although we will often only write $X \perp\!\!\!\perp Y$ when there is no ambiguity on what P is.*

More than independence, the concept of conditional independence will be fundamental for our purposes. It requires three variables, say X, Y, Z. We will say that X and Y are conditionally independent given Z if, for any pair of functions f and g as above

$$E(f(X)g(Y)|Z) = E(f(X)|Z)E(g(Y)|Z). \tag{2}$$

(Unless otherwise specified, all our r.v.'s from now on are assumed to be discrete.) Of course, this can also be interpreted via conditional probabilities, namely, for any $x \in R_X$, $y \in R_Y$ and $z \in R_Z$ such that $P(Z = z) > 0$,

$$P(X = x, Y = y \mid Z = z) = P(X = x \mid Z = z)P(Y = y \mid Z = z).$$

An equivalent statement is that, for any z such that $P(Z = z) \neq 0$, X and Y are independent when P is replaced by the conditional distribution $P(.|Z = z)$.

Multiplying both terms by $P(Z = z)^2$, we get the equivalent statement: X and Y are conditionally independent given Z if and only if,

$$\forall x, y, z : P(X = x, Y = y, Z = z)P(Z = z) = P(X = x, Z = z)P(Y = y, Z = z). \tag{3}$$

Note that the identity is meaningful, and always true, for $P(Z = z) = 0$, so that this case does not need to be excluded anymore.

Conditional independence can be interpreted by the statement that X brings no more information on Y than what is already provided by Z: one has

$$P(Y = y \mid X = x, Z = z) = \frac{P(Y = y, X = x, Z = z)}{P(X = x, Z = z)} = \frac{P(Y = y, Z = z)}{P(Z = z)}$$

as directly deduced from (3). (This computation being valid as soon as $P(X = x, Z = z) > 0$.)

NOTATION 2. *To indicate that X and Y are conditionally independent given Z for the distribution P, we will write $(X \perp\!\!\!\perp Y \mid Z)_P$ or simply $(X \perp\!\!\!\perp Y \mid Z)$.*

So we have the equivalence:

$$(X \perp\!\!\!\perp Y \mid Z)_P \Leftrightarrow \big(\forall z : P(Z = z) > 0 \Rightarrow (X \perp\!\!\!\perp Y)_{P(.,.|Z=z)}\big).$$

Absolute independence is like "independence conditional to no variable", and we will use the notation $\emptyset$ for the "empty" random variable that contains no information (for example, a set-valued random variable that always returns the empty set, or any constant variable). So we have the tautology

$$X \perp\!\!\!\perp Y \Leftrightarrow (X \perp\!\!\!\perp Y \,|\, \emptyset).$$

Note that, dealing with discrete variables, all previous definitions automatically extend to groups of variables: for example, if Z_1, Z_2 are two discrete variables, so is $Z = (Z_1, Z_2)$ and we immediately obtain a definition for the conditional independence of X and Y given Z_1 and Z_2, denoted $(X \perp\!\!\!\perp Y | Z_1, Z_2)$.

2. Fundamental properties

Proposition 1 below lists important properties of conditional independence that will be used repeatedly in these notes. All considered variables are discrete without additional notice. Before stating this proposition, we need the following definition.

DEFINITION 2. *One says that the joint distribution of the random variables $(X_1, \ldots, X_N)$ is positive if the following holds:*

$$P(X_1 = x_1, \ldots, X_N = x_N) = 0 \Rightarrow \exists k \in \{1, \ldots, N\} : P(X_k = x_k) = 0.$$

This definition states that any conjunction of events for different X_k's has positive probability, as soon as each of them has positive probability (if all events may occur, then they may occur together). Equivalently, it says that the support of the joint distribution of $(X_1, \ldots, X_N)$ is the product space of the supports of the X_k's. (The support of a discrete distribution being the set of points that have positive probability.)

PROPOSITION 1. *Let X, Y, Z and W be random variables. The following properties are true.*

(CI1) *Symmetry:* $(X \perp\!\!\!\perp Y | Z) \Rightarrow (Y \perp\!\!\!\perp X | Z)$.
(CI2) *Decomposition:* $(X \perp\!\!\!\perp (Y, W) | Z) \Rightarrow (X \perp\!\!\!\perp Y | Z)$.
(CI3) *Weak union:* $(X \perp\!\!\!\perp (Y, W) | Z) \Rightarrow (X \perp\!\!\!\perp Y | (Z, W))$.
(CI4) *Contraction:* $(X \perp\!\!\!\perp Y \,|\, Z)$ *and* $(X \perp\!\!\!\perp W \,|\, (Z, Y)) \Rightarrow (X \perp\!\!\!\perp (Y, W) \,|\, Z)$.
(CI5) *Intersection: assume that the joint distribution of W, Y and Z is positive. Then*

$$(X \perp\!\!\!\perp W \,|\, (Z, Y)) \text{ and } (X \perp\!\!\!\perp Y \,|\, (Z, W)) \Rightarrow (X \perp\!\!\!\perp (Y, W) \,|\, Z).$$

PROOF. Properties (CI1) and (CI2) are easily deduced from (3) and left to the reader. To prove the last three, we will use the notation $P(x), P(x, y)$ etc. instead of $P(X = x), P(X = x, Y = y)$, etc. to save space. Identities are assumed to hold for all x, y, z, w unless stated otherwise.

For (CI3), we must prove, according to (3), that

$$P(x, y, z, w)P(z, w) = P(x, z, w)P(y, z, w) \tag{4}$$

whenever $P(x, y, z, w)P(z) = P(x, z)P(y, z, w)$. Summing this last equation over y (or applying (CI2)) yields $P(x, z, w)P(z) = P(x, z)P(z, w)$. We can note that all terms in (4) vanish when $P(z) = 0$, so that the identity is true in this case. When $P(z) \neq 0$, the right-hand side of (4) becomes

$$\begin{aligned}(P(x, z)P(z, w)/P(z))P(y, z, w) = (P(x, z)P(y, z, w)/P(z))P(z, w) \\ = P(x, y, z, w)P(z, w),\end{aligned}$$

using once again the hypothesis. This proves (CI3).

For (CI4), the hypotheses are

$$\begin{cases} P(x,y,z)P(z) = P(x,z)P(y,z) \\ P(x,y,z,w)P(y,z) = P(x,y,z)P(y,z,w) \end{cases}$$

and the conclusion must be

$$P(x,y,z,w)P(z) = P(x,z)P(y,z,w). \tag{5}$$

Since (5) is true when $P(y,z) = 0$, we assume that this probability does not vanish and write

$$\begin{aligned} P(x,y,z,w)P(z) &= P(x,y,z)P(z)P(y,z,w)/P(y,z) \\ &= P(x,z)P(y,z)P(y,z,w)/P(y,z) \\ &= P(x,z)P(y,z,w) \end{aligned}$$

yielding (5).

For (CI5), assuming

$$\begin{cases} P(x,y,z,w)P(y,z) = P(x,y,z)P(y,z,w) \\ P(x,y,z,w)P(z,w) = P(x,z,w)P(y,z,w), \end{cases} \tag{6}$$

we want to show that

$$P(x,y,z,w)P(z) = P(x,z)P(y,z,w).$$

Since this identity is obviously true when any of the events $W = w, Y = y$ or $Z = z$ has zero probability, we can assume that their probabilities are positive, which, by assumption, also implies that all joint probabilities are positive. From the two identities, we get

$$P(x,y,z,w)/P(y,z,w) = P(x,y,z)/P(y,z) = P(x,z,w)/P(z,w)$$

This implies, in particular

$$P(x,y,z) = P(y,z)P(x,z,w)/P(z,w)$$

that we can sum over y to obtain

$$P(x,z) = P(z)P(x,z,w)/P(z,w)$$

We therefore get

$$P(x,y,z,w)/P(y,z,w) = P(x,z,w)/P(z,w) = P(x,z)/P(z),$$

which is what we wanted. □

A counter-example of (CI5) when the positivity assumption is not satisfied can be built as follows: let X be a Bernoulli random variable, and $Y = W = X$. Let Z be any Bernoulli random variable, independent from X. Given Z and W, X and Y are constant and therefore independent. Similarly, given Z and Y, X and W are constant and therefore independent. However, given Z, X and (Y, W) are not independent (they are equal and non constant).

3. Mutual Independence

Another concept of interest is the mutual (conditional) independence of more than two random variables. The random variables $(X_1, \ldots, X_N)$ are mutually conditionally independent given Z if and only if

$$E(f_1(X_1) \cdots f_N(X_N)|Z) = E(f_1(X_1)|Z) \cdots E(f_N(X_N)|Z)$$

for any functions $f_1, \ldots, f_N$. In terms of discrete probabilities, this can be written as

$$P(X_1 = x_1, \ldots, X_N = x_N, Z = z)P(Z = z)^{N-1} = \\ P(X_1 = x_1, Z = z) \cdots P(X_N = x_N, Z = z).$$

This will be summarized with the notation

$$(X_1 \perp\!\!\!\perp \cdots \perp\!\!\!\perp X_N \mid Z).$$

We have the proposition

PROPOSITION 2. *For variables $X_1, \ldots, X_N$ and Z, the following properties are equivalent.*

(i) $(X_1 \perp\!\!\!\perp \cdots \perp\!\!\!\perp X_N \mid Z)$;
(ii) *For all $S, T \subset \{1, \ldots, N\}$ with $S \cap T = \emptyset$, we have:*

$$((X_i, i \in S) \perp\!\!\!\perp (X_j, j \in T) \mid Z);$$

(iii) *For all $s \in \{1, \ldots, N\}$, we have: $(X_s \perp\!\!\!\perp (X_t, t \neq s) \mid Z)$;*
(iv) *For all $s \in \{2, \ldots, N\}$, we have: $(X_s \perp\!\!\!\perp (X_1, \ldots, X_{s-1}) \mid Z)$.*

PROOF. We prove the proposition in the discrete case. It is clear that (i) $\Rightarrow \cdots \Rightarrow$ (iv) so it suffices to prove that (iv) $\Rightarrow$ (i). For this, simply write (applying (iv) repeatedly to $s = N - 1.N - 2. \ldots$)

$$\begin{aligned} & P(X_1 = x_1, \ldots, X_N = x_N, Z = z)P(Z = z)^{N-1} \\ = \; & P(X_1 = x_1, \ldots, X_{N-1} = x_{N-1}, Z = z)P(Z = z)^{N-2}P(X_N = x_N, Z = z) \\ \vdots \; & \\ = \; & P(X_1, Z = z) \cdots P(X_N = x_N, Z = z). \end{aligned}$$

□

4. Relation with Information Theory

Several concepts in information theory are directly related to independence between random variables. The entropy of a discrete probability distribution over a finite set Ω is defined by

$$H(P) = -\sum_{\omega \in \Omega} \ln P(\omega)P(\omega). \tag{7}$$

Similarly, the entropy of a random variable X is defined by

$$H(X) = -\sum_{x \in R_X} \ln P(X = x)P(X = x). \tag{8}$$

The entropy is always positive, and provides a measure of the uncertainty associated to P. For a given finite set Ω, it is maximal when P is uniform over Ω, and minimal (and vanishes) when P is supported by a single $\omega \in \Omega$ (i.e. $P(\omega) = 1$).

One defines the entropy of two or more random variables as the entropy of their joint distribution, so that, for example,

$$H(X,Y) = -\sum_{(x,y)\in R_{X,Y}} \ln P(X=x, Y=y) P(X=x, Y=y).$$

We have the proposition:

PROPOSITION 3. *For random variables $X_1, \ldots, X_n$, one has*

$$H(X_1, \ldots, X_n) \leq H(X_1) + \cdots + H(X_n)$$

with equality if and only if $(X_1, \ldots, X_n)$ are mutually independent.

The proof of this proposition uses properties of the Kullback-Leibler divergence, which compares two probability distributions, and is defined as follows.

DEFINITION 3. *The Kullback-Leibler (KL) divergence between two probability distributions π and π' on a finite set Ω is defined by*

$$D(\pi \| \pi') = \sum_{\omega\in\Omega} \pi(\omega) \ln \frac{\pi(\omega)}{\pi'(\omega)}.$$

with the convention $\pi \log(\pi/\pi') = 0$ if $\pi = 0$ and $= \infty$ if $\pi > 0$ and $\pi' = 0$.

This divergence has several interesting properties, the most important for us is that it can be used to assess the proximity between two distributions. The following proposition is a standard result that we state without proof [**5**].

PROPOSITION 4. *Given two probability distributions π and π' on Ω, we have $D(\pi\|\pi') \geq 0$ with equality if and only if $\pi = \pi'$.*

Returning to Proposition 3, a straightforward computation (which is left to the reader) shows that

$$H(X_1) + \cdots + H(X_n) - H(X_1, \ldots, X_n) = D(\pi\|\pi')$$

with $\pi(x_1, \ldots, x_n) = P(X_1 = x_1, \ldots, X_n = x_n)$ and $\pi'(x_1, \ldots, x_n) = \prod_{k=1}^n P(X_k = x_k)$. This makes Proposition 3 a direct consequence of Proposition 4.

The mutual information between two random variables X and Y is defined by

$$I(X,Y) = H(X) + H(Y) - H(X,Y). \tag{9}$$

From Proposition 3, $I(X,Y)$ is nonnegative and vanishes if and only if X and Y are independent. Also from the proof of Proposition 3, $I(X,Y)$ is equal to $D(P_{(X,Y)}\|P_X \otimes P_Y)$ where the first probability is the joint distribution of X and Y and the second one the product of the marginals of X and Y, which coincides with $P_{X,Y}$ if and only if X and Y are independent.

If X and Y are two random variables, and $y \in R_Y$ with $P(Y = y) > 0$, the entropy of the conditional probability $x \mapsto P(X = x|Y = y)$ is denoted $H(X \mid Y = y)$, and is a

function of y. The conditional entropy of X given Y, denoted $H(X \mid Y)$ is the expectation of $H(X \mid Y = y)$ for the distribution of Y, i.e.,

$$H(X \mid Y) = \sum_{y \in R_Y} H(X \mid Y = y) P(Y = y)$$
$$= - \sum_{x \in R_X} \sum_{y \in R_Y} \ln P(X = x | Y = y) P(X = x, Y = y).$$

So, we have (with a straightforward proof)

PROPOSITION 5. *Given two random variables X and Y, we have*

$$\begin{aligned} H(X \mid Y) &= -E_{X,Y}(\ln P(X = \cdot \mid Y = \cdot)) \\ &= H(X, Y) - H(Y) \end{aligned} \tag{10}$$

This proposition also immediately yields:

$$I(X, Y) = H(X) - H(X \mid Y) = H(Y) - H(Y \mid X). \tag{11}$$

The identity $H(X, Y) = H(X \mid Y) + H(Y)$ that is deduced from (10) can be generalized to more than two random variables (the proof being left to the reader), yielding, if $X_1, \ldots, X_n$ are random variables:

$$H(X_1, \ldots, X_n) = \sum_{k=1}^{n} H(X_k \mid X_1, \ldots, X_{k-1}). \tag{12}$$

If Z is an additional random variable, the following identity is obtained by applying the previous one to conditional distributions given $Z = z$ and taking averages over z:

$$H(X_1, \ldots, X_n \mid Z) = \sum_{k=1}^{n} H(X_k \mid X_1, \ldots, X_{k-1}, Z). \tag{13}$$

The following proposition characterizes conditional independence in terms of entropy.

PROPOSITION 6. *Let X, Y and Z be three random variables. The following statements are equivalent.*

(i) *X and Y are conditionally independent given Z.*
(ii) $H(X, Y \mid Z) = H(X \mid Z) + H(Y \mid Z)$
(iii) $H(X \mid Y, Z) = H(X \mid Y)$

Moreover, when (i) to (iii) are satisfied, we have:

(iv) $I(X, Y) \leq \min(I(X, Z), I(Y, Z))$.

PROOF. From Proposition 3, we have, for any three random variables X, Y, Z, and any z such that $P(Z = z) > 0$,

$$H(X, Y | Z = z) \leq H(X | Z = z) + H(Y | Z = z).$$

Taking expectations on both sides implies the important inequality

$$H(X, Y | Z) \leq H(X | Z) + H(Y | Z) \tag{14}$$

and equality occurs if and only if $P(X = x, Y = y|Z = z) = P(X = x|Z = z)P(Y = y|Z = z)$ whenever $P(Z = z) > 0$, that is, if and only if X and Y are conditionally independent given Z. This proves that (i) and (ii) are equivalent. The fact that (ii) and (iii) are equivalent comes from (13), which gives, for any three random variables

$$H(X, Y|Z) = H(X|Y, Z) + H(Y|Z). \tag{15}$$

To prove that (i)-(iii) implies (iv), we note that (14) and (15) imply that, for any three random variables:

$$H(X|Y, Z) \leq H(X|Y).$$

If X and Y are conditionally independent given Z, then the right-hand side is equal to $H(X|Z)$ and this yields

$$I(X, Y) = H(X) - H(X|Y) \leq H(X) - H(X|Z) = I(X, Z).$$

By symmetry, we must also have $I(X, Y) \leq I(Y, Z)$ so that (iv) is true. □

Statement (iv) is often called the data-processing inequality, and has been used to infer conditional independence within gene networks [**26**].

CHAPTER 2

Models on Undirected Graphs

1. Graphical Representation of Conditional Independence

1.1. Definitions. An undirected graph is a collection of vertices and edges, in which edges link pairs of vertices without order. Edges can therefore be identified to *subsets* of cardinality two of the set of vertices, V. This yields the definition:

DEFINITION 4. *An undirected graph G is a pair $G = (V, E)$ where V is a finite set of vertices and elements $e \in E$ are subsets $e = \{s, t\} \subset V$.*

Note that edges in undirected graphs are defined as sets, i.e., unordered pairs, which are delimited with braces in these notes. Later on, we will use parentheses to represent ordered pairs, $(s, t) \neq (t, s)$. We will write $s \sim_G t$, or simply $s \sim t$ to indicate that s and t are connected by an edge in G (we also say that s and t are neighbors in G).

DEFINITION 5. *A path in an undirected graph $G = (V, E)$ is a finite sequence $(s_0, \dots, s_N)$ of vertices such that $s_{k-1} \sim s_k \in E$. (A sequence, (s_0), of length 1 is also a path by extension.)*

We say that s and t are connected by a path if either $s = t$ or there exists a path $(s_0, \dots, s_N)$ such that $s_0 = s$ and $s_N = t$.

A subset $S \subset G$ is connected if any pair of elements in S can be connected by a path.

A subset $T \subset G$ separates two other subsets S and S' if all paths between S and S' must pass in T. We will write $(S \perp\!\!\!\perp S' \,|\, T)$ in such a case.

One of the goals of this chapter is to relate the notion of conditional independence within a set of variables to separation in a suitably chosen undirected graph with vertices in one to one correspondence with the variables. This will also justifies the similarity of notation used for separation and conditional independence.

We have the following simple fact:

LEMMA 1. *Let $G = (V, E)$ be an undirected graph, and $S, S', T \subset V$. Then*

$$(S \perp\!\!\!\perp S' \,|\, T) \Rightarrow S \cap S' \subset T.$$

Indeed, if $(S \perp\!\!\!\perp S' \,|\, T)$ and $s_0 \in S \cap S'$, the path (s_0) links S and S' and therefore must pass in T.

Proposition 1 translates into similar properties for separation:

PROPOSITION 7. *Let (V, E) be an undirected graph and S, T, U, R be subsets of V. The following properties hold*

(i) $(S \perp\!\!\!\perp T | U) \Leftrightarrow (T \perp\!\!\!\perp S | U)$.
(ii) $(S \perp\!\!\!\perp T \cup R | U) \Rightarrow (S \perp\!\!\!\perp T | U)$.
(iii) $(S \perp\!\!\!\perp T \cup R | U) \Rightarrow (S \perp\!\!\!\perp T | U \cup R)$.
(iv) $(S \perp\!\!\!\perp T \,|\, U)$ *and* $(S \perp\!\!\!\perp R \,|\, U \cup T) \Leftrightarrow (S \perp\!\!\!\perp T \cup R \,|\, U)$.

(v) $U \cap R = \emptyset, (S \perp\!\!\!\perp R \,|\, U \cup T)$ *and* $(S \perp\!\!\!\perp U \,|\, T \cup R) \Rightarrow (S \perp\!\!\!\perp U \cup R \,|\, T)$.

PROOF. (i) is obvious, and for (ii) (and (iii)), if any path between S and $T \cup R$ must pass by U, the same is obviously true for a path between S and T.

For the $\Rightarrow$ part of (iv), if a path links S and $T \cup R$, then it either links S and T and must pass through U by the first assumption, or link S and R and therefore pass through U or T by the second assumption. But if the path passes through T, it must also pass through U before by the first assumption. In all cases, the path passes through U. The $\Leftarrow$ part of (iv) is obvious.

Finally, consider (v) and take a path between two distinct elements in S and $U \cup R$. Consider the first time the path hits U or R, and assumes that it hits U (the other case being treated similarly by symmetry). Notice that the path cannot hit both U and R at the same point since $U \cap R = \emptyset$. From the assumptions, the path must hit $T \cup R$ before passing by U, and the intersection cannot be in R, so it is in T, which is the conclusion we wanted. □

To make a connection between separation in graphs and conditional independence between random variables, we consider a graph $G = (V, E)$ and a family of random variables $(X_s, s \in V)$ indexed by V. If $S \subset V$, we will use the notation X_S for the family $(X_s, s \in S)$. Such a collection of variables is often called *a random field* over V. We can now write the definition:

DEFINITION 6. *Let $G = (V, E)$ be an undirected graph and $X = X_V = (X_s, s \in V)$ a set of random variables indexed by V. We say that X is Markov (or has the Markov property) relative to G (or is G-Markov, or is a Markov random field on G) if and only if*

$$(S \perp\!\!\!\perp T \,|\, U) \Rightarrow (X_S \perp\!\!\!\perp X_T \,|\, X_U). \tag{16}$$

Letting the observation over an empty set S be empty, i.e., $X_\emptyset = \emptyset$, this definition includes the statement that, if S and T are disconnected (i.e., there is no path between them: they are separated by the empty set), then $(X_S \perp\!\!\!\perp X_T \,|\, \emptyset)$: X_S and X_T are independent.

1.2. Reduction of the Markov Property. We now proceed, in a series of steps, to a simplification of Definition 6 in order to obtain a minimal number of conditional independence statements. Note that, in its current form, Definition 6 requires to check (16) for any three subsets of V, which provides a huge number of conditions. Fortunately, as we will see, these conditions are not independent, and checking a much smaller number of them will ensure that all of them are true.

The first step for our reduction is provided by the following lemma.

LEMMA 2. *Let $G = (V, E)$ be an undirected graph and $X_V = (X_s, s \in V)$ a set of random variables indexed by V. Then X is G-Markov if and only if, for $S, T, U \subset V$,*

$$S \cap U = T \cap U = \emptyset \text{ and } (S \perp\!\!\!\perp T \,|\, U) \Rightarrow (X_S \perp\!\!\!\perp X_T \,|\, X_U). \tag{17}$$

PROOF. Assume that (17) is true, and take any S, T, U with $(S \perp\!\!\!\perp T \,|\, U)$. Let $A = S \cap U$, $B = T \cap U$ and $C = A \cup B$. Partition S in $S = S_1 \cup A$, T in $T_1 \cup B$ and U in $U_1 \cup C$. From

$(S \perp\!\!\!\perp T \,|\, U)$, we get $(S_1 \perp\!\!\!\perp T_1 \,|\, U)$. Since $S_1 \cap U = T_1 \cap U = \emptyset$, this implies $(X_{S_1} \perp\!\!\!\perp X_{T_1} \,|\, X_U)$. But this implies $((X_{S_1}, X_A) \perp\!\!\!\perp (X_{T_1}, X_B) \,|\, X_U)$. Indeed, this property requires

$$P^X(x_{S_1}, x_A, x_{T_1}, x_B, x_{U_1}, y_C) P^X(x_{U_1}, y_C) = P^X(x_{S_1}, x_A, x_{U_1}, y_C) P^X(x_{T_1}, x_B, x_{U_1}, y_C)$$

If the configurations x_A, x_B, y_C are not consistent (i.e., $x_t \neq y_t$ for some $t \in C$), then both sides vanish. So we can assume $x_C = y_C$ and remove x_A and x_B from the expression, since they are redundant. The resulting identity is true since it exactly states $(X_{S_1} \perp\!\!\!\perp X_{T_1} \,|\, X_U)$. □

Define the set of neighbors of $s \in V$ (relative to the graph G) as the set of $t \neq s$ such that $\{s, t\} \in E$ and denote this set by $\mathcal{V}_s$. For $S \subset V$ define also

$$\mathcal{V}_S = S^c \cap \bigcup_{s \in S} \mathcal{V}_s$$

which is the set of neighbors of all vertices in S that do not belong to S. (Here S^c denotes the complementary set of S, $S^c = V \setminus S$.) Finally, let $\mathcal{R}_S$ denote the vertices that are "remote" from S, $\mathcal{R}_S = (S \cup \mathcal{V}_S)^c$.

We have the following important reduction of the condition in Definition 6.

PROPOSITION 8. *X is Markov relative to G if and only if, for any $S \subset V$,*

$$(X_S \perp\!\!\!\perp X_{\mathcal{R}_S} \,|\, X_{\mathcal{V}_S}). \tag{18}$$

This says that

$$P(X_S = x_S | X_{S^c} = x_{S^c})$$

only depends (when defined) on variables x_t for $t \in S \cup \mathcal{V}_S$.

PROOF. First note that $(S \perp\!\!\!\perp \mathcal{R}_S \,|\, \mathcal{V}_S)$ is always true, since any path reaching S from S^c must pass through $\mathcal{V}_S$. This immediately proves the "only if" part of the proposition.

Consider now the "if" part. Take S, T, U such that $(S \perp\!\!\!\perp T \,|\, U)$. We want to prove that $(X_S \perp\!\!\!\perp X_T \,|\, X_U)$. According to Lemma 2, we can assume, without loss of generality, that $S \cap U = T \cap U = \emptyset$.

Define R as the set of vertices v in V such that there exists a path between S and v that does not pass in U. Then:

1. $S \subset R$: the path (s) for $s \in S$ does not pass in U since $S \cap U = \emptyset$.
2. $U \cap R = \emptyset$ by definition.
3. $\mathcal{V}_R \subset U$: assume that there exists a point r in $\mathcal{V}_R$ which is not in U. Then r has a neighbor, say r' in R. By definition of R, there exists a path from S to r' that does not hit U, and this path can obviously be extended by adding r at the end to obtain a path that still does not hit U. But this implies that $r \in R$, which contradicts the fact that $\mathcal{V}_R \cap R = \emptyset$.
4. $T \cap (R \cup \mathcal{V}_R) = \emptyset$: if $t \in T$, then $t \notin R$ from $(S \perp\!\!\!\perp T \,|\, U)$ and $t \notin \mathcal{V}_R$ from $T \cap U = \emptyset$.

Using these, we can write (each decomposition being a partition, defining the sets A, B and C, see Fig. 1) $R = S \cup A$, $U = \mathcal{V}_R \cup C$, $(R \cup \mathcal{V}_R)^c = T \cup C \cup B$, and from $(X_R \perp\!\!\!\perp X_{\mathcal{R}_R} \,|\, X_{\mathcal{V}_R})$, we get

$$((X_S, X_A) \perp\!\!\!\perp (X_T, X_C, X_B) \,|\, X_{\mathcal{V}_R})$$

which implies

$$((X_S, X_A) \perp\!\!\!\perp (X_T, X_B) \,|\, X_U)$$

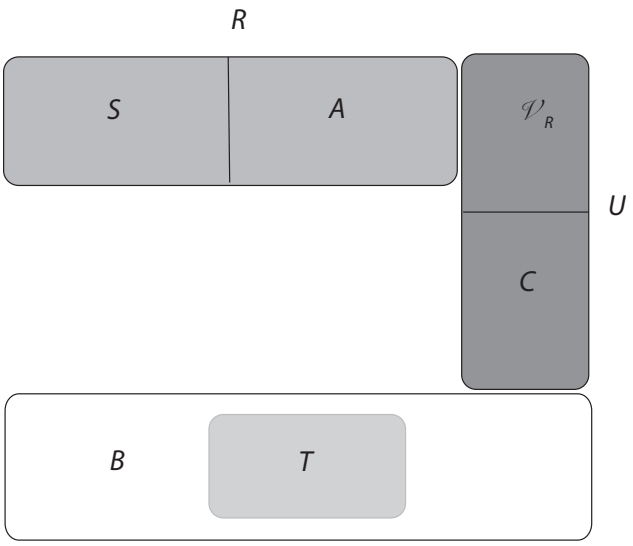

FIGURE 1. See proof of Proposition 8 for details

by (CI3), which finally implies $(X_S \perp\!\!\!\perp X_T \,|\, X_U)$ by (CI2). □

For positive probabilities, it suffices to consider singletons in Proposition 8.

PROPOSITION 9. *If the joint distribution of $(X_s, s \in V)$ is positive and, for any $s \in V$,*

$$(X_s \perp\!\!\!\perp X_{\mathcal{R}_s} \,|\, X_{\mathcal{V}_s}),$$

then X is Markov relative to G. The converse statement is true without the positivity assumption.

PROOF. By induction, it suffices to prove that, if (18) is true for S and $T \subset V$, with $T \cap S = \emptyset$, it is also true for $S \cup T$. Using this, we can extend the property, true for singletons in Proposition 9, to any subset of V.

So, let $U = \mathcal{V}_{S\cup T}$ and $R = \mathcal{R}_{S\cup T} = V \setminus (S \cup T \cup U)$. Then, we have

$$(X_S \perp\!\!\!\perp X_{\mathcal{R}_S} \,|\, X_{\mathcal{V}_S}) \Rightarrow (X_S \perp\!\!\!\perp X_R \,|\, (X_U, X_T))$$

because $R \subset \mathcal{R}_S$ (if $s \in \mathcal{V}_S$, then it is either in U or in T and therefore cannot be in R). Similarly, $(X_T \perp\!\!\!\perp X_R \,|\, (X_U, X_S))$, and (CI5) (for which we need $P > 0$) now implies $((X_T, X_S) \perp\!\!\!\perp X_R \,|\, X_U)$. □

To see that the positivity assumption is needed, consider the following example with six variables $X_1, \ldots, X_6$, and a graph linking consecutive integers and closing with an edge between 1 and 6. Assume that $X_1 = X_2 = X_4 = X_5$, and that X_1, X_3 and X_6 are independent. Then the hypotheses of Proposition 9 are true, since, for $k = 1, 2, 4, 5$, X_k is constant given its neighbors, and X_3 (resp. X_6) is independent of the rest of the variables. But (X_1, X_2) is not independent of (X_4, X_5) given the neighbors X_3, X_6.

Finally, another statement equivalent to Proposition 9 is the following:

PROPOSITION 10. *If the joint distribution of* $(X_s, s \in V)$ *is positive and, for any* $s, t \in V$,

$$s \not\sim_G t \Rightarrow (X_s \perp\!\!\!\perp X_t \,|\, X_{V\setminus\{s,t\}}),$$

then X *is Markov relative to* G. *The converse statement is true without the positivity assumption.*

PROOF. Fix $s \in V$ and assume that $(X_s \perp\!\!\!\perp X_R \,|\, X_{V\setminus R})$ for any $R \subset \mathcal{R}_s$ with cardinality at most k (the statement is true for $k = 1$ by assumption). Consider a set $\tilde{R} \subset \mathcal{R}_s$ of cardinality $k+1$, that we decompose into $R \cup \{t\}$ for some $t \in \tilde{R}$. We have $(X_s \perp\!\!\!\perp X_t \,|\, X_{V\setminus\tilde{R}}, X_R)$ from the initial hypothesis and $(X_s \perp\!\!\!\perp X_R \,|\, X_{V\setminus\tilde{R}}, X_t)$ from the induction hypothesis. Using property (CI5), this yields $(X_s \perp\!\!\!\perp X_{\tilde{R}} \,|\, X_{V\setminus\tilde{R}})$. This proves the proposition by induction. □

1.3. Remark. It is obvious from the definition of a G-Markov process that, if X_V is Markov for a graph $G = (V, E)$, it is automatically Markov for any richer graph, i.e., any graph $\tilde{G} = (V, \tilde{E})$ with $E \subset \tilde{E}$. This is because separation in $\tilde{G}$ implies separation in G. Moreover, any X_V is G-Markov for the *complete graph* on V, for which $s \sim t$ for all $s \neq t \in V$. This is because no pair of sets can be separated in a complete graph.

Any graph with respect to which X_V is Markov must be richer than the graph $G_X = (V, E_X)$ defined by $s \not\sim_{G_X} t$ if and only $(X_s \perp\!\!\!\perp X_t \,|\, X_{\{s,t\}^c})$. This is true because, for any graph G for which X is Markov, we have

$$s \not\sim_G t \Rightarrow (X_s \perp\!\!\!\perp X_t \,|\, X_{\{s,t\}^c}) \Rightarrow s \not\sim_{G_X} t.$$

Interestingly, Proposition 10 states that X_V is G_X-Markov as soon as its joint distribution is positive. This implies that G_X is the minimal graph over which X_V is Markov in this case.

1.4. Restricted Graph and Partial Evidence. Assume that some variables $X_T = (X_t, t \in T)$ (with $T \subset V$) have been observed, with observed values $x_T = (x_t, t \in T)$. One would like to use this so-called *partial evidence* to get additional information on the remaining variables, X_S where $S = V \setminus T$. From the probabilistic point of view, this means computing the conditional distribution of X_S given $X_T = x_T$.

One important property of G-Markov models is that the Markov property is essentially conserved when passing to conditional distributions. We introduce for this the following definitions.

DEFINITION 7. *If* $G = (V, E)$ *is an undirected graph, a subgraph of* G *is a graph* $G' = (V', E')$ *with* $V' \subset V$ *and* $E' \subset E$.

If $S \subset V$, *the* restricted graph, G_S, *of* G *to* S *is defined by*

$$G_S = (S, E_S) \text{ with } E_S = \{e = \{s, t\} : s, t \in S \text{ and } e \in E\}. \tag{19}$$

We have the following proposition.

PROPOSITION 11. *Let* $G = (V, E)$ *be an undirected graph and* X *be* G*-Markov. Let* $S \subset V$ *and* $T = S^c$. *Given a partial evidence* x_T *such that* $P(X_T = x_T) > 0$, X_S *conditionally to* $X_T = x_T$ *is* G_S*-Markov.*

PROOF. The proof is straightforward once it is noticed that

$$(A \perp\!\!\!\perp B \,|\, C)_{G_S} \Rightarrow (A \perp\!\!\!\perp B \,|\, C \cup T)_G$$

so that

$$\begin{aligned}(A \perp\!\!\!\perp B \,|\, C)_{G_S} &\Rightarrow (X_A \perp\!\!\!\perp X_B \,|\, X_C, X_T)_P \\ &\Rightarrow (X_A \perp\!\!\!\perp X_B \,|\, X_C)_{P(.|X_T = x_T)}\end{aligned}$$

□

1.5. Marginal Distributions. The effect of taking marginal distributions for a G-Markov model is, unfortunately, not such a mild operation as computing conditional distributions, in the sense that the conditional independence structure of the marginal distribution may be much more complex than the original one.

Let $G = (V, E)$ be an undirected graph, and let S be a subset of V. Define the graph $G^S = (S, E^S)$ by $\{s, t\} \in E^S$ if and only if $\{s, t\} \in E$ or there exists $u, u' \in S^c$ such that $\{s, u\} \in E$, $\{t, u'\} \in E$ and u and u' are connected by a path in S^c. In other terms E^S links all $s, t \in S$ that can be connected by a path, all but the extremities of which are included in S^c. With this notation, the following proposition holds.

PROPOSITION 12. *Let $G = (V, E)$ be an undirected graph, and $S \subset V$. Assume that $X_V = (X_s, s \in V)$ is a family of random variables which is G-Markov. Then $X_S = (x_s, s \in S)$ is G^S-Markov.*

PROOF. It suffices to prove that, for $A, B, C \subset S$,

$$(A \perp\!\!\!\perp B \,|\, C)_{G^S} \Rightarrow (A \perp\!\!\!\perp B \,|\, C)_G.$$

So, assume that A and B are separated by C in G^S. If a path connects A and B in G, we can, by definition of E^S, remove from this path any portion that passes in S^c and obtain a valid path in G^S. By assumption, this path must pass in C, and therefore so does the original path. □

The graph G^S can be much more complex than the restricted graph G_S introduced in the previous section (note that, by definition, G^S is richer than G_S). Take, for example, the graph that corresponds to Hidden Markov models, for which (cf. Fig, 2)

$$V = \{1, \ldots, N\} \times \{0, 1\}$$

and edges $\{s, t\} \in E$ being either $s = (k, 0)$ and $t = (l, 0)$ with $|k - l| = 1$, or $s = (k, 0)$ and $t = (k, 1)$. Let $S = \{1, \ldots, N\} \times \{1\}$. Then, G_S is totally disconnected ($E_S = \emptyset$), since no edge in G links two elements of S. In contrast, any pair of elements in S is connected by a path in S^c, so that G^S is a complete graph.

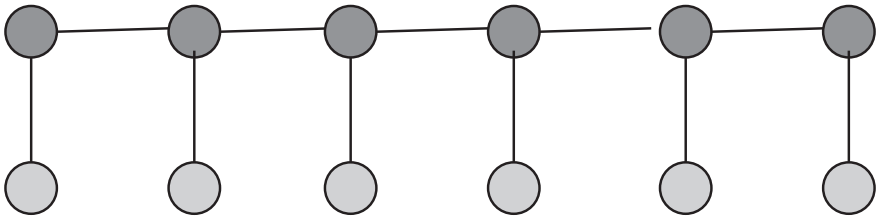

FIGURE 2. In this graph, variables in the lower row are conditionally independent given the first row, while their marginal distribution requires a completely connected graph.

2. The Hammersley-Clifford Theorem

The Hammersley-Clifford theorem, which will be proved in this section, gives a complete description of positive Markov processes relative to a given graph, G. It states that positive G-Markov models are associated to families of positive local interactions indexed by cliques in the graph. We now introduce each of these concepts.

2.1. Families of Local Interactions.

DEFINITION 8. *Let V be a set of vertices and $(F_s, s \in V)$ a collection of state spaces. A family of local interactions is a collection of non-negative functions $\Phi = (\varphi_C, C \in \mathcal{C})$ indexed over some subset $\mathcal{C}$ of $\boldsymbol{\mathcal{P}}(V)$, such that each φ_C is defined on $F_C = \prod_{s\in C} F_s$, with values in $[0, +\infty)$. (Here, $\boldsymbol{\mathcal{P}}(V)$ is the set of all subsets of V.)*

Such a family has order p if no $C \in \mathcal{C}$ has cardinality larger than p. A family of local interactions of order 2 is also called a family of pair interactions.

Such a family is said to be consistent, if there exists an $x \in F_V$ such that

$$\prod_{C\in\mathcal{C}} \varphi_C(x_C) \neq 0.$$

To a consistent family of local interactions, one associates the probability distribution $\pi = \pi_\Phi$ on F_V uniquely defined by

$$\pi(x_V) = \frac{1}{Z} \prod_{C\in\mathcal{C}} \varphi_C(x_C) \tag{20}$$

for all $x_V \in F_V$, where $Z = Z_\Phi$ is a normalizing constant.

Given a probability distribution (like π above) on F_V, one can always define a canonical process X_V such that $P(X_V = x_V) = \pi(x_V)$, by simply taking $\Omega = F_V$ and, for $\omega = x_V$, $X_s(\omega) = x_s$. We will always assume such a construction, using X_V without further definition. Given this construction, we will also refer to π as, say, a G Markov process for some graph G, as soon as the associated X_V is G-Markov.

To a given subset $\mathcal{C} \subset \boldsymbol{\mathcal{P}}(V)$, one can associate a graph $G_\mathcal{C} = (V, E_\mathcal{C})$ by letting $\{s, t\} \in E_\mathcal{C}$ if and only if there exists $C \in \mathcal{C}$ such that $\{s, t\} \in \mathcal{C}$. We then have the following proposition.

PROPOSITION 13. *Let $\Phi = (\varphi_C, C \subset \mathcal{C})$ be a consistent family of local interactions, associated to some $\mathcal{C} \subset \boldsymbol{\mathcal{P}}(V)$. Then the associated distribution π_Φ is $G_\mathcal{C}$-Markov.*

PROOF. According to Proposition 8, we must show that, for any $S \subset V$, one has

$$(X_S \perp\!\!\!\perp X_{\mathcal{R}_S} \,|\, X_{\mathcal{V}_S})$$

where $\mathcal{V}_S$ is the set of neighbors of S in $G_\mathcal{C}$ and $\mathcal{R}_S = V \setminus (\mathcal{V}_S \cup S)$. Define the set U_S by

$$U_S = \bigcup_{C\in\mathcal{C}, S\cap C\neq\emptyset} C$$

so that $\mathcal{V}_S = U_S \setminus S$ and $\mathcal{R}_S = V \setminus U_S$. To prove conditional independence, we need to prove that, for any $x_V \in F_V$:

$$\pi(x_V)\pi_{\mathcal{V}_S}(x_{\mathcal{V}_S}) = \pi_{U_S}(x_{U_S})\pi_{V\setminus S}(x_{V\setminus S}) \tag{21}$$

(where we denote π_A the marginal distribution of π on A.)

From the definition of π, we have

$$\begin{aligned}\pi(x_V) &= \frac{1}{Z}\prod_{C\in\mathcal{C}}\varphi_C(x_C)\\ &= \frac{1}{Z}\prod_{C:C\cap S\neq\emptyset}\varphi_C(x_C)\prod_{C:C\cap S=\emptyset}\varphi_C(x_C)\end{aligned}$$

The first term in the last product only depends on x_{U_S}, and the second one only on $x_{V\setminus S}$. Introduce the notation

$$\begin{cases}\mu_1(x_{\mathcal{V}_S}) = \sum_{y_{U_S}:y_{\mathcal{V}_S}=x_{\mathcal{V}_S}}\prod_{C:C\cap S\neq\emptyset}\varphi_C(x_C)\\ \mu_2(x_{\mathcal{V}_S}) = \sum_{y_{V\setminus S}:y_{\mathcal{V}_S}=x_{\mathcal{V}_S}}\prod_{C:C\cap S=\emptyset}\varphi_C(x_C)\end{cases}$$

With this notation, we have:

$$\begin{cases}\pi_{U_S}(x_{U_S}) = (\mu_2(x_{\mathcal{V}_S})/Z)\prod_{C:C\cap S\neq\emptyset}\varphi_C(x_C)\\ \pi_{V\setminus S}(x_{V\setminus S}) = (\mu_1(x_{\mathcal{V}_S})/Z)\prod_{C:C\cap S=\emptyset}\varphi_C(x_C)\\ \pi_{\mathcal{V}_S}(x_{\mathcal{V}_S}) = \mu_1(x_{\mathcal{V}_S})\mu_2(x_{\mathcal{V}_S})/Z\end{cases}$$

from which (21) can be easily obtained. □

We now discuss conditional distributions and marginals for processes associated with local interactions. Starting with conditionals, let π be associated with Φ, and let $S\subset V$ and $T = V\setminus S$. Assume that a configurations y_T is given, such that $\pi_T(y_T) > 0$, and consider the conditional distribution

$$\pi_{S|T}(x_S|y_T) = \pi(x_S\wedge y_T)/\pi_T(y_T) \tag{22}$$

where we use the notation $x_B\wedge y_C$, with $B\cap C=\emptyset$, to refer to the configuration $z\in F_{B\cup C}$ with $z_s = x_s$ if $s\in B$ and $z_s = y_s$ for $s\in C$. We have the following proposition.

PROPOSITION 14. *With the notation above, $\pi_{S|T}(\cdot|y_T)$ is associated to the family of local interactions $\Phi_{|y_T} = (\varphi_{\tilde{C}|y_T}, \tilde{C}\in\mathcal{C}_S)$ with*

$$\mathcal{C}_S = \left\{\tilde{C} : \tilde{C}\subset S, \exists C\in\mathcal{C} : \tilde{C} = C\cap S\right\}$$

and

$$\varphi_{\tilde{C}|y_T}(x_{\tilde{C}}) = \prod_{C\in\mathcal{C}:C\cap S=\tilde{C}}\varphi_C(x_{\tilde{C}}\wedge y_{C\cap T}).$$

PROOF. From (22) and the definition of π, it is easy to sees that

$$\pi_{S|T}(x_S|y_T) = \frac{1}{Z(y_T)}\prod_{C:C\cap S\neq\emptyset}\varphi_C(x_{C\cap S}\wedge y_{C\cap T}),$$

where $Z(y_T)$ is a constant that only depends on y_T. The fact that $\pi_{S|T}(\cdot|y_T)$ is associated to $\Phi_{|y_T}$ is then obtained by reorganizing the product over distinct $S\cap C$'s. □

This result, combined with Proposition 13, is consistent with Proposition 11, in the sense that the restriction to $G_{\mathcal{C}}$ to S coincides with the graph $G_{\mathcal{C}_S}$. The easy proof is left to the reader.

We now consider marginals, and more specifically marginals when only one node is removed, which is node elimination.

PROPOSITION 15. *Let π be associated to $\Phi = (\varphi_C, C \in \mathcal{C})$ as above. Let $t \in V$ and $S = V \setminus \{t\}$. Define $\mathcal{C}_t$ by $\tilde{C} \in \mathcal{C}_t$ if either $\tilde{C} \in \mathcal{C}$ and $t \notin \tilde{C}$, or*

$$\tilde{C} = \tilde{C}_t := \bigcup_{C \in \mathcal{C}: t \in C} \setminus \{t\}.$$

Define a family of local interactions $\Phi_t = (\tilde{\varphi}_{\tilde{C}}, \tilde{C} \in \mathcal{C}_t)$ by $\tilde{\varphi}_{\tilde{C}} = \varphi_{\tilde{C}}$ if $\tilde{C} \neq \tilde{C}_t$ and:

- *If $\tilde{C}_t \notin \mathcal{C}$:*

$$\tilde{\varphi}_{\tilde{C}_t} = \sum_{y_t \in F_t} \prod_{C \in \mathcal{C}, t \in C} \varphi_C(x_{C_t} \wedge y_t).$$

- *If $\tilde{C}_t \in \mathcal{C}$:*

$$\tilde{\varphi}_{\tilde{C}_t} = \varphi_{C_t}(x_{C_t}) \sum_{y_t \in F_t} \prod_{C \in \mathcal{C}, t \in C} \varphi_C(x_{C_t} \wedge y_t)$$

Then the marginal, π_S, of π over S is the distribution associated to Φ_t.

The proof is almost straightforward by summing over possible values of y_t in the expression of π and left to the reader.

2.2. Characterization of positive G-Markov processes. Using families of local interactions is a typical way to build graphical models in applications. The previous section describes a graph with respect to which the obtained process is Markov. Conversely, given a graph G, the Hammersley-Clifford theorems states that families of local interactions over the cliques of G are the only ways to build positive graphical models, which reinforces the importance of this construction. We now pass to the statement and proof of this theorem, starting with the following definition.

DEFINITION 9. *Let $G = (V, E)$ be an undirected graph. A clique in G is a nonempty subset $C \subset V$ such that $s \sim_G t$ whenever $s, t \in C$, $s \neq t$. (In particular, subsets of cardinality one are always cliques.) Cliques therefore form complete subgraphs of G.*

The set of cliques of a graph G will be denoted $\mathcal{C}_G$.

A clique that cannot be strictly included in any other clique is called a maximal clique, and their set denoted $\mathcal{C}_G^*$.

Note that some authors call cliques what we refer to as maximal cliques in these notes.

Given $G = (V, E)$, consider a family of random variables $X = X_V = (X_s, s \in V)$. We assume that X_s takes values in a finite set F_s with $P(X_s = a) > 0$ for any $a \in F_s$ (this is no loss of generality since one can always restrict F_s to such a's. If $S \subset V$, we denote $F_S = \prod_{s \in S} F_s$, so that X_S takes values in F_S. With this notation, X_V is positive, according to Definition 2, if and only if $P(X_V = x_V) > 0$ for all $x_V \in F_V$. We will let $\pi = P^X$ be the

probability distribution of X, so that $\pi(x_V) = P(X_V = x_V)$ and use as above the following notation: for $S, T \subset V$

$$\begin{cases} \pi_S(x_S) = P(X_S = x_S) \\ \pi_{S|T}(x_S|x_T) = P(X_S = x_s|X_T = x_T). \end{cases} \tag{23}$$

(For the first notation, we will simply write π if $S = V$.)

We will also need to fix a reference, or "zero", element in each F_s. We can choose it arbitrarily, and will denote it 0_s. Given this, we have the theorem:

THEOREM 1 (Hammersley-Clifford). *With the previous notation, X is a positive G-Markov process if and only if its distribution, π, is associated to a family of local interactions $\Phi = (\varphi_C, C \subset \mathcal{C}_G)$ such that $\varphi_C(x_C) > 0$ for all $x_C \in F_C$.*

Moreover, Φ is uniquely characterized by the additional constraint: $\varphi_C(x_C) = 1$ as soon as there exists $s \in C$ such that $x_s = 0_s$.

Letting $\lambda_C = -\ln \varphi_C$, we get an equivalent formulation of the theorem in terms of potentials, where a potential is defined as a family of functions

$$\Lambda = (\lambda_C, C \in \mathcal{C})$$

indexed by a subset $\mathcal{C}$ of $\boldsymbol{\mathcal{P}}(V)$, such that λ_C only depends on x_C. The distribution associated to Λ is

$$\pi(x_V) = \frac{1}{Z_\Lambda} \exp\Big(-\sum_{C \in \mathcal{C}} \lambda_C(x_C) \Big). \tag{24}$$

With this terminology, we trivially have an equivalent formulation:

THEOREM 2. *X is a positive G-Markov process if and only if its distribution, pi, is associated to a potential $\Lambda = (\lambda_C, C \subset \mathcal{C}_G)$.*

Moreover, Λ is uniquely characterized by the additional constraint: $\lambda_C(x_C) = 0$ as soon as there exists $s \in C$ such that $x_s = 0_s$.

We now prove this theorem.

PROOF. Let's start with the "if" part. If π is associated to a potential over $\mathcal{C}_G$, we have already proved that π is $G_{\mathcal{C}_G}$-Markov, so that it suffices to prove that $G_{\mathcal{C}_G} = G$, which is almost obvious: If $s \sim_G t$, then $\{s, t\} \in \mathcal{C}_G$ and $s \sim_{G_{\mathcal{C}_G}} t$ by definition of $G_{\mathcal{C}_G}$. Conversely, if $s \sim_{G_{\mathcal{C}_G}} t$, there exists $C \in \mathcal{C}_G$ such that $\{s, t\} \subset C$, which implies that $s \sim_G t$ by definition of a clique.

We now prove the "only if" part, which relies on a combinatorial lemma, which is one of Möbius's inversion formulae. Recall the notation $\boldsymbol{\mathcal{P}}(A)$ to indicate the set of all subsets of a given set A (its power set).

LEMMA 3. *Let A be a finite set and f a numeric function $f : \boldsymbol{\mathcal{P}}(A) \to \mathbb{R}$. Then there is a unique function $\lambda : \boldsymbol{\mathcal{P}}(A) \to \mathbb{R}$ such that*

$$\forall B \subset A,\ f_B = \sum_{C \subset B} \lambda_C, \tag{25}$$

and λ is given by

$$\lambda_C = \sum_{B \subset C} (-1)^{|C|-|B|} f_B. \tag{26}$$

To prove the lemma, first notice that the space $\mathcal{F}$ of functions $f : \boldsymbol{\mathcal{P}}(A) \to \mathbb{R}$ is a vector space of dimension $2^{|A|}$ and that the transformation $\varphi : \lambda \mapsto f$ with $f_B = \sum_{C \subset B} \lambda_C$ is linear. It therefore suffices to prove that, given any f, the function λ given in (26) satisfies $\varphi(\lambda) = f$, since this proves that φ is onto from $\mathcal{F}$ to $\mathcal{F}$ and therefore necessarily one to one.

So consider f and λ given by (26). Then

$$\begin{aligned}
\varphi(\lambda)(B) &= \sum_{C \subset B} \lambda_C \\
&= \sum_{C \subset B} \sum_{\tilde{B} \subset C} (-1)^{|C|-|\tilde{B}|} f_{\tilde{B}} \\
&= \sum_{\tilde{B} \subset B} \left(\sum_{C \supset \tilde{B}, C \subset B} (-1)^{|C|-|\tilde{B}|} \right) f_{\tilde{B}} \\
&= f_B
\end{aligned}$$

The last identity comes from the fact that, for any finite set $\tilde{B} \subset B, \tilde{B} \neq B$, we have

$$\sum_{C \supset \tilde{B}, C \subset B} (-1)^{|C|-|\tilde{B}|} = 0$$

(for $\tilde{B} = B$, the sum is obviously equal to 1). Indeed, if $s \in B$, $s \notin \tilde{B}$, we have

$$\begin{aligned}
\sum_{C \supset \tilde{B}, C \subset B} (-1)^{|C|-|\tilde{B}|} &= \sum_{C \supset \tilde{B}, C \subset B, s \in C} (-1)^{|C|-|\tilde{B}|} + \sum_{C \supset \tilde{B}, C \subset B, s \notin C} (-1)^{|C|-|\tilde{B}|} \\
&= \sum_{C \supset \tilde{B}, C \subset B, s \notin C} \left((-1)^{|C \cup \{s\}|-|\tilde{B}|} + (-1)^{|C|-|\tilde{B}|} \right) \\
&= 0.
\end{aligned}$$

So the lemma is proved. We now proceed to proving the existence and uniqueness statements in Theorem 2. So, assume that X is G-Markov and positive. Fix $x = x_V \in F_V$ and consider the function, defined on $\boldsymbol{\mathcal{P}}(V)$ by

$$f_B(x) = -\ln \frac{\pi(x_B \wedge 0_{B^c})}{\pi(0_V)}.$$

Then, letting

$$\lambda_C(x) = \sum_{B \subset C} (-1)^{|C|-|B|} f_B(x),$$

we have

$$f_B(x) = \sum_{C \subset B} \lambda_C(x).$$

In particular, for $B = V$, this gives

$$\pi(x_V) = \frac{1}{Z} \exp\Big(- \sum_{C \subset V} \lambda_C(x_C) \Big)$$

with $Z = P(0_V)$. We now prove that $\lambda_C(x) = 0$ if $x_s = 0_s$ for some $s \in V$ or if $C \notin \mathcal{C}_G$. This will prove (24) and the existence statement in Theorem 2.

So, assume $x_s = 0_s$. Then, for any B such that $s \notin B$, we have $f_B(x) = f_{\{s\} \cup B}(x)$. Now take C with $s \in C$. We have

$$\begin{aligned}
\lambda_C(x) &= \sum_{B \subset C, s \in B} (-1)^{|C|-|B|} f_B(x) + \sum_{B \subset C, s \notin B} (-1)^{|C|-|B|} f_B(x) \\
&= \sum_{B \subset C, s \notin B} (-1)^{|C|-|B \cup \{s\}|} f_{B \cup \{s\}}(x) + \sum_{B \subset C, s \notin B} (-1)^{|C|-|B|} f_B(x) \\
&= \sum_{B \subset C, s \notin B} ((-1)^{|C|-|B \cup \{s\}|} + (-1)^{|C|-|B|}) f_B(x) \\
&= 0.
\end{aligned}$$

Now assume that C is not a clique, and let $s \neq t \in C$ such that $s \not\sim t$. We can write, using decompositions like above,

$$\lambda_C(x) = \sum_{B \subset C \setminus \{s,t\}} (-1)^{|C|-|B|} \big(f_{B \cup \{s,t\}}(x) - f_{B \cup \{s\}}(x) - f_{B \cup \{t\}}(x) + f_B(x) \big)$$

But, for $B \subset C \setminus \{s,t\}$, we have

$$\begin{aligned}
f_{B \cup \{s,t\}}(x) - f_{B \cup \{s\}}(x) &= -\ln \frac{\pi(x_{B \cup \{s,t\}} \wedge 0_{B^c \setminus \{s,t\}})}{\pi(x_{B \cup \{s\}} \wedge 0_{B^c \setminus \{s\}})} \\
&= \ln \frac{\pi_t(x_t | x_{B \cup \{s\}} \wedge 0_{B^c \setminus \{s,t\}})}{\pi_t(0_t | x_{B \cup \{s\}} \wedge 0_{B^c \setminus \{s,t\}})}
\end{aligned}$$

and

$$\begin{aligned}
f_{B \cup \{t\}}(x) - f_B(x) &= -\ln \frac{\pi(x_{B \cup \{t\}} \wedge 0_{B^c \setminus \{t\}})}{\pi(x_B \wedge 0_{B^c})} \\
&= \ln \frac{\pi_t(x_t | x_B \wedge 0_{B^c \setminus \{t\}})}{\pi_t(0_t | x_B \wedge 0_{B^c \setminus \{t\}})}
\end{aligned}$$

So we can write

$$\lambda_C(x) = \sum_{B \subset C \setminus \{s,t\}} (-1)^{|C|-|B|} \ln \frac{\pi_t(x_t | x_{B \cup \{s\}} \wedge 0_{B^c \setminus \{s,t\}}) \pi_t(0_t | x_B \wedge 0_{B^c \setminus \{t\}})}{\pi_t(0_t | x_{B \cup \{s\}} \wedge 0_{B^c \setminus \{s,t\}}) \pi_t(x_t | x_B \wedge 0_{B^c \setminus \{t\}})}$$

which vanishes, since

$$\pi_t(x_t | x_{B \cup \{s\}} \wedge 0_{B^c \setminus \{s,t\}}) = \pi_t(x_t | x_B \wedge 0_{B^c \setminus \{t\}})$$

because $s \not\sim t$.

To prove uniqueness, note that, for any zero-normalized Λ satisfying (24), we must have $\pi(0_V) = 1/Z$ and therefore, for any x,

$$-\ln \frac{\pi(x_B \wedge 0_{B^c})}{\pi(0_V)} = \sum_{C \subset B} \lambda_C(x)$$

(extending Λ so that $\lambda_C = 0$ for $C \notin \mathcal{C}_G$). But, from Lemma 3, this uniquely defines Λ. □

The exponential form of the distribution in the Hammersley-Clifford theorem is related to what is called a Gibbs distribution in Statistical Mechanics. More precisely:

DEFINITION 10. *Let Ω be a finite set and $W : \Omega \to \mathbb{R}$ be a scalar function. The Gibbs distribution with energy W at temperature $T > 0$ is defined by*

$$\pi(x) = \frac{1}{Z_T} e^{-\frac{W(x)}{T}}$$

The normalizing constant $Z_T = \sum_{y \in \Omega} \exp(-W(y)/T)$ is called the partition function.

If $\Lambda = (\lambda_C, C \subset V)$ is a potential then its associated energy is

$$W(x) = \sum_{C \subset V} \lambda_C(x_C).$$

So the Hammersley-Clifford theorem implies that any positive G-Markov model is associated to a unique zero-normalized potential defined over the cliques of G. This representation can also be used to provide an alternate proof of Proposition 10, which is left to the reader. Finally, one can restate Proposition 14 in terms of potentials, yielding:

PROPOSITION 16. *Let P be a Gibbs distribution associated with a zero-normalized potential $\lambda = (\lambda_C, C \subset V)$. Let $S \subset V$ and $T = S^c$. Then the conditional distribution of X_S given $X_T = x_T$ is the Gibbs distribution associated with the zero-normalized potential $\tilde{\lambda} = (\tilde{\lambda}_C, C \subset S)$ where*

$$\tilde{\lambda}_C(y_S) = \sum_{C' \subset V, C' \cap S = C} \lambda_{C'}((y_S \wedge x_T)_{C'}).$$

3. Examples

3.1. Finite Markov Chains. We now review a few important examples of Markov processes X associated to specific graphs $G = (V, E)$. We will always denote by F_s the space in which X_s takes his values, for $s \in V$.

The simplest example of G-Markov process (for any graph G) is the case when $X = (X_s, s \in V)$ is a collection of independent random variables. In this case, we can take $G_X = (V, \emptyset)$, the totally disconnected graph on V. Another simple fact is that, as already remarked, any X is Markov for the complete graph $(V, \boldsymbol{\mathcal{P}}_2(V))$ where $\boldsymbol{\mathcal{P}}_2(V)$ contains all subsets of V with cardinality 2.

Beyond these trivial (but nonetheless important) cases, the simplest graph-Markov processes are those associated with linear graphs, providing finite Markov chains. For this, we let V be a finite ordered set, say

$$V = \{0, \ldots, N\}.$$

We say that $X = X_V$ is a finite Markov chain if, for any $k = 1, \ldots, N$

$$(X_k \perp\!\!\!\perp (X_0, \ldots, X_{k-2}) \,|\, X_{k-1}).$$

So we have the identity

$$P(X_0 = x_0, \ldots, X_k = x_k)P(X_{k-1} = x_{k-1}) \\ = P(X_0 = x_0, \ldots, X_{k-1} = x_{k-1})P(X_{k-1} = x_{k-1}, X_k = x_k).$$

The distribution of a Markov chain is therefore fully specified by $P(X_0 = x_0), x_0 \in F_0$ (the initial distribution) and the conditional probabilities

$$p_k(x_k|x_{k-1}) = P(X_k = x_k|X_{k-1} = x_{k-1}) \tag{27}$$

(with an arbitrary choice when $P(X_{k-1} = x_{k-1}) = 0$). Indeed, assume that $P(X_0 = x_0, \ldots, X_{k-1} = x_{k-1})$ is known (for all $x_0, \ldots, x_{k-1}$). Then, either $P(X_0 = x_0, \ldots, X_{k-1} = x_{k-1}) = 0$, in which case

$$P(X_0 = x_0, \ldots, X_k = x_k) = 0$$

for any x_k, or $P(X_0 = x_0, \ldots, X_{k-1} = x_{k-1}) > 0$, in which case, necessarily, $P(X_{k-1} = x_{k-1}) > 0$, and

$$P(X_0 = x_0, \ldots, X_k = x_k) = p_k(x_k|x_{k-1})P(X_0 = x_0, \ldots, X_{k-1} = x_{k-1}).$$

We next give the definition of a *transition probability.*

DEFINITION 11. *Let F_1 and F_2 be two finite sets; a transition probability from F_1 to F_2 is a function $p : F_2 \times F_1 \to [0, 1], (x, y) \mapsto p(x|y)$ such that, for all $y \in F_1$, the function $p(.|y)$ is a probability on F_2.*

Therefore p_k in (27) is a transition probability between F_{k-1} and F_k.

We have the following identification of a finite Markov chain with a graph-Markov process:

PROPOSITION 17. *Let $X = (X_0, \ldots, X_N)$ be a finite Markov chain, such that X is positive. Then X is G Markov for the linear graph $G = (V, E)$ with*

$$\begin{aligned} V &= \{1, \ldots, N\} \\ E &= \{\{1, 2\}, \ldots, \{N - 1, N\}\}. \end{aligned}$$

The converse is true without the positivity assumption: a G-Markov process for the graph above is always a finite Markov chain.

PROOF. We prove the direct statement (the converse one being obvious). Let s and t be nonconsecutive distinct integers, with, say, $s < t$. From the Markov chain assumption, we have

$$(X_t \perp\!\!\!\perp (X_s, X_{\{1,t-2\}\setminus\{s,\}}) \,|\, X_{t-1}),$$

which, using (CI3), yields $(X_t \perp\!\!\!\perp X_s \,|\, X_{\{1,\ldots,t-1\}\setminus\{s\}})$. Define $Y_u = X_{\{1,\ldots,u\}\setminus\{s,t\}}$: what we have proved is $(X_t \perp\!\!\!\perp X_s \,|\, Y_t)$.

We now proceed by induction and assume that $(X_t \perp\!\!\!\perp X_s \,|\, Y_u)$ for some $u \geq t$. Then, we have $(X_{u+1} \perp\!\!\!\perp (X_s, X_t, Y_{u-1}) \,|\, X_u)$, which implies (from (CI3)) $(X_{u+1} \perp\!\!\!\perp X_t \,|\, X_s, Y_u)$. Applying (CI4) to $(X_t \perp\!\!\!\perp X_s \,|\, Y_u)$ and $(X_t \perp\!\!\!\perp X_{u+1} \,|\, X_s, Y_u)$, we obtain $(X_t \perp\!\!\!\perp (X_s, X_{u+1}) \,|\, Y_u)$ and finally, $(X_t \perp\!\!\!\perp X_s \,|\, Y_{u+1})$. By induction, this gives $(X_t \perp\!\!\!\perp X_s \,|\, Y_N)$ and therefore Proposition 10 now implies that X is G-Markov.

(The proposition can also be proved as a direct consequence of the decomposition

$$P(X_0 = x_0, \dots, X_N = x_N) = P(X_0 = x_0)p_1(x_1|x_0) \dots p_N(x_N|x_{N-1}).)$$

□

3.2. Undirected Acyclic Graph models and Trees. The situation with acyclic graphs is only slightly more complex than with chains, but will require a few new definitions, including directed graphs and trees.

The difference between directed and undirected graphs is that the edges of the former are ordered pairs, namely:

DEFINITION 12. *A (finite) directed graph G is a pair $G = (V, E)$ where V is a finite set of vertices and E is a subset of*

$$V \times V \setminus \{(s, s), s \in V\},$$

which satisfies, in addition,

$$(s, t) \in E \Rightarrow (t, s) \notin E.$$

So, for directed graphs, edges (s, t) and (t, s) differ. Because of this, we can say that the edge $e = (s, t)$ stems from s and points to t. The *parents* of a vertex s are the vertices t such that $(t, s) \in E$, and its children are the vertices t such that $(s, t) \in E$. We will also use the notation $s \to_G t$ to indicate that $(s, t) \in E$ (compare to $s \sim_G t$ for undirected graphs).

DEFINITION 13. *A path in a directed graph $G = (V, E)$ is a sequence $(s_0, \dots, s_N)$ such that, for all $k = 1, \dots, N$, $s_k \to_G s_{k+1}$ (this includes one-vertex paths (s_0)). The definition is the same for undirected graph, replacing $s_k \to_G s_{k+1}$ by $s_k \sim_G s_{k+1}$. For both direct and undirect cases, one says that a path is closed if $s_0 = s_N$.*

In an undirected graph, a path is folded if it can be written as $(s_0, \dots, s_{N-1}, s_N, s_{N-1}, \dots, s_0)$.

If $G = (V, E)$ is directed, one says that $t \in V$ is a descendant of $s \in V$ (or that s is an ancestor of t) if there exists a path starting at s and ending at t. In particular, every vertex is both a descendant and an ancestor of itself.

We finally define acyclic graphs.

DEFINITION 14. *A loop in a directed (resp. an undirected) graph G is a nontrivial path $(s_0, s_1, \dots, s_{N-1})$, with $N \geq 3$, such that $s_N \to s_0$ (resp. $s_N \sim s_0$), which passes only once through $s_0, \dots, s_{N-1}$ (no self-intersection).*

A (directed or undirected) graph G is acyclic if it contains no loop.

The following property will be useful.

PROPOSITION 18. *In a directed graph, any non trivial closed path contains a loop (i.e., one can delete vertices from it to finally obtain a loop.)*

In an undirected graph, any non trivial closed path which is not a union of folded paths contains a loop.

PROOF. Take $\gamma = (s_0, s_1, \dots, s_N = s_0)$. First take the case of a directed graph. Consider the first occurrence of a repetition, i.e., the first index for which

$$s_j \in \{s_0, \dots, s_{j-1}\}.$$

Then there is a unique $j' \in \{0, \dots, j-1\}$ such that $s_{j'} = s_j$, and the path $(s_{j'}, \dots, s_{j-1})$ must be a loop (any repetition in the sequence would contradict the fact that j was the first occurrence.

Consider now the undirected case. We can remove all folded subpaths, by keeping everything but their initial point, since each such operation still provide a path at the end. Assume that this is done, still denoting the remaining path $(s_0, s_1, \dots, s_N = s_0)$. We must have $N \geq 3$ since $N = 1$ implies that the original path was a union of folded paths, and $N = 2$ provides a folded path. Let, $0 \leq j' < j$ be as in the directed case. Note that one must have $j' < j - 2$, since $j' = j - 1$ would imply an edge between j and itself and $j' = j - 2$ induces a folded subpath. But this implies that $(s_{j'}, \dots, s_{j-1})$ is a loop. □

Directed acyclic graphs (DAG) will be important for us, because they are associated to the class of probability distributions known as Bayesian networks that we will discuss later. For now, we are interested with undirected acyclic graphs and their relation to trees, which form a subclass of directed acyclic graphs as follows.

Definition 15. *A forest is a directed acyclic graph with the additional requirement that each of its vertices has at most one parent.*

A root in a forest is a vertex that has no parent. A forest with a single root is called a tree.

It is clear that a forest has at least one root, since one could otherwise describe a nontrivial cycle by starting from a any vertex and passing to its parent until the sequence self-intersects (which must happen since V is finite).

From a directed graph $G = (V, E)$, we can build an undirected graph, denoted $G^\flat = (V, E^\flat)$ by forgetting the edge ordering, namely

$$\{s, t\} \in E^\flat \Leftrightarrow (s, t) \in E \text{ or } (t, s) \in E.$$

The following proposition relates forests and undirected acyclic graphs.

Proposition 19. *If G is a forest, then $G^\flat$ is an undirected acyclic graph.*

Conversely, if G is an undirected acyclic graph, there exists a forest $\tilde{G}$ such that $\tilde{G}^\flat = G$.

Proof. Let $G = (V, E)$ be a forest and, in order to reach a contradiction, assume that $G^\flat$ has a loop, $s_0, \dots, s_{N-1}, s_N = s_0$. Assume that $(s_0, s_1) \in E$; then, also $(s_1, s_2) \in E$ (otherwise s_1 would have two parents), and this propagates to all (s_k, s_{k+1}) for $k = 0, \dots, N-1$. But, since $s_N = s_0$, this provides a loop in G which is not possible. This proves thet $G^\flat$ has no loop since the case $(s_1, s_0) \in E$ is treated similarly.

Now, let G be an undirected acyclic graph. Fix a vertex $s \in V$ and define the following recursive procedure initialized with $S_0 = \{s\}$ (the processed vertices) and $E_0 = \emptyset$ (the oriented edges).

– At step k, assume that vertices in S_k have been processed and edges in $\tilde{E}_k$ have been oriented so that (S_k, E_k) is a forest, and that $\tilde{E}_k^\flat$ is the set of edges $\{s, t\} \in E$ such that $s, t \in S_k$ (so, oriented edges at step k can only involve processed vertices).

– If $S_k = V$: stop, the proposition is proved.

– Otherwise, apply the following construction. Let F_k be the set of edges in E that contain exactly one element of S_k.

(1) If $F_k = \emptyset$, take any $s \in V \setminus S_k$ as a new root and let $S_{k+1} = S_k \cup \{s\}$, $\tilde{E}_{k+1} = \tilde{E}_k$.

(2) Otherwise, add to $\tilde{E}_k$ the oriented edges (s,t) such that $s \in S_k$ and $\{s,t\} \in F_k$, yielding $\tilde{E}_{k+1}$ and add to S_k the corresponding children (t's) yielding S_{k+1}.

We need to justify the fact that $\tilde{G}_{k+1} = (S_{k+1}, \tilde{E}_{k+1})$ above is still a forest. This is obvious after Case 1, so consider Case 2. First $\tilde{G}_{k+1}$ is acyclic, since any oriented loop is a fortiori an unoriented loop and G is acyclic. So we need to prove that no vertex in S_{k+1} has two parents. Since we did not add any parent to the vertices in S_k and, by assumption, $(S_k, \tilde{E}_k)$ is a forest, the only possibility for a vertex to have two parents in S_{k+1} is the existence of t such that there exists $s, s' \in S_k$ with $\{s,t\}$ and $\{s',t\}$ in E. But, since s and s' have unaccounted edges containing them, they cannot have been introduced in S_k before the previously introduced root has been added, so they are both connected to this root: but the two connections to t would create a loop in G which is impossible.

So the procedure carries on, and must end with $S_k = V$ at some point since we keep adding points to S_k at each step. □

Note that the previous proof shows that the orientation of a connected undirected tree into a tree is not unique, although uniquely specified once a root is chosen. It is constructive, and provides an algorithm building a forest from an undirected acyclic graph.

We now define graphical models supported by trees, which constitute our first Markov models associated to directed graphs. Define the depth of a vertex in a tree $G = (V,E)$ to be the number of edges in the unique path that links it to the root. We will denote by G_d the set of vertices in G that are at depth d, so that G_0 contains only the root, G_1 the children of the root and so on. Using this, we have the definition:

DEFINITION 16. *Let $G = (V,E)$ be a tree. A process $X_V = (x_s, s \in V)$ is G-Markov if and only, for each $d \geq 1$, and for each $s \in G_d$, we have*

$$(X_s \perp\!\!\!\perp (X_{G_d \setminus \{s\}}, X_{G_q \setminus \{s^-\}}, q < d) \,|\, X_{s^-}) \tag{28}$$

where s^- is the parent of s.

So, conditional to its parent, X_s is independent from all other variables at depth smaller or equal to the depth of s.

Note that, from (CI3), we have, for all $s \in G_d$,

$$(X_s \perp\!\!\!\perp X_{G_d \setminus \{s\}} \,|\, X_{G_q}, q < d),$$

which, using Proposition 2, implies that the variables $(X_s, s \in G_d)$ are mutually independent given $X_{G_q}, q < d$. This implies that, for $d = 1$ (letting s_0 denote the root in G):

$$P(X_{G_1} = x_{G_1}, X_{s_0} = x_{s_0}) = P(X_{s_0} = x_{s_0}) \prod_{s \in G_1} P(X_s = x_s | X_{s_0} = x_{s_0}).$$

(If $P(X_{s_0} = x_{s_0}) = 0$, the choice for the conditional probabilities can be made arbitrarily without changing the left-hand side which vanishes.) More generally, we have, letting $G_{<d} = G_0 \cup \cdots \cup G_{d-1}$,

$$P^X_{G_{\leq d}}(x_{G_{\leq d}}) = \prod_{s \in G_d} P(X_s = x_s | X_{s^-} = x_{s^-}) P^X_{G_{<d}}(x_{G_{<d}})$$

(with again an arbitrary choice for the conditional probabilities that are not defined) so that, we obtain, by induction

$$P^X(x_V) = P^X_{s_0}(x_{s_0}) \prod_{s \neq s_0} p_s(x_s|x_{s^-}) \tag{29}$$

where $p_s(x_s|x_{s^-}) := P(X_s = x_s|X_{s^-} = x_{s^-})$ are the tree transition probability between a parent and a child. So we have the following proposition.

PROPOSITION 20. *A process $X = X_V$ is Markov relative to a tree $G = (V, E)$ if and only if there exists a probability distribution p_0 on F_{s_0} and a family $(p_{st}, (s,t) \in E)$ such that p_{st} is a transition probability from F_s to F_t and*

$$P(x_V) = p_0(x_{s_0}) \prod_{(s,t)\in E} p_{st}(x_t|x_s). \tag{30}$$

We only have proved the "only if" part, but the if part is obvious from (30). Another property that becomes obvious with this expression is the first part of the following proposition.

PROPOSITION 21. *If a process $X = X_V$ is Markov relative to a tree $G = (V, E)$ then it is $G^\flat$ Markov. Conversely, if $G = (V, E)$ is an undirected acyclic graph and X is G-Markov, then X is Markov relative to any tree $\tilde{G}$ such that $\tilde{G}^\flat = G$.*

PROOF. To prove the converse part, assume that $G = (V, E)$ is undirected acyclic and that X is G-Markov. Take $\tilde{G}$ such that $\tilde{G}^\flat = G$. For $s \in V$ and its parent s^- in $\tilde{G}$, the sets $\{s\}$ and $\tilde{G}_{\leq d} \setminus \{s, s^-\}$ are separated by s^- in G. To see this, assume that there exists a $t \in \tilde{G}_{\leq d} \setminus \{s, s^-\}$ with a path from t to s that does not pass through s^-. Then we can complete this path with the path from t to the first common ancestor (in $\tilde{G}$) of t and s and back to s to create a path from s to s that passes only once through $\{s^-, s\}$ and therefore contains a loop by Proposition 18.

The G-Markov property now implies

$$(X_s \perp\!\!\!\perp (X_{\tilde{G}_d \setminus \{s\}}, X_{\tilde{G}_q \setminus \{s^-\}}, q < d) \,|\, X_{s^-})$$

which proves that X is $\tilde{G}$-Markov. □

So we see that there is no real gain in generality when passing from undirected to directed graphs when working with trees. This is an important remark, because directionality in graphs is often interpreted as causality. For example, there is a natural causal order in the statements

$$\text{(It rains)} \rightarrow \text{(Car windshields get wet)} \rightarrow \text{(Wipers are on)}$$

in the sense that each event can be seen as a logical precursor to the next one. However, because one can pass from this directed chain to an equivalent undirected chain and then back to a equivalent directed tree by choosing any of the three variables as roots, there is no way to infer, from the observation of the three events (It rains, Car windshields get wet, Wipers are on), any causality relation between them: the joint distribution cannot resolve whether wipers are on because it rains, or whether turning wipers on automatically wets windshields which in turn triggers a shower !

To infer causality relations, one needs a different kind of observation, that would modify the distribution of the system. Such an operation (called an intervention), can be done, for example, by preventing the windshields from being wet (doing, for example, the observation in a parking garage), or forcing them to be wet (using a hose). Then, one can compare observations made with these new conditions, and those made with the original system, and check, for example, whether they modified the probability that rain occurs outside. The answer (likely to be negative!) would refute any causality relation from 'windshields are wet' to 'it rains'. On the other hand, the intervention might modify how wipers are used, which would indicate a possible causal relationship from "windshields are wet" to "wipers are on".

3.3. General "Loopy" Markov Random Fields. We will see that acyclic models have very nice computational properties that make them attractive in designing distributions. However, the absence of loops is a very restrictive constraint, which is not realistic in many practical situations. Feedback effects are often needed, for example. Most models in statistical physics are supported by a lattice, in which natural translation/rotation invariance relations forbid using any non-trivial acyclic model. As an example, we now consider the 2D Ising model on a finite grid, which is a model for (anti)-ferromagnetic interaction in a spin system.

Let $G = (V, E)$. A (positive) G-Markov model is said to have only pair nainteractions if and only if can be written in the form

$$P(x) = \frac{1}{Z} \exp\Big(- \sum_{s \in G} h_s(x_s) - \sum_{\{s,t\} \in E} h_{\{s,t\}}(x_{s,t}) \Big).$$

Relating to Theorem 2, this says that P is associated to a potential involving cliques of order 2 at most (note that this does not mean that the cliques of the associated graph have order 2 at most; there can be higher-order cliques, which would therefore have a zero potential). The functions in the potential are indexed by sets, as they should be from the general definition. However, models with pair interactions are often written in the form

$$P(x) = \frac{1}{Z} \exp\Big(- \sum_{s \in G} h_s(x_s) - \sum_{\{s,t\} \in E} \tilde{h}_{st}(x_s, x_t) \Big)$$

with $\tilde{h}_{st}(\lambda, \mu) = \tilde{h}_{ts}(\mu, \lambda)$ (which is equivalent, taking $\tilde{h} = h/2$).

The *Ising model* is a special case of models with pair interactions, for which the state space, F_s is equal to $\{-1, 1\}$ for all s and

$$h_s(x_s) = \alpha_s x_s, \quad h_{\{s,t\}}(x_s, x_t) = \beta_{st} x_s x_t.$$

In fact, for binary variables, this is the most general pair interaction model.

The Ising model is moreover usually defined on a regular lattice, which, in two dimensions, implies that V is a finite rectangle in $\mathbb{Z}^2$, for example $V = \{-N, \dots, N\}^2$. The simplest choice of a translation- and 90-degree rotation-invariant graph is the nearest-neighbor graph for which $\{(i, j), (i', j')\} \in E$ if and only if $|i - i'| + |j - j'| = 1$ (see Fig. 3). With this graph, one can furthermore simplify the model to obtain the *isotropic Ising model* given by

$$\pi(x) = \frac{1}{Z} \exp\Big(- \alpha \sum_{s \in V} x_s - \beta \sum_{s \sim t} x_s x_t \Big).$$

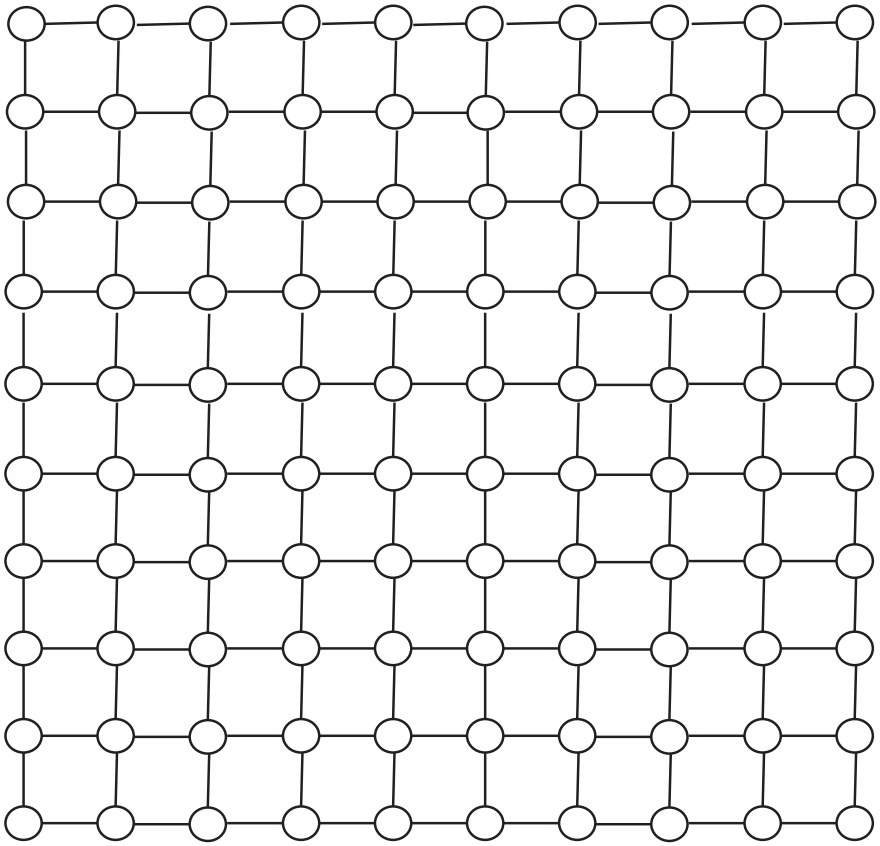

FIGURE 3. Graph forming a two-dimensional regular grid.

When $\beta < 0$, the model is *ferromagnetic*: each pair of neighbors with identical signs brings a negative contribution to the energy, making the configuration more likely (since lower energy implies higher probability).

The Potts model generalizes the Ising model to finite, but non-necessarily binary, state spaces, say $F_s = F = \{1, \ldots, n\}$. Define the function $\delta(\lambda, \mu) = 1$ is $\lambda = \mu$ and (-1) otherwise. Then the Potts model is simply given by

$$\pi(x) = \frac{1}{Z} \exp\Big(- \alpha \sum_{s \in V} h(x_s) - \beta \sum_{s \sim t} \delta(x_s, x_t) \Big)$$

for some function h defined on F.

CHAPTER 3

Probabilistic Inference

Defining probability models is only useful if they can provide a better understanding of the modeled systems, and obtain qualitative or quantitative conclusions on their behavior.

In particular, they should allow one to compute probabilities of events, or expectations of random variables of interest. For example, when building a system that describes a medical condition, in which variables like diagnosis, age, gender, clinical evidence can interact, one may want to compute, say, the probability of being both 25 years old and sick; or the probability of being sick given other observable factors. Note that, being able to compute expectations for G-Markov processes also ensures that one can compute conditional expectations, since, by Proposition 11, conditional G-Markov distributions are Markov over restricted graphs. So, our basic issue is to compute $P(X_S = x_S)$ when X is G-Markov and $S \subset V$, starting with one-vertex marginals, $P(X_s = x_s)$.

The Hammersley-Clifford theorem provides a generic form for general positive G-Markov processes, in the form

$$P_V^X(x_V) = \frac{1}{Z} \exp\Big(- \sum_{C \in \mathcal{C}_G} h_C(x_C)\Big).$$

So, formally, marginal distributions are given by the ratio

$$P_S^X(x_S) = \frac{\sum_{y \in F_V, y_S = x_S} \exp\Big(- \sum_{C \in \mathcal{C}_G} h_C(y_C)\Big)}{\sum_{y \in F_V} \exp\Big(- \sum_{C \in \mathcal{C}_G} h_C(y_C)\Big)}.$$

The problem is that the sums involved in this ratio involve a number of terms that grows exponentially with the size of V. Unless V is very small, a direct computation of these sums is intractable. An exception to this is the case of acyclic graphs, as we will see below. But for general, loopy, graphs, the sums can only be approximated, using, for example, Monte-Carlo sampling. We now review these issues, starting with the simple case of acyclic graphs.

1. Inference with Acyclic Graphs

We here consider a directed acyclic graph $G = (V, E)$. As we have seen, Markov processes for acyclic graphs are also Markov for any tree structure associated to the graph. Introducing such a tree, $\tilde{G} = (V, \tilde{E})$ with $\tilde{G}^\flat = G$, we know that a Markov process on G can be written in the form (letting s_0 denote the root in $\tilde{G}$):

$$\pi(x_V) = p_{s_0}(x_{s_0}) \prod_{(s,t) \in \tilde{E}} p_{st}(x_t | x_s) \tag{31}$$

where p_{s_0} is a probability and p_{st} a transition probability.

We now show how to compute marginal probabilities of configurations x_S, denoted $\pi_S(x_S)$, for a set $S \subset V$, starting with singletons $S = \{s\}$. The computation can be done by propagating down the tree as follows. For $s = s_0$, the probability is known, with $\pi_{s_0} = p_{s_0}$. Now take an arbitrary $s \neq s_0$ and let s^- be its parent. Then

$$\pi_s(x_s) = P(X_s = x_s) = \sum_{y_{s^-} \in F_{s^-}} P(X_s = x_s | X_{s^-} = y_{s^-}) P(x_{s^-} = y_{s^-})$$
$$= \sum_{y_{s^-} \in F_{s^-}} \pi_{s^-}(y_{s^-}) p_{s^- s}(x_s | y_{s^-})$$

so that the marginal probability at any $s \neq s_0$ can be computed given the marginal probability of its parent. We can propagate the computation down the tree, with a total cost for computing π_s proportional to $\sum_{k=1}^n |F_{t_{k-1}}| \, |F_{t_k}|$ where $t_0 = s_0, t_1, \ldots, t_n = s$ is the unique path between s_0 and s. This is linear in the depth of the tree, and quadratic (not exponential) in the sizes of the state spaces. The computation of all singleton marginals requires an order of $\sum_{(s,t) \in E} |F_s| \, |F_t|$ operations.

Now, assume that probabilities of singletons have been computed and consider an arbitrary set $S \subset V$. Let $s \in V$ be an ancestor of every vertex in S, maximal in the sense that none of its children also satisfy this property. Consider the subtrees of $\tilde{G}$ starting from each of the children of s, denoted $\tilde{G}_1, \ldots, \tilde{G}_n$ with $\tilde{G}_k = (V_k, \tilde{E}_k)$. Let $S_k = S \cap V_k$. From the conditional independence,

$$\begin{aligned} \pi_S(x_S) &= \sum_{y_s \in F_s} P(X_{S \setminus \{s\}} = x_{S \setminus \{s\}} | X_s = y_s) \pi_s(y_s) \\ &= \sum_{y_s \in F_s} \prod_{k=1, S_k \neq \emptyset}^{n} P(X_{S_k} = x_{S_k} | X_s = y_s) \pi_s(y_s) \end{aligned}$$

Now, for all $k = 1, \ldots, n$, we have $|S_k| < |S|$: this is obvious if S is not completely included in one of the V_k's. But if $S \subset V_k$ then the root, s_k, of V_k is an ancestor of all the elements in S and is a child of s, which contradicts the assumption that s is maximal. So we have reduced the computation of $\pi_S(x_S)$ to the computations of n probabilities of smaller sets, namely $P(X_{S_k} = x_{S_k} | X_s = y_s)$ for $S_k \neq \emptyset$. Because the distribution of X_{V_k} conditioned at s is a $\tilde{G}_k$-Markov model, we can reiterate the procedure until only sets of cardinality one remain, for which we know how to explicitly compute probabilities.

This provides a computationally feasible algorithm to compute marginal probabilities with trees, at least when its distribution is given in tree-form, like in (31). We now address the situation in which one starts with a form that derives from the Hammersley-Clifford theorem for the underlying acyclic graph G, that is:

$$\pi(x_V) = \frac{1}{Z} \exp\big(- \sum_{s \in V} h_s(x_s) - \sum_{\{s,t\} \in E} h_{st}(x_s, x_t) \big)$$

which is also of the form

$$\pi(x_V) = \frac{1}{Z}\prod_{s\in V}\varphi_s(x_s)\prod_{\{s,t\}\in E}\varphi_{st}(x_s,x_t). \tag{32}$$

Equation (32) is therefore based on pair interactions (cf. Definition 8), that we will assume to be consistent (but still allows for some vanishing $\psi_{st}(x_s,x_t)$).

Putting π in the form (31) is equivalent to computing all joint probability distributions $\pi_{st}(x_s,x_t)$ for $\{s,t\}\in E$, and we now describe this computation. Denote

$$U(x_V) = \prod_{s\in V}\varphi_s(x_s)\prod_{\{s,t\}\in E}\varphi_{st}(x_s,x_t)$$

so that $Z=\sum_{y_V\in F_V}U(y_V)$. For the tree $\tilde{G}=(V,\tilde{E})$, and $t\in V$, we let $\tilde{G}_t=(V_t,\tilde{E}_t)$ be the subtree of G rooted at t. For $S\subset V$, define

$$U_S(x_S) = \prod_{s\in S}\varphi_s(x_s)\prod_{\{s,s'\}\in E,s,s'\in D}\varphi_{ss'}(x_s,x_{s'})$$

and

$$Z_t(x_t) = \sum_{y_{V_t^*}\in F_{V_t^*}} U_{V_t}(x_t\wedge y_{V_t^*}).$$

with $V_t^*=V_t\setminus\{t\}$.

Lemma 4. *Let $G=(V,E)$ be a directed acyclic graph and $\pi=P^X$ be the G-Markov distribution given by* (32)*. With the notation above, we have*

$$\pi_{s_0}(x_{s_0}) = \frac{Z_{s_0}(x_{s_0})}{\sum_{y_{s_0}\in F_{s_0}}Z_{s_0}(y_{s_0})} \tag{33}$$

and, for $(s,t)\in\tilde{E}$

$$p_{st}(x_t|x_s) = P(X_t=x_t|X_s=x_s) = \frac{\varphi_{st}(x_s,x_t)Z_t(x_t)}{\sum_{y_t\in F_t}\varphi_{st}(x_s,y_t)Z_t(y_t)} \tag{34}$$

Proof. Let $W_t=V\setminus V_t$. Clearly, $Z=\sum_{x_0\in F_{s_0}}Z_{s_0}(x_0)$ and $\pi_{s_0}(x_0)=Z_{s_0}(x_0)/Z$ which gives (33). Moreover, if $s\in V$, we have

$$P(X_{V_s^*}=x_{V_s^*}|X_s=x_s) = \frac{\sum_{y_{W_s}}U(x_{V_s}\wedge y_{W_s})}{\sum_{y_{V_s^*},y_{W_s}}U(x_s\wedge y_{V_s^*}\wedge y_{W_s})}.$$

We can write

$$U(x_s\wedge y_{V_s^*}\wedge y_{W_s}) = U_{V_s}(x_s\wedge y_{V_s^*})U_{\{s\}\cup W_s}(x_s\wedge y_{W_s})\varphi_s(x_s)^{-1}$$

yielding the simplified expression

$$\begin{aligned}
&P(X_{V_s^*}=x_{V_s^*}|X_s=x_s)\\
&= \frac{U_{V_s}(x_{V_s})\varphi_s(x_s)^{-1}\sum_{y_{W_s}}U_{\{s\}\cup W_s}(x_s\wedge y_{W_s})}{\varphi_s(x_s)^{-1}\big(\sum_{y_{V_s^*}}U_{V_s}(x_s\wedge y_{V_s^*})\big)\big(\sum_{y_{W_s}}U_{\{s\}\cup W_s}(x_s\wedge y_{W_s})\big)}\\
&= \frac{U_{V_s}(x_{V_s})}{Z_t(x_s)}
\end{aligned}$$

Now, if $t_1, \ldots, t_n$ are the children of s, we have

$$U_{V_s}(x_{V_s}) = \varphi_s(x_s) \prod_{k=1}^{n} \varphi_{st_k}(x_s, x_{t_k}) \prod_{k=1}^{n} U_{V_{t_k}}(x_{V_{t_k}})$$

so that

$$\begin{aligned} & P(X_{t_k} = x_{t_k}, k = 1, \ldots, n | X_s = x_s) \\ &= \frac{1}{Z_s(x_s)} \sum_{y_{V^*_{t_k}}, k=1,\ldots,n} \varphi_s(x_s) \prod_{k=1}^{n} \varphi_{st_k}(x_s, x_{t_k}) \prod_{k=1}^{n} U_{V_{t_k}}(x_{t_k} \wedge y_{V^*_{t_k}}) \\ &= \frac{\varphi_s(x_s) \prod_{k=1}^{n} \varphi_{st_k}(x_t, x_{t_k}) \prod_{k=1}^{n} Z_{t_k}(x_{t_k})}{Z_s(x_s)} \end{aligned}$$

This implies that the transition probability needed for the tree model, $p_{st_1}(x_{t_1}|x_s)$, must be proportional to $\varphi_{st_1}(x_s, x_{t_1}) Z_{t_1}(x_{t_1})$ which proves the lemma. □

So this lemma reduces the computation of the transition probabilities to the computation of the $Z_s(x_s)$, for $s \in V$. This can be done efficiently, going upward in the tree (from terminal vertices to the root). Indeed, if s is terminal, then $V_s = \{s\}$ and $Z_s(x_s) = \varphi_s(x_s)$. Now, if s is non-terminal and $t_1, \ldots, t_n$ are its children, then, it is easy to see that

$$\begin{aligned} Z_s(x_s) &= \varphi_s(x_s) \sum_{x_{t_1} \in F_{t_1}, \ldots, x_{t_n} \in F_{t_n}} \prod_{k=1}^{n} \varphi_{st_k}(x_s, x_{t_k}) Z_{t_k}(x_{t_k}) \\ &= \varphi_s(x_s) \prod_{k=1}^{n} \Big(\sum_{x_{t_k} \in F_{t_k}} \varphi_{st_k}(x_s, x_{t_k}) Z_{t_k}(x_{t_k}) \Big) \end{aligned} \tag{35}$$

So, $Z_s(x_s)$ can be easily computed once the $Z_t(x_t)$'s are known for the children of s.

Equations (33), (34) and (35) therefore provide the necessary relations in order to compute the singleton and edge marginal probabilities on the tree. It is important to note that these relations are valid for any tree structure consistent with the acyclic graph we started with. We now rephrase them with notation that only depend on this graph and not on the selected orientation.

Let $s = \{s, t\}$ be an edge in E. Then s separates the graph $G \setminus \{s\}$ into two components. Let V_{st} be the component that contains t, and $V^*_{st} = V_{st} \setminus t$. Given this, define

$$Z_{st}(x_t) = \sum_{y_{V^*_{st}} \in F_{V^*_{st}}} U_{V_{st}}(x_t \wedge y_{V^*_{st}}).$$

This Z_{st} coincides with the previously introduced Z_t, computed with any tree in which the edge $\{s, t\}$ is oriented from s to t. Formula (35) can be rewritten with this new notation in the form:

$$Z_{st}(x_t) = \varphi_t(x_t) \prod_{t' \in \mathcal{V}_t \setminus \{s\}} \Big(\sum_{x_{t'} \in F_{t'}} \varphi_{tt'}(x_t, x_{t'}) Z_{tt'}(x_{t'}) \Big). \tag{36}$$

This equation is usually written in terms of "messages" defined by

$$m_{ts}(x_s) = \sum_{x_t \in F_t} \varphi_{st}(x_s, x_t) Z_{st}(x_t)$$

which yields

$$Z_{st}(x_t) = \varphi_t(x_t) \prod_{t' \in \mathcal{V}_t \setminus \{s\}} m_{t't}(x_t)$$

and the message consistency relation

$$m_{ts}(x_s) = \sum_{x_t \in F_t} \varphi_{st}(x_s, x_t) \varphi_t(x_t) \prod_{t' \in \mathcal{V}_t \setminus \{s\}} m_{t't}(x_t) \tag{37}$$

Also, because one can start building a tree from $G^\flat$ using any vertex as a root, Equation (33) is valid for any $s \in V$, in the form (applying (35) to the root)

$$\pi_s(x_s) = \frac{1}{Z_s} \varphi_s(x_s) \prod_{t \in \mathcal{V}_s} m_{ts}(x_s) \tag{38}$$

where Z_s is chosen to ensure that the sum of probabilities is 1. (In fact, looking at Lemma 4, we have $Z_s = Z$, independent of s.)

Similarly, Equation (34) can be written

$$p_{st}(x_t | x_s) = m_{ts}(x_s)^{-1} \varphi_{st}(x_s, x_t) \varphi_t(x_t) \prod_{t' \in \mathcal{V}_t \setminus \{s\}} m_{t't}(x_t) \tag{39}$$

which provides the edge transition probabilities. Combining this with (38), we get the edge marginal probabilities:

$$\pi_{st}(x_s, x_t) = \frac{1}{Z} \varphi_{st}(x_s, x_t) \varphi_s(x_s) \varphi_t(x_t) \prod_{t' \in \mathcal{V}_t \setminus \{s\}} m_{t't}(x_t) \prod_{s' \in \mathcal{V}_s \setminus \{t\}} m_{s's}(x_s). \tag{40}$$

We can modify (37) by multiplying the right-hand side by an arbitrary constant α_{ts} without changing the resulting estimation of probabilities: this only multiplies the messages by a constant, which cancels after normalization. This remark can be useful in particular to avoid numerical overflow; one can, for example, define $\alpha_{ts} = 1/\sum_{x_s \in F_s} m_{ts}(x_s)$ so that the messages always sum to 1. This is also useful when applying belief propagation (see next section) to loopy networks, for which (37) may diverge while the normalized version converges. An alternative normalization will also be discussed in the next section.

2. Belief Propagation and Free Energy Approximation

2.1. BP Stationarity. Equations (38) and (37) are the basic consistency equations that define the *belief propagation* algorithm. This algorithm iterates the "message passing" rule

$$m_{ts}(x_s) \leftarrow \sum_{x_t \in F_t} \varphi_{st}(x_s, x_t) \varphi_t(x_t) \prod_{t' \in \mathcal{V}_t \setminus \{s\}} m_{t't}(x_t) \tag{41}$$

given some initial choice for the messages, until stabilization. When this algorithm is run on an acyclic graph, it is easy to see that messages stabilize in finite time. Indeed, messages

starting from a terminal t (a vertex with only one neighbor) are automatically set to their correct value in (37),

$$m_{ts}(x_s) = \sum_{x_t \in F_t} \varphi_{st}(x_s, x_t)\varphi_t(x_t),$$

at the first update. These values then propagate to provide messages that satisfy (37) starting from the next-to-terminal vertices (those that have only one neighbor left when the terminals are removed) and so on. The interest of this formulation is that it does not depend on the instantiation of the acyclic graph as a tree, although correctly ordering the message updating scheme according to a tree structure will ensure a faster convergence.

For loopy graphs, the message passing rule and the corresponding equations (38) and (39) are not justified anymore, but are still well-defined in terms of the graph structure, since their expression does not assume the acyclicity of the graph. The associated *loopy belief propagation* algorithm has no guaranteed convergence, but can provide surprisingly good results in a number of applications, including the decoding of error-correcting codes, for which they provided the first method that reached quasi-optimal compression rates. We will refer to solutions provided by loopy belief propagation as BP-stationary points, as formally stated in the next definition, which allows for a possible normalization of messages, which is needed with loopy networks.

DEFINITION 17. *Let $G = (V, E)$ be an undirected graph and $\Phi = (\varphi_{st}, \{s,t\} \in E, \varphi_s, s \in V)$ a consistent family of pair interactions. We say that a family of joint probability distributions $(\pi'_{st}, \{s,t\} \in E)$ is BP-stationary for (G, Φ) if there exists messages $x_t \in F_t \mapsto m_{st}(x_t)$, constants ζ_{st} for $t \sim s$ and α_s for $s \in V$ satisfying*

$$m_{ts}(x_s) = \frac{\alpha_s}{\zeta_{ts}} \sum_{x_t \in F_t} \varphi_{st}(x_s, x_t)\varphi_t(x_t) \prod_{t' \in \mathcal{V}_t \setminus \{s\}} m_{t't}(x_t) \tag{42}$$

such that π'_{st} can be put in the form

$$\pi'_{st}(x_s, x_t) = \frac{1}{\zeta_{st}} \varphi_{st}(x_s, x_t)\varphi_s(x_s)\varphi_t(x_t) \prod_{t' \in \mathcal{V}_t \setminus \{s\}} m_{t't}(x_t) \prod_{s' \in \mathcal{V}_s \setminus \{t\}} m_{s's}(x_s). \tag{43}$$

There is no loss of generality in the definition of the normalizing constants in (42) and (43), in the sense that, if the messages satisfy (43) and

$$m_{ts}(x_s) = q_{ts} \sum_{x_t \in F_t} \varphi_{st}(x_s, x_t)\varphi_t(x_t) \prod_{t' \in \mathcal{V}_t \setminus \{s\}} m_{t't}(x_t)$$

for some constants q_{ts}, then

$$\begin{aligned} \zeta_{st} &= \sum_{x_s, x_t} \varphi_{st}(x_s, x_t)\varphi_s(x_s)\varphi_t(x_t) \prod_{t' \in \mathcal{V}_t \setminus \{s\}} m_{t't}(x_t) \prod_{s' \in \mathcal{V}_s \setminus \{t\}} m_{s's}(x_s) \\ &= \frac{1}{q_{ts}} \sum_{x_s} \varphi_s(x_s) \prod_{s' \in \mathcal{V}_s} m_{s's}(x_s) \end{aligned}$$

so that $\zeta_{st} q_{ts}$ (which has been denoted α_s) does not depend on t. Of course, the relevant questions regarding BP-stationarity is whether such π''s exist, how to compute them, and

whether $\pi'(x_s, x_t)$ provides a good approximation of the marginals of the probability distribution π that is associated to Φ, namely

$$\pi(x_V) = \frac{1}{Z} \prod_{s \in V} \varphi_s(x_s) \prod_{\{s,t\} \in E} \varphi_{st}(x_s, x_t).$$

A reassuring statement for BP-stationarity is that it is not affected when the functions in Φ are multiplied by constants, which does not affect the underlying probability π. This is stated in the next proposition.

PROPOSITION 22. *Let Φ be as above a family of edge and vertex interactions. Let $c_{st}, \{s,t\} \in E, c_s, s \in V$ be a family of positive constants, and define $\tilde{\Phi} = (\tilde{\varphi}_{st}, \tilde{\varphi}_s)$ by $\tilde{\varphi}_{st} = c_{st}\varphi_{st}$ and $\tilde{\varphi}_s = c_s\varphi_s$. Then,*

$$\pi' \text{ is BP-stationary for } (G, \Phi) \Leftrightarrow \pi' \text{ is BP-stationary for } (G, \tilde{\Phi}).$$

PROOF. Indeed, if (42) and (43) are true for (G, Φ), it suffices to replace α_s by $\alpha_s c_s$ and z_{st} by $z_{st}c_{st}c_t$ to obtain (42) and (43) for $(G, \tilde{\Phi})$. □

It is also important to notice that, if G is acyclic, Definition 17 is no more general than the message-passing rule we had considered earlier. More precisely, we have (see the remark at the end of Section 1).

PROPOSITION 23. *Let $G = (V, E)$ be undirected acyclic and $\Phi = (\varphi_{st}, \{s,t\} \in E, \varphi_s, s \in V)$ a consistent family of pair interactions. Then, the only BP-stationary distributions are the marginals of the distribution π associated to Φ.*

To conclude this section, we define what we mean by a belief propagation algorithm, in view of Definition 17.

DEFINITION 18. *Given a family of pair interactions $\Phi = (\varphi_s, \varphi_{st})$, a belief-propagation scheme (BP scheme) is any recursive algorithm that implements a message-passing rule of the form*

$$m_{ts}(x_s) \leftarrow q_{st} \sum_{x_t \in F_t} \varphi_{st}(x_s, x_t)\varphi_t(x_t) \prod_{t' \in \mathcal{V}_t \setminus \{s\}} m_{t't}(x_t) \tag{44}$$

where q_{st} is some well-defined positive function of Φ and of the current messages.

If such a scheme converges, the limit provides a BP-stationary family of pairwise distributions π', as given by (43), provided that the numerators in these expressions are not identically zero.

For example, one gets the basic BP-scheme by letting $q_{st} = 1$ for all s, t. We know that this algorithm converges for acyclic graphs, but it can diverge when loops are present. The simplest normalization takes

$$(1/q_{st}) = \sum_{x_s \in F_s} \sum_{x_t \in F_t} \varphi_{st}(x_s, x_t)\varphi_t(x_t) \prod_{t' \in \mathcal{V}_t \setminus \{s\}} m_{t't}(x_t)$$

which ensures that messages sum to 1.

2.2. Free Energy Approximations. A partial justification of the good behavior of BP with general graphs has been provided in terms of a measure introduced in statistical mechanics, called the Bethe free energy. We let $G = (V, E)$ be an undirected graph and assume that a consistent family of pair interactions is given (denoted $\Phi = (\varphi_s, s \in V, \varphi_{st}, \{s,t\} \in E)$) and consider the associated distribution, π, on F_V, given by

$$\pi(x_V) = \frac{1}{Z} \prod_{s \in V} \varphi_s(x_s) \prod_{\{s,t\} \in E} \varphi_{st}(x_s, x_t). \tag{45}$$

It will also be convenient to use the function

$$\psi_{st}(x_s, x_t) = \varphi_s(x_s)\varphi_t(x_t)\varphi_{st}(x_s, x_t)$$

so that

$$\pi(x_V) = \frac{1}{Z} \prod_{s \in V} \varphi_s(x_s)^{1-|\mathcal{V}_s|} \prod_{\{s,t\} \in E} \psi_{st}(x_s, x_t). \tag{46}$$

We will consider approximations π' of π that minimize the Kullback-Leibler divergence, $D(\pi'\|\pi)$ (see Definition 3), subject to some constraints. We can write

$$\begin{aligned} D(\pi'\|\pi) &= -E_{\pi'}(\ln \pi) - H(\pi') \\ &= -\ln Z - \sum_{s \in V} (1 - |\mathcal{V}_s|) E_{\pi'}(\ln \varphi_s) - \sum_{\{s,t\} \in E} E_{\pi'}(\ln \psi_{st}) - H(\pi') \end{aligned}$$

(where $H(\pi')$ is the entropy of π'). Introduce the one- and two-dimensional marginals of π', denoted π'_s ad π'_{st}. Then

$$\begin{aligned} D(\pi'\|\pi) = -\ln Z - \sum_{s \in V} (1 - |\mathcal{V}_s|) E_{\pi'}(\ln \frac{\varphi_s}{\pi'_s}) - \sum_{\{s,t\} \in E} E_{\pi'}(\ln \frac{\psi_{st}}{\pi'_{st}}) \\ + \sum_{s \in V} (1 - |\mathcal{V}_s|) H(\pi'_s) + \sum_{\{s,t\} \in E} H(\pi'_{st}) - H(\pi'). \end{aligned}$$

The Bethe free energy is the function F_β defined by

$$F_\beta(\pi') = -\sum_{s \in V} (1 - |\mathcal{V}_s|) E_{\pi'}(\ln \frac{\varphi_s}{\pi'_s}) - \sum_{\{s,t\} \in E} E_{\pi'}(\ln \frac{\psi_{st}}{\pi'_{st}})\,; \tag{47}$$

so that

$$D(\pi'\|\pi) = F_\beta(\pi') - \ln Z + \Delta_G(\pi')$$

with

$$\Delta_G(\pi') = \sum_{s \in V} (1 - |\mathcal{V}_s|) H(\pi'_s) + \sum_{\{s,t\} \in E} H(\pi'_{st}) - H(\pi').$$

Using this computation, one can consider the approximation problem: find $\hat{\pi}'$ that minimizes $D(\pi'\|\pi)$ over a class of distributions π' for which the computation of the first and second order marginals is easy. This problem has an explicit solution when the distribution π' is such that all variables are independent, leading to what is called the *mean field*

approximation of π. Indeed, in this case, we have

$$\Delta_G(\pi') = \sum_{\{s,t\}\in G} (H(\pi'_s) + H(\pi'_t)) + \sum_{s\in S}(1-|\mathcal{V}_s|)H(\pi'_s) - \sum_{s\in S} H(\pi'_s) = 0$$

and

$$F_\beta(\pi') = -\sum_{s\in V}(1-|\mathcal{V}_s|)E_{\pi'}(\ln\frac{\varphi_s}{\pi'_s}) - \sum_{\{s,t\}\in E} E_{\pi'}(\ln\frac{\psi_{st}}{\pi'_s\pi'_t})\,.$$

F_β must be minimized with respect to the variables $\pi'_s(x_s), s\in S, x_s\in F_S$ subject to the constraints $\sum_{x_s\in F_s}\pi'_s(x_s) = 1$. The corresponding Euler-Lagrange equations provide the mean-field consistency equations, described in the following definition.

PROPOSITION 24. *A local minimum of $F_\beta(\pi')$ over all probability distributions π' of the form*

$$\pi'(x_V) = \prod_{s\in V}\pi'_s(x_s)$$

must satisfy the mean field consistency equations:

$$\pi_s(x_s) = \frac{1}{Z_s}\varphi_s(x_s)^{1-|\mathcal{V}_s|}\prod_{t\sim s}\exp\left(E_{\pi_t}(\ln\psi_{st}(x_s,.))\right). \tag{48}$$

PROOF. Introduce Lagrange multipliers $(\lambda_s, s\in S)$ for each of the constraints. The associated Euler-Lagrange equations are

$$\frac{\partial F_\beta}{\partial\pi_s(x_s)} - \lambda_s = 0, \quad s\in S, x_s\in F_s.$$

This gives:

$$-(1-|\mathcal{V}_s|)\left(\ln\frac{\varphi_s(x_s)}{\pi_s(x_s)} - 1\right) - \sum_{t\sim s}\sum_{x_t\in F_t}\left(\ln\frac{\psi_{st}(x_s,x_t)}{\pi_s(x_s)\pi_t(x_t)} - 1\right)\pi_t(x_t) = \lambda_s.$$

Solving this with respect to $\pi_s(x_s)$ and regrouping all constant terms (independent from x_s) in the normalizing constant Z_s yields (48). □

The mean field consistency equations can be solved using a root-finding algorithm or by directly solving the minimization problem. Unfortunately, beyond the mean field theory and the approximation of π by a distribution that factorizes over all variables, there is no feasible method that would use more general approximations. The only exception is the particular case in which G is acyclic and the approximation is made by G-Markov processes. Obviously, in this case, the Kullback-Leibler distance is minimized with $\pi' = \pi$ (since π belongs to the approximating class). A slightly non-trivial remark is that π is optimal also for the minimization of the Bethe free energy F_β, because this energy coincides, up to the constant term $\ln Z$, with the Kullback-Leibler divergence, as proved by the following proposition.

PROPOSITION 25. *If G is acyclic and π' is G-Markov, then $\Delta_G(\pi') = 0$.*

This proposition is a consequence of the following lemma that has its own interest:

LEMMA 5. *If G is acyclic and π is a G-Markov distribution, then*

$$\pi(x_V) = \prod_{s\in V} \pi_s(x_s)^{1-|\mathcal{V}_s|} \prod_{\{s,t\}\in E} \pi_{st}(x_s,x_t). \tag{49}$$

PROOF OF LEMMA 5. We know that, if $\tilde{G} = (V,\tilde{E})$ is a tree such that $\tilde{G}^\flat = G$, we have, letting s_0 be the root in $\tilde{G}$

$$\begin{aligned}\pi(x_V) &= \pi_{s_0}(x_{s_0}) \prod_{(s,t)\in\tilde{E}} \pi_{t|s}(x_t|x_s) \\ &= \pi_{s_0}(x_{s_0}) \prod_{(s,t)\in\tilde{E}} (\pi_{st}(x_s,x_t)\pi(x_s)^{-1}).\end{aligned}$$

Each vertex s in V has $|\mathcal{V}_s|-1$ children in $\tilde{G}$, except s_0 which has $|\mathcal{V}_{s_0}|$ children. Using this, we get

$$\begin{aligned}\pi(x_V) &= \pi_{s_0}(x_{s_0})\pi_{s_0}(x_{s_0})^{-|\mathcal{V}_{s_0}|} \prod_{s\in V\setminus\{s_0\}} \pi_s(x_s)^{1-|\mathcal{V}_s|} \prod_{(s,t)\in\tilde{E}} \pi_{st}(x_s,x_t) \\ &= \prod_{s\in V} \pi_s(x_s)^{1-|\mathcal{V}_s|} \prod_{\{s,t\}\in E} \pi_{st}(x_s,x_t).\end{aligned}$$

□

PROOF OF PROPOSITION 25. If π' is given by (49), then

$$\begin{aligned}H(\pi') &= -E_{\pi'} \ln \pi' \\ &= -\sum_{s\in V}(1-|\mathcal{V}_s|)E_{\pi'} \ln \pi'_s - \sum_{\{s,t\}\in E} E_{\pi'} \ln \pi'_{st} \\ &= \sum_{s\in V}(1-|\mathcal{V}_s|)H(\pi'_s) + \sum_{\{s,t\}\in E} H(\pi'_{st})\end{aligned}$$

which proves that $\Delta_G(\pi') = 0$. □

In view of this, it is tempting to "generalize" the mean field optimization procedure and minimize $F_\beta(\pi')$ over all possible consistent singletons and pair marginals (π'_s and π'_{st}), and use the optimal ones as an approximation of π_s and π_{st}. What we have just proved is that this procedure provides the exact expression of the marginals when G is acyclic. For loopy graphs, however, it is not justified, and is at best an approximation. A very interesting fact is that this procedure provides the same consistency equations as belief propagation. To see this, we first start with the characterization of minimizers of F_β.

PROPOSITION 26. *Let $G = (V,E)$ be an undirected graph and π be given by* (45)*. Consider the problem of minimizing the Bethe free energy F_β in* (47) *with respect to all possible choices of probability distributions $(\pi'_{st}, \{s,t\}\in E)$, $(\pi'_s, s\in V)$ with the constraints*

$$\pi'_s(x_s) = \sum_{x_t\in F_t} \pi'_{st}(x_s,x_t), \forall x_s \in F_s \text{ and } t\sim s.$$

Then a local minimum of this problem must take the form

$$\pi'_{st}(x_s, x_t) = \frac{1}{Z_{st}} \psi_{st}(x_s, x_t) \mu_{st}(x_t) \mu_{ts}(x_s) \tag{50}$$

where the functions $\mu_{st} : F_t \to [0, +\infty)$ *are defined for all* (s,t) *such that* $\{s,t\} \in E$ *and satisfy the consistency conditions:*

$$\mu_{ts}(x_s)^{-(|\mathcal{V}_s|-1)} \prod_{s' \sim s} \mu_{s's}(x_s) = \left(\frac{e}{Z_{st}} \sum_{x_t \in F_t} \psi_{st}(x_s, x_t) \varphi_t(x_t) \mu_{st}(x_t) \right)^{|\mathcal{V}_s|-1}. \tag{51}$$

PROOF. We introduce Lagrange multipliers: $\lambda_{ts}(x_s)$ for the constraint

$$\pi'_s(x_s) = \sum_{x_t \in F_t} \pi'_{st}(x_s, x_t)$$

and γ_{ts} for

$$\sum_{x_s, x_t} \pi'_{st}(x_s, x_t) = 1,$$

which covers all constraints associated to the minimization problem. The associated Lagrangian is

$$F_\beta(\pi') - \sum_{s \in V} \sum_{x_s \in F_s} \sum_{t \sim s} \lambda_{ts}(x_s) \left(\sum_{x_t \in F_t} \pi'_{st}(x_s, x_t) - \pi'_s(x_s) \right) - \sum_{\{s,t\} \in E} \gamma_{st} \left(\sum_{x_s \in F_s, x_t \in F_t} \pi'_{st}(x_s, x_t) - 1 \right).$$

The derivative with respect to $\pi'_{st}(x_s, x_t)$ yields the condition

$$\ln \pi'_{st}(x_s, x_t) - \ln \psi_{st}(x_s, x_t) + 1 - \lambda_{ts}(x_s) - \lambda_{st}(x_t) - \gamma_{st} = 0.$$

which implies

$$\pi'_{st}(x_s, x_t) = \varphi_{st}(x_s, x_t) \exp(\gamma_{st} - 1) \exp(\lambda_{ts}(x_s) + \lambda_{st}(x_t)).$$

We let $Z_{st} = \exp(1 - \gamma_{st})$, with γ_{st} chosen so that π'_{st} is a probability. The derivative with respect to $\pi'_s(x_s)$ gives

$$(1 - |\mathcal{V}_s|)(\ln \pi'_s(x_s) - \ln \varphi_s(x_s) + 1) + \sum_{t \sim s} \lambda_{ts}(x_s) = 0.$$

Combining this with the expression just obtained for π'_{st}, we get, for $t \sim s$,

$$(1 - |\mathcal{V}_s|) \ln \sum_{x_t \in F_t} \psi_{st}(x_s, x_t) e^{\lambda_{st}(x_t)} + (1 - |\mathcal{V}_s|) \lambda_{ts}(x_s) + (1 - |\mathcal{V}_s|)(1 - \ln Z_{st} - \ln \varphi_s(x_s)) + \sum_{s' \sim s} \lambda_{s's}(x_s) = 0$$

which gives (51) with $\mu_{st} = \exp(\lambda_{st})$. □

A family π'_{st} satisfying the conditions of proposition 26 will be called Bethe-stationary. A very interesting remark states that Bethe-stationarity is in fact equivalent to BP-stationarity, as stated below.

PROPOSITION 27. *Let $G = (V, E)$ be an undirected graph and $\Phi = (\varphi_{st}, \{s,t\} \in E, \varphi_s, s \in V)$ a consistent family of pair interactions. Then a family π' of joint probability distributions is BP-stationary if and only if it is Bethe-stationary.*

PROOF. First assume that π' is BP-stationary with messages m_{st}, so that (42) and (43) are satisfied. Take

$$\mu_{st} = a_t \prod_{t' \in \mathcal{V}_t, t' \neq s} m_{t't}(x_t)$$

for some constant a_t that will be determined later. Then, the left-hand side of (51) is

$$\begin{aligned}
&\mu_{ts}(x_s)^{-(|\mathcal{V}_s|-1)} \prod_{s' \in \mathcal{V}_s} \mu_{s's}(x_s) \\
&= a_s \left(\prod_{s' \in \mathcal{V}_s, s' \neq t} m_{s's}(x_s) \right)^{-(|\mathcal{V}_s|-1)} \prod_{s' \in \mathcal{V}_s} \prod_{s'' \in \mathcal{V}_s, s'' \neq s'} m_{s''s}(x_s) \\
&= a_s m_{ts}(x_s)^{|\mathcal{V}_s|-1}.
\end{aligned}$$

The right-hand side is equal to (using (42))

$$\left(\frac{e a_t \zeta_{st}}{Z_{st} \alpha_s} m_{ts}(x_s) \right)^{|\mathcal{V}_s|-1},$$

so that we need to have

$$a_s = \left(\frac{e a_t \zeta_{st}}{Z_{st} \alpha_s} \right)^{|\mathcal{V}_s|-1}.$$

We also need

$$Z_{st} = \sum_{x_s, x_t} \psi_{st}(x_s, x_t) \mu_{st}(x_t) \mu_{ts}(x_s) = a_s a_t \zeta_{st}.$$

Solving these equations, we find that (50) and (51) are satisfied with

$$\begin{cases} a_s = (e/\alpha_s)^{(|\mathcal{V}_s|-1)/|\mathcal{V}_s|} \\ Z_{st} = \zeta_{st} a_s a_t \end{cases}$$

which proves that π' is Bethe-consistent.

Conversely, take a Bethe-consistent π', and μ_{st}, Z_{st} satisfying (50) and (51). For s such that $|\mathcal{V}_s| > 1$, define, for $t \in \mathcal{V}_s$,

$$m_{ts}(x_s) = \mu_{ts}(x_s)^{-1} \prod_{s' \sim s} \mu_{s's}(x_s)^{1/(|\mathcal{V}_s|-1)}. \tag{52}$$

Define also, for $|\mathcal{V}_s| > 1$,

$$\rho_{ts}(x_s) = \prod_{s' \in \mathcal{V}_s, s' \neq t} m_{s's}(x_s).$$

(If $|\mathcal{V}_s| = 1$, take $\rho_{ts} \equiv 1$.) Using (52), we find $\rho_{ts} = \mu_{ts}$ when $|\mathcal{V}_s| > 1$, and this identity is still valid when $|\mathcal{V}_s| = 1$, since in this case, (51) implies that $\mu_{ts}(x_s) = 1$.

We need to find constants α_t and ζ_{st} such that (42) and (43) are satisfied. But (43) implies

$$\zeta_{ts} = \sum_{x_t, x_s} \psi_{st}(x_s, x_t) \rho_{st}(x_t) \rho_{ts}(x_t)$$

and (50) implies $\zeta_{ts} = Z_{ts}$.

We now consider (42), which requires

$$m_{ts}(x_s) = \frac{\alpha_s}{\zeta_{st}} \sum_{x_t} \varphi_{st}(x_s, x_t)\varphi_t(x_t)\rho_{st}(x_t).$$

It is now easy to see that this identity to the power $|\mathcal{V}_s| - 1$ coincides with (51) as soon as one takes $\alpha_s = e$. Take $\alpha_s = e$ also when $|\mathcal{V}_s| = 1$ and define m_{ts} so that (42) is satisfied. □

3. Computing the Most Likely Configuration

3.1. Acyclic Graphs. We now address the problem of finding the configuration that maximizes $\pi(x_V)$. This problem turns out to be very similar to the computation of marginals, that we have considered so far, and we will obtain similar algorithms.

Assume that G is undirected and acyclic and that π can be written as

$$\pi(x_V) = \frac{1}{Z} \prod_{\{s,t\}\in E} \varphi_{st}(x_s, x_t) \prod_{s\in V} \varphi_s(x_s).$$

Since we want to maximize $\pi(x_V)$, the constant Z has no influence on the solution, and we want to maximize

$$U(x_V) = \prod_{\{s,t\}\in E} \varphi_{st}(x_s, x_t) \prod_{s\in V} \varphi_s(x_s). \tag{53}$$

Assume that a root has been chosen in G, with the resulting edge orientation yielding a tree $\tilde{G} = (V, \tilde{E})$ such that $\tilde{G}^\flat = G$. We partially order the vertices according to $\tilde{G}$, writing $s \le t$ if there exists a path from s to t in $\tilde{G}$ (s is an ancestor of t). Let V_s^+ contain all $t \in V$ with $t \ge s$, and define

$$U_s(x_{V_s^+}) = \prod_{\{t,u\}\in E_{V_s^+}} \varphi_{tu}(x_t, x_u) \prod_{t>s} \varphi_t(x_t).$$

and

$$U_s^*(x_s) = \max\left\{U_s(y_{V_s^+}), y_s = x_s\right\}. \tag{54}$$

Since we can write

$$U_s(x_{V_s^+}) = \prod_{t\in s^+} \varphi_{st}(x_s, x_t)\varphi_t(x_t)U_t(x_{V_t^+}), \tag{55}$$

we have

$$\begin{aligned} U_s^*(x_s) &= \max_{x_t, t\in s^+} \left(\prod_{t\in s^+} \varphi_t(x_t)\varphi_{st}(x_s, x_t)U_t^*(x_t) \right) \\ &= \prod_{t\in s^+} \max_{x_t\in F_t} (\varphi_t(x_t)\varphi_{st}(x_s, x_t)U_t^*(x_t)) \end{aligned} \tag{56}$$

This provides an iterative method to compute $U_s^*(x_s)$ for all s, starting with the leaves and progressively updating the parents. (When s is a leaf, $U_s^*(x_s)$ is directly given by $U_s^*(x_s) = 1$.)

Once all $U_s^*(x_s)$ have been computed, it is possible to obtain a configuration x^* that maximizes π. This is because an optimal configuration must satisfy $U_s^*(x_s^*) = U_s(x_{V_s^+}^*)$ for

all $s \in V$, i.e., $x^*_{V_s^+ \setminus \{s\}}$ must solve the maximization problem in (54). But because of (55), we can separate this problem over the children of s and obtain the fact that, it $t \in s^+$,

$$x_t^* = \operatorname*{argmax}_{x_t} \left(\varphi_t(x_t)\varphi_{st}(x_s^*, x_t)U_t^*(x_t)\right).$$

This procedure can be rewritten in a slightly different form using messages similar to the belief propagation algorithm. It $s \in t^+$, define

$$\mu_{st}(x_t) = \max_{x_s \in F_s}(\varphi_t(x_t)\varphi_{ts}(x_t, x_s)U_s^*(x_s))$$

and

$$\xi_{st}(x_t) = \operatorname*{argmax}_{x_s \in F_s}(\varphi_t(x_t)\varphi_{ts}(x_t, x_s)U_s^*(x_s)).$$

Using (56), we get

$$\begin{aligned}
\mu_{st}(x_t) &= \max_{x_s \in F_s} \left(\varphi_{ts}(x_t, x_s)\varphi_s(x_s) \prod_{u \in s^+} \mu_{us}(x_s)\right), \\
\xi_{st}(x_t) &= \operatorname*{argmax}_{x_s \in F_s} \left(\varphi_{ts}(x_t, x_s)\varphi_s(x_s) \prod_{u \in s^+} \mu_{us}(x_s)\right).
\end{aligned}$$

An optimal configuration can now be computed using $x_t^* = \xi_{ts}(x_s^*)$, with $s \in t^-$.

This resulting algorithm therefore first operates upwards on the tree (from leaves to root) to compute the μ_{st}'s and ξ_{st}'s, then downwards to compute x_V^*. This is summarized in the following proposition.

PROPOSITION 28. *A most likely configuration for*

$$\pi(x_V) = \frac{1}{Z} \prod_{\{s,t\} \in E} \varphi_{st}(x_s, x_t) \prod_{s \in V} \varphi_s(x_s).$$

can be computed after iterating the following updates, based on any acyclic orientation of G:

1. *Compute, from leaves to root:*

$$\mu_{st}(x_t) = \max_{x_s \in F_s} \left(\varphi_{ts}(x_t, x_s)\varphi_s(x_s) \prod_{u \in s^+} \mu_{us}(x_s)\right)$$

and $\xi_{st}(x_t) = \operatorname*{argmax}_{x_s \in F_s} \left(\varphi_{ts}(x_t, x_s)\varphi_s(x_s) \prod_{u \in s^+} \mu_{us}(x_s)\right)$.

2. *Compute, from root to leaves: $x_t^* = \xi_{ts}(x_s^*)$, with $s \in t^-$.*

Like with the computation of marginals, this algorithm can be rewritten in an orientation-independent form. The main remark is that the value of $\mu_{st}(x_t)$ does not depend on the tree orientation, as long as it is chosen such that $s \in t^+$, i.e., the edge $\{s, t\}$ is oriented from t to s. This is because such a choice uniquely prescribes the orientation of the edges of the descendants of s for any such tree, and μ_{st} only depends on this structure. Since the same

remark holds for ξ_{st}, this provides a definition of these two quantities for any pair s, t such that $\{s,t\} \in E$. The updating rule now becomes

$$\mu_{st}(x_t) = \max_{x_s \in F_s} \left(\varphi_{ts}(x_t, x_s)\varphi_s(x_s) \prod_{u \in \mathcal{V}_s \setminus \{t\}} \mu_{us}(x_s) \right), \tag{57}$$

$$\xi_{st}(x_t) = \operatorname*{argmax}_{x_s \in F_s} \left(\varphi_{ts}(x_t, x_s)\varphi_s(x_s) \prod_{u \in \mathcal{V}_s \setminus \{t\}} \mu_{us}(x_s) \right) \tag{58}$$

with $x_t^* = \xi_{ts}(x_s^*)$ for any pair $s \sim t$. Like with the m_{ts} in the previous section, looping over updating all μ_{ts} in any order will finally stabilize to their correct values, although, if an orientation is given, going from leaves to roots is obviously more efficient.

The previous analysis is not valid for loopy graphs, but, like with belief propagation, (57) and (58) provide well defined iterations when G is an arbitrary undirected graph, and can therefore be used as such, without any guaranteed behavior.

4. General Sum-Prod and Max-Prod Algorithms

4.1. Factor Graphs. The expressions we obtained for message updating with belief propagation and with mode determination respectively took the form

$$m_{ts}(x_s) \leftarrow \sum_{x_t \in F_t} \varphi_{st}(x_s, x_t)\varphi_t(x_t) \prod_{t' \in \mathcal{V}_t \setminus \{s\}} m_{t't}(x_t)$$

and

$$\mu_{ts}(x_s) \leftarrow \max_{x_t \in F_t} \left(\varphi_{st}(x_s, x_t)\varphi_t(x_t) \prod_{t' \in \mathcal{V}_t \setminus \{s\}} \mu_{t't}(x_t) \right).$$

They first one is often referred to as the "Sum-Prod" update rule, and the second as the "Max-Prod". In our construction, the sum-prod algorithm provided us with a method computing

$$\sigma_s(x_s) = \sum_{y_{V \setminus \{s\}}} U(x_s \wedge y_{V \setminus \{s\}})$$

with

$$U(x) = \prod_s \varphi_s(x_s) \prod_{\{s,t\} \in E} \varphi_{st}(x_s, x_t).$$

Indeed, we have, according to (38)

$$\sigma_s(x_s) = \varphi_s(x_s) \prod_{t \in \mathcal{V}_s} m_{ts}(x_s).$$

Similarly, the max-prod algorithm computes

$$\rho_s(x_s) = \max_{y_{V \setminus \{s\}}} U(x_s \wedge y_{V \setminus \{s\}})$$

via the relation

$$\rho_s(x_s) = \varphi_s(x_s) \prod_{t \in \mathcal{V}_s} \mu_{ts}(x_s).$$

We now discuss generalizations of these algorithms to situations in which the function U does not decompose as a product of bivariate functions. More precisely, let $\mathcal{S}$ be a subset of $\boldsymbol{\mathcal{P}}(V)$, and assume the decomposition

$$U(x) = \prod_{C \subset \mathcal{S}} \varphi_C(x_C).$$

The previous algorithms can be generalized using the concept of *factor graphs* associated to the decomposition. The vertices of this graph are either indices $s \in V$ or sets $C \in \mathcal{S}$, and edges are only between indices and sets that contain them. The formal definition is as follows.

DEFINITION 19. *Let V be a finite set of indices and $\mathcal{S}$ a subset of $\boldsymbol{\mathcal{P}}(V)$. The factor graph associated to V and $\mathcal{S}$ is the graph $G = (V \cup \mathcal{S}, E)$, E being constituted of all pairs $\{s, C\}$ with $C \in \mathcal{S}$ and $s \in C$.*

We assign the variable x_s to a vertex $s \in V$ of the factor graph, and the function φ_C to $C \in \mathcal{S}$. With this in mind, the sum-prod and max-prod algorithms are extended to factor graphs as follows.

DEFINITION 20. *Let $G = (V \cup \mathcal{S}, E)$ be a factor graph, with associated functions $\varphi_C(x_C)$. The sum-prod algorithm on G updates messages $m_{sC}(x_s)$ and $m_{Cs}(x_s)$ according to the rules*

$$\begin{cases} m_{sC}(x_s) \leftarrow \prod_{\tilde{C}, s \in \tilde{C}, \tilde{C} \neq C} m_{\tilde{C}s}(x_s) \\ m_{Cs}(x_s) \leftarrow \sum_{y_C : y_s = x_s} \varphi_C(y_C) \prod_{t \in C \setminus \{s\}} m_{tC}(y_t) \end{cases} \tag{59}$$

Similarly, the max-prod algorithm iterates

$$\begin{cases} \mu_{sC}(x_s) \leftarrow \prod_{\tilde{C}, s \in \tilde{C}, \tilde{C} \neq C} \mu_{\tilde{C}s}(x_s) \\ \mu_{Cs}(x_s) \leftarrow \max_{y_C : y_s = x_s} \varphi_C(y_C) \prod_{t \in C \setminus \{s\}} \mu_{tC}(y_t) \end{cases} \tag{60}$$

These algorithms reduce to the original ones when only single vertex and pair interactions exist. Let's check this with sum-prod. In this case, the set $\mathcal{S}$ contains all singletons $C = \{s\}$, with associated function φ_s, and all edges $\{s, t\}$ with associated function φ_{st}. We have links between s and $\{s\}$ and s and $\{s, t\} \in E$. For singletons, we have

$$m_{s\{s\}}(x_s) \leftarrow \prod_{t \sim s} m_{s\{s,t\}}(x_s) \text{ and } m_{\{s\}s}(x_s) \leftarrow \varphi_s(x_s).$$

For pairs,

$$m_{s\{s,t\}}(x_s) \leftarrow \varphi_s(x_s) \prod_{\tilde{t} \in \mathcal{V}_s \setminus \{t\}} m_{\{s,\tilde{t}\}s}(x_s)$$

and

$$m_{\{s,t\}s}(x_s) \leftarrow \sum_{y_t} \varphi_{st}(x_s, y_t) m_{t\{s,t\}}(y_t)$$

and, combining the last two assignments, it becomes clear that we retrieve the initial algorithm with $m_{\{s,t\}s}$ taking the role of what we previously denoted m_{ts}.

The important question, obviously, is whether the algorithms converge. The following result shows that this is true when the factor graph is acyclic.

PROPOSITION 29. *Let $G = (V \cup \mathcal{S}, E)$ be a factor graph with associated functions φ_C. Assume that G is acyclic. Then the sum-prod and max-prod algorithms converge in finite time.*

After convergence, we have $\sigma_s(x_s) = \prod_{C,s\in C} m_{Cs}(x_s)$ and $\rho_s(x_s) = \prod_{C,s\in C} \mu_{Cs}(x_s)$.

PROOF. Let's assume that G is connected, which is without loss of generality, since the argument can be applied to each component of G separately. Since G is acyclic, we can arbitrarily select one of its vertices as a root to form a tree. This being done, we can see that the messages going upward in the tree (from children to parent) progressively stabilize, starting with leaves. Leaves in the factor graph indeed are either singletons, $C = \{s\}$, or vertices $s \in V$ that belong to only one set $C \in \mathcal{S}$. In the first case, the algorithm imposes (taking, for example, the sum-prod case) $m_{\{s\}s}(x_s) = \varphi_s(x_s)$, and in the second case $m_{sC} = 1$. So the messages sent upward by the leaves are fixed at the first step. Since the messages going from a child to its parents only depend on the messages that it received from its other neighbors in the acyclic graph, which are its children in the tree, it is clear that all upward messages progressively stabilize until the root is reached. Once this is done, messages propagate downward from each parent to its children. This stabilizes as soon as all incoming messages to the parent are stabilized, since outgoing messages only depend on those. At the end of the upward phase, this is true for the root, which can then send its stable message to its children. The children now have all their incoming messages and can now send their messages to their own children and so on down to the leaves.

We now consider the second statement. Let's proceed by induction assuming that the result is true for any smaller graph than the one considered. Let s_0 be the root, and consider all vertices $s \neq s_0$ such that there exists $C_s \in \mathcal{S}$ such that s_0 and s both belong to C_s. Given s, there cannot be more than one such C_s since this would create a loop in the graph. For each such s, consider the part G_s of G containing all descendants of s. Let V_s be the set of vertices among the descendants of s and $\mathcal{C}_s$ the set of C's below s. Define

$$U_s(x_{V_s}) = \prod_{C\in\mathcal{C}_s} \varphi_C(x_C).$$

Since the upward phase of the algorithm does not depend on the ancestors of s, the messages incoming to s for the sum-prod algorithm restricted to G_s are the same as with the general algorithm, so that, using the induction hypothesis

$$\sum_{y_{V_s}, y_s = x_s} U_s(y_{V_s}) = \prod_{C\in\mathcal{C}_s, s\in C} m_{Cs}(x_s) = m_{sC_s}(x_s).$$

Now let $C_1, \ldots, C_n$ be the sets $C \in \mathcal{C}$ that contain s_0, which must be non-intersecting (excepted at $\{s_0\}$), again not to create loops. Write

$$C_1 \cup \cdots \cup C_n = \{s_0, s_1, \ldots, s_q\}.$$

Then, we have

$$U(x_V) = \prod_{j=1}^{n} \varphi_{C_j}(x_{C_j}) \prod_{i=1}^{q} U_{s_i}(x_{V_{s_i}})$$

and

$$\begin{aligned}
\sigma_{s_0}(x_{s_0}) &= \sum_{y_V : y_{s_0} = x_{s_0}} \prod_{j=1}^{n} \varphi_{C_j}(y_{C_j}) \prod_{i=1}^{q} U_{s_i}(y_{V_{s_i}}) \\
&= \sum_{y_S : y_{s_0} = x_{s_0}} \prod_{j=1}^{n} \varphi_{C_j}(y_{C_j}) \prod_{i=1}^{q} m_{s_i C_{s_i}}(y_{s_i}) \\
&= \prod_{j=1}^{n} \sum_{y_{C_j} : y_{s_0} = x_{s_0}} \varphi_{C_j}(y_{C_j}) \prod_{s \in C_j \setminus \{s_0\}} m_{s C_s}(y_s) \\
&= \prod_{j=1}^{n} m_{C_j s_0}(x_{s_0})
\end{aligned}$$

which proves the required result (note that, when factorizing the sum, we have used the fact that the sets $C_j \setminus \{s_0\}$ are non intersecting). An almost identical argument holds for the max-prod algorithm. □

Note that these algorithms are not always feasible. For example, it is always possible to represent a function U on F_V with the trivial factor graph in which $\mathcal{S} = \{V\}$ and E contains all $\{s, V\}, s \in V$ (using $\varphi_V = U$), but computing m_{Vs} is identical to directly computing σ_s with a sum over all configurations on $V \setminus \{s\}$ which grows exponentially. In fact, the complexity of the sum-prod and max-prod algorithms is exponential in the size of the largest C in $\mathcal{S}$ which should therefore remain small.

It is not always possible to decompose a function so that the resulting factor graph is acyclic with small degree (maximum number of edges per vertex). Sum-prod and max-prod can still be used with loopy networks, sometimes with excellent results, but without theoretical support.

One can sometimes transform a given factor graph into an acyclic one by grouping vertices. Assume that the set $\mathcal{S} \subset \boldsymbol{\mathcal{P}}(V)$ is given. We will say that a partition $\Delta = (D_1, \dots D_k)$ of V is $\mathcal{S}$-admissible if, for any $C \in \mathcal{S}$ and any $j \in \{1, \dots, k\}$, one has either $D_j \cap C = \emptyset$ or $D_j \subset C$.

If Δ is $\mathcal{S}$-admissible, one can define a new factor graph $\tilde{G}$ as follows. We first let $\tilde{V} = \{1, \dots, k\}$. To define $\tilde{\mathcal{S}} \subset \boldsymbol{\mathcal{P}}(\tilde{V})$ assign to each $C \in \mathcal{S}$ the set J_C of indices j such that $D_j \subset C$. From the admissibility assumption,

$$C = \bigcup_{j \in J_C} D_j, \tag{61}$$

so that $C \mapsto J_C$ is one-to-one. Let $\tilde{\mathcal{S}} = \{J_C, C \in \mathcal{S}\}$. Group variables using $\tilde{x}_k = x_{D_k}$, so that $\tilde{F}_k = F_{D_k}$. Define $\tilde{\Phi} = (\tilde{\varphi}_{\tilde{C}}, \tilde{C} \in \tilde{\mathcal{S}})$ by $\tilde{\varphi}_{\tilde{C}} = \varphi_C$ where C is given by (61).

In other terms, one groups variables $(x_s, s \in V)$ together, to create a simpler factor graph, which may be acyclic even if the original one was not. For example, if $V = \{a, b, c, d\}$, $\mathcal{S} = \{A, B\}$ with $A = \{a, b, c\}$ and $B = \{b, c, d\}$, then (A, c, B, b) is a cycle in the associated factor graph. If, however, one takes $D_1 = \{a\}$, $D_2 = \{b, c\}$ and $D_3 = \{d\}$, then (D_1, D_2, D_3) is $\mathcal{S}$-admissible and the associated factor graph is acyclic. In fact, in such a case, the resulting

factor graph, considered as a graph with vertices given by subsets of V, is a special case of junction tree, which is defined in the next section.

4.2. Junction Trees. We consider a probability distribution written in the form

$$\pi(x) = \frac{1}{Z} \prod_{C \in \mathcal{S}} \varphi_C(x_C)$$

and we now make the assumption that $\mathcal{S}$ can be organized as a junction tree, which is defined as follows.

DEFINITION 21. *Let V be a finite set. A junction tree on V is an undirected acyclic graph $\mathbb{G} = (\mathcal{S}, \mathbb{E})$ where $\mathcal{S} \subset \boldsymbol{\mathcal{P}}(V)$ is a family of subsets of V that satisfy the following property, called the running intersection constraint: if $C, C' \in \mathcal{S}$ and $s \in C \cap C'$, then all sets C'' in the (unique) path connecting C and C' in $\mathbb{G}$ must also contain s.*

Let us check that the clustered factor graph defined in the end of the previous section is equivalent to a junction tree when acyclic. Using the same notation, let $\hat{\mathcal{S}} = \{D_1, \dots, D_k\} \cup \mathcal{S}$, removing if needed the $C \in \mathcal{S}$ that coincide with one of the D_j's. Place an edge between D_j and C if and only if $D_j \subset C$.

Let $(C_1, D_{i_1}, \dots, D_{i_{n-1}}, C_n)$ be a path in that graph. Assume that $s \in C_1 \cap C_2$. Let D_{i_n} be the unique D_j that contains s. It is such that from the the admissibility assumption, $D_{i_n} \subset C_1$ and $D_{i_n} \subset C_n$, which implies that $(C_1, D_{i_1}, \dots, C_n, D_{i_n}, C_1)$ is a path in $\mathbb{G}$. Since $\mathbb{G}$ is acyclic, this path must be a union of folded paths. But it is easy to see that any folded satisfies the running intersection constraint. (Note that there was no loss of generality in assuming that the path started and ended with a "C", since any "D" must be contained in the C that follows or precedes it.)

We now describe how belief propagation can be extended to junction trees. Fixing a root $C_0 \in \mathcal{S}$, we first choose an orientation on $\mathbb{G}$, which induces as usual a partial order on $\mathcal{S}$. For $C \in \mathcal{S}$, define $\mathcal{S}_C^+$ as the set of all $B \in \mathcal{S}$ such that $B > C$. Define also

$$V_C^+ = \bigcup_{B \in \mathcal{S}_C^+} B.$$

We want to compute sums

$$\sigma_C(x_C) = \sum_{y_{V \setminus C}} U(x_C \wedge y_{V \setminus C}).$$

We have

$$\sigma_C(x_C) = \sum_{y_{V \setminus C}} \varphi_C(x_C) \prod_{B \in \mathcal{S} \setminus \{C\}} \varphi_B(x_{B \cap C} \wedge y_{B \setminus C}).$$

Define

$$\sigma_C^+(x_C) = \sum_{y_{V_C^+ \setminus C}} \prod_{B > C} \varphi_B(x_{B \cap C} \wedge y_{B \setminus C}).$$

Note that we have $\sigma_{C_0} = \varphi_{C_0}\sigma^+_{C_0}$ at the root. We have the recursion formula

$$\begin{aligned}\sigma^+_C(x_C) &= \sum_{y_{V^+_C\setminus C}} \prod_{C\to B} \left(\varphi_B(x_{B\cap C}\wedge y_{B\setminus C}) \prod_{B'>B} \varphi_{B'}(x_{B'\cap C}\wedge y_{B'\setminus C})\right)\\ &= \prod_{C\to B} \sum_{y_{B\cup V^+_B\setminus C}} \varphi_B(x_{B\cap C}\wedge y_{B\setminus C}) \prod_{B'>B} \varphi_{B'}(x_{B'\cap C}\wedge y_{B'\setminus C})\\ &= \prod_{C\to B} \sum_{y_{B\setminus C}} \varphi_B(x_{B\cap C}\wedge y_{B\setminus C})\sigma^+_B(x_{B\cap C}\wedge y_{B\setminus C}).\end{aligned}$$

The inversion between the sum and product in the second equation above was possible because the sets $B\cup V^+_B\setminus C$, $C\to B$ are disjoint. Indeed, if there existed B, B' such that $C\to B$ and $C\to B'$, and descendants C' of B' and C'' of B'' with a non-empty intersection, then this intersection must be included in every set in the (non-oriented) path connecting C' and C'' in $\mathbb{G}$. Since this path contains C, the intersection must also be included in C, so that the sets $B\cup V^+_B\setminus C$, with $C\to B$ are disjoint.

Introduce messages

$$m^+_B(x_C) = \sum_{y_{B\setminus C}} \varphi_B(x_{B\cap C}\wedge y_{B\setminus C})\sigma^+_B(x_{B\cap C}\wedge y_{B\setminus C})$$

where C is the parent of B. Then

$$m^+_B(x_C) = \sum_{y_{B\setminus C}} \varphi_B(x_{B\cap C}\wedge y_{B\setminus C}) \prod_{B\to B'} m^+_{B'}(x_{B\cap C}\wedge y_{B\setminus C})$$

with

$$\sigma^+_C(x_C) = \prod_{C\to B} m^+_B(x_C)$$

which provides σ_C at the root. Reinterpreting this discussion in terms of the undirected graph, we are led to introducing messages $m_{BC}(x_C)$ for $B\sim C$ in $\mathbb{G}$, with the message-passing rule

$$m_{BC}(x_C) = \sum_{y_{B\setminus C}} \varphi_B(x_{B\cap C}\wedge y_{B\setminus C}) \prod_{B'\sim B, B'\neq C} m_{B'B}(x_{B\cap C}\wedge y_{B\setminus C}) \tag{62}$$

Messages progressively stabilize when applied in $\mathbb{G}$, and at convergence, we have

$$\sigma_C(x_C) = \varphi_C(x_C)\prod_{B\sim C} m_{BC}(x_C). \tag{63}$$

Note that the complexity of the junction tree algorithm is exponential in the cardinality of the largest $C\in\mathcal{S}$. This algorithm will therefore be unfeasible if $\mathcal{S}$ contains sets that are too large.

5. Building Junction Trees

There is more than one family of set interactions with respect to which a given probability π can be decomposed (notice that, unlike in the Hammersley-Clifford Theorem, we do not assume that the interactions are normalized), and not all of them can be organized as a

junction tree. One can however extend a given family into a new one on which one can build a junction tree.

DEFINITION 22. *Let V be a set of vertices, and $\mathcal{S}_0 \subset \boldsymbol{\mathcal{P}}(V)$. We say that a set $\mathcal{S} \subset \boldsymbol{\mathcal{P}}(V)$ is an extension of $\mathcal{S}_0$ if, for any $C_0 \in \mathcal{S}_0$, there exists a $C \in \mathcal{S}$ such that $C_0 \subset C$.*

A tree $\mathbb{G} = (\mathcal{S}, E)$ is a junction-tree extension of $\mathcal{S}_0$ if $\mathcal{S}$ is an extension of $\mathcal{S}_0$ and $\mathbb{G}$ is a junction tree.

If $\Phi^0 = (\varphi^0_C, C \in \mathcal{S}_0)$ is a consistent family of set interactions, and $\mathcal{S}$ is an extension of $\mathcal{S}_0$, one can build a new family, $\Phi = (\varphi_C, C \in \mathcal{S})$, of set interactions which yields the same probability distribution, i.e., such that, for all $x_V \in F_V$,

$$\prod_{C \in \mathcal{S}} \varphi_C(x_C) \propto \prod_{C_0 \in \mathcal{S}_0} \varphi^0_{C_0}(x_{C_0}).$$

For this, it suffices to build a mapping say $T : \mathcal{S}_0 \to \mathcal{S}$ such that $C_0 \subset T(C_0)$ for all $C_0 \in \mathcal{S}_0$, which is always possible since $\mathcal{S}$ is an extension of $\mathcal{S}_0$ (for example, arbitrarily order the elements of $\mathcal{S}$ and let $T(\mathcal{S}_0)$ be the first element of $\mathcal{S}$, according to this order, that contains C_0). One can then define

$$\varphi_C(x_C) = \prod_{C_0 : T(C_0) = C} \varphi^0_{C_0}(x_{C_0}).$$

Given Φ^0, our goal is to design a junction-tree extension which is as feasible as possible. So, we are not interested by the trivial extension $\mathbb{G} = (V, \emptyset)$, since the resulting junction-tree algorithm is unfeasible as soon as V is large. Theorem 3 in the next section will be the first step in the design of an algorithm that computes junction trees on a given graph.

5.1. Triangulated Graphs.

DEFINITION 23. *Let $G = (V, E)$ be an undirected graph. Let $(s_1, s_2, \ldots, s_n)$ be a path in G. One says that this path has a chord at s_j, with $j \in \{2, \ldots, n\}$, if $s_{j-1} \sim s_{j+1}$, and we will refer to (s_{j-1}, s_j, s_{j+1}) as a chordal triangle. A path in G is achordal if it has no chord.*

One says that G is triangulated (or chordal) if it has no achordal loop.

The graph G is decomposable if it satisfies the following recursive condition: it is either complete, or there exists disjoint subsets (A, B, C) of V such that

- *$V = A \cup B \cup C$,*
- *A and B are not empty,*
- *C is clique in G, C separates A and B,*
- *the restricted graphs, $G_{A \cup C}$ and $G_{B \cup C}$ are decomposable.*

The last two properties are in fact equivalent, as stated in the following proposition.

PROPOSITION 30. *An undirected graph is triangulated if and only if it is decomposable*

PROOF. To prove the "if" part, we proceed by induction on $n = |V|$. Note that every graph for $n \leq 3$ is both decomposable and triangulated (we leave the verification to the reader). Assume that the statement "decomposable $\Rightarrow$ triangulated" holds for graphs with less than n vertices, and take G with n vertices. Assume that G is decomposable. If it is complete, it is obviously triangulated. Otherwise, there exists A, B, C such that $V = A \cup B \cup C$, with A and B non-empty such that $G_{A \cup C}$ and $G_{B \cup C}$ are decomposable, hence

triangulated from the induction hypothesis, and such that C is a clique which separates A and B. Let us prove that G is triangulated. Assume that γ is an achordal loop in G. Since it cannot be included in $A \cup C$ or $B \cup C$, γ must go from A to B and back, which implies that it passes at least twice in C. Since C is complete, the original loop can be shortcut to form subloops in $A \cup C$ and $B \cup C$. If one of (or both) these loops has cardinality 3, this would provide γ with a chord, which contradicts the assumption. Otherwise, the following lemma also provides a contradiction, since one of the two chords that it implies must also be a chord in the original γ.

LEMMA 6. *Let $(s_1, \ldots, s_n, s_{n+1} = s_1)$ be a loop in a triangulated graph, with $n \geq 4$. Then the path has a chord at two non-contiguous vertices at least.*

To prove the lemma, assume the contrary and let $(s_1, \ldots, s_n, s_{n+1} = s_1)$ be a loop that does not satisfy the condition, with n as small as possible. If $n > 4$, the loop must have a chord, say at s_j, and one can remove s_j from the loop to still obtain a smaller loop that must satisfy the condition in the lemma, since n was as small as possible. One of the two chords must be at a vertex other than the two neighbors of s_j, and thus provide a second chord in the original loop, which is a contradiction. Thus $n = 4$, but G being triangulated implies that this 4-point loop has a diagonal, so that the condition in the lemma holds also which provides a contradiction.

For the "only if" part, assume that G is triangulated. We prove that the graph is decomposable by induction on $|G|$. The induction will work if we can show that, if G is triangulated, it is either complete or there exists a clique in G such that $V \setminus C$ is disconnected, i.e., there exist two elements $a, b \in V \setminus C$ which are related by no path in $V \setminus C$. Indeed, we will then be able to decompose $V = A \cup B \cup C$, where A and B are unions of (distinct) connected components of $V \setminus C$. Take, for example, A to be the set of vertices connected to A in $G \setminus C$, and $B = V \setminus (A \cup C)$, which is not empty since it contains b. Note that restricted graphs from triangulated graphs are triangulated too.

So, assume that G is triangulated, and not complete. Let C be a subset of V that satisfies the property that $V \setminus C$ is disconnected, and take C minimal, so that $V \setminus C'$ is connected for any $C' \subset C$, $C' \neq C$. We want to show that C is a clique, so take s and t in C and assume that they are not neighbors to reach a contradiction.

Let A and B be two connected components of $V \setminus C$. For any $a \in A$, $b \in B$, and $s, t \in C$, we know that there exists a path between a and b in $V \setminus C \cup \{s\}$ and another one in $V \setminus C \cup \{t\}$, the first one passing by s (because it would otherwise connect a and b in $V \setminus C$) and the second one passing by t. Any point before s (or t) in these paths must belong to A, and any point after them must belong to B. Concatenating these two paths, and removing multiple points if needed, we obtain a loop passing in A, then by s, then in B, then by t. We can recursively remove all points at which these paths have a chord. We can also notice that we cannot remove s nor t in this process, since this would imply an edge between A and B, and that we must leave at least one element in A and one in B because removing the last one would require $s \sim t$. So, at the end, we obtain an achordal loop with at least four points, which contradicts the fact that G is triangulated.

□

We can now characterize graphs that admit junction trees over the set of their maximal cliques.

THEOREM 3. *Let $G = (V, E)$ be an undirected graph, and $\mathcal{C}_G^*$ be the set of all maximum cliques in G The following two properties are equivalent.*

(i) *There exists a junction tree over $\mathcal{C}_G^*$.*
(ii) *G is triangulated/decomposable.*

PROOF. The proof works by induction on the number of maximal cliques, $|\mathcal{C}_G^*|$. If G has only one maximal clique, then G is complete, because any point not included in this clique will have to be included in another maximal clique, which leads to a contradiction. So G is decomposable, and, since any single node obviously provides a junction tree, (i) is true also.

Now, fix G and assume that the theorem is true for any graph with fewer maximal cliques. First assume that $\mathcal{C}_G^*$ has a junction tree, $\mathbb{T}$. Let C_1 be a leaf in $\mathbb{T}$, connected, say, to C_2, and let $\mathbb{T}_2$ be $\mathbb{T}$ restricted to $\mathcal{C}_2 = \mathcal{C}_G^* \setminus \{C_1\}$. Let V_2 be the unions of maximal cliques from nodes in $\mathbb{T}_2$. A maximal clique C in G_{V_2} is a clique in G_V and therefore included in some maximal clique $C' \in \mathcal{C}_V$. If $C' \in \mathcal{C}_2$, then C' is also a clique in G_{V_2}, and for C to be maximal, we need $C = C'$. If $C' = C_1$, we note that we must also have

$$C = \bigcup_{\tilde{C} \in \mathcal{C}_2} C \cap \tilde{C}$$

and whenever $C \cap \tilde{C}$ is not empty, this set must be included in any node in the path in $\mathbb{T}$ that links $\tilde{C}$ to C_1. Since this path contains C_2, we have $C \cap \tilde{C} \subset C_2$ so that $C \subset C_2$, but, since C is maximal, this would imply that $C = C_2 = C_1$ which is impossible.

This shows that $\mathcal{C}_{G_2}^* = \mathcal{C}_2$. This also shows that $\mathbb{T}_2$ is a junction tree over $\mathcal{C}_2$. So, by the induction hypothesis, G_{V_2} is decomposable. If $s \in V_2 \cap C_1$, then s also belongs to some clique $C' \in \mathcal{C}_2$, and therefore belongs to any clique in the path between C' and C_1, which includes C_2. So $s \in C_1 \cap C_2$ and $C_1 \cap V_2 = C_1 \cap C_2$. So, letting $A = C_1 \setminus (C_1 \cap C_2)$, $B = V_1 \setminus (C_1 \cap C_2)$, $S = C_1 \cap C_2$, we know that $G_{A \cup S}$ and $G_{B \cup S}$ are decomposable (the first one being complete), and that S is a clique. To show that G is decomposable, it remains to show that S separates A from B.

If a path connects A to B in G, it must contain an edge, say $\{s, t\}$, with $s \in V \setminus S$ and $t \in S$; $\{s, t\}$ must be included in a maximal clique in G. If this clique is C_1, we have $s \in C_1 \cap V_2 = S$. The same argument shows that this is the only possibility, because, if $\{s, t\}$ is included in some maximal clique in $\mathcal{C}_2$, then we would find $t \in C_1 \cap C_2$. So S separates A and B in G.

Let us now prove the converse statement, and assume that G is decomposable. If G is complete, it has only one maximal clique and we are done. Otherwise, there exists a partition $V = A \cup B \cup S$ such that $G_{A \cup S}$ and $G_{B \cup S}$ are decomposable, A and B separated by S which is complete. Let $\mathcal{C}_A^*$ be the maximal cliques in $G_{A \cup S}$ and $\mathcal{C}_B^*$ the maximal cliques in $G_{B \cup S}$. By hypothesis, there exist junction trees $\mathbb{T}_A$ and $\mathbb{T}_B$ over $\mathcal{C}_A^*$ and $\mathcal{C}_B^*$.

Let C be a maximal clique in $G_{A \cup S}$. Assume that C intersect A; C can be extended to a maximal clique, C', in G, but C' cannot intersect B (since this would imply a direct edge between A and B) and is therefore included in $A \cup S$, so that $C = C'$. Similarly, all maximal cliques in $G_{B \cup S}$ that intersect B also are maximal cliques in G.

The clique S is included in some maximal clique $S_A^* \in \mathcal{C}_A^*$. From the previous discussion, we have either $S_A^* = S$ or $S_A^* \in \mathcal{C}_G^*$. Similarly, S can be extended to a maximal clique $S_B^* \in \mathcal{C}_B^*$, with $S_B^* = S$ or $S_B^* \in \mathcal{C}_G^*$. Notice also that at least one of S_A^* or S_B^* must be a maximal clique in G: indeed, assume that both sets are equal to S, which, as a clique, can extended to a maximal clique S^* in G; S^* must be included either in $A \cup S$ or in $B \cup S$, and therefore be a maximal clique in the corresponding graph which yields $S^* = S$. Reversing the notation if needed, we will assume that $S_A^* \in \mathcal{C}_G^*$.

All elements of $\mathcal{C}_G^*$ must belong either to $\mathcal{C}_A^*$ or $\mathcal{C}_B^*$ since any maximal clique, say C, in G must be included in either $A \cup S$ or $B \cup S$, and therefore also provide a maximal clique in the related graph. So the nodes in $\mathbb{T}_A$ and $\mathbb{T}_B$ enumerate all maximal cliques in G, and we can build a tree $\mathbb{T}$ over $\mathcal{C}_G^*$ by identifying S_A^* and S_B^* to S^* and merging the two trees at this node. To conclude our proof, it only remains to show that the running intersection property is satisfied. So consider two nodes C, C' in $\mathbb{T}$ and take $s \in C \cap C'$. If the path between these nodes remain in $\mathcal{C}_A^*$, or in $\mathcal{C}_B^*$, then s will belong to any set along that path, since the running intersection is true on $\mathbb{T}_A$ and $\mathbb{T}_B$. Otherwise, we must have $s \in S$, and the path must contain S^* to switch trees, and s must still belong to any clique in the path (applying the running intersection property between the beginning of the path and S^*, and between S^* and the end of the path). □

This theorem delineates a strategy in order to build a junction tree that is adapted to a given family of local interactions $\Phi = (\varphi_C, C \in C)$. Letting G be the graph induced by these interactions, i.e., $s \sim_G t$ if and only if there exists $C \in \mathcal{C}$ such that $\{s, t\} \subset C$, the method proceeds as follows.

(JT1) Extend G by adding edges to obtain a triangulated graph G^*.
(JT2) Compute the set $\mathcal{C}^*$ of maximal cliques in G^*, which therefore extend $\mathcal{C}$.
(JT3) Build a junction tree over $\mathcal{C}^*$.
(JT4) Assign interaction φ_C to a clique $C^* \in \mathcal{C}^*$ such that $C \subset C^*$.
(JT5) Run the junction-tree belief propagation algorithm to compute the marginal of π (associated to Φ) over each set $C^* \in \mathcal{C}^*$.

Steps (JT4) and (JT5) have already been discussed, and we now explain how the first three steps can be implemented.

5.2. Building Triangulated Graphs. First consider step (JT1). To triangulate a graph $G = (V, E)$, it suffices to order its vertices so that $V = \{s_1, \ldots, s_n\}$, and then run the following algorithm, starting with $k = n$ and $E_n = E$:

- *Given E_k, add an edge to any pair of neighbors of s_k (unless, of course, they are already linked).*
- *Let E_{k-1} be the new set of edges.*

Then the graph $G^* = (V, E_0)$ is triangulated. Indeed taking any achordal loop, and selecting the vertex with highest index in the loop, say s_k, brings a contradiction, since the neighbors of s_k have been linked when building E_{k-1}.

However, the quality of the triangulation, which can be measured by the number of added edges, or by the size of the maximal cliques, highly depends on the way vertices have been numbered. Take the simple example of the linear graph with three vertices $A \sim B \sim C$. If the point of highest index is B, then the previous algorithm will return the three-point

loop $A \sim B \sim C \sim A$. Any other ordering will leave the linear graph, which is already triangulated, invariant.

So, one must be careful about the order with which nodes will be processed. Finding an optimal ordering for a given global cost is an NP-complete problem. However, a very simple modification of the previous algorithms, which starts with s_n having the minimal number of neighbors, and at each step defines s_k to be the one with fewest neighbors that hasn't been visited yet, provides a rather efficient way for building triangulations. (It has the merit of leaving G invariant if it is a tree, for example). Another criterion may be preferred to the number of neighbors (for example, the number of new edges that would be needed if s is added).

If G is triangulated, there exists an ordering of V such that the algorithm above leaves G invariant. We now proceed to a proof of this statement and also show that such an ordering can be computed using an algorithm called maximum cardinality search, which, in addition, allows one to decide whether a graph is triangulated. We start with a definition that formalizes the sequence of operations in the triangulation algorithm.

DEFINITION 24. *Let $G = (V, E)$ be an undirected graph. A node elimination consists in selecting a vertex $s \in V$ and building the graph $G^{(s)} = (V^{(s)}, E^{(s)})$ with $V^{(s)} = V \setminus \{s\}$, and $E^{(s)}$ containing all pairs $\{t, t'\} \subset V^{(s)}$ such that either $\{t, t'\} \in E$ or $\{t, t'\} \subset \mathcal{V}_s$.*

$G^{(s)}$ is called the s-elimination graph of G. The set of added edges, namely $E^{(s)} \setminus (E \cap E^{(s)})$ is called the deficiency set of s and denoted $D(s)$ (or $D_G(s)$).

So, the triangulation algorithm implements a sequence of node eliminations, successively applied to s_n, s_{s-1}, etc. One says that such an elimination process is *perfect* if, for all $k = 1, \ldots, n$, the deficiency set of s_k in the graph obtained after elimination of $s_n, \ldots, s_{k+1}$ is empty (so that no edge is added during the process). We will also say that $(s_1, \ldots, s_n)$ provides a perfect ordering for G.

THEOREM 4. *An undirected graph $G = (V, E)$ admits a perfect ordering if and only if it is triangulated.*

PROOF. The "only if" part is obvious, since, the triangulation algorithm following a perfect ordering does not add any edge to G, which must therefore have been triangulated to start with.

We now proceed to the "if" part. For this it suffices to prove that for any triangulated graph, there exists a vertex s such that $D_G(s) = \emptyset$. One can then easily prove the result by induction, since, after removing this s, the remaining graph $G^{(s)}$ is still triangulated and would admit (by induction) a perfect ordering that completes this first step.

To prove that such an s exists, we take a decomposition $V = A \cup S \cup B$, in which S is complete and separates A and B, such that $|A \cup S|$ is minimal (or $|B|$ maximal). We claim that $A \cup S$ must be complete. Otherwise, since $A \cup S$ is still triangulated, There exists a similar decomposition $A \cup S = A' \cup S' \cup B'$. One cannot have $S \cap A'$ and $S \cap B'$ non empty simultaneously, since this would imply a direct edge from A' to B' (S is complete). Say that $S \cap A' = \emptyset$, so that $A' \subset A$. Then the decomposition $V = A' \cup S' \cup (B' \cup B)$ is such that S' separates A' from $B \cup B'$. Indeed, a path from A' to $b \in B \cup B'$ must pass in S' if $b \in B'$, and, if $b \in B$, it must pass in S (since it links A and B). But $S \subset S' \cup B'$ so that the path must intersect S'. We therefore obtain a decomposition that enlarges B, which is

a contradiction and shows that $A \cup S$ is complete. Given this, any element $s \in A$ can only have neighbors in $A \cup S$ and is therefore such that $D_G(s) = \emptyset$, which concludes the proof. □

If a graph is triangulated, there is more than one perfect ordering of its vertices. One of these orderings is provided the *maximum cardinality search* algorithm, which also allows one to decide whether the graph is triangulated. We start with a definition/notation.

DEFINITION 25. *If $G = (V, E)$ is an undirected graph, with $|V| = n$, any ordering $V = (s_1, \dots, s_n)$ can be identified with the bijection $\alpha : V \to \{1, \dots, n\}$ defined by $\alpha(s_k) = k$. In other terms, $\alpha(s)$ is the rank of s in the ordering. We will refer to α as an ordering, too.*

Given an ordering α, we define incremental neighborhoods $\mathcal{V}_s^{\alpha,k}$, for $s \in V$ and $k = 1, \dots, n$ to be the intersections of $\mathcal{V}_s$ with the sets $\alpha^{-1}(\{1, \dots, k\})$, i.e.,

$$\mathcal{V}_s^{\alpha,k} = \{t \in V, t \sim s, \alpha(t) \leq k\}.$$

One says that α satisfies the maximum cardinality property if, for all $k = \{2, \dots, n\}$

$$|\mathcal{V}_{s_k}^{\alpha,k-1}| = \max_{\alpha(s) \geq k} |\mathcal{V}_s^{\alpha,k-1}|. \tag{64}$$

where $s_k = \alpha^{-1}(k)$.

Given this, we have the proposition:

PROPOSITION 31. *If $G = (V, E)$ is triangulated, then any ordering that satisfies the maximum cardinality property is perfect.*

Equation (64) immediately provides an algorithm that constructs an ordering satisfying the maximum cardinality property given a graph G. From Proposition 31, we see that, if for some k, the largest set $\mathcal{V}_{s_k}^{\alpha,k-1}$ is not a clique, then G is not triangulated. We now proceed to the proof of this proposition.

PROOF. Let G be triangulated, and assume that α is an ordering that satisfies (64). Assume that α is not proper in order to reach a contradiction.

Let k be the first index for which $\mathcal{V}_{s_k}^{\alpha,k-1}$ is not a clique, so that s_k has two neighbors, say t and u, such that $\alpha(t) < k$, $\alpha(u) < k$ and $t \not\sim u$. Assume that $\alpha(t) > \alpha(u)$. Then t must have a neighbor that is not neighbor of s, say t', such that $\alpha(t') < \alpha(t)$ (otherwise, s would have more neighbors than t at order less than $\alpha(t)$, which contradicts the maximum cardinality property). The sequence t', t, s, u forms a path that is such that α increases from t' to s, then decreases from s to u, and contains no chord. Moreover, t' and u cannot be neighbors, since this would yield an achordal loop and a contradiction. The proof of Proposition 31 consists in showing that this construction can be iterated until a contradiction is reached.

More precisely, assume that an achordal path $s_1, \dots, s_k$ has been obtained, such that $\alpha(s)$ is first increasing, then decreasing along the path, and such that, at extremities one either has $\alpha(s_1) < \alpha(s_k) < \alpha(s_2)$ or $\alpha(s_k) < \alpha(s_1) < \alpha(s_{k-1})$. In fact, one can switch between these last two cases by reordering the path backward. Both paths (u, s, t) and (u, s, t, t') in the discussion above satisfy this property.

- Assume, without loss of generality, that $\alpha(s_1) < \alpha(s_k) < \alpha(s_2)$ and note that, in the considered path, s_1 and s_k cannot be neighbors (for, if j is the last index smaller than $k-1$ such that s_j and s_k are neighbors, then j must also be smaller than $k-2$ and the loop $s_j, \dots, s_{k-1}, s_k$ would be achordal).

- Since $\alpha(s_2) > \alpha(s_k)$, and s_1 and s_2 are neighbors, s_k must have a neighbor, say s'_k, such that s'_k is not neighbor of s_2 and $\alpha(s'_k) < \alpha(s_k)$.
- Select the first index $j > 2$ such that $s_j \sim s'_k$, and consider the path $(s_1, \ldots, s_j, s'_k)$. This path is achordal, by construction, and one cannot have $s_1 \sim s'_k$ since this would create an achordal loop. Let us show that α first increases and then decreases along this path. Since s_2 is in the path, α must first increase, and it suffices to show that $\alpha(s'_k) < \alpha(s_j)$. If α increases from s_1 to s_j, then $\alpha(s_j) > \alpha(s_2) > \alpha(s_k) > \alpha(s'_k)$. If α started decreasing at some point before s_j, then $\alpha(s_j) > \alpha(s_k) > \alpha(s'_k)$.
- Finally, we need to show that the α-value at one extremity is between the first two α-values on the other end of the path. If $\alpha(s'_k) < \alpha(s_1)$, and since we have just seen that $\alpha(s_j) > \alpha(s_k) > \alpha(s_1)$, we do get $\alpha(s'_k) < \alpha(s_1) < \alpha(s_j)$. If $\alpha(s'_k) > \alpha(s_1)$, then, since by construction $\alpha(s_2) > \alpha(s_k) > \alpha(s'_k)$, we have $\alpha(s_2) > \alpha(s'_k) > \alpha(s_1)$.
- So, we have obtained a new path that satisfies the same property that the one we started with, but with a maximum value at end points smaller than the initial one, i.e.,
$$\max(\alpha(s_1), \alpha(s'_k)) < \max(\alpha(s_1), \alpha(s_k)).$$
Since α takes a finite number of values, this process cannot be iterated indefinitely, which yields our contradiction.

□

5.3. Computing Maximal Cliques. At this point, we know that a graph must be triangulated for its maximal cliques to admit junction trees, and we have an algorithm to decide whether a graph is triangulated, and extend it into a triangulated one if needed. This provides the first step, (JT1), of our description of the junction tree algorithm. The next step, (JT2), requires computing a list of maximal cliques. Computing maximal cliques in general graph is an NP complete, problem, for which a large number of algorithms has been developed (see, for example, [**30**] for a review). For graphs with a perfect ordering, however, this problem can always be solved in a polynomial time.

Indeed, assume that a perfect ordering is given for $G = (V, E)$, so that $V = \{s_1, \ldots, s_n\}$ is such that, for all k, $\mathcal{V}'_{s_k} := \mathcal{V}_{s_k} \cap \{s_1, \ldots, s_{k-1}\}$ is a clique. Let G_k be G restricted to $\{s_1, \ldots, s_k\}$ and $\mathcal{C}^*_k$ be the set of maximal cliques in G_k. Then the set $C_k := \{s_k\} \cup \mathcal{V}'_{s_k}$ is the only maximal clique in G_k that contains s_k: it is a clique because the ordering is perfect, and any clique that contains s_k must be included in it (because its elements are either s_k or neighbors of s_k). It follows from this that the set $\mathcal{C}^*_k$ can be deduced from C^*_{k-1} by

$$\begin{cases} \mathcal{C}^*_k = \mathcal{C}^*_{k-1} \cup \{C_k\} \text{ if } \mathcal{V}'_k \notin \mathcal{C}^*_{k-1} \\ \mathcal{C}^*_k = (\mathcal{C}^*_{k-1} \cup \{C_k\}) \setminus \{\mathcal{V}'_k\} \text{ if } \mathcal{V}'_k \in \mathcal{C}^*_{k-1} \end{cases}$$

This allows one to enumerate all elements in $\mathcal{C}^*_G = \mathcal{C}^*_n$, starting with $\mathcal{C}^*_1 = \{\{s_1\}\}$.

5.4. Characterization of Junction Trees. We now discuss the last remaining point, (JT3). For this, we need to form the *clique graph* of G, which is the undirected graph $\mathbb{G} = (\mathcal{C}^*_G, \mathbb{E})$ defined by $(C, C') \in \mathbb{E}$ if and only if $C \cap C' \neq \emptyset$. We then have the following fact:

PROPOSITION 32. *The clique graph $\mathbb{G}$ of a connected triangulated undirected graph G is connected.*

PROOF. We proceed by induction, and assume that the result is true if $|V| = n-1$ (the proposition obviously holds if $|V| = 1$). Assume that a perfect order on G has been chosen, say $V = \{s_1, \ldots, s_n\}$. Let G' be G restricted to $\{s_1, \ldots, s_{n-1}\}$, and $\mathbb{G}'$ the associated clique graph. Because $\{s_n\} \cup \mathcal{V}_{s_n}$ is a clique, any path in G provides a valid path in G' after removing all occurrences of s_n (because any two neighbors of s_n are linked). The induction hypothesis also implies that $\mathbb{G}'$ is connected. Since G is connected, $\mathcal{V}_{s_n}$ is not empty. Moreover, $C := \{s_n\} \cup \mathcal{V}_{s_n}$ must be a maximal clique in G (since we assume that the order is perfect) and it is the only maximal clique in G that contains s_n (all other maximal cliques in G therefore are maximal cliques in G' also). To prove that $\mathbb{G}$ is connected, it suffices to prove that C is connected to any other maximal clique, C', in G by a path in $\mathbb{G}$. If $t \in C$, $t \neq s_n$, there exists a maximal clique, say C'', in G' that contains t, and, since $\mathbb{G}'$ is connected, there exists a path $(C_1 = C', \ldots, C_q = C'')$ connecting C' to C'' in $\mathbb{G}'$. Let j be the first integer such that $C_j = \mathcal{V}_n$ (take $j = q+1$ if this never happens). Then $(C_1, \ldots, C_{j-1}, C)$ is a path linking C' and C in $\mathbb{G}$. □

We hereafter assume that G, and hence $\mathbb{G}$, is connected. This is not real loss of generality because connected components in undirected graphs yields independent processes that can be handled separately. We assign weights to edges of the clique graph of G by defining $w(C, C') = |C \cap C'|$. Recall that a subgraph $\tilde{T}$ of any given graph $\tilde{G}$ is called a spanning tree if $\tilde{T}$ is a tree with set of vertices equal to the set of vertices of $\tilde{G}$. If $\mathbb{T} = (\mathcal{C}_G^*, \mathbb{E}')$ is a spaning tree of $\mathbb{G}$, we define the total weight

$$w(\mathbb{T}) = \sum_{\{C,C'\} \in \mathbb{E}'} w(C, C').$$

We then have the proposition:

PROPOSITION 33. [**20**] *If G is a connected triangulated graph, the set of junction trees over $\mathcal{C}_G^*$ coincides with the set of maximizers of $w(\mathbb{T})$ over all spanning trees of $\mathbb{G}$.*

(Notice that $\mathbb{G}$ being connected implies that spanning trees over $\mathbb{G}$ exist.)

Before proving this proposition, we discuss some properties related to maximal (or maximum-weight) spanning trees over an undirected graph. For this discussion, we let $G = (V, E)$ be any undirected graph with weight $(w(e), e \in E)$. We will then apply these results to a clique graph when will switch back to the general notation of this section. Maximal spanning trees can be computed using the so-called Prim's algorithm [**19, 34, 9**]. This algorithm builds a sequence of trees $T_k = (V_k, E_k)$, starting with $T_1 = (\{s_1\}, \emptyset)$, for some arbitrary $s_1 \in V$. Given T_{k-1}, the next tree is defined with

$$V_k = \{s_k\} \cup V_{k-1} \ (s_k \notin V_{k-1})$$

and $E_k = \{e_k\} \cup E_{k-1}$, such that $e_k = \{s_k, s\}$ for some $s \in V_{k-1}$ satisfying

$$w(e_k) = \max\Big(w(\{t, t'\}), \{t, t'\} \in E, t \notin V_{k-1}, t' \in V_{k-1}\Big) \tag{65}$$

This ability of this algorithm to always build a maximal spanning tree is summarized in the following proposition [**17, 27**].

PROPOSITION 34. *If $G = (V, E)$ is a weighted, connected undirected graph, Prim's algorithm, as described above, provides a sequence $T_k = (V_k, E_k)$, for $k = 1, \ldots, n$ of subtrees of*

G such that $V_n = V$ and, for all k, T_k is a maximal spanning tree for the restriction G_{V_k} of G to V_k.

Moreover, any maximal spanning tree of G, can be realized as T_n, where $(T_1, \ldots, T_n)$ is a sequence provided by Prim's algorithm.

PROOF. We first prove that, for all k, T_k is maximal spanning tree on the graph G_{V_k}.

We will prove a slightly stronger statement, namely, that, for all k, T_k can be extended to form a maximal spanning tree of G. This is stronger, because, if $T_k = (V_k, E_k)$ can be extended to a maximal spanning tree $T = (V, E)$, and if $T'_k = (V_k, E'_k)$ is a spanning tree for G_{V_k} such that $(w(T_k) < w(T'_k)$, then the graph $T' = (V, E')$ with

$$E' = (E \setminus E_k) \cup E'_k$$

would be a spanning tree for G with $w(T) < w(T')$, which is impossible. To see that T' is a tree, notice that paths in T' are in one-to-one correspondence with paths in T by replacing any subpath within T'_k by the unique subpath in T_k that has the same extremities.

Clearly, T_1, which only has one vertex, can be extended to a maximal spanning tree. Let $k \geq 1$ be the last integer for which this property is true for all $j = 1, \ldots, k$. If $k = n$, we are done. Otherwise, take a maximum spanning tree, T, that extends T_k. This tree cannot contain the new edge added when building T_{k+1}, namely $e_{k+1} = \{s_{k+1}, s\}$ as defined in Prim's algorithm, since it would otherwise also extend T_{k+1}. Consider the path γ in T that links s to s_k. This path must have an edge $e = \{t, t'\}$ such that $t \in V_k$ and $t' \notin V_k$, and by definition of e_{k+1}, we must have $w(e) \leq w(e_{k+1})$. Notice that e is uniquely defined, because a path leaving V_k cannot return in this set, since one would be otherwise able to close it into a loop by inserting the only path in T_k that connects its extremities.

Replace e by e_{k+1} in T. The resulting graph, say T', is still a spanning tree for G. From any path in T, one can create a path in T' with the same extremities by replacing any occurrence of the edge, e, by the concatenation of the unique path in T going from t to s, followed by (s, s_{k+1}), followed by the unique path in T going from s_{k+1} to t'. This implies that T' is connected. It is also acyclic, since any loop in T would have to contain e_{k+1} (since T is acyclic), but there is no other path than (s, s_{k+1}) in T' that links s and s_k, because this path would have to be in T, and we have removed the only possible one from T by deleting the edge e.

As a conclusion, T' is an extension of T_{k+1}, and a spanning tree with total weight larger or equal to the one of T, and must therefore be optimal, too. But this contradicts the fact that T_{k+1} cannot be extended to a maximal tree, so that $k = n$ and the sequence of trees provided by Prim's algorithm is optimal.

To prove the second statement, let T be an optimal spanning tree. Let k be the largest integer such that there exists a sequence $(T_1, \ldots, T_k)$ generated by Prim's algorithm, such that, for all $j = 1, \ldots, k$, T_j is a subtree of T. One necessarily has $j \geq 1$, since T extends any one-vertex tree. If $k = n$, we are done. Assuming otherwise, let $T_k = (V_k, E_k)$ and make one more step of Prim's algorithm, selecting an edge $e_{k+1} = (s_{k+1}, s)$ satisfying (65). By assumption, e_{k+1} is not in T. Take as before the unique path linking s and s_{k+1} in T and let e be the unique edge at which this path leaves V_k. Replacing e by e_{k+1} in T provides a new spanning tree, T'. One must have $w(e) \geq w(e_{k+1})$ because T is optimal, and $w(e_{k+1}) \geq w(e)$ by (65). So $w(e) = w(e_{k+1})$, and one can use e instead of e_{k+1} for the $(k+1)$th step of

Prim's algorithm. But this contradicts the fact that k was the largest integer in a sequence of subtrees of T that is generated by Prim's algorithm, and one therefore has $k = n$. □

The proof of Proposition 33, that we provide now, uses very similar "edge-switching" arguments.

Proof of Proposition 33. Let us start with an maximum-weight spanning tree for $\mathbb{G}$, say $\mathbb{T}$, and show that it is a junction tree. Since $\mathbb{T}$ has maximum weight, we know that it can be obtained via Prim's algorithm, and that there exists a sequence $\mathbb{T}_1, \ldots, \mathbb{T}_n = \mathbb{T}$ of trees constructed by this algorithm. Let $\mathbb{T}_k = (\mathcal{C}_k, \mathbb{E}_k)$.

We proceed by contradiction. Let k be the largest index such that $\mathbb{T}_k$ can be extended to a junction tree for $\mathcal{C}_G^*$, and let $\mathbb{T}'$ be a junction tree extension of $\mathbb{T}_k$. Assume that $k < n$, and let $e_{k+1} = (C_{k+1}, C')$ be the edge that has been added when building $\mathbb{T}_{k+1}$, with $\mathcal{C}_{k+1} = \{C_{k+1}\} \cup \mathcal{C}_k$. This edge is not in $\mathbb{T}'$, and there therefore exists a unique edge $e = (B, B')$ in the path between C_k and C' in $\mathbb{T}'$ such that $B \in \mathcal{C}_k$ and $B' \notin \mathcal{C}_k$. We must have $w(e) = |B \cap B'| \leq w(e_{k+1}) = |C_{k+1} \cap C'|$. But, since the running intersection property is true for $\mathbb{T}'$, both B and B' must contain $C_{k+1} \cap C'$ so that $B \cap B' = C_{k+1} \cap C'$. This implies that, if one modifies $\mathbb{T}'$ by replacing edge e by edge e_{k+1}, yielding a new spanning tree $\mathbb{T}''$, the running intersection property is still satisfied in $\mathbb{T}'$. Indeed if a vertex $s \in V$ belongs to both extremities of a path containing B and B' in $\mathbb{T}'$, then it must belong to $B \cap B'$, and hence to $C_{k+1} \cap C'$, and therefore to any set in the path in $\mathbb{T}'$ that linked C_{k+1} and C'. So we found a junction tree extension of $\mathbb{T}_{k+1}$, which contradicts our assumption that k was the largest. So we must have $k = n$ and $\mathbb{T}$ is a junction tree.

Let us now consider the converse statement and assume that $\mathbb{T}$ is a junction tree. Let k be the largest integer such that there exists a sequence of subgraphs of $\mathbb{T}$ that is provided by Prim's algorithm. Denote such a sequence by $(\mathbb{T}_1, \ldots, \mathbb{T}_k)$, with $\mathbb{T}_j = (\mathcal{C}_j, \mathbb{E}_j)$. Assume (to get a contradiction) that $k < n$, and consider a new step for Prim's algorithm, adding a new edge $e_{k+1} = \{C_{k+1}, C'\}$ to $\mathbb{T}_k$. Take as before the path in $\mathbb{T}$ linking C' to C_{k+1} in $\mathbb{T}$, and select the edge e at which this path leaves $\mathcal{C}_k$. If $e = (B, B')$, we must have $w(e) = |B \cap B'| \leq w(e_k) = |C_{k+1} \cap C'|$, and the running intersection property in $\mathbb{T}$ implies that $C_{k+1} \cap C' \subset B \cap B'$, which implies that $w(e) = w(e_{k+1})$. This implies that adding e instead of e_{k+1} at step $k+1$ is a valid choice for Prim's algorithm, and contradicts the fact that k was the largest number of such steps that could provide a subtree of $\mathbb{T}$. So $k = n$ and $\mathbb{T}$ is maximal. □

6. Monte-Carlo Sampling

The previous algorithms provided deterministic methods to compute, or approximate marginal probabilities of graphical models. In this section, we describe alternate methods that estimate the same quantities based on random samples.

More precisely, the goal of this section is to describe how, from a basic random number generator that provides samples from a uniform distribution on $[0, 1]$, one can generate samples that follow, or approximately follow, the distribution of a G-Markov process, with very few conditions on the structure of the graph. This, combined with the Law of large numbers, permits to approximate probabilities or expectations by empirical averages over a large collection of generated samples.

We assume that as many as needed independent samples of the uniform distribution are available, which is only an approximation of the truth. In practice, computer programs are only able to generate pseudo-random numbers, which are highly chaotic recursive sequences, but still deterministic. Also, these numbers are generated as integers, which only provide, after normalization, a distribution on a discretization of the unit interval. We will neglect these facts, however, and work as if the output of the function *random* (or any similar name) in a computer program is a true realization of the uniform distribution.

Sampling algorithms provide inference methods for graphical models that are applicable in a large variety of contexts. Unlike belief propagation and junction trees, they do not require that the underlying graph can be reduced to an acyclic structure. In counterpart, they generally are iterative methods, that do not converge in finite time, so that they can only – for most of them – provide approximations of marginal distributions, and they may require large amounts of computation to provide good ones.

Before describing sampling methods that are well adapted to graphical models, we start by a brief presentation of the basic sampling algorithm for distributions over finite sets.

6.1. Direct Sampling. Given a variable U, uniform on $[0, 1]$, it is conceptually very easy to generate a sample of any finite probability distribution π. Indeed, let Ω be a finite set, with elements ordered in some way, writing

$$\Omega = \{x_1, \ldots, x_n\}, \tag{66}$$

let π be a probability distribution on Ω and Z its associated cumulative distribution function

$$Z_k = \sum_{l=1}^{k} \pi(x_l)$$

with $Z_0 = 0$ and by construction $Z_n = 1$. Then, the random variable X defined by $X_k = x_k$ if and only if $U \in [Z_{k-1}, Z_k)$, $k \geq 1$ has distribution π. This is obvious, since the probability for $U \in [Z_{k-1}, Z_k)$ is $Z_k - Z_{k-1} = \pi(x_k)$.

This very simple algorithm can however become computationally prohibitive when the number of elements of Ω is very large, which is almost always the case with graphical models, for which Ω is a product of state spaces F_s and therefore grows exponentially with the size of the graph.

One possible alternative to this direct sampling method is to use rejection sampling. For this, one first selects a probability $q(x_k)$ that is easy to sample (for example, a uniform probability, or, with several random variables, a probability for which all variables are independent). One then iterate the following process:

1. Sample X according to q.
2. Sample a binary variable $\alpha \sim \text{bernoulli}(a(X))$, with $a(x) \in [0, 1]$. This function will be described below.
3. If $\alpha = 1$ (accept): stop; otherwise (reject): return to step 1.

Let $\mu = \sum_{l=1}^{n} q(x_l)a(x_l)$ be the probability of accepting X at step 3. The probability that $X = x_k$ at the end of the process (which terminates as soon as $\mu > 0$) is

$$q(x_k)a(x_k) \sum_{l=0}^{\infty} (1 - \mu)^l = q(x_k)a(x_k)/\mu$$

so that the process simulates π if and only if $qa/\mu = \pi$, i.e., a is proportional to π/q. To reduce computation, one should take a as large as possible, so that μ, in turn, is as large as possible. We of course need that $a = \mu\pi/q \leq 1$. Taking the largest possible choice lead to select $\mu = \inf_l q(x_l)/\pi(x_l)$ and

$$a(x_l) = \mu \frac{\pi(x_k)}{q(x_k)}.$$

Although this method only requires to compute as many $\pi(x_k)$'s as rejected attempts, it may still be limited for high-dimensional models because in that case, the acceptance probability, μ, is likely to be extremely small, yielding a very large number of trials before termination. Also, sometimes, π is simply not computable.

In fact, for all but very low-dimensional graphical models, there is no direct simulation method available (at the exception of "perfect sampling" methods which are applicable to a limited range of models, see section 6.3), and the feasible procedures, which are based on Markov chains, require an infinite time to provide exact samples from X. Such methods belong in the category of Markov Chain Monte Carlo (MCMC) methods.

6.2. Markov Chain Sampling.

6.2.1. *General Properties.* Without loss of generality (since one can always reduce the set Ω), we will assume, in this section, that $\pi(x) > 0$ for all $x \in \Omega$.

Providing a complete presentation of the theory of Markov chains, even in their simplest setting (finite state space) is out of the scope of these notes, and we will restrict to the few notions that are needed in order to understand and design basic sampling algorithms.

A Markov chain is the probabilistic analogous of a recursive sequence $X_{n+1} = F(X_n)$, which is fully defined by the function F and the initial value X_0. For Markov chains, X_0 is a random variable, which therefore does not have a fixed value, but follows a probability distribution that we will generally denote μ_0: $\mu_0(x) = P(X_0 = x)$. The computation of X_{n+1} given X_n is not deterministic either, but given by a transition probability

$$p^{n.n+1}(y|x) = P(X_{n+1} = y|X_n = x).$$

For finite sets, the transition probabilities are rather written in the form $p_{xy}^{n,n+1}$ instead of $p^{n.n+1}(y|x)$. When $p_{xy}^{n,n+1} = p_{xy}$ does not depend on n, the Markov chain is said to be homogeneous. To simplify notation, we will restrict to homogeneous chains, although some of those used in MCMC sampling may be inhomogeneous. This is not a very strong loss of generality, however, because inhomogeneous Markov chains can be considered as homogeneous by extending the space Ω on which they are defined to $\Omega \times \mathbb{N}$, and defining the transition probability

$$\tilde{p}((y,r)|(x,n)) = \delta_{r=n+1} p^{n,n+1}(y|x).$$

Considering that transitions are over the finite set Ω as in (66), we can consider p_{xy} as the coefficients of a matrix $\mathbb{P} = (p_{x_k x_l}, k, l = 1, \ldots, N)$. Such a matrix, which has non-negative entries and row sums equal to 1, is called a stochastic matrix. Of main interest for our purposes is the distribution of X_n when n tends to infinity. Denoting $\mu_n(x) = P(X_n = x)$, we get, from the law of total probabilities:

$$\mu_{n+1}(x) = \sum_{y \in \Omega} \mu_n(y) p_{yx}$$

which can be written in matrix form $\mu_{n+1} = \mu_n \mathbb{P}$ (μ_n is here considered as a row vector). Iterating backward to $n = 0$ yields $\mu_n = \mu_0 \mathbb{P}^n$ which gives a very simple expression of the distribution of X_n.

We will be interested in the limit behavior of μ_n. More precisely, we want to ensure that μ_n converges to π (the distribution we want to sample from) when n tends to infinity, independently of the choice that was made for μ_0.

A distribution μ_* is such that $\mu_0 P^n \to \mu_*$ *for some* μ_0 if and only if $\mu_* P = \mu_*$ (the proof is obvious). Such a distribution is said to be *invariant* by the Markov chain. Invariant distributions always exist. To see why this is true, consider the map $\mu \to \mu P$, defined over all probabilities on Ω. Since the set of probability distributions is compact, the continuous function $\mu \to |\mu - \mu P|_1$ (with $|\nu|_1 = \sum_x |\nu(x)|$) has a minimum, say at μ_*. But

$$\begin{aligned} |\mu P - \mu P^2|_1 &= \sum_{y \in \Omega} \Big| \sum_{x \in \Omega} (\mu_*(x) - \mu_* P(x)) p_{xy} \Big| \\ &\leq \sum_{x,y \in \Omega} |\mu_*(x) - (\mu_* P)(x)| p_{xy} \\ &= |\mu_* - \mu_* P|_1. \end{aligned}$$

and the inequality is strict unless, for each y, the numbers $((\mu_*(x) - \mu_* P(x)) p_{xy}, x \in \Omega)$ have identical signs. But since these numbers sum to 0, the only possibility if that they all vanish: μ_* is invariant.

An invariant distribution is, by definition, a left-eigenvector of P (i.e., an eigenvector of P^T) for the eigenvalue 1. Note that it is easy to justify that 1 is an eigenvalue for P, since the rows of P summing to 1 is equivalent to $P\mathbf{1} = \mathbf{1}$ where $\mathbf{1}$ is the column vector with all coefficients equal to 1. Another interesting fact is that no (real of complex) eigenvalue can have a modulus larger than 1. Let us justify this since the proof is not too difficult: if $Pz = \lambda z$, for a nonzero column vector z, then, for $x \in \Omega$

$$\begin{aligned} |\lambda| \, |z(x)| &\leq \sum_{y \in \Omega} p_{xy} |z(y)| \\ &\leq \max_{y \in \Omega} |z(y)| \end{aligned}$$

which implies that $|\lambda| \max_y |z(y)| \leq \max_y |z(y)|$ so that $|\lambda| \leq 1$.

The interesting situation for us is when 1 (with a single multiplicity) is the unique eigenvalue of P with modulus 1. In this case the invariant distribution, μ_*, is unique and there exist constants $C > 0$ and $\rho \in [0, 1)$ such that

$$|\mu_0 P^n - \mu_*|_1 \leq C \rho^n \tag{67}$$

for any initial distribution μ_0. Such stochastic matrices (and corresponding Markov chains) are called *ergodic*. They can be characterized with the following very simple criterion, stated without proof.

PROPOSITION 35. *A stochastic matrix P is ergodic if and only if there exists a integer n such that P^n has only positive coefficients.*

This statement can be rephrased as: there exists n such that the Markov chain with transition P can pass from any state to any other in n steps with positive probability. An

important fact is that the law of large numbers is true also for ergodic chains, which makes them useful to evaluate expectations.

THEOREM 5. *If P is an ergodic stochastic matrix, and π its invariant probability, then, if $X_0, X_1, \ldots$ is a Markov chain with transition P, and f a function on Ω,*

$$\lim_{n\to\infty} \frac{1}{n} \sum_{k=0}^{n} f(X_k) = E_\pi(f)$$

with probability 1.

Ergodic Markov chains provide one of the requirements we need for our sampling procedure, the convergence to a limit distribution. It remains to ensure that this distribution is the one we want: π. This is usually done by ensuring that the detailed balance condition is satisfied.

DEFINITION 26. *A stochastic matrix P is in detailed balance with respect to π if and only if:*

$$\forall x, y \in \Omega : \pi(x)p_{xy} = \pi(y)p_{yx}.$$

One also says that P is π-reversible.

The following proposition states that detailed balance is a sufficient condition for invariance.

PROPOSITION 36. *If P is in detailed balance with respect to π, then π is P-invariant.*

PROOF. Indeed, if P is in detailed balance, then

$$(\pi P)(x) = \sum_{y\in\Omega} \pi(y)p_{yx} = \sum_{y\in\Omega} \pi(x)p_{xy} = \pi(x)$$

so that π is invariant. □

So, building a valid MCMC sampling procedure for π can be done by finding an ergodic P such that the detailed balance condition is satisfied. Note that reversibility is only a sufficient condition for invariance: some distributions can be P-invariant without P being reversible. But it provides a condition that is so easy to verify that it is used in almost all practical cases.

The performance of the resulting chain, as an approximation method for π, is directly related to the size of ρ in equation (67), smaller ρ's obviously yielding faster convergence. Unfortunately, explicit and accurate evaluations of ρ are rarely available (evaluating speeds of convergence for Markov chains is still a major research subject in probability theory today).

6.2.2. *Metropolis-Hastings Scheme.* One simple way to build a stochastic matrix, P, which is π-reversible, is to use a rejection sampling rule similar to the one introduced for direct sampling, but with Markov chains. Given a state x, the sampling procedure is run by proposing a new state y with probability q_{xy}, and accepting it with probability $a(x, y)$

(keeping x otherwise). The transition probabilities for this process are $p_{xy} = q_{xy}a(x,y)$ if $x \neq y$ and $p_{xx} = 1 - \sum_{y \neq x} p_{xy}$. Detailed balance is ensured by

$$\pi(x)q_{xy}a(x,y) = \pi(y)q_{yx}a(y,x).$$

Since we assume that π is positive, and since there is no interest in having vanishing acceptance probabilities, we see that this condition requires, as a necessary condition on q (which is part of the design of the method), that the following weak symmetry condition is satisfied:

$$\forall x, y \in \Omega : q_{xy} = 0 \Leftrightarrow q_{yx} = 0. \tag{68}$$

Also, because $a(x,y)$ must be smaller than 1, we see that detailed balance requires that

$$a(y,x) \leq \frac{\pi(x)q_{xy}}{\pi(y)q_{yx}}.$$

(There is no need to define $a(y,x)$ when $q_{yx} = 0$.) One can in fact take $a(y,x)$ as large as possible given this constraint:

$$a(y,x) = \min\left(1, \frac{\pi(x)q_{xy}}{\pi(y)q_{yx}}\right).$$

This leads to what is known as the general Metropolis-Hastings procedure.

DEFINITION 27. *Let π be a positive probability distribution on a set Ω and q a transition probability satisfying the weak symmetry condition* (68)*. The Metropolis-Hastings sampling procedure generates a Markov chain by iterating the following steps, given that $x \in \Omega$ is the current state of the process.*

1. *Propose a new state y according to the transition probability q_{yx}.*
2. *Accept y as the new current state with probability*

$$a(x,y) = \min\left(1, \frac{\pi(y)q_{yx}}{\pi(x)q_{xy}}\right)$$

and keep x as current state otherwise.

The Metropolis-Hastings procedure therefore generates a Markov Chain with a π-reversible transition probability. Note that, when designing such a chain, the ergodicity condition provided by Proposition 35 must also be checked to ensure, for example, the validity of Theorem 5. This is true as soon as the proposal stochastic matrix, Q, is ergodic.

6.2.3. *Application to Graphical Models.* Let $G = (V, E)$ be an undirected graph and $\Omega = F_V$. The Metropolis-Hastings procedure is usually defined with local updating rules, for which a V-indexed family $q^s = (q^s_{xy}), s \in V$ of transition probabilities is designed, with the property that q^s only updates the value of the configuration at s; in other terms: if there exists $t \neq s$ such that $x_t \neq y_t$, then $q^s_{xy} = 0$. This is obviously consistent with the weak symmetry condition (68). For local updating rules, the acceptance probability is given by

$$\begin{aligned} a^s(x,y) &= \min\left(1, \frac{\pi(y)q^s_{yx}}{\pi(x)q^s_{xy}}\right) \\ &= \min\left(1, \frac{\pi_s(y_s|x_{s^c})q^s_{yx}}{\pi(x_s|x_{s^c})q^s_{xy}}\right) \end{aligned}$$

so that the transition only depend on the conditional probabilities at one vertex given the rest. The expression gets even simpler if one chooses q^s to be symmetric in x and y, for example $q^s_{xy} = 1/|F_s|$, which gives

$$a^s(x,y) = \min\left(1, \frac{\pi_s(y_s|x_{s^c})}{\pi(x_s|x_{s^c})}\right).$$

The transition is usually applied after sampling s at random from V, yielding the basic Metropolis algorithm:

DEFINITION 28. *The Metropolis algorithm for a distribution π on a product space $\Omega = F_V$ iterates the following steps, given the current $x \in \Omega$:*

1. *Pick, at random, a vertex s in V, according to the uniform distribution.*
2. *Propose a new value, y_s, for x_s, according to the uniform distribution on F_s.*
3. *Accept this value with probability* $\min(1, \pi_s(y_s|x_{s^c})/\pi(x_s|x_{s^c}))$.

This algorithm is in fact associated to the transition probability

$$p_{xy} = \frac{1}{|V|}\sum_{s\in V} p^s_{xy}$$

where $p^s = q^s a^s$ are single-vertex transitions. It is also possible to replace the random choice of s at step 1 with a systematic scan of V. The obtained Markov chain is not homogeneous anymore, but retain its limit properties, like in Theorem 5.

Another common choice is to take $q^s(x,y) = \pi_s(y_s|y_{s^c})$, which yields $a^s = 1$. This yields the Gibbs sampling, or heat-bath algorithm.

DEFINITION 29. *The Gibbs sampling (or heat bath) algorithm for a distribution π on a product space $\Omega = F_V$ iterates the following steps, given the current $x \in \Omega$:*

1. *Sample, at random, a vertex s in V, according to the uniform distribution.*
2. *Propose (and always accept) a new value, y_s, for x_s according to the distribution $\pi_s(.|x_{s^c})$ on F_s.*

6.3. Perfect Sampling. The problem with "forward" Markov chain methods, as described above, is that they do not provide exact samples from the distribution π, but only increasingly accurate approximations. Perfect sampling algorithms use Markov chains "backward" to achieve this result. To describe them, it is easier to describe a Markov chain as a stochastic recursive equation of the form

$$X_{n+1} = f(X_n, U_{n+1}) \tag{69}$$

where U_{n+1} is independent of $X_n, X_{n-1}, \ldots$, and the U_k's are identically distributed. Given a stochastic matrix P, one can take U_n to be the uniformly distributed variable used to sample from $(p_{X_n x}, x \in \Omega)$. Conversely, the transition probability associated to (69) is $p_{xy} = P(f(x,U) = y)$.

It will be convenient to consider negative times also. For $n > 0$, recursively define $F_{-n}(x, u_{-n+1}, \ldots, u_0)$ by

$$F_{-n-1}(x, u_{-n}, \ldots, u_0) = F_{-n}(f(x, u_{-n}), u_{-n+1}, \ldots, u_0)$$

and $F_{-1}(x, u_0) = f(x, u_0)$. Denote, for short, $u^0_{-n} = (U_{-n}, \ldots, U_0)$. The function $F_{-n}(x, u^0_{-n+1})$ provides the value of X_0 when $X_{-n} = x$ and $U^0_{-n+1} = u^0_{-n+1}$.

For an infinite past sequence, $u_{-\infty}^0$, let $\nu(u_{-\infty}^0)$ denote the first integer n such that $F_{-n}(x, u_{-n+1}^0)$ does not depend on x (the function "coalesces"). Then, the following theorem is true:

THEOREM 6. *Assume that the chain defined by* (69) *is ergodic, with invariant distribution* π. *Then* $\nu = \nu(U_{-\infty}^0)$ *is finite with probability 1, and*

$$X_* := F_{-\nu}(x, U_{-\nu+1}^0) \tag{70}$$

(which is independent of x*) has distribution* π.

PROOF. Because the chain is ergodic, we know that there exists an integer N such that one can pass from any state to any other with positive probability. So the chain can, starting from anywhere, coalesce with positive probability in N steps; ν being infinite would imply that this event never occurs in an infinite number of trials, and this has probability 0.

For any $k > 0$ and any $x \in \Omega$, we have

$$X_* = F_{-\nu}(f_{-k}(x, U_{-\nu-k+1}^{-\nu}), U_{-\nu+1}^0) = F_{-\nu-k}(x, U_{-\nu-k+1}^0). \tag{71}$$

But, because the chain is ergodic, we have, for any $x \in \Omega$

$$\lim_{k\to\infty} P(F_{-k}(x, U_{-k+1}^0) = y) = \pi(y).$$

We can write

$$\begin{aligned}
&P(F_{-k}(x, U_{-k+1}^0) = y) \\
&\quad = P(F_{-k}(x, U_{-k+1}^0) = y, \nu \le k) + P(F_{-k}(x, U_{-k+1}^0) = y, \nu > k) \\
&\quad = P(X_* = y, \nu \le k) + P(F_{-k}(x, U_{-k+1}^0) = y, \nu > k)
\end{aligned}$$

The right-hand side tends to $P(X_* = y)$ when k tends to infinity (because $P(\nu > k)$ tends to 0), and the left-hand side tends to $\pi(y)$, which gives the second part of the theorem. □

From equation (71), which is the key step in proving that X^* follows the invariant distribution, one can see why it is important to consider sampling that expands backward in time rather than forward. More specifically, consider the coalescence time for the forward chain, letting $\tilde{\nu}(u_0^\infty)$ be the first index for which

$$\tilde{X}_* := F_{\tilde{\nu}}(x, u_0^{\tilde{\nu}})$$

is independent from the starting point, x. For any $k \ge 0$, one still has the fact that $F_{\tilde{\nu}+k}(x, u_0^{\tilde{\nu}+k})$ does not depend on x, but its value depends on k and will not be equal to $\tilde{X}_*$ anymore, which prevents the rest of the proof of Theorem 6 to carry on.

An equivalent algorithm is described in the next proposition (the proof is easy and left to the reader).

PROPOSITION 37. *Using the same notation as above, the following algorithm generates a perfect sample,* ξ_*, *of the invariant distribution of an ergodic Markov chain.*

Assume that an infinite sample $u_{-\infty}^0$ *of* U *is available. Given this sequence, the algorithm, starting with* $t_0 = 2$, *is:*

1. *For all* $x \in \Omega$, *define* $\xi_{-t}^x, t = -t_0, \ldots, 0$ *by* $\xi_{-t_0}^x = x$ *and* $\xi_{-t+1}^x = f(\xi_{-t}^x, u_{-t+1})$.

2. *If ξ_0^x is constant (independent of x), let ξ_* be equal to this constant value and stop. Otherwise, return to step 1 replacing t_0 with $2t_0$.*

In practice, the u_{-k}'s are only generated if they are needed. But it is important to consider the sequence as fixed: once u_{-k} is generated, it must be stored (or identically regenerated, using the same seed) for further use. It is important to strengthen the fact that this algorithm works backward in time, in the sense that the first states of the sequence are not identical at each iteration, because they are generated using random numbers with indices further in the past.

Such an algorithm is not feasible when $|\Omega|$ is too large, since one would have to consider an intractable number of Markov chains (one for each $x \in \Omega$). So they do not directly apply to general graphical models, but there are, interestingly, some cases in which the constancy of ξ_0^x over all Ω can be decided from its constancy over a small subset of Ω.

One situation in which this is true is when the Markov chain is monotone, according to the following definition. Assume that Ω can be partially ordered, and that f in equation (69) is increasing in x, i.e.,

$$x \leq x' \Rightarrow \forall u, f(x,u) \leq f(x',u). \tag{72}$$

Let $\Omega_{\min}$ and $\Omega_{\max}$ be the set of minimal and maximal elements in Ω. Then the sequence coalesces for the algorithm above if and only if it coalesces over $\Omega_{\min} \cup \Omega_{\max}$. Indeed, any $x \in \Omega$ is smaller than some maximal element, and larger than some minimal element in Ω. By (72), these inequalities remain true at each step of the sampling process, which implies that when chains initialized with extremal elements coalesce, so do the other ones. Therefore, it suffices to run the algorithm with extremal configurations only.

One can rewrite (72) in terms of transition probabilities p_{xy}, assuming that U follows a uniform distribution on $[0,1]$ and, for all $x \in \Omega$, there exists a partition $(I_{xy}, y \in \Omega)$ of Ω, such that

$$f(x,u) = y \Leftrightarrow u \in I_{x,y}$$

and I_{xy} is an interval with length p_{xy}. Condition (72) is then equivalent to

$$x \leq x' \Rightarrow \forall y \in \Omega, I_{xy} \subset \bigcup_{y' \geq y} I_{x'y'}.$$

This requires in particular that $\sum_{y \geq y_0} p_{xy} \leq \sum_{y \geq y_0} p_{x'y}$ whenever $x \leq x'$ (one says that $p_x(.)$ is stochastically smaller than $p_{x'}(.)$). This does not, however, provide a sufficient condition.

One example in which this reduction works is with the ferromagnetic Ising model, for which $\Omega = \{-1,1\}^V$ and

$$\pi(x) = \frac{1}{Z} \exp\Big(\sum_{\{s,t\} \in E} \beta_{st} x_s x_t \Big)$$

with $\beta_{st} \geq 0$ for all $\{s,t\}$. Then, the heat bath sampling algorithm iterates the following steps: take a random $s \in V$ and update x_s according to the conditional distribution

$$\pi_s(y_s | x_{s^c}) = \frac{e^{y_s v_s(x)}}{e^{-v_s(x)} + e^{v_s(x)}}$$

with $v_s(x) = \sum_{t \in \mathcal{V}_s} \beta_{st} x_t$. Order Ω so that $x \leq x'$ if and only if $x_s \leq x'_s$ for all $s \in V$. The minimal and maximal elements are unique in this case, with $x_{\min,s} \equiv -1$ and $x_{\max,s} \equiv 1$.

Moreover, because all β_{st} are non-negative, v_s is an increasing function of x so that, if $x \leq x'$, then $\pi_s(1|x_s) \leq \pi_s(1|x'_s)$. To define the stochastic iterations, first introduce

$$f_s(x,u) = \begin{cases} 1_s \wedge x_{s^c} \text{ if } u \leq \pi_s(1|x_s) \\ (-1)_s \wedge x_{s^c} \text{ if } u > \pi_s(1|x_s), \end{cases}$$

which satisfies (72). The whole updating scheme can then be implemented with the function

$$f(x,(u,\tilde{u})) = \sum_{s \in V} \delta_{I_s}(\tilde{u}) f_s(x,u)$$

where $(I_s, s \in V)$ is any partition of $[0,1]$ in intervals of length $1/|V|$. This is still monotonic. The algorithm described in Proposition 37 can therefore be applied to sample exactly, in finite time, from the ferromagnetic Ising model.

CHAPTER 4

Bayesian Networks

1. Definitions

Bayesian networks are graphical models supported by directed acyclic graphs (DAG), which provide them with an ordered (causal), organization (directed graphs were introduced in Definition 12).

We first introduce some notation. Let $G = (V, E)$ be a directed acyclic graph. The parents of $s \in V$ are vertices t such that $(t, s) \in E$, and its children are t's such that $(s, t) \in E$. The set of parents of s is denoted s^-, and the set of its children is s^+, with $\mathcal{V}_s = s^+ \cup s^-$.

The vertices of G can be partially ordered, like with trees, by $s \leq_G t$ if and only if there exists a path going from s to t. Unlike trees, however, there can be more than one minimal element in V, and we still call roots vertices that have no parent, denoting

$$V_0 = \left\{ s \in V : s^- = \emptyset \right\}.$$

We also call leaves, or terminal nodes, vertices that have no children. Unless otherwise specified, we assume that all graphs are connected.

Bayesian networks over G are defined as follows.

DEFINITION 30. *A random variable $X = X_V$ is a Bayesian network over a DAG $G = (V, E)$ if and only if its distribution can be written in the form*

$$P^X(x) = \prod_{s \in V_0} p_s(x_s) \prod_{s \in V \setminus V_0} p_s(x_s | x_{s^-}) \tag{73}$$

where p_s is, for all $s \in V$, a probability distribution with respect to x_s.

Using the convention that conditional distributions given the empty set are just absolute distributions, we can rewrite (73) as

$$P^X(x) = \prod_{s \in V} p_s(x_s | x_{s^-}). \tag{74}$$

One can verify that $\sum_{x \in \Omega} P^X(x) = 1$ (otherwise the definition would not make sense). Indeed, when summing over x, we can start summing over all x_s with $s^+ = \emptyset$ (the leaves). Such x_s's only appear in the corresponding p_s's, which disappear since they sum to 1. What remains is the sum of the product over V minus the leaves, and the argument can be iterated until the remaining sum is 1 (alternatively, work by induction on $|V|$). This fact is also a consequence of Proposition 39 below, applied with $A = \emptyset$.

2. Conditional Independence Graph

2.1. Moral Graph. Bayesian networks have a conditional independence structure which is not exactly given by G, but can be deduced from it. Indeed, fixing a set $S \subset V$, we can see, when computing the probability of $X_S = x_s$ given $X_{S^c} = x_{S^c}$, which is

$$P_S^X(x_S|x_{S^c}) = \frac{1}{Z(x_{S^c})} \prod_{s \in V} p_s(x_s|x_{s^-}),$$

that the only variables $x_t, t \notin S$ that can be factorized in the normalizing constant are those that are neither parent or children of vertices in S, and do not share a child with a vertex in S (i.e., they intervene in no $p_s(x_s|x_{s^-})$ that also involves elements of S). This suggests the following definition.

DEFINITION 31. *Let G be a directed acyclic graph. We denote $G^\sharp = (V, E^\sharp)$ the undirected graph on V such that $\{s,t\} \in E^\sharp$ if one of the following conditions is satisfied*

- *Either $(s,t) \in E$ or $(t,s) \in E$.*
- *There exists $u \in V$ such that $(s,u) \in E$ and $(t,u) \in E$.*

$G^\sharp$ is sometimes called the *moral graph* of G (because it forces parents to marry!). A path in $G^\sharp$ can be visualized as a path in $G^\flat$ which is allowed to jump between parents of the same vertex even if they were not connected originally.

The previous discussion implies:

PROPOSITION 38. *Let X be a Bayesian network on G. We have*

$$(S \perp\!\!\!\perp T \,|\, U)_{G^\sharp} \Rightarrow (X_S \perp\!\!\!\perp X_T \,|\, X_U),$$

i.e., X is $G^\sharp$-Markov.

This proposition can be refined by noticing that the joint distribution of X_S, X_T, X_U can be deduced from a Bayesian network on a graph restricted to the ancestors of $S \cup T \cup U$. First, we define a restricted graph.

DEFINITION 32. *Let $G = (V, E)$ be a graph (directed or undirected), and $A \subset V$. The restricted graph $G_A = (A, E_A)$ is such that the elements of E_A are the edges (s,t) (or $\{s,t\}$) in E such that both s and t belong to A.*

Moreover, for a directed acyclic graph G and $s \in V$, we define the set of ancestors of s by

$$\mathcal{A}_s = \{t \in V, t \leq_G s\} \tag{75}$$

for the partial order on V induced by G.

If $S \subset V$, we denote $\mathcal{A}_S = \bigcup_{s \in S} \mathcal{A}_s$. Note that, by definition, $S \subset \mathcal{A}_S$. The following proposition is true.

PROPOSITION 39. *Let X be a Bayesian network on $G = (V, E)$ with distribution given by (74). Let $S \subset V$ and $A = \mathcal{A}_S$. Then the distribution of X_A is a Bayesian network over G_A given by*

$$P_A^X(x_A) = \prod_{s \in A} p_s(x_s|x_{s^-}). \tag{76}$$

Note that there is no ambiguity in the notation s^-, since the parents of $s \in A$ are the same in G_A as in G.

PROOF. One needs to show that

$$\prod_{s\in A} p_s(x_s|x_{s^-}) = \sum_{x_{A^c}} \prod_{s\in V} p_s(x_s|x_{s^-}).$$

This can be done by induction on the cardinality of V. Assume that the result is true for graphs of size n, and let $|V| = n+1$ (the result is obvious for graphs of size 1).

If $A = V$, there is nothing to prove, so assume that A^c is not empty. Then A^c must contain a leaf in G, since otherwise, A would contain all leaves and their ancestors which would imply that $A = V$.

If $s \in A^c$ is a leaf in G, one can remove the variable x_s from the sum, since it only appear in p_s and conditional probabilities sum to one. But one can now apply the induction assumption to the restriction of G to $V \setminus \{s\}$. □

Given Proposition 39, Proposition 38 can therefore be refined as follows.

PROPOSITION 40. *Let X be a Bayesian network on G. We have*

$$(S \perp\!\!\!\perp T \,|\, U)_{(G_{\mathcal{A}_{S\cup T\cup U}})^\sharp} \Rightarrow (X_S \perp\!\!\!\perp X_T \,|\, X_U).$$

Proposition 39 is also used in the proof of the following proposition, which justifies the notation representing the p_s's as conditional distributions.

PROPOSITION 41. *Let $G = (V, E)$ be a directed acyclic graph, and X be a Bayesian network over G. Then, for all $s \in S$*

$$P(X_s = x_s|X_{\mathcal{A}_s\setminus\{s\}} = x_{\mathcal{A}_s\setminus\{s\}}) = P(X_s = x_s|X_{s^-} = x_{s^-}) = p_s(x_s|x_{s^-}).$$

PROOF. By Proposition 39, we can without loss of generality assume that $V = \mathcal{A}_s$. Then

$$\begin{aligned} P(X_s = x_s|X_{\mathcal{A}_s\setminus\{s\}} = x_{\mathcal{A}_s\setminus\{s\}}) &\propto P(X_{\mathcal{A}_s} = x_{\mathcal{A}_s}) \\ &= p_s(x_s|x_{s^-})Z(x_{\mathcal{A}_s\setminus\{s\}}) \end{aligned}$$

where

$$Z(x_{\mathcal{A}_s\setminus\{s\}}) = \prod_{t\in\mathcal{A}_s\setminus\{s\}} p_t(x_t|x_{t^-})$$

disappears when the conditional probability is normalized. □

2.2. Reduction to d-Separation. We now want to reformulate Proposition 40 in terms of the unoriented graph $G^\flat$ and specific features in G called v-junctions, that we now define.

DEFINITION 33. *Let $G = (V, E)$ be a directed graph. A v-junction is a triple of distinct vertices, $(s, t, u) \in V^3$ such that $\{s, u\} \subset t^-$ (i.e., s and u are parents of t).*

We will say that a path $(s_1, \ldots, s_N)$ in $G^\flat$ passes at $s = s_k$ with a v-junction if (s_{k-1}, s_k, s_{k+1}) is a v-junction in G.

We have the lemma:

LEMMA 7. *Two vertices s and t in G are separated by a set U in $(G_{\mathcal{A}_{\{s,t\}\cup U}})^\sharp$ if and only if any path between s and t in $G^\flat$ must either*

(1) *Pass at a vertex in U without a v-junction.*
(2) *Pass in $V \setminus \mathcal{A}_{\{s,t\}\cup U}$ at a v-junction.*

PROOF.
Step 1. We first note that the v-junction clause is redundant in (2). It can be removed without affecting the condition. Indeed, if a path in $G^\flat$ passes in $V \setminus \mathcal{A}_{\{s,t\}\cup U}$ one can follow this path downward (i.e., following the orientation in G) until a v-junction is met. This has to happen before reaching the extremities of the path, since u would be an ancestor of s or t otherwise. We can therefore work with the weaker condition (that we will denote (2)') in the rest of proof.

Step 2. Assume that U separates s and t in $(G_{\mathcal{A}_{\{s,t\}\cup U}})^\sharp$. Take a path γ between s and t in $G^\flat$. We need to show that the path satisfies (1) or (2)'. So assume that (2)' is false (otherwise we are done) so that γ is included in $\mathcal{A}_{\{s,t\}\cup U}$. We can modify γ by removing all the central nodes in v-junctions and still keep a valid path in $(G_{\mathcal{A}_{\{s,t\}\cup U}})^\sharp$ (since parents are connected in the moral graph). The remaining path must intersect U by assumption, and this cannot be at a v-junction in γ since we have removed them. So (1) is true.

Step 3. Conversely, assume that (1) or (2) is true for any path in $G^\flat$. Consider a path γ in $(G_{\mathcal{A}_{\{s,t\}\cup U}})^\sharp$ between s and t. Any edge in γ that is not in $G^\flat$ must involve parents of a common child in $\mathcal{A}_{\{s,t\}\cup U}$. Insert this child between the parents every time this occurs, resulting in a v-junction added to γ. Since the added vertices are still in $\mathcal{A}_{\{s,t\}\cup U}$, the new path still has no intersection with $V \setminus \mathcal{A}_{\{s,t\}\cup U}$ and must therefore satisfy (1). So there must be an intersection with U without a v-junction, and since the new additions are all at v-junctions, the intersection must have been originally in γ, which therefore passes in U. This shows that U separates s and t in $(G_{\mathcal{A}_{\{s,t\}\cup U}})^\sharp$. □

Condition (2) can be further restricted to provide the notion of d-separation.

DEFINITION 34. *One says that two vertices s and t in G are d-separated by a set U if and only if any path between s and t in $G^\flat$ must either*
(D1) *Pass at a vertex in U without a v-junction.*
(D2) *Pass in $V \setminus \mathcal{A}_U$ with a v-junction.*

Then we have:

THEOREM 7. *Two vertices s and t in G are separated by a set U in $(G_{\mathcal{A}_{\{s,t\}\cup U}})^\sharp$ if and only if they are d-separated by U.*

PROOF. It suffices to show that if condition ((D1) or (D2)) holds for any path between s and t in $G^\flat$, then so does ((1) or (2)). So take a path between s and t: if (D1) is true for this path, the conclusion is obvious, since (D1) and (1) are the same. So assume that (D1) (and therefore (1)) is false and that (D2) is true. Let u be a vertex in $V \setminus \mathcal{A}_U$ at which γ passes with a v-junction.

Assume that (2) is false. Then u must be an ancestor of either s or t. Say it is an ancestor of s: there is a path in G going from u to s without passing by U (otherwise u would be an ancestor of U); one can replace the portion of the old path between s and u by this new one, which does not pass by u with a v-junction anymore. So the new path still does not

satisfy (D1) and must satisfy (D2). Keep on removing all intersections with ancestors of s and t that have v-junctions to finally obtain a path that satisfies neither (D1) or (D2) and a contradiction to the fact that s and t are d-separated by U. □

3. Chain Graph Representation

The d-separability property involves both unoriented and oriented edges. It is in fact a property of the hybrid graph in which the orientation is removed from the edges that are not involved in a v-junction, and retained otherwise. Such graphs are particular instances of chain graphs.

DEFINITION 35. *A chain graph $G = (V, E, \tilde{E})$ is composed with a finite set V of vertices, a set $E \subset \boldsymbol{\mathcal{P}}_2(V)$ of unoriented edges and a set $\tilde{E} \subset E \times E \setminus \{(t,t), t \in E\}$ of oriented edges with the property that $E \cap \tilde{E}^\flat = \emptyset$, i.e., two vertices cannot be linked by both an oriented and an unoriented edge.*

A path in a chain graph is a a sequence of vertices $s_0, \ldots, s_N$ such that for all $k \geq 1$, s_{k-1} and s_k form an edge, which means that either $\{s_{k-1}, s_k\} \in E$ or $(s_{k-1}, s_k) \in \tilde{E}$.

A chain graph is acyclic if it contains no loop. It is semi-acyclic if it contains no loop containing oriented edges.

We start with the following equivalence relation within vertices in a semi-acyclic chain graph.

PROPOSITION 42. *Let $G = (V, E, \tilde{E})$ be a semi-acyclic chain graph. Define the relation $s \mathcal{R} t$ if and only if there exists a path in the unoriented subgraph (V, E) that links s and t. Then $\mathcal{R}$ is an equivalence relation.*

The proposition is obvious. This relation partitions V in equivalence classes, the set of which being denoted $V_{\mathcal{R}}$. If $S \in V_{\mathcal{R}}$, then any pair s, t in S is related by an unoriented path, and if $S \neq S' \in V_{\mathcal{R}}$, no elements $s \in S$ and $t \in S'$ can be related by such a path.

Moreover, no path in G between two elements of $S \in V_{\mathcal{R}}$, can contain a directed edge, since these elements must also be related by an undirected path, and this would create a loop in G containing an undirected edge. So the restriction of G to S is an undirected graph.

One can define a directed graph over equivalence classes as follows. Let $G_{\mathcal{R}} = (V_{\mathcal{R}}, E_{\mathcal{R}})$ be such that $(S, S') \in E_{\mathcal{R}}$ if and only if there exists $s \in S$ and $t \in S'$ such that $(s, t) \in \tilde{E}$. The graph $G_{\mathcal{R}}$ is acyclic: any loop in $G_{\mathcal{R}}$ would induce a loop in G containing at least one oriented edge.

We now can formally define a probability distribution on a semi-acyclic chain graph.

DEFINITION 36. *Let $G = (V, E, \tilde{E})$ be a semi-acyclic chain graph. One says that a random variable $X = X_V$ decomposes on G if and only if: $(X_S, S \in V_{\mathcal{R}})$ is a Bayesian network on $G_{\mathcal{R}}$ and the conditional distribution of X_S given $X_{S'}, S' \in S^-$ is G_S-Markov, such that, for $s \in S$, $P(X_s = x_s | X_t, t \in S, X_{S'}, S' \in S^-)$ only depends on x_t with $\{s, t\} \in E$ or $(t, s) \in \tilde{E}$.*

Returning to our discussion on Bayesian networks, we have the following. Associate to a DAG $G = (V, E)$ the chain graph $G^\dagger = (V, E^\dagger, \tilde{E}^\dagger)$ defined by: $\{s, t\} \in E^\dagger$ if and only if (s, t) or $(t, s) \in E$ and is not involved in a v-junction, and $(s, t) \in \tilde{E}^\dagger$ if $(s, t) \in E$ and is involved in a v-junction. This graph is acyclic; indeed, take any loop in $G^\dagger$: when its edges

are given their original orientations in E, the sequence cannot contain a v-junction, since the orientation in v-junctions are kept in $G^\dagger$; the path therefore constitutes a loop in G which is a contradiction.

All, excepted at most one, vertices in an equivalence class $S \in G^\dagger_{\mathcal{R}}$ have all their parents in S. Indeed, assume that two vertices, s and t, in S have parents outside of S. There exists an unoriented path, $s_0 = s, s_1, \ldots, s_N = t$, in $G^\dagger$ connecting them, since they belong to the same equivalence class. The edge at s must be oriented from s to s_1 in G, since otherwise s_1 would be a second parent to s in G, creating a v-junction, and the edge would have remained oriented in $G^\dagger$. Similarly, the last edge in the path must be oriented from t to s_{N-1} in G. But this implies that there exists a v-junction in the original orientation along the path, which cannot be constituted with only unoriented edges in $G^\dagger$. So we get a contradiction.

Thus, random variables that decompose on $G^\dagger$ are "Bayesian networks" of acyclic graphs, or trees since we know these are equivalent. The root of each tree must have multiple (vertex) parents in the parent tree in $G_{\mathcal{R}}$. The following theorem states that all Bayesian networks are equivalent to such a process.

THEOREM 8. *Let $G = (V, E)$ be a DAG. The random variable X is a Bayesian network on G if and only if it decomposes over $G^\dagger$.*

PROOF. Assume that X is a Bayesian network on G. We can obviously rewrite the probability distribution of X in the form

$$\pi(x) = \prod_{S \in G^\dagger_{\mathcal{R}}} \prod_{s \in S} p_s(x_s | x_{s^-}).$$

Since every vertex in S has its parents in S or in $\bigcup_{T \in S^-} T$, this *a fortiori* takes the form

$$\pi(x) = \prod_{S \in G^\dagger_{\mathcal{R}}} p_S(x_S | x_T, T \in S^-).$$

So $X_S, S \in V_{\mathcal{R}}$ is a Bayesian network. Moreover,

$$p_S(x_S | x_T, T \in S^-) = \prod_{s \in S} p_s(x_s | x_{s^-})$$

is a tree distribution with the required form of the individual conditional distributions.

Now assume that X decomposes on $G^\dagger$. Then the conditional distribution of X_S given $X_T, T \in S^-$ is Markov for the acyclic undirected graph G_S, and can therefore be expressed as a tree distribution consistent with the orientation of G. □

4. Markov Equivalence

While the previous discussion provides a rather simple description of Bayesian networks in terms of chain graphs, it does not go all the way in reducing the number of oriented edges in the definition of a Bayesian network. The issue is, in some way, addressed by the notion of Markov equivalence, which is defined as follows.

DEFINITION 37. *Two directed acyclic graphs on the same set of vertices $G = (V, E)$ and $\tilde{G} = (V, \tilde{E})$ are Markov-equivalent if any family of random variables that decomposes as a (positive) Bayesian network over one of them also decomposes as a Bayesian network over the other.*

The notion of Markov equivalence is exactly described by d-separation. This is stated in the following theorem, due to Geiger and Pearl, that we state without proof.

THEOREM 9. *G and $\tilde{G}$ are Markov equivalent if and only if, whenever two vertices are d-separated by a set in one of them, the same separation is true with the other.*

This property can be expressed in a strikingly simple condition. One says that a v-junction (s,t,u) in a DAG is *unlinked* if s and u are not neighbors.

THEOREM 10. *G and $\tilde{G}$ are Markov equivalent if and only if $G^\flat = \tilde{G}^\flat$ and both graphs have the same unlinked v-junctions.*

PROOF. *Step 1.* We first show that a given pair of vertices in a DAG is unlinked if and only if it can be d-separated by some set in the graph. Clearly, if they are linked, they cannot be d-separated (which is the "if" part), so what really needs to be proved is that unlinked vertices can be d-separated. Let s and t be these vertices and let $U = \mathcal{A}_{\{s,t\}} \setminus \{s,t\}$. Then U d-separates s and t since any path between s and t in $(G_{\mathcal{A}_{\{s,t\}\cup U}})^\sharp = (G_{\mathcal{A}_{\{s,t\}}})^\sharp$ must obviously pass in U.

Step 2. Let's now prove the only-if part of Theorem 10 and therefore assume that G and $\tilde{G}$ are Markov equivalent, or, as stated in Theorem 9, that d-separation coincides in G and $\tilde{G}$. We want to prove that $G^\flat = \tilde{G}^\flat$ and unlinked v-junctions are the same.

Step 2.1. The first statement is obvious from Step 1: d-separation determines the existence of a link, so if d-separation coincides in the two graphs, then the same holds for links and $G^\flat = \tilde{G}^\flat$.

Step 2.2. So let us proceed to the second statement and let (s,t,u) be an unlinked v-junction in G. We want to show that it is also a v-junction in $\tilde{G}$ (obviously unlinked since links coincide).

We will denote by $\tilde{\mathcal{A}}_S$ the ancestors of some set $S \subset V$ in $\tilde{G}$ (while $\mathcal{A}_S$ still denotes its ancestors in G). Let $U = \mathcal{A}_{\{s,u\}} \setminus \{s,u\}$. Then, as we have shown in Step 1, U d-separates s and u in G, so that, by assumption it also d-separates them in $\tilde{G}$.

We know that $t \notin U$, because it cannot be both a child and an ancestor of $\{s,u\}$ in G (this would induce a loop). The path (s,t,u) links s and u and does not pass in U, which is only possible (since U d-separates s and t in $\tilde{G}$) if it passes in $V - \tilde{\mathcal{A}}_U$ at a v-junction: so (s,t,u) is a v-junction in $\tilde{G}$, which is what we wanted to prove.

Step 3. We now consider the converse statement and assume that $G^\flat = \tilde{G}^\flat$ and unlinked v-junctions coincide. We want to show that d-separation is the same in G and $\tilde{G}$. So, we assume that U d-separates s and t in G, and we want to show that the same is true in $\tilde{G}$. Thus, what we need to prove is:

Claim 1 Consider a path γ between s and t in $\tilde{G}^\flat = G^\flat$. Then γ either (D1) passes in U without a v-junction in $\tilde{G}$, or (D2) in $V \setminus \tilde{\mathcal{A}}_U$ with a v-junction in $\tilde{G}$.

We will prove Claim 1 using a series of lemmas. We say that γ has a three-point loop at u if (v,u,w) are three consecutive points in γ such that v and w are linked. So (v,u,w,v) forms a loop in the undirected graph.

LEMMA 8. *If γ is a path between s and t that does not satisfy (D2) for G and passes in U without three-point loops, then γ satisfies (D1) for $\tilde{G}$.*

The proof is easy: since γ does not satisfy (D2) in G, it satisfies (D1) and passes in U without a v-junction in G. But this intersection cannot be a v-junction in $\tilde{G}$ since it would otherwise have to be linked and constitute a three-point loop in γ, which proves that (D1) is true for γ in $\tilde{G}$.

The next step is to remove the three-point loop condition in Lemma 8. This will be done using the next two results.

LEMMA 9. *Let γ be a path with a three-point loop at $u \in U$ for G. Assume that $\gamma \setminus u$ (which is a valid path in $G^\flat$) satisfies (D1) or (D2) in $\tilde{G}$. Then γ satisfies (D1) or (D2) in $\tilde{G}$.*

To prove the lemma, let v and w be the predecessor and successor of u in γ. First assume that $\gamma \setminus u$ satisfies (D1) in $\tilde{G}$. If this does not happen at v or at w, then this will apply also to γ and we're done, so let's assume that $v \in U$ and that (v', v, w) is not a v-junction in $\tilde{G}$, where v' is the predecessor of v. If (v', v, u) is not a v-junction in $\tilde{G}$, then (D1) is true for γ in $\tilde{G}$. If it is a v-junction, then (v, u, w) is not and (D1) is true too.

Assume now that (D2) is true for $\gamma \setminus u$ in $\tilde{G}$. Again, there is no problem if (D2) occurs for some point other than v or w, so let's consider the case for which it happens at v. This means that $v \notin \tilde{\mathcal{A}}_U$ and (v', v, w) is a v-junction. But, since $u \in U$, the link between u and v must be from u to v in $\tilde{G}$ so that there is no v-junction at u and (D1) is true in $\tilde{G}$. This proves Lemma 9.

LEMMA 10. *Let γ be a path with a three-point loop at $u \in U$ for G. Assume that γ does not satisfy (D2) in G. Then $\gamma \setminus u$ does not satisfy this property either.*

Let's assume that $\gamma \setminus u$ satisfies (D2) and reach a contradiction. Letting (v, u, w) be the three-point loop, (D2) can only happen in $\gamma \setminus u$ at v or w, and let's assume that this happens at v, so that, v' being the predecessor of v, (v', v, w) is a v-junction in G with $v \notin \mathcal{A}_U$. Since $v \notin \mathcal{A}_U$, the link between u and v in G must be from u to v, but this implies that (v', v, u) is a v-junction in G with $v \notin \mathcal{A}_U$ which is a contradiction: this proves Lemma 10.

The previous three lemmas directly imply the next one.

LEMMA 11. *If γ is a path between s and t that does not satisfy (D2) for G, then γ satisfies (D1) or (D2) for $\tilde{G}$.*

Indeed, if we start with γ that does not satisfy (D2) for G, Lemma 10 allows us to progressively remove three-point loops from γ until none remains with a final path that satisfies the assumptions of Lemma 8 and therefore satisfies (D1) in $\tilde{G}$, and Lemma 9 allows us to add the points that we have removed in reverse order while always satisfying (D1) or (D2) in $\tilde{G}$.

We now partially relax the hypothesis that (D2) is not satisfied with the next lemma.

LEMMA 12. *If γ is a path between s and t that does not pass in $V \setminus \mathcal{A}_U$ at a linked v-junction for G, then γ satisfies (D1) or (D2) for $\tilde{G}$.*

Assume that γ does not satisfy (D2) for $\tilde{G}$ (otherwise the result is proved). By Lemma 11, γ must satisfy (D2) for G. So, take an intersection of γ with $V \setminus \mathcal{A}_U$ that occurs at a v-junction in G, that we will denote (v, u, w). This is still a v-junction in $\tilde{G}$ since we assume it to be unlinked. Since (D2) is false in $\tilde{G}$, we must have $u \in \tilde{\mathcal{A}}_U$, and there is an oriented path, τ, from u to U in $\tilde{G}$.

We can assume that τ has no v-junction in G. If a v-junction exists in τ, then this v-junction must be linked (otherwise this would also be a v-junction in $\tilde{G}$ and contradict the fact that τ is consistently oriented in $\tilde{G}$), and this link must be oriented from u to U in $\tilde{G}$ to avoid creating a loop in this graph. This implies that we can bypass the v-junction while keeping a consistently oriented path in $\tilde{G}$, and iterate this until τ has no v-junction in G. But this implies that τ is consistently oriented in G, necessarily from U to u since $u \notin \mathcal{A}_U$.

Denote $\tau = (u_0 = u, v_1, \dots, u_n \in U)$. We now prove by induction that each (v, u_k, w) is an unlinked v-junction. This is true when $k = 0$, and let's assume that it is true for $k - 1$. Then (u_k, u_{k-1}, v) is a v-junction in G but not in $\tilde{G}$: so it must be linked and there exists an edge between v and u_k. In $\tilde{G}$, this edge must be oriented from v to u_k, since (v, u_{k-1}, u_k, v) would form a loop otherwise. For the same reason, there must be an edge in $\tilde{G}$ from w to u_k so that (v, u_k, w) is an unlinked v-junction.

Since this is true for $k = n$, we can replace u by u_n in γ and still obtain a valid path. This can be done for all intersections of γ with $V \setminus \mathcal{A}_U$ that occur at v-junctions. This finally yields a path (denote it $\bar{\gamma}$) which does not satisfy (D2) in G anymore, and therefore satisfies (D1) or (D2) in $\tilde{G}$: so $\bar{\gamma}$ must either pass in U without a v-junction or in $V \setminus \tilde{A}_U$ at a v-junction. None of the nodes that were modified can satisfy any of these conditions, since they were all in U with a v-junction, so that the result is true for the original γ also. This proves Lemma 12.

So the only unsolved case is when γ is allowed to pass in $V \setminus \mathcal{A}_U$ at linked v-junctions. We define an algorithm that removes them as follows. Let $\gamma_0 = \gamma$ and let γ_k be the path after step k of the algorithm. One passes from γ_k to γ_{k+1} as follows.

- If γ_k has no linked v-junctions in $V \setminus \mathcal{A}_U$ for G, stop.
- Otherwise, pick such a v-junction and let (v, u, w) be the three nodes involved in it.
  - (i) If $v \in U, v' \notin U$ and (v', v, u) is a v-junction in $\tilde{G}$, remove v from γ_k to define γ_{k+1}.
  - (ii) Otherwise, if $w \in U, w' \notin U$ and (u, w, w') is a v-junction in $\tilde{G}$, remove w from γ_k to define γ_{k+1}.
  - (iii) Otherwise, remove u from γ_k to define γ_{k+1}.

None of the considered cases can disconnect the path. This is clear for case (iii) since v and w are linked. For case (i), note that, in G, (v', v, u) cannot be a v-junction since (v, u, w) is one. This implies that the v-junction in $\tilde{G}$ must be linked and that v' and u are connected.

The algorithm will stop at some point with some γ_n that does not have any linked v-junction in $V \setminus \mathcal{A}_U$ anymore, which implies that (D1) or (D2) is true in $\tilde{G}$ for γ_n. To prove that this statement holds for γ, it suffices to show that if (D1) or (D2) is true in $\tilde{G}$ with

γ_{k+1}, it must have been true with γ_k at each step of the algorithm. So let's assume that γ_{k+1} satisfies (D1) or (D2) in $\tilde{G}$.

First assume that we passed from γ_k to γ_{k+1} via case (iii). Assume that (D2) is true for γ_{k+1}, with as usual the only interesting case being when this occurs at v or w. Assume it occurs at v so that (v', v, w) is a v-junction and $v \notin \tilde{\mathcal{A}}_U$. If (v', v, u) is a v-junction, then (D2) is true with γ_k. Otherwise, there is an edge from v to u in $\tilde{G}$ which also implies an edge from w to u since (v, u, w, v) would be a loop otherwise. So (v, u, w) is a v-junction in $\tilde{G}$, and u cannot be in $\tilde{\mathcal{A}}_U$ since its parent, v would be in that set also. So (D2) is true in $\tilde{G}$. Now, assume that (D1) is true at v, so that (v', v, w) is not a v-junction and $v \in U$. If (v', v, u) is not a v-junction either, we are done, so assume the contrary. If $v' \in U$, then we cannot have a v-junction at v' and (D1) is true. But $v' \notin U$ is not possible since this leads to case (i).

Now assume that we passed from γ_k to γ_{k+1} via case (i). Assume that (D1) is true for γ_k: this cannot be at v' since $v' \notin U$, neither at u since $u \notin \mathcal{A}_U$, so it will also be true for γ_{k+1}. The same statement holds with (D2) since (v', v, u) is a v-junction in $\tilde{G}$ with $v \in U$ which implies that both v' and u are in $\tilde{\mathcal{A}}_U$. Case (ii) is obviously addressed similarly.

With this, the proof of Theorem 10 is complete.

□

5. Probabilistic Inference

5.1. Sum-prod Algorithm. We now discuss the issue of using the sum-prod algorithm to compute marginal probabilities, $P_s^X(x_s)$ for $s \in V$ when X is a Bayesian network on $G = (V, E)$. By definition, $P^X(x_V)$ can be written in the form

$$P^X(x_V) = \prod_{C \in \mathcal{C}} \varphi_C(x_C)$$

where $\mathcal{C}$ contains all subsets $C_s := \{s\} \cup s^-$, $s \in V$. Marginal probabilities can therefore be computed easily when the factor graph associated to $\mathcal{C}$ is acyclic, according to Proposition 29. However, because of the specific form of the φ_C's (they are conditional probabilities), the sum-prod algorithm can be analyzed in more detail, and provide correct results even when the factor graph is not acyclic.

The general rules for the sum-prod algorithm are

$$\begin{cases} m_{sC}(x_s) \leftarrow \prod_{\tilde{C}, s \in \tilde{C}, \tilde{C} \neq C} m_{\tilde{C}s}(x_s) \\ m_{Cs}(x_s) \leftarrow \sum_{y_C : y_s = x_s} \varphi_C(y_C) \prod_{t \in C \setminus \{s\}} m_{tC}(y_t) \end{cases}$$

They take a particular form for Bayesian networks, using the fact that a vertex s belongs to C_s, and to all C_t for $t \in s^+$.

$$
\begin{aligned}
m_{sC_s}(x_s) &\leftarrow \prod_{t\in s^+} m_{C_t s}(x_s), \\
m_{sC_t}(x_s) &\leftarrow m_{C_s s}(x_s) \prod_{u\in s^+, u\neq t} m_{C_u s}(x_s), \text{ for } t \in s^+, \\
m_{C_s s}(x_s) &\leftarrow \sum_{y_{C_s}, y_s = x_s} p_s(x_s|y_{s^-}) \prod_{t\in s^-} m_{tC_s}(y_t), \\
m_{C_t s}(x_s) &\leftarrow \sum_{y_{C_t}, y_s = x_s} p_t(y_t|x_s, y_u, u \in s^-, u \neq t) m_{tC_t}(y_t) \prod_{u\in s^-, u\neq t} m_{uC_t}(y_u), \\
&\quad \text{for } t \in s^+.
\end{aligned}
$$

These relations imply that, if $s^- = \emptyset$ (s is a root), then $m_{C_s s} = p_s(x_s)$. Also, if $s^+ = \emptyset$ (s is a leaf) then $m_{sC_s} = 1$. The following proposition shows that many of the messages become constant over time.

PROPOSITION 43. *All upward messages, m_{sC_s} and $m_{C_t s}$ with $t \in s^+$ become constant (independent from x_s) in finite time.*

PROOF. This can be shown recursively as follows. Assume that, for a given s, m_{tC_t} is constant for all $t \in s^+$ (this is true if s is a leaf). Then,

$$
\begin{aligned}
m_{C_t s}(x_s) &\leftarrow \sum_{y_{C_t}, y_s = x_s} p_t(y_t|x_s, y_u, u \in s^-, u \neq t) m_{tC_t}(y_t) \prod_{u\in s^-, u\neq t} m_{uC_t}(y_u), \\
&= m_{tC_t} \sum_{y_{C_t}, y_s = x_s} p_t(y_t|x_s, y_u, u \in s^-, u \neq t) \prod_{u\in s^-, u\neq t} m_{uC_t}(y_u) \\
&= m_{tC_t} \sum_{y_{C_t\setminus\{t\}}, y_s = x_s} \prod_{u\in s^-, u\neq t} m_{uC_t}(y_u) \\
&= m_{tC_t} \prod_{u\in s^-, u\neq t} \sum_{y_u} m_{uC_t}(y_u)
\end{aligned}
$$

which is constant. Now

$$
m_{sC_s}(x_s) \leftarrow \prod_{t\in s^+} m_{C_t s}(x_s)
$$

is also constant. This proves that all m_{sC_s} progressively become constant, and, as we have just seen, this implies the same property for $m_{C_t s}$, $t \in s^+$.

□

This proposition implies that, if initialized with constant messages (or after a finite time), the sum-prod algorithm iterates

$$\begin{aligned} m_{sC_s} &\leftarrow \prod_{t\in s^+} m_{C_t s} \\ m_{C_s s}(x_s) &\leftarrow \sum_{y_{C_s}, y_s = x_s} p_s(x_s|y_{s^-}) \prod_{t\in s^-} m_{tC_s}(y_t) \\ m_{sC_t}(x_s) &\leftarrow m_{C_s s}(x_s) \prod_{u\in s^+, u\neq t} m_{C_u s}, \quad t \in s^+ \\ m_{C_t s} &\leftarrow m_{sC_s} \prod_{u\in t^-, u\neq s} \sum_{y_u} m_{uC_s}(y_u), \quad t \in s^+. \end{aligned}$$

From this expression, we can conclude

PROPOSITION 44. *If the previous algorithm is first initialized with upward messages, $m_{sC_s} = m_{C_t s}$ all equal to 1, and if downward messages are computed top down from the roots to the leaves, the obtained configuration of messages is invariant for the sum-prod algorithm.*

PROOF. If all upward messages are equal to 1, then clearly, the downward messages sum to 1 once they are updated from roots to leaves, and this implies that the upward messages will remain equal to 1 for the next round. The obtained configuration is invariant since the downward messages are recursively uniquely defined by their value at the roots. □

The downward messages, under the previous assumptions, satisfy $m_{sC_t}(x_s) = m_{C_s s}(x_s)$ for all $t \in s^+$ and therefore

$$m_{C_s s}(x_s) = \sum_{y_{C_s}, y_s = x_s} \pi(x_s|y_{s^-}) \prod_{t\in s^-} m_{C_t t}(y_t). \tag{77}$$

Note that the associated "marginals" inferred by the sum-prod algorithm are

$$\sigma_s(x_s) = \prod_{C, s\in C} m_{Cs}(x_s) = m_{C_s s}(x_s)$$

since $m_{C_t s}(x_s) = 1$ when $t \in s^+$.

Although the sum-prod algorithm initialized with unit messages converges to a stable configuration if run top-down, the obtained σ_s's do not necessarily provide the correct single site marginals. There is a situation for which this is true, however, which is when the initial directed graph is singly connected, as we will see below. Before this, let's analyze the complexity resulting from an iterative computation of the marginal probabilities, similar to what we have done with trees.

We define the depth of a vertex in G as follows.

DEFINITION 38. *Let $G = (V, E)$ be a DAG. The depth of a vertex s in V is defined recursively by*

- $\text{depth}(s) = 0$ *if s has no parent.*
- $\text{depth}(s) = 1 + \max\big(\text{depth}(t), t \in s^-\big)$ *otherwise.*

The recursive computation of marginal distributions is made possible (although not always feasible) with the following remark.

LEMMA 13. *Let X be a Bayesian network on the DAG $G = (V, E)$, and $S \subset V$, such that all elements in S have the same depth. Let S^- be the set of parents of elements in S, and $T = \mathrm{depth}^-(S)$ the set of vertices in V with depth strictly smaller than the depth of S. Then $(X_S \perp\!\!\!\perp X_{T \setminus S^-} \,|\, X_{S^-})$ and the variables $X_s, s \in S$ are conditionally independent given X_{S^-}.*

PROOF. It suffices to show that vertices in S are separated from $T \setminus S^-$ and from other elements of S by S^- for the graph $(G_{S \cup T})^\sharp$. Any path starting at $s \in S$ must either pass by a parent of s (which is what we want), or by one of its children, or by another vertex that shares a child with s in $G_{S \cup T}$. But s cannot have any child in $G_{S \cup T}$, since this child cannot have a smaller depth than s, and it cannot be in S either since all elements in S have the same depth. □

This lemma allows us to work recursively as follows. Assume that we can compute marginal distributions over sets S with maximal depth no larger than d. Take a set S of maximal depth $d+1$, and let S_0 be the set of elements of depth $d+1$ in S. Then, letting $T = \mathrm{depth}^-(S) = \mathrm{depth}^-(S_0)$, and $S_1 = S \setminus S_0$,

$$\begin{aligned} P_S^X(x_S) &= \sum_{y_{T \setminus S_1}} P^X(x_{S_0} | y_{T \setminus S_1}, x_{S_1}) P^X_{T \cup S_1}(y_{T \setminus S_1} \wedge x_{S_1}) \\ &= \sum_{y_{S^- \setminus S_1}} \prod_{s \in S_0} p_s(x_s | (y \wedge x)_{s^-}, x_{S_1}) P^X_{S_0^- \cup S_1}(y_{S_0^- \setminus S_1} \wedge x_{S_1}) \end{aligned} \tag{78}$$

Since $S^- \cup S_1$ has maximal depth strictly smaller than the maximal depth of S, this indeed provides a recursive formula for the computation of marginal over subsets of V with increasing maximal depths. However, because one needs to add parents to the considered set when reducing the depth, one may end up having to compute marginals over very large sets, which becomes intractable without further assumptions.

A way to reduce the complexity is to assume that the graph G is singly connected, as defined below.

DEFINITION 39. *A DAG G is singly connected if there exists at most one path in G that connects any two vertices.*

Such a property is true for a tree, but also holds for some networks with multiple parents. We have the following nice property in this case.

PROPOSITION 45. *Let G be a singly connected DAG and X a Bayesian network on G. If s is a vertex in G, the variables $(X_t, t \in s^-)$ are mutually independent.*

PROOF. We have, using Proposition 39,

$$\pi_{s^-}(x_{s^-}) = \sum_{y_{\mathcal{A}_{s^-}}, y_{s^-} = x_{s^-}} \prod_{u \in \mathcal{A}_{s^-}} p_u(y_u | y_{u^-}).$$

Because the graph is singly connected, two parents of s cannot have a common ancestor (since there would then be two paths from this ancestor to S). So $\mathcal{A}_{s^-}$ is the disjoint union

of the $\mathcal{A}_t$'s for $t \in s^-$ and we can write

$$\begin{aligned}\pi_{s^-}(x_{s^-}) &= \sum_{y_{\mathcal{A}_{s^-}}, y_{s^-}=x_{s^-}} \prod_{t\in s^-} \prod_{u\in\mathcal{A}_t} p_u(y_u|y_{u^-}) \\ &= \prod_{t\in s^-} \sum_{y_{\mathcal{A}_t}, y_t=x_t} \prod_{u\in\mathcal{A}_t} p_u(y_u|y_{u^-}) \\ &= \prod_{t\in s^-} \pi_t(x_t)\end{aligned}$$

This proves the lemma. □

Equation (78) can be simplified under the assumption of a singly connected graph, at least for the computation of single vertex marginals; we have, if $s \in V$ and G is singly connected

$$P_s^X(x_s) = \sum_{y_{s^-}} p_s(x_s|y_{s^-}) \prod_{t\in s^-} P_t^X(y_t). \tag{79}$$

This is now recursive in single vertex marginal probabilities. It moreover coincides with the recursive equation that defines the messages $m_{C_s s}$ in (77), which shows that the sum-prod algorithm provides the correct answer in this case.

5.2. Conditional Probabilities and Intervention. One of the main interests of graphical models is to provide an ability to infer the behavior of hidden variables of interest given other, observed, variables. When dealing with oriented graphs the way this should be analyzed is, however, ambiguous.

Let's consider an example, provided by the graph in Figure 1. The Bayesian network

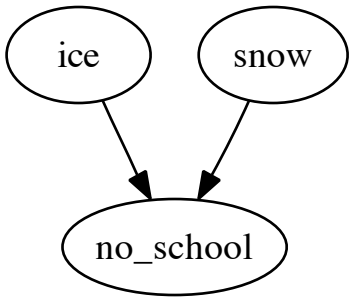

FIGURE 1.

interpretation of this graph is that both events 'ice' and 'snow' happen first, and that they are independent. Then, given their observation, the 'no_school' even may occur, probably more likely if there is ice or snow, and even more when ice and snow have both occurred.

Now consider the following passive observation: you wake up, you haven't checked the weather yet, and someone tells you: there is no school today. Then you may infer that

there is more chances than usual for ice and/or snow. Given this information, snow and ice now become correlated, even if they were initially independent. So, even if the 'no-school' even is considered as a probabilistic consequence of its parents, observing it influences our knowledge on them.

Now, here is an intervention, or manipulation: the school superintendent has declared that he has given enough snow days for the year and declared that there would be school today whatever happens. So you know that the 'no-school' event will not happen. Does it change the risk of ice or snow? Obviously not: intervention on a node does not affect the distribution of the parents.

Manipulation and passive observation are two very different ways of affecting unobserved variables in Bayesian networks. Both of them may be relevant in applications. Of the two, the simplest to analyze is intervention, since it merely consists in clamping one of the variables while letting the rest of the network dynamic unchanged. This leads to the following formal definition of manipulation.

DEFINITION 40. *Let $G = (V, E)$ be a directed acyclic graph and X a Bayesian network on G. Let S be a subset of G and $x_S \in F_S$ a given configuration on S. Then the manipulated distribution of X with fixed values x_S on S is the Bayesian network on the restricted graph G_S, with the same conditional probabilities, using the value x_s every time a vertex $s \in S$ is a parent of $t \in V \setminus S$ in G.*

So, if the distribution of X is given by (74), then its distribution after manipulation on S is

$$\tilde{\pi}(y_{V\setminus S}) = \prod_{t\in V\setminus S} p_s(y_t|y_{t^-})$$

where t^- is the set of parents of t in G, and $y_s = x_s$ whenever $s \in t^- \cap S$.

The distribution of a Bayesian network X after passive observation $X_S = x_S$ is not so easily described. It is obviously the conditional distribution $P(X_{V\setminus S} = y_{V\setminus S}|X_S = x_S)$ and therefore requires using the conditional dependency structure, involving the moral graph and/or d-separation.

Let us discuss this first in the simpler case of trees, for which the moral graph is the undirected acyclic graph underlying the tree, and d-separation is simple separation on this acyclic graph. We can then use Proposition 11 to understand the new structure after conditioning: it is a $G^\flat_{V\setminus S}$-Markov random field, and, for $t \in V \setminus S$, the conditional distribution of $X_t = y_t$ given its neighbors is the same as before, using the value x_s when $s \in S$. But note that when doing this (passing to $G^\flat$), we broke the causality relation between the variables. We can however always go back to a tree (or forest, since connectedness may have been broken) with the same edge orientation as they initially were, but this requires reconstituting the edge joint probabilities from the new acyclic graph, and therefore using (acyclic) belief propagation.

With general Bayesian networks, we know that the moral graph can be loopy and therefore a source of serious difficulties. The following proposition states that the damage is circumscribed to the ancestors of S.

PROPOSITION 46. *Let $G = (V, E)$ be a directed acyclic graph, X a Bayesian network on G, $S \subset V$ and $x_{\mathcal{A}_S} \in F_{\mathcal{A}_S}$. Then the conditional distribution of $X_{\mathcal{A}_S^c}$ given by $X_{\mathcal{A}_S} = x_{\mathcal{A}_S}$ coincides with the manipulated distribution in Definition 40.*

PROOF. The conditional distribution is proportional to

$$\prod_{s \in V} p(y_s | y_{s^-})$$

with $y_t = x_t$ if $t \in \mathcal{A}_S$. Since $s \in \mathcal{A}_S$ implies $s^- \subset \mathcal{A}_S$, all terms with $s \in \mathcal{A}_S$ are constant in the sum and can be factored out after normalization. So the conditional distribution is proportional to

$$\prod_{s \in \mathcal{A}_S^c} p(y_s | y_{s^-})$$

with $y_t = x_t$ if $t \in \mathcal{A}_S$. But we know that such products sum to 1, so that the conditional distribution is equal to this expression and therefore provides a Bayesian network on $G_{\mathcal{A}_S^c}$. □

CHAPTER 5

Parameter Estimation

1. Learning Bayesian Networks

1.1. Learning a Single Probability. Since Bayesian networks are specified by probabilities and conditional probabilities of configurations of variables, we start with a discussion of the apparently simple problem of estimating discrete probability distributions.

The obvious way to estimate the probability of an event A based on a series of N independent experiments is by using relative frequencies

$$f_A = \frac{\#\{A \text{ occurs}\}}{N}.$$

This estimation is unbiased ($E(f_A) = P(A)$) and its variance is $P(A)(1 - P(A))/N$. This implies that the relative error $\delta_A = f_A/P(A) - 1$ has zero mean and variance

$$\sigma^2 = \frac{1 - P(A)}{NP(A)}.$$

This number can become very large when $P(A) \sim 0$. In particular, when $P(A)$ is small compared to $1/N$, the relative frequency will often be $f_A = 0$, leading to the false conclusion that A is not just rare, but impossible. If there are reasons to expect beforehand that A is indeed possible, it is important to inject this prior belief in the procedure. This can be done with Bayesian estimation methods.

The main assumption for these methods is to consider the unknown probability, $p = P(A)$, as a random variable, yielding the generative process in which a random probability is first obtained, and then N instances of A or not-A are generated using this probability.

The prior belief can be represented as a probability distribution on p. Denote its density by $q(p)$, over the unit interval; this represents the *prior distribution* of the parameter p. Based on N independent observations of occurrences of A, each following a Bernoulli distribution $b(p)$, we can build a joint likelihood given by

$$\binom{N}{k} p^k (1-p)^{N-k} q(p),$$

where k is the number of times the event A has been observed.

The conditional density of p given the observation (k occurrences of A) is called the *posterior distribution*. Here, it is given by

$$q(p|k) = \frac{q(p)}{C_k} p^k (1-p)^{N-k}$$

where C_k is a normalizing constant. If there was no specific prior knowledge on p (so that $q(p) = 1$), the resulting distribution is a beta distribution with parameters $k+1$ and $N-k+1$, the beta distribution being defined as follows.

DEFINITION 41. *The beta distribution with parameters a and b (abbreviated $\beta(a, b)$) has density*

$$\rho(t) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} t^{a-1}(1-t)^{b-1} \text{ if } t \in [0,1]$$

and $\rho(t) = 0$ otherwise, with

$$\Gamma(x) = \int_0^\infty t^{x-1} e^{-t} dt.$$

From the definition of a beta distribution, it is clear also that, if we choose the prior in the Bayesian analysis to be $\beta(a+1, \nu-a+1)$ then the posterior is $\beta(k+a+1, N+\nu-(k+a)+1)$. The posterior therefore belongs to the same family of distributions as the prior, and one says that the beta distribution is a *conjugate prior* for the binomial distribution. The mode of the posterior distribution (which is the maximum a posteriori (MAP) estimator) is given by

$$\hat{p} = \frac{k+a}{N+\nu}.$$

This estimator now provides a positive value even if $k = 0$. By selecting a and ν, we therefore can incorporate our prior belief on p being small, but positive.

1.2. Learning a Finite Probability Distribution. Now assume that F is a finite space and that we want to estimate a probability distribution $(p(x), x \in F)$ using a Bayesian approach as above. We cannot use the previous approach to estimate each $p(x)$ separately, since the probabilities are linked by the fact that they sum to 1. We can however come up with a good (conjugate) prior, identified, as done above, by computing the posterior associated to a uniform prior distribution.

Letting N_x be the number of times x is observed among N independent samples of a random variable X with distribution p, the joint distribution of $(N_x, x \in F)$ is multinomial, given by

$$P(N_x, x \in F|p) = \frac{N!}{\prod_{x\in F} N_x!} \prod_{x\in F} p(x)^{N_x}.$$

The posterior distribution of p given the observations with a uniform prior is proportional to $\prod_{x\in F} p(x)^{N_x}$. It belongs to the family of Dirichlet distributions, described in the following definition.

DEFINITION 42. *Let F be a finite set and $\mathcal{S}_F$ be the simplex defined by*

$$\mathcal{S}_F = \left\{ (p(x), x \in F) : p(x) \geq 0, x \in F \text{ and } \sum_{x\in F} p(x) = 1 \right\}.$$

The Dirichlet distribution with parameters $a = (a(x), x \in F)$ (abbreviated $\mathrm{Dir}(a)$) has density

$$\rho(p) = \frac{\Gamma(\nu)}{\prod_{x\in F} \Gamma(a(x))} \prod_{x\in F} p(x)^{a(x)-1}, \text{ if } x \in \mathcal{S}_F$$

and 0 otherwise, with $\nu = \sum_{x\in F} a(x)$.

Note that, if F has cardinality 2, the Dirichlet distribution coincides with the beta distribution. Like with the beta for the binomial, and almost by construction, the Dirichlet distribution is a conjugate prior for the multinomial. More precisely, if the prior distribution for p is $\mathrm{Dir}(1 + a(x), x \in F)$, then the posterior after N observations of X is $\mathrm{Dir}(1 + N_x + a(x), x \in F)$, and the MAP estimator is given by

$$\hat{p}(x) = \frac{N_x + a(x)}{N + \nu}$$

with $\nu = \sum_{x \in F} a(x)$.

1.3. Conjugate Prior for Bayesian Networks. We now consider the issue of estimating the conditional probabilities in the representation

$$P(x) = \prod_{s \in V} p_s(x_s | x_{s^-}).$$

Assume that N independent observations of X have been made. Define the counts $N_s(x_s, x_{s^-})$ to be the number of times the observation $x_{\{s\} \cup s^-}$ has been made. Then, it is straightforward to see that, assuming a uniform prior for the p_s, their posterior distribution is proportional to

$$\prod_{s \in V} \prod_{x_{s^-} \in F_{s^-}} \prod_{x_s \in F_s} p_s(x_s | x_{s^-})^{N_s(x_s, x_{s^-})}.$$

This implies that, for the posterior distribution, the conditional probabilities $p_s(.|x_{s^-})$ are independent and follow a Dirichlet distribution with parameters $(1 + N_s(x_s, x_{s^-}), x_s \in F_s)$.

So, independent Dirichlet distributions indexed by configurations of parents of nodes provide a conjugate prior for the general Bayesian network model. This prior is specified by a family of positive numbers

$$(a_s(x_s, x_{s^-}), s \in V, x_s \in F_s, x_{s^-} \in F_{s^-}), \tag{80}$$

yielding a prior probability proportional to

$$\prod_{s \in V} \prod_{x_{s^-} \in F_{s^-}} \prod_{x_s \in F_s} p_s(x_s | x_{s^-})^{a_s(x_s . x_{s^-}) - 1}.$$

and a MAP estimator

$$\hat{p}_s(x_s | x_{s^-}) = \frac{N_s(x_s, x_{s^-}) + a_s(x_s, x_{s^-})}{N_s(x_{s^-}) + \nu_s(x_{s^-})} \tag{81}$$

where $N_s(x_{s^-}) = \sum_{x_s \in F_s} N_s(x_s, x_{s^-})$ and $\nu_s(x_{s^-}) = \sum_{x_s \in F_s} a_s(x_s, x_{s^-})$.

One can restrict the huge class of coefficients described by (80) to a smaller class by imposing the following consistency condition.

Definition 43. *One says that the family of coefficients*

$$a = (a_s(x_s, x_{s^-}), s \in V, x_s \in F_s, x_{s^-} \in F_{s^-}),$$

is consistent if there exists a positive scalar ν and a probability distribution P' on F_V such that

$$a_s(x_s, x_{s^-}) = \nu P'_{\{s\} \cup s^-}(x_s, x_{s^-}).$$

The class of products of Dirichlet distributions with consistent families of coefficients still provides a conjugate prior for Bayesian networks (the proof being left to the reader). Within this class, the simplest choice (and most natural in the absence of additional information) is to assume that P' is uniform, so that

$$a_s(x_s, x_{s^-}) = \nu' P'(x_s, x_{s^-}) = \frac{\nu'}{|F_{\{s\}\cup s^-}|}. \tag{82}$$

With this choice, ν' is the only parameter that must be specified. It is often called the equivalent sample size for the prior distribution.

We can see from (81) that using a prior distribution is quite important for Bayesian networks, since, when the number of parents increases, some configurations on F_{s^-} may not be observed, resulting in an undetermined value for the ratio $N_s(x_s, x_{s^-})/N_s(x_{s^-})$, although, for the estimated model, the probability of observing x_{s^-} may not be zero.

1.4. Structure Scoring. Given a prior defined as a family of Dirichlet distributions associated to $a = (a_s(x_s, x_{s^-}), s \in V, x_s \in F_s, x_{s^-} \in F_{s^-})$, the joint distribution of the observations and parameters is given by

$$P(\mathbf{x}, \theta) = \prod_{s, x_{s^-}} \mathcal{D}(a_s(\cdot, x_{s^-})) \prod_{s, x_s, x_{s^-}} p(x_s|x_{s^-})^{N_s(x_s;x_{s^-})+a_s(x_s;x_{s^-})-1}$$

with

$$\mathcal{D}(a(\lambda), \lambda \in F) = \frac{\Gamma(\nu)}{\prod_\lambda \Gamma(a(\lambda))}$$

and $\nu = \sum_\lambda a(\lambda)$. Here, th represents the parameters of the model, i.e., the conditional distributions that specify the Bayesian network. The marginal of this likelihood over all possible parameters, i.e.,

$$P(\mathbf{x}) = \int P(\mathbf{x}, \theta) d\theta$$

provides the expected likelihood of the sample relative to the distribution of the parameters, and only depends on the structure of the network. In our case, integrating with respect to th yields

$$\ln P(\mathbf{x}) = \sum_{s, x_{s^-}} \ln \frac{\mathcal{D}(a_s(\cdot, x_{s^-}))}{\mathcal{D}(a_s(\cdot, x_{s^-}) + N_s(\cdot, x_{s^-}))}.$$

Letting

$$\gamma(s, s^-) = \sum_{x_{s^-}} \ln \frac{\mathcal{D}(a_s(\cdot, x_{s^-}))}{\mathcal{D}(a_s(\cdot, x_{s^-}) + N_s(\cdot, x_{s^-}))},$$

the decomposition

$$\ln P(\mathbf{x}) = \sum_{s \in V} \gamma(s, s^-)$$

expresses this likelihood as a sum of scores (associated to each node and its parents), which depends on the observed sample. The scores that are computed above are often called

Bayesian scores because they derive from a Bayesian construction. One can also consider more simple scores, like penalized likelihood:

$$\gamma(s, s^-) = -\sum_{x_{s^-}} \hat{H}(X_s|X_{s^-})|F_{s^-}| - \rho|s^-|,$$

where $\hat{H}$ is the conditional entropy for the empirical distribution based on observed samples.

Structure learning algorithms are designed to optimize such scores.

1.5. Reducing the Parametric Dimension. In the previous section, we estimated all conditional probabilities intervening in the network. This is obviously a lot of parameters and, even with a regularizing prior, the estimated values will be inaccurate for small sample sizes. Simple procedures can be used to simplify the parametric complexity of the model.

When the sets F_s are not too large, which is common in practice, the parametric explosion is due to the multiplicity of parents, since the number of conditional probabilities $p_s(.|x_{s^-})$ grows exponentially with $|s^-|$.

One way to simplify this is to assume that the conditional probability at s only depends on x_{s^-} via some "global-effect" statistic $g_s(x_{s^-})$. The idea, of course, is that the number of values taken by g_s should remain small, even if the number of parents is large.

Examples of some functions g_s can be $\max(x_t, t \in s^-)$, or the min, of some simple (quantized) function of the sum. With binary variables ($F_s = \{0, 1\}$), logical operators are also available ("and", "or", "xor"), as well as combinations of them. The choice made for the functions g_s is part of building the model, and would rely on the specific context and prior information on the process, which is always important to account for, in any statistical problem.

Once the g_s's are fixed, learning the network distribution, which is now given by

$$\pi(x) = \prod_{s \in V} p_s(x_s|g_s(x_{s^-}))$$

can be done exactly as before, the parameters being all $p_s(\lambda|w), \lambda \in F_s, w \in W_s$, where W_s is the range of g_s, and Dirichlet priors can be associated to each $p_s(., w)$ for $s \in V$ and $w \in W_s$. The counts provided in (82) now can be chosen as

$$a_s(x_s, w) = \frac{\nu'}{|F|\,|g_s^{-1}(w)|}. \tag{83}$$

2. Learning Loopy Markov Random Fields

Like everything else, parameter estimation for loopy networks is much harder than with trees or Bayesian networks. There is usually no closed form expression for the estimators, and their computation relies on more or less tractable numerical procedures.

2.1. Maximum Likelihood with Exponential Models. In this section, we consider a parameterized model for a Gibbs distribution

$$\pi_\theta(x) = \frac{1}{Z_\theta} \exp(-\theta^T U(x)) \tag{84}$$

where θ is a d-dimensional parameter and U is a function from F_V to $\mathbb{R}^d$. For example, if π is an Ising model with

$$\pi(x) = \frac{1}{Z} \exp\big(\alpha \sum_{s \in V} x_s + \beta \sum_{s \sim t} x_s x_t\big),$$

we would take $\theta = (\alpha, \beta)$ and $U(x) = -(\sum_s x_s, \sum_{s \sim t} x_s x_t)$. Most of the Markov random fields models that are used in practice can be put in this form.

The constant Z_θ in (84) is

$$Z_\theta = \sum_{x \in F_V} \exp(-\theta^T U(x))$$

and is usually not computable.

Now, assume that an N-sample, $x^{(1)}, \ldots, x^{(N)}$ is observed for this distribution. The maximum likelihood estimator maximizes

$$\ell(\theta) = \frac{1}{N} \sum_{k=1}^{N} \ln \pi_\theta(x^{(k)}) = -\theta^T \bar{U}_N - \ln Z_\theta$$

with $\bar{U}_N = (U(x^{(1)}) + \cdots + U(x^{(N)}))/N$.

We have the following proposition, which is a well-known property of exponential families of probabilities.

PROPOSITION 47. *The log-likelihood, ℓ, is a concave function of θ, with*

$$\nabla \ell(\theta) = E_\theta(U) - \bar{U}_N \tag{85}$$

and

$$D^2(\ell)(\theta) = -\,Var_\theta(U) \tag{86}$$

where E_θ denotes the expectation with respect to π_θ and Var_θ the covariance matrix under the same distribution.

This proposition implies that a local maximum of $\theta \mapsto \ell(\theta)$ must also be global. Any such maximum must be a solution of

$$E_\theta(U) = V(x_0)$$

and conversely. There are some situations in which the maximum does not exist, or is not unique. Let's first discuss the second case.

If several solutions exist, the log-likelihood cannot be strictly concave: there must exist at least one θ for which $\mathrm{Var}_\theta(U)$ is not definite. This implies that there exists a nonzero vector u such that $\mathrm{var}_\theta(u^T U) = u^T \mathrm{Var}_\theta(U) u = 0$. This is only possible when $u^T U(x) = \mathrm{cst}$ for all $x \in F_V$. Conversely, if this is true, $\mathrm{Var}_\theta(U)$ is degenerate *for all* θ.

So, the non-uniqueness of the solutions is only possible when a deterministic affine relation exists between the components of U, i.e., the model is over-dimensioned. Such situations are usually easily dealt with. In all other cases, there exists at most one maximum.

For a concave function like ℓ to have no maximum, there must exist what is called a direction of recession, which is a direction $u \in \mathbb{R}^d$ such that, for all θ, the function

$t \mapsto \ell(\theta + t\alpha)$ is increasing. In this case the maximum is attained "at infinity". Denoting $U_\alpha(x) = \langle \alpha \, , \, U(x) \rangle$, the derivative in t of $\ell(\theta + t\alpha)$ is

$$E_{\theta+t\alpha}(U_\alpha) - \bar{U}_\alpha.$$

This derivative is positive for all t if and only if

$$\bar{U}_\alpha = U_\alpha^* := \min\{U_\alpha(x), x \in F_V\} \tag{87}$$

and U_α is not constant. To prove this, assume that the derivative is positive. Then U_α is not constant (otherwise, the derivative would be zero). Let $F_\alpha^* \subset F_V$ be the set of configurations x for which $U_\alpha(x) = U_\alpha^*$. Then

$$\begin{aligned}
&E_{\theta+t\alpha}(U_\alpha) \\
&= \frac{\sum_{x \in F_V} U_\alpha(x) \exp(-\theta^T U(x) - t U_\alpha(x))}{\sum_{x \in F_V} \exp(-\theta^T U(x) - t U_\alpha(x))} \\
&= \frac{\sum_{x \in F_V} U_\alpha(x) \exp(-\theta^T U(x) - t(U_\alpha(x) - U_\alpha^*))}{\sum_{x \in F_V} \exp(-\theta^T U(x) - t(U_\alpha(x) - U_\alpha^*))} \\
&= \frac{U_\alpha^* \sum_{x \in F_\alpha^*} \exp(-\theta^T U(x)) + \sum_{x \notin F_\alpha^*} U_\alpha(x) \exp(-\theta^T U(x) - t(U_\alpha(x) - U_\alpha^*))}{\sum_{x \in F_\alpha^*} \exp(-\theta^T U(x)) + \sum_{x \notin F_\alpha^*} \exp(-\theta^T U(x) - t(U_\alpha(x) - U_\alpha^*))}.
\end{aligned}$$

When t tends to $+\infty$, the sums over $x \notin F_\alpha^*$ tend to 0, which implies that $E_{\theta+t\alpha}(U_\alpha)$ tends to U_α^*. So, if $E_{\theta+t\alpha}(U_\alpha) - \bar{U}_\alpha > 0$ for all t, then $\bar{U}_\alpha = U_\alpha^*$ and U_α is not constant. The converse statement is obvious.

As a conclusion, the function ℓ has a finite maximum if and only if there is no direction $u \in \mathbb{R}^d$ such that $\left\langle \alpha \, , \, U(x) - \bar{U}_N \right\rangle \leq 0$ for all $x \in F_V$. Equivalently, $\bar{U}_N$ must belong to the interior of the convex hull of the finite set

$$\{U(x), x \in F_V\} \subset \mathbb{R}^d.$$

In such a case, that we hereafter assume, computing the maximum likelihood estimator boils down to solving the equation

$$E_\theta(U) = \bar{U}_N.$$

Because the maximization problem is concave, we know that numerical algorithms like gradient descent,

$$\theta(t+1) = \theta(t) + \varepsilon(E_{\theta(t)}(U) - \bar{U}_N), \tag{88}$$

or Newton-Raphson,

$$\theta(t+1) = \theta(t) + \varepsilon \mathrm{Var}_{\theta(t)}(U)^{-1}(E_{\theta(t)}(U) - \bar{U}_N), \tag{89}$$

which is more efficient, converge to the optimal parameter. Unfortunately, the computation of the expectations and covariance matrices can only be made explicitly for acyclic models, for which parameter estimation is not a problem anyway. For general loopy graphical models, the expectation can be estimated iteratively using Monte-Carlo methods. It turns out that this estimation can be synchronized with gradient descent to obtain a consistent algorithm, which is described in the next section.

2.2. Maximum Likelihood with Stochastic Gradient. As remarked above, for fixed θ, we have designed Markov chain Monte Carlo algorithms that asymptotically sample form π_θ. Select one of these algorithms, and let P_θ be the corresponding stochastic matrix that provides the associated transition probabilities for a given θ. Then, define the iterative algorithm, initialized with arbitrary $\theta(0)$ and $x(0) \in F_V$, that loops over the following two steps.

(SG1) Sample from the distribution $P_{\theta(t)}(x(t), \cdot)$ to obtain a new configuration $x(t+1)$.
(SG2) Update the parameter using

$$\theta(t+1) = \theta(t) + \gamma(t+1)(U(x(t+1)) - \bar{U}_N). \tag{90}$$

Then, the following holds.

THEOREM 11. *If P_θ corresponds to the Gibbs sampler or Metropolis algorithm, and $\gamma(t+1) = \varepsilon/(t+1)$ for small enough ε, the algorithm that iterates (SG1) and (SG2) converges to the maximum likelihood estimator.*

This is a particular instance of a stochastic gradient algorithm, in which the expectation in (88) is crudely estimated at each step by the evaluation of U on the random sample $x(t)$. One of the reasons for the convergence (and almost a necessary condition for it) is that the average of (90) over $x(t+1)$ for the invariant distribution of $P_{\theta(t+1)}$ is precisely (88). This averaging effect then takes place over time to provide the correct limit.

The speed of convergence of such algorithms depends both on the speed of convergence of the Monte-Carlo sampling and of the original gradient descent. The latter can be improved somewhat with variants that try to reproduce (89). For high dimensional systems, however, convergence may be hard to achieve, and may require some tuning of the iteration gains $\gamma(t)$.

2.3. Relation with Maximum Entropy. The maximum likelihood estimator is closely related to what is called the maximum entropy extension of a set of constraints. Let the function U from F_V to $\mathbb{R}^d$ be given. An element $u \in \mathbb{R}^d$ is said to be a consistent assignment for U if there exists a probability distribution π on F_V such that $E_\pi(U) = u$. An example of consistent assignment is any empirical average $\bar{U}$ based on a sample $(x^{(1)}, \ldots, x^{(N)})$, since $\bar{U} = E_\pi(U)$ for

$$\pi = \frac{1}{N} \sum_{k=1}^{N} \delta_{x^{(k)}}.$$

Given U and a consistent assignment, u, the associated maximum entropy extension is defined as probability distribution π maximizing the entropy, $H(\pi)$, subject to the constraint $E_\pi(U) = u$. This is a convex optimization problem, with constraints

$$\begin{cases} \sum_{x \in F_V} \pi(x) = 1 \\ \sum_{x \in F_V} U_j(x)\pi(x) = u_j, j = 1, \ldots, d \\ \pi(x) \geq 0, x \in F_V \end{cases} \tag{91}$$

Because the entropy is strictly convex, there is a unique solution to this problem. We first discuss non-positive solutions, i.e., solutions for which $\pi(x) = 0$ for some x. An important fact is that, if, for a given x, there exists π_1 such that $E_{\pi_1}(U) = u$ and $\pi_1(x) > 0$, then

the optimal π must also satisfy $\pi(x) > 0$. This is because, if $\pi(x) = 0$, then, letting $\pi_\varepsilon = (1-\varepsilon)\pi + \varepsilon\pi_1$, we have $E_{\pi_\varepsilon}(U) = u$ since this constraint is linear, $\pi_\varepsilon(x) > 0$ and

$$\begin{aligned} H(\pi_\varepsilon) - H(\pi) &= -\sum_{y,\pi(y)>0} (\pi_\varepsilon(y)\ln\pi_\varepsilon(y) - \pi(y)\ln\pi(y)) \\ &\quad -\sum_{y,\pi(y)=0} \varepsilon\pi_1(y)(\ln(\varepsilon) + \ln\pi_1(y)) \\ &= -\varepsilon\ln\varepsilon \sum_{y,\pi(y)=0} \pi_1(y) + O(\varepsilon) \end{aligned}$$

which is positive for small enough ε, contradicting the fact that π is a maximizer.

Introduce the set $\mathcal{N}_u$ containing all configurations $x \in F_V$ such that $\pi(x) = 0$ for all π such that $E_\pi(U) = u$. Then we know that the maximum entropy extension satisfies $\pi(x) > 0$ if $x \notin \mathcal{N}_u$. Introduce Lagrange multipliers $\theta_0, \theta_1, \ldots, \theta_d$ for the $d+1$ equality constraints in (91), and the Lagrangian

$$L = H(\pi) + \sum_{x \in F_V \setminus \mathcal{N}_u} (\theta_0 + \langle \theta \, , \, U(x) \rangle)\pi(x)$$

in which we have set $\theta = (\theta_1, \ldots, \theta_d)$, we find that the optimal π must satisfy

$$\begin{cases} \ln\pi(x) = -\theta_0 - 1 - \langle \theta \, , \, U(x) \rangle \\ \sum_x \pi(x) = 1 \\ E_\pi(U) = \bar{u} \end{cases}$$

In other terms, the maximum entropy extension is characterized by

$$\pi(x) = \frac{1}{Z_\theta} \exp(-\langle \theta \, , \, U(x) \rangle)\delta_{\mathcal{N}_u^c}(x)$$

and $E_\pi(U) = u$.

In particular, if $\mathcal{N}_u = \emptyset$, then the maximum entropy extension is positive. If, in addition, $u = \bar{U}$ for some observed sample, then it coincides with the maximum likelihood estimator for (84). Notice that, in this case, the condition $\mathcal{N}_u \neq \emptyset$ coincide with the condition that there exists α such that $\alpha^T U(x) \geq \alpha^T u$ for all x, with $\alpha^T U(x)$ not constant. Indeed, assume that the latter condition is true. Then, if $E_\pi(U) = u$, then $E_\pi(\alpha^T U) = \alpha^T u$, which is only possible if $\pi(x) = 0$ for all x such that $\alpha^T U(x) < \alpha^T u$. Such x's exist by assumption, and therefore $\mathcal{N}_u \neq \emptyset$. Conversely, assume $\mathcal{N}_u \neq \emptyset$. If condition (87) is not satisfied, then we have shown when discussing maximum likelihood that an optimal parameter for the exponential model would exist, leading to a positive distribution for which $E_\pi(U) = u$, which is a contradiction.

2.4. Iterative Scaling. Iterative scaling is a method that is well-adapted to learning distributions given by (84), when U can be interpreted as a random histogram, or a collection of them.

More precisely, assume that for all $x \in F_V$, one has

$$U(x) = (U_1(x), \ldots, U_q(x))$$

with

$$\sum_{j=1}^{q} U_j(x) = 1 \text{ and } U_j(x) \geq 0.$$

Let the parameter be given by $\theta = (\theta_1, \ldots, \theta_q)$. Assume that $x^{(1)}, \ldots, x^{(N)}$ have been observed, and let $u \in \mathbb{R}^d$ be a consistent assignment for U, with $u_j > 0$ for $j = 1, \ldots, d$ and such that $\mathcal{N}_u = \emptyset$. Iterative scaling computes the maximum entropy extension of $E_\pi(U) = u$, that we will denote π^*. It is supported by the following lemma.

LEMMA 14. *Let π be a probability on F_V with $\pi > 0$ and define*

$$\pi'(x) = \frac{\pi(x)}{\zeta} \prod_{j=1}^{d} \left(\frac{u_j}{E_\pi(U_j)} \right)^{U_j(x)}$$

where ζ is chosen so that π' is a probability. Then $\pi' > 0$ and

$$D(\pi^* \| \pi') - D(\pi^* \| \pi) \leq -D(u \| E_\pi(U)) \leq 0 \tag{92}$$

PROOF. Note that, since $\pi > 0$, $E_\pi(U_j)$ must also be positive for all j, since $E_\pi(U_j) = 0$ would otherwise imply $U_j = 0$ and $u_j = 0$ for u to be consistent. So, π' is well defined and obviously positive.

We have

$$\begin{aligned} D(\pi^* \| \pi') - D(\pi^* \| \pi) &= \ln \zeta - \sum_{x \in F_V} \pi^*(x) \sum_{j=1}^{d} U_j(x) \ln \frac{u_j}{E_\pi(U_j)} \\ &= \ln \zeta - \sum_{j=1}^{d} u_j \ln \frac{u_j}{E_\pi(U_j)} \\ &= \ln \zeta - D(u \| E_\pi(U)). \end{aligned}$$

(We have used the identity $E_{\pi^*}(U) = u$.) So it suffices to prove that $\zeta \leq 1$. We have

$$\begin{aligned} \zeta &= \sum_x \pi(x) \prod_{j=1}^{d} \left(\frac{u_j}{E_\pi(U_j)} \right)^{U_j(x)} \\ &\leq \sum_{j=1}^{d} \sum_x \pi(x) U_j(x) \frac{u_j}{E_\pi(U_j)} \\ &= \sum_{j=1}^{d} E_\pi(U_j) \frac{u_j}{E_\pi(U_j)} \\ &= 1, \end{aligned}$$

which proves the lemma (we have used the fact that, for x_i, w_i positive numbers with $\sum_i w_i = 1$, one has $\prod_i x_i^{w_i} \leq \sum_i w_i x_i$, which is a consequence of the concavity of the logarithm). □

Consider the iterative algorithm

$$\pi^{(n+1)}(x) = \frac{\pi^{(n)}(x)}{\zeta_n} \prod_{j=1}^{d} \left(\frac{u_j}{E_{\pi^{(n)}}(U_j)} \right)^{U_j(x)}$$

initialized with a uniform distribution. Equivalently, using the exponential formulation, define, for $j = 1, \ldots, d$,

$$\theta_j^{(n+1)} = \theta_j^{(n)} + \ln \frac{E_{\theta^{(n)}}(U_j)}{u_j} + D(u \| E_{\theta_n}(U)), \tag{93}$$

with π_θ given by (84), initialized with $\theta^{(0)} = 0$. Note that adding a term that is independent of j to θ does not change the value of π_θ, because the U_j's sum to 1. The model is in fact over parametrized, and the addition of the KL distance in (93) ensures that $\sum_{i=1}^{d} u_i \theta_i = 0$ at all steps.

This algorithm always reduces the Kullback-Leibler distance to the maximum entropy extension. This distance being always positive therefore converges to a limit, which, still according to Lemma 14, is only possible if $D(u \| E_{\pi_n}(U))$ also tends to 0, that is $E_{\pi_n}(U) \to u$. Since the space of probability distributions is compact, the Heine-Borel theorem implies that the sequence π_{θ_n} has at least one accumulation point, that we now identify. If π is such a point, one must have $E_\pi(U) = u$. Moreover, we have $\pi > 0$, since otherwise $D(\pi^* \| \pi) = +\infty$. To prove that $\pi = \pi^*$ (and therefore the limit of the sequence), it remains to show that it can be put in the form (84). For this, define the vector space $\mathcal{V}$ of functions $v : F_V \to \mathbb{R}$ which can be written in the form

$$v(x) = \alpha_0 + \sum_{j=1}^{g} \alpha_j U_j(x),$$

so that it suffices to prove that $\ln \pi$ belongs to $\mathcal{V}$. But this result is obvious, since $\ln \pi_{\theta_n}$ is in $\mathcal{V}$ for all n, and therefore so does its limit. This proves the following proposition.

PROPOSITION 48. *Assume that for all $x \in F_V$, one has $U(x) = (U_1(x), \ldots, U_d(x))$ with*

$$\sum_{j=1}^{d} U_j(x) = 1 \text{ and } U_j(x) \geq 0.$$

Let u be a consistent assignment for the expectation of U such that $\mathcal{N}_u = \emptyset$. Then, the algorithm described in equation (93) *converges to the maximum entropy extension of u.*

This is the iterative scaling algorithm. This method can be extended in a straightforward way to handle the maximum entropy extension for a family of functions $U^{(1)}, \ldots, U^{(K)}$, such that, for all x and for all k, $U^{(k)}(x)$ is a d_k-dimensional vector such that

$$\sum_{j=1}^{d_k} U_j^{(k)}(x) = 1.$$

The maximum entropy extension takes the form

$$\pi_\theta(x) = \frac{1}{Z_\theta} \exp\Big(- \sum_{k=1}^{K} \big\langle \theta^{(k)} \, , \, U^{(k)}(x) \big\rangle \Big),$$

where $\theta^{(k)}$ is d_k-dimensional, and iterative scaling can then be implemented by updating only one of these vectors at a time, using (93) with $U = U^{(k)}$.

The restriction to $U(x)$ providing a discrete probability distribution for all x is, in fact, no loss of generality. This is because adding a constant to U does not change the resulting exponential model in (84), and multiplying U by a constant can be also compensated by dividing θ by the same constant in the same model. So, if u_- is a lower bound for $\min_{j,x} U_j(x)$, one can replace U by $(U - u_-)$, and therefore assume that $U \geq 0$, and if u_+ is an upper bound for $\sum_j U_j(x)$, we can replace U by U/u_+ and therefore assume that $\sum_j U_j(x) \leq 1$. Define

$$U_{d+1}(x) = 1 - \sum_{j=1}^{d} U_j(x) \geq 0.$$

Then, the maximum entropy extension for $(U_1, \ldots, U_d)$ with assignment $(u_1, \ldots, u_d)$ is obviously also the extension for $(U_1, \ldots, U_{d+1})$, with assignment $(u_1, \ldots, u_{d+1})$, where

$$u_{d+1} = 1 - \sum_{j=1}^{d} u_j,$$

and the latter is in the form required in Proposition 48.

2.5. Pseudo Likelihood. Maximum likelihood estimation is a special case of *minimal contrast estimators*. These estimators are based on the definition of a measure of dissimilarity, say $C(\pi \| \tilde{\pi})$, between two probability distributions π and $\tilde{\pi}$. The usual assumptions on C are that $C(\pi \| \tilde{\pi}) \geq 0$, with equality if and only if $\pi = \tilde{\pi}$, and that C is — at least — continuous in π and $\tilde{\pi}$. Minimal contrast estimators approximate the problem of minimizing $\theta \mapsto C(\pi_{\text{true}} \| \pi_\theta)$ over a parameter θ, (which is not feasible, since π_{true}, the true distribution of the data, is unknown) by the minimization of $\theta \mapsto C(\hat{\pi} \| \pi_\theta)$ where $\hat{\pi}$ is the empirical distribution computed from observed data. Under mild conditions on C, these estimators are generally consistent when N tends to infinity, which means that the estimated parameter asymptotically (in the sample size N) provides the best (according to C) approximation of π_{true} by a π_θ.

The contrast that is associated to maximum likelihood is the Kullback-Leibler divergence, defined by

$$D(\pi \| \tilde{\pi}) = E_\pi(\ln \frac{\pi}{\tilde{\pi}}).$$

Indeed, given a sample $x_1, \ldots, x_N$, we have

$$\begin{aligned} D(\hat{\pi} \| \pi_\theta) &= E_{\hat{\pi}} \ln \hat{\pi} - E_{\hat{\pi}} \ln \pi_\theta \\ &= E_{\hat{\pi}} \ln \hat{\pi} - \sum_{k=1}^{N} \ln \pi_\theta(x_k). \end{aligned}$$

Since $E_{\hat{\pi}} \ln \hat{\pi}$ does not depend on θ, minimizing $D(\hat{\pi} \| \pi_\theta)$ is equivalent to maximizing $\sum_{k=1}^{N} \ln \pi_\theta(x_k)$ which is the log-likelihood.

Maximum pseudo-likelihood estimators form another class of minimal contrast estimators for graphical models. Given a distribution π on F_V, define the local specifications

$\pi_s(x_s|x_t, t \neq s)$ to be conditional distributions at one vertex given the others, and the contrast

$$C(\pi\|\tilde{\pi}) = \sum_{s\in V} E_\pi(\ln \frac{\pi_s}{\tilde{\pi}_s}).$$

Because we can write, using standard properties of conditional expectations,

$$C(\pi\|\tilde{\pi}) = \sum_{s\in V} E_\pi\left(E_{\pi_s}(\ln \frac{\pi_s}{\tilde{\pi}_s})\right) = \sum_{s\in V} E(D(\pi_s(.|X_t, t \neq s)\|\tilde{\pi}_s(.|X_t, t \neq s)),$$

we see that $C(\pi, \tilde{\pi})$ is always positive, and vanishes (under the assumption of positive π) only if all the local specifications for π and $\tilde{\pi}$ coincide, and this can be shown to imply that $\pi = \tilde{\pi}$. Indeed, for any $x, y \in F_V$, and choosing some order $V = \{s_1, \ldots, s_n\}$ on V, we can write

$$\frac{\pi(x)}{\pi(y)} = \prod_{k=1}^{n} \frac{\pi(x_{s_k}|x_{s_1}, \ldots, x_{s_{k-1}}, y_{s_{k+1}}, \ldots, y_{s_n})}{\pi(x_{s_k}|x_{s_1}, \ldots, x_{s_{k-1}}, y_{s_{k+1}}, \ldots, y_{s_n})}$$

and the ratios $\pi(x)/\pi(y)$, for $x \in F_V$, combined with the constraint that $\sum_x \pi(x) = 1$ uniquely define π.

So C is a valid contrast and

$$C(\hat{\pi}\|\pi_\theta) = \sum_{s\in V} E_{\hat{\pi}} \ln \hat{\pi}_s - \sum_{s\in V} \sum_{k=1}^{N} \ln \pi_{\theta,s}(x_s^{(k)}|x_t^{(k)}, t \neq s).$$

This yields the *maximum pseudo-likelihood estimator* (or pseudo maximum likelihood) defined as a maximizer of the function (called log-pseudo-likelihood)

$$\theta \mapsto \sum_{s\in V} \sum_{k=1}^{N} \ln \pi_{\theta,s}(x_s^{(k)}|x_t^{(k)}, t \neq s).$$

Although maximum likelihood is known to provide the most accurate approximations in many cases, maximum of pseudo likelihood has the important advantage to be, most of the time, computationally feasible. This is because, for a model like (84), local specifications are given by

$$\pi_{\theta,s}(x_s|x_t, t \neq s) = \frac{\exp(-\theta^T U(x))}{\sum_{y_s\in F_s} \exp(-U(y_s \wedge x_{V\setminus s}))}.$$

and therefore include no intractable normalizing constant. Maximum of pseudo-likelihood estimators can be computed using standard maximization algorithms, including Newton-Raphson. For exponential models like (84), the log-pseudo-likelihood is, like the log-likelihood, a concave function.

3. Incomplete Observations

3.1. The EM Algorithm. We now address the issue of identifying graphical models when some of the node variables are not observed, which brings the problem to a new level of difficulty. Missing variables are unfortunately a common issue with graphical models. They may correspond to real processes that cannot be measured, which is common, for example, with biological data. They may be more conceptual objects that are interpretable but are not parts of the data acquisition process, like phonemes in speech recognition, or edges and

labels in image processing and object recognition. They may also be variables that have been added to the model to increase its parametric dimension without increasing the complexity of the graph.

The most important and widely used algorithm to deal with hidden variables is the EM algorithm. We first describe it in a general context before applying it to graphical models.

Let $X = (\xi, \eta)$ be a pair of random variables taking values in a set $M = M_\xi \times M_\eta$. Let (π_θ) a family of densities with respect to a product measure $\mu = \mu_\xi \otimes \mu_\eta$ (if you are unfamiliar with measure theory, think of $d\mu = d\xi\, d\eta$, a product of Lebesgue's measures). Assume that a sample, $(\xi^{(1)}, \ldots, \xi^{(N)})$ of ξ (the visible part of X; η is the hidden part) is observed. Let ψ_θ be the marginal distribution of V when the distribution of X is π_θ. We want to compute the maximum likelihood estimator, θ, which maximizes

$$\sum_{k=1}^{N} \ln \psi_\theta(\xi^{(k)}) = \sum_{k=1}^{N} \ln \int \pi_\theta(\xi^{(k)}, \eta) d\mu_\eta(\eta).$$

It is generally the case that the computation of the MLE for complete observations, ie. the maximization of

$$\sum_{k=1}^{N} \ln \pi_\theta(\xi^{(k)}, \eta^{(k)})$$

is easy (or well behaved, say, concave). For example, π_θ can be an exponential family of distributions. However, maximizing the log-likelihood of the incomplete data is much harder.

The key formula that justifies the EM algorithm is the following: we have

$$\ln \psi_\theta(\xi) = \max_\pi E_\pi \ln \left(\frac{\pi_\theta(\xi, \eta)}{\pi(\eta)} \right), \tag{94}$$

the maximum being over all densities π relative to μ_η. Let's prove this. Introduce the conditional density $\pi_\theta(\eta|\xi) = \pi_\theta(\xi, \eta)/\psi_\theta(\eta)$. We have

$$\begin{aligned} E_\pi \left(\ln \frac{\pi_\theta(\xi, \eta)}{\pi(\eta)} \right) &= \int \ln \frac{\pi_\theta(\xi, \eta)}{\pi(\eta)} \pi(\eta) d\mu_\eta(\eta) \\ &= \int \ln \frac{\pi_\theta(\eta|\xi)\psi_\theta(\xi)}{\pi(\eta)} \pi(\eta) d\mu_\eta(\eta) \\ &= \int \ln \frac{\pi_\theta(\eta|\xi)}{\pi(\eta)} \pi(\eta) d\mu_\eta(\eta) + \ln \psi_\theta(\xi) \\ &= -D(\pi \| \pi_\theta(.|\xi)) + \ln \psi_\theta(\xi) \end{aligned}$$

where D is the Kullback-Leibler divergence. Since we know that $D(f\|g)$ is always positive and vanishes only for $f = g$, the result is proved, with the additional fact that the maximum is attained for $\pi = \pi_\theta(.|\xi)$.

As a consequence, the log-likelihood can be written

$$\sum_{k=1}^{N} \ln \psi_\theta(\xi^{(k)}) = \sum_{k=1}^{N} \max_{\pi^{(k)}} E_{\pi^{(k)}} \left(\ln \frac{\pi_\theta(\xi^{(k)}, \eta)}{\pi^{(k)}(\eta)} \right) \tag{95}$$

Defining

$$\Delta_\theta(\pi', \xi) = E_{\pi'}\left(\ln \frac{\pi_\theta(\xi, \eta)}{\pi'(\eta)}\right), \tag{96}$$

maximum likelihood is equivalent to computing

$$\max_{\theta, \pi^{(1)}, \ldots, \pi^{(N)}} \sum_{k=1}^{N} \Delta_\theta(\pi^{(k)}, \xi^{(k)}).$$

This can be done by looping over two steps which are: (i) solve the maximization problem above with respect to θ with fixed $\pi^{(1)}, \ldots, \pi^{(N)}$, and (ii) maximize over the $\pi^{(k)}$'s with fixed θ. The solution of the last problem is already known, since we must have $\pi^{(k)} = \pi_\theta(.|\xi^{(k)})$. Therefore, the EM algorithm is, letting θ_n be the current parameter after loop n:

$$\text{Compute } \theta_{n+1} \text{ by maximizing } \theta \mapsto \textstyle\sum_{k=1}^{N} E_{\theta_n}\left(\ln \pi_\theta(\xi^{(k)}, \eta)|\xi = \xi^{(k)}\right).$$

(We have used the fact that $\ln \pi^{(k)}(\eta)$ does not depend on θ.)

The computation of the conditional expectations (which usually goes with a simplification of the expression in a closed form function of θ) is called the E (expectation) step of the algorithm. An important example is when the distribution of the complete data forms an exponential family, i.e.,

$$\pi_\theta(\xi, \eta) = \frac{1}{Z(\theta)} \exp\big(-\langle \theta\,,\, U(\xi, \eta)\rangle\big). \tag{97}$$

(One generally allows the dot product in the exponential to depend on a possibly nonlinear function of θ, say $h(\theta)$. We here take $h(\theta) = \theta$ to simplify (or, equivalently, implicitly replace θ by $h(\theta)$ without changing notation).)

From (97), one gets

$$\ln \pi_\theta(\xi, \eta) = -\ln Z(\theta) - \langle \theta\,,\, U(\xi, \eta)\rangle$$

and the transition from θ_n to θ_{n+1} is done by maximizing

$$-\ln Z(\theta) - \big\langle \theta\,,\, \bar{U}_n\big\rangle \tag{98}$$

where

$$\bar{U}_n = \frac{1}{N} \sum_{k=1}^{N} E_{\theta_n}(U(\xi^{(k)}, \eta)|\xi = \xi^{(k)}). \tag{99}$$

So, the M-step of the EM, which maximizes (98), coincides in fact with the complete-data maximum-likelihood problem for which the empirical average of U is replaced by the average of its conditional expectations given the observations, as given in (99), which constitues the E-step.

By construction, the EM increases the likelihood of the observed data at each step. If this likelihood decreases at infinity (in the sense that, for any compact set K in parameter space, there exists a compact set $\tilde{K} \supset K$ such that the likelihood at any point in K is larger than the maximum likelihood over $\tilde{K}^c$), then the sequence generated by the EM remains bounded and converges to a limit which is a stationary point of the likelihood. This limit,

being approached from below, cannot be a local minimum, so it is either a local maximum, or a saddle point (the latter option cannot be excluded in general).

Since the log-likelihood of the observations is not convex in general, the number of possible limits is generally larger than one, and the one which is obtained depends on the starting point, θ_0. Because of this, it is most of the time very useful to initialize the algorithm with a good guess of the true parameters. Saddle points can be avoided to some extent by trying a few reruns of the algorithm starting with small perturbations of the limit parameter (say, adding white noise with a small variance).

Another possibility is to run the EM algorithm multiple times with different initial conditions, and take the one with the highest likelihood. Note that this requires to be able, at least, to evaluate the marginal distribution of the observed data, i.e.,

$$\ln \psi_\theta(\xi) = \ln \int \pi_\theta(\xi, \eta) d\mu_\eta(\eta),$$

which, unfortunately, is not always tractable.

3.2. Working with a Bayesian Prior. The procedure is only slightly modified in the presence of a prior distribution of the parameter, say $q(\theta)$. The maximization of

$$\ln q(\theta) + \sum_{k=1}^{N} \ln \psi_\theta(\xi^{(k)})$$

can be solved by iterating $\theta_n \to \theta_{n+1}$ with

$$\theta_{n+1} = \text{argmax}_\theta \Big(\ln q(\theta) + \sum_{k=1}^{N} E_{\theta_n} \left(\ln \pi_\theta(\xi^{(k)}, \eta) | \xi = \xi^{(k)} \right) \Big).$$

3.3. Stochastic Approximation EM. The EM algorithm is not always feasible, and practical difficulties often arises because conditional expectations can be difficult to compute, which makes the E-step impossible to solve explicitly. It is however often possible to sample from the conditional distributions, either by direct sampling, or using Markov chains. Let us assume the latter, since it is more general, and therefore introduce transition probabilities $P_{\theta,\xi}$ on M_η, which provide ergodic Markov chains with invariant distributions given by $\pi_\theta(\cdot|\xi)$. (For this discussion, the reader can assume that η is a discrete random variables, and M_η a discrete set, although the framework extends to a large class of Markov chains on continuous spaces.)

The E-step of the EM can then be replaced by a stochastic approximation step, leading to the stochastic approximation EM algorithm, or SAEM. We here describe it when the distribution of the complete data belongs to an exponential family, and is given by (97), which is mainly the situation in which it is feasible. In addition to the sequence of parameters, (θ_n), generated by the EM, this algorithm also updates a sequence of hidden variable associated to each hidden observation, with realization at step n denoted by $\eta_n^{(k)}$, $k \in \{1, \ldots, N\}$. In addition, it also updates of current estimates of the conditional expectations of the sufficient statistic, U, which will be denoted $\bar{U}_n^{(k)}$, for $k = 1, \ldots, N$. Given a sequence of gains, $(\gamma_n, n \geq 1)$, the SAEM algorithm is then defined as follows:

SA-E step: For $k = 1, \ldots, N$, sample a new hidden sample $\eta_{n+1}^{(k)}$ according to the transition probability $P_{\theta_n, \xi^{(k)}}(\eta_n^{(k)}, \cdot)$. Update the sufficient statistics via

$$\bar{U}_{n+1}^{(k)} = \bar{U}_n^{(k)} + \gamma_{n+1}\big(U(\xi^{(k)}, \eta_{n+1}^{(k)}) - \bar{U}_n^{(k)}\big).$$

M-step: let θ_{n+1} be a maximizer of

$$-\ln Z(\theta) - \left\langle \theta \,,\, \bar{U}_{n+1}^{*} \right\rangle$$

with

$$\bar{U}_{n+1}^{*} = \frac{1}{N} \sum_{k=1}^{N} \bar{U}_{n+1}^{(k)}.$$

The SAEM algorithm converges, like the EM, to a point or zero gradient of the log-likelihood, under the main assumption that $\sum_n \gamma_n = \infty$ while $\sum_n \gamma_n^{1+p} < \infty$ for some $p > 0$, plus a few technical assumptions (see [**7, 25**]).

3.4. Approximation to the E-step. One other way to deal with the difficulty of performing the E-step is to use tractable approximations of the conditional distribution. A direct way to implement this is via (95), by constraining the $\pi^{(k)}$'s in the formula to belong to some manageable class of distributions.

The simplest approximation only considers $\pi^{(k)}$'s that are concentrated at a single point (Dirac distribution), say $\pi^{(k)} = \delta_{\eta^{(k)}}$. In this case, since the entropy of a Dirac is zero, one has

$$E_{\pi^{(k)}}\left(\ln \frac{\pi_\theta(\xi^{(k)}, \eta)}{\pi^{(k)}(\eta)}\right) = \ln \pi_\theta(\xi^{(k)}, \eta^{(k)})$$

which is maximized when $\eta^{(k)}$ maximizes $\eta \mapsto \pi_\theta(\xi^{(k)}, \eta)$, or, equivalently, when $\eta^{(k)}$ is the mode of the posterior distribution $\pi_\theta(\cdot|\xi^{(k)})$. This results in what is sometimes called *mode approximation to EM*, or MAEM. This algorithm can be written as (given θ_n at step n):

MAE step: Let $\eta^{(k)}$ be a maximizer of $\ln \pi_{\theta_n}(\cdot|\xi^{(k)})$ for $k = 1, \ldots, N$.

M step: Let θ_{n+1} maximize

$$\sum_{k=1}^{N} \ln \pi_{\theta_n}(\xi^{(k)}, \eta^{(k)}).$$

If η is a collection of a large number of variables, which is typical with Bayesian networks, one can also try to restrict the distributions $\pi^{(k)}$ to only those for which the components of η are independent. Maximizing

$$E_{\pi^{(k)}}\left(\ln \frac{\pi_\theta(\xi^{(k)}, \eta)}{\pi^{(k)}(\eta)}\right)$$

subject to this constraint is identical to the mean-field approximation problem for the conditional distribution $\pi(\eta|\xi^{(k)})$, which was described in Proposition 24. This leads to the mean-field EM algorithm [**44, 3**] and results in the following iterations.

Mean-field step: For each $k = 1, \ldots, N$, compute a mean-field approximation, $\pi^{(k)}$, of the conditional density $\pi(\cdot|\xi^{(k)})$.

M step: Let θ_{n+1} maximize

$$\sum_{k=1}^{N} E_{\pi^{(k)}} \ln \pi_{\theta_n}(\xi^{(k)}, \eta).$$

Note that this algorithm provides a better approximation than the previous MAEM, since a Dirac distribution is a special case of a product distribution.

3.5. Pseudo-EM Algorithm. Another way to amend equation (95) is to replace the likelihood ratio by expressions that are more amenable to computation. We here consider a situation, which includes interesting problems with graphical models, in which η is a collection of a large number of variables and the conditional distribution of ξ given η is "simple" (for example, ξ may be a collection of variables which are independent given η). So we let $\eta = (\eta_s, s \in V)$ and define, in replacement of (96):

$$\tilde{\Delta}_\theta(\pi', \xi) = \sum_{s \in V} E_{\pi'} \left(\ln \frac{\pi_{\theta,s}(\xi, \eta_s | \eta_t, t \neq s)}{\pi'_s(\eta_s | \eta_t, t \neq s)} \right). \tag{100}$$

We now compute

$$\max_{\theta, \pi^{(1)}, \ldots, \pi^{(N)}} \sum_{k=1}^{N} \tilde{\Delta}_\theta(\pi^{(k)}, \xi^{(k)}).$$

Using an argument similar to the one done for proving that pseudo-likelihood provided a contrast, one sees that $\tilde{\Delta}(\pi', \xi)$ is maximized when

$$\pi'_s(\eta_s | \eta_t, t \neq s) = \pi_{\theta,s}(\eta_s | \xi, \eta_t, t \neq s),$$

for which the only solution is $\pi'(\eta) = \pi_\theta(\eta|\xi)$. This provides a new, "pseudo-EM", iteration, for which θ_{n+1} maximizes

$$\theta \mapsto \sum_{s \in V} \sum_{k=1}^{N} E_{\theta_n} \big(\ln \pi_{\theta,s}(\xi^{(k)}, \eta_s | \eta_t, t \in V) | \xi^{(k)} \big).$$

The pseudo-EM algorithm was essentially introduced in [**4**] (in which it was called the EM-Gibbs algorithm) to learn parameters for imperfectly observed Gibbs distributions. In general, the expression of the conditional expectations, as functions of θ, cannot be computed explicitly, and must be estimated via Monte-Carlo simulation.

3.6. Partially-Observed Bayesian Networks. We now consider the situation in which the joint distribution of $X = (\xi, \eta)$ is a Bayesian network over a directed acyclic graph $G = (V, E)$. We assume that the set of vertices splits into two parts: $V = (S, H)$ and $\xi = X_S$, $\eta = X_H$.

Assume that $x_S^{(1)}, \ldots, x_S^{(N)}$ are observed. The parameter θ is the collection of all $p(x_s | x_{s^-})$ for $s \in V$. Define the random variables $I_{s,x}(y)$ equal to one if $y_{s \cup s^-} = x_{s \cup s^-}$ and zero otherwise. We can write

$$\begin{aligned} \ln \pi(y) &= \sum_{s \in S} \ln p_s(y_s | y_{s^-}) \\ &= \sum_{s \in S} \sum_{x_{s \cup s^-} \in F_{s \cup s^-}} \ln p_s(x_s | x_{s^-}) I_{s,x}(y) \end{aligned}$$

This implies that

$$\sum_{k=1}^{N} E_{\theta_n}\left(\ln \pi(x_S^{(k)}, X_H)|X_S = x_S^{(k)}\right) = \sum_{x_{s\cup s^-}\in F_{s\cup s^-}} \ln p_s(x_s|x_{s^-}) \sum_{k=1}^{N} E_{\theta_n}(I_{s,x}(X)|X_S = x_S^{(k)})$$

$$= \sum_{x_{s\cup s^-}\in F_{s\cup s^-}} \ln p_s(x_s|x_{s^-}) \sum_{k=1}^{N} \pi_{\theta_n}(x_{s\cup s^-}|X_S = x_S^{(k)}).$$

The EM iteration at step n then is

$$p_s^{(n+1)}(x_s|x_{s^-}) = \frac{1}{Z_s(x_{s^-})} \sum_{k=1}^{N} \pi_{\theta_n}(x_{s\cup s^-}|X_S = x_S^{(k)})$$

with

$$\pi_{\theta_n}(x) = \prod_{s\in V} p^{(n)}(x_s|x_{s^-}),$$

Z_s being a normalization constant.

If the estimation is solved with a Dirichlet prior $\mathrm{Dir}(1 + a_s(x_s, x_{s^-}))$, the update formula becomes

$$p_s^{(n+1)}(x_s|x_{s^-}) = \frac{1}{Z_s(x_{s^-})}\left(a_s(x_s, x_{s^-}) + \sum_{k=1}^{N} \pi_{\theta_n}(x_{s\cup s^-}|X_S = x_S^{(k)})\right). \tag{101}$$

Trees and Hidden Markov Models. This algorithm is very simple when the conditional distributions $\pi_{\theta_n}(x_{s\cup s^-}|X_S = x_S^{(k)})$ can be easily computed, which is not always the case for a general Bayesian network, since the conditional distribution does not always have a structure of Bayesian network. The computation is simple enough for trees, however, since conditional tree distributions are still trees (or forests). More precisely, the conditional distribution given the observed variables can be written in the form

$$\pi(x_S|y_H) = \frac{1}{Z(y_H)} \prod_{s\in S} \varphi_{s,y}(x_s) \prod_{t\sim s, s,t\subset S} \varphi_{st}(x_s, x_t)$$

with $\varphi_{s,s^-}(x_s, x_{s^-}) = p_s(x_s|x_{s^-})$ and, letting $\varphi_s(x_s) = p_s(x_s)$ is $s^- = \emptyset$ and 1 otherwise,

$$\varphi_{s,y}(x_s) = \varphi_s(x_s) \prod_{t\sim s, t\in H} \varphi_{st}(x_s, x_t).$$

So, the marginal joint distribution of a vertex and its parents are directly given by belief propagation, using the just defined interactions.

We therefore get a nice training algorithm for tree distributions with partial observations, that we summarize as follows. Starting with some initial guess of the conditional probabilities (for example, those given by the prior), the transition from step n to step $n+1$ is as follows.

(1) For $k = 1, \ldots, N$, use belief propagation (or sum-prod) to compute all $\pi_{\theta_n}(x_{s\cup s^-}|X_S = x_S^{(k)})$. Note that these probabilities can be 0 or 1 when s and/or s^- are observed.
(2) Use (101) to compute the next set of parameters.

The tree case includes the important example of hidden Markov models, which are defined as follows. S and H are ordered, with same cardinality, say $S = \{s_1, \ldots, s_q\}$ and $H = \{h_1, \ldots, h_q\}$. Edges are $(h_1, h_2), \ldots, (h_{q-1}, h_q)$ and $(h_1, s_1), \ldots, (h_q, s_q)$. The interpretation generally is that the hidden variables, h_s, are the variables of interest, and behave like a Markov chain, and that the observations, x_s, are either noisy or transformed versions of them. A major application is in speech recognition, where the h_s's are labels that represent specific phonemes (little pieces of spoken words) and the x_s's are measured signals. The $h \to h$ transitions then describe how phonemes are likely to appear in sequence for a given language, and the $h \to s$ links describe how each phoneme is likely to be pronounced and heard.

General Bayesian Networks. The algorithm in the general case can move from tractable to intractable depending on the situation. This is generally to be handled in a case by case basis, by analyzing the conditional structure, for a given model, knowing the observations.

In practice, it is always possible to use loopy belief propagation to obtain some approximation of the conditional probabilities, even if it is not sure that the algorithm will converge to the correct marginals. When feasible, junction trees can be used, too. Monte-Carlo sampling is also an option, although quite computational.

3.7. Partially Observed Loopy Models. Since parameter estimation with loopy graphs and complete observations is already quite difficult, it can be expected that having hidden variables in addition won't make the problem simple. One may arguably declare that parameter estimation for general loopy networks, or Markov random fields, is still an open problem, in the sense that no satisfactory, fully general solution has been discovered yet.

Let us consider Gibbs distributions given by

$$\pi_\theta(\xi, \eta) = \frac{1}{Z_\theta} \exp(-\theta^T U(\xi, \eta))$$

in which only ξ is observable. Given a sample, $\xi^{(1)}, \ldots, \xi^{(N)}$, the EM algorithm is in principle applicable. It requires solving at each step the following problem: compute

$$\begin{aligned} \theta_{n+1} &= \text{argmax}_\theta \sum_{k=1}^{N} E_{\theta_n}(\ln \pi_\theta(\xi^{(k)}, \eta) | \xi = \xi^{(k)}) \\ &= \text{argmax}_\theta - \ln Z_\theta - \theta^T \left(\frac{1}{N} \sum_{k=1}^{N} E_{\theta_n}(U(\xi^{(k)}, \eta) | \xi = \xi^{(k)}) \right). \end{aligned}$$

This is a problem similar to the one we have discussed with perfect observations, with $\bar{U}$ replaced by

$$\bar{U}_{\theta_n} = \frac{1}{N} \sum_{k=1}^{N} E_{\theta_n}(U(\xi^{(k)}, \eta) | \xi = \xi^{(k)}).$$

From our previous discussion on maximum likelihood with complete observations, we see that the EM algorithm must iterate the following (E and M) steps at each update of the parameter:

- E step: Compute $E_{\theta_n}(U(\xi^{(k)},\eta)|\xi=\xi^{(k)})$ for each $k=1,\dots,N$, and the average $\bar{U}_{\theta_n}$.
- M step: Solve the equation

$$E_\theta(U)=\bar{U}_{\theta_n}$$

to obtain θ_{n+1}.

The problem is that, as we have seen, none of these steps can be solved exactly, and require Monte Carlo approximations. For example, the E step requires computing expectations for the conditional distributions

$$\pi_{\theta_n}(\eta|\xi^{(k)})=\frac{1}{Z_{\theta_n}(\xi^{(k)})}\exp(-\theta_n^T U(\xi^{(k)},\eta))$$

which has a Markov distribution which is, by Proposition 11, no more complex that the graph for which the joint distribution is Markov, but usually still loopy.

Given that these two steps may have to be iterated a large number of times until the parameter stabilizes, using the EM in its standard form is generally impractical. In some cases, one can use one of the approximations we previously discussed, like the mean-field and pseudo-EM. One problem, with the mean field, is that the statistical quality of the obtained estimator is unclear. Also, the mean-field approximation is not always easy to compute. The pseudo-EM can in turn be fairly computational, and generally requires that the conditional distributions of a hidden variable given the rest take a relatively simple form.

An additional option, in this setting, is to run a stochastic gradient descent algorithm that directly minimizes the likelihood of the observations. We first prove a proposition that describes the derivatives of the log likelihood.

PROPOSITION 49. *Let $(\xi,\eta)\mapsto\pi_\theta(\xi,\eta)$ be a probability density function with respect to a product measure $d\mu_\xi\, d\mu_\eta$. Let*

$$\psi_\theta(\xi)=\int \pi_\theta(\xi,\eta)d\mu_\eta.$$

Then

$$\frac{d}{d\theta}\ln\psi_\theta(\xi)=E_\theta\Big(\frac{d}{d\theta}\ln\pi_\theta(\xi,\eta)|\xi\Big). \tag{102}$$

PROOF. Indeed, writing ∂_θ for $d/d\theta$,

$$\begin{aligned}\partial_\theta\ln\psi_\theta(\xi) &= \int\frac{\partial_\theta\pi_\theta(\xi,\eta)}{\psi_\theta(\xi)}d\mu_\eta(\eta)\\ &= \int\partial_\theta\ln\pi_\theta(\xi,\eta)\frac{\pi_\theta(\xi,\eta)}{\psi_\theta(\xi)}d\mu_\eta(\eta)\\ &= E_\theta(\partial_\theta\ln\pi_\theta(\xi,\eta)|\xi).\end{aligned}$$

□

Now, consider a model given, as in (97), by

$$\pi_\theta(\xi,\eta)=\frac{1}{Z(\theta)}\exp\big(-\langle\theta\,,\,U(\xi,\eta)\rangle\big).$$

Then, from Proposition 47,

$$\partial_\theta \ln \psi_\theta = E_\theta(E_\theta(U) - U|\xi) = E_\theta(U) - E_\theta(U|\xi).$$

Given an observed sample $\xi^{(1)}, \dots, \xi^{(N)}$, one can build a stochastic gradient algorithm similar to the one provided in section 2.2, in which Monte-Carlo simulations are used for both expectations. Assume for this that one has an ergodic stochastic matrix P_θ on $M_\xi \times M_\eta$, and a family of ergodic stochastic matrices P_θ^ξ, $\xi \in M_\xi$, on M_η, such that the invariant distribution of P_θ is π_θ, and the one of P_θ^ξ is $\pi_\theta(\,\cdot\,|\xi)$. Start the algorithm with initial parameter $\theta(0)$ and initial configurations $\xi(0), \eta(0)$ and $\eta^{(k)}(0)$, $k = 1, \dots, N$. Then, at step n,

(SGH1) Sample from the distribution $P_{\theta(n)}((\xi(n), \eta(n)), \cdot)$ to obtain new configurations $\xi(n+1), \eta(n+1)$.

(SGH2) For $k = 1, \dots, N$, sample from the distribution $P_{\theta(n)}^{\xi^{(k)}}(\eta^{(k)}(n), \cdot)$ to obtain a new configuration $\eta^{(k)}(n+1)$.

(SGH3) Update the parameter using

$$\theta(n+1) = \theta(n) + \gamma(n+1)\Big(U(\xi(n+1), \eta(n+1)) - \frac{1}{N}\sum_{k=1}^{N} U(\xi^{(k)}, \eta^{(k)}(n+1))\Big). \tag{103}$$

References and further reading

The literature on graphical models, Markov random fields and Bayesian networks is huge, and this section will not provide an extensive description of its current state. We will instead refer the reader to textbooks, in which the authors did make the effort of providing an accurate list of references, and to a small selection of papers, obviously partial and definitely subjective.

Here are a few textbooks on graphical models that contributed to the writing of these notes: Pearl's book [**31**] is a basic reference, at least for historical purposes. The book by Cowell et. al. [**6**] is a good introduction to the field, and so is the one written by Neapolitan [**29**], and the book from Koller and Friedman [**23**]. Winkler's book, on Markov random fields [**40**] is also a good choice. Other references are [**21, 11, 2, 33**].

Here is a list of research papers that can also be of interest. On Markov random fields, [**1, 14, 13, 16**] for the basic theory, [**1, 42, 43, 15**] for parametric estimation; On MCMC sampling (a few out of a huge literature) [**32, 37**], with [**35, 36, 10**] on perfect sampling; On belief propagation [**38, 39, 24, 41**]; see also [**22**]; On Bayesian networks: [**12, 18**] and the numerous references in the cited textbooks.

The original reference for the EM algorithm is [**8**], the presentation in these notes being inpired by [**28**]. Extension on parametric estimation for Bayesian networks, including the estimation of the structure can be found in [**29**] and in [**18**].

Bibliography

[1] J. Besag, *Spatial interaction and the statistical analysis of lattice systems*, J. of Roy. Stat. Soc., 36 (1974), pp. 192–236.

[2] C. Borgelt and R. Kruse, *Graphical models: Methods for data analysis and mining*, Wiley, 2002.

[3] G. Celeux, F. Forbes, and N. Peyrard, *EM procedures using mean field-like approximations for Markov model-based image segmentation*, Pattern recognition, 36 (2003), pp. 131–144.

[4] B. Chalmond, *An iterative Gibbsian technique for reconstruction of m-ary images*, Pattern recognition, 22 (1989), pp. 747–761.

[5] T. Cover and J. Thomas, *Elements of information theory*, Wiley, 1991.

[6] R. G. Cowell, A. P. Dawid, S. L. Lauritzen, and D. J. Spiegelhalter, *Probabilistic networks and expert systems*, Springer, 2007.

[7] B. Delyon and M. Lavielle, *Convergence of a Stochastic Approximation Version of the EM Algorithm*, Annals of Statistics, 27 (1999), pp. 94–128.

[8] A. P. Dempster, N. M. Laird, and D. B. Rubin, *Maximum likelihood from incomplete data via the em algorithm*, J. Royal Stat. Soc., 39 (1977), pp. 1–38.

[9] E. Dijkstra, *A note on two problems in connexion with graphs*, Numerische mathematik, 1 (1959), pp. 269–271.

[10] J. A. Fill, *An interruptible algorithm for perfect sampling via Markov chains*, The Annals of Applied Probability, 8 (1998), pp. 131–162.

[11] B. J. Frey, *Graphical models for machine learning and digital communication*, MIT Press, 1998.

[12] D. Geiger, V. T., and J. Pearl, *Identifying independence in bayesian networks*, Networks, 20 (1990), pp. 507–534.

[13] D. Geman, *Random fields and inverse problems in imaging*, in Ecole d't de Saint-Flour, Lecture Notes in Mathematics, vol. 1427, New York, 1990, Springer-Verlag.

[14] S. Geman and D. Geman, *Stochastic relaxation, gibbs distributions and the bayesian restoration of images*, IEEE PAMI, 9 (1984), pp. 721–741.

[15] C. J. Geyer and E. Thompson, *Constrained monte carlo maximum likelihood for dependent data (with discussion)*, J. of Royal Stat. Soc., 54 (1992).

[16] B. Gidas, *Markov Random Fields: Theory and Applications*, Academic Press, 1991, ch. Parameter estimation for Gibbs distributions, I: Fully Observed Data.

[17] M. Gondran and M. Minoux, *Graphs and algorithms.*, JOHN WILEY & SONS, INC., 605 THIRD AVE., NEW YORK, NY 10158, USA, 1983, 550, (1983).

[18] D. Heckerman, D. Geiger, and D. M. Chickering, *Learning bayesian networks: the combination of knowledge and statistical data*, Machine learning, 20 (1995), pp. 197–243.

[19] V. Jarnik, *O jistem problemu minimalnim (about a certain minimal problem)*, Prace Moravske Prirodovedecke Spolecnosti, 6 (1930), pp. 57–63.

[20] F. Jensen and F. Jensen, *Optimal junction trees*, in Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence, 1994, pp. 360–366.

[21] F. V. Jensen, *An introduction to Bayesian networks*, Springer-Verlag, 1996.

[22] M. I. Jordan, Z. Ghahramani, T. S. Jaakola, and L. K. Saul, *An introduction to variational methods for graphical models*, Machine learning, 37 (1999), pp. 183–233.

[23] D. Koller and N. Friedman, *Probabilistic graphical models: principles and techniques*, The MIT Press, 2009.

[24] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, *Factor graphs and the sum-product algorithm*, IEEE trans. Inf. Th., 47 (2001), pp. 520–548.
[25] E. Kuhn and M. Lavielle, *COUPLING A STOCHASTIC APPROXIMATION VERSION OF EM*, ESAIM: Probability and Statistics, 8 (2004), pp. 115–131.
[26] A. a. Margolin, I. Nemenman, K. Basso, C. Wiggins, G. Stolovitzky, R. Dalla Favera, and A. Califano, *ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context.*, BMC bioinformatics, 7 Suppl 1 (2006), p. S7.
[27] J. Mchugh, *Algorithm graph theory*, New Jersey: Prentice-Hall Inc, 1990.
[28] R. M. Neal and G. E. Hinton, *A view of the em algorithm that justifies incremental, sparse and other variants*, in Learning in graphical models, M. I. Jordan, ed., Kluwer, 1998, pp. 355–368.
[29] R. E. Neapolitan, *Learning Bayesian networks*, Prentice Hall, 2004.
[30] P. Pardalos and J. Xue, *The maximum clique problem*, Journal of Global Optimization, 4 (1994), pp. 301–328.
[31] J. Pearl, *Probabilistic reasoning in intelligent systems*, Morgan Kaufmann, 1988.
[32] H. Peskun, P, *Optimum monte carlo sampling using markov chains*, Biometrika, 60 (1973), pp. 607–612.
[33] O. Pourret, P. Naï m, and B. Marcot, eds., *Bayesian netowrks: a practical guide to applications*, Wiley, 2008.
[34] R. Prim, *Shortest connection networks and some generalizations*, Bell system technical journal, 36 (1957), pp. 1389–1401.
[35] J. G. Propp and D. B. Wilson, *Exact sampling with coupled Markov chains and applications to statistical mechanics*, Random Structures and Algorithms, 9 (1996), pp. 223–252.
[36] ——, *How to get a perfectly random sample from a generic Markov chain and generate a random spanning tree of a directed graph*, Journal of Algorithms, 27 (1998), pp. 170–217.
[37] A. Sokal, *Monte carlo methods in statistical mechanics*. Lecture notes: Cours de troisième cycle de la physique en Suisse Romande, Lausanne, 1989.
[38] Y. Weiss, *Correctness of local probability propagation in graphical models with loops*, Neural computation, 12 (2000), pp. 1–41.
[39] Y. Weiss and W. T. Freeman, *Correctness of belief propagation in gaussian graphical models of arbitrary topology*, Neural computation, 13 (2001), pp. 2173–2200.
[40] G. Winkler, *Image analysis, random fields and Markov chain Monte Carlo methods*, Springer, 1995,2003.
[41] J. S. Yedidia, W. T. Freeman, and Y. Weiss, *Constructing free-energy approximations and generalized belief propagation algorithms*, IEEE trans. Inf. Th., 51 (2005), pp. 2282–2312.
[42] L. Younes, *Estimation and annealing for gibbsian fields*, Ann. de l'Inst. Henri Poincaré, 2 (1988).
[43] ——, *Parametric inference for imperfectly observed gibbsian fields*, Prob. Thry. Rel. Fields, 82 (1989), pp. 625–645.
[44] J. Zhang, *The mean field theory in EM procedures for Markov random fields*, Signal Processing, IEEE Transactions on, 40 (1992), pp. 2570–2583.