Chapter 10

Undirected Graphical Models

This chapter covers the following topics:

- Directed versus undirected graphical models.
- Global, local, and pairwise Markov properties.
- Log-linear models.
- Calculating marginal and conditional probabilities.
- Parameter estimation
- Learning the structured of undirected graphical model.

10.1 Introduction

Graphical models come in two basic varieties, corresponding to directed (acyclic) graphs and undirected graphs. In the previous chapter we discussed directed graphs as a representation of the independence relations for families of probability distributions. In the current chapter we turn to undirected graphical models.

Let X, Y and Z be three random variables. We write $X \perp\!\!\!\perp Y$ to mean that X and Y are independent. We write $X \perp\!\!\!\perp Y \mid Z$ to mean that X and Y are independent given Z. A single random vector drawn from a distribution P is denoted by $X = (X_1, \ldots, X_d)$. Besides directed graphs, another way to explore the structure of the distribution P for the random vector X is to estimate its undirected graph G, which consists of a set of vertices V and an edge set E of unordered pairs of vertices. Each vertex here corresponds to one random variable from $X_1, \ldots, X_d$. When we observe n random vectors, $x_1, \ldots, x_n$, drawn from P, we can write this in more detail as:

$$\begin{array}{ccccc} x_1, & x_2, & \ldots, & x_n & \text{i.i.d. drawn from } P \\ \| & \| & & \| & \end{array}$$

$$\begin{pmatrix} x_{11} \\ \vdots \\ x_{1d} \end{pmatrix} \quad \begin{pmatrix} x_{21} \\ \vdots \\ x_{2d} \end{pmatrix} \quad \vdots \quad \begin{pmatrix} x_{n1} \\ \vdots \\ x_{nd} \end{pmatrix}.$$

There are n independent observations with d features each. The nodes of the graph denote the d features but not the individual observations. For example, suppose we measure blood pressure, age and cholesterol on 100 people. Then $n = 100$ and $d = 3$. A possible graph for this example is provided in Figure 10.1. If we define $X = (X_1, X_2, X_3) =$

Figure 10.1: An undirected graph of three random variables: blood pressure, age, and cholesterol.

$(\text{blood pressure}, \text{age}, \text{cholesterol})$ then we can draw the graph as in 10.1. Keep in mind that the observations (random vectors) are independent, but the features (coordinates of each random vector) are not. The key idea of the undirected graph representation of probability distributions is this:

> The undirected graph G associated with P has d vertices corresponding to the components $X_1, \ldots, X_d$ of the random vector $X = (X_1, \ldots, X_d)$. We omit an edge between two nodes X_i and X_j if and only if X_i and X_j are conditionally independent given the other variables. This is called the Markov property encoded in the graph.

A simple example is shown in Figure 10.2. In this figure, there is a random vector $X = (X_1, X_2, X_3, X_4)$. Since there is no edge between X_1 and X_4 we conclude that X_1 and X_4 are independent given X_2 and X_3. Similarly, since there is no edge between X_2 and X_3 we conclude that X_2 and X_3 are independent given X_1 and X_4.

Undirected graphs are used as a natural representation of the dependencies between random variables in many applications. One example is image processing, where each pixel in an image may correspond to a random variable. Such lattice based models were first studied in statistical physics, where they were used to represent spin configurations of atoms in a crystal. Other applications where undirected graphical models arise naturally include error correcting codes, protein interaction networks, and social networks; for suggestive illustrations see Figures 10.3, 10.4, and 10.5.

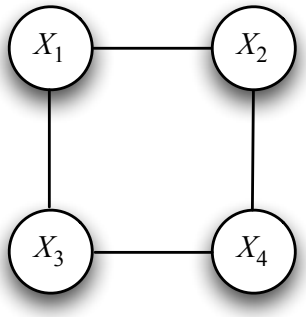


Figure 10.2: Graph for a random vector $X = (X_1, X_2, X_3, X_4)$. Since there is no edge between X_1 and X_4 we conclude that X_1 and X_4 are independent given X_2 and X_3. Similarly, since there is no edge between X_2 and X_3 we conclude that X_2 and X_3 are independent given X_1 and X_4.

The estimation and structure learning methods discussed in this chapter are based on parametric models. In later chapters we introduce nonparametric methods for estimating undirected graphs. In the case where each X_i is a continuous random variable, taking values in $\mathbb{R}$, perhaps the most popular and useful model is the multivariate Gaussian $X \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. As will be explained in the following sections, the graphical structure in this case is encoded in the inverse covariance matrix $\boldsymbol{\Omega} = \boldsymbol{\Sigma}^{-1}$; the edge between X_j and X_k is missing if and only if $\boldsymbol{\Omega}_{jk} = 0$. The case of discrete data, where each variable $X_i \in \{0, 1\}$ is binary, is much more difficult to work with. For this case, the analogue of the Gaussian graphical model is the Ising model.

10.2 Probability and Undirected Graphs

Let $G = (V, E)$ be an undirected graph with vertex set V and edge set E, and let A, B, and C be subsets of vertices. We say that C separates A and B if every path from a node in A to a node in B passes through a node in C. Now consider a random vector $X = (X_1, \ldots, X_d)$ where X_j corresponds to node j in the graph. If $A \subset \{1, \ldots, d\}$ then we write $X_A = (X_j : \ j \in A)$. In this section, we discuss the relationship between probability distributions and undirected graphs.

10.2.1 Markov Properties on Undirected Graphs

A probability distribution P for a random vector $X = (X_1, \ldots, X_d)$ may satisfy a range of different Markov properties with respect to a graph $G = (V, E)$:

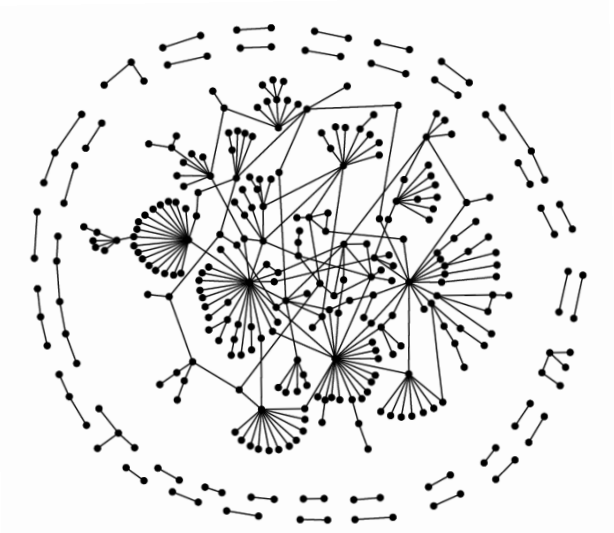


Figure 10.3: The graph depicts certain protein networks [59].

Definition 160. (Global Markov Property) A probability distribution P for a random vector $X = (X_1, \ldots, X_d)$ satisfies the global Markov property with respect to a graph G if for any disjoint vertex subsets A, B, and C such that C separates A and B, the random variables X_A are conditionally independent of X_B given X_C.

Definition 161. (Local Markov Property) A probability distribution P for a random vector $X = (X_1, \ldots, X_d)$ satisfies the local Markov property with respect to a graph G if the conditional distribution of a variable given its neighbors is independent of the remaining nodes. That is, let $N(s) = \{t \in V \,|\, (s,t) \in E\}$ denote the set of neighbors of a node $s \in V$. Then the local Markov property is that

$$p(x_s \,|\, x_t,\ t \neq s) = p\left(x_s \,|\, x_t, t \in N(s)\right) \tag{10.1}$$

for each node s.

Definition 162. (Pairwise Markov Property) A probability distribution P for a random vector $X = (X_1, \ldots, X_d)$ satisfies the pairwise Markov property with respect to a graph G if for any pair of non-adjacent nodes $s, t \in V$, we have

$$X_s \perp\!\!\!\perp X_t \,|\, X_{V \setminus \{s,t\}}. \tag{10.2}$$

Consider for example the graph in Figure 10.6. Here the set C separates A and B. Thus, a distribution that satisfies the global Markov property for this graph must have the property that the random variables in A are conditionally independent of the random variables in B given the random variables C. This is seen to generalize the usual Markov property for simple chains, where $X_A \longrightarrow X_C \longrightarrow X_B$ forms a Markov chain in case X_A and X_B are

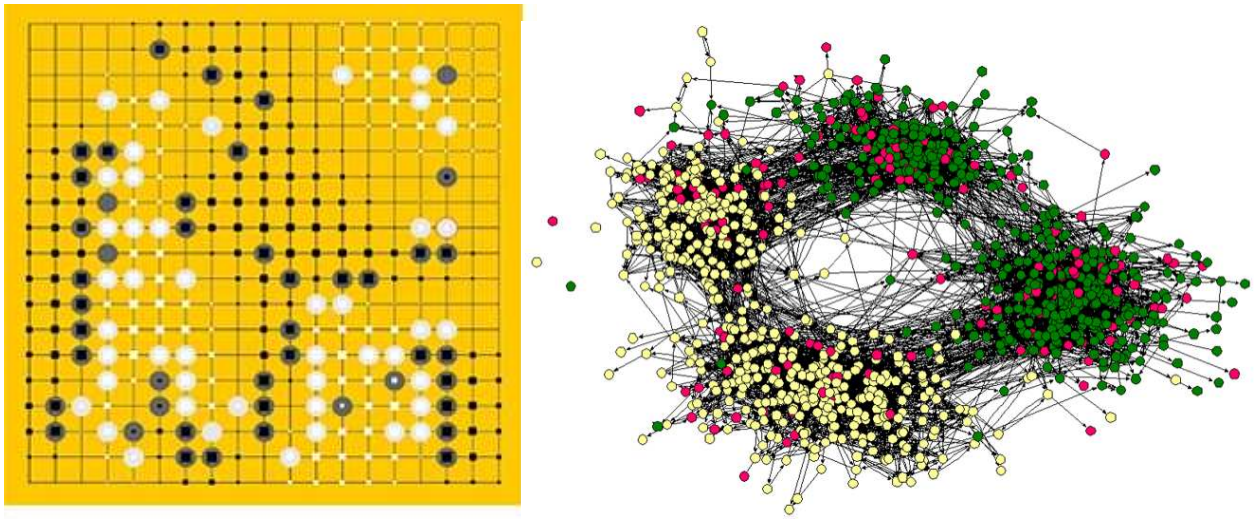


Figure 10.4: Left: The game of Go is modeled probabilistically using undirected graphical models in [82]. Here the graph is the grid of the game board, and each node takes on the binary value `black` or `white` depending on which player owns the position at the end of the game. Right: Graphs have been used for many years to represent social networks [64]. Recent work as renewed interest in probabilistic graphical models in social networks.

independent given X_C. A distribution that satisfies the global Markov property is said to be a Markov random field or Markov network with respect to the graph. The local Markov property is depicted in Figure 10.7.

From the definitions, the relationships of different Markov properties can be characterized as:

Proposition 163. For any undirected graph G and any distribution P, we have

$$\text{global Markov property} \implies \text{local Markov property} \implies \text{pairwise Markov property}.$$

Proof. The global Markov property implies the local Markov property because for each node $s \in V$, its neighborhood $N(s)$ separates $\{s\}$ and $V \setminus \{N(s) \cup \{s\}\}$. Assume next that the local Markov property holds. Any t that is not adjacent to s is an element of $t \in V \setminus \{N(s) \cup \{s\}\}$. Therefore

$$N(s) \cup [(V \setminus \{N(s) \cup \{s\}\}) \setminus \{t\}] = V \setminus \{s, t\}, \tag{10.3}$$

and it follows from the local Markov property that

$$X_s \perp\!\!\!\perp X_{V \setminus \{N(s) \cup \{s\}\}} \,|\, X_{V \setminus \{s,t\}}. \tag{10.4}$$

This implies $X_s \perp\!\!\!\perp X_t \,|\, X_{V \setminus \{s,t\}}$, which is the pairwise Markov property. □

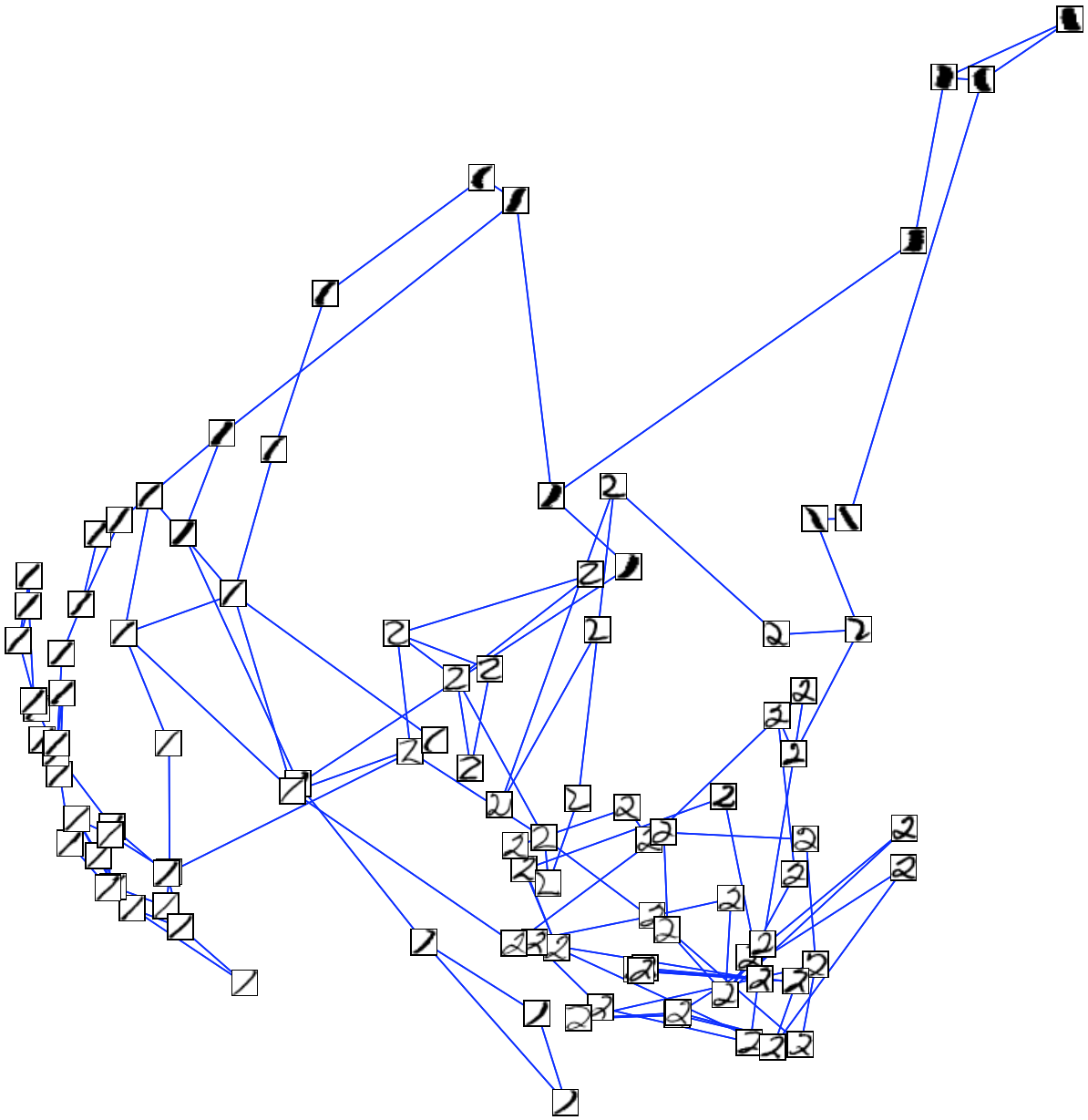


Figure 10.5: Undirected graphs have been used in semi-supervised learning; the plot on the right shows a subset of scanned digits of ones and twos, with edges constructed in terms of a nearest neighbor metric, projected into two dimensions using PCA (courtesy of Jerry Zhu).

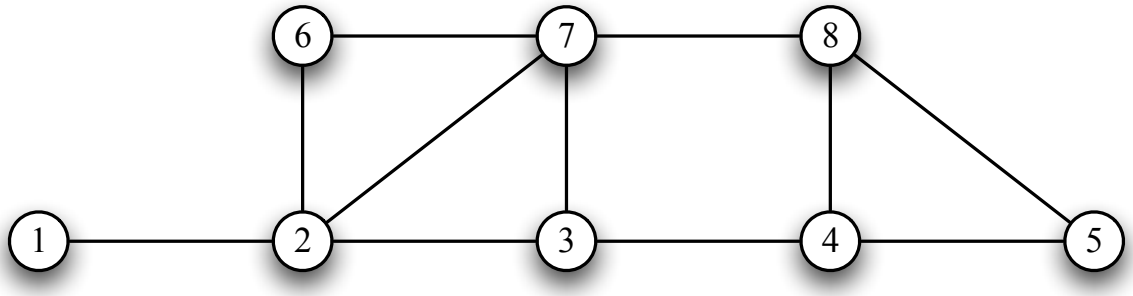


Figure 10.6: An undirected graph. $C = \{3, 7\}$ separates $A = \{1, 2\}$ and $B = \{4, 8\}$.

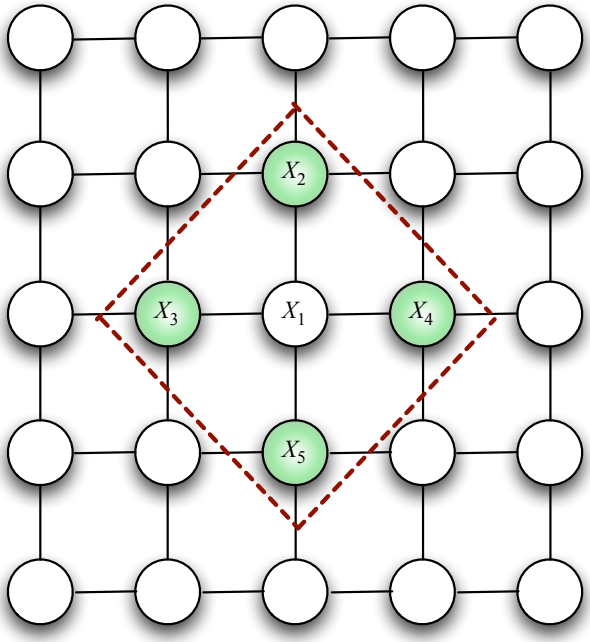


Figure 10.7: The local Markov property: Conditioned on its four neighbors X_2, X_3, X_4, and X_5, node X_1 is independent of the remaining nodes in the graph.

In general, the global, local, and pairwise Markov properties are different, as illustrated by the following examples 164 and 165. However, if the distributions have positive continuous densities, all three Markov properties are equivalent.

Example 164. Define the joint distribution of five binary random variables U, W, X, Y, Z as follows: U and Z are independent with

$$\mathbb{P}(U = 1) = \mathbb{P}(Z = 1) = 1/2, \tag{10.5}$$

$W = U$, $Y = Z$, and $X = WY$. The joint distribution so defined is easily seen to satisfy the local Markov property but not the global Markov property for the chain graph U—W—X—Y—Z.

Example 165. Let $X = Y = Z$ be three binary random variables with $\mathbb{P}(X = 1) = 1/2$. We consider a graph G which only contains one edge that connects Y and Z. It is easy to see that this distribution satisfies the pairwise Markov property with respect to the graph G but it does not satisfy the local Markov property with respect to G, Since X is not independent of Y.

The next theorem, due to [69], provides a sufficient condition for equivalence.

Theorem 166. [69] If it holds that for all disjoint subsets $A, B, C, D \subset V$, we have

$$\text{if } X_A \perp\!\!\!\perp X_B \,|\, X_{C\cup D} \text{ and } X_A \perp\!\!\!\perp X_C \,|\, X_{B\cup D}, \text{ then } X_A \perp\!\!\!\perp X_{B\cup C} \,|\, X_D, \tag{10.6}$$

then the global, local, and pairwise Markov properties are equivalent.

Proof. It is enough to show that the pairwise Markov property implies the global Markov property under the given condition. Let $S, A, B \subset V$ witn S separating A from B in the graph G. Without loss of generality both A and B are assumed to be non-empty. The proof can be carried out using backward induction on the number of nodes in S, denoted by $m = |S|$. Let $d = |V|$, for the base case, if $m = d - 1$ then both A and B only consist of single vertex and the result follows from pairwise Markov property.

Now assume that $m < d - 1$ and separation implies conditional independence for all separating sets S with more than m nodes. We proceed in two cases: (i) $A \cup B \cup S = V$ and (ii) $A \cup B \cup S \subset V$.

For case (i), we know that at least one of A and B must have more than one element. Without loss of generality, we assume A has more than one element. If $s \in A$, then $S \cup \{s\}$ separates $A \setminus \{s\}$ from B and also $S \cup (A \setminus \{s\})$ separates s from B. Thus by the induction hypothesis

$$X_{A\setminus\{s\}} \perp\!\!\!\perp X_B \,|\, X_{S\cup\{s\}} \text{ and } X_s \perp\!\!\!\perp X_B \,|\, S \cup (A \setminus \{s\}). \tag{10.7}$$

Now the condition (10.6) implies $X_A \perp\!\!\!\perp X_B \,|\, X_S$. For case (ii), we could choose $s \in V \setminus (A \cup B \cup S)$. Then $S \cup \{s\}$ separates A and B, implying $A \perp\!\!\!\perp B \,|\, S \cup \{s\}$. We then proceed in two cases, either $A \cup S$ separates B from s or $B \cup S$ separates A from s. For both cases, the condition (10.6) implies that $A \perp\!\!\!\perp B \,|\, S$. □

The next proposition provides a stronger condition that implies (10.6).

Proposition 167. Let $X = (X_1, \ldots, X_d)$ be a random vector with distribution P and joint density $p(x)$. If the joint density $p(x)$ is positive and continuous with respect to a product measure, then condition (10.6) holds.

Proof. Without loss of generality, it suffices to assume that $d = 3$. We want to show that

$$\text{if } X_1 \perp\!\!\!\perp X_2 \,|\, X_3 \text{ and } X_1 \perp\!\!\!\perp X_3 \,|\, X_2 \text{ then } X_1 \perp\!\!\!\perp \{X_2, X_3\}. \tag{10.8}$$

Since the density is positive and $X_1 \perp\!\!\!\perp X_2 \,|\, X_3$ and $X_1 \perp\!\!\!\perp X_3 \,|\, X_2$, we know that there must exist some positive functions $f_{13}, f_{23}, g_{12}, g_{23}$ such that the joint density takes the following factorization:

$$p(x_1, x_2, x_3) = f_{13}(x_1, x_3) f_{23}(x_2, x_3) = g_{12}(x_1, x_2) g_{23}(x_2, x_3). \tag{10.9}$$

Since the density is continuous and positive, we have

$$g_{12}(x_1, x_2) = \frac{f_{13}(x_1, x_3) f_{23}(x_2, x_3)}{g_{23}(x_2, x_3)}. \tag{10.10}$$

For each fixed $X_3 = x_3'$, we see that $g_{12}(x_1, x_2) = h(x_1)\ell(x_2)$ where $h(x_1) = f_{13}(x_1, x_3')$ and $\ell(x_2) = f_{23}(x_2, x_3')/g_{23}(x_2, x_3')$. This implies that

$$p(x_1, x_2, x_3) = h(x_1)\ell(x_2) g_{23}(x_2, x_3) \tag{10.11}$$

and hence $X_1 \perp\!\!\!\perp \{X_2, X_3\}$ as desired. □

From Proposition 167, we see that for distributions with positive continuous densities, the global, local, and pairwise Markov properties are all equivalent. If a distribution P satisfies global Markov property with respect to a graph G, we say that P is Markov to G

10.2.2 Clique Decomposition

Unlike a directed graph which encodes a factorization of the joint probability distribution in terms of conditional probability distributions. An undirected graph encodes a factorization of the joint probability distribution in terms of clique potentials. Recall that a clique in a graph is a fully connected subset of vertices. Thus, every pair of nodes in a clique

is connected by an edge. A clique is a maximal clique if it is not contained in any larger clique. Consider, for example, the graph shown in the right plot of Figure 10.8. The pairs $\{X_4, X_5\}$ and $\{X_1, X_3\}$ form cliques; $\{X_4, X_5\}$ is a maximal clique, while $\{X_1, X_3\}$ is not maximal since it is contained in a larger clique $\{X_1, X_2, X_3\}$.

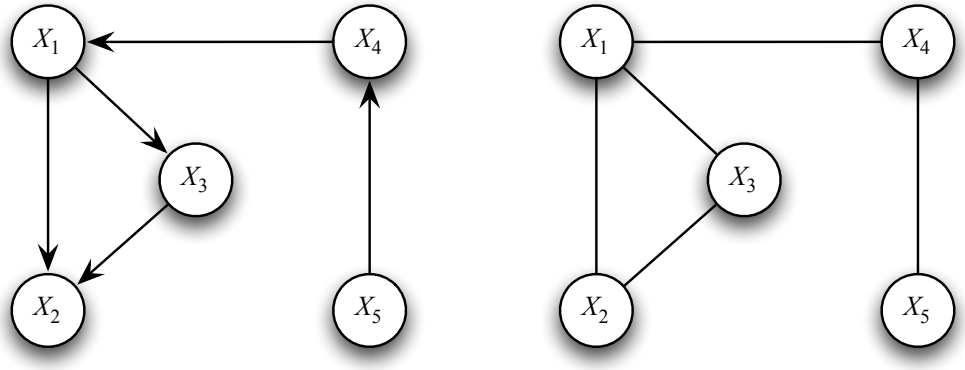


Figure 10.8: A directed graph encodes a factorization of the joint probability distribution in terms of conditional probability distributions. An undirected graph encodes a factorization of the joint probability distribution in terms of clique potentials.

Let $\mathcal{C}$ be the set of all maximal cliques in a graph. A probability distribution *factors with respect to this graph* in case it can be written as a product of factors, one for each of the maximal cliques in the graph:

$$p(x_1, \ldots, x_{|V|}) = \prod_{C \in \mathcal{C}} \psi_C(x_C). \tag{10.12}$$

Similarly, a set of clique potentials $\{\psi_C(x_C) \geq 0\}_{C \in \mathcal{C}}$ determines a probability distribution that factors with respect to the graph by normalizing:

$$p(x_1, \ldots, x_{|V|}) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(x_C). \tag{10.13}$$

The normalizing constant or partition function Z sums (or integrates) over all settings of the random variables:

$$Z = \int_{x_1, \ldots, x_{|V|}} \prod_{C \in \mathcal{C}} \psi_C(x_C) dx_1 \ldots dx_{|V|}. \tag{10.14}$$

Thus, the family of distributions represented by the undirected graph in Figure 10.8 can be written as

$$p(x_1, x_2, x_3, x_4, x_5) = \psi_{1,2,3}(x_1, x_2, x_3)\, \psi_{1,4}(x_1, x_4)\, \psi_{4,5}(x_4, x_5). \tag{10.15}$$

In contrast, the family of distributions represented by the directed graph in Figure 10.8 can be factored into conditional distributions according to

$$p(x_1, x_2, x_3, x_4, x_5) = p(x_5)\, p(x_4 \,|\, x_5)\, p(x_1 \,|\, x_4)\, p(x_3 \,|\, x_1)\, p(x_2 \,|\, x_1, x_3). \tag{10.16}$$

Theorem 168. For any undirected graph $G = (V, E)$, a distribution P that factors with respect to the graph also satisfies the global Markov property on the graph.

Proof. Let $A, B, S \subset V$ such that S separates A and B. We want to show $X_A \perp\!\!\!\perp X_B \,|\, X_S$. For a subset $D \subset V$, we denote G_D to be the subgraph induced by the vertex set D. We define $ldeA$ to be the connectivity components in $G_{V \setminus S}$ which contain A and $ldeB = V \setminus (ldeA \cup S)$. Since A and B are separated by S, they must belong to different connectivity components of $G_{V \setminus S}$ and any clique of G must be a subset of either $ldeA \cup S$ or $ldeB \cup S$. Let $\mathcal{C}_A$ be the set of cliques contained in $ldeA \cup S$, the joint density $p(x)$ takes the following factorization

$$p(x) = \prod_{C \in \mathcal{C}} \psi_C(x_C) = \prod_{C \in \mathcal{C}_A} \psi_C(x_C) \prod_{C \in \mathcal{C} \setminus \mathcal{C}_A} \psi_C(x_C). \tag{10.17}$$

This implies that $ldeA \perp\!\!\!\perp ldeB \,|\, S$ and thus $A \perp\!\!\!\perp B \,|\, S$. □

It is worth remembering that while we think of the set of maximal cliques as given in a list, the problem of enumerating the set of maximal cliques in a graph is NP-hard, and the problem of determining the largest maximal clique is NP-complete [2, 12]. However, many graphs of interest in statistical analysis are sparse, with the number of cliques of size $O(|V|)$.

Theorem 168 shows that factoring with respect to a graph implies global Markov property. The next question is, under what conditions the Markov properties imiply factoring with respect to a graph. In fact, in the case where P has a positive and continuous density we can show that the pairwise Markov property implies factoring with respect to a graph. Thus all Markov properties are equivalent. The results have been discovered by many authors but is usually referred to as Hammersley and Clifford due to one of their unpublished manuscript in 1971. They proved the result in the discrete case. The following result is usually referred to as the Hammersley-Clifford theorem, a proof appears in [10]. The extension to the continuous case is left as an exercise (See Exercise 4).

Theorem 169 (Hammersley-Clifford-Besag)**.** Suppose that $G = (V, E)$ is a graph and X_i, $i \in V$ are random variables that take on a finite number of values. If $\mathbb{P}(x) > 0$ is strictly positive and satisfies the local Markov property with respect to G, then it factors with respect to G.

Proof. Let $d = |V|$. By re-indexing the values of X_i, we may assume without loss of generality that each X_i takes on the value 0 with positive probability, and $\mathbb{P}(0, 0, \ldots, 0) > 0$. Let $X_{0\setminus i}$ denote the vector $X_{0\setminus i} = (X_1, X_2, \ldots, X_{i-1}, 0, X_{i+1}, \ldots, X_d)$ obtained by setting $X_i = 0$, and let $X_{\setminus i} = (X_1, X_2, \ldots, X_{i-1}, X_{i+1}, \ldots, X_d)$ denote the vector of all components except X_i. Then

$$\frac{\mathbb{P}(x)}{\mathbb{P}(x_{i\setminus 0})} = \frac{\mathbb{P}(x_i \,|\, x_{\setminus i})}{\mathbb{P}(0 \,|\, x_{\setminus i})}. \tag{10.18}$$

Now, let

$$Q(x) = \log\left(\frac{\mathbb{P}(x)}{\mathbb{P}(0)}\right). \tag{10.19}$$

Then for any $i \in \{1, 2, \ldots, d\}$ we have that

$$\begin{aligned} Q(x) &= \log\left(\frac{\mathbb{P}(x)}{\mathbb{P}(0)}\right) && (10.20)\\ &= \log\left(\frac{\mathbb{P}(0, \ldots, x_i, 0, \ldots, 0)}{\mathbb{P}(0)}\right) + \log\left(\frac{\mathbb{P}(x)}{\mathbb{P}(0, \ldots, x_i, 0, \ldots, 0)}\right) && (10.21)\\ &= \frac{1}{d}\sum_{i=1}^{d}\left\{\log\left(\frac{\mathbb{P}(0, \ldots, x_i, 0, \ldots, 0)}{\mathbb{P}(0)}\right) + \log\left(\frac{\mathbb{P}(x)}{\mathbb{P}(0, \ldots, x_i, 0, \ldots, 0)}\right)\right\} && (10.22) \end{aligned}$$

Recursively, we obtain

$$Q(x) = \sum_i \phi_i(x_i) + \sum_{i<j} \phi_{ij}(x_i, x_j) + \sum_{i<j<k} \phi_{ijk}(x_i, x_j, x_k) + \cdots + \phi_{12\cdots d}(x)$$

for functions ϕ_A that satisfy $\phi_A(x_A) = 0$ if $i \in A$ and $x_i = 0$. Consider node $i = 1$, we have

$$\begin{aligned} Q(x) - Q(x_{0\setminus i}) &= \log\left(\frac{\mathbb{P}(x_i \,|\, x_{\setminus i})}{\mathbb{P}(0 \,|\, x_{\setminus i})}\right) && (10.23)\\ &= \phi_1(x_1) + \sum_{i>1} \phi_{1i}(x_1, x_i) + \sum_{j>i>1} \phi_{1ij}(x_1, x_i, x_j) + \cdots + \phi_{12\cdots d}(x) \end{aligned}$$

depends only on x_1 and the neighbors of node 1 in the graph. Thus, from the local Markov property, if k is not a neighbor of node 1, then the above expression does not depend of x_k. In particular, $\phi_{1k}(x_1, x_k) = 0$, and more generally all $\phi_A(x_A)$ with $1 \in A$ and $k \in A$ are identically zero. Similarly, if i, j are not neighbors in the graph, then $\phi_A(x_A) = 0$ for any A containing i and j. Thus, $\phi_A \neq 0$ only holds for the subsets A that form cliques in the graph. Since it is obvious that $\exp(\phi_A(x)) > 0$, we finish the proof. □

Since factoring with respect to the graph implies the global Markov property, we may summarize this result as follows:

For positive distributions, global Markov $\Leftrightarrow$ local Markov $\Leftrightarrow$ factored

For strictly positive distributions, the global Markov property, the local Markov property, and factoring with respect to the graph are equivalent.

In the following, if a probability distribution factors with respect to a graph, we call it Gibbs distribution.

Definition 170. (Gibbs Distribution) A probability distribution P (with density $p(x)$) on an undirected graph G is called a Gibbs distribution if it can be factorized into positive functions defined on cliques that cover all the nodes and edges of G. That is,

$$p(x) = \frac{1}{Z}\prod_{C\in\mathcal{C}}\psi_C(x_C) = \frac{1}{Z}\exp\left(\sum_{C\in\mathcal{C}}\log\psi_C(x_C)\right) \tag{10.24}$$

where $\mathcal{C}$ is the set of all (maximal) cliques in G and Z is the normalization constant.

10.3 Directed vs. Undirected Graphs

Directed graphical models are naturally viewed as generative; the graph specifies a straightforward (in principle) procedure for sampling from the underlying distribution. For instance, a sample from a distribution represented from the DAG in left plot of Figure 10.9 can be sampled as follows:

$$\begin{aligned}
X_1 &\sim P(X_1) && (10.25)\\
X_2 &\sim P(X_2) && (10.26)\\
X_3 &\sim P(X_3) && (10.27)\\
X_5 &\sim P(X_5) && (10.28)\\
X_4 \,|\, X_1, X_2 &\sim P(X_4 \,|\, X_1, X_2) && (10.29)\\
X_6 \,|\, X_3, X_4, X_5 &\sim P(X_6 \,|\, X_3, X_4, X_5). && (10.30)
\end{aligned}$$

As long as each of the conditional probability distributions can be efficiently sampled, the full model can be efficiently sampled as well. In contrast, there is no straightforward way to sample from an distribution from the family specified by an undirected graph. We will return to this when we discuss simulation.

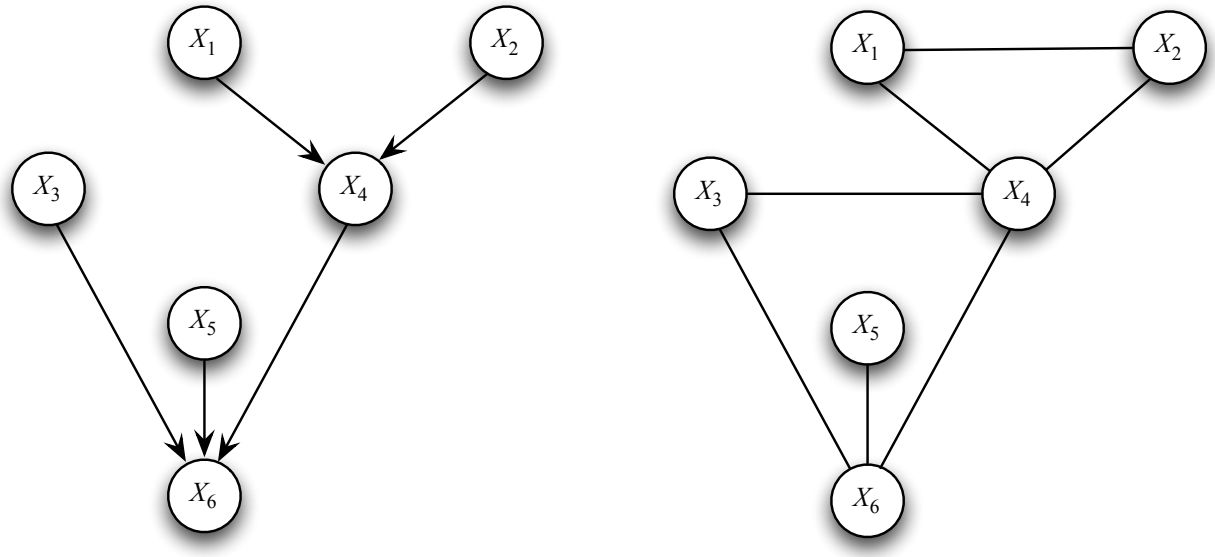


Figure 10.9: A DAG and its corresponding moral graph. A probability distribution that factors according to a DAG obeys the global Markov property on the undirected moral graph.

10.3.1 Converting Directed Graphs to Undirected Graphs

In certain cases, a Bayesian network (or DAGs) can also be written as a Gibbs distribution, note that the distribution in (10.16) also takes the form in (10.15) with

$$\psi_{1,2,3}(x_1, x_2, x_3) \equiv p(x_2 \,|\, x_1, x_3)\, p(x_3 \,|\, x_1) \tag{10.31}$$
$$\psi_{1,4}(x_1, x_4) \equiv p(x_1 \,|\, x_4) \tag{10.32}$$
$$\psi_{4,5}(x_4, x_5) \equiv p(x_4 \,|\, x_5)\, p(x_5). \tag{10.33}$$

However, more generally edges must be added to the skeleton of a DAG in order for the distribution to satisfy the global Markov property on the graph. Consider the example in Figure 10.9. Here the directed model has a distribution

$$p(x_1)\, p(x_2)\, p(x_3)\, p(x_5)\, p(x_4 \,|\, x_1, x_2)\, p(x_6 \,|\, x_3, x_4, x_5). \tag{10.34}$$

The corresponding undirected graphical model has two maximal cliques, and factors as

$$\psi_{1,2,4}(x_1, x_2, x_4)\, \psi_{3,4,5,6}(x_3, x_4, x_5, x_6). \tag{10.35}$$

More generally, let P be a probability distribution that is Markov to a DAG G. We define the moralized graph of G as the following:

Definition 171. (Moral graph) The moral graph M of a DAG G is an undirected graph that contains an undirected edge between two nodes X_i and X_j if (i) there is a directed edge between X_i and X_j in G, or (ii) X_i and X_j are both parents of the same node.

Theorem 172. If a probability distribution factors with respected to a DAG G, then it obeys the global Markov property with respect to the undirected moral graph of G.

Proof. Directly follows from the definition of Bayesian networks and Theorem 168. □

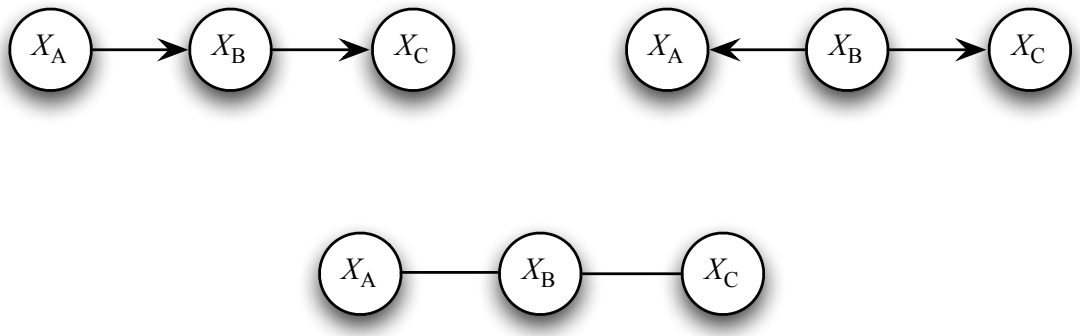


Figure 10.10: These three graphs encode distributions with identical independence relations. Conditioned on variable X_C, the variables X_A and X_B are independent; thus C separates A and B in the undirected graph.

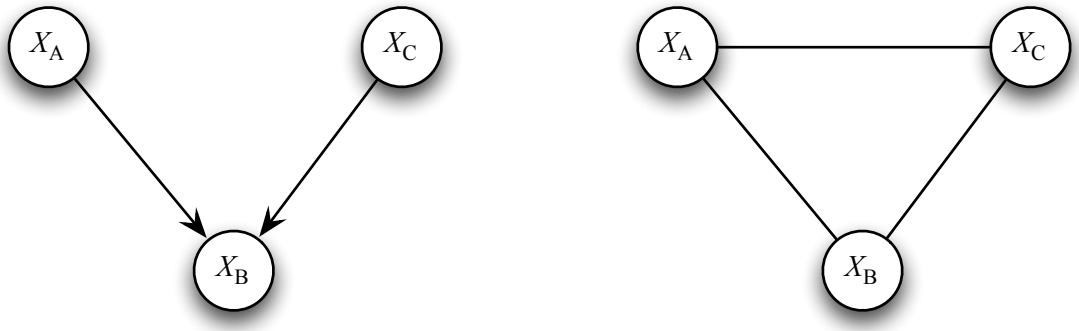


Figure 10.11: A directed graph whose conditional independence properties can not be perfectly expressed by its undirected moral graph. In the directed graph, the node C is a collider; therefore, X_A and X_B are not independent conditioned on X_C. In the corresponding moral graph, A and B are not separated by C. However, in the directed graph, we have the independence relationship $X_A \perp\!\!\!\perp X_B$, which is missing in the moral graph.

Example 173. (Basic Directed and Undirected Graphs) To illustrate some basic cases, consider the graphs in Figure 10.10. Each of the top three graphs encodes the same family of probability distributions. In the two directed graphs, by d-separation the variables X_A and X_B are independent conditioned on the variable X_C. In the corresponding undirected graph, which simply removes the arrows, node C separates A and B.

The two graphs in Figure 10.11 provide an example of a directed graph which encodes a set of conditional independence relationships that can not be perfectly represented by the

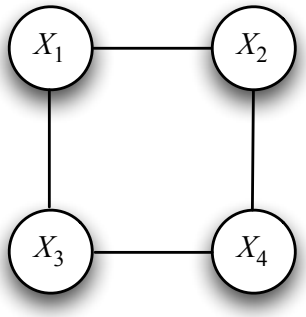


Figure 10.12: This undirected graph encodes a family of distributions that cannot be represented by a directed graph on the same set of nodes.

corresponding moral graph. In this case, for the directed graph the node C is a collider, and deriving an equivalent undirected graph requires joining the parents by an edge. In the corresponding undirected graph, A and B are not separated by C. However, in the directed graph, X_A and X_B are marginally independent, such an independence relationship is lost in the moral graph. Conversely, Figure 10.12 provides an undirected graph over four variables. There is no directed graph over four variables that implies the same set of conditional independence properties.

The upper plot in Figure 10.13 shows the directed graph underlying a hidden Markov model. There are no colliders in this graph, and therefore the undirected skeleton represents an equivalent set of independence relations. Thus, hidden Markov models are equivalent to hidden Markov fields with an underlying tree graph.

10.4 Faithfulness Revisited

The set of all distributions that are Markov to a graph G is denoted by $\mathcal{P}(G)$ [1]. To understand $\mathcal{P}(G)$ more clearly, we introduce some more notation. Given a distribution P let $\mathcal{I}(P)$ denote all conditional independence statements that are true for P. For example, if P has density p and $p(x_1, x_2, x_3) = p(x_1)p(x_2)p(x_3|x_1, x_2)$ then $\mathcal{I}(P) = \{X_1 \perp\!\!\!\perp X_2\}$. On the other hand, if $p(x_1, x_2, x_3) = p(x_1)p(x_2, x_3)$ then

$$\mathcal{I}(P) = \Big\{X_1 \perp\!\!\!\perp X_2,\ X_1 \perp\!\!\!\perp X_3,\ X_1 \perp\!\!\!\perp X_2|X_3,\ X_1 \perp\!\!\!\perp X_3|X_2\Big\}.$$

Similarly, given a graph G let $\mathcal{I}(G)$ denote all independence statements implied by the graph. For example, if G is the graph in Figure 10.2, then

$$\mathcal{I}(G) = \Big\{X_1 \perp\!\!\!\perp X_4 |\ \{X_2, X_3\}\,,\quad X_2 \perp\!\!\!\perp X_3 |\ \{X_1, X_4\}\Big\}.$$

[1] This section may be quite confusing at a first reading. We suggest you skim it and return to it later for a second reading.

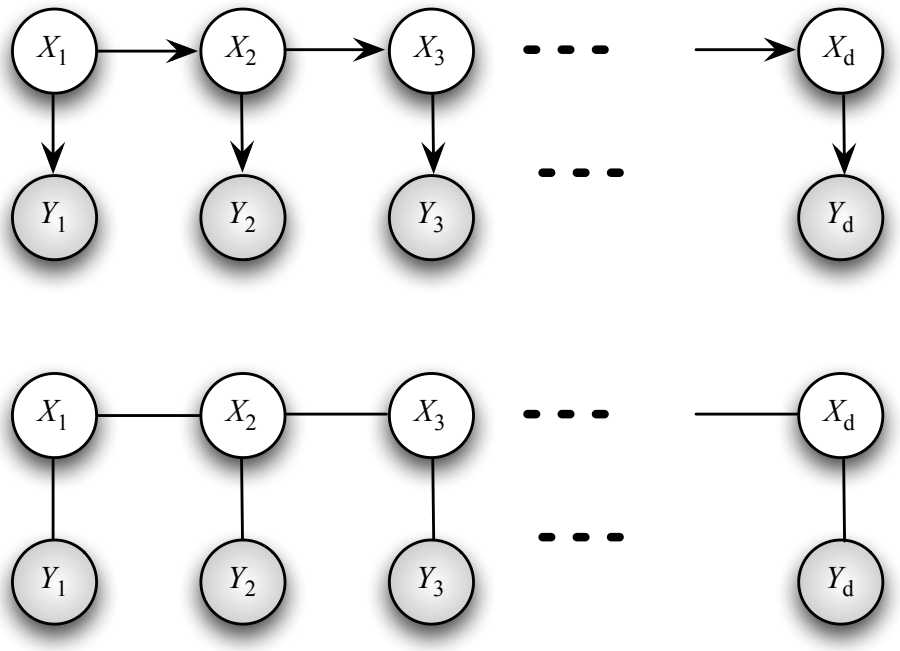


Figure 10.13: The top graph is a directed graph representing a hidden Markov model. The shaded nodes are observed, but the unshaded nodes, representing states in a latent Markov chain, are unobserved. Replacing the directed edges by undirected edges (bottom) does not changed the independence relations.

From definition, we could write $\mathcal{P}(G)$ as

$$\mathcal{P}(G) = \Big\{P : \ \mathcal{I}(G) \subseteq \mathcal{I}(P)\Big\}. \tag{10.36}$$

This result often leads to confusion since you might have expected that $\mathcal{P}(G)$ would be equal to $\big\{P : \ \mathcal{I}(G) = \mathcal{I}(P)\big\}$. But this is incorrect. For example, consider the undirected graph X_1 — X_2, in this case, $\mathcal{I}(G) = \emptyset$ and $\mathcal{P}(G)$ consists of all distributions $p(x_1, x_2)$. Since, $\mathcal{P}(G)$ consists of all distributions, it also includes distributions of the form $p_0(x_1, x_2) = p_0(x_1)p_0(x_2)$. For such a distribution we have $\mathcal{I}(P_0) = \{X_1 \perp\!\!\!\perp X_2\}$. Hence, $\mathcal{I}(G)$ is a strict subset of $I(P_0)$.

In fact, you can think of $\mathcal{I}(G)$ as the set of independence statements that are common to all $P \in \mathcal{P}(G)$. In other words,

$$\mathcal{I}(G) = \bigcap \Big\{\mathcal{I}(P) : \ P \in \mathcal{P}(G)\Big\}. \tag{10.37}$$

Every $P \in \mathcal{P}(G)$ has the independence properties in $\mathcal{I}(G)$. But some P's in $\mathcal{P}(G)$ might have extra independence properties.

We say that P is faithful to G if $\mathcal{I}(P) = \mathcal{I}(G)$. We define

$$\mathcal{F}(G) = \Big\{P : \ \mathcal{I}(G) = \mathcal{I}(P)\Big\} \tag{10.38}$$

and we note that $\mathcal{F}(G) \subset \mathcal{P}(G)$. A distribution P that is in $\mathcal{P}(G)$ but is not in $\mathcal{F}(G)$ is said to be unfaithful with respect to G. The independence relation expressed by G are correct for such a P. It's just that P has extra independence relations not expressed by G. A distribution P is also Markov to some graph. For example, any distribution is Markov to the complete graph. But there exist distributions P that are not faithful to any graph. This means that there will be some independence relations of P that cannot be expressed using a graph.

Example 174. The directed graph in Figure 10.11 implies that $X_A \perp\!\!\!\perp X_B$ but that X_A and X_B are not independent given X_C. There is no undirected graph G for (X_A, X_B, X_C) such that $\mathcal{I}(G)$ contains $X_A \perp\!\!\!\perp X_B$ but excludes $X_A \perp\!\!\!\perp X_B \,|\, X_C$. The only way to represent P is to use the complete graph. Then P is Markov to G since $\mathcal{I}(G) = \emptyset \subset \mathcal{I}(P) = \{X_A \perp\!\!\!\perp X_B\}$ but P is unfaithful to G since it has an independence relation not represented by G, namely, $\{X_A \perp\!\!\!\perp X_B\}$.

Example 175 (Unfaithful Gaussian distribution)**.** . Let $\xi_1, \xi_2, \xi_3 \sim N(0,1)$ be independent.

$$X_1 = \xi_1 \tag{10.39}$$
$$X_2 = aX_1 + \xi_2 \tag{10.40}$$
$$X_3 = bX_2 + cX_1 + \xi_3 \tag{10.41}$$

where a, b, c are nonzero. See Figure 10.14. Now suppose that $c = -\frac{b(a^2+1)}{a}$. Then

$$\begin{aligned}
\mathrm{Cov}\,(X_2, X_3) &= \mathbb{E}(X_2X_3) - \mathbb{E}X_2\mathbb{E}X_3 && (10.42)\\
&= \mathbb{E}(X_2X_3) && (10.43)\\
&= \mathbb{E}\left[(aX_1 + \xi_2)(bX_2 + cX_1 + \xi_3)\right] && (10.44)\\
&= \mathbb{E}\left[(a\xi_1 + \xi_2)(b(a\xi_1 + \xi_2) + cX_1 + \xi_3)\right] && (10.45)\\
&= (a^2b + ac)\mathbb{E}\xi_1^2 + b\mathbb{E}\xi_2^2. && (10.46)\\
&= a^2b + ac + b = 0. && (10.47)
\end{aligned}$$

Thus, we know that $X_2 \perp\!\!\!\perp X_3$. We would like to drop the edge between X_2 and X_3. But this would imply that $X_2 \perp\!\!\!\perp X_3 \,|\, X_1$ which is not true.

Generally, the set of unfaithful distributions $\mathcal{P}(G) \setminus \mathcal{F}(G)$ is a small set. In fact, it has Lebesgue measure zero if we restrict ourselves to nice parametric families. However, these unfaithful distributions are scattered thoughout $\mathcal{P}(G)$ in a complex way; see Figure 10.15.

10.5 Exponential Family Representation

By the Hammersley-Clifford-Besag theorem, we see that the essential property of undirected graphical models is the exponential family representation. In particular, a strictly

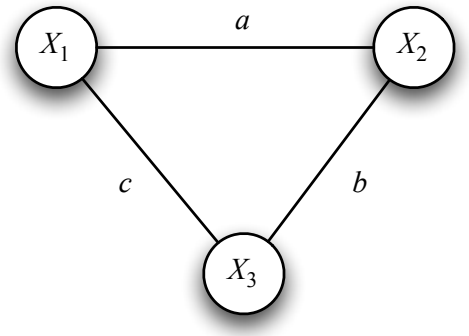


Figure 10.14: An unfaithful Gaussian distribution.

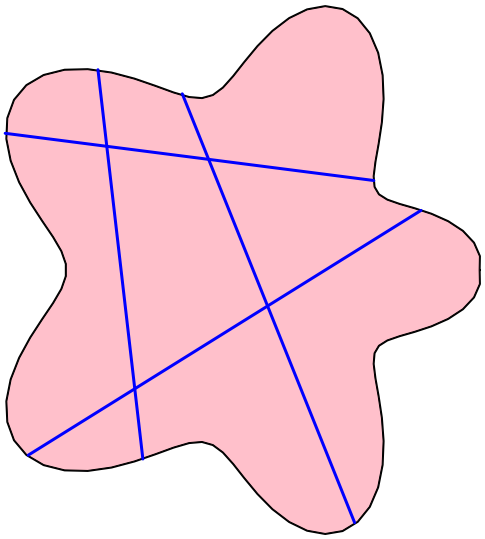


Figure 10.15: The blob represents the set $\mathcal{P}(G)$ of distributions that are Markov to some graph G. The lines are a stylized representation of the members of $\mathcal{P}(G)$ that are not faithful to G. Hence the lines represent the set $\mathcal{P}(G) \setminus \mathcal{F}(G)$. These distributions have extra independence relations not captured by the graph G. The set $\mathcal{P}(G) \setminus \mathcal{F}(G)$ is small but these distributions are scattered throughout $\mathcal{P}(G)$.

positive distribution that is locally Markov with respect to a graph can be represented as

$$p(x) = \frac{1}{Z(f)} \exp\left(\sum_{C \in \mathcal{C}} f_C(x_C)\right) \tag{10.48}$$

where the sum is over all maximal cliques in the graph, and

$$Z(f) = \sum_x \exp\left(\sum_{C \in \mathcal{C}} f_C(x_C)\right) \quad \text{or} \quad Z(f) = \int \exp\left(\sum_{C \in \mathcal{C}} f_C(x_C)\right) dx \tag{10.49}$$

in either discrete or continuous settings.

A sufficient condition for positivity is that the potential functions f_C satisfy

$$\inf_x \min_{C \in \mathcal{C}} f_C(x_C) > -\infty. \tag{10.50}$$

Thus, the Hammersley-Clifford-Besag theorem characterizes Markov fields in terms of arbitrary functions of the local configuration in the graph.

This family is often restricted to a linear functions of a set of pre-given feature vector $\{f_C(x_C)\}_{C \in \mathcal{C}}$:

$$p(x; \theta) = \frac{1}{Z(\theta)} \exp\left(\sum_{C \in \mathcal{C}} \theta_C f_C(x_C)\right) \tag{10.51}$$

where $f(x) = (f_C(x_C) : \ C \in \mathcal{C})$ is a vector of sufficient statistics and $\theta = (\theta_C : \ C \in \mathcal{C})$ is the parameter vector. Curved exponential families can also be employed, but these are used less often in practice. Note the difference between (10.48) and (10.51); in the latter, the function $f_C(x_C)$ are fixed, and not free parameters of the model as in (10.48).

10.6 Gaussian Random Fields for Continuous Data

We now represent multivariate Gaussian distributions using undirected graphical models. A Gaussian random field is another name for a multivariate Gaussian distribution, where the structure of the underlying undirected graph is emphasized. This is a simple, but important graphical model that is appropriate for real-valued data $X = (X_1, \ldots, X_d) \in \mathbb{R}^d$.

Let X be distributed as

$$p(x_1, x_2, \ldots, x_d) = \frac{1}{(2\pi)^{d/2}|\mathbf{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T \mathbf{\Sigma}^{-1}(x-\mu)\right) \tag{10.52}$$

$$\propto \exp\left(-\frac{1}{2}x^T \mathbf{\Omega} x + x^T \mathbf{\Omega}\mu\right) \tag{10.53}$$

where $\mathbf{\Omega} = [\omega_{ij}] = \mathbf{\Sigma}^{-1}$ is the inverse covariance matrix, also known as the precision matrix. Let $G = (V, E)$ be the graph with $V = \{1, 2, \ldots, d\}$, and $(i, j) \in E$ in case $\omega_{ij} \neq 0$. Thus, the inverse of the covariance matrix encodes the graph. The distribution can then be expressed as

$$p(x_1, x_2, \ldots, x_d) \propto \prod_{(i,j)\in E} \psi_{ij}(x_i, x_j) \prod_{i\in V} \psi_i(x_i) \tag{10.54}$$

where

$$\psi_{ij}(x_i, x_j) = \exp\left(-\frac{1}{2}\omega_{ij}x_i x_j\right) \text{ and } \psi_i(x_i) = \exp\left(x_i \sum_{j=1}^{d} \omega_{ij}\mu_j\right). \tag{10.55}$$

Writing this as a product of factors with respect to the edges, we have

$$p(x_1, x_2, \ldots, x_d) \propto \prod_{(i,j)\in E} lde\psi_{ij}(x_i, x_j) \tag{10.56}$$

where $lde\psi_{ij}(x_i, x_j) = \psi_{ij}(x_i, x_j)\psi_i(x_i)^{1/n(i)}\psi_j(x_j)^{1/n(j)}$ and $n(i) = |\{(i, j) : (i, j) \in E\}|$ is the number of neighbors of node i in the graph. Therefore, the distribution P is Markov to the graph G. One thing to note is that there can be many cycles in the graph.

The following summarizes the above discussion.

Theorem 176. Let $X_1, \ldots, X_d \sim N(0, \mathbf{\Sigma})$ and $\mathbf{\Omega} = \mathbf{\Sigma}^{-1}$. Let $G = (V, E)$ be the graph defined above, and assume that the distribution P is faithful to G. Then

$$(i, j) \notin E \iff \omega_{ij} = 0 \iff X_i \perp\!\!\!\perp X_j \,|\, X_{V\setminus\{i,j\}}. \tag{10.57}$$

Proof. The result directly follows from (10.56) and Theorem 168. □

10.7 Log-linear Models for Discrete Data

We now represent multivariate discrete distributions using undirected graphical models. The log-linear models are exponential parameterizations of multinomials. Suppose $X_j \in \{0, 1, \ldots, m-1\}$, for $j \in V$, with $V = \{1, \ldots, d\}$; thus each of the d variables takes one of m possible values.

Definition 177. Let $X = (X_1, \ldots, X_d)$ be a discrete random vector with probability function $p(x) = \mathbb{P}(X = x) = \mathbb{P}(X_1 = x_1, \ldots, X_d = x_d)$ where $x = (x_1, \ldots, x_d)$. The probability mass function $p(x)$ is in log-linear form in case

$$\log p(x) = \sum_{A\subset V} \psi_A(x_A) \tag{10.58}$$

with the constraints that ψ_Φ is a constant , and if $j \in A$ and $x_j = 0$ then $\psi_A(x_A) = 0$.

The formula in (10.58) is called the log-linear expansion of $p(x)$. Each $\psi_A(x_A)$ may depend on some unknown parameters θ_A. Note that the total number of parameters satisfies $\sum_{j=1}^{d}\binom{d}{j}(m-1)^j = m^d$, however one of the parameters is the normalizing constant, and is determined by the constraint that the sum of the probabilities is one. Thus, there are m^d-1 free parameters, and this is a minimal exponential parameterization of the multinomial. Let $\theta = (\theta_A : A \subset V)$ be the set of all these parameters. We will write $p(x) = p(x;\theta)$ when we want to emphasize the dependence on the unknown parameters θ.

The next theorem provides an easy way to read out conditional independence in a log-linear model.

Theorem 178. Let (X_A, X_B, X_C) be a partition of $X = (X_1, \ldots, X_d)$. Then $X_B \perp\!\!\!\perp X_C \,|\, X_A$ if and only if all the ψ-terms in the log-linear expansion that have at least one coordinate in B and one coordinate in C are zero.

Proof. From the definition of conditional independence, we know that $X_B \perp\!\!\!\perp X_C \,|\, X_A$ if and only if $p(x_A, x_B, x_C) = f(x_A, x_B)g(x_A, x_C)$ for some functions f and g.

Suppose that ψ_t is 0 whenever t has coordinates in B and C. Hence, ψ_t is 0 if $t \not\subseteq A \cup B$ or $t \not\subseteq A \cup C$. Therefore

$$\log p(x) = \sum_{t \subset A\cup B} \psi_t(x_t) + \sum_{t \subset A\cup C} \psi_t(x_t) - \sum_{t \subset A} \psi_t(x_t). \tag{10.59}$$

Exponentiating, we see that the joint density is of the form $f(x_A, x_B)g(x_A, x_C)$. Therefore $X_B \perp\!\!\!\perp X_C \,|\, X_A$. The reverse follows by reversing the argument. □

A graphical log-linear model with respect to a graph G is a log-linear model for which the parameters ψ_A satisfy $\psi_A(x_A) \neq 0$ if and only if A is a clique of G. Thus, a graphical log-linear model has potential functions on each clique, both maximal and non-maximal, with the restriction that $\psi_A(x_A) = 0$ in case $x_j = 0$ for any $j \in A$. In a hierarchical log-linear model, if $\psi_A(x_A) = 0$ then $\psi_B(x_B) = 0$ whenever $A \subset B$. Thus, the parameters in a hierarchical model are nested, in the sense that if a parameter is identically zero for some subset of variables, the parameter for supersets of those variables must also be zero. Every graphical log-linear model is hierarchical, but a hierarchical model need not be graphical; Such a relationship is shown in Figure 10.16 and is characterized by the next lemma.

Lemma 179. A graphical log-linear model is hierarchical but the reverse need not be true.

Proof. We assume there exists a model that is graphical but not hierarchical. There must exist two sets A and B, such that $A \subset B$ with $\psi_A(x_A) = 0$ and $\psi_B(x_B) \neq 0$. Since the model is graphical, $\psi_B(x_B) \neq 0$ implies that B is a clique. We then know that A must also be a clique due to $A \subset B$, which implies that $\psi_A(x_A) \neq 0$. A contradiction.

To see that a hierarchical model does not have to be graphical. We consider the following example. Let

$$\log p(x) = \psi_\Phi + \sum_{i=1}^{3} \psi_i(x_i) + \sum_{1 \le j < k \le 3} \psi_{jk}(x_{jk}). \tag{10.60}$$

This model is hierarchical but not graphical. The graph corresponding to this model is a complete graph with three nodes X_1, X_2, X_3. It is not graphical since $\psi_{123}(x) = 0$, which is contradict with the fact that the graph is complete. □

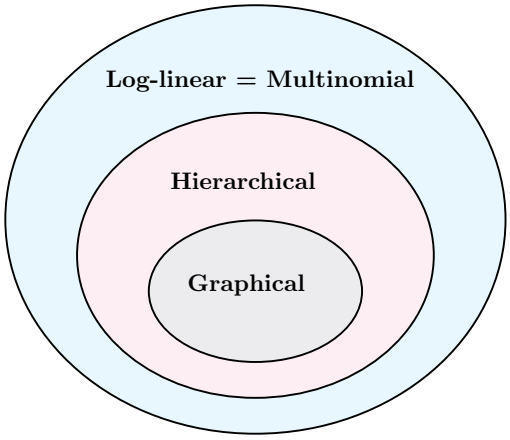


Figure 10.16: Every graphical log-linear model is hierarchical but the reverse may not be true.

10.7.1 Overcomplete Representations of log-linear Models

While log-linear models are, by convention, minimal exponential family models, it is often more convenient to work with overcomplete representation (or non-minimal representations). Consider a graph G, and consider the following exponential family model with parameters for each edge and vertex, where we assume that each random variable $X_s \in \{0, 1, \ldots, m-1\}$:

$$p_\theta(x) \propto \exp\left(\sum_{(s,t) \in E} \sum_{k,\ell=0}^{m-1} \theta_{s,t;k,\ell} \delta(x_s, k) \delta(x_t, \ell) + \sum_{s \in V} \sum_{k=0}^{m-1} \theta_{s;k} \delta(x_s, k) \right).$$

Here $\delta(x_s, k) = 1$ if $x_s = k$ and $\delta(x_s, k) = 0$ otherwise. This model has $m^2|E| + m|V|$ free parameters, where $|E|$ is the number of edges and $|V|$ is the number of vertices, compared with the number of free parameters in the corresponding hierarchical log-linear model of

$(m-1)^2|E|+(m-1)|V|$. Thus, there are linear dependencies among the sufficient statistics; for instance,

$$\sum_{k,\ell} \delta(x_s, k)\, \delta(x_t, \ell) = 1 \tag{10.61}$$

for every edge $(s,t) \in E$. It is often more convenient to work in this overparameterized form. This can be viewed as a discrete version of the multivariate Gaussian.

10.7.2 Ising and Potts Models

The Ising and Potts models are special cases of hierarchical log-linear models that originated in statistical and solid state physics, but also arise naturally in image analysis, models of social networks, and other areas. The Ising model can be thought of as a discrete analogue of the Gaussian graphical model. In essence, an Ising model is a hierarchical model where the nodes have binary variables and the distribution has pairwise interactions. Thus, the model is hierarchical with nonzero parameters on vertices and cliques only. The Potts model is the extension of the Ising model to the case of $m \geq 2$ possible labels at each node.

In more detail, let $G = (V, E)$ be a graph, and let $X_s \in \{0, 1\}$ be a binary labeling of each node. The Ising model takes the form

$$p(x;\theta) \;\propto\; \exp\left(\sum_{s\in V} \theta_s x_s + \sum_{(s,t)\in E} \theta_{s,t} x_s x_t\right). \tag{10.62}$$

For a given node $s \in V$, the conditional distribution, by the local Markov property, is given by a logistic regression model

$$p_s(x;\theta) \;\equiv\; p(x_s = 1 \,|\, x_t, t \neq s; \theta) = p(x_s = 1 \,|\, x_t, (s,t) \in E) \tag{10.63}$$

$$= \frac{\exp\left(\theta_s + \sum_{t:\,(s,t)\in E} \theta_{s,t} x_t\right)}{1 + \exp\left(\theta_s + \sum_{t:\,(s,t)\in E} \theta_{s,t} x_t\right)}. \tag{10.64}$$

The model is identifiable, and the Fisher information matrix at node $s \in V$ is given by

$$Q_s(\theta) = \mathbb{E}\left[p_s(X;\theta)(1 - p_s(X;\theta)) X_{V\setminus\{s\}} X_{V\setminus\{s\}}^T\right]. \tag{10.65}$$

Note that the matrix $Q_s(\theta)$ is the Fisher information matrix associated with the local conditional probability distribution. Intuitively, it serves as the counterpart for discrete graphical models of the covariance matrix $\mathbb{E}[XX^T]$ of Gaussian graphical models.

The m-dimensional Potts model allows $X_s \in \{0, 1, \ldots, m-1\}$, and has potential functions that test whether neighboring states are the same:

$$p(x;\theta) \quad \propto \quad \exp\left(\sum_{(s,t)\in E} \theta_{s,t}\delta(x_s, x_t)\right) \tag{10.66}$$

An "external field" can be incorporated through vertex potentials, so

$$p(x;\theta) \quad \propto \quad \exp\left(\sum_{s\in V} \theta_s x_s + \sum_{(s,t)\in E} \theta_{s,t}\delta(x_s, x_t)\right). \tag{10.67}$$

A simplified version of the model is used to study the properties of random colorings of a graph; here $X_s \in \{0, 1, \ldots, m-1\}$ denotes the color of node s, and $S(X) = \sum_{s,t} \delta(X_s, X_t)$ is the number of edges in the graph whose endpoints are colored with the same color. The associated Gibbs distribution is then

$$p(x;\theta) \quad \propto \quad \exp\left(-\beta J S(x)\right) \tag{10.68}$$

where $J \in \mathbb{R}$ and $\beta > 0$ is the "inverse temperature." Thus, the parameters in (10.66) satisfy $\theta_{s,t} = -\beta J$. When J is negative, higher probability is assigned to colorings X where the endpoints of edges have the same color. This is referred to as the ferromagnetic Potts model. If J is positive, however, then colorings X where the endpoints are identically colored have low probability; this is known as the antiferromagnetic Potts model. The parameter β is interpreted as inverse temperature: as $\beta \to \infty$ the temperature cools to zero. As the temperature cools, the antiferromagnetic model becomes more likely to sample a proper coloring of the graph, where no two adjacent nodes have the same color.

The Ising model corresponds to a Potts model with two colors, and the configurations can be thought of as weighted cuts in the graph, which divide the graph into two parts corresponding to the color of each node. When the temperature is low, and $J < 0$, with high probability the cut will be large. Since determining the largest cut in a graph is an NP-complete problem, it is (believed to be) not possible to efficiently sample from the Ising model in this regime. An interesting simulation of the Potts model with four states on a square two-dimensional lattice is shown in Figure 10.17.

10.8 Computing Marginal and Conditional Probabilities*

We now describe exact algorithms for calculating marginal and conditional distributions of probability distributions Markov to undirected graphs. Sometimes people call this exact

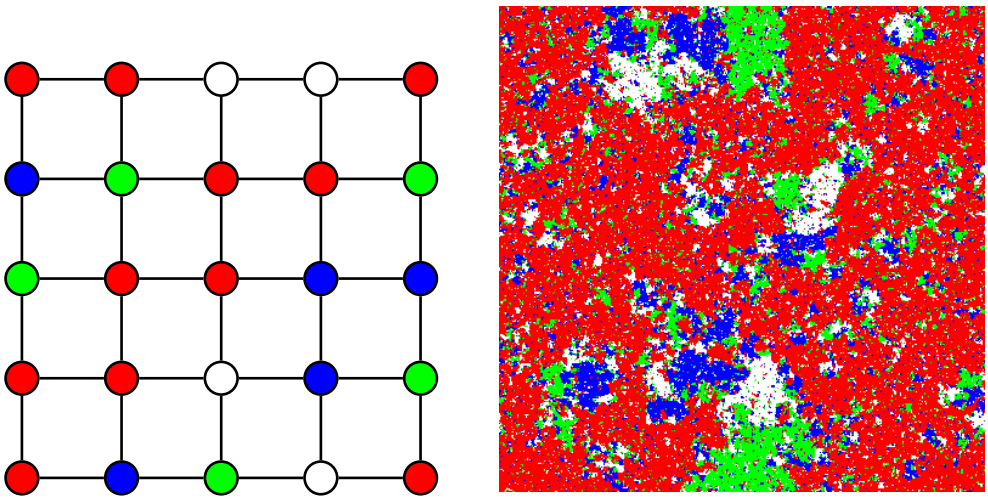


Figure 10.17: An important example of an undirected graphical model is the simple 2-dimensional grid graph. This arises in image processing, for instance, where each node may represent a pixel in an image. Random fields (or probability distributions) on such graphs were first studied in statistical physics. The right plot shows a sample from a Potts model over four states on a 2-dimensional lattice at the critical temperature (D. Wilson).

inference for undirected graphical models. But this is just calculating marginal and conditional probabilities, and is different from the more standard meaning of the term statistical inference, as used in this book. In this section, we briefly introduce three topics: (i) inference using variable elimination, (ii) inference using factor graphs, and (iii) inference over junction trees.

10.8.1 Variable Elimination

We illustrate the concept of variable elimination by a concrete example. Let $X = (X_1, \ldots, X_6)$ be a random vector whose distribution is Markov to the graph shown in Figure 10.18 (a). Suppose that we want to compute the conditional probability $\mathbb{P}(X_1 \,|\, X_6 = x_6)$, this can be done using an "elimination" algorithm. The basic scheme is very simple: for factored distributions it is better to interleave multiplication and addition. Such a baisc idea can be made more sophisticated with bucket elimination schemes. In this section, we describe the idea of variable elimination only using a DAG example. This algorithm can be easily extended to handle undirected graphs.

First, to calculate this conditional probability we need to marginalize

$$p(x_1, x_6) = \sum_{x_2, x_3, x_4, x_5} p(x_1, x_2, x_3, x_4, x_5, x_6) \tag{10.69}$$

We define

$$m_5(x_3, x_4, x_6) = \sum_{x_5} p(x_5 \mid x_3, x_4) p(x_6 \mid x_5) \tag{10.70}$$

$$m_{5,4}(x_2, x_3, x_6) = \sum_{x_4} p(x_4 \mid x_2) m_5(x_3, x_4, x_6) \tag{10.71}$$

$$m_{5,4,3}(x_2, x_6) = \sum_{x_3} p(x_3 \mid x_2) m_{5,4}(x_2, x_3, x_6) \tag{10.72}$$

$$m_{5,4,3,2}(x_1, x_6) = \sum_{x_2} p(x_2 \mid x_1) m_{5,4,3}(x_2, x_6). \tag{10.73}$$

Using the DAG factorization, the marginal probability $\mathbb{P}(x_1, x_6)$ can be written as

$$p(x_1, x_6) = \sum_{x_2, x_3, x_4, x_5} p(x_1) p(x_2 \mid x_1) p(x_3 \mid x_2) p(x_4 \mid x_2) p(x_5 \mid x_3, x_4) p(x_6 \mid x_5) \tag{10.74}$$

$$= p(x_1) \sum_{x_2} p(x_2 \mid x_1) \sum_{x_3} p(x_3 \mid x_2) \sum_{x_4} p(x_4 \mid x_2) \sum_{x_5} p(x_5 \mid x_3, x_4) p(x_6 \mid x_5)$$

$$= p(x_1) \sum_{x_2} p(x_2 \mid x_1) \sum_{x_3} p(x_3 \mid x_2) \sum_{x_4} p(x_4 \mid x_2) m_5(x_3, x_4, x_6) \tag{10.75}$$

$$= p(x_1) \sum_{x_2} p(x_2 \mid x_1) \sum_{x_3} p(x_3 \mid x_2) m_{5,4}(x_2, x_3, x_6) \tag{10.76}$$

$$= p(x_1) \sum_{x_2} p(x_2 \mid x_1) m_{5,4,3}(x_2, x_6) \tag{10.77}$$

$$= p(x_1) m_{5,4,3,2}(x_1, x_6). \tag{10.78}$$

We then determine the conditional probability as

$$p(x_1 \mid x_6) = \frac{p(x_1) m_{5,4,3,2}(x_1, x_6)}{\sum_{x_1'} p(x_1') m_{5,4,3,2}(x_1', x_6)}. \tag{10.79}$$

Note that the largest message table was $m_{5,4}(x_2, x_3, x_6)$; if each variable has m values, this requires m^3 entries. In general, it is difficult to choose an elimination order to minimize the table storage or amount of computation.

Graphically, we can view these operations as iteratively collapsing nodes in the graph, as shown in Figure 10.18. A difficulty with this procedure, in addition to its large computational cost in general, is that it only works for a particular "query" probability $p(x_1 \mid x_6)$ that we want to compute. To calculate another, for example $p(x_4 \mid x_3)$, requires a separate calculation, and it is not evident how to share the work across them. This is a motivation for studying more sophisticated graph representation methods, including factor graphs and tree representations of undirected graphical models.

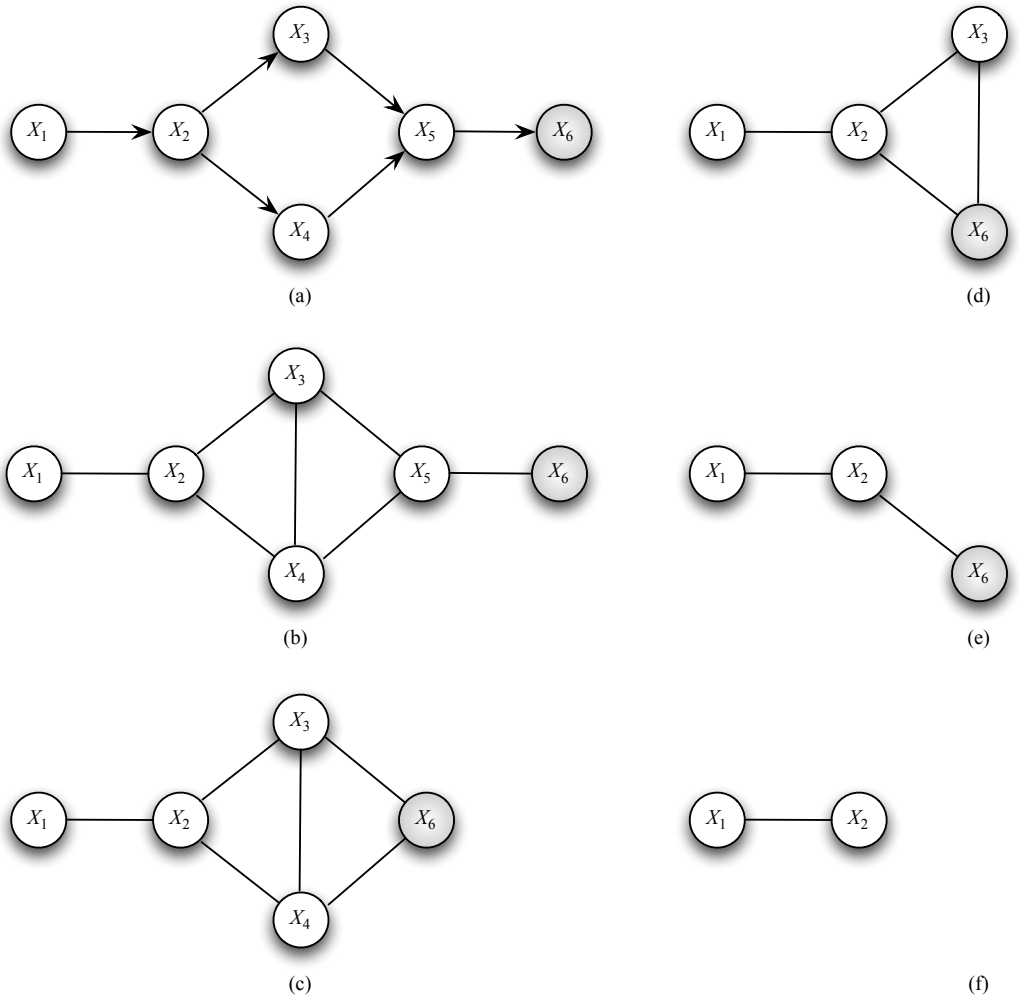


Figure 10.18: Elimination of nodes in a graph reduces the graph one node at a time, while creating "message" tables that store intermediate sums.

10.8.2 Factor Graphs

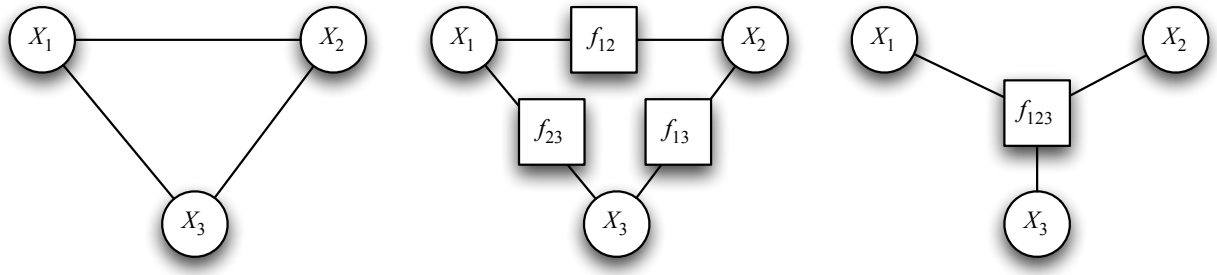
In the previous chapter, we described the message-passing or sum-product algorithm for exact inference on directed graphs. In this section, we show that the same algorithm can be adapted to undirected graphs. This can be done through an alternative graph representation called the factor graph. Unlike directed and undirected graphs that aim to capture the conditional independence relationships of many variables, factor graphs are used to capture the factorization of a multivariate distribution. Recall our point of view that the exponential representation of Markov random fields is the key, not necessarily the graph structure. However, in some cases an alternative graph structure provides a convenient representation.

Definition 180 (Factor Graph)**.** A factor graph is a bipartite graph representing the factorization of a probability distribution. Suppose that the distribution factors in a form

$$p(x_1, \ldots, x_d) = \prod_{j=1}^{m} f_j(x_{C_j}) \tag{10.80}$$

where $C_j \subset \{1, \ldots, d\}$ indexes a subset of all the variables. It is not required that C_j forms a clique in a graph (other than the complete graph). The corresponding factor graph $G = (V, F, E)$ consists of variable vertices $X = (X_1, \ldots, X_d)$, factor vertices $F = \{f_1, f_2, \ldots, f_m\}$, and edge set E. There is an undirected edge connecting a variable vertex and factor vertex f_j when $k \in S_j$. In general, we represent a variable vertex using round nodes and a factor vertex using square nodes.

From the above definition, it is easy to see that both directed and undirected graphs can be trivially converted into factor graphs. Compare to the corresponding directed and undirected graphs, factor graphs provide a more refined factorization information of probability distributions. However, the factor graph representation is not unique. For example, for the undirected graph in Figure 10.19 (a), we plot two different factor graphs: (i) Figure 10.19 (b) corresponds to the factorized potential $\psi_{x_1,x_2,x_3} = f_{12}(x_1, x_2) f_{23}(x_2, x_3) f_{13}(x_1, x_3)$; while Figure 10.19 (c) illustrates a different factor graph corresponding to the factorized potential $\psi_{x_1,x_2,x_3} = f_{123}(x_1, x_2, x_3)$.



(a) Original undirected graph (b) The first factor graph (c) The second factor graph

Figure 10.19: (a) An undirected graph providing no information about possible factorization of the potential function associated with a given clique. (b) The factor graph corresponding to the factorized potential $\psi_{x_1,x_2,x_3} = f_{12}(x_1, x_2) f_{23}(x_2, x_3) f_{13}(x_1, x_3)$. (c) A different factor graph corresponding to the factorized potential $\psi_{x_1,x_2,x_3} = f_{123}(x_1, x_2, x_3)$.

One thing to note is that by introducing additional random variables into a directed or undirected graph, we can always represent more subtle factorization information as can be represented by a factor graph, so factor graph does not really provide much representation power. However, factor graphs can be combined with message passing algorithms (or belief propagation algorithms) to efficiently compute the marginal and conditional distributions of Markov random fields.

10.8.3 Sum-Product and Max-Product Algorithms for Factor Trees

We first use an illustrative example to explain how the sum-product algorithm can be applied to compute the marginal distribution using an illustrative example. For simplicity, we only consider tree-structured factor graph, i.e., the undirected graph obtained by ignoring the distinction between variable nodes and factor nodes is a tree. On these graphs exact inference is possible.

Given a factor graph, we denote $N(X)$ to be the neighborhood of a node X. For example, in Figure 10.20, $N(X_1) = \{f_{123}, f_1, f_{16}\}$ and $N(f_{123}) = \{X_1, X_2, X_3\}$. Since a factor graph is bipartite, we know that the neighborhood of a variable node only contains factor nodes, while the neighborhood of a factor node only contains variable nodes. The detailed sum-product algorithm for factor tree is shown in the following:

Sum-Product Algorithm for Factor Tree The marginal probability of X_i can be evaluated as

$$p(x_i) = \prod_{f_C \in N(x_i)} m_{f_C \to x_i}(x_i). \tag{10.81}$$

(R1) If f_C is a leaf node, then $m_{f_C \to x}(x_C) = f_C(x_C)$.

(R2) If x is a leaf node, then $m_{x \to f_C}(x) = 1$.

(R3) The message sent from a non-leaf factor node f_C to a variable node X_i is:

$$m_{f_C \to x_i}(x_i) = \sum_{x_{N(f_C)\setminus\{x_i\}}} f(x_C) \left(\prod_{x' \in N(f_C)\setminus\{x\}} m_{x' \to f_C}(x') \right). \tag{10.82}$$

(R4) The message sent from a non-leaf variable node X_i to a factor node f_C is:

$$m_{x_i \to f_C}(x_i) = \prod_{f_{C'} \in N(x_i)\setminus\{f_C\}} m_{f_{C'} \to x_i}(x_i). \tag{10.83}$$

To better understand the sum-product algorithm, we apply it on a concrete example.

Example 181 (Sum-product algorithm)**.** Consider the factor graph in Figure 10.20 for a discrete distribution, the joint distribution has the factorization

$$p(x) = f_{123}(x_1, x_2, x_3) f_{24}(x_2, x_4) f_1(x_1) f_{16}(x_1, x_6). \tag{10.84}$$

We want to calculate the marginal distribution $p(x_1)$.

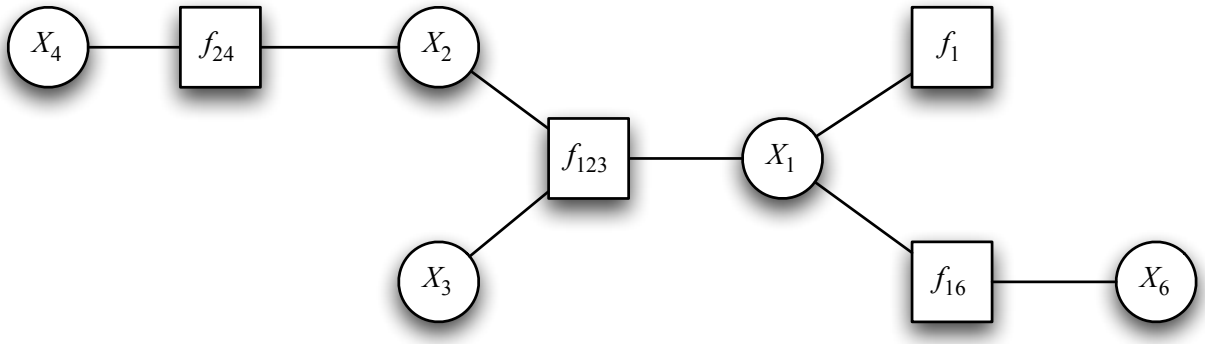


Figure 10.20: A factor graph illustrating the sum-product algorithm.

First, the neighborhood of X_1 is $N(X_1) = \{f_{123}, f_1, f_6\}$. By the rules (R1) and (R2) we have that message $m_{f_1 \to x_1}(x_1) \equiv f_1(x_1)$ and $m_{x_6 \to f_{16}}(x_6) \equiv 1$.

Further, by rules (R3) and (R4), we have $m_{f_{16} \to x_1}(x_1) = \sum_{x_6} f_{16}(x_1, x_6) m_{x_6 \to f_{16}}(x_6) = \sum_{x_6} f_{16}(x_1, x_6)$ and $m_{f_{123} \to x_1} = \sum_{x_2, x_3, x_4} f_{123}(x_1, x_2, x_3) f_{24}(x_2, x_4)$.

$$
\begin{aligned}
p(x_1) &= \prod_{f_C \in N(x_1)} m_{f_C \to x_1}(x_1). && (10.85)\\
&= \Big[m_{f_{123} \to x_1}(x_1)\Big] \cdot \Big[m_{f_1 \to x_1}(x_1)\Big] \cdot \Big[m_{f_{16} \to x_1}(x_1)\Big] && (10.86)\\
&= \Big[\sum_{x_2, x_3, x_4} f_{123}(x_1, x_2, x_3) f_{24}(x_2, x_4)\Big] \cdot \Big[f_1(x_1)\Big] \cdot \Big[\sum_{x_6} f_{16}(x_1, x_6)\Big] && (10.87)\\
&= \sum_{x_2} \cdots \sum_{x_6} p(x) && (10.88)
\end{aligned}
$$

Therefore, we see the sum-product algorithm gives us desired result.

Let $E \subset V$ and X_E be a subset of variables. If $X_E = x'_E$ is observed, the above sum-product algorithm can be easily modified to calculate the conditional marginal $p(x_i \mid x'_E)$. To do this, we only need to multiply all the messages by an indicator function $\delta(X_E, x'_E)$. The sum-product algorithm can also be straightforwardly modified to evaluate $\max_{x_{V \setminus \{i\}}} p(x)$. For this, we only need to replace all the summation operations by maximization. The resulting procedure is called max-product algorithm. To handle numerical underflow problem, we could also work in the logarithmic-space, which leads to the max-sum algorithm.

10.8.4 Chordal Graphs, Junction Trees, and Decomposable Graphs

In graph theory, a tree decomposition is a mapping of a graph into a tree that can be used to speed up solving certain problems on the original graph. A *tree reprtesentation* of a graph can be used to solve problems on the original graph in terms of dynamic programming on the tree. When studying probabilistic graphical models, the most popular tree representations are junction trees. They play important roles in problems like probabilistic inference and query optimization. In this section we introduce the concepts of chordal graphs, decomposable graphs, and junction trees. These concepts are fundamental since they provide a unified framework to inference that makes systematic use of the Markov properties of graphical models.

Let $G = (V, E)$ be an undirected graph. We denote $V = \{1, \ldots, d\}$ to be the vertex set and E to be the edge set. We say a cycle in G is chordless is all pairs of vertices that are not adjacent in the cycle are not neighbors. We then have the following definitions:

Definition 182 (Chordal graph)**.** A graph is chordal (also called triangulated) if it contains no chordless cycles of length greater than 3.

Definition 183 (Decomposable graph)**.** A graph $G = (V, E)$ is decomposable if either

1. G is complete, or
2. There exist two non-empty subsets $A, B \subset V$ and a complete subset $S \subset V$ which separates A and B in G, such that A, B, S form a partition of V and $A \cup B$ and $A \cup C$ are decomposable.

Definition 184 (Recursively simplicial graph)**.** A vertex in a graph G is simplicial if it has no neighbor or its neighbors form a complete subgraph. A graph G is recursively simplicial if it contains a simplicial vertex and when this vertex is removed the obtained subgraph is recursively simplicial.

To get more intuition on the recursively simplicial graph, recall that tree has a recursive definition. A graph G is a tree if it has only one single vertex, or there exists a vertex (leaf) adjacent to no more than one node. Removal of that vertex results in another tree. It is easy to see that the tree graph is recursively simplicial.

Definition 185 (Clique tree)**.** A clique tree for a graph $G = (V, E)$ is a tree $T = (V_T, E_T)$ where V_T is a set of cliques of G that contains all maximal cliques of G. The edge connecting two cliques $C_1, C_2 \in V_T$ are labeled with the intersection set $C_1 \cap C_2$[2].

Definition 186 (Junction tree)**.** A junction tree for a graph G is a clique tree that satisfies the running intersection property, i.e., for any cliques C_1 and C_2 in the tree, every clique on the path connecting C_1 and C_2 contains $C_1 \cup C_2$.

[2]One thing to note is that different edges may have the same edge labels

From the above definitions, we see that both decomposable graphs and recursively simplicial graphs are defined in a recursive way. The next theorem unifies the above definitions by showing their equivalence.

Theorem 187 (Equivalence)**.** The following properties of G are equivalent

1. G is chordal (or triangulated).
2. G is decomposable.
3. G is recursively simplicial.
4. G has a junction tree representation.
5. There is an orientation of the edges of G that gives a directed acyclic graph whose moral graph is G.
6. There is a directed graphical model with conditional independencies identical to those implied by G.

Proof. To prove the equivalence of 1, 2, 3, 4, we only need to show that $1 \implies 2 \implies 3 \implies 4 \implies 1$. Since most definition are recursive, all the proofs are based on induction on the size of the graph. We could then show $2 \implies 5 \implies 6 \implies 2$. Detailed proof is left an exercise (See Exercise 6). □

With above definitions of decomposable graphs, we can further define decomposable models. Decomposable models are a restricted family of undirected graphical models that have interesting relationships with directed graphical models (e.g., each decomposable model has a representation as either an undirected or a directed model) as is shown in the following definition:

Definition 188 (Decomposable models)**.** We call a probability distribution P decomposable if it satisfies any of the following (equivalent) properties:

1. P is Markov to an undirected chordal (or decomposable) graph.
2. P is Markov to a directed acyclic graph which has no unshielded colliders.
3. P is simultaneously Markov to a DAG and faithful to an undirected graph.
4. P is Markov to a graph which has a junction tree representation.

The above definition implies that the set of decomposable models is in fact the intersection of directed and undirected graphical models. A big picture illustrating the relationships of different graphical models and graphs are provided in Figure 10.21.

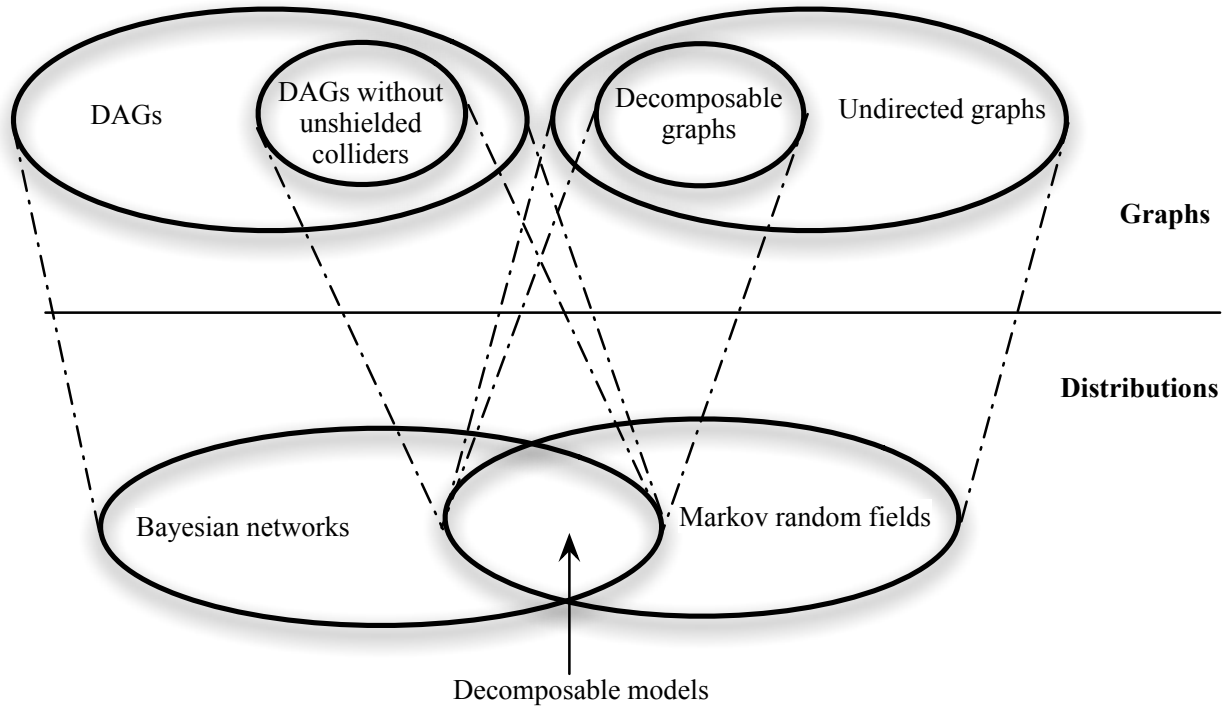


Figure 10.21: The big picture of decomposable graphs, Bayesian networks, and Markov random fields.

The next two theorems summarize important properties of junction trees.

Theorem 189. Let $T = (V_T, E_T)$ be a junction tree and $S = \{C_1, C_2\} \in E_T$. Let $T_1 = (V_{T_1}, E_{T_1})$ and $T_2 = (V_{T_2}, E_{T_2})$ be the maximal subtrees with roots C_1 and C_2 that does not contain the edge S. Then

$$[V_{T_1} \setminus (C_1 \cap C_2)] \perp\!\!\!\perp [V_{T_2} \setminus (C_1 \cap C_2)] \,|\, C_1 \cap C_2. \tag{10.89}$$

Proof. Since T is a junction tree we have $[V_{T_1} \setminus (C_1 \cap C_2)] \cap [V_{T_2} \setminus (C_1 \cap C_2)] = \Phi$. Since V_T only contain the maximal cliques, the neighbors of any vertex in $V_{T_1} \setminus (C_1 \cap C_2)$ are in V_{T_1}. Therefore $V_{T_1} \setminus (C_1 \cap C_2)$ is separated from $V_{T_2} \setminus (C_1 \cap C_2)$ by $C_1 \cap C_2$. □

Theorem 190 (Junction tree decomposition)**.** Let G be a graph with a junction tree representation $T = (V_T, E_T)$. Let P be a probability distribution that is Markov to G. The density $p(x)$ of P can be factorized as

$$p(x) = \frac{\prod_{C \in V_T} p(x_C)}{\prod_{S \in E_T} p(x_S)}, \tag{10.90}$$

where if $S = \{C_1, C_2\}$, then x_S denotes to be $x_{C_1 \cap C_2}$.

Proof. Let $n = |V_T|$. Since T is a tree, we can find a topological order $\{C_1, \ldots, C_n\}$ of all the vertices in V_T (i.e. each node?s parent appears earlier in the ordering). Let $C_{\pi(i)}$ be parent node of the vertex C_i. From Theorem 189 we have

$$X_{C_1}, \ldots, X_{C_{i-1}} \perp\!\!\!\perp X_{C_i} \,|\, X_{C_{\pi(i)} \cap C_i}. \tag{10.91}$$

Then, we have

$$\begin{aligned} p(x) &= \prod_{i=1}^{n} p\left(x_{C_i} \,|\, x_{C_1}, \ldots, x_{C_{i-1}}\right) && (10.92)\\ &= \prod_{i=1}^{n} p(x_{C_i} \,|\, x_{C_{\pi(i)} \cap C_i}) && (10.93)\\ &= \prod_{i=1}^{n} \frac{p(x_{C_i})}{p(x_{C_{\pi(i)} \cap C_i})} && (10.94)\\ &= \frac{\prod_{C \in V_T} p(x_C)}{\prod_{S \in E_T} p(x_S)}, && (10.95) \end{aligned}$$

where (10.93) follows from (10.91). □

Let $G = (V, E)$ be a graph with a junction tree representation $T = (V_T, E_T)$ and $V_T = \{C_1, \ldots, C_n\}$. The size C_j of the j-th clique is the number of elements it contains, and the width of a junction tree is defined as

$$\text{width}(T) = \max_{C \in V_T} |C| - 1 \tag{10.96}$$

and the treewidth of G is defined as

$$\text{treewidth}(G) = \min_{T} \text{width}(T), \tag{10.97}$$

where the minimization is over all possible junction tree representation of G. If G is a tree, then $\text{treewidth}(G) = 1$. Determining the treewidth of a graph is an NP-hard problem.

From previous discussions, we see that junction tree can be viewed as an extension of the tree graph. Thus the sum-product and max-product algorithms can be applied on junction trees using similar ideas. Given a graph G, one important quantity that affects the computational complexity of these algorithms is the treewidth of G. In general, the computational time is exponential time is exponential to $\text{treewidth}(G)$.

10.9 Parameter Estimation

The problem of parameter estimation is difficult for Markov random fields. The main challenge lies in the normalizing constant. Let $x_1, \ldots, x_n$ be n data points sampled from a model of the form $p(x;\theta) \propto \exp(\sum_C \theta_C f_C(x_C))$, the log-likelihood is given by

$$\ell_n(\theta) = \frac{1}{n}\sum_{i=1}^{n}\sum_{C}\theta_C f_C(x_{iC}) - \log Z(\theta) \tag{10.98}$$

and the gradient satisfies

$$\begin{aligned}\frac{\partial \ell_n(\theta)}{\partial \theta_C} &= \frac{1}{n}\sum_{i=1}^{n} f_C(x_{iC}) - \int p(x;\theta) f_C(x_C) dx_C & (10.99)\\ &= \mathbb{E}_{\widehat{P}_n}[f_C(X_C)] - \mathbb{E}_\theta[f_C(X_C)] & (10.100)\end{aligned}$$

where $\widehat{P}_n$ denotes the empirical distribution. In general, neither the likelihood nor the gradient can be efficiently computed. The global normalization factor couples the parameters, and makes parameter estimation challenging. An exception is the family of decomposable models, for which the probability distribution can be written as in (10.90), which does not explicitly involve the normalizing constant. Therefore, if we parameterize $p(x_C)$ and $p(x_S)$ in a consistent way, maximum likelihood estimation of the parameters is tractable for decomposable models. For general undirected graphical models, we must design algorithms that can effectively exploit the graph structure. In the following, we briefly discuss the pseudo likelihood method for parameter estimation.

10.9.1 Pseudo-Likelihood Consistency

The *pseudo-likelihood estimator*, first proposed by [10], instead of maximizing the log-likelihood, maximizes the product of the local conditional probabilities

$$\ell_n^P(\theta) = \frac{1}{n}\sum_{i}\sum_{s\in V}\log p_\theta\left(x_{is} \,|\, x_{it},\, t \in N(s)\right). \tag{10.101}$$

In the usual case, $p_\theta(x_s \,|\, x_t,\, t \in N(s))$ is a logistic regression model. For example, in the Ising model, the pseudo-likelihood takes the form

$$\ell_n^P(\theta) = \frac{1}{n}\sum_{i}\sum_{s,t}\left\{\theta_{st}x_{is}x_{it} - \log\left(1+\exp\left(\sum_t \theta_{st}x_{it}\right)\right)\right\} \tag{10.102}$$

The maximum pseudo-likelihood estimate is given by

$$\widehat{\theta}_n^{PL} = \arg\max_\theta \ell_n^P(\theta) \tag{10.103}$$

[17] proved the following consistency result.

Theorem 191 ([17]). Suppose that $p_\theta(x)$ defines a translation-invariant Gibbs distribution on $\Lambda_n \subset \mathbb{Z}^d$ with fixed boundary conditions. Moreover, suppose that θ is identifiable with the true value θ^*. Then for all $\epsilon > 0$, there exist $c > 0$ and $\delta > 0$ such that

$$\mathbb{P}\left(\|\widehat{\theta}_n^{PL} - \theta^*\|_2 > \epsilon\right) \leq c \exp\left(-|\Lambda_n|\delta\right). \tag{10.104}$$

10.10 Structure Learning

In this section we describe methods that could estimate the undirected graphs based on observational samples. We are especially interested in the high dimensional settings. To handle high dimensions, the most common assumption is that G is sparse. This means that $\mathbf{\Omega} = \mathbf{\Sigma}^{-1}$ has many zeroes. (Alternatively, you can take the point of view that we are finding a sparse approximation to the true graph G.)

10.10.1 Structure Learning for Gaussian and Ising Models

In the high dimensional setting, we cannot estimate G unless we make some extra assumption. The most popular method for estimating G when the dimension d is large is to assume that X has a multivariate Gaussian distribution, $X \in N(\boldsymbol{\mu}, \mathbf{\Sigma})$. From Theorem 176, we know that the graph is encoded by the sparsity pattern of the inverse covariance matrix $\mathbf{\Omega} = [\omega_{ij}] = \mathbf{\Sigma}^{-1}$.

Inspired by the success of the lasso, [63] proposed a pseudo-likelihood approach to estimate the graph. They use the lasso to regress X_i on $(X_j : \ j \neq i)$. This is repeated for each X_i. Let

$$\beta^i = \text{argmin}_\beta \mathbb{E}(X_i - \sum_{j \neq i} \beta_j X_j)^2$$

and define the neighborhood of i, $N_i = \{j : \ \beta_j^i \neq 0\}$. The lasso gives estimates $\widehat{\beta}^i$ for all i. Let $\widehat{N}_i = \{j : \ \widehat{\beta}_j^i \neq 0\}$ and $\widehat{E}$ be the set of edges (i, j) such that $i \in \widehat{N}_j$ and $j \in \widehat{N}_i$. Under suitable sparsity assumptions they prove that $\mathbb{P}(N_i = \widehat{N}_i) \to 1$ as $n \to \infty$ even if $d = n^\gamma$ for some $\gamma > 0$. Similarly, $\mathbb{P}(E = \widehat{E}) \to 1$ as $n \to \infty$.

Alternatively, [98] suggested a penalized likelihood estimator

$$\widehat{\mathbf{\Omega}} = \arg\max_{\Omega \succ 0} \{\text{loglikelihood}(\mathbf{\Omega}) - \lambda \sum_{j,k} |\omega_{jk}|\},$$

where the loglikelihood of $\mathbf{\Omega}$ is evaluated under the Gaussian model. The estimator $\widehat{\mathbf{\Omega}}$ can be efficiently computed using the glasso algorithm [33, 5], which is a block coordinate

descent procedure that uses the standard lasso to estimate a single row and column of $\boldsymbol{\Omega}$ in each iteration. In this subsection, we focus on introducing the glasso algorithm, which provides a good example of convex duality, and in particular how efficient algorithms can be derived by formulating the dual problem.

Suppose that we have n observations $x_1, \ldots, x_n$ where $x_i \in \mathbb{R}^d$. The maximum likelihood estimate of μ and $\boldsymbol{\Omega}$ are obtained in terms of the sample mean and sample covariance,

$$\widehat{\mu}_n = \frac{1}{n}\sum_{i=1}^{n} x_i, \tag{10.105}$$

$$\widehat{\boldsymbol{\Omega}}_n = \boldsymbol{S}_n^{-1} \text{ where } \boldsymbol{S}_n = \frac{1}{n}\sum_{i=1}^{n}(x_i - \widehat{\mu}_n)(x_i - \widehat{\mu}_n)^T. \tag{10.106}$$

In the following, we will simply assume that the mean μ is known. For $d < n$, the maximum likelihood estimate is easily obtained by noting that the log-likelihood of the data is

$$\ell(\boldsymbol{\Omega}) = \frac{n}{2}\log|\boldsymbol{\Omega}| - \frac{n}{2}\text{tr}(\boldsymbol{\Omega}\boldsymbol{S}_n) - \frac{nd}{2}\log(2\pi) \tag{10.107}$$

and the gradient of $\log|\boldsymbol{X}|$ as a function of $\boldsymbol{X} \in \mathcal{S}_{++}^d$ is $\nabla \log|\boldsymbol{X}| = \boldsymbol{X}^{-1}$.

If $d > n$, the maximum likelihood estimate is no longer valid because the matrix $\boldsymbol{S}_n$ is singular. We need to use regularized estimator. The negative log-likelihood, rescaled for convenience, is

$$f(\boldsymbol{\Omega}) = \text{tr}(\boldsymbol{\Omega}\boldsymbol{S}_n) - \log|\boldsymbol{\Omega}|. \tag{10.108}$$

Just as for the lasso, we can look for a sparse matrix $\boldsymbol{\Omega}$ by imposing an L_1 penalty, leading to the optimization problem

$$\min_{\Omega \succ 0} \quad \text{tr}(\boldsymbol{\Omega}\boldsymbol{S}_n) - \log|\boldsymbol{\Omega}| \tag{10.109}$$

$$\text{such that} \quad \|\boldsymbol{\Omega}\|_1 \leq L, \tag{10.110}$$

where $\|\boldsymbol{\Omega}\|_1 = \sum_{i,j}|\omega_{ij}|$. As is often the case, the dual form leads to insight into the optimization problem and its properties. To derive the dual, first note that we can write

$$\|\boldsymbol{X}\|_1 = \max_{\|U\|_\infty \leq 1} \text{tr}(\boldsymbol{X}\boldsymbol{U}), \tag{10.111}$$

where $\boldsymbol{U} = [u_{ij}]$ and $\|\boldsymbol{U}\|_\infty = \max_{i,j}|u_{ij}|$. Therefore, the Lagrangian of the problem can be written as

$$\mathcal{L}(\boldsymbol{\Omega}, \lambda) = \text{tr}(\boldsymbol{\Omega}\boldsymbol{S}_n) - \log|\boldsymbol{\Omega}| + \lambda\|\boldsymbol{\Omega}\|_1 \tag{10.112}$$

$$= \max_{\|U\|_\infty \leq \lambda}\Big\{\text{tr}(\boldsymbol{\Omega}\boldsymbol{S}_n) - \log|\boldsymbol{\Omega}| + \text{tr}(\boldsymbol{\Omega}\boldsymbol{U})\Big\} \tag{10.113}$$

$$= \max_{\|U\|_\infty \leq \lambda}\Big\{\text{tr}(\boldsymbol{\Omega}(\boldsymbol{S}_n + \boldsymbol{U})) - \log|\boldsymbol{\Omega}|\Big\}. \tag{10.114}$$

The dual is thus

$$h(\lambda) = \min_{\Omega \succ 0} \max_{\|U\|_\infty \leq \lambda} \Big\{ \text{tr}(\boldsymbol{\Omega}(\boldsymbol{S}_n + \boldsymbol{U})) - \log |\boldsymbol{\Omega}| \Big\}. \tag{10.115}$$

Interchanging the min and max (which is justified by strong duality), we can then perform the minimization analytically, to obtain $\boldsymbol{\Omega} = (\boldsymbol{S}_n + \boldsymbol{U})^{-1}$, and thus

$$h(\lambda) = \max_{\|U\|_\infty \leq \lambda} \log |\boldsymbol{S}_n + \boldsymbol{U}| + d. \tag{10.116}$$

The log determinant acts as a barrier function that ensures $\boldsymbol{S}_n + \boldsymbol{U}$ is positive definite. Letting $\boldsymbol{\Sigma} = \boldsymbol{S}_n + \boldsymbol{U}$, we can then re-express the dual optimization as estimating the covariance $\boldsymbol{W}$ according to

$$\widehat{\boldsymbol{\Sigma}}_n = \underset{W \,:\, \|W - S_n\|_\infty \leq \lambda}{\text{argmax}} \log |\boldsymbol{W}|. \tag{10.117}$$

Thus, the dual estimates the covariance matrix while the primal estimates the inverse covariance.

To carry out this dual optimization, [5] propose a block-coordinate ascent algorithm, optimizing over one row and column of $\boldsymbol{W}$ at a time. To derive the algorithm, first note that

$$\widehat{\boldsymbol{\Sigma}}_{jj} = \boldsymbol{S}_{jj} + \lambda \tag{10.118}$$

(where we drop the sample size subscript on S_n for notational clarity). Thus, the diagonal is fixed in closed form. Suppose that we are optimizing over the jth row and column; then after reordering rows by moving the jth row into the first position, we can write the current estimate as

$$\widehat{\boldsymbol{\Sigma}} = \begin{pmatrix} \boldsymbol{S}_{jj} + \lambda & y^T \\ y & \boldsymbol{W}_{\setminus jj} \end{pmatrix} \tag{10.119}$$

where $\boldsymbol{W}_{\setminus jj}$ is the matrix $\boldsymbol{W}$ with the jth row and column removed. By Schur complements, we have

$$\log |\widehat{\boldsymbol{\Sigma}}| = \log |\boldsymbol{W}_{\setminus jj}| + \log \left(\boldsymbol{S}_{jj} + \lambda - y^T \boldsymbol{W}_{\setminus jj}^{-1} y \right). \tag{10.120}$$

Therefore, optimizing over y is equivalent to the following quadratic program:

$$\min_{y} \qquad y^T \boldsymbol{W}_{\setminus jj}^{-1} y \tag{10.121}$$

$$\text{such that} \qquad \|y - \boldsymbol{S}_j\|_\infty \leq \lambda \tag{10.122}$$

where $\boldsymbol{S}_j$ is the jth column of $\boldsymbol{S}$.

Now, we can recast this inner optimization problem into a more convenient form by forming the dual quadratic program, which is given as follows:

$$\min_x x^T \boldsymbol{W}_{\backslash jj} x - \boldsymbol{S}_j^T x + \lambda \|x\|_1, \tag{10.123}$$

where the mapping between these primal and dual subproblems is $y = \boldsymbol{W}_{\backslash jj} x$. This is equivalent to a lasso regression problem. To see this, let $\mathbf{Q}$ be the square root $\boldsymbol{W}_{\backslash jj}^{1/2}$ and let $b = \frac{1}{2}\mathbf{Q}^{-1}\boldsymbol{S}_j$. Then the problem above is equivalent to

$$\min_x \|\mathbf{Q}x - b\|_2^2 + \lambda \|x\|_1. \tag{10.124}$$

Thus, we estimate the covariance by an iterated sequence of lasso problems. This can be implemented in an efficient manner using iterative soft thresholding. In particular, Equation (10.123) is solved iteratively by

$$x \leftarrow \mathcal{S}_\lambda^{(1)}\left(x + \boldsymbol{S}_j - W_{\backslash jj} x\right) \tag{10.125}$$

where $\mathcal{S}_\lambda^{(1)}$ is the soft thresholding operator (assuming that $\boldsymbol{W}_{\backslash jj}$ has norm less than or equal to one; the problem can be rescaled to ensure this is the case). After convergence, y in Equation (10.119) is updated with $\boldsymbol{W}_{\backslash jj} x$. At each step, the inverse $\boldsymbol{W}^{-1}$ can be efficiently updated using Schur complements, to yield the estimate $\widehat{\boldsymbol{\Omega}}_n$.

Figure 10.22 shows a gene-gene interaction graph estimated using this method. The data are based on Affymetrix GeneChip microarrays for the plant *Arabidopsis thaliana*, [95], with sample size is $n = 118$. The expression levels for each chip are pre-processed by log-transformation and standardization. A subset of 40 genes from the isoprenoid pathway are chosen, and the graphs for two different values of the regularization parameter λ are shown.

To handle discrete data, [71] proposed a pseudo-likelihood based graph estimation procedure for high dimensional Ising models (or discrete Markov random fields). Their idea is similar to that of [63] but replacing the lasso with L_1-regularized logistic regression.

10.10.2 Learning Forest Graphical Models

Besides Gaussian and Ising models, another tractable family for structure learning is forest graphical models. If F is a d-node undirected forest with vertex set $V_F = \{1, \ldots, d\}$ and edge set $E_F \subset \{1, \ldots, d\} mes \{1, \ldots, d\}$, the number of edges satisfies $|E_F| < d$, noting that we do not restrict the graph to be connected. From (10.90) we get that a probability density function $p(x)$ that is Markov to F can be written as

$$p(x) = \prod_{(i,j)\in E_F} \frac{p(x_i, x_j)}{p(x_i)\, p(x_j)} \prod_{k \in V_F} p(x_k), \tag{10.126}$$

Figure 10.22: Estimated gene-gene interaction graphs for the *Arabidopsis thaliana* data, for the regularization levels $\lambda = 0.2448, 0.30857$.

where each $p(x_i, x_j)$ is a bivariate density, and each $p(x_k)$ is a univariate density. Using (10.126), we have

$$\begin{aligned} & \mathbb{E} \log p(X) \qquad (10.127) \\ &= -\int p(x) \left(\sum_{(i,j) \in E_F} \log \frac{p(x_i, x_j)}{p(x_i)p(x_j)} + \sum_{k \in V_F} \log \left(p(x_k) \right) \right) dx \\ &= -\sum_{(i,j) \in E_F} \int p(x_i, x_j) \log \frac{p(x_i, x_j)}{p(x_i)p(x_j)} dx_i dx_j - \sum_{k \in V_F} \int p(x_k) \log p(x_k) dx_k \\ &= -\sum_{(i,j) \in E_F} I(X_i; X_j) + \sum_{k \in V_F} H(X_k), \qquad (10.128) \end{aligned}$$

where

$$I(X_i; X_j) \equiv \int p(x_i, x_j) \log \frac{p(x_i, x_j)}{p(x_i)\, p(x_j)} \, dx_i dx_j \qquad (10.129)$$

is the mutual information between the pair of variables X_i, X_j and

$$H(X_k) \equiv -\int p(x_k) \log p(x_k) \, dx_k \qquad (10.130)$$

is the entropy.

The optimal forest F^* can be found by minimizing the right hand side of (10.128). Since the entropy term $H(X) = \sum_k H(X_k)$ is constant across all forests, this can be recast as the problem of finding the maximum weight spanning forest for a weighted graph, where the weight $w(i, j)$ of the edge connecting nodes i and j is $I(X_i; X_j)$. Kruskal's algorithm

[50] is a greedy algorithm that is guaranteed to find a maximum weight spanning tree of a weighted graph. In the setting of density estimation, this procedure was proposed by [16] as a way of constructing a tree approximation to a distribution. At each stage the algorithm adds an edge connecting that pair of variables with maximum mutual information among all pairs not yet visited by the algorithm, if doing so does not form a cycle. When stopped early, after $k < d$ edges have been added, it yields the best k-edge weighted forest.

Of course, the above procedure is not practical since the true density $p(x)$ is unknown. In applications, we parameterize bivariate and univariate distributions to be $p_{\theta_{ij}}(x_i, x_j)$ and $p_{\theta_k}(x_k)$. We replace the population mutual information $I(X_i; X_j)$ in (10.128) by the plug-in estimate $\widehat{I}_n(X_i, X_j)$, defined as

$$\widehat{I}_n(X_i, X_j) = \int p_{\widehat{\theta}_{ij}}(x_i, x_j) \log \frac{p_{\widehat{\theta}_{ij}}(x_i, x_j)}{p_{\widehat{\theta}_i}(x_i)\, p_{\widehat{\theta}_j}(x_j)}\, dx_i dx_j \tag{10.131}$$

where $\widehat{\theta}_{ij}$ and $\widehat{\theta}_k$ are maximum likelihood estimates. Given this estimated mutual information matrix $\widehat{M} = \left[\widehat{I}_n(X_i, X_j)\right]$, we can apply Kruskal's algorithm (equivalently, the Chow-Liu algorithm) to find the best forest structure $\widehat{F}$. The detailed algorithm is described in the following:

Chow-Liu Algorithm for Learning Forest Graphs Initialize $E^{(0)} = \emptyset$ and the desired forest size $K < d$.
Calculate the mutual information matrix $\widehat{M} = \left[\widehat{I}_n(X_i, X_j)\right]$ according to (10.131).
For $k = 1, \ldots, K$

(a) $(i^{(k)}, j^{(k)}) \leftarrow \text{argmax}_{(i,j)}\, \widehat{M}(i, j)$ such that $E^{(k-1)} \cup \{(i^{(k)}, j^{(k)})\}$ does not contain a cycle.

(b) $E^{(k)} \leftarrow E^{(k-1)} \cup \{(i^{(k)}, j^{(k)})\}$.

Output the obtained edge set $E^{(K)}$.

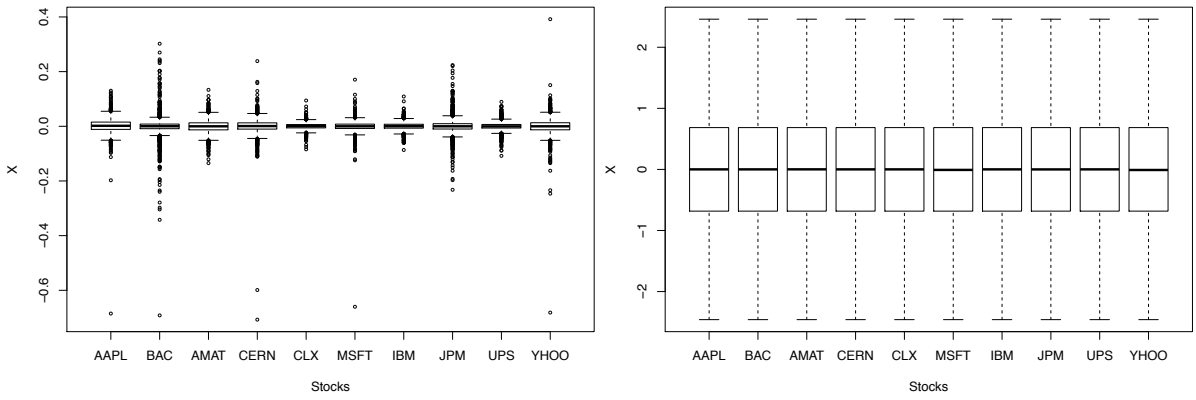
Example 192 (Learning Gaussian maximum weight spanning tree)**.** For Gaussian data $X \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, we know that the mutual information between two variables are

$$I(X_i; X_j) = -\frac{1}{2} \log\left(1 - \rho_{ij}^2\right), \tag{10.132}$$

where ρ_{ij} is the correlation between X_i and X_j. To obtain an empirical estimator, we simply plug-in the sample correlation $\widehat{\rho}_{ij}$. Once the mutual information matrix is calculated, we could apply the Chow-Liu algorithm to get the maximum weight spanning tree.

Example 193 (Graphs for Equities Data)**.** We collect the daily closing prices were obtained for 452 stocks that were consistently in the S&P 500 index between January 1, 2003 through January 1, 2011. This gave us altogether 2,015 data points, each data point

corresponds to the vector of closing prices on a trading day. With $S_{t,j}$ denoting the closing price of stock j on day t, we consider the variables $X_{tj} = \log\left(S_{t,j}/S_{t-1,j}\right)$ and build graphs over the indices j. We simply treat the instances X_t as independent replicates, even though they form a time series. We truncate every stock so that its data points are within six times the mean absolute deviation from the sample average. In Figure 10.23(a) we show boxplots for 10 randomly chosen stocks. It can be seen that the data contains outliers even after truncation; the reasons for these outliers includes splits in a stock, which increases the number of shares. In Figure 10.23(b) we show the boxplots of the data after the nonparanormal transformation (the details of nonparanormal transformation will be explained in the nonparametric graphical model chapter). In this analysis, we use the subset of the data between January 1, 2003 to January 1, 2008, before the onset of the "financial crisis." There are altogether $n = 1,257$ data points and $d = 452$ dimensions.



(a) original data (b) after nonparanormal transformation

Figure 10.23: Boxplots of $X_t = \log(S_t/S_{t-1})$ for 10 stocks. As can be seen, the original data has many outliers, which is addressed by the nonparanormal transformation on the re-scaled data (right).

The 452 stocks are categorized into 10 Global Industry Classification Standard (GICS) sectors, including `Consumer Discretionary` (70 stocks), `Energy` (37 stocks), `Financials` (74 stocks), `Consumer Staples` (35 stocks), `Telecommunications Services` (6 stocks), `Health Care` (46 stocks), `Industrials` (59 stocks), `Information Technology` (64 stocks), `Materials` (29 stocks), and `Utilities` (32 stocks). It is expected that stocks from the same GICS sectors should tend to be clustered together, since stocks from the same GICS sector tend to interact more with each other. In the graphs shown below, the nodes are colored according to the GICS sector of the corresponding stock.

With Gaussian assumption, we directly apply Chow-Liu algorithm to obtain a full spanning tree of $d - 1 = 451$ edges. The resulting graph is shown in Figure 10.24. We see that the stocks from the same GICS sector are clustered very well.

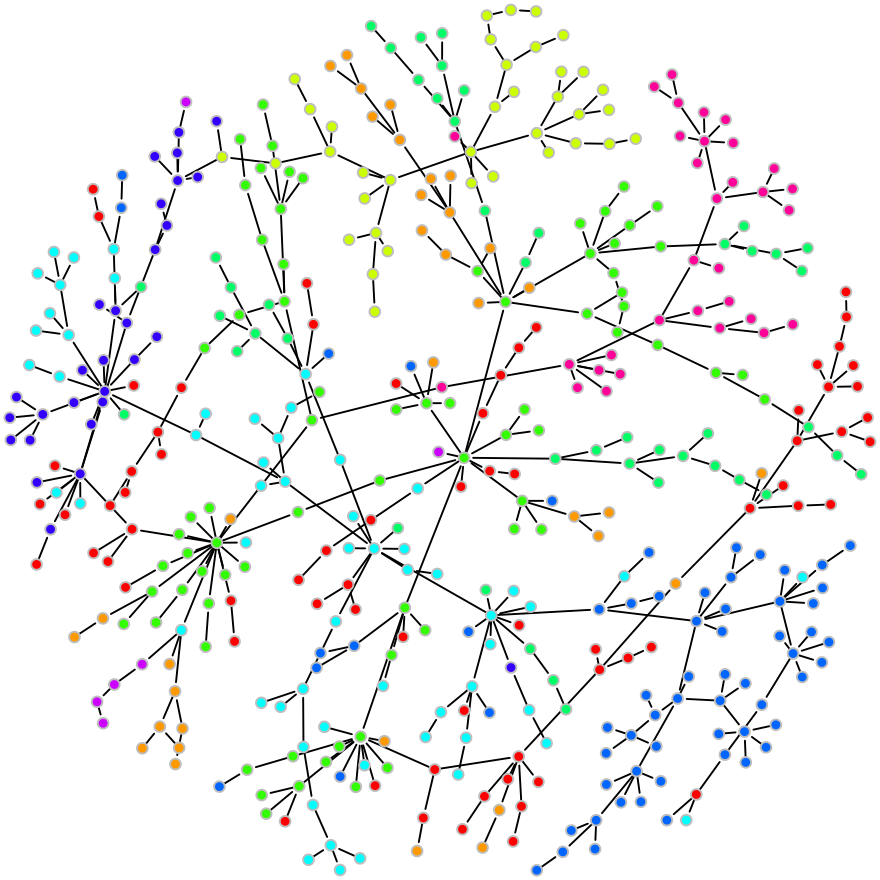


Figure 10.24: Tree graph learned from S&P 500 stock data from Jan. 1, 2003 to Jan. 1, 2008. The graph is estimated using the Chow-Liu algorithm under the Gaussian model. The nodes are colored according to their GICS sector categories.

10.10.3 Neighborhood Selection

Another approach for structure learning is through model selection, which in the context of Gibbs distributions means to determine the structure of the local neighborhood.

Consider for example the problem of selecting the order of a Markov chain. In this case the underlying graph is simply the line (or chain), and the neighborhood of a node contains $m+1$ nodes for an order-m Markov chain. For a more general Gibbs distribution, where the underlying graph is a lattice $\mathbb{Z}^d$, there are many more possibilities for the shape of the local neighborhood.

Consider a translation-invariant field as in Figure 10.25 where the neighborhood Γ^i of node $i \in \mathbb{Z}^d$ is the translation of a neighborhood Γ of the origin, we assume that

$$p(x_i \,|\, x_j,\, j \neq i) = p(x_i \,|\, x_j,\, j \in \Gamma^i). \tag{10.133}$$

[19] proposed an analogue of the Bayesian information criterion. Recall that BIC is usually defined as

$$\text{BIC}(\mathcal{M}) = -\log p(x_1, \ldots, x_n \,|\, \mathcal{M}) + \frac{|\mathcal{M}|}{2} \log n \tag{10.134}$$

where $|\mathcal{M}|$ is the number of parameters in the model $\mathcal{M}$. Then

$$\widehat{\mathcal{M}}_n = \operatorname*{argmin}_{\mathcal{M}} \text{BIC}(\mathcal{M}) \tag{10.135}$$

is the model selected by the BIC criterion.

For Gibbs distributions, the difficulty with this procedure is that the log-likelihood cannot be computed, since it involves the partition function. [19] instead use pseudo-likelihood, and an approximation for the number of parameters (see also [42]). The pseudo-likelihood for a given neighborhood Γ is

$$\ell_n^P(\Gamma) = \sum_{i=1}^{n} \sum_{j=1}^{d} \log p(x_{ij} \,|\, x_{i\Gamma^j}) \tag{10.136}$$

where the parameter estimates are given by simple normalized counts, as multinomial probabilities. This leads to the following pseudo-Bayesian information criterion.

$$\text{PIC}(\Gamma) = -\ell_n^P(\Gamma) + k^{|\Gamma|} \log |\Lambda_n| \tag{10.137}$$

where Λ_n is the subset of $\mathbb{Z}^d$ that the data are over, k is the number of values each variable X_i can take, and Γ is the neighborhood.

Theorem 194 ([19])**.** Let

$$\widehat{\Gamma}_{\Lambda_n} = \operatorname*{argmin}_{\Gamma:\, r(\Gamma) \le r_n} \text{PIC}(\Gamma) \tag{10.138}$$

where $r(\Gamma)$ is the maximum of the absolute value of a component of $i \in \Gamma$, and r_n satisfies $r_n = o\big(\log^{1/2d} |\Lambda_n|\big)$. Then $\mathbb{P}\big(\widehat{\Gamma}_{\Lambda_n} = \Gamma^*\big) \longrightarrow 1$ where Γ^* is the true neighborhood.

10.11 Bibliographic Remarks

Textbooks on undirected graphical models include [94], [26], [53], and [44]. Nice treatments on undirected graphical model inference appears in [11]. More details about the graphical lasso algorithm can be found in [40]. A thorough treatment of undirected graphical models can be found in [49]. Some discussions on chordal graphs and junction trees are drawn from an unpublished notes from Peter Bartlett.

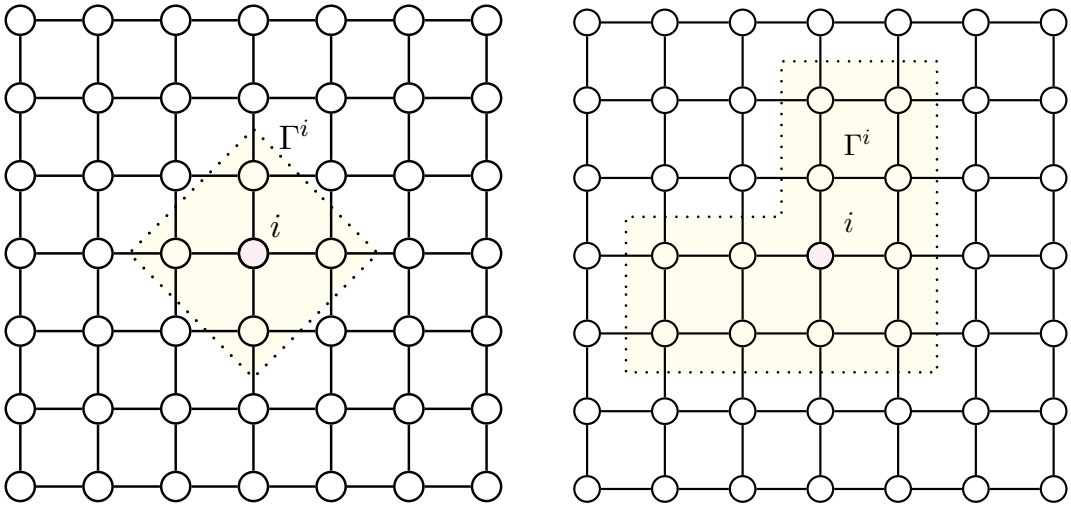


Figure 10.25: Two neighborhood structures for a Gibbs field on $\mathbb{Z}^2$.

Exercises

10.1 Prove Equation (10.37).

10.2 Prove Theorem 168.

10.3 Prove Theorem 172.

10.4 State and prove the continuous version of Hammersley-Clifford theorem.

10.5 Prove that the undirected graph (the square) of Figure 10.12 represents a family of probability distributions that cannot be represented by a directed graph on the same set of vertices.

10.6 Prove Theorem 187.

10.7 Consider random variables (X_1, X_2, X_3, X_4). Suppose the log-density is

$$\log p(x) = \psi_\Phi + \psi_{12}(x_1, x_2) + \psi_{13}(x_1, x_3) + \psi_{24}(x_2, x_4) + \psi_{34}(x_3, x_4). \quad (10.139)$$

(a) Draw the graph G for these variables.

(b) Write down all independence and conditional independence relationships implied by the graph.

(c) Is this model graphical? Is it hierarchical?

10.8 Suppose that the parameters $p(x_1, x_2, x_3)$ are proportional to the following values:

$$\mathbb{P}(0,0,0) = 2, \;\; \mathbb{P}(0,0,1) = 8, \;\; \mathbb{P}(0,1,0) = 4, \;\; \mathbb{P}(0,1,1) = 16, \quad (10.140)$$

$$\mathbb{P}(1,0,0) = 16, \;\; \mathbb{P}(1,0,1) = 128, \;\; \mathbb{P}(1,1,0) = 32, \;\; \mathbb{P}(1,1,1) = 256. \quad (10.141)$$

Find the ψ-terms for the log-linear expansion. Comment on the model.

10.9 Let $X_1, \ldots, X_4$ be binary. Draw the independence graphs corresponding to the following log-linear models (where $\alpha > 0$). Also, identify whether each is graphical and/or hierarchical (or neither).

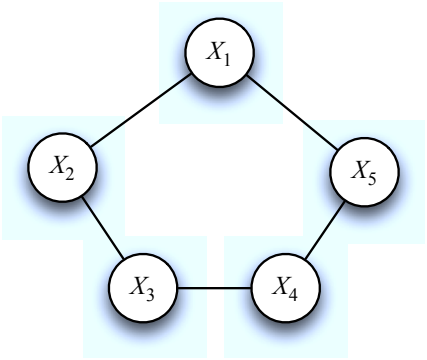
(a) $\log p(x) = \alpha + 11x_1 + 2x_2 + 1.5x_3 + 17x_4$

(b) $\log p(x) = \alpha + 11x_1 + 2x_2 + 1.5x_3 + 17x_4 + 12x_2x_3 + 78x_2x_4 + 3x_3x_4 + 32x_2x_3x_4$

(c) $\log p(x) = \alpha + 11x_1 + 2x_2 + 1.5x_3 + 17x_4 + 12x_2x_3 + 3x_3x_4 + x_1x_4 + 2x_1x_2$

(d) $\log p(x) = \alpha + 5055x_1x_2x_3x_4$.

10.10 This problem is based on the following graph:



(a) Can you construct a DAG that has the same conditional independencies of the form $p(x_i \,|\, x_j, \; j \neq i) = p(x_i \,|\, x_{i_1}, x_{i_2})$ as those implied by the above graph?

(b) For each of the following families of distributions, list any independence relations, if any, that are implied *in addition to* the independence relations for a general distribution that is Markov to the above graph.

(i) $p(x) = \frac{1}{Z}\psi_{12}(x_1, x_2)\, \psi_{23}(x_2, x_3)\, \psi_{34}(x_3, x_4)\, \psi_{45}(x_4, x_5)\, \psi_{51}(x_5, x_1)$

(ii) $p(x) = \frac{1}{Z}\psi_{12}(x_1, x_2)\, \psi_{23}(x_2, x_3)\, \psi_{34}(x_3, x_4)\, \psi_{45}(x_4, x_5)$

(iii) $p(x) = \frac{1}{Z}\psi_{12}(x_1, x_2)\, \psi_{34}(x_3, x_4)\, \psi_{5}(x_5)$

(c) Now suppose that each X_i is *binary*, so that $X_i \in \{0, 1\}$, and consider the following family of distributions:

$$p(x) \;\propto\; \psi_\theta(x_1, x_2)\, \psi_\theta(x_2, x_3)\, \psi_\theta(x_3, x_4)\, \psi_\theta(x_4, x_5)\, \psi_\theta(x_5, x_1)$$

where $\psi_\theta(x, y) = \theta^{xy}$ with $\theta > 0$. For this family of models, calculate the conditional probability $P(X_1 = 1 \,|\, X_3 = 1, X_4 = 1)$ as a function of θ.

10.11 Let $X_1, \ldots, X_6$ be random variables with joint distribution of the form

$$p(x) \;\propto\; f(x_1, x_2, x_3)\, g(x_3, x_4)\, g(x_1, x_5)\, g(x_4, x_5)\, f(x_1, x_5, x_6)$$

where $f : \mathbb{R}^3 \longrightarrow \mathbb{R}_+$ and $g : \mathbb{R}^2 \longrightarrow \mathbb{R}_+$ are arbitrary non-negative functions.

(a) What is the graph with the fewest edges that represents the independence relations for this family of distributions?

(b) For each of the following sets of variables $C \subset \{X_1, \ldots, X_6\}$, give non-empty sets of variables A and B such that $A \perp\!\!\!\perp B \,|\, C$.

(i) $C = \{X_1, X_4\}$

(ii) $C = \{X_1, X_3\}$

(ii) $C = \{X_1, X_3, X_4\}$

(c) Now suppose that $X_i \in \{0, 1\}$ are binary and that

$$f(x, y, z) = \alpha^{xyz} \quad g(x, y) = \alpha^{xy}$$

Find the conditional probability $\boldsymbol{P}(X_4 = 1 \,|\, X_1 = 1, X_3 = 1)$.

10.12 Let $X = (X_1, X_2, X_3, X_4, X_5)$ be a random vector distributed as $X \sim N(0, \boldsymbol{\Sigma})$ where the covariance matrix $\boldsymbol{\Sigma}$ is given by

$$\boldsymbol{\Sigma} = \frac{1}{15}\begin{pmatrix} 9 & -3 & -3 & -3 & -3 \\ -3 & 6 & 1 & 1 & 1 \\ -3 & 1 & 6 & 1 & 1 \\ -3 & 1 & 1 & 6 & 1 \\ -3 & 1 & 1 & 1 & 6 \end{pmatrix} \quad \text{with inverse} \quad \boldsymbol{\Sigma}^{-1} = \begin{pmatrix} 3 & 1 & 1 & 1 & 1 \\ 1 & 3 & 0 & 0 & 0 \\ 1 & 0 & 3 & 0 & 0 \\ 1 & 0 & 0 & 3 & 0 \\ 1 & 0 & 0 & 0 & 3 \end{pmatrix}.$$

(a) What is the graph for X, viewed as an undirected graphical model?

(b) Which of the following independence statements are true?

(a) $X_2 \perp\!\!\!\perp X_3 \,|\, X_1$

(b) $X_3 \perp\!\!\!\perp X_4$

(c) $X_1 \perp\!\!\!\perp X_3 \,|\, X_2$

(d) $X_1 \perp\!\!\!\perp X_5$

(c) List the local Markov properties for this graphical model.

(d) Find the conditional density $p(x_2 \,|\, X_1 = -3)$.

10.13 Consider a chain graph X_1—X_2—X_3—X_4—X_5, we assume all variables are binary. One log-linear model that is consistent with this graph is

$$\log p(x) = \beta_0 + 5\big(x_1x_2 + x_2x_3 + x_3x_4 + x_4x_5\big).$$

Simulate $n = 100$ random vectors from this distribution. Fit the model

$$\log p(x) = \beta_0 + \sum_j \beta_j x_j + \sum_{k<\ell} \beta_{k\ell} x_k x_\ell$$

using maximum likelihood. Report your estimators. Use forward model selection with BIC to choose a submodel. Compare the selected model to the true model.

10.14 Let $X \sim N(0, \mathbf{\Sigma})$ where $X = (X_1, \ldots, X_d)^T$, $d = 10$. Let $\mathbf{\Omega} = \mathbf{\Sigma}^{-1}$ and suppose that $\mathbf{\Omega}(i,i) = 1$, $\mathbf{\Omega}(i, i-1) = .5$, $\mathbf{\Omega}(i-1, i) = .5$ and $\mathbf{\Omega}(i,j) = 0$ otherwise. Simulate 50 random vectors and use the glasso method to estimate the covariance matrix. Compare your estimated graph to the true graph.

10.15 Let $X = (X_1, X_2, X_3, X_4)$ be a random vector satisfying

$$f(X) \sim N(0, \mathbf{\Sigma})$$

where $f(x) = (x_1^3, x_2^3, x_3^3, x_4^3)$ and the covariance matrix $\mathbf{\Sigma}$ is given by

$$\Sigma = \frac{1}{6}\begin{pmatrix} -4 & -2 & -2 & -2 \\ -2 & 4 & 1 & 1 \\ -2 & 1 & 4 & 1 \\ -2 & 1 & 1 & 4 \end{pmatrix}, \quad \text{with inverse} \quad \mathbf{\Sigma}^{-1} = \begin{pmatrix} 3 & 1 & 1 & 1 \\ 1 & 2 & 0 & 0 \\ 1 & 0 & 2 & 0 \\ 1 & 0 & 0 & 2 \end{pmatrix}.$$

(a) Give an expression for the density of X.

(b) What is the graph for X, viewed as a graphical model?

(c) List the local Markov properties for this graphical model.

10.16 Let $X = (X_1, \ldots, X_d)$ where each $X_j \in \{0, 1\}$. Consider the log-linear model

$$\log p(x) = \beta_0 + \sum_{j=1}^{d} \beta_j x_j + \sum_{j<k}^{d} \beta_{jk} x_j x_k + \cdots + \sum_{j<k<\ell}^{d} \beta_{jk\ell} x_j x_k x_\ell + \cdots +$$

Suppose that $\beta_A = 0$ whenever $\{1, 2\} \subset A$. Show that $X_1 \perp\!\!\!\perp X_2 \,|\, X_3, \ldots, X_d$.

10.17 Let $X = (X_1, \ldots, X_d) \in \mathbb{R}^d$ be a random vector, with bivariate marginal densities $p_{ij}(x_i, x_j) > 0$ and univariate marginal densities $p_i(x_i)$. Let $G = (V, E)$ be a tree graph on $\{1, \ldots, d\}$, so that G does not contain any cycles. Consider the family of functions

$$f_m(x_1, \ldots, x_d) = \prod_{i=1}^{d} \theta_i(x_i)^{m_i} \prod_{(i,j)\in E} \theta_{ij}(x_i, x_j),$$

where $m_i \in \mathbb{Z}$ are integers. Find a set of integers $m_1, \ldots, m_d \in \mathbb{Z}$ for which the function f_m is a probability density; i.e., f_m is nonnegative and integrates to one.

10.18 Given $X = (Y, Z) \in \mathbb{R}^6$, let $X \sim N(0, \mathbf{\Sigma})$ be a random Gaussian vector where $Y = (Y_1, Y_2) \in \mathbb{R}^2$ and $Z = (Z_1, Z_2, Z_3, Z_4) \in \mathbb{R}^4$, with $\mathbf{\Sigma}^{-1} = \mathbf{\Omega} = \begin{pmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{B}^T & \boldsymbol{C} \end{pmatrix}$ where

$$\boldsymbol{A} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \quad \boldsymbol{B} = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ -1 & \frac{1}{2} & -\frac{1}{3} & \frac{1}{4} \end{bmatrix} \quad \boldsymbol{C} = \begin{bmatrix} 2 & \frac{1}{2} & 0 & 0 \\ \frac{1}{2} & 2 & 0 & 0 \\ 0 & 0 & 2 & \frac{1}{2} \\ 0 & 0 & \frac{1}{2} & 2 \end{bmatrix}.$$

(a) Draw the undirected graph of X.

(b) Draw the undirected graph of Z. Hint: Recall that

$$\begin{bmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{B}^T & \boldsymbol{C} \end{bmatrix}^{-1} = \begin{bmatrix} \boldsymbol{A}^{-1} + \boldsymbol{A}^{-1}\boldsymbol{B}\boldsymbol{S}^{-1}\boldsymbol{B}^T\boldsymbol{A}^{-1} & -\boldsymbol{A}^{-1}\boldsymbol{B}\boldsymbol{S}^{-1} \\ -\boldsymbol{S}^{-1}\boldsymbol{B}^T\boldsymbol{A}^{-1} & \boldsymbol{S}^{-1} \end{bmatrix}$$

where $\boldsymbol{S} = \boldsymbol{C} - \boldsymbol{B}^T\boldsymbol{A}^{-1}\boldsymbol{B}$ is the Schur complement.

(c) Which of the following independence statements hold?

1. $Y_1 \perp\!\!\!\perp Y_2 \,|\, Z$
2. $Z_1 \perp\!\!\!\perp Z_4 \,|\, Z_2$
3. $Z_1 \perp\!\!\!\perp Z_4 \,|\, Y_1$
4. $Z_1 \perp\!\!\!\perp Z_2$.