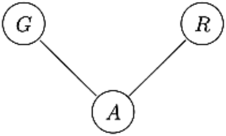
Graphical Models for Discrete Data
Part 2: Directed Graphs

1 Introduction

To introduce directed graphs and their models, we borrow the following example from Edwards ([Edw00]). A market researcher wants to find out who likes noodles, and to do this he interviews a representative sample of people, recording their race (R), gender (G) and answer (A) to the question "Do you like noodles?". Suppose the results are as shown in table 1.

Figure 1: $G \perp\!\!\!\perp R | A$



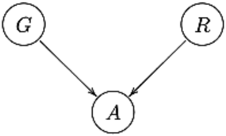
Race	Gender	Do you like noodles?	
		Yes	No
Black	Male	32	86
	Female	35	121
White	Male	61	73
	Female	42	70

Table 1: the noodles data

The simplest undirected graphical model consistent with these data is the one shown in figure 1. But, as Edwards remarks, this model is obviously inappropriate. How can we suppose that race and gender are conditionally independent *given the response*? The respondents' race and gender, characteristics determined decades before, cannot be affected by whether or not they like noodles. Race and gender might me marginally independent, but they can hardly be conditionally independent given the response.

The problem arises because we have not taken the *ordering* of the variables into account. Here race and gender are clearly prior to the response. If we analyse the data using directed graphs and the associated models, then we obtain the graph shown in figure 2.

Figure 2: $G \perp\!\!\!\perp R$



This resembles the previous graph, except that the edges are replaced by arrows pointing towards the response. As we shall see, directed graphs have different rules for the derivation of conditional independence relations. Now the missing arrow between race and gender means that they are marginally independent, not conditionally independent given the response.

2 Definition and Notation

A directed graph is a pair $G = (K, E)$, where K is a set of vertices and E is a set of edges with *ordered* pairs of vertices. If there is an arrow from i to j, then we write this as $i \to j$, or equivalently as $(ij) \in E$. We restrict attention to directed graphs with no directed cycles, i.e. acyclic directed graphs (DAGs). If $i \to j$, then X_i is called a parent of X_j, and X_j is called a child of X_i. The set of coordinates of the parents of X_j is denoted $\mathrm{pa}(j)$, so $X_{\mathrm{pa}(j)}$ denotes the set of parents of X_j. If there is a directed path from i to j, then X_i is called an ancestor of X_j. The set of coordinates of the ancestors of X_j is denoted $\mathrm{an}(j)$. These definitions can be extended to apply

to sets of nodes in the obvious way. For example, for a set $S \subseteq K$ we define $\mathrm{pa}(S) = \bigcup_{i \in S} \mathrm{pa}(i) \setminus S$, that is, the set of nodes not in S that are parent to a node in S. The definition of ancestor is extended similarly. Furthermore we define $\mathrm{an}^{+}(S) = S \cup \mathrm{an}(S)$ to be the *ancestral set* of S.

The absence of any directed cycles is equivalent to the existence of an ordering of the nodes $\{1, 2, \ldots, k\}$ such that $i \to j$ only when $i < j$. In other words, there exists an ordering of the nodes such that arrows point only from lower-numbered nodes to higher-numbered nodes. Suppose that a priori knowledge tells us the variables can be labeled $X_1, X_2, \ldots, X_k$ such that X_i is prior to X_{i+1}. Corresponding to this ordering we can factorize the joint density of $X_1, X_2, \ldots, X_k$ as

$$P(X) = P(X_1)P(X_2 \mid X_1) \cdots P(X_k \mid X_{k-1}, X_{k-2}, \ldots, X_1) \qquad (1)$$

In constructing a DAG, an arrow is drawn from i to j, where $i < j$, unless $P(X_j \mid X_{j-1}, \ldots, X_1)$ does not depend on X_i, in other words, unless

$$i \perp\!\!\!\perp j | \{1, \ldots, j\} \setminus \{i, j\} \qquad (2)$$

This is the key difference between DAGs and undirected graphs. In both types of graph a missing edge between X_i and X_j is equivalent to a conditional independence relation between X_i and X_j. In undirected graphs they are conditionally independent given all the remaining variables, whereas in DAGs they are conditionally independent given all prior variables. Thus in figure 2 the missing arrow between G and R means that $G \perp\!\!\!\perp R$, not that $G \perp\!\!\!\perp R | A$.

Having constructed the DAG from (2), we can write the joint density (1) more elegantly as

$$P(X) = \prod_{i=1}^{k} P(X_i \mid X_{pa(i)}) \qquad (3)$$

The pairwise conditional independence relations corresponding to a missing arrow between i and j can be expressed more elegantly as

$$i \perp\!\!\!\perp j | \mathrm{an}(\{i,j\})$$

3 Interpretation

In this section we discuss the independence properties of directed independence graphs. For undirected graphs, we saw that a simple criterion of separation in the graph-theoretic sense was equivalent to conditional independence in the statistical sense. A similar result is true of DAGs, though the

graph-theoretic property, usually called d-separation, is somewhat more complicated than the separation criterion in undirected graphs. There are in fact two different formulations of the criterion. The original formulation is due to Pearl [Pea86a, Pea86b]. The alternative formulation due to Lauritzen et al. [LDLL90] is discussed here.

To do this we need to define the *moral graph* of a DAG. Given a DAG $G = (K, E)$ we construct the moral graph G^m by marrying parents, and deleting directions, that is,

1. For each $i \in K$, we connect all vertices in $\mathrm{pa}(i)$ with lines.

2. We replace all arrows in E with lines.

Now the directed independence graph G possesses the conditional independence properties of its associated moral graph G^m. We can see this as follows. The joint distribution factorizes as

$$\begin{aligned} P(X) &= \prod_{i=1}^{k} P(X_i \mid X_{pa(i)}) \\ &= \prod_{i=1}^{k} g(X_i, X_{pa(i)}) \qquad (4) \end{aligned}$$

by setting $g(X_i, X_{pa(i)}) = P(X_i \mid X_{pa(i)})$. We thus have an expansion for the joint density function in terms of functions $g(X_a)$ for $a = i \cup \mathrm{pa}(i)$, $i = 1, 2, \ldots, k$. Recall that random vectors X and Y are conditionally independent given Z, $X \perp\!\!\!\perp Y \mid Z$ if and only if there exist functions g and h such that

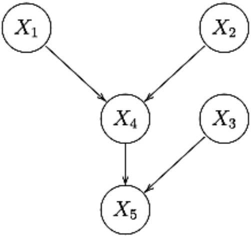
$$P(x, y, z) = g(x, z)h(y, z)$$

for all (x, y) and for all z for which $P(z) > 0$. By application of the factorisation criterion to the expansion (4), we can deduce all pairwise conditional independence statements of the form $i \perp\!\!\!\perp j \mid \text{rest}$. The edges of the undirected independence graph for $P(X)$ are characterised as edges between i and each of its parents, and edges between each pair of parents of i. That is, the edge set of the moral graph, G^m.

Consider for example the directed graph in figure 3. This graph corresponds to the factorisation

$$\begin{aligned} P(X) &= P(X_1)P(X_2)P(X_3)P(X_4|X_1, X_2)P(X_5|X_3, X_4) \\ &= g_1(X_1)g_2(X_2)g_3(X_3)g_4(X_1, X_2, X_4)g_5(X_3, X_4, X_5) \qquad (5) \end{aligned}$$

Figure 3: example directed graph

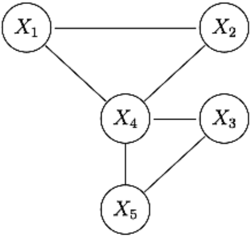


Using the factorisation criterion for conditional independence, we can read the following pairwise independences from this factorisation

$2 \perp\!\!\!\perp 3 |$ rest
$1 \perp\!\!\!\perp 3 |$ rest
$1 \perp\!\!\!\perp 5 |$ rest
$2 \perp\!\!\!\perp 5 |$ rest

Representing these conditional independences in an undirected graph gives the graph in figure 4 which is indeed the moral graph of the graph in figure 3.

Figure 4: moral graph



We can use the moral graph to answer for example the question whether $X_1 \perp\!\!\!\perp X_3|X_5$. Since $\{5\}$ does not separate $\{1\}$ from $\{3\}$ in the moral graph, the answer is "no". It does follow from the moral graph however that $X_1 \perp\!\!\!\perp X_3|X_4$, since $\{4\}$ separates $\{1\}$ from $\{3\}$. The full moral graph can obscure certain independencies however. In this example X_1 and X_2 are independent, but this can not be inferred from the full moral graph. This fact may be deduced though by strengthening the assertion of the equivalence of directed and moral graphs to refer to the graph on the ancestral set of the variables involved in the conditional independence statement. To determine whether X_1 and X_2 are independent, we only have to look at "the smallest marginal distribution that includes them both", and since $P(X_1, X_2) = P(X_1)P(X_2)$ according to the factorisation in (5), it is clear that they are independent.

More generally, suppose we want to check whether $i \perp\!\!\!\perp j|S$ for some set $S \subseteq K$. The first step is to consider the ancestral set of $\{i, j\} \cup S$, that is $\mathrm{an}^+(\{i, j\} \cup S) = A$, say. Since for $i \in A$, $\mathrm{pa}(i) \in A$, we know that the joint distribution of X_A is given by

$$\prod_{i \in A} P(X_i | X_{pa(i)})$$

which corresponds to the subgraph G_A of G. This is a product of factors $P(X_i|X_{pa(i)})$, that is, involving the variables $X_{i \cup pa(i)}$ only. So it factorizes according to G_A^m, and thus the global Markov properties for undirected graphs apply. So, if S separates i and j in G_A^m, then $i \perp\!\!\!\perp j | S$.

The criterion is easily extended to sets of variables, in the following sense. The directed version of the global Markov property states that for three disjoint sets S_1, S_2 and S_3, $S_1 \perp\!\!\!\perp S_2 | S_3$ whenever S_3 separates S_1 and S_2 in G_A^m, where $A = \mathrm{an}^+(S_1 \cup S_2 \cup S_3)$.

4 Maximum Likelihood Estimation of Bayesian Networks

In this section we consider the maximum likelihood estimation of the parameters of a given Bayesian network structure. This turns out to be pretty straightforward: it is a collection of independent multinomial estimation problems. Therefore we start with a discussion of the ML estimation of the parameters of a multinomial distribution. After that we show how this applies to the estimation of the parameters of a Bayesian network.

4.1 ML Estimation for multinomial distribution

We want to estimate the probabilities $p_1, p_2, \ldots, p_J$ of getting outcomes $1, 2, \ldots, J$. If in n trials, we observe n_1 outcomes 1, n_2 of 2, …, n_J of J, then the obvious guess is to estimate $p_j, j = 1, \ldots, J$, by n_j/n. This is also the maximum likelihood estimate because the probability of getting the sequence $x_1, \ldots, x_n$ of outcomes is given by

$$P(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) = p_1^{n_1} p_2^{n_2} \cdots p_J^{n_J}$$

and so the log-likelihood function is

$$\mathcal{L} = n_1 \log p_1 + n_2 \log p_2 + \ldots + n_J \log p_J$$

We want to maximize this expression, but we have to satisfy the constraint

$$p_1 + p_2 + \ldots + p_J = 1$$

To apply the method of Lagrange multipliers, we form the auxiliary function

$$F(p_1, \ldots, p_J, \lambda) = n_1 \log p_1 + n_2 \log p_2 + \ldots + n_J \log p_J + \lambda \left(\sum_{j=1}^{J} p_j - 1 \right)$$

Taking the derivative with respect to p_j, $j = 1, \ldots, J$ and equating to zero we get

$$\frac{n_j}{p_j} + \lambda = 0, \quad j = 1, \ldots, J$$

Solving for p_j

$$p_j = -\frac{n_j}{\lambda}$$

Taking the derivative of F with respect to λ and equating to zero yields

$$p_1 + p_2 + \ldots + p_J - 1 = 0 \qquad (6)$$

Substituting the p_j in 6 gives

$$-\left(\frac{n_1}{\lambda} + \frac{n_2}{\lambda} + \ldots + \frac{n_J}{\lambda} \right) = -\frac{1}{\lambda}(n_1 + n_2 + \ldots + n_J) = 1$$

Hence

$$-\frac{1}{\lambda} = \frac{1}{\sum n_j}$$

So $\lambda = -n$, which immediately gives $p_j = n_j/n$

Consider a random variable X with three possible values, and in a sample of size 100, we observe $n_1 = 20, n_2 = 70, n_3 = 10$. Let p_i denote $P(X = i), i = 1, 2, 3$. The common sense estimator of p_1 is of course to take the relative frequency of the value 1 observed in the sample, i.e. $\hat{p}_1 = 20/100 = 0.2$. Similar reasoning leads to $\hat{p}_2 = 70/100 = 0.7$, and $\hat{p}_3 = 10/100 = 0.1$.

Below we show that our common sense estimates coincide with the maximum likelihood estimates. For any value of p_1, p_2, p_3, the probability of the observed sample is

$$L(p_1, p_2, p_3) = p_1^{20} \times p_2^{70} \times p_3^{10}$$

Therefore the log-likelihood of the observed sample is

$$\mathcal{L}(p_1, p_2, p_3) = 20 \log p_1 + 70 \log p_2 + 10 \log p_3$$

We want to find the values of p_1, p_2, p_3 that maximize the log-likelihood function, subject to the constraint that $p_1 + p_2 + p_3 = 1$.

Rather than using the Lagrange multiplier method, we simply substitute $(1 - p_1 - p_2)$ for p_3 in the log-likelihood function:

$$\mathcal{L}(p_1, p_2) = 20 \log p_1 + 70 \log p_2 + 10 \log(1 - p_1 - p_2)$$

Now we simply take the derivative of $\mathcal{L}$ with respect to p_1 and p_2:

$$\frac{\partial \mathcal{L}}{\partial p_1} = \frac{20}{p_1} - \frac{10}{1 - p_1 - p_2} \qquad \frac{\partial \mathcal{L}}{\partial p_2} = \frac{70}{p_2} - \frac{10}{1 - p_1 - p_2}$$

Here we used the fact that the derivative of $\log x$ is $1/x$. To find the values of p_1 and p_2 for which $\mathcal{L}$ is maximized, we equate the partial derivatives to zero, and solve for p_1 and p_2. Upon doing so, we find $p_1 = 0.2$ and $p_2 = 0.7$ as expected.

4.2 ML estimation of Bayesian Networks

The probability of each observation is given by

$$P(X) = \prod_{i=1}^{k} p(X_i \mid X_{pa(i)})$$

where we use lower case p for the network parameters (probabilities and conditional probabilities). So the joint probability for n independent observations is

$n \quad k$

$$P(X^{(1)}, \ldots, X^{(n)}) = \prod_{j=1} \prod_{i=1} p(X_i^{(j)} \mid X_{pa(i)}^{(j)})$$

If we write $n(x_i, x_{pa(i)})$ for the number of observations with $X_i = x_i$ and $X_{pa(i)} = x_{pa(i)}$, we can write

$$L = \prod_{i=1}^{k} \prod_{x_i, x_{pa(i)}} p(x_i \mid x_{pa(i)})^{n(x_i, x_{pa(i)})}$$

Taking the log-likelihood, this becomes

$$\mathcal{L} = \sum_{i=1}^{k} \sum_{x_i, x_{pa(i)}} n(x_i, x_{pa(i)}) \log p(x_i \mid x_{pa(i)})$$

Assuming the parameters are not related, this boils down to a whole bunch of independent multinomial estimation problems (one for each possible parent configuration). From this it follows that we get maximum likelihood estimates

$$\hat{p}(x_i \mid x_{pa(i)}) = \frac{n(x_i, x_{pa(i)})}{n(x_{pa(i)})}$$

So the value of the likelihood function evaluated at its maximum is

$$\mathcal{L} = \sum_{i=1}^{k} \sum_{x_i, x_{pa(i)}} n(x_i, x_{pa(i)}) \log \frac{n(x_i, x_{pa(i)})}{n(x_{pa(i)})}$$

As an example, consider the data in table 2. Suppose we want to estimate from this data set the network

$$P(X_1, X_2, X_3, X_4) = p_1(X_1) p_2(X_2) p_{3|12}(X_3|X_1, X_2) p_{4|3}(X_4|X_3)$$

Now we have to estimate the following parameters:

obs	X_1	X_2	X_3	X_4
1	1	1	1	1
2	1	1	1	1
3	1	1	2	1
4	1	2	2	1
5	1	2	2	2
6	2	1	1	2
7	2	1	2	3
8	2	1	2	3
9	2	2	2	3
10	2	2	1	3

Table 2: Example data set

$p_1(1)$	$p_1(2) = 1 - p_1(1)$	
$p_2(1)$	$p_2(2) = 1 - p_2(1)$	
$p_{3\|1,2}(1\|1,1)$	$p_{3\|1,2}(2\|1,1) = 1 - p_{3\|1,2}(1\|1,1)$	
$p_{3\|1,2}(1\|1,2)$	$p_{3\|1,2}(2\|1,2) = 1 - p_{3\|1,2}(1\|1,2)$	
$p_{3\|1,2}(1\|2,1)$	$p_{3\|1,2}(2\|2,1) = 1 - p_{3\|1,2}(1\|2,1)$	
$p_{3\|1,2}(1\|2,2)$	$p_{3\|1,2}(2\|2,2) = 1 - p_{3\|1,2}(1\|2,2)$	
$p_{4\|3}(1\|1)$	$p_{4\|3}(2\|1)$	$p_{4\|3}(3\|1) = 1 - p_{4\|3}(1\|1) - p_{4\|3}(2\|1)$
$p_{4\|3}(1\|2)$	$p_{4\|3}(2\|2)$	$p_{4\|3}(3\|2) = 1 - p_{4\|3}(1\|2) - p_{4\|3}(2\|2)$

This means we have to estimate 10 probabilities in total. The contribution of observation 1 to the likelihood function is

$$L(1,1,1,1) = p_1(1)p_2(1)p_{3|1,2}(1|1,1)p_{4|3}(1|1)$$

Likewise, the contribution of observation 3 to the likelihood function is

$$L(1,1,2,1) = p_1(1)p_2(1)(1 - p_{3|1,2}(1|1,1))p_{4|3}(1|2)$$

Their joint contribution is

$$p_1(1)^2 p_2(1)^2 p_{3|1,2}(1|1,1)(1 - p_{3|1,2}(1|1,1))p_{4|3}(1|1)p_{4|3}(1|2)$$

Doing this for all observations, we get

$$\begin{aligned} L(\mathcal{D}) \;=\; & p_1(1)^5(1-p_1(1))^5 p_2(1)^6(1-p_2(1))^4 p_{3|1,2}(1|1,1)^2(1-p_{3|1,2}(1|1,1)) \\ & (1-p_{3|1,2}(1|1,2))^2 p_{3|1,2}(1|2,1)(1-p_{3|1,2}(1|2,1))^2 p_{3|1,2}(1|2,2)(1-p_{3|1,2}(1|2,2)) \end{aligned}$$

$$p_{4|3}(1|1)^2 p_{4|3}(2|1)(1 - p_{4|3}(1|1) - p_{4|3}(2|1))$$
$$p_{4|3}(1|2)^2 p_{4|3}(2|2)(1 - p_{4|3}(1|2) - p_{4|3}(2|2))^3$$

Or in log form

$$\begin{aligned}
\mathcal{L}(\mathcal{D}) &= 5\log p_1(1) + 5\log(1 - p_1(1)) + 6\log p_2(1) + 4\log(1 - p_2(1)) \\
&\quad +2\log p_{3|1,2}(1|1,1) + \log(1 - p_{3|1,2}(1|1,1)) \\
&\quad +2\log(1 - p_{3|1,2}(1|1,2)) + \log p_{3|1,2}(1|2,1) + 2\log(1 - p_{3|1,2}(1|2,1)) \\
&\quad + \log p_{3|1,2}(1|2,2) + \log(1 - p_{3|1,2}(1|2,2)) \\
&\quad +2\log p_{4|3}(1|1) + \log p_{4|3}(2|1) + \log(1 - p_{4|3}(1|1) - p_{4|3}(2|1)) \\
&\quad +2\log p_{4|3}(1|2) + \log p_{4|3}(2|2) + 3\log(1 - p_{4|3}(1|2) - p_{4|3}(2|2))
\end{aligned}$$

This looks like a very complicated function to optimize at first sight, but upon closer inspection we see that it decomposes into a number of unrelated optimization problems. In fact, only parameters corresponding to the same parent configuration are related, because they are constrained to sum to one. Otherwise, we can optimize all parameters separately. For example, to find the optimal value of $p_1(1)$, we simply find the value that maximizes $p_1(1)^5(1 - p_1(1))^5$, regardless of the other parameters. Hence, in the end we just have a bunch of multinomial estimation problems (one for each parent configuration), which we know how to solve.
For example

$$\hat{p}_1(1) = \frac{n(x_1 = 1)}{n} = \frac{5}{10} \qquad \hat{p}_{3|1,2}(2|1,2) = \frac{n(x_1 = 1, x_2 = 2, x_3 = 2)}{n(x_1 = 1, x_2 = 2)} = 1$$

5 Estimation from Incomplete Data

In this section we consider the problem of maximum likelihood estimation of a Bayesian network when we have incomplete data, that is some values are missing. We partition the complete data in an observed part X_{obs} and a missing part X_{mis}, i.e. $X = (X_{obs}, X_{mis})$. For observation j we write $X^{(j)} = (X^{(j)}_{obs}, X^{(j)}_{mis})$.

First we should be clear about what it is we want to maximize: we want to find those parameter values that maximize the probability of the *observed* data. This means that if some values are missing, we have to obtain the marginal probability of the observed data by summing out the missing data. For observation j, the probability thus is:

$$P(X_{obs}^{(j)}) = \sum_{X_{mis}^{(j)}} P(X^{(j)})$$

For all observations together the probability is

$$\prod_{j=1}^{n} P(X_{obs}^{(j)}) = \prod_{j=1}^{n} \left(\sum_{X_{mis}^{(j)}} P(X^{(j)}) \right)$$

So for example, if we have three binary variables $X = (X_1, X_2, X_3)$, and we have an observation $(1, 0, ?)$, the probability is

$$P(1, 0, ?) = P(1, 0, 0) + P(1, 0, 1)$$

and for observation $(?, 1, ?)$ the probability is

$$P(?, 1, ?) = P(0, 1, 0) + P(0, 1, 1) + P(1, 1, 0) + P(1, 1, 1)$$

So if we have a Bayesian network with X_1 and X_2 the parents of X_3 (see figure 5), then the probability of $(1, 0, ?)$ is

$$\begin{aligned} P(1, 0, ?) &= P(1, 0, 0) + P(1, 0, 1) \\ &= p_1(1)p_2(0)p_{3|12}(0|1, 0) + p_1(1)p_2(0)p_{3|12}(1|1, 0) \\ &= p_1(1)p_2(0) \end{aligned}$$

since $p_{3|12}(0|1, 0) + p_{3|12}(1|1, 0) = 1$. In this case, we still have a closed form solution. Suppose however that we an observation $(1, ?, 0)$. Its probability is

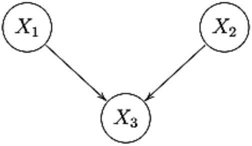
$$\begin{aligned} P(1, ?, 0) &= P(1, 0, 0) + P(1, 1, 0) \\ &= p_1(1)p_2(0)p_{3|12}(0|1, 0) + p_1(1)p_2(1)p_{3|12}(0|1, 1) \end{aligned}$$

Now if we want to maximize the log-likelihood, we get a sum of parameters inside the log, making analytical maximization impossible.

Therefore direct maximization of the observed data likelihood is complicated: in most cases there is no closed form solution of the ML estimates as in the complete data case.

There is however an ingenious iterative scheme to compute the ML estimates, called Expectation Maximization (EM). EM [DLR77] is a general method for doing maximum likelihood estimation with incomplete data. The computational scheme consists of the alternated application of an Expectation step and a Maximization step; hence the name EM. In the E-step, the expected value of the complete-data loglikelihood is calculated, by integrating over the possible values of the missing data under its distribution given the current parameter estimate $\theta^{(t)}$ and the observed data. In the M-step we choose the value of $\theta^{(t+1)}$ that maximizes the loglikelihood in the last E-step. It can be shown that under mild conditions the sequence $\theta^{(0)}, \theta^{(1)}, \ldots$ converges to a maximum likelihood estimate of the observed data likelihood.

Figure 5: A Simple Bayesian Network



Application of the EM-algorithm to Bayesian networks is conceptually straightforward. We proceed as follows:

1. Pick initial values for network paramaters.

2. Use inference to find the expected values of the sufficient statistics.

3. Compute new estimates using the expected values of the sufficient statistics.

4. If convergenced then stop, otherwise return to (2).

Now inference in a Bayesian network is a complicated affair, and clever algorithms have been developed to do this efficiently. We won't go into that; if your interested, follow the course *probabilistic reasoning*. We will do inference the simple way by computing the full joint distribution and computing any probability we might need from that. For large networks this is of course computationally not feasible.

To illustrate how EM works, we consider an extremely simple Bayesian network: just one binary parent and a binary child (see figure 6).

Now suppose we pick the following initial values for the network parameters: $\hat{p}^{(0)}(X_1 = 1) = 0.8$, $\hat{p}^{(0)}(X_2 = 1|X_1 = 1) = 0.6$, $\hat{p}^{(0)}(X_2 = 1|X_1 = 0) = 0.2$. This gives the joint distribution $\hat{P}^{(0)}$ as given in the left part of table 3.

We observe data as given in table 4. For the incomplete cases, columns 3 and 4 give the probabilities of different completions given the initial parameter estimates. For example, the probability that $(0, ?)$ is completed to $(0, 0)$ is equal to 0.8 because $\hat{p}^{(0)}(X_2 = 0|X_1 = 0) = 0.8$.

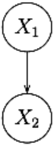
x_1, x_2	$\hat{P}^{(0)}(x_1, x_2)$	$\hat{P}^{(1)}(x_1, x_2)$
(0,0)	$0.2 \times 0.8 = 0.16$	$0.24 \times 0.64 = 0.1536$
(0,1)	$0.2 \times 0.2 = 0.04$	$0.24 \times 0.36 = 0.0864$
(1,0)	$0.8 \times 0.4 = 0.32$	$0.76 \times 0.36 = 0.2736$
(1,1)	$0.8 \times 0.6 = 0.48$	$0.76 \times 0.64 = 0.4864$

Table 3: Joint distribution of (X_1, X_2). Left: on basis of initial parameter estimates. Right: on basis of parameter estimates after one iteration.

x_1, x_2	count		
(0,0)	12		
(0,1)	8		
(1,0)	20		
(1,1)	40		
(0,?)	2	$\hat{P}^{(0)}(X_2 = 0 \mid X_1 = 0) = 0.8$	$\hat{P}^{(0)}(X_2 = 1 \mid X_1 = 0) = 0.2$
(1,?)	8	$\hat{P}^{(0)}(X_2 = 0 \mid X_1 = 1) = 0.4$	$\hat{P}^{(0)}(X_2 = 1 \mid X_1 = 1) = 0.6$
(?,0)	6	$\hat{P}^{(0)}(X_1 = 0 \mid X_2 = 0) = 0.33$	$\hat{P}^{(0)}(X_1 = 1 \mid X_2 = 0) = 0.67$
(?,1)	4	$\hat{P}^{(0)}(X_1 = 0 \mid X_2 = 1) = 0.077$	$\hat{P}^{(0)}(X_1 = 1 \mid X_2 = 1) = 0.923$

Table 4: Observed data, and results of inference on basis of initial estimates.

Figure 6: Simple BN for EM example.



Now we can compute the expected values of the sufficient statistics. For example, the observation $(?, 0)$ contributes for 0.67 to $\hat{n}_1(1)$, because

$$\hat{P}^{(0)}(X_1 = 1 | X_2 = 0) = \frac{\hat{P}^{(0)}(X_1 = 1, X_2 = 0)}{\hat{P}^{(0)}(X_2 = 0)} = \frac{0.32}{0.32 + 0.16} = 0.67.$$

Continuing in this fashion, we get the following expected sufficient statistics:

$$\begin{aligned}
\hat{n}_1(1) &= 20 + 40 + 8 + 6 \times 0.67 + 4 \times 0.923 = 75.712 \\
\hat{n}_1(0) &= 100 - 75.712 = 24.288 \\
\hat{n}_{12}(0,0) &= 12 + 2 \times 0.8 + 6 \times 0.33 = 15.58 \\
\hat{n}_{12}(0,1) &= 24.288 - 15.58 = 8.708 \\
\hat{n}_{12}(1,0) &= 20 + 8 \times 0.4 + 6 \times 0.67 = 27.22 \\
\hat{n}_{12}(1,1) &= 75.712 - 27.22 = 48.492
\end{aligned}$$

From these expected sufficient statistics, we compute the new parameter estimates:

$$\begin{aligned}
\hat{p}^{(1)}(X_1 = 1) &= \frac{\hat{n}_1(1)}{n} = \frac{75.712}{100} \approx 0.76 \\
\hat{p}^{(1)}(X_2 = 1 | X_1 = 1) &= \frac{\hat{n}_{12}(1,1)}{\hat{n}_1(1)} = \frac{48.492}{75.712} \approx 0.64 \\
\hat{p}^{(1)}(X_2 = 1 | X_1 = 0) &= \frac{\hat{n}_{12}(0,1)}{\hat{n}_1(0)} = \frac{8.708}{24.288} \approx 0.36
\end{aligned}$$

From these new parameter estimates, we compute the new joint distribution as given in the right part of table 3. Then we perform inference again using $\hat{P}^{(1)}$, and compute the new expected values of the sufficient statistics. This procedure is iterated until the parameter estimates converge. In figure 7 the iterates of $\hat{p}(X_1 = 1)$ are shown; the sequence converges to approximately 0.7515.

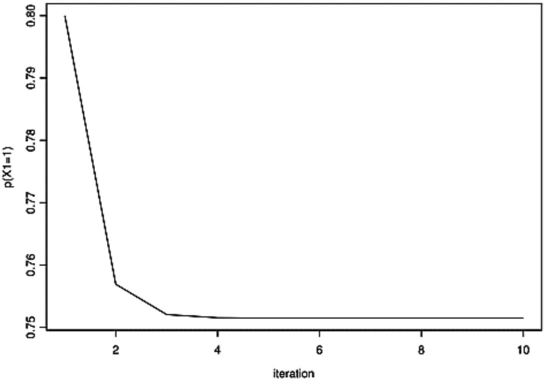


Figure 7: EM iterations for $\hat{p}(X_1 = 1)$.

6 Model Selection with Complete Data

We have seen that maximum likelihood estimation of a given Bayesian network structure is pretty straightforward: you could do the required calculations by hand if the dataset is not too big. In many cases the structure is not known however, so we would like to use the data to find a good structure. The approach we take is very similar to the one for undirected graphs: we define a measure for the quality of a structure, and then search for a model with high quality. As you probably know by now, it is not a good idea to use a quality measure that only takes into account how well the model fits the data. This is a sure way to get an overfitted model; in fact the saturated

Algorithm 1 EM for Bayesian Networks

```
$\hat{\mathbf{p}}^{(0)} \leftarrow$ available case estimates of parameters
$t \leftarrow 0$
repeat
    for all $x_i, x_{pa(i)}$ do
        $\hat{n}^{(t+1)}(x_i, x_{pa(i)}) \leftarrow \sum_{j=1}^{n} P(X_i = x_i, X_{pa(i)} = x_{pa(i)} | X^{(j)}_{obs}, \hat{\mathbf{p}}^{(t)})$
        $\hat{n}^{(t+1)}(x_{pa(i)}) \leftarrow \sum_{x_i} \hat{n}^{(t+1)}(x_i, x_{pa(i)})$
        $\hat{p}^{(t+1)}(x_i | x_{pa(i)}) \leftarrow \hat{n}^{(t+1)}(x_i, x_{pa(i)}) / \hat{n}^{(t+1)}(x_{pa(i)})$
    end for
    $t \leftarrow t + 1$
until $\sum |\hat{\mathbf{p}}^{(t)} - \hat{\mathbf{p}}^{(t-1)}| < \varepsilon$
return $\hat{\mathbf{p}}$
```

model (a fully connected graph) will always have the best fit. We saw this problem before, and one way to deal with it is to include a penalty term for the complexity of the model. Let $\mathcal{L}^M$ denote the value of the log-likelihood function evaluated at $\hat{p}^M$; the ML estimates of p under model M. Akaike's Information Criterion would give the following quality for model M:

$$\text{AIC}(M) = -2\mathcal{L}^M + 2\text{dim}(M)$$

where $\text{dim}(M)$ is the number of parameters of the model. It is customary to define the criterion in such a way that a higher value means higher quality, so we divide by -2 to get

$$\text{AIC}(M) = \mathcal{L}^M - \text{dim}(M)$$

AIC gives a relatively low penalty for complexity, and therefore has a tendency to select overly complex models. A more popular quality measure for Bayesian networks is the Bayesian Information Criterion (BIC):

$$\text{BIC}(M) = \mathcal{L}^M - \frac{\log n}{2}\text{dim}(M)$$

This measure has an asymptotic justification from a Bayesian statistics viewpoint, but we won't be bothered with that here. This score can also be justified by the Minimum Discription Length (MDL) principle, but again we omit the details.

So now we have a well-defined optimization problem. Given

1. Training data
2. Scoring function (BIC)
3. Space of possible models (all DAG's)

find a network (or the networks) that maximizes the score. Unfortunately, finding the maximal scoring network structure (i.e. model) is NP-hard in general. This means that for all practical purposes we have to resort to heuristic search algorithms. We define which models are neighbours of a given model (typically: addition, removal, reversal of an arc) and then traverse the search space looking for high scoring models. The simplest approach is to use a greedy hill-climbing search. This works as follows. Start with a given network (e.g. the empty network, or a random network), and compute the score of this network and all its neighbours. Then apply the change that leads to the biggest improvement in the score. Compute the score of all neighbours of the new model, and again apply the change that gives the biggest improvement in the score, and so on. The iteration stops when none of the neighbours improves the score. You might get stuck in local maxima. One way to escape local maxima is to use random restarts.

An important observation is that the score is *decomposable*: it is a sum of terms, where each term contains the variables $i \cup \mathrm{pa}(i)$. This means that when we move from one model to another, we don't have to compute the score all over again. We only have to recompute the score for those variables for which the parent set has changed. This means the score computations can be done efficiently. As an example, consider again the data in table 2. Suppose the current model is:

$$P(X_1, X_2, X_3, X_4) = p_1(X_1)p_2(X_2)p_{3|12}(X_3|X_1, X_2)p_{4|3}(X_4|X_3)$$

Now suppose we consider adding an edge from X_1 to X_2. Only the parent set of X_2 changes, but the rest of the score is unaffected. The part of the log-likelihood score of the current model that is affected by adding an arc from X_1 to X_2 is boxed in the formula below:

$$\begin{aligned}
\mathcal{L}(\mathcal{D}) &= 5\log\frac{5}{10} + 5\log\frac{5}{10} + \boxed{6\log\tfrac{6}{10} + 4\log\tfrac{4}{10}} \\
&\quad +2\log\frac{2}{3} + \log\frac{1}{3} \\
&\quad +2\log 1 + \log\frac{1}{3} + 2\log\frac{2}{3} \\
&\quad +\log\frac{1}{2} + \log\frac{1}{2} \\
&\quad +2\log\frac{2}{4} + \log\frac{1}{4} + \log\frac{1}{4} \\
&\quad +2\log\frac{2}{6} + \log\frac{1}{6} + 3\log\frac{3}{6} \approx -29.09
\end{aligned}$$

After we add an edge from X_1 to X_2 the log-likelihood score becomes:

$$\begin{aligned}
\mathcal{L}(\mathcal{D}) &= 5\log\frac{5}{10} + 5\log\frac{5}{10} + \boxed{3\log\tfrac{3}{5} + 2\log\tfrac{2}{5} + 3\log\tfrac{3}{5} + 2\log\tfrac{2}{5}} \\
&\quad +2\log\frac{2}{3} + \log\frac{1}{3} \\
&\quad +2\log 1 + \log\frac{1}{3} + 2\log\frac{2}{3} \\
&\quad +\log\frac{1}{2} + \log\frac{1}{2} \\
&\quad +2\log\frac{2}{4} + \log\frac{1}{4} + \log\frac{1}{4}
\end{aligned}$$

$$+2\log\frac{2}{6}+\log\frac{1}{6}+3\log\frac{3}{6}\approx -29.09$$

In this particular case the score doesn't increase, because X_1 and X_2 are independent in the data. Since the model with the extra edge has one extra parameter, it scores lower on AIC or BIC.

Now suppose we consider adding an edge from X_1 to X_4. Again we boxed the part of the log-likelihood score that would be affected by this.

$$\begin{aligned}\mathcal{L}(\mathcal{D}) &= 5\log\frac{5}{10}+5\log\frac{5}{10}+6\log\frac{6}{10}+4\log\frac{4}{10}\\ &\quad +2\log\frac{2}{3}+\log\frac{1}{3}\\ &\quad +2\log 1+\log\frac{1}{3}+2\log\frac{2}{3}\\ &\quad +\log\frac{1}{2}+\log\frac{1}{2}\\ &\quad +\boxed{2\log\tfrac{2}{4}+\log\tfrac{1}{4}+\log\tfrac{1}{4}}\\ &\quad +\boxed{2\log\tfrac{2}{6}+\log\tfrac{1}{6}+3\log\tfrac{3}{6}}\approx -29.09\end{aligned}$$

If we add the edge, the log-likelihood score becomes:

$$\begin{aligned}\mathcal{L}(\mathcal{D}) &= 5\log\frac{5}{10}+5\log\frac{5}{10}+6\log\frac{6}{10}+4\log\frac{4}{10}\\ &\quad +2\log\frac{2}{3}+\log\frac{1}{3}\\ &\quad +2\log 1+\log\frac{1}{3}+2\log\frac{2}{3}\end{aligned}$$

$$
\begin{aligned}
&+\log\frac{1}{2}+\log\frac{1}{2} \\
&+\boxed{2\log 1+2\log\tfrac{2}{3}+\log\tfrac{1}{3}} \\
&+\boxed{\log\tfrac{1}{2}+\log\tfrac{1}{2}+3\log 1}\approx -22.16
\end{aligned}
$$

We see this leads to an improvement of the log-likelihood score, and depending on the complexity penalty to an improvement of the overall score. We added 4 parameters and improved the log-likelihood score by $-22.16 + 29.09 = 6.93$. Both AIC and BIC give the more complex model a higher score in this case.

Algorithm 2 gives the pseudo-code for a simple Bayesian Network structure learning algorithm.

Algorithm 2 BN Structure Learning

```
$G \leftarrow$ initial graph
max $\leftarrow$ score($G$)
repeat
  nb $\leftarrow$ neighbours($G$)
  for all $G' \in$ nb do
    if score ($G'$) > max then
      max $\leftarrow$ score($G'$)
      $G \leftarrow G'$
    end if
  end for
until no change to $G$
return $G$
```

Appendix: The EM-algorithm

The distribution of the complete data $X = (X_{obs}, X_{mis})$ can be factored as

$$P(X|\theta) = P(X_{obs}|\theta)P(X_{mis}|X_{obs}, \theta) \tag{7}$$

Viewing each term as a function of θ it follows that

$$\mathcal{L}(\theta|X) = \mathcal{L}(\theta|X_{obs}) + \log P(X_{mis}|X_{obs}, \theta) \tag{8}$$

where $\mathcal{L}(\theta|X) = \log P(X|\theta)$ denotes the complete-data loglikelihood and $\mathcal{L}(\theta|X_{obs}) = \log L(\theta|X_{obs})$ the observed-data loglikelihood. Because X_{mis} is unknown, we cannot calculate the second term on the right-hand side of equation (8), so instead we take the average of (8) over the predictive distribution $P(X_{mis}|X_{obs}, \theta^{(t)})$, where $\theta^{(t)}$ is a preliminary estimate of the unknown parameter θ. Taking the expectation left and right with respect to $P(X_{mis}|X_{obs}, \theta^{(t)})$ yields

$$Q(\theta|\theta^{(t)}) = \mathcal{L}(\theta|X_{obs}) + H(\theta|\theta^{(t)}) \tag{9}$$

where

$$\begin{aligned} Q(\theta|\theta^{(t)}) &= E_{\theta^{(t)}}\mathcal{L}(\theta|X) \\ &= \int \mathcal{L}(\theta|X)P(X_{mis}|X_{obs}, \theta^{(t)})dX_{mis} \end{aligned}$$

is the expected complete-data loglikelihood, and

$$H(\theta|\theta^{(t)}) = \int \log P(X_{mis}|X_{obs}, \theta)P(X_{mis}|X_{obs}, \theta^{(t)})dX_{mis}$$

A central result of [DLR77] is that if we let $\theta^{(t+1)}$ be the value of θ that maximizes $Q(\theta|\theta^{(t)})$, then $\theta^{(t+1)}$ is a better estimate than $\theta^{(t)}$ in the sense that its observed-data loglikelihood is at least as high as that of $\theta^{(t)}$,

$$\mathcal{L}(\theta^{(t+1)}) \geq \mathcal{L}(\theta^{(t)})$$

This can be seen by writing

$$\mathcal{L}(\theta^{(t+1)}|X_{obs}) - \mathcal{L}(\theta^{(t)}|X_{obs}) \;\; = \;\; \{Q(\theta^{(t+1)}|\theta^{(t)}) - Q(\theta^{(t)}|\theta^{(t)})\}$$

$$-\{H(\theta^{(t+1)}|\theta^{(t)}) - H(\theta^{(t)}|\theta^{(t)})\}$$

The first difference is nonnegative since $\theta^{(t+1)}$ is chosen so that

$$Q(\theta^{(t+1)}|\theta^{(t)}) \geq Q(\theta|\theta^{(t)})$$

for all θ. It remains to show that the second difference is nonpositive, that is

$$H(\theta^{(t+1)}|\theta^{(t)}) - H(\theta^{(t)}|\theta^{(t)}) \leq 0$$

Now for any θ

$$\begin{aligned}
& H(\theta^{(t+1)}|\theta^{(t)}) - H(\theta^{(t)}|\theta^{(t)}) \\
= & \int \log P(X_{mis}|X_{obs},\theta) P(X_{mis}|X_{obs},\theta^{(t)}) dX_{mis} \\
& - \int \log P(X_{mis}|X_{obs},\theta^{(t)}) P(X_{mis}|X_{obs},\theta^{(t)}) dX_{mis} \\
= & \int \log \frac{P(X_{mis}|X_{obs},\theta)}{P(X_{mis}|X_{obs},\theta^{(t)})} P(X_{mis}|X_{obs},\theta^{(t)}) dX_{mis} \\
= & E_{\theta^{(t)}} \left\{ \log \frac{P(X_{mis}|X_{obs},\theta)}{P(X_{mis}|X_{obs},\theta^{(t)})} \right\} \\
\leq & \log E_{\theta^{(t)}} \left\{ \frac{P(X_{mis}|X_{obs},\theta)}{P(X_{mis}|X_{obs},\theta^{(t)})} \right\} \\
= & \log \int \frac{P(X_{mis}|X_{obs},\theta)}{P(X_{mis}|X_{obs},\theta^{(t)})} P(X_{mis}|X_{obs},\theta^{(t)}) dX_{mis} \\
= & \log \int P(X_{mis}|X_{obs},\theta) dX_{mis} \\
= & 0
\end{aligned}$$

where the inequality is a consequence of Jensen's inequality and the concavity of the logarithmic function.

Thus, we have established that the observed-data likelihood is not decreased after an EM iteration. So for a bounded sequence of likelihood values $\{\theta^{(t)}\}$, $\theta^{(t)}$ converges monotonically to some L^*, which is almost always a stationary value of L. Moreover, in many practical applications, L^* will be a local maximum. For a detailed discussion of the convergence properties of

EM, see [MK97].

A Simple Example

We illustrate the EM-algorithm with a particularly simple example that does not require EM for its solution. This allows us to discuss the computational steps of EM without being distracted by technical detail. Consider a sequence of 4 independent coin tosses with the following outcome (1,1,0,?), where we use 1 to denote that heads has come up, and 0 for tails. The question mark for the fourth toss indicates that its outcome was not observed for some reason. The parameter of interest is the probability of heads, which we denote by θ. We partition the complete data X into the observed part and the missing part, i.e. $X = (X_{obs}, X_{mis})$. The probability of the observed data is obtained from the probability of the complete data by summing out the missing data, i.e.

$$P(X_{obs} \mid \theta) = \sum_{X_{mis}} P(X \mid \theta) = P((1,1,0,0) \mid \theta) + P((1,1,0,1) \mid \theta) =$$
$$\theta^3(1-\theta) + \theta^2(1-\theta)^2 = \theta^2(1-\theta)\{\theta + (1-\theta)\} = \theta^2(1-\theta),$$

since $\theta + (1-\theta) = 1$. As was to be expected the observed data likelihood reduces to the likelihood obtained by ignoring the fourth toss altogether. Hence the maximum likelihood estimate is simply the fraction of heads observed, i.e. $\hat{\theta} = 2/3$.

For illustratory purposes we consider how we would arrive at this estimate using the EM computational scheme. In the E-step we form the expected complete-data loglikelihood based on the current estimate $\theta^{(t)}$,

$$\begin{aligned} Q(\theta|\theta^{(t)}) &= \theta^{(t)}(3\log\theta + \log(1-\theta)) + (1-\theta^{(t)})(2\log\theta + 2\log(1-\theta)) \\ &= (2+\theta^{(t)})\log\theta + (1 + (1-\theta^{(t)}))\log(1-\theta) \end{aligned}$$

Since we choose $\theta^{(t+1)}$ to maximize this function with respect to θ, it is obvious that

$$Q(\theta^{(t+1)}|\theta^{(t)}) \geq Q(\theta^{(t)}|\theta^{(t)})$$

Now the observed-data loglikelihood is

$$\mathcal{L}(\theta|X_{obs}) = 2\log\theta + \log(1-\theta)$$

and

$$H(\theta|\theta^{(t)}) = \theta^{(t)}\log\theta + (1-\theta^{(t)})\log(1-\theta)$$

Verify that

$$Q(\theta|\theta^{(t)}) = \mathcal{L}(\theta|X_{obs}) + H(\theta|\theta^{(t)})$$

Now

$$\begin{aligned}\frac{d}{d\theta}H(\theta|\theta^{(t)}) &= \frac{d}{d\theta}\left\{\theta^{(t)}\log\theta + (1-\theta^{(t)})\log(1-\theta)\right\} \\ &= \frac{\theta^{(t)}}{\theta} - \frac{1-\theta^{(t)}}{1-\theta}\end{aligned}$$

Equating to zero and solving for θ yields $\theta = \theta^{(t)}$, and hence

$$H(\theta|\theta^{(t)}) \leq H(\theta^{(t)}|\theta^{(t)})$$

for any value of θ.

In this particularly simple case one may obtain a closed-form solution for the iterates: $\theta^{(t+1)} = 1/2 + 1/4\ \theta^{(t)}$. Thus if we make an initial guess $\theta^{(0)} = 0.25$, we obtain the sequence $0.2500, 0.5625, 0.6406, 0.6602, 0.6650, \ldots$, which converges to $2/3$.

References

[DLR77] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society B*, 39:1–38, 1977.

[Edw00] D. Edwards. *Introduction to Graphical Modelling (second edition).* Springer, New York, 2000.

[LDLL90] S.L. Lauritzen, A.P. Dawid, B.N. Larsen, and H.-G. Leimer. Independence properties of directed markov fields. *Networks*, 20:491–505, 1990.

[MK97] G.J. McLachlan and T. Krishnan. *The EM algorithm and extensions.* Wiley, New York, 1997.

[Pea86a] J. Pearl. A constraint propagation approach to probabilistic reasoning. In *Uncertainty in Artificial Intelligence*, pages 357–370. North-Holland, 1986.

[Pea86b] J. Pearl. Fusion, propagation and structuring in belief networks. *Artificial Intelligence*, 29:241–288, 1986.