4 Undirected Graphical Models

So far, we have dealt only with directed graphical models, or Bayesian networks. These models are useful because both the structure and the parameters provide a natural representation for many types of real-world domains. In this chapter, we turn our attention to another important class of graphical models, defined on the basis of undirected graphs.

As we will see, these models are useful in modeling a variety of phenomena where one cannot naturally ascribe a directionality to the interaction between variables. Furthermore, the undirected models also offer a different and often simpler perspective on directed models, in terms of both the independence structure and the inference task. We also introduce a combined framework that allows both directed and undirected edges. We note that, unlike our results in the previous chapter, some of the results in this chapter require that we restrict attention to distributions over discrete state spaces.

4.1 The Misconception Example

To motivate our discussion of an alternative graphical representation, let us reexamine the Misconception example of section 3.4.2 (example 3.8). In this example, we have four students who get together in pairs to work on their homework for a class. The pairs that meet are shown via the edges in the undirected graph of figure 3.10a.

As we discussed, we intuitively want to model a distribution that satisfies $(A \perp C \mid \{B, D\})$ and $(B \perp D \mid \{A, C\})$, but no other independencies. As we showed, these independencies cannot be naturally captured in a Bayesian network: any Bayesian network I-map of such a distribution would necessarily have extraneous edges, and it would not capture at least one of the desired independence statements. More broadly, a Bayesian network requires that we ascribe a directionality to each influence. In this case, the interactions between the variables seem symmetrical, and we would like a model that allows us to represent these correlations without forcing a specific direction to the influence.

Markov network

A representation that implements this intuition is an undirected graph. As in a Bayesian network, the nodes in the graph of a *Markov network* represent the variables, and the edges correspond to a notion of direct probabilistic interaction between the neighboring variables — an interaction that is not mediated by any other variable in the network. In this case, the graph of figure 3.10, which captures the interacting pairs, is precisely the Markov network structure that captures our intuitions for this example. As we will see, this similarity is not an accident.

$\phi_1(A,B)$			$\phi_2(B,C)$			$\phi_3(C,D)$			$\phi_4(D,A)$		
a^0	b^0	30	b^0	c^0	100	c^0	d^0	1	d^0	a^0	100
a^0	b^1	5	b^0	c^1	1	c^0	d^1	100	d^0	a^1	1
a^1	b^0	1	b^1	c^0	1	c^1	d^0	100	d^1	a^0	1
a^1	b^1	10	b^1	c^1	100	c^1	d^1	1	d^1	a^1	100
(a)			(b)			(c)			(d)		

Figure 4.1 Factors for the Misconception **example**

The remaining question is how to parameterize this undirected graph. Because the interaction is not directed, there is no reason to use a standard CPD, where we represent the distribution over one node given others. Rather, we need a more symmetric parameterization. Intuitively, what we want to capture is the *affinities* between related variables. For example, we might want to represent the fact that Alice and Bob are more likely to agree than to disagree. We associate with A, B a general-purpose function, also called a *factor*:

Definition 4.1 factor

Let $\mathbf{D}$ be a set of random variables. We define a factor ϕ *to be a function from $Val(\mathbf{D})$ to $\mathbb{R}$. A factor is* nonnegative *if all its entries are nonnegative. The set of variables $\mathbf{D}$ is called the* scope *of the factor and denoted $Scope[\phi]$.* ■

scope

Unless stated otherwise, we restrict attention to nonnegative factors.

In our example, we have a factor $\phi_1(A, B) : Val(A, B) \mapsto \mathbb{R}^+$. The value associated with a particular assignment a, b denotes the affinity between these two values: the higher the value $\phi_1(a, b)$, the more compatible these two values are.

Figure 4.1a shows one possible compatibility factor for these variables. Note that this factor is not normalized; indeed, the entries are not even in $[0, 1]$. Roughly speaking, $\phi_1(A, B)$ asserts that it is more likely that Alice and Bob agree. It also adds more weight for the case where they are both right than for the case where they are both wrong. This factor function also has the property that $\phi_1(a^1, b^0) < \phi_1(a^0, b^1)$. Thus, if they disagree, there is less weight for the case where Alice has the misconception but Bob does not than for the converse case.

In a similar way, we define a compatibility factor for each other interacting pair: $\{B, C\}$, $\{C, D\}$, and $\{A, D\}$. Figure 4.1 shows one possible choice of factors for all four pairs. For example, the factor over C, D represents the compatibility of Charles and Debbie. It indicates that Charles and Debbie argue all the time, so that the most likely instantiations are those where they end up disagreeing.

As in a Bayesian network, the parameterization of the Markov network defines the local interactions between directly related variables. To define a global model, we need to combine these interactions. As in Bayesian networks, we combine the local models by multiplying them. Thus, we want $P(a, b, c, d)$ to be $\phi_1(a, b) \cdot \phi_2(b, c) \cdot \phi_3(c, d) \cdot \phi_4(d, a)$. In this case, however, we have no guarantees that the result of this process is a normalized joint distribution. Indeed, in this example, it definitely is not. Thus, we define the distribution by taking the product of

Assignment				*Unnormalized*	*Normalized*
a^0	b^0	c^0	d^0	300,000	0.04
a^0	b^0	c^0	d^1	300,000	0.04
a^0	b^0	c^1	d^0	300,000	0.04
a^0	b^0	c^1	d^1	30	$4.1 \cdot 10^{-6}$
a^0	b^1	c^0	d^0	500	$6.9 \cdot 10^{-5}$
a^0	b^1	c^0	d^1	500	$6.9 \cdot 10^{-5}$
a^0	b^1	c^1	d^0	5,000,000	0.69
a^0	b^1	c^1	d^1	500	$6.9 \cdot 10^{-5}$
a^1	b^0	c^0	d^0	100	$1.4 \cdot 10^{-5}$
a^1	b^0	c^0	d^1	1,000,000	0.14
a^1	b^0	c^1	d^0	100	$1.4 \cdot 10^{-5}$
a^1	b^0	c^1	d^1	100	$1.4 \cdot 10^{-5}$
a^1	b^1	c^0	d^0	10	$1.4 \cdot 10^{-6}$
a^1	b^1	c^0	d^1	100,000	0.014
a^1	b^1	c^1	d^0	100,000	0.014
a^1	b^1	c^1	d^1	100,000	0.014

Figure 4.2 Joint distribution for the Misconception **example.** The unnormalized measure and the normalized joint distribution over A, B, C, D, obtained from the parameterization of figure 4.1. The value of the partition function in this example is 7,201,840.

the local factors, and then normalizing it to define a legal distribution. Specifically, we define

$$P(a, b, c, d) = \frac{1}{Z}\phi_1(a, b) \cdot \phi_2(b, c) \cdot \phi_3(c, d) \cdot \phi_4(d, a),$$

where

$$Z = \sum_{a,b,c,d} \phi_1(a, b) \cdot \phi_2(b, c) \cdot \phi_3(c, d) \cdot \phi_4(d, a)$$

partition function

Markov random field

is a normalizing constant known as the *partition function.* The term "partition" originates from the early history of Markov networks, which originated from the concept of *Markov random field* (or *MRF*) in statistical physics (see box 4.C); the "function" is because the value of Z is a function of the parameters, a dependence that will play a significant role in our discussion of learning.

In our example, the unnormalized measure (the simple product of the four factors) is shown in the next-to-last column in figure 4.2. For example, the entry corresponding to a^1, b^1, c^0, d^1 is obtained by multiplying:

$$\phi_1(a^1, b^1) \cdot \phi_2(b^1, c^0) \cdot \phi_3(c^0, d^1) \cdot \phi_4(d^1, a^1) = 10 \cdot 1 \cdot 100 \cdot 100 = 100,000.$$

The last column shows the normalized distribution.

We can use this joint distribution to answer queries, as usual. For example, by summing out A, C, and D, we obtain $P(b^1) \approx 0.26$ and $P(b^0) \approx 0.74$; that is, Bob is 26 percent likely to have the misconception. On the other hand, if we now observe that Charles does not have the misconception (c^0), we obtain $P(b^1 \mid c^0) \approx 0.06$.

The benefit of this representation is that it allows us great flexibility in representing interactions between variables. For example, if we want to change the nature of the interaction between A and B, we can simply modify the entries in that factor, without having to deal with normalization constraints and the interaction with other factors. The flip side of this flexibility, as we will see later, is that the effects of these changes are not always intuitively understandable.

As in Bayesian networks, there is a tight connection between the factorization of the distribution and its independence properties. The key result here is stated in exercise 2.5: $P \models (\boldsymbol{X} \perp \boldsymbol{Y} \mid \boldsymbol{Z})$ if and only if we can write P in the form $P(\mathcal{X}) = \phi_1(\boldsymbol{X}, \boldsymbol{Z})\phi_2(\boldsymbol{Y}, \boldsymbol{Z})$. In our example, the structure of the factors allows us to decompose the distribution in several ways; for example:

$$P(A, B, C, D) = \left[\frac{1}{Z}\phi_1(A, B)\phi_2(B, C)\right] \phi_3(C, D)\phi_4(A, D).$$

From this decomposition, we can infer that $P \models (B \perp D \mid A, C)$. We can similarly infer that $P \models (A \perp C \mid B, D)$. These are precisely the two independencies that we tried, unsuccessfully, to achieve using a Bayesian network, in example 3.8. Moreover, these properties correspond to our intuition of "paths of influence" in the graph, where we have that B and D are separated given A, C, and that A and C are separated given B, D. Indeed, as in a Bayesian network, independence properties of the distribution P correspond directly to separation properties in the graph over which P factorizes.

4.2 Parameterization

We begin our formal discussion by describing the parameterization used in the class of undirected graphical models that are the focus of this chapter. In the next section, we make the connection to the graph structure and demonstrate how it captures the independence properties of the distribution.

To represent a distribution, we need to associate the graph structure with a set of parameters, in the same way that CPDs were used to parameterize the directed graph structure. However, the parameterization of Markov networks is not as intuitive as that of Bayesian networks, since the factors do not correspond either to probabilities or to conditional probabilities. As a consequence, the parameters are not intuitively understandable, making them hard to elicit from people. As we will see in chapter 20, they are also significantly harder to estimate from data.

4.2.1 Factors

A key issue in parameterizing a Markov network is that the representation is undirected, so that the parameterization cannot be directed in nature. We therefore use factors, as defined in definition 4.1. Note that a factor subsumes both the notion of a joint distribution and the notion of a CPD. A joint distribution over $\boldsymbol{D}$ is a factor over $\boldsymbol{D}$: it specifies a real number for every assignment of values of $\boldsymbol{D}$. A conditional distribution $P(X \mid \boldsymbol{U})$ is a factor over $\{X\} \cup \boldsymbol{U}$. However, both CPDs and joint distributions must satisfy certain normalization constraints (for example, in a joint distribution the numbers must sum to 1), whereas there are no constraints on the parameters in a factor.

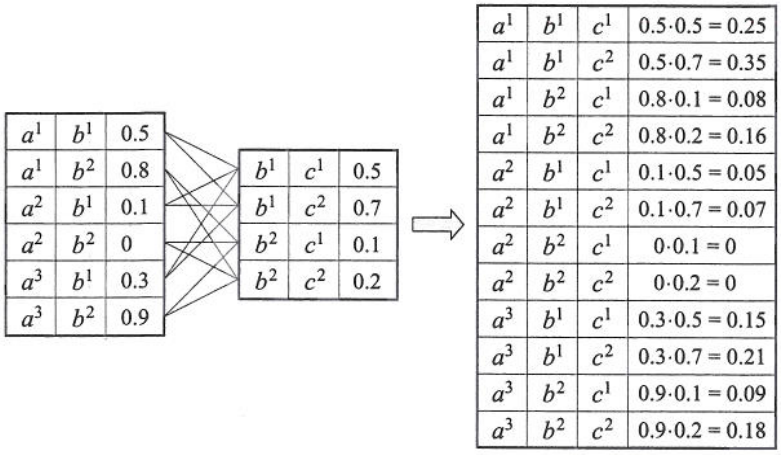


Figure 4.3 An example of factor product

As we discussed, we can view a factor as roughly describing the "compatibilities" between different values of the variables in its scope. We can now parameterize the graph by associating a set of a factors with it. One obvious idea might be to associate parameters directly with the edges in the graph. However, a simple calculation will convince us that this approach is insufficient to parameterize a full distribution.

Example 4.1

Consider a fully connected graph over $\mathcal{X}$; in this case, the graph specifies no conditional independence assumptions, so that we should be able to specify an arbitrary joint distribution over $\mathcal{X}$. If all of the variables are binary, each factor over an edge would have 4 parameters, and the total number of parameters in the graph would be $4\binom{n}{2}$. However, the number of parameters required to specify a joint distribution over n binary variables is $2^n - 1$. Thus, pairwise factors simply do not have enough parameters to encompass the space of joint distributions. More intuitively, such factors capture only the pairwise interactions, and not interactions that involve combinations of values of larger subsets of variables. ■

A more general representation can be obtained by allowing factors over arbitrary subsets of variables. To provide a formal definition, we first introduce the following important operation on factors.

Definition 4.2
factor product

Let $\boldsymbol{X}$, $\boldsymbol{Y}$, and $\boldsymbol{Z}$ be three disjoint sets of variables, and let $\phi_1(\boldsymbol{X}, \boldsymbol{Y})$ and $\phi_2(\boldsymbol{Y}, \boldsymbol{Z})$ be two factors. We define the factor product *$\phi_1 \times \phi_2$ to be a factor $\psi : Val(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}) \mapsto \mathbb{R}$ as follows:*

$$\psi(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}) = \phi_1(\boldsymbol{X}, \boldsymbol{Y}) \cdot \phi_2(\boldsymbol{Y}, \boldsymbol{Z}).$$ ■

The key aspect to note about this definition is the fact that the two factors ϕ_1 and ϕ_2 are multiplied in a way that "matches up" the common part $\boldsymbol{Y}$. Figure 4.3 shows an example of the product of two factors. We have deliberately chosen factors that do not correspond either to probabilities or to conditional probabilities, in order to emphasize the generality of this operation.

As we have already observed, both CPDs and joint distributions are factors. Indeed, the chain rule for Bayesian networks defines the joint distribution factor as the product of the CPD factors. For example, when computing $P(A, B) = P(A)P(B \mid A)$, we always multiply entries in the $P(A)$ and $P(B \mid A)$ tables that have the same value for A. Thus, letting $\phi_{X_i}(X_i, \mathrm{Pa}_{X_i})$ represent $P(X_i \mid \mathrm{Pa}_{X_i})$, we have that

$$P(X_1, \ldots, X_n) = \prod_i \phi_{X_i}.$$

4.2.2 Gibbs Distributions and Markov Networks

We can now use the more general notion of factor product to define an undirected parameterization of a distribution.

Definition 4.3 Gibbs distribution

A distribution P_Φ is a Gibbs distribution *parameterized by a set of factors $\Phi = \{\phi_1(\boldsymbol{D}_1), \ldots, \phi_K(\boldsymbol{D}$ if it is defined as follows:*

$$P_\Phi(X_1, \ldots, X_n) = \frac{1}{Z}\tilde{P}_\Phi(X_1, \ldots, X_n),$$

where

$$\tilde{P}_\Phi(X_1, \ldots, X_n) = \phi_1(\boldsymbol{D}_1) \times \phi_2(\boldsymbol{D}_2) \times \cdots \times \phi_m(\boldsymbol{D}_m)$$

is an unnormalized measure and

$$Z = \sum_{X_1, \ldots, X_n} \tilde{P}_\Phi(X_1, \ldots, X_n)$$

partition function

is a normalizing constant called the partition function. ■

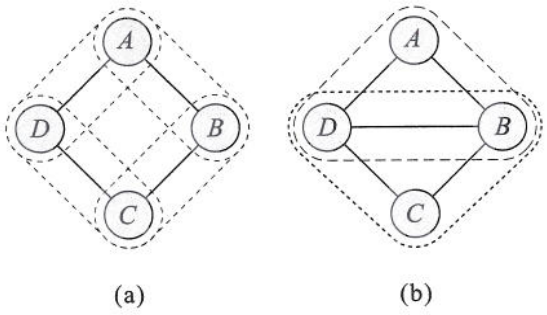
It is tempting to think of the factors as representing the marginal probabilities of the variables in their scope. Thus, looking at any individual factor, we might be led to believe that the behavior of the distribution defined by the Markov network as a whole corresponds to the behavior defined by the factor. However, this intuition is overly simplistic. **A factor is only one contribution to the overall joint distribution. The distribution as a whole has to take into consideration the contributions from all of the factors involved.**

Example 4.2

Consider the distribution of figure 4.2. The marginal distribution over A, B, is

a^0	b^0	0.13
a^0	b^1	0.69
a^1	b^0	0.14
a^1	b^1	0.04

The most likely configuration is the one where Alice and Bob disagree. By contrast, the highest entry in the factor $\phi_1(A, B)$ in figure 4.1 corresponds to the assignment a^0, b^0. The reason for the discrepancy is the influence of the other factors on the distribution. In particular, $\phi_3(C, D)$ asserts that Charles and Debbie disagree, whereas $\phi_2(B, C)$ and $\phi_4(D, A)$ assert that Bob and Charles agree and that Debbie and Alice agree. Taking just these factors into consideration, we would conclude that Alice and Bob are likely to disagree. In this case, the "strength" of these other factors is much stronger than that of the $\phi_1(A, B)$ factor, so that the influence of the latter is overwhelmed. ■



(a) (b)

Figure 4.4 The cliques in two simple Markov networks. In (a), the cliques are the pairs $\{A, B\}$, $\{B, C\}$, $\{C, D\}$, and $\{D, A\}$. In (b), the cliques are $\{A, B, D\}$ and $\{B, C, D\}$.

We now want to relate the parameterization of a Gibbs distribution to a graph structure. If our parameterization contains a factor whose scope contains both X and Y, we are introducing a direct interaction between them. Intuitively, we would like these direct interactions to be represented in the graph structure. Thus, if our parameterization contains such a factor, we would like the associated Markov network structure $\mathcal{H}$ to contain an edge between X and Y.

Definition 4.4 Markov network factorization

We say that a distribution P_Φ with $\Phi = \{\phi_1(\boldsymbol{D}_1), \ldots, \phi_K(\boldsymbol{D}_K)\}$ factorizes *over a Markov network $\mathcal{H}$ if each $\boldsymbol{D}_k$ $(k = 1, \ldots, K)$ is a complete subgraph of $\mathcal{H}$.* ■

clique potentials

The factors that parameterize a Markov network are often called *clique potentials*.

As we will see, if we associate factors only with complete subgraphs, as in this definition, we are not violating the independence assumptions induced by the network structure, as defined later in this chapter.

Note that, because every complete subgraph is a subset of some (maximal) clique, we can reduce the number of factors in our parameterization by allowing factors only for maximal cliques. More precisely, let $\boldsymbol{C}_1, \ldots, \boldsymbol{C}_k$ be the cliques in $\mathcal{H}$. We can parameterize P using a set of factors $\phi_1(\boldsymbol{C}_1), \ldots, \phi_l(\boldsymbol{C}_l)$. Any factorization in terms of complete subgraphs can be converted into this form simply by assigning each factor to a clique that encompasses its scope and multiplying all of the factors assigned to each clique to produce a clique potential. In our Misconception example, we have four cliques: $\{A, B\}$, $\{B, C\}$, $\{C, D\}$, and $\{A, D\}$. Each of these cliques can have its own clique potential. One possible setting of the parameters in these clique potential is shown in figure 4.1. Figure 4.4 shows two examples of a Markov network and the (maximal) cliques in that network.

☞ **Although it can be used without loss of generality, the parameterization using maximal clique potentials generally obscures structure that is present in the original set of factors.** For example, consider the Gibbs distribution described in example 4.1. Here, we have a potential for every pair of variables, so the Markov network associated with this distribution is a single large clique containing all variables. If we associate a factor with this single clique, it would be exponentially large in the number of variables, whereas the original parameterization in terms of edges requires only a quadratic number of parameters. See section 4.4.1.1 for further discussion.

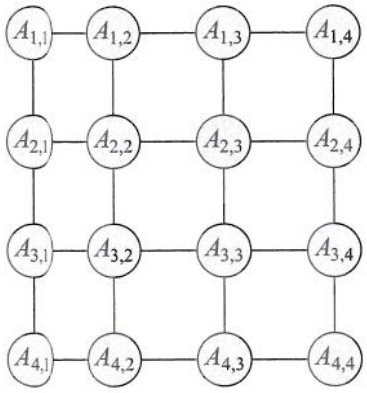


Figure 4.A.1 — A pairwise Markov network (MRF) structured as a grid.

pairwise Markov network

node potential

edge potential

Box 4.A — Concept: Pairwise Markov Networks. *A subclass of Markov networks that arises in many contexts is that of* pairwise Markov networks, *representing distributions where all of the factors are over single variables or pairs of variables. More precisely, a pairwise Markov network over a graph $\mathcal{H}$ is associated with a set of* node potentials $\{\phi(X_i) : i = 1, \ldots, n\}$ *and a set of* edge potentials $\{\phi(X_i, X_j) : (X_i, X_j) \in \mathcal{H}\}$. *The overall distribution is (as always) the normalized product of all of the potentials (both node and edge). Pairwise MRFs are attractive because of their simplicity, and because interactions on edges are an important special case that often arises in practice (see, for example, box 4.C and box 4.B).*

A class of pairwise Markov networks that often arises, and that is commonly used as a benchmark for inference, is the class of networks structured in the form of a grid, as shown in figure 4.A.1. As we discuss in the inference chapters of this book, although these networks have a simple and compact representation, they pose a significant challenge for inference algorithms.

4.2.3 Reduced Markov Networks

We end this section with one final concept that will prove very useful in later sections. Consider the process of conditioning a distribution on some assignment $\boldsymbol{u}$ to some subset of variables $\boldsymbol{U}$. Conditioning a distribution corresponds to eliminating all entries in the joint distribution that are inconsistent with the event $\boldsymbol{U} = \boldsymbol{u}$, and renormalizing the remaining entries to sum to 1. Now, consider the case where our distribution has the form P_Φ for some set of factors Φ. Each entry in the unnormalized measure $\tilde{P}_\Phi$ is a product of entries from the factors Φ, one entry from each factor. If, in some factor, we have an entry that is inconsistent with $\boldsymbol{U} = \boldsymbol{u}$, it will only contribute to entries in $\tilde{P}_\Phi$ that are also inconsistent with this event. Thus, we can eliminate all such entries from every factor in Φ.

More generally, we can define:

a^1	b^1	c^1	0.25
a^1	b^2	c^1	0.08
a^2	b^1	c^1	0.05
a^2	b^2	c^1	0
a^3	b^1	c^1	0.15
a^3	b^2	c^1	0.09

Figure 4.5 Factor reduction: The factor computed in figure 4.3, reduced to the context $C = c^1$.

Definition 4.5 factor reduction

Let $\phi(\boldsymbol{Y})$ be a factor, and $\boldsymbol{U} = \boldsymbol{u}$ an assignment for $\boldsymbol{U} \subseteq \boldsymbol{Y}$. We define the reduction *of the factor ϕ to the context $\boldsymbol{U} = \boldsymbol{u}$, denoted $\phi[\boldsymbol{U} = \boldsymbol{u}]$ (and abbreviated $\phi[\boldsymbol{u}]$), to be a factor over scope $\boldsymbol{Y}' = \boldsymbol{Y} - \boldsymbol{U}$, such that*

$$\phi[\boldsymbol{u}](\boldsymbol{y}') = \phi(\boldsymbol{y}', \boldsymbol{u}).$$

For $\boldsymbol{U} \not\subset \boldsymbol{Y}$, we define $\phi[\boldsymbol{u}]$ to be $\phi[\boldsymbol{U}' = \boldsymbol{u}']$, where $\boldsymbol{U}' = \boldsymbol{U} \cap \boldsymbol{Y}$, and $\boldsymbol{u}' = \boldsymbol{u}\langle \boldsymbol{U}' \rangle$, where $\boldsymbol{u}\langle \boldsymbol{U}' \rangle$ denotes the assignment in $\boldsymbol{u}$ to the variables in $\boldsymbol{U}'$. ■

Figure 4.5 illustrates this operation, reducing the of figure 4.3 to the context $C = c^1$.

Now, consider a product of factors. An entry in the product is consistent with $\boldsymbol{u}$ if and only if it is a product of entries that are all consistent with $\boldsymbol{u}$. We can therefore define:

Definition 4.6 reduced Gibbs distribution

Let P_Φ be a Gibbs distribution parameterized by $\Phi = \{\phi_1, \ldots, \phi_K(\})$ and let $\boldsymbol{u}$ be a context. The reduced Gibbs distribution *$\tilde{P}[\boldsymbol{u}]$ is the Gibbs distribution defined by the set of factors $\Phi[u] = \{\phi_1[\boldsymbol{u}], \ldots, \phi_K[\boldsymbol{u}]\}$.* ■

Reducing the set of factors defining P_Φ to some context $\boldsymbol{u}$ corresponds directly to the operation of conditioning P_Φ on the observation $\boldsymbol{u}$. More formally:

Proposition 4.1

Let $P_\Phi(\boldsymbol{X})$ be a Gibbs distribution. Then $P_\Phi[\boldsymbol{u}] = P_\Phi(\boldsymbol{W} \mid \boldsymbol{u})$ where $\boldsymbol{W} = \boldsymbol{X} - \boldsymbol{U}$.

Thus, to condition a Gibbs distribution on a context $\boldsymbol{u}$, we simply reduce every one of its factors to that context. Intuitively, the renormalization step needed to account for $\boldsymbol{u}$ is simply folded into the standard renormalization of any Gibbs distribution. This result immediately provides us with a construction for the Markov network that we obtain when we condition the associated distribution on some observation $\boldsymbol{u}$.

Definition 4.7 reduced Markov network

Let $\mathcal{H}$ be a Markov network over $\boldsymbol{X}$ and $\boldsymbol{U} = \boldsymbol{u}$ a context. The reduced Markov network *$\mathcal{H}[\boldsymbol{u}]$ is a Markov network over the nodes $\boldsymbol{W} = \boldsymbol{X} - \boldsymbol{U}$, where we have an edge X—Y if there is an edge X—Y in $\mathcal{H}$.* ■

Proposition 4.2

Let $P_\Phi(\boldsymbol{X})$ be a Gibbs distribution that factorizes over $\mathcal{H}$, and $\boldsymbol{U} = \boldsymbol{u}$ a context. Then $P_\Phi[\boldsymbol{u}]$ factorizes over $\mathcal{H}[\boldsymbol{u}]$.

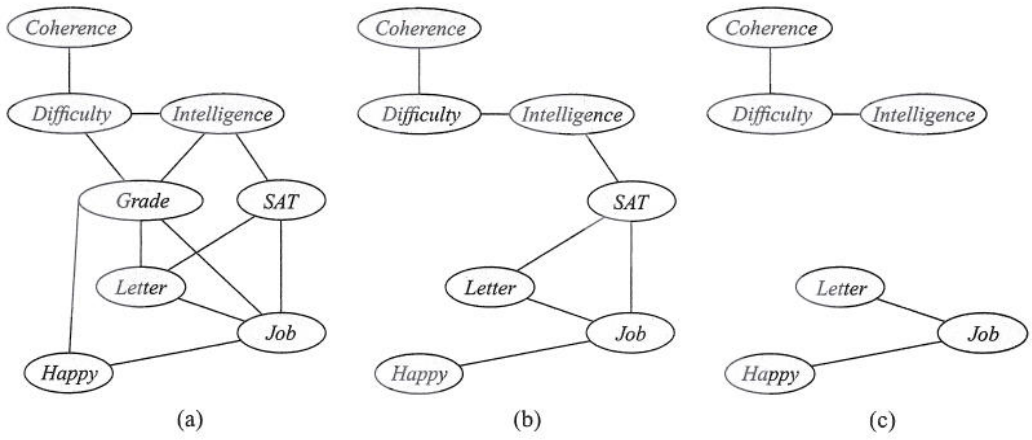


Figure 4.6 Markov networks for the factors in an extended Student **example:** (a) The initial set of factors; (b) Reduced to the context $G = g$; (c) Reduced to the context $G = g, S = s$.

Note the contrast to the effect of conditioning in a Bayesian network: Here, conditioning on a context $\boldsymbol{u}$ only eliminates edges from the graph; in a Bayesian network, conditioning on evidence can activate a v-structure, creating new dependencies. We return to this issue in section 4.5.1.1.

Example 4.3

Consider, for example, the Markov network shown in figure 4.6a; as we will see, this network is the Markov network required to capture the distribution encoded by an extended version of our Student *Bayesian network (see figure 9.8). Figure 4.6b shows the same Markov network reduced over a context of the form $G = g$, and (c) shows the network reduced over a context of the form $G = g, S = s$. As we can see, the network structures are considerably simplified.* ■

Box 4.B — Case Study: Markov Networks for Computer Vision. *One important application ar for Markov networks is computer vision. Markov networks, typically called MRFs in this vision community, have been used for a wide variety of visual processing tasks, such as image segmentation, removal of blur or noise, stereo reconstruction, object recognition, and many more.*

In most of these applications, the network takes the structure of a pairwise MRF, where the variables correspond to pixels and the edges (factors) to interactions between adjacent pixels in the grid that represents the image; thus, each (interior) pixel has exactly four neighbors. The value space of the variables and the exact form of factors depend on the task. These models are usually formulated in terms of energies (negative log-potentials), so that values represent "penalties," and a lower value corresponds to a higher-probability configuration.

image denoising

In image denoising, *for example, the task is to restore the "true" value of all of the pixels given possibly noisy pixel values. Here, we have a node potential for each pixel X_i that penalizes large discrepancies from the observed pixel value y_i. The edge potential encodes a preference for continuity between adjacent pixel values, penalizing cases where the inferred value for X_i is too*

far from the inferred pixel value for one of its neighbors X_j. However, it is important not to overpenalize true disparities (such as edges between objects or regions), leading to oversmoothing of the image. Thus, we bound the penalty, using, for example, some truncated norm, as described in box 4.D: $\epsilon(x_i, x_j) = \min(c\|x_i - x_j\|_p, \text{dist}_{\max})$ (for $p \in \{1, 2\}$).

stereo reconstruction

Slight variants of the same model are used in many other applications. For example, in stereo reconstruction, the goal is to reconstruct the depth disparity of each pixel in the image. Here, the values of the variables represent some discretized version of the depth dimension (usually more finely discretized for distances close to the camera and more coarsely discretized as the distance from the camera increases). The individual node potential for each pixel X_i uses standard techniques from computer vision to estimate, from a pair of stereo images, the individual depth disparity of this pixel. The edge potentials, precisely as before, often use a truncated metric to enforce continuity of the depth estimates, with the truncation avoiding an overpenalization of true depth disparities (for example, when one object is partially in front of the other). Here, it is also quite common to make the penalty inversely proportional to the image gradient between the two pixels, allowing a smaller penalty to be applied in cases where a large image gradient suggests an edge between the pixels, possibly corresponding to an occlusion boundary.

image segmentation

In image segmentation, the task is to partition the image pixels into regions corresponding to distinct parts of the scene. There are different variants of the segmentation task, many of which can be formulated as a Markov network. In one formulation, known as multiclass segmentation, each variable X_i has a domain $\{1, \ldots, K\}$, where the value of X_i represents a region assignment for pixel i (for example, grass, water, sky, car). Since classifying every pixel can be computationally expensive, some state-of-the-art methods for image segmentation and other tasks first oversegment the image into superpixels (or small coherent regions) and classify each region — all pixels within a region are assigned the same label. The oversegmented image induces a graph in which there is one node for each superpixel and an edge between two nodes if the superpixels are adjacent (share a boundary) in the underlying image. We can now define our distribution in terms of this graph.

Features are extracted from the image for each pixel or superpixel. The appearance features depend on the specific task. In image segmentation, for example, features typically include statistics over color, texture, and location. Often the features are clustered or provided as input to local classifiers to reduce dimensionality. The features used in the model are then the soft cluster assignments or local classifier outputs for each superpixel. The node potential for a pixel or superpixel is then a function of these features. We note that the factors used in defining this model depend on the specific values of the pixels in the image, so that each image defines a different probability distribution over the segment labels for the pixels or superpixels. In effect, the model used here is a conditional random field, a concept that we define more formally in section 4.6.1.

conditional random field

The model contains an edge potential between every pair of neighboring superpixels X_i, X_j. Most simply, this potential encodes a contiguity preference, with a penalty of λ whenever $X_i \neq X_j$. Again, we can improve the model by making the penalty depend on the presence of an image gradient between the two pixels. An even better model does more than penalize discontinuities. We can have nondefault values for other class pairs, allowing us to encode the fact that we more often find tigers adjacent to vegetation than adjacent to water; we can even make the model depend on the relative pixel location, allowing us to encode the fact that we usually find water below vegetation, cars over roads, and sky above everything.

Figure 4.B.1 shows segmentation results in a model containing only potentials on single pixels (thereby labeling each of them independently) versus results obtained from a model also containing

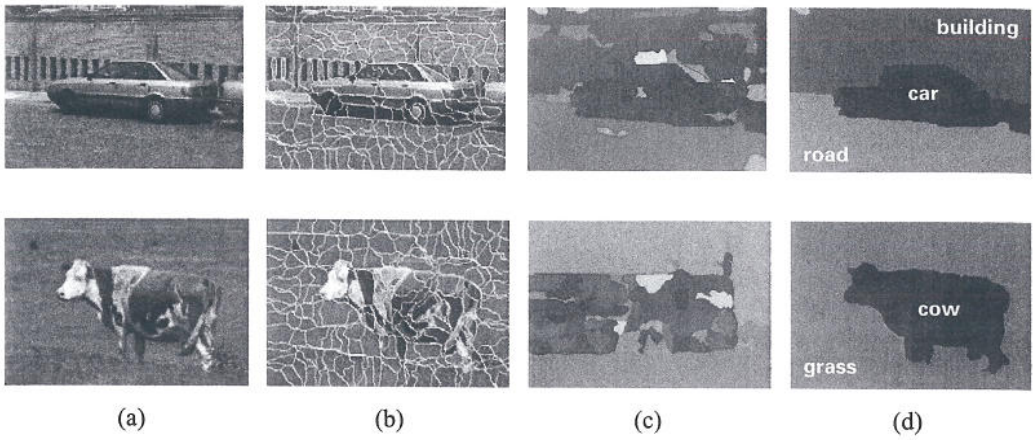


Figure 4.B.1 — Two examples of image segmentation results (a) The original image. (b) An oversegmentation known as superpixels; each superpixel is associated with a random variable that designates its segment assignment. The use of superpixels reduces the size of the problems. (c) Result of segmentation using node potentials alone, so that each superpixel is classified independently. (d) Result of segmentation using a pairwise Markov network encoding interactions between adjacent superpixels.

pairwise potentials. The difference in the quality of the results clearly illustrates the importance of modeling the correlations between the superpixels.

4.3 Markov Network Independencies

In section 4.1, we gave an intuitive justification of why an undirected graph seemed to capture the types of interactions in the Misconception example. We now provide a formal presentation of the undirected graph as a representation of independence assertions.

4.3.1 Basic Independencies

As in the case of Bayesian networks, the graph structure in a Markov network can be viewed as encoding a set of independence assumptions. Intuitively, in Markov networks, probabilistic influence "flows" along the undirected paths in the graph, but it is blocked if we condition on the intervening nodes.

observed variable

active path

Definition 4.8 *Let $\mathcal{H}$ be a Markov network structure, and let $X_1 — \ldots — X_k$ be a path in $\mathcal{H}$. Let $\mathbf{Z} \subseteq \mathcal{X}$ be a set of* observed variables. *The path $X_1 — \ldots — X_k$ is* active *given $\mathbf{Z}$ if none of the X_i's, $i = 1, \ldots, k$, is in $\mathbf{Z}$.* ■

Using this notion, we can define a notion of *separation* in the graph.

Definition 4.9 separation global independencies

We say that a set of nodes $\mathbf{Z}$ separates *$\mathbf{X}$ and $\mathbf{Y}$ in $\mathcal{H}$, denoted* $\text{sep}_{\mathcal{H}}(\mathbf{X}; \mathbf{Y} \mid \mathbf{Z})$*, if there is no active path between any node $X \in \mathbf{X}$ and $Y \in \mathbf{Y}$ given $\mathbf{Z}$. We define the* global independencies *associated with $\mathcal{H}$ to be:*

$$\mathcal{I}(\mathcal{H}) = \{(\mathbf{X} \perp \mathbf{Y} \mid \mathbf{Z}) \ : \ \text{sep}_{\mathcal{H}}(\mathbf{X}; \mathbf{Y} \mid \mathbf{Z})\}. \quad \blacksquare$$

As we will discuss, the independencies in $\mathcal{I}(\mathcal{H})$ are precisely those that are guaranteed to hold for every distribution P over $\mathcal{H}$. In other words, the separation criterion is sound for detecting independence properties in distributions over $\mathcal{H}$.

Note that the definition of separation is monotonic in $\mathbf{Z}$, that is, if $sep_{\mathcal{H}}(\mathbf{X}; \mathbf{Y} \mid \mathbf{Z})$, then $sep_{\mathcal{H}}(\mathbf{X}; \mathbf{Y} \mid \mathbf{Z}')$ for any $\mathbf{Z}' \supset \mathbf{Z}$. Thus, if we take separation as our definition of the independencies induced by the network structure, we are effectively restricting our ability to encode nonmonotonic independence relations. Recall that in the context of intercausal reasoning in Bayesian networks, nonmonotonic reasoning patterns are quite useful in many situations — for example, when two diseases are independent, but dependent given some common symptom. The nature of the separation property implies that such independence patterns cannot be expressed in the structure of a Markov network. We return to this issue in section 4.5.

As for Bayesian networks, we can show a connection between the independence properties implied by the Markov network structure, and the possibility of factorizing a distribution over the graph. As before, we can now state the analogue to both of our representation theorems for Bayesian networks, which assert the equivalence between the Gibbs factorization of a distribution P over a graph $\mathcal{H}$ and the assertion that $\mathcal{H}$ is an I-map for P, that is, that P satisfies the Markov assumptions $\mathcal{I}(\mathcal{H})$.

4.3.1.1 Soundness

soundness

We first consider the analogue to theorem 3.2, which asserts that a Gibbs distribution satisfies the independencies associated with the graph. In other words, this result states the *soundness* of the separation criterion.

Theorem 4.1

Let P be a distribution over $\mathcal{X}$, and $\mathcal{H}$ a Markov network structure over $\mathcal{X}$. If P is a Gibbs distribution that factorizes over $\mathcal{H}$, then $\mathcal{H}$ is an I-map for P.

PROOF Let $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ be any three disjoint subsets in $\mathcal{X}$ such that $\mathbf{Z}$ separates $\mathbf{X}$ and $\mathbf{Y}$ in $\mathcal{H}$. We want to show that $P \models (\mathbf{X} \perp \mathbf{Y} \mid \mathbf{Z})$.

We start by considering the case where $\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z} = \mathcal{X}$. As $\mathbf{Z}$ separates $\mathbf{X}$ from $\mathbf{Y}$, there are no direct edges between $\mathbf{X}$ and $\mathbf{Y}$. Hence, any clique in $\mathcal{H}$ is fully contained either in $\mathbf{X} \cup \mathbf{Z}$ or in $\mathbf{Y} \cup \mathbf{Z}$. Let $\mathcal{I}_{\mathbf{X}}$ be the indexes of the set of cliques that are contained in $\mathbf{X} \cup \mathbf{Z}$, and let $\mathcal{I}_{\mathbf{Y}}$ be the indexes of the remaining cliques. We know that

$$P(X_1, \ldots, X_n) = \frac{1}{Z} \prod_{i \in \mathcal{I}_{\mathbf{X}}} \phi_i(\mathbf{D}_i) \cdot \prod_{i \in \mathcal{I}_{\mathbf{Y}}} \phi_i(\mathbf{D}_i).$$

As we discussed, none of the factors in the first product involve any variable in $\mathbf{Y}$, and none in the second product involve any variable in $\mathbf{X}$. Hence, we can rewrite this product in the form:

$$P(X_1, \ldots, X_n) = \frac{1}{Z} f(\mathbf{X}, \mathbf{Z}) g(\mathbf{Y}, \mathbf{Z}).$$

From this decomposition, the desired independence follows immediately (exercise 2.5).

Now consider the case where $\boldsymbol{X} \cup \boldsymbol{Y} \cup \boldsymbol{Z} \subset \mathcal{X}$. Let $\boldsymbol{U} = \mathcal{X} - (\boldsymbol{X} \cup \boldsymbol{Y} \cup \boldsymbol{Z})$. We can partition $\boldsymbol{U}$ into two disjoint sets $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$ such that $\boldsymbol{Z}$ separates $\boldsymbol{X} \cup \boldsymbol{U}_1$ from $\boldsymbol{Y} \cup \boldsymbol{U}_2$ in $\mathcal{H}$. Using the preceding argument, we conclude that $P \models (\boldsymbol{X}, \boldsymbol{U}_1 \perp \boldsymbol{Y}, \boldsymbol{U}_2 \mid \boldsymbol{Z})$. Using the decomposition property (equation (2.8)), we conclude that $P \models (\boldsymbol{X} \perp \boldsymbol{Y} \mid \boldsymbol{Z})$. ■

Hammersley-Clifford theorem

The other direction (the analogue to theorem 3.1), which goes from the independence properties of a distribution to its factorization, is known as the *Hammersley-Clifford theorem*. Unlike for Bayesian networks, this direction does not hold in general. As we will show, it holds only under the additional assumption that P is a positive distribution (see definition 2.5).

Theorem 4.2

Let P be a positive distribution over $\mathcal{X}$, and $\mathcal{H}$ a Markov network graph over $\mathcal{X}$. If $\mathcal{H}$ is an I-map for P, then P is a Gibbs distribution that factorizes over $\mathcal{H}$.

To prove this result, we would need to use the independence assumptions to construct a set of factors over $\mathcal{H}$ that give rise to the distribution P. In the case of Bayesian networks, these factors were simply CPDs, which we could derive directly from P. As we have discussed, the correspondence between the factors in a Gibbs distribution and the distribution P is much more indirect. The construction required here is therefore significantly more subtle, and relies on concepts that we develop later in this chapter; hence, we defer the proof to section 4.4 (theorem 4.8).

This result shows that, **for positive distributions, the global independencies imply that the distribution factorizes according the network structure. Thus, for this class of distributions, we have that a distribution P factorizes over a Markov network $\mathcal{H}$ if and only if $\mathcal{H}$ is an I-map of P.** The positivity assumption is necessary for this result to hold:

Example 4.4

Consider a distribution P over four binary random variables X_1, X_2, X_3, X_4, which gives probability $1/8$ to each of the following eight configurations, and probability zero to all others:

(0,0,0,0) (1,0,0,0) (1,1,0,0) (1,1,1,0)
(0,0,0,1) (0,0,1,1) (0,1,1,1) (1,1,1,1)

Let $\mathcal{H}$ be the graph X_1—X_2—X_3—X_4—X_1. Then P satisfies the global independencies with respect to $\mathcal{H}$. For example, consider the independence $(X_1 \perp X_3 \mid X_2, X_4)$. For the assignment $X_2 = x_2^1, X_4 = x_4^0$, we have that only assignments where $X_1 = x_1^1$ receive positive probability. Thus, $P(x_1^1 \mid x_2^1, x_4^0) = 1$, and X_1 is trivially independent of X_3 in this conditional distribution. A similar analysis applies to all other cases, so that the global independencies hold. However, the distribution P does not factorize according to $\mathcal{H}$. The proof of this fact is left as an exercise (see exercise 4.1). ■

4.3.1.2 Completeness

completeness

The preceding discussion shows the soundness of the separation condition as a criterion for detecting independencies in Markov networks: any distribution that factorizes over $\mathcal{G}$ satisfies the independence assertions implied by separation. The next obvious issue is the *completeness* of this criterion.

As for Bayesian networks, the strong version of completeness does not hold in this setting. In other words, it is not the case that every pair of nodes X and Y that are not separated in $\mathcal{H}$ are dependent in every distribution P which factorizes over $\mathcal{H}$. However, as in theorem 3.3, we can use a weaker definition of completeness that does hold:

Theorem 4.3 *Let $\mathcal{H}$ be a Markov network structure. If X and Y are not separated given $\mathbf{Z}$ in $\mathcal{H}$, then X and Y are dependent given $\mathbf{Z}$ in some distribution P that factorizes over $\mathcal{H}$.*

PROOF The proof is a constructive one: we construct a distribution P that factorizes over $\mathcal{H}$ where X and Y are dependent. We assume, without loss of generality, that all variables are binary-valued. If this is not the case, we can treat them as binary-valued by restricting attention to two distinguished values for each variable.

By assumption, X and Y are not separated given $\mathbf{Z}$ in $\mathcal{H}$; hence, they must be connected by some unblocked trail. Let $X = U_1 — U_2 — \ldots — U_k = Y$ be some minimal trail in the graph such that, for all i, $U_i \notin \mathbf{Z}$, where we define a minimal trail in $\mathcal{H}$ to be a path with no shortcuts: thus, for any i and $j \neq i \pm 1$, there is no edge $U_i — U_j$. We can always find such a path: If we have a nonminimal path where we have $U_i — U_j$ for $j > i + 1$, we can always "shortcut" the original trail, converting it to one that goes directly from U_i to U_j.

For any $i = 1, \ldots, k - 1$, as there is an edge $U_i — U_{i+1}$, it follows that U_i, U_{i+1} must both appear in some clique $\mathbf{C}_i$. We pick some very large weight W, and for each i we define the clique potential $\phi_i(\mathbf{C}_i)$ to assign weight W if $U_i = U_{i+1}$ and weight 1 otherwise, regardless of the values of the other variables in the clique. Note that the cliques $\mathbf{C}_i$ for U_i, U_{i+1} and $\mathbf{C}_j$ for U_j, U_{j+1} must be different cliques: If $\mathbf{C}_i = \mathbf{C}_j$, then U_j is in the same clique as U_i, and we have an edge $U_i — U_j$, contradicting the minimality of the trail. Hence, we can define the clique potential for each clique $\mathbf{C}_i$ separately. We define the clique potential for any other clique to be uniformly 1.

We now consider the distribution P resulting from multiplying all of these clique potentials. Intuitively, the distribution $P(U_1, \ldots, U_k)$ is simply the distribution defined by multiplying the pairwise factors for the pairs U_i, U_{i+1}, regardless of the other variables (including the ones in $\mathbf{Z}$). One can verify that, in $P(U_1, \ldots, U_k)$, we have that $X = U_1$ and $Y = U_k$ are dependent. We leave the conclusion of this argument as an exercise (exercise 4.5). ■

We can use the same argument as theorem 3.5 to conclude that, for almost all distributions P that factorize over $\mathcal{H}$ (that is, for all distributions except for a set of measure zero in the space of factor parameterizations) we have that $\mathcal{I}(P) = \mathcal{I}(\mathcal{H})$.

Once again, we can view this result as telling us that our definition of $\mathcal{I}(\mathcal{H})$ is the maximal one. For any independence assertion that is not a consequence of separation in $\mathcal{H}$, we can always find a counterexample distribution P that factorizes over $\mathcal{H}$.

4.3.2 Independencies Revisited

When characterizing the independencies in a Bayesian network, we provided two definitions: the local independencies (each node is independent of its nondescendants given its parents), and the global independencies induced by d-separation. As we showed, these two sets of independencies are equivalent, in that one implies the other.

So far, our discussion for Markov networks provides only a global criterion. While the global criterion characterizes the entire set of independencies induced by the network structure, a local criterion is also valuable, since it allows us to focus on a smaller set of properties when examining the distribution, significantly simplifying the process of finding an I-map for a distribution P.

Thus, it is natural to ask whether we can provide a local definition of the independencies induced by a Markov network, analogously to the local independencies of Bayesian networks. Surprisingly, as we now show, in the context of Markov networks, there are three different possible definitions of the independencies associated with the network structure — two local ones and the global one in definition 4.9. While these definitions are related, they are equivalent only for positive distributions. As we will see, nonpositive distributions allow for deterministic dependencies between the variables. Such deterministic interactions can "fool" local independence tests, allowing us to construct networks that are not I-maps of the distribution, yet the local independencies hold.

4.3.2.1 Local Markov Assumptions

The first, and weakest, definition is based on the following intuition: Whenever two variables are directly connected, they have the potential of being directly correlated in a way that is not mediated by other variables. Conversely, when two variables are not directly linked, there must be some way of rendering them conditionally independent. Specifically, we can require that X and Y be independent given all other nodes in the graph.

Definition 4.10 *Let $\mathcal{H}$ be a Markov network. We define the* pairwise independencies *associated with $\mathcal{H}$ to be:*

pairwise independencies

$$\mathcal{I}_p(\mathcal{H}) = \{(X \perp Y \mid \mathcal{X} - \{X,Y\}) : X - Y \notin \mathcal{H}\}.$$ ■

Using this definition, we can easily represent the independencies in our Misconception example using a Markov network: We simply connect the nodes up in exactly the same way as the interaction structure between the students.

The second local definition is an undirected analogue to the local independencies associated with a Bayesian network. It is based on the intuition that we can block all influences on a node by conditioning on its immediate neighbors.

Definition 4.11 *For a given graph $\mathcal{H}$, we define the* Markov blanket *of X in $\mathcal{H}$, denoted $\mathrm{MB}_{\mathcal{H}}(X)$, to be the neighbors of X in $\mathcal{H}$. We define the* local independencies *associated with $\mathcal{H}$ to be:*

Markov blanket

local independencies

$$\mathcal{I}_\ell(\mathcal{H}) = \{(X \perp \mathcal{X} - \{X\} - \mathrm{MB}_{\mathcal{H}}(X) \mid \mathrm{MB}_{\mathcal{H}}(X)) : X \in \mathcal{X}\}.$$ ■

In other words, the local independencies state that X is independent of the rest of the nodes in the graph given its immediate neighbors. We will show that these local independence assumptions hold for any distribution that factorizes over $\mathcal{H}$, so that X's Markov blanket in $\mathcal{H}$ truly does separate it from all other variables.

4.3.2.2 Relationships between Markov Properties

We have now presented three sets of independence assertions associated with a network structure $\mathcal{H}$. For general distributions, $\mathcal{I}_p(\mathcal{H})$ is strictly weaker than $\mathcal{I}_\ell(\mathcal{H})$, which in turn is strictly weaker than $\mathcal{I}(\mathcal{H})$. However, all three definitions are equivalent for positive distributions.

Proposition 4.3 *For any Markov network $\mathcal{H}$, and any distribution P, we have that if $P \models \mathcal{I}_\ell(\mathcal{H})$ then $P \models \mathcal{I}_p(\mathcal{H})$.*

The proof of this result is left as an exercise (exercise 4.8).

Proposition 4.4 *For any Markov network $\mathcal{H}$, and any distribution P, we have that if $P \models \mathcal{I}(\mathcal{H})$ then $P \models \mathcal{I}_\ell(\mathcal{H})$.*

The proof of this result follows directly from the fact that if X and Y are not connected by an edge, then they are necessarily separated by all of the remaining nodes in the graph.

The converse of these inclusion results holds only for positive distributions (see definition 2.5). More specifically, if we assume the intersection property (equation (2.11)), all three of the Markov conditions are equivalent.

Theorem 4.4 *Let P be a positive distribution. If P satisfies $\mathcal{I}_p(\mathcal{H})$, then P satisfies $\mathcal{I}(\mathcal{H})$.*

PROOF We want to prove that for all disjoint sets $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$:

$$sep_{\mathcal{H}}(\boldsymbol{X}; \boldsymbol{Y} \mid \boldsymbol{Z}) \Longrightarrow P \models (\boldsymbol{X} \perp \boldsymbol{Y} \mid \boldsymbol{Z}). \tag{4.1}$$

The proof proceeds by descending induction on the size of $\boldsymbol{Z}$.

The base case is $|\boldsymbol{Z}| = n - 2$; equation (4.1) follows immediately from the definition of $\mathcal{I}_p(\mathcal{H})$.

For the inductive step, assume that equation (4.1) holds for every $\boldsymbol{Z}'$ with size $|\boldsymbol{Z}'| = k$, and let $\boldsymbol{Z}$ be any set such that $|\boldsymbol{Z}| = k - 1$. We distinguish between two cases.

In the first case, $\boldsymbol{X} \cup \boldsymbol{Z} \cup \boldsymbol{Y} = \mathcal{X}$. As $|\boldsymbol{Z}| < n - 2$, we have that either $|\boldsymbol{X}| \geq 2$ or $|\boldsymbol{Y}| \geq 2$. Without loss of generality, assume that the latter holds; let $A \in \boldsymbol{Y}$ and $\boldsymbol{Y}' = \boldsymbol{Y} - \{A\}$. From the fact that $sep_{\mathcal{H}}(\boldsymbol{X}; \boldsymbol{Y} \mid \boldsymbol{Z})$, we also have that $sep_{\mathcal{H}}(\boldsymbol{X}; \boldsymbol{Y}' \mid \boldsymbol{Z})$ on one hand and $sep_{\mathcal{H}}(\boldsymbol{X}; A \mid \boldsymbol{Z})$ on the other hand. As separation is monotonic, we also have that $sep_{\mathcal{H}}(\boldsymbol{X}; \boldsymbol{Y}' \mid \boldsymbol{Z} \cup \{A\})$ and $sep_{\mathcal{H}}(\boldsymbol{X}; A \mid \boldsymbol{Z} \cup \boldsymbol{Y}')$. The separating sets $\boldsymbol{Z} \cup \{A\}$ and $\boldsymbol{Z} \cup \boldsymbol{Y}'$ are each at least size $|\boldsymbol{Z}| + 1 = k$ in size, so that equation (4.1) applies, and we can conclude that P satisfies:

$$(\boldsymbol{X} \perp \boldsymbol{Y}' \mid \boldsymbol{Z} \cup \{A\}) \quad \& \quad (\boldsymbol{X} \perp A \mid \boldsymbol{Z} \cup \boldsymbol{Y}').$$

Because P is positive, we can apply the intersection property (equation (2.11)) and conclude that $P \models (\boldsymbol{X} \perp \boldsymbol{Y}' \cup \{A\} \mid \boldsymbol{Z})$, that is, $(\boldsymbol{X} \perp \boldsymbol{Y} \mid \boldsymbol{Z})$.

The second case is where $\boldsymbol{X} \cup \boldsymbol{Y} \cup \boldsymbol{Z} \neq \mathcal{X}$. Here, we might have that both $\boldsymbol{X}$ and $\boldsymbol{Y}$ are singletons. This case requires a similar argument that uses the induction hypothesis and properties of independence. We leave it as an exercise (exercise 4.9). ∎

Our previous results entail that, for positive distributions, the three conditions are equivalent.

Corollary 4.1 *The following three statements are equivalent for a positive distribution P:*

1. $P \models \mathcal{I}_\ell(\mathcal{H})$.
2. $P \models \mathcal{I}_p(\mathcal{H})$.
3. $P \models \mathcal{I}(\mathcal{H})$.

This equivalence relies on the positivity assumption. In particular, for nonpositive distributions, we can provide examples of a distribution P that satisfies one of these properties, but not the stronger one.

Example 4.5

Let P be any distribution over $\mathcal{X} = \{X_1, \ldots, X_n\}$; let $\mathcal{X} = \{X'_1, \ldots, X'_n\}$. We now construct a distribution $P'(\mathcal{X}, \mathcal{X}')$ whose marginal over $X_1, \ldots, X_n$ is the same as P, and where X'_i is deterministically equal to X_i. Let $\mathcal{H}$ be a Markov network over $\mathcal{X}, \mathcal{X}'$ that contains no edges other than X_i—X'_i. Then, in P', X_i is independent of the rest of the variables in the network given its neighbor X'_i, and similarly for X'_i; thus, $\mathcal{H}$ satisfies the local independencies for every node in the network. Yet clearly $\mathcal{H}$ is not an I-map for P', since $\mathcal{H}$ makes many independence assertions regarding the X_i's that do not hold in P (or in P'). ■

Thus, for nonpositive distributions, the local independencies do not imply the global ones.

A similar construction can be used to show that, for nonpositive distributions, the pairwise independencies do necessarily imply the local independencies.

Example 4.6

Let P be any distribution over $\mathcal{X} = \{X_1, \ldots, X_n\}$, and now consider two auxiliary sets of variables $\mathcal{X}'$ and $\mathcal{X}''$, and define $\mathcal{X}^* = \mathcal{X} \cup \mathcal{X}' \cup \mathcal{X}''$. We now construct a distribution $P'(\mathcal{X}^*)$ whose marginal over $X_1, \ldots, X_n$ is the same as P, and where X'_i and X''_i are both deterministically equal to X_i. Let $\mathcal{H}$ be the empty Markov network over $\mathcal{X}^*$. We argue that this empty network satisfies the pairwise assumptions for every pair of nodes in the network. For example, X_i and X'_i are rendered independent because $\mathcal{X}^* - \{X_i, X'_i\}$ contains X''_i. Similarly, X_i and X_j are independent given X'_i. Thus, $\mathcal{H}$ satisfies the pairwise independencies, but not the local or global independencies. ■

4.3.3 From Distributions to Graphs

Based on our deeper understanding of the independence properties associated with a Markov network, we can now turn to the question of encoding the independencies in a given distribution P using a graph structure. As for Bayesian networks, the notion of an I-map is not sufficient by itself: The complete graph implies no independence assumptions and is hence an I-map for any distribution. We therefore return to the notion of a minimal I-map, defined in definition 3.13, which was defined broadly enough to apply to Markov networks as well.

How can we construct a minimal I-map for a distribution P? Our discussion in section 4.3.2 immediately suggests two approaches for constructing a minimal I-map: one based on the pairwise Markov independencies, and the other based on the local independencies.

In the first approach, we consider the pairwise independencies. They assert that, if the edge $\{X, Y\}$ is not in $\mathcal{H}$, then X and Y must be independent given all other nodes in the graph, regardless of which other edges the graph contains. Thus, at the very least, to guarantee that $\mathcal{H}$ is an I-map, we must add direct edges between all pairs of nodes X and Y such that

$$P \not\models (X \perp Y \mid \mathcal{X} - \{X, Y\}). \tag{4.2}$$

We can now define $\mathcal{H}$ to include an edge X—Y for all X, Y for which equation (4.2) holds.

In the second approach, we use the local independencies and the notion of minimality. For each variable X, we define the neighbors of X to be a minimal set of nodes $\mathbf{Y}$ that render X independent of the rest of the nodes. More precisely, define:

Definition 4.12 Markov blanket

A set $\boldsymbol{U}$ is a Markov blanket *of X in a distribution P if $X \notin \boldsymbol{U}$ and if $\boldsymbol{U}$ is a minimal set of nodes such that*

$$(X \perp \mathcal{X} - \{X\} - \boldsymbol{U} \mid \boldsymbol{U}) \in \mathcal{I}(P). \tag{4.3}$$

■

We then define a graph $\mathcal{H}$ by introducing an edge $\{X, Y\}$ for all X and all $Y \in \mathrm{MB}_P(X)$. As defined, this construction is not unique, since there may be several sets $\boldsymbol{U}$ satisfying equation (4.3). However, theorem 4.6 will show that there is only one such minimal set. In fact, we now show that any positive distribution P has a unique minimal I-map, and that both of these constructions produce this I-map.

We begin with the proof for the pairwise definition:

Theorem 4.5

Let P be a positive distribution, and let $\mathcal{H}$ be defined by introducing an edge $\{X, Y\}$ for all X, Y for which equation (4.2) holds. Then the Markov network $\mathcal{H}$ is the unique minimal I-map for P.

PROOF The fact that $\mathcal{H}$ is an I-map for P follows immediately from fact that P, by construction, satisfies $\mathcal{I}_p(\mathcal{H})$, and, therefore, by corollary 4.1, also satisfies $\mathcal{I}(\mathcal{H})$. The fact that it is minimal follows from the fact that if we eliminate some edge $\{X, Y\}$ from $\mathcal{H}$, the graph would imply the pairwise independence $(X \perp Y \mid \mathcal{X} - \{X, Y\})$, which we know to be false for P (otherwise, the edge would have been omitted in the construction of $\mathcal{H}$). The uniqueness of the minimal I-map also follows trivially: By the same argument, any other I-map $\mathcal{H}'$ for P must contain at least the edges in $\mathcal{H}$ and is therefore either equal to $\mathcal{H}$ or contains additional edges and is therefore not minimal. ■

It remains to show that the second definition results in the same minimal I-map.

Theorem 4.6

Let P be a positive distribution. For each node X, let $\mathrm{MB}_P(X)$ be a minimal set of nodes $\boldsymbol{U}$ satisfying equation (4.3). We define a graph $\mathcal{H}$ by introducing an edge $\{X, Y\}$ for all X and all $Y \in \mathrm{MB}_P(X)$. Then the Markov network $\mathcal{H}$ is the unique minimal I-map for P.

The proof is left as an exercise (exercise 4.11).

Both of the techniques for constructing a minimal I-map make the assumption that the distribution P is positive. As we have shown, for nonpositive distributions, neither the pairwise independencies nor the local independencies imply the global one. Hence, for a nonpositive distribution P, constructing a graph $\mathcal{H}$ such that P satisfies the pairwise assumptions for $\mathcal{H}$ does not guarantee that $\mathcal{H}$ is an I-map for P. Indeed, we can easily demonstrate that both of these constructions break down for nonpositive distributions.

Example 4.7

Consider a nonpositive distribution P over four binary variables A, B, C, D that assigns nonzero probability only to cases where all four variables take on exactly the same value; for example, we might have $P(a^1, b^1, c^1, d^1) = 0.5$ and $P(a^0, b^0, c^0, d^0) = 0.5$. The graph $\mathcal{H}$ shown in figure 4.7 is one possible output of applying the local independence I-map construction algorithm to P: For example, $P \models (A \perp C, D \mid B)$, and hence $\{B\}$ is a legal choice for $\mathrm{MB}_P(A)$. A similar analysis shows that this network satisfies the Markov blanket condition for all nodes. However, it is not an I-map for the distribution.

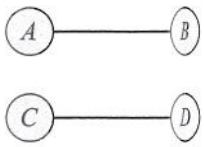


Figure 4.7 An attempt at an I-map for a nonpositive distribution P

If we use the pairwise independence I-map construction algorithm for this distribution, the network constructed is the empty network. For example, the algorithm would not place an edge between A and B, because $P \models (A \perp B \mid C, D)$. Exactly the same analysis shows that no edges will be placed into the graph. However, the resulting network is not an I-map for P. ■

Both these examples show that deterministic relations between variables can lead to failure in the construction based on local and pairwise independence. Suppose that A and B are two variables that are identical to each other and that both C and D are variables that correlated to both A and B so that $(C \perp D \mid A, B)$ holds. Since A is identical to B, we have that both $(A, D \perp C \mid B)$ and $(B, D \perp C \mid A)$ hold. In other words, it suffices to observe one of these two variables to capture the relevant information both have about C and separate C from D. In this case the Markov blanket of C is not uniquely defined. This ambiguity leads to the failure of both local and pairwise constructions. Clearly, identical variables are only one way of getting such ambiguities in local independencies. Once we allow nonpositive distribution, other distributions can have similar problems.

Having defined the notion of a minimal I-map for a distribution P, we can now ask to what extent it represents the independencies in P. More formally, we can ask whether every distribution has a perfect map. Clearly, the answer is no, even for positive distributions:

Example 4.8

Consider a distribution arising from a three-node Bayesian network with a v-structure, for example, the distribution induced in the Student *example over the nodes Intelligence, Difficulty, and Grade (figure 3.3). In the Markov network for this distribution, we must clearly have an edge between I and G and between D and G. Can we omit the edge between I and D? No, because we do not have that $(I \perp D \mid G)$ holds for the distribution; rather, we have the opposite: I and D are* dependent *given G. Therefore, the only minimal I-map for this P is the fully connected graph, which does not capture the marginal independence $(I \perp D)$ that holds in P.* ■

This example provides another counterexample to the strong version of completeness mentioned earlier. The only distributions for which separation is a sound and complete criterion for determining conditional independence are those for which $\mathcal{H}$ is a perfect map.

4.4 Parameterization Revisited

Now that we understand the semantics and independence properties of Markov networks, we revisit some alternative representations for the parameterization of a Markov network.

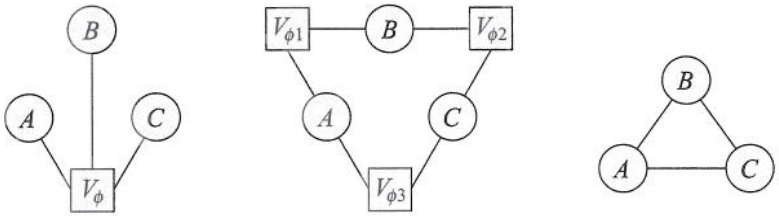


Figure 4.8 Different factor graphs for the same Markov network: (a) One factor graph over A, B, C, with a single factor over all three variables. (b) An alternative factor graph, with three pairwise factors. (c) The induced Markov network for both is a clique over A, B, C.

4.4.1 Finer-Grained Parameterization

4.4.1.1 Factor Graphs

A Markov network structure does not generally reveal all of the structure in a Gibbs parameterization. In particular, one cannot tell from the graph structure whether the factors in the parameterization involve maximal cliques or subsets thereof. Consider, for example, a Gibbs distribution P over a fully connected pairwise Markov network; that is, P is parameterized by a factor for each pair of variables $X, Y \in \mathcal{X}$. The clique potential parameterization would utilize a factor whose scope is the entire graph, and which therefore uses an exponential number of parameters. On the other hand, as we discussed in section 4.2.1, the number of parameters in the pairwise parameterization is quadratic in the number of variables. Note that the complete Markov network is not redundant in terms of conditional independencies — P does not factorize over any smaller network. Thus, although the finer-grained structure does not imply additional independencies in the distribution (see exercise 4.6), it is still very significant.

An alternative representation that makes this structure explicit is a *factor graph*. A factor graph is a graph containing two types of nodes: one type corresponds, as usual, to random variables; the other corresponds to factors over the variables. Formally:

Definition 4.13 (factor graph, factorization)

A factor graph $\mathcal{F}$ *is an undirected graph containing two types of nodes:* variable nodes *(denoted as ovals) and* factor nodes *(denoted as squares). The graph only contains edges between variable nodes and factor nodes. A factor graph $\mathcal{F}$ is parameterized by a set of factors, where each factor node V_ϕ is associated with precisely one factor ϕ, whose scope is the set of variables that are neighbors of V_ϕ in the graph. A distribution P* factorizes *over $\mathcal{F}$ if it can be represented as a set of factors of this form.* ■

Factor graphs make explicit the structure of the factors in the network. For example, in a fully connected pairwise Markov network, the factor graph would contain a factor node for each of the $\binom{n}{2}$ pairs of nodes; the factor node for a pair X_i, X_j would be connected to X_i and X_j; by contrast, a factor graph for a distribution with a single factor over $X_1, \ldots, X_n$ would have a single factor node connected to all of $X_1, \ldots, X_n$ (see figure 4.8). Thus, although the Markov networks for these two distributions are identical, their factor graphs make explicit the

$\epsilon_1(A,B)$			$\epsilon_2(B,C)$			$\epsilon_3(C,D)$			$\epsilon_4(D,A)$		
a^0	b^0	-3.4	b^0	c^0	-4.61	c^0	d^0	0	d^0	a^0	-4.61
a^0	b^1	-1.61	b^0	c^1	0	c^0	d^1	-4.61	d^0	a^1	0
a^1	b^0	0	b^1	c^0	0	c^1	d^0	-4.61	d^1	a^0	0
a^1	b^1	-2.3	b^1	c^1	-4.61	c^1	d^1	0	d^1	a^1	-4.61
(a)			(b)			(c)			(d)		

Figure 4.9 Energy functions for the Misconception **example**

difference in their factorization.

4.4.1.2 Log-Linear Models

Although factor graphs make certain types of structure more explicit, they still encode factors as complete tables over the scope of the factor. As in Bayesian networks, factors can also exhibit a type of context-specific structure — patterns that involve particular values of the variables. These patterns are often more easily seen in terms of an alternative parameterization of the factors that converts them into log-space.

More precisely, we can rewrite a factor $\phi(\boldsymbol{D})$ as

$$\phi(\boldsymbol{D}) = \exp(-\epsilon(\boldsymbol{D})),$$

energy function

where $\epsilon(\boldsymbol{D}) = -\ln \phi(\boldsymbol{D})$ is often called an *energy function.* The use of the word "energy" derives from statistical physics, where the probability of a physical state (for example, a configuration of a set of electrons), depends inversely on its energy. In this logarithmic representation, we have

$$P(X_1, \ldots, X_n) \propto \exp\left[-\sum_{i=1}^{m} \epsilon_i(\boldsymbol{D}_i)\right].$$

The logarithmic representation ensures that the probability distribution is positive. Moreover, the logarithmic parameters can take any value along the real line.

Any Markov network parameterized using positive factors can be converted to a logarithmic representation.

Example 4.9 *Figure 4.9 shows the logarithmic representation of the clique potential parameters in figure 4.1. We can see that the "1" entries in the clique potentials translate into "0" entries in the energy function.*■

This representation makes certain types of structure in the potentials more apparent. For example, we can see that both $\epsilon_2(B,C)$ and $\epsilon_4(D,A)$ are constant multiples of an energy function that ascribes 1 to instantiations where the values of the two variables agree, and 0 to the instantiations where they do not.

We can provide a general framework for capturing such structure using the following notion:

Definition 4.14 *Let $\boldsymbol{D}$ be a subset of variables. We define a* feature *$f(\boldsymbol{D})$ to be a function from $\boldsymbol{D}$ to $\mathbb{R}$.* ■

feature

indicator feature

A feature is simply a factor without the nonnegativity requirement. One type of feature of particular interest is the *indicator feature* that takes on value 1 for some values $\boldsymbol{y} \in Val(\boldsymbol{D})$ and 0 otherwise.

Features provide us with an easy mechanism for specifying certain types of interactions more compactly.

Example 4.10 *Consider a situation where A_1 and A_2 each have ℓ values $a^1, \ldots, a^\ell$. Assume that our distribution is such that we prefer situations where A_1 and A_2 take on the same value, but otherwise have no preference. Thus, our energy function might have the following form:*

$$\epsilon(A_1, A_2) = \begin{cases} 3 & A_1 = A_2 \\ 0 & \text{otherwise} \end{cases}$$

Represented as a full factor, this clique potential requires ℓ^2 values. However, it can also be represented as a log-linear function in terms of a feature $f(A_1, A_2)$ that is an indicator function for the event $A_1 = A_2$. The energy function is then simply a constant multiple 3 of this feature. ■

Thus, we can provide a more general definition for our notion of log-linear models:

Definition 4.15 *A distribution P is a* log-linear model *over a Markov network $\mathcal{H}$ if it is associated with:*

log-linear model

- *a set of features $\mathcal{F} = \{f_1(\boldsymbol{D}_1), \ldots, f_k(\boldsymbol{D}_k)\}$, where each $\boldsymbol{D}_i$ is a complete subgraph in $\mathcal{H}$,*
- *a set of weights $w_1, \ldots, w_k$,*

such that

$$P(X_1, \ldots, X_n) = \frac{1}{Z} \exp\left[-\sum_{i=1}^{k} w_i f_i(\boldsymbol{D}_i)\right].$$

■

Note that we can have several features over the same scope, so that we can, in fact, represent a standard set of table potentials. (See exercise 4.13.)

The log-linear model provides a much more compact representation for many distributions, especially in situations where variables have large domains such as text (such as box 4.E).

4.4.1.3 Discussion

We now have three representations of the parameterization of a Markov network. The Markov network denotes a product over potentials on cliques. A factor graph denotes a product of factors. And a set of features denotes a product over feature weights. Clearly, each representation is finer-grained than the previous one and as rich. A factor graph can describe the Gibbs distribution, and a set of features can describe all the entries in each of the factors of a factor graph.

Depending on the question of interest, different representations may be more appropriate. For example, a Markov network provides the right level of abstraction for discussing independence queries: The finer-grained representations of factor graphs or log-linear

models do not change the independence assertions made by the model. On the other hand, as we will see in later chapters, factor graphs are useful when we discuss inference, and features are useful when we discuss parameterizations, both for hand-coded models and for learning.

Ising model

Box 4.C — Concept: Ising Models and Boltzmann Machines. *One of the earliest types of Markov network models is the* Ising model, *which first arose in statistical physics as a model for the energy of a physical system involving a system of interacting atoms. In these systems, each atom is associated with a binary-valued random variable $X_i \in \{+1, -1\}$, whose value defines the direction of the atom's spin. The energy function associated with the edges is defined by a particularly simple parametric form:*

$$\epsilon_{i,j}(x_i, x_j) = -w_{i,j} x_i x_j \tag{4.4}$$

This energy is symmetric in X_i, X_j; it makes a contribution of $w_{i,j}$ to the energy function when $X_i = X_j$ (so both atoms have the same spin) and a contribution of $-w_{i,j}$ otherwise. Our model also contains a set of parameters u_i that encode individual node potentials; these bias individual variables to have one spin or another.

As usual, the energy function defines the following distribution:

$$P(\xi) = \frac{1}{Z} \exp\left(-\sum_{i<j} w_{i,j} x_i x_j - \sum_i u_i x_i \right).$$

As we can see, when $w_{i,j} > 0$ the model prefers to align the spins of the two atoms; in this case, the interaction is called ferromagnetic. *When $w_{i,j} < 0$ the interaction is called* antiferromagnetic. *When $w_{i,j} = 0$ the atoms are* non-interacting.

Much work has gone into studying particular types of Ising models, attempting to answer a variety of questions, usually as the number of atoms goes to infinity. For example, we might ask the probability of a configuration in which a majority of the spins are $+1$ or -1, versus the probability of more mixed configurations. The answer to this question depends heavily on the strength of the interaction between the variables; so, we can consider adapting this strength (by multiplying all weights by a temperature parameter*) and asking whether this change causes a phase transition in the probability of skewed versus mixed configurations. These questions, and many others, have been investigated extensively by physicists, and the answers are known (in some cases even analytically) for several cases.*

temperature parameter

Boltzmann distribution

Related to the Ising model is the Boltzmann distribution*; here, the variables are usually taken to have values $\{0, 1\}$, but still with the energy form of equation (4.4). Here, we get a nonzero contribution to the model from an edge (X_i, X_j) only when $X_i = X_j = 1$; however, the resulting energy can still be reformulated in terms of an Ising model (exercise 4.12).*

The popularity of the Boltzmann machine was primarily driven by its similarity to an activation model for neurons. To understand the relationship, we note that the probability distribution over each variable X_i given an assignment to its neighbors is $\text{sigmoid}(z)$ *where*

$$z = -\left(\sum_j w_{i,j} x_j\right) - w_i.$$

This function is a sigmoid of a weighted combination of X_i's neighbors, weighted by the strength and direction of the connections between them. This is the simplest but also most popular mathematical approximation of the function employed by a neuron in the brain. Thus, if we imagine a process by which the network continuously adapts its assignment by resampling the value of each variable as a stochastic function of its neighbors, then the "activation" probability of each variable resembles a neuron's activity. This model is a very simple variant of a stochastic, recurrent neural network.

---

labeling MRF

Box 4.D — Concept: Metric MRFs. *One important class of MRFs comprises those used for* labeling. *Here, we have a graph of nodes $X_1, \ldots, X_n$ related by a set of edges $\mathcal{E}$, and we wish to assign to each X_i a in the space $\mathcal{V} = \{v_1, \ldots, v_K\}$. Each node, taken in isolation, has its preferences among the possible labels. However, we also want to impose a soft "smoothness" constraint over the graph, in that neighboring nodes should take "similar" values.*

We encode the individual node preferences as node potentials in a pairwise MRF and the smoothness preferences as edge potentials. For reasons that will become clear, it is traditional to encode these models in negative log-space, using energy functions. As our objective in these models is inevitably the MAP objective, we can also ignore the partition function, and simply consider the energy function:

$$E(x_1, \ldots, x_n) = \sum_i \epsilon_i(x_i) + \sum_{(i,j)\in\mathcal{E}} \epsilon_{i,j}(x_i, x_j). \tag{4.5}$$

Our goal is then to minimize the energy:

$$\arg \min_{x_1,\ldots,x_n} E(x_1, \ldots, x_n).$$

We now need to provide a formal definition for the intuition of "smoothness" described earlier. There are many different types of conditions that we can impose; different conditions allow different methods to be applied.

Ising model

One of the simplest in this class of models is a slight variant of the Ising model, *where we have that, for any i, j:*

$$\epsilon_{i,j}(x_i, x_j) = \begin{cases} 0 & x_i = x_j \\ \lambda_{i,j} & x_i \neq x_j, \end{cases} \tag{4.6}$$

for $\lambda_{i,j} \geq 0$. In this model, we obtain the lowest possible pairwise energy (0) when two neighboring nodes X_i, X_j take the same value, and a higher energy $\lambda_{i,j}$ when they do not.

Potts model

This simple model has been generalized in many ways. The Potts model *extends it to the setting of more than two labels. An even broader class contains models where we have a distance function on the labels, and where we prefer neighboring nodes to have labels that are a smaller distance*

metric function

apart. More precisely, a function $\mu : \mathcal{V} \times \mathcal{V} \mapsto [0, \infty)$ is a metric *if it satisfies:*

- **Reflexivity:** $\mu(v_k, v_l) = 0$ *if and only if* $k = l$;
- **Symmetry:** $\mu(v_k, v_l) = \mu(v_l, v_k)$;

$\epsilon_1'(A,B)$			$\epsilon_2'(B,C)$		
a^0	b^0	-4.4	b^0	c^0	-3.61
a^0	b^1	1.61	b^0	c^1	$+1$
a^1	b^0	-1	b^1	c^0	0
a^1	b^1	2.3	b^1	c^1	4.61

(a) (b)

Figure 4.10 Alternative but equivalent energy functions

- **Triangle Inequality:** $\mu(v_k, v_l) + \mu(v_l, v_m) \geq \mu(v_k, v_m)$.

semimetric

We say that μ is a semimetric *if it satisfies reflexivity and symmetry. We can now define a* metric MRF *(or a* semimetric MRF*) by defining $\epsilon_{i,j}(v_k, v_l) = \mu(v_k, v_l)$ for all i, j, where μ is a metric (semimetric). We note that, as defined, this model assumes that the distance metric used is the same for all pairs of variables. This assumption is made because it simplifies notation, it often holds in practice, and it reduces the number of parameters that must be acquired. It is not required for the inference algorithms that we present in later chapters. Metric interactions arise in many applications, and play a particularly important role in computer vision (see box 4.B and box 13.B).*

truncated norm

For example, one common metric used is some form of truncated p-norm *(usually $p = 1$ or $p = 2$):*

$$\epsilon(x_i, x_j) = \min(c\|x_i - x_j\|_p, \text{dist}_{\max}). \tag{4.7}$$

4.4.2 Overparameterization

Even if we use finer-grained factors, and in some cases, even features, the Markov network parameterization is generally overparameterized. That is, for any given distribution, there are multiple choices of parameters to describe it in the model. Most obviously, if our graph is a single clique over n binary variables $X_1, \ldots, X_n$, then the network is associated with a clique potential that has 2^n parameters, whereas the joint distribution only has $2^n - 1$ independent parameters.

A more subtle point arises in the context of a nontrivial clique structure. Consider a pair of cliques $\{A, B\}$ and $\{B, C\}$. The energy function $\epsilon_1(A, B)$ (or its corresponding clique potential) contains information not only about the interaction between A and B, but also about the distribution of the individual variables A and B. Similarly, $\epsilon_2(B, C)$ gives us information about the individual variables B and C. The information about B can be placed in either of the two cliques, or its contribution can be split between them in arbitrary ways, resulting in many different ways of specifying the same distribution.

Example 4.11

Consider the energy functions $\epsilon_1(A, B)$ and $\epsilon_2(B, C)$ in figure 4.9. The pair of energy functions shown in figure 4.10 result in an equivalent distribution: Here, we have simply subtracted 1 from $\epsilon_1(A, B)$ and added 1 to $\epsilon_2(B, C)$ for all instantiations where $B = b^0$. It is straightforward to

check that this results in an identical distribution to that of figure 4.9. In instances where $B \neq b^0$ the energy function returns exactly the same value as before. In cases where $B = b^0$, the actual values of the energy functions have changed. However, because the sum of the energy functions on each instance is identical to the original sum, the probability of the instance will not change. ■

Intuitively, the standard Markov network representation gives us too many places to account for the influence of variables in shared cliques. Thus, the same distribution can be represented as a Markov network (of a given structure) in infinitely many ways. It is often useful to pick one of this infinite set as our chosen parameterization for the distribution.

4.4.2.1 Canonical Parameterization

canonical parameterization

The *canonical parameterization* provides one very natural approach to avoiding this ambiguity in the parameterization of a Gibbs distribution P. This canonical parameterization requires that the distribution P be positive. It is most convenient to describe this parameterization using energy functions rather then clique potentials. For this reason, it is also useful to consider a log-transform of P: For any assignment ξ to $\mathcal{X}$, we use $\ell(\xi)$ to denote $\ln P(\xi)$. This transformation is well defined because of our positivity assumption.

The canonical parameterization of a Gibbs distribution over $\mathcal{H}$ is defined via a set of energy functions over all non-empty cliques. Thus, for example, the Markov network in figure 4.4b would have energy functions for the two cliques $\{A, B, D\}$ and $\{B, C, D\}$, energy functions for all possible pairs of variables except the pair $\{A, C\}$ (a total of five pairs), and energy functions for all four singleton sets.

At first glance, it appears that we have only increased the number of parameters in the specification. However, as we will see, this approach uniquely associates the interaction parameters for a subset of variables with that subset, avoiding the ambiguity described earlier. As a consequence, many of the parameters in this canonical parameterization are often zero.

The canonical parameterization is defined relative to a particular fixed assignment $\xi^* = (x_1^*, \ldots, x_n^*)$ to the network variables $\mathcal{X}$. This assignment can be chosen arbitrarily. For any subset of variables $\boldsymbol{Z}$, and any assignment $\boldsymbol{x}$ to some subset of $\mathcal{X}$ that contains $\boldsymbol{Z}$, we define the assignment $\boldsymbol{x}_{\boldsymbol{Z}}$ to be $\boldsymbol{x}\langle \boldsymbol{Z} \rangle$, that is, the assignment in $\boldsymbol{x}$ to the variables in $\boldsymbol{Z}$. Conversely, we define $\xi^*_{-\boldsymbol{Z}}$ to be $\xi^*\langle \mathcal{X} - \boldsymbol{Z} \rangle$, that is, the assignment in ξ^* to the variables outside $\boldsymbol{Z}$. We can now construct an assignment $(\boldsymbol{x}_{\boldsymbol{Z}}, \xi^*_{-\boldsymbol{Z}})$ that keeps the assignments to the variables in $\boldsymbol{Z}$ as specified in $\boldsymbol{x}$, and augments it using the default values in ξ^*.

canonical energy function

The *canonical energy function* for a clique $\boldsymbol{D}$ is now defined as follows:

$$\epsilon^*_{\boldsymbol{D}}(\boldsymbol{d}) = \sum_{\boldsymbol{Z} \subseteq \boldsymbol{D}} (-1)^{|\boldsymbol{D}-\boldsymbol{Z}|} \ell(\boldsymbol{d}_{\boldsymbol{Z}}, \xi^*_{-\boldsymbol{Z}}), \tag{4.8}$$

where the sum is over all subsets of $\boldsymbol{D}$, including $\boldsymbol{D}$ itself and the empty set $\emptyset$. Note that all of the terms in the summation have a scope that is contained in $\boldsymbol{D}$, which in turn is part of a clique, so that these energy functions are legal relative to our Markov network structure.

This formula performs an inclusion-exclusion computation. For a set $\{A, B, C\}$, it first subtracts out the influence of all of the pairs: $\{A, B\}$, $\{B, C\}$, and $\{C, A\}$. However, this process oversubtracts the influence of the individual variables. Thus, their influence is added back in, to compensate. More generally, consider any subset of variables $\boldsymbol{Z} \subseteq \boldsymbol{D}$. Intuitively, it

$\epsilon_1^*(A,B)$			$\epsilon_2^*(B,C)$			$\epsilon_3^*(C,D)$			$\epsilon_4^*(D,A)$		
a^0	b^0	0	b^0	c^0	0	c^0	d^0	0	d^0	a^0	0
a^0	b^1	0	b^0	c^1	0	c^0	d^1	9.21	d^0	a^1	9.21
a^1	b^0	0	b^1	c^0	0	c^1	d^0	0	d^1	a^0	0
a^1	b^1	4.09	b^1	c^1	9.21	c^1	d^1	0	d^1	a^1	18.4

$\epsilon_5^*(A)$		$\epsilon_6^*(B)$		$\epsilon_7^*(C)$		$\epsilon_8^*(D)$		$\epsilon_9^*(\emptyset)$
a^0	0	b^0	0	c^0	0	d^0	0	-3.18
a^1	-8.01	b^1	-6.4	c^1	0	d^1	-9.21	

Figure 4.11 Canonical energy function for the Misconception **example**

makes a "contribution" once for every subset $\boldsymbol{U} \supseteq \boldsymbol{Z}$. Except for $\boldsymbol{U} = \boldsymbol{D}$, the number of times that $\boldsymbol{Z}$ appears is even — there is an even number of subsets $\boldsymbol{U} \supseteq \boldsymbol{Z}$ — and the number of times it appears with a positive sign is equal to the number of times it appears with a negative sign. Thus, we have effectively eliminated the net contribution of the subsets from the canonical energy function.

Let us consider the effect of the canonical transformation on our Misconception network.

Example 4.12

Let us choose (a^0, b^0, c^0, d^0) as our arbitrary assignment on which to base the canonical parameterization. The resulting energy functions are shown in figure 4.11. For example, the energy value $\epsilon_1^*(a^1, b^1)$ was computed as follows:

$$\ell(a^1, b^1, c^0, d^0) - \ell(a^1, b^0, c^0, d^0) - \ell(a^0, b^1, c^0, d^0) + \ell(a^0, b^0, c^0, d^0) = \\ -13.49 - -11.18 - -9.58 + -3.18 = 4.09$$

Note that many of the entries in the energy functions are zero. As discussed earlier, this phenomenon is fairly general, and occurs because we have accounted for the influence of small subsets of variables separately, leaving the larger factors to deal only with higher-order influences. We also note that these canonical parameters are not very intuitive, highlighting yet again the difficulties of constructing a reasonable parameterization of a Markov network by hand. ■

This canonical parameterization defines the same distribution as our original distribution P:

Theorem 4.7

Let P be a positive Gibbs distribution over $\mathcal{H}$, and let $\epsilon^*(\boldsymbol{D}_i)$ for each clique $\boldsymbol{D}_i$ be defined as specified in equation (4.8). Then

$$P(\xi) = \exp\left[-\sum_i \epsilon^*_{\boldsymbol{D}_i}(\xi\langle \boldsymbol{D}_i\rangle)\right].$$

The proof for the case where $\mathcal{H}$ consists of a single clique is fairly simple, and it is left as an exercise (exercise 4.4). The general case follows from results in the next section.

The canonical parameterization gives us the tools to prove the Hammersley-Clifford theorem, which we restate for convenience.

Theorem 4.8 *Let P be a positive distribution over $\mathcal{X}$, and $\mathcal{H}$ a Markov network graph over $\mathcal{X}$. If $\mathcal{H}$ is an I-map for P, then P is a Gibbs distribution over $\mathcal{H}$.*

Proof To prove this result, we need to show the existence of a Gibbs parameterization for any distribution P that satisfies the Markov assumptions associated with $\mathcal{H}$. The proof is constructive, and simply uses the canonical parameterization shown in section 4.4.2. Given P, we define an energy function for all subsets $\boldsymbol{D}$ of nodes in the graph, regardless of whether they are cliques in the graph. This energy function is defined exactly as in equation (4.8), relative to some specific fixed assignment ξ^* used to define the canonical parameterization. The distribution defined using this set of energy functions is P: the argument is identical to the proof of theorem 4.7, for the case where the graph consists of a single clique (see exercise 4.4).

It remains only to show that the resulting distribution is a Gibbs distribution over $\mathcal{H}$. To show that, we need to show that the factors $\epsilon^*(\boldsymbol{D})$ are identically 0 whenever $\boldsymbol{D}$ is not a clique in the graph, that is, whenever the nodes in $\boldsymbol{D}$ do not form a fully connected subgraph. Assume that we have $X, Y \in \boldsymbol{D}$ such that there is no edge between X and Y. For this proof, it helps to introduce the notation

$$\sigma_{\boldsymbol{Z}}[\boldsymbol{x}] = (\boldsymbol{x}_{\boldsymbol{Z}}, \xi^*_{-\boldsymbol{Z}}).$$

Plugging this notation into equation (4.8), we have that:

$$\epsilon^*_{\boldsymbol{D}}(\boldsymbol{d}) = \sum_{\boldsymbol{Z} \subseteq \boldsymbol{D}} (-1)^{|\boldsymbol{D}-\boldsymbol{Z}|} \ell(\sigma_{\boldsymbol{Z}}[\boldsymbol{d}]).$$

We now rearrange the sum over subsets $\boldsymbol{Z}$ into a sum over groups of subsets. Let $\boldsymbol{W} \subseteq \boldsymbol{D} - \{X, Y\}$; then $\boldsymbol{W}$, $\boldsymbol{W} \cup \{X\}$, $\boldsymbol{W} \cup \{Y\}$, and $\boldsymbol{W} \cup \{X, Y\}$ are all subsets of $\boldsymbol{Z}$. Hence, we can rewrite the summation over subsets of $\boldsymbol{D}$ as a summation over subsets of $\boldsymbol{D} - \{X, Y\}$:

$$\begin{aligned} \epsilon^*_{\boldsymbol{D}}(\boldsymbol{d}) \quad = \quad & \sum_{\boldsymbol{W} \subseteq \boldsymbol{D}-\{X,Y\}} (-1)^{|\boldsymbol{D}-\{X,Y\}-\boldsymbol{W}|} \\ & (\ell(\sigma_{\boldsymbol{W}}[\boldsymbol{d}]) - \ell(\sigma_{\boldsymbol{W}\cup\{X\}}[\boldsymbol{d}]) - \ell(\sigma_{\boldsymbol{W}\cup\{Y\}}[\boldsymbol{d}]) + \ell(\sigma_{\boldsymbol{W}\cup\{X,Y\}}[\boldsymbol{d}])). \end{aligned} \tag{4.9}$$

Now consider a specific subset $\boldsymbol{W}$ in this sum, and let $\boldsymbol{u}^*$ be $\xi^*\langle \mathcal{X} - \boldsymbol{D} \rangle$ — the assignment to $\mathcal{X} - \boldsymbol{D}$ in ξ. We now have that:

$$\begin{aligned} \ell(\sigma_{\boldsymbol{W}\cup\{X,Y\}}[\boldsymbol{d}])) - \ell(\sigma_{\boldsymbol{W}\cup\{X\}}[\boldsymbol{d}]) &= \ln \frac{P(x, y, \boldsymbol{w}, \boldsymbol{u}^*)}{P(x, y^*, \boldsymbol{w}, \boldsymbol{u}^*)} \\ &= \ln \frac{P(y \mid x, \boldsymbol{w}, \boldsymbol{u}^*) P(x, \boldsymbol{w}, \boldsymbol{u}^*)}{P(y^* \mid x, \boldsymbol{w}, \boldsymbol{u}^*) P(x, \boldsymbol{w}, \boldsymbol{u}^*)} \\ &= \ln \frac{P(y \mid x^*, \boldsymbol{w}, \boldsymbol{u}^*) P(x, \boldsymbol{w}, \boldsymbol{u}^*)}{P(y^* \mid x^*, \boldsymbol{w}, \boldsymbol{u}^*) P(x, \boldsymbol{w}, \boldsymbol{u}^*)} \\ &= \ln \frac{P(y \mid x^*, \boldsymbol{w}, \boldsymbol{u}^*) P(x^*, \boldsymbol{w}, \boldsymbol{u}^*)}{P(y^* \mid x^*, \boldsymbol{w}, \boldsymbol{u}^*) P(x^*, \boldsymbol{w}, \boldsymbol{u}^*)} \\ &= \ln \frac{P(x^*, y, \boldsymbol{w}, \boldsymbol{u}^*)}{P(x^*, y^*, \boldsymbol{w}, \boldsymbol{u}^*)} \\ &= \ell(\sigma_{\boldsymbol{W}\cup\{Y\}}[\boldsymbol{d}])) - \ell(\sigma_{\boldsymbol{W}}[\boldsymbol{d}]), \end{aligned}$$

where the third equality is a consequence of the fact that X and Y are not connected directly by an edge, and hence we have that $P \models (X \perp Y \mid \mathcal{X} - \{X, Y\})$. Thus, we have that each term in the outside summation in equation (4.9) adds to zero, and hence the summation as a whole is also zero, as required. ■

For positive distributions, we have already shown that all three sets of Markov assumptions are equivalent; putting these results together with theorem 4.1 and theorem 4.2, we obtain that, **for positive distributions, all four conditions — factorization and the three types of Markov assumptions — are all equivalent.**

4.4.2.2 Eliminating Redundancy

An alternative approach to the issue of overparameterization is to try to eliminate it entirely. We can do so in the context of a feature-based representation, which is sufficiently fine-grained to allow us to eliminate redundancies without losing expressive power. The tools for detecting and eliminating redundancies come from linear algebra.

linear dependence

We say that a set of features $f_1, \ldots, f_k$ is *linearly dependent* if there are constants $\alpha_0, \alpha_1, \ldots, \alpha_k$, not all of which are 0, so that for all ξ

$$\alpha_0 + \sum_i \alpha_i f_i(\xi) = 0.$$

This is the usual definition of linear dependencies in linear algebra, where we view each feature as a vector whose entries are the value of the feature in each of the possible instantiations.

Example 4.13 *Consider again the* Misconception *example. We can encode the log-factors in example 4.9 as a set of features by introducing indicator features of the form:*

$$f_{a,b}(A, B) = \begin{cases} 1 & A = a, B = b \\ 0 & \text{otherwise.} \end{cases}$$

Thus, to represent $\epsilon_1(A, B)$, we introduce four features that correspond to the four entries in the energy function. Since A, B take on exactly one of these possible four values, we have that

$$f_{a^0,b^0}(A, B) + f_{a^0,b^1}(A, B) + f_{a^1,b^0}(A, B) + f_{a^1,b^1}(A, B) = 1.$$

Thus, this set of features is linearly dependent. ■

Example 4.14 *Now consider also the features that capture $\epsilon_2(B, C)$ and their interplay with the features that capture $\epsilon_1(A, B)$. We start by noting that the sum $f_{a^0,b^0}(A, B) + f_{a^1,b^0}(A, B)$ is equal to 1 when $B = b^0$ and 0 otherwise. Similarly, $f_{b^0,c^0}(B, C) + f_{b^0,c^1}(B, C)$ is also an indicator for $B = b^0$. Thus we get that*

$$f_{a^0,b^0}(A, B) + f_{a^1,b^0}(A, B) - f_{b^0,c^0}(B, C) - f_{b^0,c^1}(B, C) = 0.$$

And so these four features are linearly dependent. ■

As we now show, linear dependencies imply non-unique parameterization.

Proposition 4.5 *Let $f_1, \ldots, f_k$ be a set of features with weights $w = \{w_1, \ldots, w_k\}$ that form a log-linear representation of a distribution P. If there are coefficients $\alpha_0, \alpha_1, \ldots, \alpha_k$ such that for all ξ*

$$\alpha_0 + \sum_i \alpha_i f_i(\xi) = 0 \tag{4.10}$$

then the log-linear model with weights $w' = \{w_1 + \alpha_1, \ldots, w_k + \alpha_k\}$ also represents P.

PROOF Consider the distribution

$$P_{w'}(\xi) \propto \exp\left\{-\sum_i (w_i + \alpha_i) f_i(\xi)\right\}.$$

Using equation (4.10) we see that

$$-\sum_i (w_i + \alpha_i) f_i(\xi) = \alpha_0 - \sum_i w_i f_i(\xi).$$

Thus,

$$P_{w'}(\xi) \propto e^{\alpha_0} \exp\left\{-\sum_i w_i f_i(\xi)\right\} \propto P(\xi).$$

We conclude that $P_{w'}(\xi) = P(\xi)$. ∎

redundant

Motivated by this result, we say that a set of linearly dependent features is *redundant*. A *nonredundant* set of features is one where the features are not linearly dependent on each other. In fact, if the set of features is nonredundant, then each set of weights describes a unique distribution.

Proposition 4.6 *Let $f_1, \ldots, f_k$ be a set of nonredundant features, and let $w, w' \in \mathbb{R}^k$. If $w \neq w'$ then $P_w \neq P_{w'}$.*

Example 4.15 *Can we construct a nonredundant set of features for the* Misconception *example? We can determine the number of nonredundant features by building the 16×16 matrix of the values of the 16 features (four factors with four features each) in the 16 instances of the joint distribution. This matrix has rank of 9, which implies that a subset of 8 features will be a nonredundant subset. In fact, there are several such subsets. In particular, the canonical parameterization shown in figure 4.11 has nine features of nonzero weight, which form a nonredundant parameterization. The equivalence of the canonical parameterization (theorem 4.7) implies that this set of features has the same expressive power as the original set of features. To verify this, we can show that adding any other feature will lead to a linear dependency. Consider, for example, the feature f_{a^1,b^0}. We can verify that*

$$f_{a^1,b^0} + f_{a^1,b^1} - f_{a^1} = 0.$$

Similarly, consider the feature f_{a^0,b^0}. Again we can find a linear dependency on other features:

$$f_{a^0,b^0} + f_{a^1} + f_{b^1} - f_{a^1,b^1} = 1.$$

Using similar arguments, we can show that adding any of the original features will lead to redundancy. Thus, this set of features can represent any parameterization in the original model. ∎

4.5 Bayesian Networks and Markov Networks

We have now described two graphical representation languages: Bayesian networks and Markov networks. Example 3.8 and example 4.8 show that these two representations are incomparable as a language for representing independencies: each can represent independence constraints that the other cannot. In this section, we strive to provide more insight about the relationship between these two representations.

4.5.1 From Bayesian Networks to Markov Networks

Let us begin by examining how we might take a distribution represented using one of these frameworks, and represent it in the other. One can view this endeavor from two different perspectives: Given a Bayesian network $\mathcal{B}$, we can ask how to represent the distribution $P_{\mathcal{B}}$ as a parameterized Markov network; or, given a graph $\mathcal{G}$, we can ask how to represent the independencies in $\mathcal{G}$ using an undirected graph $\mathcal{H}$. In other words, we might be interested in finding a minimal I-map for a distribution $P_{\mathcal{B}}$, or a minimal I-map for the independencies $\mathcal{I}(\mathcal{H})$. We can see that these two questions are related, but each perspective offers its own insights.

Let us begin by considering a distribution $P_{\mathcal{B}}$, where $\mathcal{B}$ is a parameterized Bayesian network over a graph $\mathcal{G}$. Importantly, the parameterization of $\mathcal{B}$ can also be viewed as a parameterization for a Gibbs distribution: We simply take each CPD $P(X_i \mid \mathrm{Pa}_{X_i})$ and view it as a factor of scope X_i, Pa_{X_i}. This factor satisfies additional normalization properties that are not generally true of all factors, but it is still a legal factor. This set of factors defines a Gibbs distribution, one whose partition function happens to be 1.

What is more important, **a Bayesian network conditioned on evidence $\boldsymbol{E} = \boldsymbol{e}$ also induces a Gibbs distribution: the one defined by the original factors *reduced* to the context $\boldsymbol{E} = \boldsymbol{e}$.**

Proposition 4.7 *Let $\mathcal{B}$ be a Bayesian network over $\mathcal{X}$ and $\boldsymbol{E} = \boldsymbol{e}$ an observation. Let $\boldsymbol{W} = \mathcal{X} - \boldsymbol{E}$. Then $P_{\mathcal{B}}(\boldsymbol{W} \mid \boldsymbol{e})$ is a Gibbs distribution defined by the factors $\Phi = \{\phi_{X_i}\}_{X_i \in \mathcal{X}}$, where*

$$\phi_{X_i} = P_{\mathcal{B}}(X_i \mid \mathrm{Pa}_{X_i})[\boldsymbol{E} = \boldsymbol{e}].$$

The partition function for this Gibbs distribution is $P(\boldsymbol{e})$.

The proof follows directly from the definitions. This result allows us to view any Bayesian network conditioned as evidence as a Gibbs distribution, and to bring to bear techniques developed for analysis of Markov networks.

What is the structure of the undirected graph that can serve as an I-map for a set of factors in a Bayesian network? In other words, what is the I-map for the Bayesian network structure $\mathcal{G}$? Going back to our construction, we see that we have created a factor for each family of X_i, containing all the variables in the family. Thus, in the undirected I-map, we need to have an edge between X_i and each of its parents, as well as between all of the parents of X_i. This observation motivates the following definition:

Definition 4.16 moralized graph

The moral graph *$\mathcal{M}[\mathcal{G}]$ of a Bayesian network structure $\mathcal{G}$ over $\mathcal{X}$ is the undirected graph over $\mathcal{X}$*

that contains an undirected edge between X and Y if: (a) there is a directed edge between them (in either direction), or (b) X and Y are both parents of the same node.[1] ■

For example, figure 4.6a shows the moralized graph for the $\mathcal{B}^{student}$ network of figure 3.3.

The preceding discussion shows the following result:

Corollary 4.2 *Let $\mathcal{G}$ be a Bayesian network structure. Then for any distribution $P_{\mathcal{B}}$ such that $\mathcal{B}$ is a parameterization of $\mathcal{G}$, we have that $\mathcal{M}[\mathcal{G}]$ is an I-map for $P_{\mathcal{B}}$.*

One can also view the moralized graph construction purely from the perspective of the independencies encoded by a graph, avoiding completely the discussion of parameterizations of the network.

Proposition 4.8 *Let $\mathcal{G}$ be any Bayesian network graph. The moralized graph $\mathcal{M}[\mathcal{G}]$ is a minimal I-map for $\mathcal{G}$.*

Proof We want to build a Markov network $\mathcal{H}$ such that $\mathcal{I}(\mathcal{H}) \subseteq \mathcal{I}(\mathcal{G})$, that is, that $\mathcal{H}$ is an I-map for $\mathcal{G}$ (see definition 3.3). We use the algorithm for constructing minimal I-maps based on the Markov independencies. Consider a node X in $\mathcal{X}$: our task is to select as X's neighbors the smallest set of nodes $\boldsymbol{U}$ that are needed to render X independent of all other nodes in the network. We define the *Markov blanket* of X in a Bayesian network $\mathcal{G}$, denoted $\mathrm{MB}_{\mathcal{G}}(X)$, to be the nodes consisting of X's parents, X's children, and other parents of X's children. We now need to show that $\mathrm{MB}_{\mathcal{G}}(X)$ d-separates X from all other variables in $\mathcal{G}$; and that no subset of $\mathrm{MB}_{\mathcal{G}}(X)$ has that property. The proof uses straightforward graph-theoretic properties of trails, and it is left as an exercise (exercise 4.14). ■

Markov blanket

Now, let us consider how "close" the moralized graph is to the original graph $\mathcal{G}$. Intuitively, **the addition of the moralizing edges to the Markov network $\mathcal{H}$ leads to the loss of independence information implied by the graph structure.** For example, if our Bayesian network $\mathcal{G}$ has the form $X \rightarrow Z \leftarrow Y$, with no edge between X and Y, the Markov network $\mathcal{M}[\mathcal{G}]$ loses the information that X and Y are marginally independent (not given Z). However, information is not always lost. Intuitively, moralization causes loss of information about independencies only when it introduces new edges into the graph. We say that a Bayesian network $\mathcal{G}$ is *moral* if it contains no immoralities (as in definition 3.11); that is, for any pair of variables X, Y that share a child, there is a covering edge between X and Y. It is not difficult to show that:

moral graph

Proposition 4.9 *If the directed graph $\mathcal{G}$ is moral, then its moralized graph $\mathcal{M}[\mathcal{G}]$ is a perfect map of $\mathcal{G}$.*

Proof Let $\mathcal{H} = \mathcal{M}[\mathcal{G}]$. We have already shown that $\mathcal{I}(\mathcal{H}) \subseteq \mathcal{I}(\mathcal{G})$, so it remains to show the opposite inclusion. Assume by contradiction that there is an independence $(\boldsymbol{X} \perp \boldsymbol{Y} \mid \boldsymbol{Z}) \in \mathcal{I}(\mathcal{G})$ which is not in $\mathcal{I}(\mathcal{H})$. Thus, there must exist some trail from $\boldsymbol{X}$ to $\boldsymbol{Y}$ in $\mathcal{H}$ which is active given $\boldsymbol{Z}$. Consider some such trail that is minimal, in the sense that it has no shortcuts. As $\mathcal{H}$ and $\mathcal{G}$ have precisely the same edges, the same trail must exist in $\mathcal{G}$. As, by assumption, it cannot be active in $\mathcal{G}$ given $\boldsymbol{Z}$, we conclude that it must contain a v-structure $X_1 \rightarrow X_2 \leftarrow X_3$. However, because $\mathcal{G}$ is moralized, we also have some edge between X_1 and X_3, contradicting the assumption that the trail is minimal. ■

1. The name *moralized graph* originated because of the supposed "morality" of marrying the parents of a node.

Thus, a moral graph $\mathcal{G}$ can be converted to a Markov network without losing independence assumptions. This conclusion is fairly intuitive, inasmuch as the only independencies in $\mathcal{G}$ that are not present in an undirected graph containing the same edges are those corresponding to v-structures. But if any v-structure can be short-cut, it induces no independencies that are not represented in the undirected graph.

We note, however, that very few directed graphs are moral. For example, assume that we have a v-structure $X \rightarrow Y \leftarrow Z$, which is moral due to the existence of an arc $X \rightarrow Z$. If Z has another parent W, it also has a v-structure $X \rightarrow Z \leftarrow W$, which, to be moral, requires some edge between X and W. We return to this issue in section 4.5.3.

4.5.1.1 Soundness of d-Separation

The connection between Bayesian networks and Markov networks provides us with the tools for proving the soundness of the d-separation criterion in Bayesian networks.

The idea behind the proof is to leverage the soundness of separation in undirected graphs, a result which (as we showed) is much easier to prove. Thus, we want to construct an undirected graph $\mathcal{H}$ such that active paths in $\mathcal{H}$ correspond to active paths in $\mathcal{G}$. A moment of thought shows that the moralized graph is not the right construct, because there are paths in the undirected graph that correspond to v-structures in $\mathcal{G}$ that may or may not be active. For example, if our graph $\mathcal{G}$ is $X \rightarrow Z \leftarrow Y$ and Z is not observed, d-separation tells us that X and Y are independent; but the moralized graph for $\mathcal{G}$ is the complete undirected graph, which does not have the same independence.

barren node

Therefore, to show the result, we first want to eliminate v-structures that are not active, so as to remove such cases. To do so, we first construct a subgraph where remove all *barren nodes* from the graph, thereby also removing all v-structures that do not have an observed descendant. The elimination of the barren nodes does not change the independence properties of the distribution over the remaining variables, but does eliminate paths in the graph involving v-structures that are not active. If we now consider only the subgraph, we can reduce d-separation to separation and utilize the soundness of separation to show the desired result.

upward closure

We first use these intuitions to provide an alternative formulation for d-separation. Recall that in definition 2.14 we defined the *upward closure* of a set of nodes $\boldsymbol{U}$ in a graph to be $\boldsymbol{U} \cup \mathit{Ancestors}_{\boldsymbol{U}}$. Letting $\boldsymbol{U}^*$ be the closure of a set $\boldsymbol{U}$, we can define the network induced over $\boldsymbol{U}^*$; importantly, as all parents of every node in $\boldsymbol{U}^*$ are also in $\boldsymbol{U}^*$, we have all the variables mentioned in every CPD, so that the induced graph defines a coherent probability distribution. We let $\mathcal{G}^+[\boldsymbol{U}]$ be the induced Bayesian network over $\boldsymbol{U}$ and its ancestors.

Proposition 4.10 *Let $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$ be three disjoint sets of nodes in a Bayesian network $\mathcal{G}$. Let $\boldsymbol{U} = \boldsymbol{X} \cup \boldsymbol{Y} \cup \boldsymbol{Z}$, and let $\mathcal{G}' = \mathcal{G}^+[\boldsymbol{U}]$ be the induced Bayesian network over $\boldsymbol{U} \cup \text{Ancestors}_{\boldsymbol{U}}$. Let $\mathcal{H}$ be the moralized graph $\mathcal{M}[\mathcal{G}']$. Then* $\text{d-sep}_{\mathcal{G}}(\boldsymbol{X}; \boldsymbol{Y} \mid \boldsymbol{Z})$ *if and only if* $\text{sep}_{\mathcal{H}}(\boldsymbol{X}; \boldsymbol{Y} \mid \boldsymbol{Z})$.

Example 4.16 *To gain some intuition for this result, consider the Bayesian network $\mathcal{G}$ of figure 4.12a (which extends our* Student *network). Consider the d-separation query* $\text{d-sep}_{\mathcal{G}}(D; I \mid L)$. *In this case, $\boldsymbol{U} = \{D, I, L\}$, and hence the moralized graph $\mathcal{M}[\mathcal{G}^+[\boldsymbol{U}]]$ is the graph shown in figure 4.12b, where we have introduced an undirected moralizing edge between D and I. In the resulting graph,*

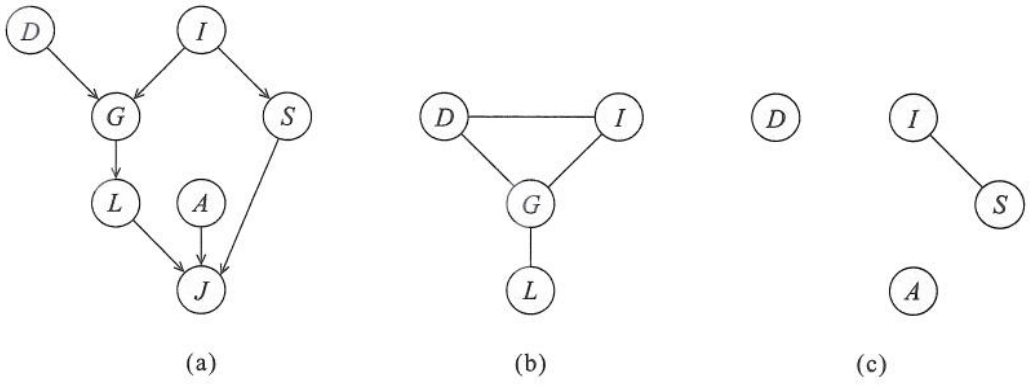


Figure 4.12 Example of alternative definition of d-separation based on Markov networks. (a) A Bayesian network $\mathcal{G}$. (b) The Markov network $\mathcal{M}[\mathcal{K}^+[D, I, L]]$. (c) The Markov network $\mathcal{M}[\mathcal{K}^+[D, I, A, S]]$.

D and I are not separated given L, exactly as we would have concluded using the d-separation procedure on the original graph.

On the other hand, consider the d-separation query $\text{d-sep}_{\mathcal{G}}(D; I \mid S, A)$. *In this case,* $\boldsymbol{U} = \{D, I, S, A\}$. *Because D and I are not spouses in $\mathcal{G}^+[\boldsymbol{U}]$, the moralization process does not add an edge between them. The resulting moralized graph is shown in figure 4.12c. As we can see, we have that* $\text{sep}_{\mathcal{M}[\mathcal{G}^+[\boldsymbol{U}]]}(D; I \mid S, A)$, *as desired.* ■

The proof for the general case is similar and is left as an exercise (exercise 4.15).

With this result, the soundness of d-separation follows easily. We repeat the statement of theorem 3.3:

Theorem 4.9 *If a distribution $P_{\mathcal{B}}$ factorizes according to $\mathcal{G}$, then $\mathcal{G}$ is an I-map for P.*

PROOF As in proposition 4.10, let $\boldsymbol{U} = \boldsymbol{X} \cup \boldsymbol{Y} \cup \boldsymbol{Z}$, let $\boldsymbol{U}^* = \boldsymbol{U} \cup \mathit{Ancestors}_{\boldsymbol{U}}$, let $\mathcal{G}_{\boldsymbol{U}^*} = \mathcal{G}^+[\boldsymbol{U}]$ be the induced graph over $\boldsymbol{U}^*$, and let $\mathcal{H}$ be the moralized graph $\mathcal{M}[\mathcal{G}_{\boldsymbol{U}^*}]$. Let $P_{\boldsymbol{U}^*}$ be the Bayesian network distribution defined over $\mathcal{G}_{\boldsymbol{U}^*}$ in the obvious way: the CPD for any variable in $\boldsymbol{U}^*$ is the same as in $\mathcal{B}$. Because $\boldsymbol{U}^*$ is upwardly closed, all variables used in these CPDs are in $\boldsymbol{U}^*$.

Now, consider an independence assertion $(\boldsymbol{X} \perp \boldsymbol{Y} \mid \boldsymbol{Z}) \in \mathcal{I}(\mathcal{G})$; we want to prove that $P_{\mathcal{B}} \models (\boldsymbol{X} \perp \boldsymbol{Y} \mid \boldsymbol{Z})$. By definition 3.7, if $(\boldsymbol{X} \perp \boldsymbol{Y} \mid \boldsymbol{Z}) \in \mathcal{I}(\mathcal{G})$, we have that $\textit{d-sep}_{\mathcal{G}}(\boldsymbol{X}; \boldsymbol{Y} \mid \boldsymbol{Z})$. It follows that $\textit{sep}_{\mathcal{H}}(\boldsymbol{X}; \boldsymbol{Y} \mid \boldsymbol{Z})$, and hence that $(\boldsymbol{X} \perp \boldsymbol{Y} \mid \boldsymbol{Z}) \in \mathcal{I}(\mathcal{H})$. $P_{\boldsymbol{U}^*}$ is a Gibbs distribution over $\mathcal{H}$, and hence, from theorem 4.1, $P_{\boldsymbol{U}^*} \models (\boldsymbol{X} \perp \boldsymbol{Y} \mid \boldsymbol{Z})$. Using exercise 3.8, the distribution $P_{\boldsymbol{U}^*}(\boldsymbol{U}^*)$ is the same as $P_{\mathcal{B}}(\boldsymbol{U}^*)$. Hence, it follows also that $P_{\mathcal{B}} \models (\boldsymbol{X} \perp \boldsymbol{Y} \mid \boldsymbol{Z})$, proving the desired result. ■

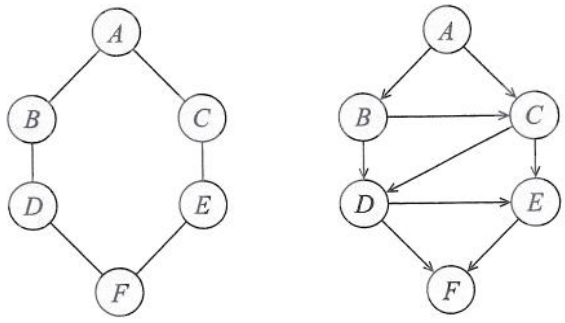


Figure 4.13 Minimal I-map Bayesian networks for a nonchordal Markov network. (a) A Markov network $\mathcal{H}_\ell$ with a loop. (b) A minimal I-map $\mathcal{G}_\ell$ Bayesian network for $\mathcal{H}$.

4.5.2 From Markov Networks to Bayesian Networks

The previous section dealt with the conversion from a Bayesian network to a Markov network. We now consider the converse transformation: finding a Bayesian network that is a minimal I-map for a Markov network. It turns out that the transformation in this direction is significantly more difficult, both conceptually and computationally. Indeed, the Bayesian network that is a minimal I-map for a Markov network might be considerably larger than the Markov network.

Example 4.17 *Consider the Markov network structure $\mathcal{H}_\ell$ of figure 4.13a, and assume that we want to find a Bayesian network I-map for $\mathcal{H}_\ell$. As we discussed in section 3.4.1, we can find such an I-map by enumerating the nodes in $\mathcal{X}$ in some ordering, and define the parent set for each one in turn according to the independencies in the distribution. Assume we enumerate the nodes in the order A, B, C, D, E, F. The process for A and B is obvious. Consider what happens when we add C. We must, of course, introduce A as a parent for C. More interestingly, however, C is not independent of B given A; hence, we must also add B as a parent for C. Now, consider the node D. One of its parents must be B. As D is not independent of C given B, we must add C as a parent for B. We do not need to add A, as D is independent of A given B and C. Similarly, E's parents must be C and D. Overall, the minimal Bayesian network I-map according to this ordering has the structure $\mathcal{G}_\ell$ shown in figure 4.13b.* ■

A quick examination of the structure $\mathcal{G}_\ell$ shows that we have added several edges to the graph, resulting in a set of triangles crisscrossing the loop. In fact, the graph $\mathcal{G}_\ell$ in figure 4.13b is chordal: all loops have been partitioned into triangles.

One might hope that a different ordering might lead to fewer edges being introduced. Unfortunately, this phenomenon is a general one: any Bayesian network I-map for this Markov network must add triangulating edges into the graph, so that the resulting graph is chordal (see definition 2.24). In fact, we can show the following property, which is even stronger:

Theorem 4.10 *Let $\mathcal{H}$ be a Markov network structure, and let $\mathcal{G}$ be any Bayesian network minimal I-map for $\mathcal{H}$. Then $\mathcal{G}$ can have no immoralities (see definition 3.11).*

PROOF Let $X_1, \ldots, X_n$ be a topological ordering for $\mathcal{G}$. Assume, by contradiction, that there is some immorality $X_i \rightarrow X_j \leftarrow X_k$ in $\mathcal{G}$ such that there is no edge between X_i and X_k; assume (without loss of generality) that $i < k < j$.

Owing to minimality of the I-map $\mathcal{G}$, if X_i is a parent of X_j, then X_i and X_j are not separated by X_j's other parents. Thus, $\mathcal{H}$ necessarily contains one or more paths between X_i and X_j that are not cut by X_k (or by X_j's other parents). Similarly, $\mathcal{H}$ necessarily contains one or more paths between X_k and X_j that are not cut by X_i (or by X_j's other parents).

Consider the parent set $\boldsymbol{U}$ that was chosen for X_k. By our previous argument, there are one or more paths in $\mathcal{H}$ between X_i and X_k via X_j. As $i < k$, and X_i is not a parent of X_k (by our assumption), we have that $\boldsymbol{U}$ must cut all of those paths. To do so, $\boldsymbol{U}$ must cut either all of the paths between X_i and X_j, or all of the paths between X_j and X_k: As long as there is at least one active path from X_i to X_j and one from X_j to X_k, there is an active path between X_i and X_k that is not cut by $\boldsymbol{U}$. Assume, without loss of generality, that $\boldsymbol{U}$ cuts all paths between X_j and X_k (the other case is symmetrical). Now, consider the choice of parent set for X_j, and recall that it is the (unique) minimal subset among $X_1, \ldots, X_{j-1}$ that separates X_j from the others. In a Markov network, this set consists of all nodes in $X_1, \ldots, X_{j-1}$ that are the first on some uncut path from X_j. As $\boldsymbol{U}$ separates X_k from X_j, it follows that X_k cannot be the first on any uncut path from X_j, and therefore X_k cannot be a parent of X_j. This result provides the desired contradiction. ■

Because any nontriangulated loop of length at least 4 in a Bayesian network graph necessarily contains an immorality, we conclude:

Corollary 4.3 *Let $\mathcal{H}$ be a Markov network structure, and let $\mathcal{G}$ be any minimal I-map for $\mathcal{H}$. Then $\mathcal{G}$ is necessarily chordal.*

triangulation

Thus, the process of turning a Markov network into a Bayesian network requires that we add enough edges to a graph to make it chordal. This process is called *triangulation*. As in the transformation from Bayesian networks to Markov networks, the addition of edges leads to the loss of independence information. For instance, in example 4.17, the Bayesian network $\mathcal{G}_\ell$ in figure 4.13b loses the information that C and D are independent given A and F. In the transformation from directed to undirected models, however, the edges added are only the ones that are, in some sense, implicitly there — the edges required by the fact that each factor in a Bayesian network involves an entire family (a node and its parents). By contrast, the transformation from Markov networks to Bayesian networks can lead to the introduction of a large number of edges, and, in many cases, to the creation of very large families (exercise 4.16).

4.5.3 Chordal Graphs

We have seen that the conversion in either direction between Bayesian networks to Markov networks can lead to the addition of edges to the graph and to the loss of independence information implied by the graph structure. It is interesting to ask when a set of independence assumptions can be represented perfectly by both a Bayesian network and a Markov network. It turns out that this class is precisely the class of undirected chordal graphs.

The proof of one direction is fairly straightforward, based on our earlier results.

Theorem 4.11 *Let $\mathcal{H}$ be a nonchordal Markov network. Then there is no Bayesian network $\mathcal{G}$ which is a perfect map for $\mathcal{H}$ (that is, such that $\mathcal{I}(\mathcal{H}) = \mathcal{I}(\mathcal{G})$).*

PROOF The proof follows from the fact that the minimal I-map for $\mathcal{G}$ must be chordal. Hence, any I-map $\mathcal{G}$ for $\mathcal{I}(\mathcal{H})$ must include edges that are not present in $\mathcal{H}$. Because any additional edge eliminates independence assumptions, it is not possible for any Bayesian network $\mathcal{G}$ to precisely encode $\mathcal{I}(\mathcal{H})$. ■

To prove the other direction of this equivalence, we first prove some important properties of chordal graphs. As we will see, chordal graphs and the properties we now show play a central role in the derivation of exact inference algorithms for graphical models. For the remainder of this discussion, we restrict attention to connected graphs; the extension to the general case is straightforward. The basic result we show is that we can decompose any connected chordal graph $\mathcal{H}$ into a *tree of cliques* — a tree whose nodes are the maximal cliques in $\mathcal{H}$ — so that the structure of the tree precisely encodes the independencies in $\mathcal{H}$. (In the case of disconnected graphs, we obtain a forest of cliques, rather than a tree.)

We begin by introducing some notation. Let $\mathcal{H}$ be a connected undirected graph, and let $\boldsymbol{C}_1, \ldots, \boldsymbol{C}_k$ be the set of maximal cliques in $\mathcal{H}$. Let $\mathcal{T}$ be any tree-structured graph whose nodes correspond to the maximal cliques $\boldsymbol{C}_1, \ldots, \boldsymbol{C}_k$. Let $\boldsymbol{C}_i, \boldsymbol{C}_j$ be two cliques in the tree that are directly connected by an edge; we define $\boldsymbol{S}_{i,j} = \boldsymbol{C}_i \cap \boldsymbol{C}_j$ to be a *sepset* between $\boldsymbol{C}_i$ and $\boldsymbol{C}_j$. Let $\boldsymbol{W}_{<(i,j)}$ ($\boldsymbol{W}_{<(j,i)}$) be all of the variables that appear in any clique on the $\boldsymbol{C}_i$ ($\boldsymbol{C}_j$) side of the edge. Thus, each edge decomposes $\mathcal{X}$ into three disjoint sets: $\boldsymbol{W}_{<(i,j)} - \boldsymbol{S}_{i,j}$, $\boldsymbol{W}_{<(j,i)} - \boldsymbol{S}_{i,j}$, and $\boldsymbol{S}_{i,j}$.

sepset

Definition 4.17 *We say that a tree $\mathcal{T}$ is a* clique tree *for $\mathcal{H}$ if:*

clique tree

- *each node corresponds to a clique in $\mathcal{H}$, and each maximal clique in $\mathcal{H}$ is a node in $\mathcal{T}$;*
- *each sepset $\boldsymbol{S}_{i,j}$ separates $\boldsymbol{W}_{<(i,j)}$ and $\boldsymbol{W}_{<(j,i)}$ in $\mathcal{H}$.* ■

Note that this definition implies that each separator $\boldsymbol{S}_{i,j}$ renders its two sides conditionally independent in $\mathcal{H}$.

Example 4.18 *Consider the Bayesian network graph $\mathcal{G}_\ell$ in figure 4.13b. Since it contains no immoralities, its moralized graph $\mathcal{H}'_\ell$ is simply the same graph, but where all edges have been made undirected. As $\mathcal{G}_\ell$ is chordal, so is $\mathcal{H}'_\ell$. The clique tree for $\mathcal{H}'_\ell$ is simply a chain $\{A, B, C\} \rightarrow \{B, C, D\} \rightarrow \{C, D, E\} \rightarrow \{D, E, F\}$, which clearly satisfies the separation requirements of the clique tree definition.* ■

Theorem 4.12 *Every undirected chordal graph $\mathcal{H}$ has a clique tree $\mathcal{T}$.*

PROOF We prove the theorem by induction on the number of nodes in the graph. The base case of a single node is trivial. Now, consider a chordal graph $\mathcal{H}$ of size > 1. If $\mathcal{H}$ consists of a single clique, then the theorem holds trivially. Therefore, consider the case where we have at least two nodes X_1, X_2 that are not connected directly by an edge. Assume that X_1 and X_2

are connected, otherwise the inductive step holds trivially. Let $\boldsymbol{S}$ be a minimal subset of nodes that separates X_1 and X_2.

The removal of the set $\boldsymbol{S}$ breaks up the graph into at least two disconnected components — one containing X_1, another containing X_2, and perhaps additional ones. Let $\boldsymbol{W}_1, \boldsymbol{W}_2$ be some partition of the variables in $\mathcal{X} - \boldsymbol{S}$ into two disjoint components, such that $\boldsymbol{W}_i$ encompasses the connected component containing X_i. (The other connected components can be assigned to $\boldsymbol{W}_1$ or $\boldsymbol{W}_2$ arbitrarily.) We first show that $\boldsymbol{S}$ must be a complete subgraph. Let Z_1, Z_2 be any two variables in $\boldsymbol{S}$. Due to the minimality of $\boldsymbol{S}$, each Z_i must lie on a path between X_1 and X_2 that does not go through any other node in $\boldsymbol{S}$. (Otherwise, we could eliminate Z_i from $\boldsymbol{S}$ while still maintaining separation.) We can therefore construct a minimal path from Z_1 to Z_2 that goes only through nodes in $\boldsymbol{W}_1$ by constructing a path from Z_1 to X_1 to Z_2 that goes only through $\boldsymbol{W}_1$, and by eliminating any shortcuts. We can similarly construct a minimal path from Z_1 to Z_2 that goes only through nodes in $\boldsymbol{W}_2$. The two paths together form a cycle of length ≥ 4. Because of chordality, the cycle must have a chord, which, by construction, must be the edge Z_1—Z_2.

Now consider the induced graph $\mathcal{H}_1 = \mathcal{H}[\boldsymbol{W}_1 \cup \boldsymbol{S}]$. As $X_2 \notin \mathcal{H}_1$, this induced graph is smaller than $\mathcal{H}$. Moreover, $\mathcal{H}_1$ is chordal, so we can apply the inductive hypothesis. Let $\mathcal{T}_1$ be the clique tree for $\mathcal{H}_1$. Because $\boldsymbol{S}$ is a complete connected subgraph, it is either a maximal clique or a subset of some maximal clique in $\mathcal{H}_1$. Let $\boldsymbol{C}_1$ be some clique in $\mathcal{T}_1$ containing $\boldsymbol{S}$ (there may be more than one such clique). We can similarly define $\mathcal{H}_2$ and $\boldsymbol{C}_2$ for X_2. If neither $\boldsymbol{C}_1$ nor $\boldsymbol{C}_2$ is equal to $\boldsymbol{S}$, we construct a tree $\mathcal{T}$ that contains the union of the cliques in $\mathcal{T}_1$ and $\mathcal{T}_2$, and connects $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ by an edge. Otherwise, without loss of generality, let $\boldsymbol{C}_1 = \boldsymbol{S}$; we create $\mathcal{T}$ by merging $\mathcal{T}_1$ minus $\boldsymbol{C}_1$ into $\mathcal{T}_2$, making all of $\boldsymbol{C}_1$'s neighbors adjacent to $\boldsymbol{C}_2$ instead.

It remains to show that the resulting structure is a clique tree for $\mathcal{H}$. First, we note that there is no clique in $\mathcal{H}$ that intersects both $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$; hence, any maximal clique in $\mathcal{H}$ is a maximal clique in either $\mathcal{H}_1$ or $\mathcal{H}_2$ (or both in the possible case of $\boldsymbol{S}$), so that all maximal cliques in $\mathcal{H}$ appear in $\mathcal{T}$. Thus, the nodes in $\mathcal{T}$ are precisely the maximal cliques in $\mathcal{H}$. Second, we need to show that any $\boldsymbol{S}_{i,j}$ separates $\boldsymbol{W}_{<(i,j)}$ and $\boldsymbol{W}_{<(j,i)}$. Consider two variables $X \in \boldsymbol{W}_{<(i,j)}$ and $Y \in \boldsymbol{W}_{<(j,i)}$. First, assume that $X, Y \in \mathcal{H}_1$; as all the nodes in $\mathcal{H}_1$ are on the $\mathcal{T}_1$ side of the tree, we also have that $\boldsymbol{S}_{i,j} \subset \mathcal{H}_1$. Any path between two nodes in $\mathcal{H}_1$ that goes through $\boldsymbol{W}_2$ can be shortcut to go only through $\mathcal{H}_1$. Thus, if $\boldsymbol{S}_{i,j}$ separates X, Y in $\mathcal{H}_1$, then it also separates them in $\mathcal{H}$. The same argument applies for $X, Y \in \mathcal{H}_2$. Now, consider $X \in \boldsymbol{W}_1$ and $Y \in \boldsymbol{W}_2$. If $\boldsymbol{S}_{i,j} = \boldsymbol{S}$, the result follows from the fact that $\boldsymbol{S}$ separates $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$. Otherwise, assume that $\boldsymbol{S}_{i,j}$ is in $\mathcal{T}_1$, on the path from X to $\boldsymbol{C}_1$. In this case, we have that $\boldsymbol{S}_{i,j}$ separates X from $\boldsymbol{S}$, and $\boldsymbol{S}$ separates $\boldsymbol{S}_{i,j}$ from Y. The conclusion now follows from the transitivity of graph separation.

We have therefore constructed a clique tree for $\mathcal{H}$, proving the inductive claim. ∎

Using this result, we can show that the independencies in an undirected graph $\mathcal{H}$ can be captured perfectly in a Bayesian network if and only if $\mathcal{H}$ is chordal.

Theorem 4.13 *Let $\mathcal{H}$ be a chordal Markov network. Then there is a Bayesian network $\mathcal{G}$ such that $\mathcal{I}(\mathcal{H}) = \mathcal{I}(\mathcal{G})$.*

Proof Let $\mathcal{T}$ be the clique tree for $\mathcal{H}$, whose existence is guaranteed by theorem 4.12. We can select an ordering over the nodes in the Bayesian network as follows. We select an arbitrary

clique C_1 to be the root of the clique tree, and then order the cliques $C_1, \ldots, C_k$ using any topological ordering, that is, where cliques closer to the root are ordered first. We now order the nodes in the network in any ordering consistent with the clique ordering: if X_l first appears in C_i and X_m first appears in C_j, for $i < j$, then X_l must precede X_m in the ordering. We now construct a Bayesian network using the procedure Build-Minimal-I-Map of algorithm 3.2 applied to the resulting node ordering $X_1, \ldots, X_n$ and to $\mathcal{I}(\mathcal{H})$.

Let $\mathcal{G}$ be the resulting network. We first show that, when X_i is added to the graph, then X_i's parents are precisely $\boldsymbol{U}_i = \mathrm{Nb}_{X_i} \cap \{X_1, \ldots, X_{i-1}\}$, where Nb_{X_i} is the set of neighbors of X_i in $\mathcal{H}$. In other words, we want to show that X_i is independent of $\{X_1, \ldots, X_{i-1}\} - \boldsymbol{U}_i$ given $\boldsymbol{U}_i$. Let C_k be the first clique in the clique ordering to which X_i belongs. Then $\boldsymbol{U}_i \subset C_k$. Let C_l be the parent of C_k in the rooted clique tree. According to our selected ordering, all of the variables in C_l are ordered before any variable in $C_k - C_l$. Thus, $\boldsymbol{S}_{l,k} \subset \{X_1, \ldots, X_{i-1}\}$. Moreover, from our choice of ordering, none of $\{X_1, \ldots, X_{i-1}\} - \boldsymbol{U}_i$ are in any descendants of C_k in the clique tree. Thus, they are all in $\boldsymbol{W}_{<(l,k)}$. From theorem 4.12, it follows that $\boldsymbol{S}_{l,k}$ separates X_i from all of $\{X_1, \ldots, X_{i-1}\} - \boldsymbol{U}_i$, and hence that X_i is independent of all of $\{X_1, \ldots, X_{i-1}\} - \boldsymbol{U}_i$ given $\boldsymbol{U}_i$. It follows that $\mathcal{G}$ and $\mathcal{H}$ have the same set of edges. Moreover, we note that all of $\boldsymbol{U}_i$ are in C_k, and hence are connected in $\mathcal{G}$. Therefore, $\mathcal{G}$ is moralized. As $\mathcal{H}$ is the moralized undirected graph of $\mathcal{G}$, the result now follows from proposition 4.9. ■

For example, the graph $\mathcal{G}_\ell$ of figure 4.13b, and its moralized network $\mathcal{H}'_\ell$ encode precisely the same independencies. By contrast, as we discussed, there exists no Bayesian network that encodes precisely the independencies in the nonchordal network $\mathcal{H}_\ell$ of figure 4.13a.

Thus, we have shown that chordal graphs are precisely the intersection between Markov networks and Bayesian networks, in that the independencies in a graph can be represented exactly in both types of models if and only if the graph is chordal.

4.6 Partially Directed Models

So far, we have presented two distinct types of graphical models, based on directed and undirected graphs. We can unify both representations by allowing models that incorporate both directed and undirected dependencies. We begin by describing the notion of *conditional random field*, a Markov network with a directed dependency on some subset of variables. We then present a generalization of this framework to the class of *chain graphs*, an entire network in which undirected components depend on each other in a directed fashion.

4.6.1 Conditional Random Fields

So far, we have described the Markov network representation as encoding a joint distribution over $\mathcal{X}$. The same undirected graph representation and parameterization can also be used to encode a *conditional distribution* $P(\boldsymbol{Y} \mid \boldsymbol{X})$, where $\boldsymbol{Y}$ is a set of *target variables* and $\boldsymbol{X}$ is a (disjoint) set of *observed variables*. We will also see a directed analogue of this concept in section 5.6. In the case of Markov networks, this representation is generally called a *conditional random field* (CRF).

target variable

observed variable

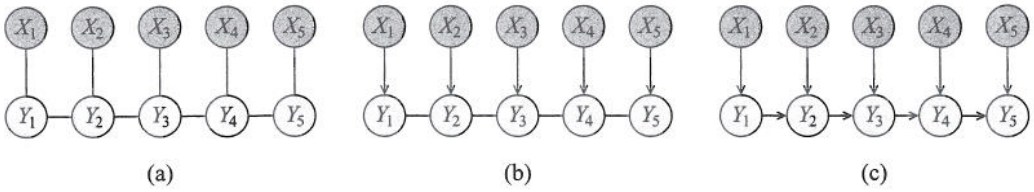


Figure 4.14 Different linear-chain graphical models: (a) a linear-chain-structured conditional random field, where the feature variables are denoted using grayed-out ovals; (b) a partially directed variant; (c) a fully directed, non-equivalent model. The X_i's are assumed to be always observed when the network is used, and hence they are shown as darker gray.

4.6.1.1 CRF Representation and Semantics

More formally, a CRF is an undirected graph whose nodes correspond to $\boldsymbol{Y} \cup \boldsymbol{X}$. At a high level, this graph is parameterized in the same way as an ordinary Markov network, as a set of factors $\phi_1(\boldsymbol{D}_1), \ldots, \phi_m(\boldsymbol{D}_m)$. (As before, these factors can also be encoded more compactly as a log-linear model; for uniformity of presentation, we view the log-linear model as encoding a set of factors.) However, rather than encoding the distribution $P(\boldsymbol{Y}, \boldsymbol{X})$, we view it as representing the conditional distribution $P(\boldsymbol{Y} \mid \boldsymbol{X})$. To have the network structure and parameterization correspond naturally to a conditional distribution, we want to avoid representing a probabilistic model over $\boldsymbol{X}$. We therefore disallow potentials that involve only variables in $\boldsymbol{X}$.

Definition 4.18 conditional random field

A conditional random field *is an undirected graph $\mathcal{H}$ whose nodes correspond to $\boldsymbol{X} \cup \boldsymbol{Y}$; the network is annotated with a set of factors $\phi_1(\boldsymbol{D}_1), \ldots, \phi_m(\boldsymbol{D}_m)$ such that each $\boldsymbol{D}_i \not\subseteq \boldsymbol{X}$. The network encodes a conditional distribution as follows:*

$$\begin{aligned} P(\boldsymbol{Y} \mid \boldsymbol{X}) &= \frac{1}{Z(\boldsymbol{X})} \tilde{P}(\boldsymbol{Y}, \boldsymbol{X}) \\ \tilde{P}(\boldsymbol{Y}, \boldsymbol{X}) &= \prod_{i=1}^{m} \phi_i(\boldsymbol{D}_i) \\ Z(\boldsymbol{X}) &= \sum_{\boldsymbol{Y}} \tilde{P}(\boldsymbol{Y}, \boldsymbol{X}). \end{aligned} \tag{4.11}$$

Two variables in $\mathcal{H}$ are connected by an (undirected) edge whenever they appear together in the scope of some factor. ■

The only difference between equation (4.11) and the standard Markov network definition in definition 4.4 is the different normalization used in the partition function $Z(\boldsymbol{X})$. The definition of a CRF induces a different value for the partition function for every assignment $\boldsymbol{x}$ to $\boldsymbol{X}$. This difference is denoted graphically by having the feature variables grayed out.

Example 4.19

Consider a CRF over $\boldsymbol{Y} = \{Y_1, \ldots, Y_k\}$ and $\boldsymbol{X} = \{X_1, \ldots, X_k\}$, with an edge Y_i—Y_{i+1} ($i = 1, \ldots, k-1$) and an edge Y_i—X_i ($i = 1, \ldots, k$), as shown in figure 4.14a. The distribution

represented by this network has the form:

$$\begin{aligned} P(\boldsymbol{Y} \mid \boldsymbol{X}) &= \frac{1}{Z(\boldsymbol{X})}\tilde{P}(\boldsymbol{Y}, \boldsymbol{X}) \\ \tilde{P}(\boldsymbol{Y}, \boldsymbol{X}) &= \prod_{i=1}^{k-1} \phi(Y_i, Y_{i+1}) \prod_{i=1}^{k} \phi(Y_i, X_i) \\ Z(\boldsymbol{X}) &= \sum_{\boldsymbol{Y}} \tilde{P}(\boldsymbol{Y}, \boldsymbol{X}). \end{aligned}$$

■

Note that, unlike the definition of a conditional Bayesian network, the structure of a CRF may still contain edges between variables in $\boldsymbol{X}$, which arise when two such variables appear together in a factor that also contains a target variable. However, these edges do not encode the structure of any distribution over $\boldsymbol{X}$, since the network explicitly does not encode any such distribution.

☞ **The fact that we avoid encoding the distribution over the variables in $\boldsymbol{X}$ is one of the main strengths of the CRF representation. This flexibility allows us to incorporate into the model a rich set of observed variables whose dependencies may be quite complex or even poorly understood. It also allows us to include continuous variables whose distribution may not have a simple parametric form. This flexibility allows us to use domain knowledge in order to define a rich set of features characterizing our domain, without worrying about modeling their joint distribution.** For example, returning to the vision MRFs of box 4.B, rather than defining a joint distribution over pixel values and their region assignment, we can define a conditional distribution over segment assignments *given* the pixel values. The use of a conditional distribution here allows us to avoid making a parametric assumption over the (continuous) pixel values. Even more important, we can use image-processing routines to define rich features, such as the presence or direction of an image gradient at a pixel. Such features can be highly informative in determining the region assignment of a pixel. However, the definition of such features usually relies on multiple pixels, and defining a correct joint distribution or a set of independence assumptions over these features is far from trivial. The fact that we can condition on these features and avoid this whole issue allows us the flexibility to include them in the model. See box 4.E for another example.

4.6.1.2 Directed and Undirected Dependencies

A CRF defines a conditional distribution of $\boldsymbol{Y}$ on $\boldsymbol{X}$; thus, it can be viewed as a partially directed graph, where we have an undirected component over Y, which has the variables in $\boldsymbol{X}$ as parents.

Example 4.20 naive Markov

Consider a CRF over the binary-valued variables $\boldsymbol{X} = \{X_1, \ldots, X_k\}$ and $\boldsymbol{Y} = \{Y\}$, and a pairwise potential between Y and each X_i; this model is sometimes known as a naive Markov *model, due to its similarity to the naive Bayes model. Assume that the pairwise potentials defined via the following log-linear model*

$$\phi_i(X_i, Y) = \exp\{w_i \mathbf{I}\{X_i = 1, Y = 1\}\}.$$

We also introduce a single-node potential $\phi_0(Y) = \exp\{w_0 \mathbf{I}\{Y = 1\}\}$. Following equation (4.11),

we now have:

$$\begin{aligned}\tilde{P}(Y = 1 \mid x_1, \ldots, x_k) &= \exp\left\{w_0 + \sum_{i=1}^{k} w_i x_i\right\} \\ \tilde{P}(Y = 0 \mid x_1, \ldots, x_k) &= \exp\{0\} = 1.\end{aligned}$$

In this case, we can show (exercise 5.16) that

$$P(Y = 1 \mid x_1, \ldots, x_k) = \text{sigmoid}\left(w_0 + \sum_{i=1}^{k} w_i x_i\right),$$

where

$$\text{sigmoid}(z) = \frac{e^z}{1 + e^z}$$

sigmoid

logistic CPD

is the sigmoid *function. This conditional distribution $P(Y \mid \boldsymbol{X})$ is of great practical interest: It defines a CPD that is not structured as a table, but that is induced by a small set of parameters $w_0, \ldots, w_k$ — parameters whose number is linear, rather than exponential, in the number of parents. This type of CPD, often called a* logistic CPD, *is a natural model for many real-world applications, inasmuch as it naturally aggregates the influence of different parents. We discuss this CPD in greater detail in section 5.4.2 as part of our general presentation of structured CPDs.* ■

The partially directed model for the CRF of example 4.19 is shown in figure 4.14b. We may be tempted to believe that we can construct an equivalent model that is fully directed, such as the one in figure 4.14c. In particular, conditioned on any assignment $\boldsymbol{x}$, the posterior distributions over $\boldsymbol{Y}$ in the two models satisfy the same independence assignments (the ones defined by the chain structure). However, the two models are not equivalent: In the Bayesian network, we have that Y_1 is independent of X_2 if we are not given Y_2. By contrast, in the original CRF, the unnormalized marginal measure of $\boldsymbol{Y}$ depends on the entire parameterization of the chain, and specifically the values of all of the variables in $\boldsymbol{X}$. A sound conditional Bayesian network for this distribution would require edges from all of the variables in $\boldsymbol{X}$ to each of the variables Y_i, thereby losing much of the structure in the distribution. See also box 20.A for further discussion.

Box 4.E — Case Study: CRFs for Text Analysis. *One important use for the CRF framework is in the domain of text analysis. Various models have been proposed for different tasks, including part-of-speech labeling, identifying named entities (people, places, organizations, and so forth), and extracting structured information from the text (for example, extracting from a reference list the publication titles, authors, journals, years, and the like). Most of these models share a similar structure: We have a target variable for each word (or perhaps short phrase) in the document, which encodes the possible labels for that word. Each target variable is connected to a set of feature variables that capture properties relevant to the target distinction. These methods are very popular in text analysis, both because the structure of the networks is a good fit for this domain, and because they produce state-of-the-art results for a broad range of natural-language processing problems.*

As a concrete example, consider the named entity recognition task, as described by Sutton and McCallum (2004, 2007). Entities often span multiple words, and the type of an entity may not be

apparent from individual words; for example, "New York" is a location, but "New York Times" is an organization. The problem of extracting entities from a word sequence of length T can be cast as a graphical model by introducing for each word, $X_t, 1 \leq t \leq T$, a target variable, Y_t, which indicates the entity type of the word. The outcomes of Y_t include B-PERSON, I-PERSON, B-LOCATION, I-LOCATION, B-ORGANIZATION, I-ORGANIZATION, *and* OTHER. *In this so-called "BIO notation,"* OTHER *indicates that the word is not part of an entity, the* B- *outcomes indicate the beginning of a named entity phrase, and the* I- *outcomes indicate the inside or end of the named entity phrase. Having a distinguishing label for the beginning versus inside of an entity phrase allows the model to segment adjacent entities of the same type.*

linear-chain CRF

A common structure for this problem is a linear-chain CRF *often having two factors for each word: one factor $\phi_t^1(Y_t, Y_{t+1})$ to represent the dependency between neighboring target variables, and another factor $\phi_t^2(Y_t, X_1, \ldots, X_T)$ that represents the dependency between a target and its context in the word sequence. Note that the second factor can depend on arbitrary features of the entire input word sequence. We generally do not encode this model using table factors, but using a log-linear model. Thus, the factors are derived from a number of feature functions, such as $f_t(Y_t, X_t) = \mathbb{I}\{Y_t =$ B-ORGANIZATION$, X_t =$ "Times"$\}$. We note that, just as logistic CPDs are the conditional analog of the naive Bayes classifier (example 4.20), the linear-chain CRF is the conditional analog of the* hidden Markov model *(HMM) that we present in section 6.2.3.1.*

hidden Markov model

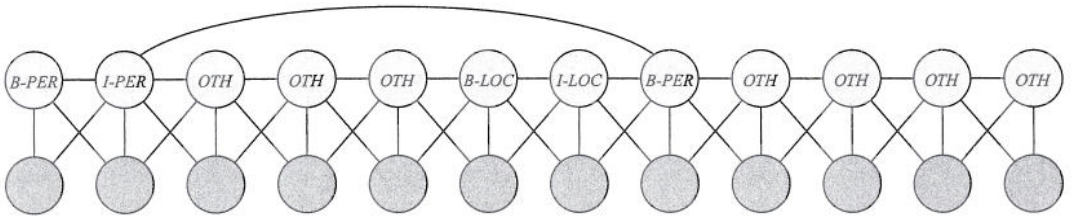
A large number of features of the word X_t and neighboring words are relevant to the named entity decision. These include features of the word itself: is it capitalized; does it appear in a list of common person names; does it appear in an atlas of location names; does it end with the character string "ton"; is it exactly the string "York"; is the following word "Times." Also relevant are aggregate features of the entire word sequence, such as whether it contains more than two sports-related words, which might be an indicator that "New York" is an organization (sports team) rather than a location. In addition, including features that are conjunctions of all these features often increases accuracy. The total number of features can be quite large, often in the hundreds of thousands or more if conjunctions of word pairs are used as features. However, the features are sparse, meaning that most features are zero for most words.

Note that the same feature variable can be connected to multiple target variables, so that Y_t would typically be dependent on the identity of several words in a window around position t. These contextual features are often highly indicative: for example, "Mrs." before a word and "spoke" after a word are both strong indicators that the word is a person. These context words would generally be used as a feature for multiple target variables. Thus, if we were using a simple naive-Bayes-style generative model, where each target variable is a parent of its associated feature, we either would have to deal with the fact that a context word has multiple parents or we would have to duplicate its occurrences (with one copy for each target variable for which it is in the context), and thereby overcount its contribution.

Linear-chain CRFs frequently provide per-token accuracies in the high 90 percent range on many natural data sets. Per-field precision and recall (where the entire phrase category and boundaries must be correct) are more often around 80–95 percent, depending on the data set.

Although the linear-chain model is often effective, additional information can be incorporated into the model by augmenting the graphical structure. For example, often when a word occurs multiple times in the same document, it often has the same label. This knowledge can be incorporated by including factors that connect identical words, resulting in a skip-chain CRF, *as shown in figure 4.E.1a. The first occurrence of the word "Green" has neighboring words that provide strong*

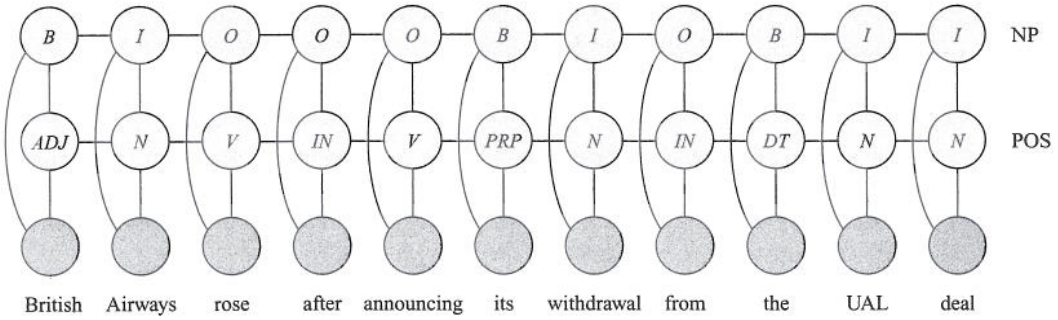
skip-chain CRF



KEY

B-PER	Begin person name	*I-LOC*	Within location name
I-PER	Within person name	*OTH*	Not an entitiy
B-LOC	Begin location name		

(a)



KEY

B	Begin noun phrase	*V*	Verb
I	Within noun phrase	*IN*	Preposition
O	Not a noun phrase	*PRP*	Possesive pronoun
N	Noun	*DT*	Determiner (e.g., a, an, the)
ADJ	Adjective		

(b)

Figure 4.E.1 — Two models for text analysis based on a linear chain CRF Gray nodes indicate $\boldsymbol{X}$ and clear nodes $\boldsymbol{Y}$. The annotations inside the $\boldsymbol{Y}$ are the true labels. (a) A skip chain CRF for named entity recognition, with connections between adjacent words and long-range connections between multiple occurrences of the same word. (b) A pair of coupled linear-chain CRFs that performs joint part-of-speech labeling and noun-phrase segmentation. Here, B indicates the beginning of a noun phrase, I other words in the noun phrase, and O words not in a noun phrase. The labels for the second chain are parts of speech.

evidence that it is a PERSON*'s name; however, the second occurrence is much more ambiguous. By augmenting the original linear-chain CRF with an additional long-range factor that prefers its connected target variables to have the same value, the model is more likely to predict correctly that the second occurrence is also a* PERSON*. This example demonstrates another flexibility of conditional models, which is that the graphical structure over $\boldsymbol{Y}$ can easily depend on the value of the $\boldsymbol{X}$'s.*

CRFs having a wide variety of model structures have been successfully applied to many different tasks. Joint inference of both part-of-speech labels and noun-phrase segmentation has been performed with two connected linear chains (somewhat analogous to a coupled hidden Markov mode, *shown in figure 6.3). This structure is illustrated in figure 4.E.1b.*

coupled HMM

4.6.2 Chain Graph Models ⋆

We now present a more general framework that builds on the CRF representation and can be used to provide a general treatment of the independence assumptions made in these partially directed models. Recall from definition 2.21 that, in a *partially directed acyclic graph* (PDAG), the nodes can be disjointly partitioned into several *chain components*. An edge between two nodes in the same chain component must be undirected, while an edge between two nodes in different chain components must be directed. Thus, PDAGs are also called *chain graphs*.

partially directed acyclic graph

chain component

chain graph

4.6.2.1 Factorization

As in our other graphical representations, the structure of a PDAG $\mathcal{K}$ can be used to define a factorization for a probability distribution over $\mathcal{K}$. Intuitively, the factorization for PDAGs represents the distribution as a product of each of the chain components given its parents. Thus, we call such a representation a *chain graph model*.

chain graph model

Intuitively, each chain component $\boldsymbol{K}_i$ in the chain graph model is associated with a CRF that defines $\mathcal{P}(\boldsymbol{K}_i \mid \mathrm{Pa}_{\boldsymbol{K}_i})$ — the conditional distribution of $\boldsymbol{K}_i$ given its parents in the graph. More precisely, each is defined via a set of factors that involve the variables in $\boldsymbol{K}_i$ and their parents; the distribution $\mathcal{P}(K_i \mid \mathrm{Pa}_{\boldsymbol{K}_i})$ is defined by using the factors associated with $\boldsymbol{K}_i$ to define a CRF whose target variables are $\boldsymbol{K}_i$ and whose observable variables are $\mathrm{Pa}_{\boldsymbol{K}_i}$.

To provide a formal definition, it helps to introduce the concept of a moralized PDAG.

Definition 4.19 moralized graph

Let $\mathcal{K}$ be a PDAG and $\boldsymbol{K}_1, \ldots, \boldsymbol{K}_\ell$ be its chain components. We define $\mathrm{Pa}_{\boldsymbol{K}_i}$ to be the parents of nodes in $\boldsymbol{K}_i$. The moralized graph *of $\mathcal{K}$ is an undirected graph $\mathcal{M}[\mathcal{K}]$ produced by first connecting, using undirected edges, any pair of nodes $X, Y \in \mathrm{Pa}_{\boldsymbol{K}_i}$ for all $i = 1, \ldots, \ell$, and then converting all directed edges into undirected edges.* ■

This definition generalizes our earlier notion of a moralized directed graph. In the case of directed graphs, each node is its own chain component, and hence we are simply adding undirected edges between the parents of each node.

Example 4.21

Figure 4.15 shows a chain graph and its moral graph. We have added the edge between A and B, since they are both parents of the chain component $\{C, D, E\}$, and edges between C, E, and H, because they are parents of the chain component $\{I\}$. Note that we did not add an edge between

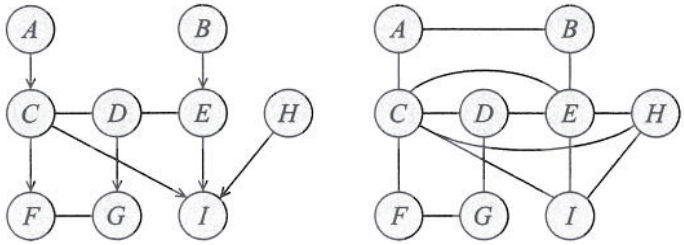


Figure 4.15 A chain graph $\mathcal{K}$ and its moralized version

D and H (even though D and C, E are in the same chain component), since D is not a parent of I. ■

We can now define the factorization of a chain graph:

Definition 4.20 chain graph distribution

Let $\mathcal{K}$ be a PDAG, and $\boldsymbol{K}_1, \ldots, \boldsymbol{K}_\ell$ be its chain components. A chain graph distribution *is defined via a set of factors $\phi_i(\boldsymbol{D}_i)$ $(i = 1, \ldots, m)$, such that each $\boldsymbol{D}_i$ is a complete subgraph in the moralized graph $\mathcal{M}[\mathcal{K}]$. We associate each factor $\phi_i(\boldsymbol{D}_i)$ with a single chain component $\boldsymbol{K}_j$, such that $\boldsymbol{D}_i \subseteq \boldsymbol{K}_i \cup \mathrm{Pa}_{\boldsymbol{K}_i}$ and define $P(\boldsymbol{K}_i \mid \mathrm{Pa}_{\boldsymbol{K}_i})$ as a CRF with these factors, and with $\boldsymbol{Y}_i = \boldsymbol{K}_i$ and $\boldsymbol{X}_i = \mathrm{Pa}_{\boldsymbol{K}_i}$. We now define*

$$P(\mathcal{X}) = \prod_{i=1}^{\ell} P(\boldsymbol{K}_i \mid \mathrm{Pa}_{\boldsymbol{K}_i}).$$

■

We say that a distribution P factorizes over $\mathcal{K}$ if it can be represented as a chain graph distribution over $\mathcal{K}$.

Example 4.22

In the chain graph model defined by the graph of figure 4.15, we require that the conditional distribution $P(C, D, E \mid A, B)$ factorize according to the graph of figure 4.16a. Specifically, we would have to define the conditional probability as a normalized product of factors:

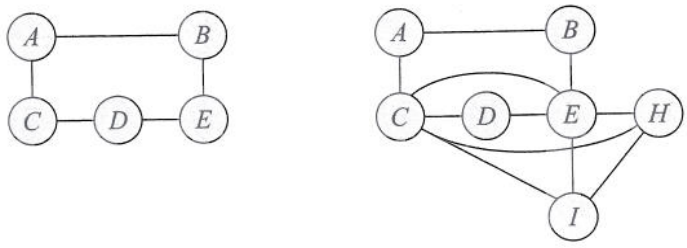
$$\frac{1}{Z(A,B)}\phi_1(A,C)\phi_2(B,E)\phi_3(C,D)\phi_4(D,E).$$

A similar factorization applies to $P(F, G \mid C, D)$. ■

4.6.2.2 Independencies in Chain Graphs

As for undirected graphs, there are three distinct interpretations for the independence properties induced by a PDAG. Recall that in a PDAG, we have both the notion of parents of X (variables Y such that $Y \rightarrow X$ is in the graph) and neighbors of X (variables Y such that $Y—X$ is in the graph). The union of these two sets is the *boundary* of X, denoted $\mathrm{Boundary}_X$. Also recall, from definition 2.15, that the descendants of X are those nodes Y that can be reached using any directed path, where a directed path can involve both directed and undirected edges but must contain at least one edge directed from X to Y, and no edges directed from Y to

boundary



(a) (b)

Figure 4.16 Example for definition of c-separation in a chain graph. (a) The Markov network $\mathcal{M}[\mathcal{G}^+[C, D, E]]$. (b) The Markov network $\mathcal{M}[\mathcal{G}^+[C, D, E, I]]$.

X. Thus, in the case of PDAGs, it follows that if Y is a descendant of X, then Y must be in a "lower" chain component.

Definition 4.21 pairwise independencies

For a PDAG $\mathcal{K}$, we define the pairwise independencies *associated with $\mathcal{K}$ to be:*

$$\mathcal{I}_p(\mathcal{K}) = \{(X \perp Y \mid (\text{NonDescendants}_X - \{X, Y\})) : X, Y \text{ non-adjacent}, Y \in \text{NonDescendants}_X\}.$$ ■

This definition generalizes the pairwise independencies for undirected graphs: in an undirected graph, nodes have no descendants, so $\text{NonDescendants}_X = \mathcal{X}$. Similarly, it is not too hard to show that these independencies also hold in a directed graph.

Definition 4.22 local independencies

For a PDAG $\mathcal{K}$, we define the local independencies *associated with $\mathcal{K}$ to be:*

$$\mathcal{I}_\ell(\mathcal{K}) = \{(X \perp \text{NonDescendants}_X - \text{Boundary}_X \mid \text{Boundary}_X) : X \in \mathcal{X}\}.$$ ■

This definition generalizes the definition of local independencies for both directed and undirected graphs. For directed graphs, NonDescendants_X is precisely the set of nondescendants, whereas Boundary_X is the set of parents. For undirected graphs, NonDescendants_X is $\mathcal{X}$, whereas $\text{Boundary}_X = \text{Nb}_X$.

We define the global independencies in a PDAG using the definition of moral graph. Our definition follows the lines of proposition 4.10.

Definition 4.23 c-separation

Let $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z} \subset \mathcal{X}$ be three disjoint sets, and let $\boldsymbol{U} = \boldsymbol{X} \cup \boldsymbol{Y} \cup \boldsymbol{Z}$. We say that $\boldsymbol{X}$ is c-separated *from $\boldsymbol{Y}$ given $\boldsymbol{Z}$ if $\boldsymbol{X}$ is separated from $\boldsymbol{Y}$ given $\boldsymbol{Z}$ in the undirected graph $\mathcal{M}[\mathcal{K}^+[\boldsymbol{X} \cup \boldsymbol{Y} \cup \boldsymbol{Z}]]$.* ■

Example 4.23

Consider again the PDAG of figure 4.15. Then C is c-separated from E given D, A, because C and E are separated given D, A in the undirected graph $\mathcal{M}[\mathcal{K}^+[\{C, D, E\}]]$, shown in figure 4.16a. However, C is not c-separated from E given only D, since there is a path between C and E via A, B. On the other hand, C is not separated from E given D, A, I. The graph $\mathcal{M}[\mathcal{K}^+[\{C, D, E, I\}]]$ is shown in figure 4.16b. As we can see, the introduction of I into the set $\boldsymbol{U}$ causes us to introduce a direct edge between C and E in order to moralize the graph. Thus, we cannot block the path between A and E using D, A, I. ■

This notion of c-separation clearly generalizes the notion of separation in undirected graphs, since the ancestors of a set $\boldsymbol{U}$ in an undirected graph are simply the entire set of nodes $\mathcal{X}$. It also generalizes the notion of d-separation in directed graphs, using the equivalent definition provided in proposition 4.10. Using the definition of c-separation, we can finally define the notion of global Markov independencies:

Definition 4.24 *Let $\mathcal{K}$ be a PDAG. We define the* local independencies *associated with $\mathcal{K}$ to be:*

local independencies

$$\mathcal{I}(\mathcal{K}) = \{(\boldsymbol{X} \perp \boldsymbol{Y} \mid \boldsymbol{Z}) : \boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z} \subset \mathcal{X}, \boldsymbol{X} \text{ is c-separated from } \boldsymbol{Y} \text{ given } \boldsymbol{Z}\}. \quad \blacksquare$$

As in the case of undirected models, these three criteria for independence are not equivalent for nonpositive distributions. The inclusions are the same: the global independencies imply the local independencies, which in turn imply the pairwise independencies. Because undirected models are a subclass of PDAGs, the same counterexamples used in section 4.3.3 show that the inclusions are strict for nonpositive distributions. For positive distributions, we again have that the three definitions are equivalent.

We note that, as in the case of Bayesian networks, the parents of a chain component are always fully connected in $\mathcal{M}[\mathcal{K}[\boldsymbol{K}_i \cup \mathrm{Pa}_{\boldsymbol{K}_i}]]$. Thus, while the structure over the parents helps factorize the distribution over the chain components containing the parents, it does not give rise to independence assertions in the conditional distribution over the child chain component. Importantly, however, it does give rise to structure in the form of the parameterization of $P(\boldsymbol{K}_i \mid \mathrm{Pa}_{\boldsymbol{K}_i})$, as we saw in example 4.20.

As in the case of directed and undirected models, we have an equivalence between the requirement of factorization of a distribution and the requirement that it satisfy the independencies associated with the graph. Not surprisingly, since PDAGs generalize undirected graphs, this equivalence only holds for positive distributions:

Theorem 4.14 *A positive distribution P factorizes over a PDAG $\mathcal{K}$ if and only if $P \models \mathcal{I}(\mathcal{K})$.*

We omit the proof.

4.7 Summary and Discussion

In this chapter, we introduced *Markov networks*, an alternative graphical modeling language for probability distributions, based on undirected graphs.

We showed that Markov networks, like Bayesian networks, can be viewed as defining a set of independence assumptions determined by the graph structure. In the case of undirected models, there are several possible definitions for the independence assumptions induced by the graph, which are equivalent for positive distributions. As in the case of Bayesian network, we also showed that the graph can be viewed as a data structure for specifying a probability distribution in a factored form. The factorization is defined as a product of factors (general nonnegative functions) over cliques in the graph. We showed that, for positive distributions, the two characterizations of undirected graphs — as specifying a set of independence assumptions and as defining a factorization — are equivalent.

Markov networks also provide useful insight on Bayesian networks. In particular, we showed how a Bayesian network can be viewed as a Gibbs distribution. More importantly, the unnormalized measure we obtain by introducing evidence into a Bayesian network is also a Gibbs

distribution, whose partition function is the probability of the evidence. This observation will play a critical role in providing a unified view of inference in graphical models.

We investigated the relationship between Bayesian networks and Markov networks and showed that the two represent different families of independence assumptions. The difference in these independence assumptions is a key factor in deciding which of the two representations to use in encoding a particular domain. There are domains where interactions have a natural directionality, often derived from causal intuitions. **In this case, the independencies derived from the network structure directly reflect patterns such as intercausal reasoning. Markov networks represent only monotonic independence patterns: observing a variable can only serve to remove dependencies, not to activate them.** Of course, we can encode a distribution with "causal" connections as a Gibbs distribution, and it will exhibit the same nonmonotonic independencies. However, these independencies will not be manifest in the network structure.

In other domains, the interactions are more symmetrical, and attempts to force a directionality give rise to models that are unintuitive and that often are incapable of capturing the independencies in the domain (see, for example, section 6.6). As a consequence, the use of undirected models has increased steadily, most notably in fields such as computer vision and natural language processing, where the acyclicity requirements of directed graphical models are often at odds with the nature of the model. The flexibility of the undirected model also allows the distribution to be decomposed into factors over multiple overlapping "features" without having to worry about defining a single normalized generating distribution for each variable. Conversely, **this very flexibility and the associated lack of clear semantics for the model parameters often make it difficult to elicit models from experts. Therefore, many recent applications use learning techniques to estimate parameters from data, avoiding the need to provide a precise semantic meaning for each of them.**

Finally, the question of which class of models better encodes the properties of the distribution is only one factor in the selection of a representation. There are other important distinctions between these two classes of models, especially when it comes to learning from data. We return to these topics later in the book (see, for example, box 20.A).

4.8 Relevant Literature

contingency table

The representation of a probability distribution as an undirected graph has its roots in the *contingency table* representation that is a staple in statistical modeling. The idea of representing probabilistic interactions in this representation dates back at least as early as the work of Bartlett (1935). This line of work is reviewed in detail by Whittaker (1990) and Lauritzen (1996), and we refer the reader to those sources and the references therein.

A parallel line of work involved the development of the Markov network (or Markov random field) representation. Here, the starting point was a graph object rather than a distribution object (such as a contingency table). Isham (1981) surveys some of the early work along these lines.

The connection between the undirected graph representation and the Gibbs factorization of the distribution was first made in the unpublished work of Hammersley and Clifford (1971). As a consequence, they also showed the equivalence of the different types of (local, pairwise, and global) independence properties for undirected graphs in the case of positive distributions.

Lauritzen (1982) made the connection between MRFs and contingency tables, and proved

some of the key results regarding the independence properties arising form the undirected representation. The line of work analyzing independence properties was then significantly extended by Pearl and Paz (1987). The history of these developments and other key references are presented by Pearl (1988) and Lauritzen (1996). The independence properties of chain graphs were studied in detail by Frydenberg (1990); see also Lauritzen (1996). Studený and Bouckaert (1998) also provide an alternative definition of the independence properties in chain graphs, one that is equivalent to c-separation but more directly analogous to the definition of d-separation in directed graphs.

Factor graphs were presented by Kschischang et al. (2001a) and extended by Frey (2003) to encompass both Bayesian networks and Markov networks. The framework of conditional random fields (CRFs) was first proposed by Lafferty, McCallum, and Pereira (2001). They have subsequently been used in a broad range of applications in natural language processing, computer vision, and many more. Skip-chain CRFs were introduced by Sutton and McCallum (2004), and factorial CRFs by Sutton et al. (2007). Sutton and McCallum (2007) also provide an overview of this framework and some of its applications.

Ising models were first proposed by Ising (1925). The literature on this topic is too vast to mention; we refer the reader to any textbook in the area of statistical physics. The connection between Markov networks and these models in statistical physics is the origin of some of the terminology associated with these models, such as partition function or energy. In fact, many of the recent developments in inference for these models arise from approximations that were first proposed in the statistical physics community. Boltzmann machines were first proposed by Hinton and Sejnowski (1983).

Computer vision is another application domain that has motivated much of the work in undirected graphical models. The applications of MRFs to computer vision are too numerous to list; they span problems in low-level vision (such as image denoising, stereo reconstruction, or image segmentation) and in high-level vision (such as object recognition). Li (2001) provides a detailed description of some early applications; Szeliski et al. (2008) describe some applications that are viewed as standard benchmark problems for MRFs in the computer vision field.

4.9 Exercises

Exercise 4.1

Complete the analysis of example 4.4, showing that the distribution P defined in the example does not factorize over $\mathcal{H}$. (Hint: Use a proof by contradiction.)

Exercise 4.2

In this exercise, you will prove that the modified energy functions $\epsilon_1'(A, B)$ and $\epsilon_2'(B, C)$ of figure 4.10 result in precisely the same distribution as our original energy functions. More generally, for any constants λ^1 and λ^0, we can redefine

$$\begin{aligned} \epsilon_1'(a, b^i) &:= \epsilon_1(a, b^i) + \lambda^i \\ \epsilon_2'(b^i, c) &:= \epsilon_2(b^i, c) - \lambda^i \end{aligned}$$

Show that the resulting energy function is equivalent.

Exercise 4.3⋆

Provide an example a class of Markov networks $\mathcal{H}_n$ over n such that the size of the largest clique in $\mathcal{H}_n$ is constant, yet *any* Bayesian network I-map for $\mathcal{H}_n$ is exponentially large in n.

Exercise 4.4⋆

Prove theorem 4.7 for the case where $\mathcal{H}$ consists of a single clique.

Exercise 4.5

Complete the proof of theorem 4.3, by showing that U_1 and U_k are dependent given $\boldsymbol{Z}$ in the distribution P defined by the product of potentials described in the proof.

Exercise 4.6⋆

Consider a factor graph $\mathcal{F}$, as in definition 4.13. Define the minimal Markov network $\mathcal{H}$ that is guaranteed to be an I-map for any distribution defined over $\mathcal{F}$. Prove that $\mathcal{H}$ is a sound and complete representation of the independencies in $\mathcal{F}$:

a. If $sep_{\mathcal{H}}(\boldsymbol{X}; \boldsymbol{Y} \mid \boldsymbol{Z})$ holds, then $(\boldsymbol{X} \perp \boldsymbol{Y} \mid \boldsymbol{Z})$ holds for all distributions over $\mathcal{F}$.
b. If $sep_{\mathcal{H}}(\boldsymbol{X}; \boldsymbol{Y} \mid \boldsymbol{Z})$ does not hold, then there is some distribution P that factorizes over $\mathcal{F}$ such that $(\boldsymbol{X} \perp \boldsymbol{Y} \mid \boldsymbol{Z})$ does not hold in P.

Exercise 4.7⋆

The canonical parameterization in the Hammersley-Clifford theorem is stated in terms of the maximal cliques in a Markov network. In this exercise, you will show that it also captures the finer-grained representation of factor graphs. Specifically, let P be a distribution that factorizes over a factor graph $\mathcal{F}$ as in definition 4.13. Show that the canonical parameterization of P also factorizes over $\mathcal{F}$.

Exercise 4.8

Prove proposition 4.3. More precisely, let P satisfy $\mathcal{I}_\ell(\mathcal{H})$, and assume that X and Y are two nodes in $\mathcal{H}$ that are not connected directly by an edge. Prove that P satisfies $(X \perp Y \mid \mathcal{X} - \{X, Y\})$.

Exercise 4.9⋆

Complete the proof of theorem 4.4. Assume that equation (4.1) holds for all disjoint sets $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$, with $|\boldsymbol{Z}| \geq k$. Prove that equation (4.1) also holds for any disjoint $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$ such that $\boldsymbol{X} \cup \boldsymbol{Y} \cup \boldsymbol{Z} \neq \mathcal{X}$ and $|\boldsymbol{Z}| = k - 1$.

Exercise 4.10

We define the following properties for a set of independencies:

strong union

- **Strong Union**:

$$(\boldsymbol{X} \perp \boldsymbol{Y} \mid \boldsymbol{Z}) \Longrightarrow (\boldsymbol{X} \perp \boldsymbol{Y} \mid \boldsymbol{Z}, \boldsymbol{W}). \tag{4.12}$$

  In other words, additional evidence $\boldsymbol{W}$ cannot induce dependence.

transitivity

- **Transitivity**: For all disjoint sets $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$ and all variables A:

$$\neg(\boldsymbol{X} \perp A \mid \boldsymbol{Z}) \& \neg(A \perp \boldsymbol{Y} \mid \boldsymbol{Z}) \Longrightarrow \neg(\boldsymbol{X} \perp \boldsymbol{Y} \mid \boldsymbol{Z}). \tag{4.13}$$

  Intuitively, this statement asserts that if $\boldsymbol{X}$ and $\boldsymbol{Y}$ are both correlated with some A (given $\boldsymbol{Z}$), then they are also correlated with each other (given $\boldsymbol{Z}$). We can also write the contrapositive of this statement, which is less obvious but easier to read. For all $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}, A$:

$$(\boldsymbol{X} \perp \boldsymbol{Y} \mid \boldsymbol{Z}) \longrightarrow (\boldsymbol{X} \perp A \mid \boldsymbol{Z}) \vee (A \perp \boldsymbol{Y} \mid \boldsymbol{Z}).$$

Prove that if $\mathcal{I} = \mathcal{I}(\mathcal{H})$ for some Markov network $\mathcal{H}$, then $\mathcal{I}$ satisfies strong union and transitivity.

Exercise 4.11⋆

In this exercise you will prove theorem 4.6. Consider some specific node X, and let $\mathcal{U}$ be the set of all subsets $\boldsymbol{U}$ satisfying definition 4.12. Define $\boldsymbol{U}^*$ to be the intersection of all $\boldsymbol{U} \in \mathcal{U}$.

a. Prove that $\boldsymbol{U}^* \in \mathcal{U}$. Conclude that $\mathrm{MB}_P(X) = \boldsymbol{U}^*$.

b. Prove that if $P \models (X \perp Y \mid \mathcal{X} - \{X, Y\})$, then $Y \notin \mathrm{MB}_P(X)$.
c. Prove that if $Y \notin \mathrm{MB}_P(X)$, then $P \models (X \perp Y \mid \mathcal{X} - \{X, Y\})$.
d. Conclude that $\mathrm{MB}_P(X)$ is precisely the set of neighbors of X in the graph defined in theorem 4.5, showing that the construction of theorem 4.6 also produces a minimal I-map.

Exercise 4.12

Show that a Boltzmann machine distribution (with variables taking values in $\{0, 1\}$) can be rewritten as an Ising model, where we use the value space $\{-1, +1\}$ (mapping 0 to -1).

Exercise 4.13

Show that we can represent any Gibbs distribution as a log-linear model, as defined in definition 4.15.

Exercise 4.14

Complete the proof of proposition 4.8. In particular, show the following:

a. For any variable X, let $\boldsymbol{W} = \mathcal{X} - \{X\} - \mathrm{MB}_{\mathcal{G}}(X)$. Then $d\text{-}sep_{\mathcal{G}}(X; \boldsymbol{W} \mid \mathrm{MB}_{\mathcal{G}}(X))$.
b. The set $\mathrm{MB}_{\mathcal{G}}(X)$ is the minimal set for which this property holds.

Exercise 4.15

Prove proposition 4.10.

Exercise 4.16⋆

Provide an example of a class of Markov networks $\mathcal{H}_n$ over n nodes for arbitrarily large n (not necessarily for every n), where the size of the largest clique is a constant independent of n, yet the size of the largest clique in any chordal graph $\mathcal{H}_n^C$ that contains $\mathcal{H}_n$ is exponential in n. Explain why the size of the largest clique is necessarily exponential in n for all $\mathcal{H}_n^C$.

Exercise 4.17⋆

In this exercise, you will prove that the chordality requirement for graphs is equivalent to two other conditions of independent interest.

Definition 4.25 Markov network decomposition

Let $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$ be disjoint sets such that $\mathcal{X} = \boldsymbol{X} \cup \boldsymbol{Y} \cup \boldsymbol{Z}$ and $\boldsymbol{X}, \boldsymbol{Y} \neq \emptyset$. We say that $(\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{Y})$ is a decomposition *of a Markov network $\mathcal{H}$ if $\boldsymbol{Z}$ separates $\boldsymbol{X}$ from $\boldsymbol{Y}$ and $\boldsymbol{Z}$ is a complete subgraph in $\mathcal{H}$.* ■

Definition 4.26

We say that a graph $\mathcal{H}$ is decomposable *if there is a decomposition $(\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{Y})$ of $\mathcal{H}$, such that the graphs induced by $\boldsymbol{X} \cup \boldsymbol{Z}$ and $\boldsymbol{Y} \cup \boldsymbol{Z}$ are also decomposable.* ■

Show that, for any undirected graph $\mathcal{H}$, the following conditions are equivalent:

a. $\mathcal{H}$ is decomposable;
b. $\mathcal{H}$ is chordal;
c. for every X, Y, every minimal set $\boldsymbol{Z}$ that separates X and Y is complete.

The proof of equivalence proceeds by induction on the number of vertices in $\mathcal{X}$. Assume that the three conditions are equivalent for all graphs with $|\mathcal{X}| \leq n$, and consider a graph $\mathcal{H}$ with $|\mathcal{X}| = n + 1$.

a. Prove that if $\mathcal{H}$ is decomposable, it is chordal.
b. Prove that if $\mathcal{H}$ is chordal, then for any X, Y, and any minimal set $\boldsymbol{Z}$ that separates X and Y, $\boldsymbol{Z}$ is complete.
c. Prove that for any X, Y, any minimal set $\boldsymbol{Z}$ that separates X and Y is complete, then $\mathcal{H}$ is decomposable.

Exercise 4.18

Let $\mathcal{G}$ be a Bayesian network structure and $\mathcal{H}$ a Markov network structure over $\mathcal{X}$ such that the skeleton of $\mathcal{G}$ is precisely $\mathcal{H}$. Prove that if $\mathcal{G}$ has no immoralities, then $\mathcal{I}(\mathcal{G}) = \mathcal{I}(\mathcal{H})$.

Exercise 4.19

Consider the PDAG of figure 4.15. Write down all c-separation statements that are valid given $\{G\}$; write down all valid statements given $\{G, D\}$; write down all statements that are valid given $\{C, D, E\}$.