Graphical Models for Discrete Data
Part 1: Undirected Graphs

1 Introduction

In this chapter we consider models that aim to represent the associations between a number of discrete variables. In contrast to for example classification trees we don't select a particular variable as a target that is to be explained or predicted by the other variables. Instead, all variables are treated on an equal footing: we simply want to model the associations between them. We confine our attention to discrete variables, although similar ideas have been developed for continuous as well as mixed discrete and continuous variables.

After giving a motivating example, we give a short review of the notions of independence and conditional independence of random variables. These notions are central to the interpretation of the type of models we are going to discuss. Next we start with the so called log-linear representation of a multi-way contingency table. This representation is convenient for our purpose because it allows us to express (conditional) independence constraints by setting certain coefficients equal to zero. In fact we define subclasses of the log-linear model that can be fully interpreted in terms of conditional independence relations. These are in order of inclusion: hierarchical models, graphical models and finally decomposable models. We discuss how such models can be estimated from data, sometimes requiring an iterative algorithm such as iterative proportional fitting. Finally, we discuss how we can test whether a model gives a reasonable fit, and how one can select a good model when little prior knowledge is available concerning the conditional independence relations between the variables. Most of the material in this chapter is based on the book of Whittaker [Whi90]. Other sources used in writing this chapter are [Edw00, Sch97, Chr97, BFH75, HEL12].

2 Example

Although we assume familiarity with the basic rules of probability, almost all results we use can be inferred from two elementary properties that we list here for reference:

$$P(X) = \sum_Y P(X,Y) \qquad \text{(sum rule)}$$

$$P(X,Y) = P(X|Y)P(Y) \qquad \text{(product rule)}$$

Consider problems where we have a collection of discrete random variables whose joint probability distribution has to be estimated from data. Now suppose we have k random variables each of which can take on m possible values. To estimate the probability of each possible combination would require the estimation of m^k probabilities. For a relatively small problem with 10 variables with 5 possible values each, this is

$$5^{10} = 9,765,625$$

say 10 million probabilities. Typically we have far fewer observations than that, so it is clear we cannot estimate all these probabilities reliably from the limited amount of data we have. This is one of the many manifestations of what is called the *curse of dimensionality*.

How can we simplify such a problem? How can we reduce the number of probabilities we have to estimate in a natural way? One of the most natural ways to do this is to exploit (conditional) independences that might hold in the problem domain. To illustrate, consider a problem with just two ternary variables. There are $3\times3 = 9$ possible value combinations, so without making any simplifying assumptions we have to estimate 8 probabilities (we subtract 1, because we have the constraint that the probabilities must sum to one). Now suppose we observe the counts displayed in Table 1.

To estimate the joint distribution of X and Y, we use

$$\hat{P}(x,y) = \frac{n(x,y)}{n},$$

that is, we just look up how many times a particular combination of values of X and Y occurs in the data and divide this number by the total number

$n(x,y)$	y			
x	1	2	3	$n(x)$
1	2	5	3	10
2	10	20	10	40
3	8	35	7	50
$n(y)$	20	60	20	100

Table 1: Cross-table of counts for two ternary variables

$\hat{P}(x,y)$	y			
x	1	2	3	$\hat{P}(x)$
1	0.02	0.05	0.03	0.10
2	0.10	0.20	0.10	0.40
3	0.08	0.35	0.07	0.50
$\hat{P}(y)$	0.20	0.60	0.20	1

Table 2: Estimated joint distribution for two ternary variables

of observations. Hence, we obtain the estimated joint distribution as given in Table 2.

Now suppose we assume that X and Y are independent random variables. In that case, we have

$$P(X,Y) = P(X)P(Y), \tag{1}$$

that is, the joint distribution can be written as the product of the marginal distributions. Now we only need to estimate the marginal distributions $P(X)$ and $P(Y)$ and plug these estimates into equation (1) to obtain an estimate of the joint probability. This requires the estimation of 2 probabilities (remember the sum to one constraint) for $P(X)$ and the same number for $P(Y)$, hence a total of just 4 probabilities. These estimates can be read from the margins (hence the name marginal probability) of Table 2 and filling them in in equation (1) gives the estimates as displayed in Table 3.

Another way of expressing the result is to compute the “fitted counts” of this model as displayed in Table 4. These are simply obtained by multiplying the estimated probabilities with the total number of observations. To determine whether the independence assumption is justified, we compare the observed counts with the fitted counts of the independence model. We ob-

$\hat{P}(x)\hat{P}(y)$	y			
x	1	2	3	$\hat{P}(x)$
1	0.02	0.06	0.02	0.10
2	0.08	0.24	0.08	0.40
3	0.10	0.30	0.10	0.50
$\hat{P}(y)$	0.20	0.60	0.20	1

Table 3: Estimated joint distribution for two ternary variables using independence assumption

$\hat{n}(x,y)$	y			
x	1	2	3	$n(x)$
1	2	6	2	10
2	8	24	8	40
3	10	30	10	50
$n(y)$	20	60	20	100

Table 4: Fitted counts for two ternary variables using independence assumption

serve that the fitted counts are not that far off, and the independence model seems to give a reasonable fit. To decide in a more justified manner whether the independence assumption should be accepted, a statistical test can be performed. We don't discuss this here, but see section 8.

Next we discuss a somewhat more complicated example. The data set displayed in Table 5 has been made famous by the book of Bishop, Fienberg and Holland [BFH75]. The data gives information on the survival rate of 715 infants attending two clinics and the amount of care received by the mother, where the amount of care is classified as either *more* or *less*. Table 6 gives the probability estimates corresponding to the saturated model, that is, the model making no independence assumptions at all. These estimates are obtained simply by dividing the count in each cell of the table by the total number of observations.

Now consider the model that assumes survival and care are independent within each clinic. This is called a conditional independence assumption because we condition on clinic: we don't state survival and care are inde-

n(clinic, care, survival)		survival	
clinic	care	no	yes
clinic 1	less	3	176
	more	4	293
clinic 2	less	17	197
	more	2	23

Table 5: Three-way tabel relating clinic, care and survival

$\hat{P}$(clinic, care, survival)		survival	
clinic	care	no	yes
clinic 1	less	.004	.246
	more	.006	.410
clinic 2	less	.024	.276
	more	.003	.032

Table 6: Estimated joint distribution of clinic, care and survival without making any independence assumptions (the so-called saturated model)

pendent per se, but that they are independent given clinic. This assumption corresponds to the following factorization (the theory is discussed in the next section):

$$\hat{P}(\text{care, survival} \mid \text{clinic}) = \hat{P}(\text{care} \mid \text{clinic})\hat{P}(\text{survival} \mid \text{clinic})$$

Multiplying left and right by $\hat{P}$(clinic) we get

$$\begin{aligned}\hat{P}(\text{care, survival, clinic}) &= \hat{P}(\text{care,clinic})\hat{P}(\text{survival} \mid \text{clinic}) \\ &= \frac{\hat{P}(\text{care,clinic})\hat{P}(\text{survival,clinic})}{\hat{P}(\text{clinic})}\end{aligned}$$

As you can see from this last expression we have to estimate the joint distribution of care and clinic, and the joint distribution of survival and clinic. The marginal distribution of clinic is obtained by summing out care from the joint distribution of care and clinic, or alternatively, by summing out survival from the joint distribution of survival and clinic. To obtain the necessary counts, we take Table 5 and sum out care respectively survival. The resulting tables of sufficient statistics are displayed in table 7.

n(clinic, care)	care	
clinic	less	more
clinic 1	179	297
clinic 2	214	25

n(clinic, survival)	survival	
clinic	no	yes
clinic 1	7	469
clinic 2	19	220

Table 7: Observed counts required to estimate the conditional independence model. These are called the sufficient statistics.

Writing $\hat{n}$ for $\hat{P}n$ we get

$$\hat{n}(\text{clinic, care, survival}) = \frac{\hat{n}(\text{clinic,care})\hat{n}(\text{clinic,survival})}{\hat{n}(\text{clinic})}.$$

Next, we replace the fitted counts on the right hand side by the corresponding observed counts:

$$\hat{n}(\text{clinic, care, survival}) = \frac{n(\text{clinic,care})n(\text{clinic,survival})}{n(\text{clinic})}.$$

The reason why this is allowed is explained in detail in section 6. Finally, we can compute the fitted values:

$\hat{n}$(clinic, care, survival)		survival	
clinic	care	no	yes
clinic 1	less	2.63	176.37
	more	4.37	292.63
clinic 2	less	17.01	196.99
	more	1.99	23.01

To give one example, the fitted count for clinic=clinic 1, care=more, survival=yes is computed as follows

$$\begin{aligned}\hat{n}(\text{clinic 1, more, yes}) &= \frac{n(\text{clinic 1,more})n(\text{clinic 1, yes})}{n(\text{clinic 1})} \\ &= \frac{297 \times 469}{179 + 297} = 292.63\end{aligned}$$

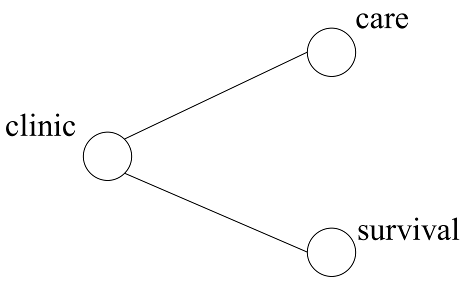


Figure 1: Graphical representation of: "survival is independent of care within each clinic".

	survival		
care	no	yes	(%)
less	20	373	5.1
more	6	316	1.9

Table 8: Cross-table of care and survival. The last column gives the mortality rate.

When we compare the fitted counts to the observed counts we see that they are very close. Hence the assumption that care and survival are independent within each clinic seems to be justified. Again, a rigorous statistical test will confirm this, see section 8. Within the first clinic the mortality rate for the less care group is practically the same as for the more care group; the same is true for the second clinic. In neither clinic is there a relationship between care and survival. In other words, given clinic, care and survival are independent. A graph that describes this structure is given in figure 2

where the vertices correspond to the variables and absence of an edge between care and survival indicates that these variables are conditionally independent given clinic.

The reason this dataset has become well-known is that a strange phenomenon occurs when we sum out clinic, and then analyse the association between care and survival. From table 8 one would conclude that the more maternal care received the lower the infant mortality rate, with the rate dropping by more than half.

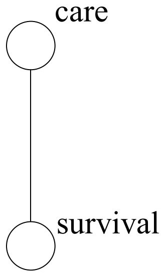


Figure 2: Graphical representation of: "survival is not independent of care".

Apparently, when the three-way table is collapsed over clinic a spurious association between care and survival is induced. The lack of independence suggests the graph given in figure 2.

A lesson to learn is that it is dangerous to analyse a three-way table solely by inspecting its two way margins. Can you explain how the spurious association between care and survival comes about?

3 Independence and Conditional Independence

We give a short review of the concepts of independence and conditional independence of random vectors. We stress that X and Y are vectors (or sets) of random variables.

Random vectors X and Y are independent iff

$$P(x, y) = P(x)P(y) \text{ for all } (x, y),$$

or (same thing) $P(x \,|\, y) = P(x)$, and $P(y \,|\, x) = P(y)$. The second formulation best captures the intuitive idea: if X and Y are independent, then learning the value of X doesn't give you any information about Y, and vice versa. We also write $X \perp\!\!\!\perp Y$. Example: gender is independent of eye color.

To establish independence, it is sufficient to show that the joint density function factorises; it is not necessary to show explicitly that it factorises into the product of the marginal distributions. This gives us the factorisation criterion for independent random vectors: random vectors X and Y are independent if and only if there exist two functions g and h such that

$$P(x, y) = g(x)h(y) \text{ for all } (x, y)$$

We will often make use of the "log-version" of this criterion:

$$\log P(x, y) = g'(x) + h'(y) \text{ for all } (x, y),$$

where $g'(x) = \log g(x)$, and $h'(y) = \log h(y)$.

Next we discuss what is in a sense the central concept of graphical models: conditional independence. Random vectors X and Y are conditionally independent given Z iff

$$P(x, y \mid z) = P(x \mid z)P(y \mid z) \tag{2}$$

for all (x, y) and for all z for which $P(z) > 0$. Again, the alternative formulation $P(x \,|\, y, z) = P(x \,|\, z)$ best captures the intuitive idea: if I already know the value of Z, then learning the value of Y doesn't give me any additional information about X. We also write $X \perp\!\!\!\perp Y \mid Z$.

For example, consider the variables "heavy smoking", "lung cancer", and "yellow fingers". Heavy smoking increases the probability of lung cancer, and also causes nicotine stain on the fingers ("yellow fingers"). Hence lung cancer and yellow fingers will be positively associated with each other, but only because they have a common cause (heavy smoking). So lunger cancer and yellow fingers are not independent of each other:

$$\text{lung cancer} \not\perp\!\!\!\perp \text{ yellow fingers}$$

However, if I know that someone is a heavy smoker, then the association between lung cancer and yellow fingers will disappear: once I know whether or not someone is a heavy smoker, the information that someone has yellow fingers does not change my assessment of the probability that this person develops lung cancer. This whole reasoning is based on the assumption that there is no other causal mechanism linking yellow fingers and lung cancer of course. Even though the example is not perfect, I hope it conveys the intuition of what conditional independence means.

An equivalent formulation is (multiply equation 2 left and right by $P(z)$):

$$P(x, y, z) = \frac{P(x, z)P(y, z)}{P(z)}$$

which shows that conditional independence can be rephrased entirely in terms of marginal densities.

Like with marginal independence we can state a simple factorisation criterion to establish conditional independence: random vectors X and Y are conditionally independent given 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 $P(z) > 0$. Again we will often use the "log-version"

$$\log P(x, y, z) = g'(x, z) + h'(y, z)$$

4 Independence Graphs

We can represent the conditional independence relations between a set of random variables in a so-called conditional independence graph. Let $X = (X_1, X_2, \ldots, X_k)$ be a k-dimensional random vector. The conditional independence graph of X is the undirected graph $G = (K, E)$, with $K = \{1, 2, \ldots, k\}$, and where $\{i, j\}$ is *not* in the edge set E iff

$$X_i \perp\!\!\!\perp X_j \mid \text{rest}$$

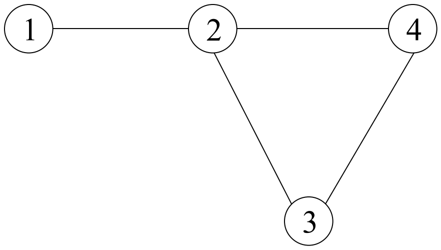
Example: Let $X = (X_1, X_2, X_3, X_4), 0 < x_i < 1$ with probability density

$$P(x) = \exp(u + x_1 + x_1x_2 + x_2x_3x_4)$$

Application of the factorisation criterion gives

$$X_1 \perp\!\!\!\perp X_4 \mid (X_2, X_3) \text{ and } X_1 \perp\!\!\!\perp X_3 \mid (X_2, X_4)$$

Hence the conditional independence graph of X is



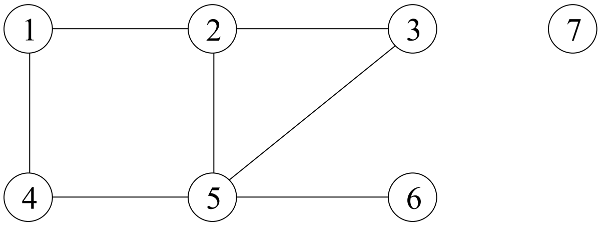


Figure 3: Example of a conditional independence graph

In fact we can reduce the conditioning set by using the concept of separation.

From the conditional independence graph in figure 3 we can read that

$$X_1 \perp\!\!\!\perp X_3 | (X_2, X_4, X_5, X_6, X_7)$$

However, since $\{2, 5\}$ separates 1 from 3 in the graph (i.e. every path from 1 to 3 must go through 2 or 5), we can make the stronger statement

$$X_1 \perp\!\!\!\perp X_3 | (X_2, X_5)$$

We defined the conditional independence graph using the rule that for all non-adjacent vertices i and j

$$X_i \perp\!\!\!\perp X_j | \text{rest}$$

This is called the pairwise Markov property. Perhaps surprisingly, the following properties turn out to be equivalent

Global Markov property: a separates b from c (a, b, c disjoint) iff

$$X_b \perp\!\!\!\perp X_c | X_a$$

where $X_a = (X_i; i \in a)$ and a separates b from c if for all $i \in b, j \in c$: a separates i from j.

Local Markov property:

$$X_i \perp\!\!\!\perp \text{ rest } | \text{ boundary}(i)$$

where the boundary of a vertex i is simply the set of adjacent vertices.

The local Markov property is particularly relevant to prediction. For example, to predict X_2 in figure 3, we only need to know the values of X_1, X_3 and X_5.

When we say that the pairwise, local, and global Markov properties are equivalent, what we mean to say is that if all pairwise independencies corresponding to graph G hold for a given probability distribution, then all the global independencies corresponding to G also hold for that distribution (and vice versa).

5 Log-linear Models

In this section we introduce the class of log-linear models and its subclasses of hierarchical, graphical log-linear, and finally decomposable models. For ease of exposition we start with log-linear models for binary variables.

5.1 Log-linear models for binary data

A random experiment that only distinguishes between two possible outcomes is called a *Bernoulli* experiment. The outcomes are usually referred to as *success* and *failure* respectively. We define a random variable X that denotes the number of successes in a Bernoulli experiment; X therefore has possible values 0 and 1. The probability distribution of X is completely determined by the probability of success, which we denote by p, and is: $P(X = 0) = 1-p$ and $P(X = 1) = p$.

A Bernoulli random variable X, has the probability function

$$P(x) = p^x(1-p)^{1-x} \quad \text{for } x = 0, 1 \text{ and } 0 \leq p \leq 1$$

This is a compact way of writing the probability of both outcomes in a single formula; check that indeed $P(1) = p$ and $P(0) = 1 - p$ as required.

Next we consider the analysis of a 2×2 table. The bivariate Bernoulli random vector (X_1, X_2), takes the values $(0, 0)$, $(0, 1)$,$(1, 0)$ and $(1, 1)$ in the Cartesian product $\{0, 1\} \times \{0, 1\}$. Its distribution is completely specified by the table of probabilities

$P(x_1, x_2)$		$x_2 = 0$	$x_2 = 1$	Total
	$x_1 = 0$	$p(0, 0)$	$p(0, 1)$	$p_1(0)$
	$x_1 = 1$	$p(1, 0)$	$p(1, 1)$	$p_1(1)$
	Total	$p_2(0)$	$p_2(1)$	1

Like before, we can write the probability distribution as a single function:

$$P(x_1, x_2) = p(0,0)^{(1-x_1)(1-x_2)} p(0,1)^{(1-x_1)x_2} p(1,0)^{x_1(1-x_2)} p(1,1)^{x_1 x_2}$$

for $x_1 = 0, 1$ and $x_2 = 0, 1$. Verify that $P(0,0) = p(0,0)$, $P(0,1) = p(0,1)$, $P(1,0) = p(1,0)$ and $P(1,1) = p(1,1)$ as required.

Taking logarithms of this identity for P, and collecting terms in x_1 and x_2 gives

$$\begin{aligned}\log P(x_1, x_2) &= \log p(0,0) + x_1 \log \frac{p(1,0)}{p(0,0)} + \\ &\qquad x_2 \log \frac{p(0,1)}{p(0,0)} + x_1 x_2 \log \frac{p(1,1)p(0,0)}{p(0,1)p(1,0)}\end{aligned}$$

Exercise: Verify this result using the following properties of logarithms:

$$\begin{aligned}\log a^b &= b \log a \\ \log ab &= \log a + \log b \\ \log \frac{a}{b} &= \log a - \log b\end{aligned}$$

Reparameterizing the right hand side leads to the so-called *log-linear expansion*

$$\log P(x_1, x_2) = u_\emptyset + x_1 u_1 + x_2 u_2 + x_1 x_2 u_{12} \quad \text{for } (x_1, x_2) \text{ in } \{0,1\}^2$$

The coefficients, $u_\emptyset, u_1, u_2, u_{12}$ are known as the u-terms. For example

$$u_1 = \log \frac{p(1,0)}{p(0,0)}$$

which is the log of the odds of the event $X_1 = 1$ to the event $X_1 = 0$ conditioned on $X_2 = 0$. The coefficient of the product $x_1 x_2$ is the logarithm of the cross product ratio

$$u_{12} = \log \frac{p(1,1)p(0,0)}{p(0,1)p(1,0)} = \log \mathrm{cpr}(X_1, X_2)$$

This makes sense, because the cross-product ratio is a well-known measure for the degree of association between two binary variables. If $\mathrm{cpr}(X_1, X_2) > 1$, then X_1 and X_2 are positively associated (e.g. smoking and lung cancer), if $\text{cpr}(X_1, X_2) < 1$ then X_1 and X_2 are negatively associated and if $\text{cpr}(X_1, X_2) = 1$ then X_1 and X_2 are not associated, i.e. independent.

This last property can be verified by applying the factorisation criterion to the log-linear expansion. The factorisation criterion states that X_1 and X_2 are independent if and only if there exist two functions g and h such that

$$\log P(x_1, x_2) = g(x_1) + h(x_2) \text{ for all } (x_1, x_2)$$

If $u_{12} = 0$, the log-linear expansion simplifies to

$$\log P(x_1, x_2) = u_\emptyset + x_1 u_1 + x_2 u_2$$

Hence, we can take $g(x_1) = u_\emptyset + x_1 u_1$ and $h(x_2) = u_2 x_2$. If $u_{12} \neq 0$ no such decomposition is possible.

Since $u_{12} = \log \text{cpr}(X_1, X_2)$, this implies that X_1 and X_2 are independent if and only if $\text{cpr}(X_1, X_2) = 1$.

The log-linear expansion of a $2 \times 2 \times 2$ table (three dimensional Bernoulli) is obtained in a similar way. The density function can be written

$$P(x_1, x_2, x_3) = p(0,0,0)^{(1-x_1)(1-x_2)(1-x_3)} \cdots p(1,1,1)^{x_1 x_2 x_3}$$

The log-linear expansion is

$$\begin{aligned}\log P(x_1, x_2, x_3) &= u_\emptyset + u_1 x_1 + u_2 x_2 + u_3 x_3 + u_{12} x_1 x_2 + \\ &\quad u_{13} x_1 x_3 + u_{23} x_2 x_3 + u_{123} x_1 x_2 x_3\end{aligned}$$

Note that for example

$$X_2 \perp\!\!\!\perp X_3 \mid X_1 \Leftrightarrow u_{23} = 0 \text{ and } u_{123} = 0$$

In general, we can enforce (conditional) independence constraints, by setting the right u-terms to zero.

5.2 Extension to non-binary data

So far we assumed all variables are binary. In general we allow discrete variables with more than two levels as well. To see how we can generalise the log-linear model to this case, consider again the 2×2 table

$$\log P(x_1, x_2) = u_\emptyset + u_1 x_1 + u_2 x_2 + u_{12} x_1 x_2$$

for $x \in \{0, 1\}^2$. What if the x_i have more than two levels? The solution is to make the u-terms *functions* of x rather than constants:

$$\log P(x_1, x_2) = u_\emptyset + u_1(x_1) + u_2(x_2) + u_{12}(x_1, x_2) \tag{3}$$

For example, we have a different u term $u_{12}(x_1, x_2)$ for each possible value combination of x_1 and x_2. However, in order not to create redundant parameters, we impose the constraint that $u_a(x_a) = 0$ whenever $x_i = 0$ and $i \in a$. Here we assume that if x_i has d_i possible values, these are numbered $0, 1, \ldots, d_i - 1$. Note however that this numbering does not imply any ordering of the values. With this constraint, the complete log-linear expansion has as many u-terms as there are cells in the probability table. It stands to reason that having more u-terms than there are combinations of values of the variables creates superfluous parameters.

So for example, suppose x_1 has two possible values (0,1) and x_2 has three possible values (0,1,2) then the following u-terms are constrained to be zero

$$u_1(0) = 0, u_2(0) = 0, u_{12}(0, 0) = u_{12}(0, 1) = u_{12}(0, 2) = u_{12}(1, 0) = 0$$

Also note that the constraint is consistent with the binary case when the u-terms were constants. For example, in the binary case the u-term u_{12} would drop from the model when either x_1 or x_2 (or both) were zero, because u_{12} was multiplied by $x_1 x_2$.

5.3 Hierarchical and Graphical Log-linear models

Definition 1 (Log-linear expansion) *The log-linear expansion of the cross-classified Multinomial distribution P_K is*

$$\log P_K(x) = \sum_{a \subseteq K} u_a(x_a)$$

where the sum is taken over all possible subsets a of $K = \{1, 2, \ldots, k\}$ and where the u-terms satisfy the constraint that $u_a(x_a) = 0$ whenever $x_i = 0$ and $i \in a$.

Next we state the conditions for independence as conditions on the u-terms.

Proposition 1 (Independence and the u-terms) *If (X_a, X_b, X_c) is a partitioned Multinomial random vector then $X_b \perp\!\!\!\perp X_c \mid X_a$ if and only if all u-terms in the log-linear expansion with coordinates from both b and c, are zero.*

The proof is a direct application of the factorisation theorem for conditional independence. Let t be an arbitrary subset of $a \cup b \cup c = \{1, 2, \ldots, k\}$. If all u-terms, u_t, are zero whenever $t \not\subseteq a \cup b$ and $t \not\subseteq a \cup c$ (i.e. whenever t contains coordinates from both b and c) then we can write

$$\log P_K = \sum_{t \subseteq a \cup b} u_t + \sum_{t \subseteq a \cup c} u_t - \sum_{t \subseteq a} u_t$$

This function is of the form $g(x_a, x_b) + h(x_a, x_c)$ and hence $X_b \perp\!\!\!\perp X_c \mid X_a$ by the factorisation criterion. Note that we had to subtract $\sum_{t \subseteq a} u_t$ because these u-terms were counted twice by the first two sums of the equation.

The importance of the log-linear expansion rests in the fact that many interesting hypotheses can be generated by setting u-terms to zero. Proposition 1 gives conditions on the u-terms for conditional independence.

In most applications it does not make sense to include the three way association u_{123} unless the two-way associations u_{12}, u_{13} and u_{23} are also present. A log-linear model is said to be hierarchical if the presence of a term implies that all lower-order terms that are contained in it are also present. This implies that a hierarchical model is identified by listing its highest order interaction terms. In table 9 we give all hierarchical models for three dimensions.

Definition 2 (Graphical Model) *Given an independence graph $G = (K, E)$, the cross-classified Multinomial distribution for the random vector X is a* graphical model *for X if the distribution of X is arbitrary apart from constraints of the form that for all pairs of coordinates not in the edge set E of G, the u-terms containing the selected coordinates are equal to zero.*

More explicitly, the density of a Multinomial graphical model is

$$\log P_K(x) = \sum_{a \subseteq K} u_a(x_a)$$

subject to the constraints that $u_a = 0$ if $\{i, j\} \subseteq a$ and (i, j) is not in the edge set E. The parameters of the graphical model are the remaining u-terms that

Model	Omitted	Interpretation
123	none	saturated
12,13,23	u_{123}	homogeneous association
12,13	u_{123}, u_{23}	$X_2 \perp\!\!\!\perp X_3 \mid X_1$
12,23	u_{123}, u_{13}	$X_1 \perp\!\!\!\perp X_3 \mid X_2$
13,23	u_{123}, u_{12}	$X_1 \perp\!\!\!\perp X_2 \mid X_3$
12,3	u_{123}, u_{13}, u_{23}	$(X_1, X_2) \perp\!\!\!\perp X_3$
13,2	u_{123}, u_{12}, u_{23}	$(X_1, X_3) \perp\!\!\!\perp X_2$
23,1	u_{123}, u_{12}, u_{13}	$(X_2, X_3) \perp\!\!\!\perp X_1$
1,2,3	$u_{123}, u_{12}, u_{13}, u_{23}$	mutual independence

Table 9: All hierarchical models with 3 variables

are not set to zero. More informally, one could say that a model is graphical if it is *completely* characterized by its independence graph; all constraints can be read from the independence graph.

In figure 4 we show four hierarchical models and their independence graphs. Note that the saturated model and the homogeneous association model have the same independence graph. The homogeneous association model is *not* a graphical model however, because it imposes the additional constraint that $u_{123} = 0$, and this constraint can not be inferred from the independence graph. In fact the homogeneous association model is the only hierarchical model in 3 dimensions that is not graphical.

6 Maximum Likelihood Estimation of Hierarchical and Graphical Models

To estimate the parameters of log-linear models, we will use the principle of maximum likelihood. The principle of maximum likelihood states that to estimate an unknown parameter θ, we should pick the value $\hat{\theta}$ that maximizes the probability of the data we actually observed. Let's be a bit more specific. If we have observations x that are assumed to be drawn from a probability distribution $P(x;\theta)$, where θ represents the vector of unknown parameters of P, then:

$$\hat{\theta} = \arg\max_{\theta} \prod_{i=1}^{n} P(x_i;\theta)$$

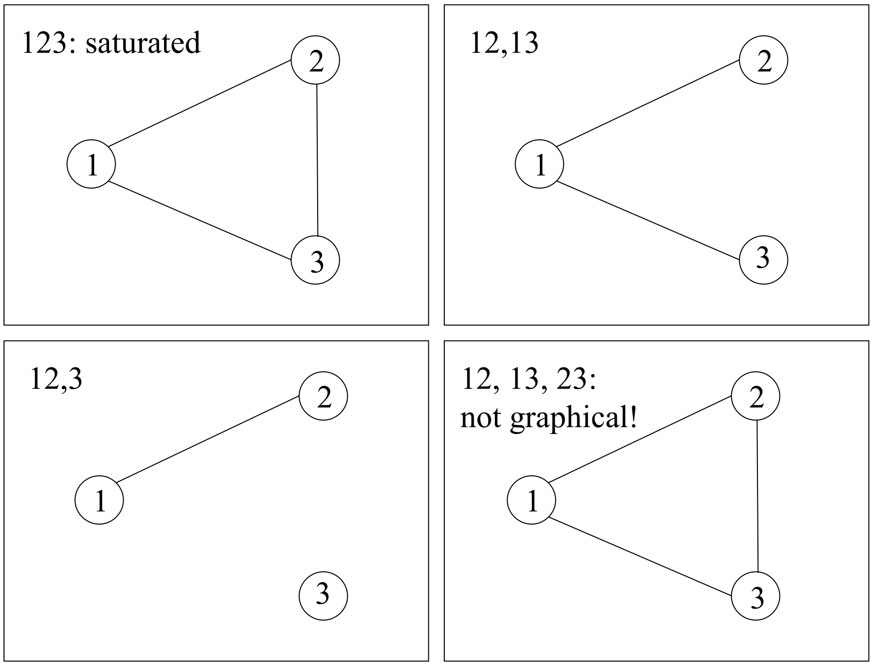


Figure 4: Four hierarchical models and their independence graphs. The hierarchical models are specified by listing their highest-order interaction terms.

In this formula, we assumed that we have n independent observations of x. Hence we were allowed to multiply their individual probabilities to obtain the joint probability of all observations together. The assumption that the observations are independent is not part of the principle of maximum likelihood however. In practice, one often use the log-likelihood instead of the likelihood:

$$\hat{\theta} = \arg\max_{\theta} \sum_{i=1}^{n} \log P(x_i; \theta)$$

We will start with a small concrete example. Suppose we have a problem with two binary variables $X_1, X_2 \in \{0,1\} \times \{0,1\}$. We estimate the independence model $X_1 \perp\!\!\!\perp X_2$, which has the log-linear representation

$$\log P(x_1, x_2) = u_\emptyset + u_1 x_1 + u_2 x_2$$

Here $u_\emptyset$, u_1, and u_2 are the unknown parameters that we collectively called θ in the general formulation of the maximum likelihood principle. Recall that in this independence model we have the following relation between u terms and cell probabilities:

$$\begin{aligned}
\log p(0,0) &= u_\emptyset \Rightarrow p(0,0) = e^{u_\emptyset} \\
\log p(1,0) &= u_\emptyset + u_1 \Rightarrow p(1,0) = e^{u_\emptyset + u_1} \\
\log p(0,1) &= u_\emptyset + u_2 \Rightarrow p(0,1) = e^{u_\emptyset + u_2} \\
\log p(1,1) &= u_\emptyset + u_1 + u_2 \Rightarrow p(1,1) = e^{u_\emptyset + u_1 + u_2}
\end{aligned}$$

Suppose we observe the following data:

$n(x_1, x_2)$	$x_2 = 0$	$x_2 = 1$	Total
$x_1 = 0$	10	20	30
$x_1 = 1$	30	40	70
Total	40	60	100

This yields the log-likelihood function

$$\ell(u; n(x)) = 100u_0 + 70u_1 + 60u_2$$

The standard approach to find the values of u_0, u_1, u_2 that maximize the log-likelihood is to take the derivatives of the log-likelihood function with respect to these parameters, equate them to zero, and solve for the u terms. We have to take into account however the constraint that the cell probabilities must sum to one:

$$\sum_x p(x) = 1 \Rightarrow e^{u_\emptyset} + e^{u_\emptyset + u_1} + e^{u_\emptyset + u_2} + e^{u_\emptyset + u_1 + u_2} = 1$$

Using the method of Lagrange to take into account this equality constraint, the new objective function becomes:

$$\ell(u; n(x); \lambda) = 100u_0 + 70u_1 + 60u_2 - \lambda \left(\sum_x p(x) - 1 \right)$$

Next we take derivatives with respect to the Lagrange multiplier λ, and the

observed	$x_2 = 0$	$x_2 = 1$	Total
$x_1 = 0$	0.1	0.2	0.3
$x_1 = 1$	0.3	0.4	0.7
Total	0.4	0.6	1.0

fitted	$x_2 = 0$	$x_2 = 1$	Total
$x_1 = 0$			0.3
$x_1 = 1$			0.7
Total	0.4	0.6	1.0

Table 10: Observed relative frequencies (left) and margins fitted by the independence model (right).

u terms, and equate these derivatives to zero:

$$\begin{aligned}
\frac{\partial \ell}{\partial \lambda} &: 1 - \sum_x p(x) = 0 \\
\frac{\partial \ell}{\partial u_0} &: 100 - \lambda \sum_x p(x) = 0 \\
\frac{\partial \ell}{\partial u_1} &: 70 - \lambda \sum_x x_1 p(x) = 0 \\
\frac{\partial \ell}{\partial u_2} &: 60 - \lambda \sum_x x_2 p(x) = 0
\end{aligned}$$

From the first two equations, it follows that $\lambda = 100$. Filling in $\lambda = 100$ in the third equation gives:

$$70 - 100 p_1(1) = 0 \Rightarrow \hat{p}_1(1) = \frac{70}{100} = 0.7$$

Likewise, filling in $\lambda = 100$ in the fourth equation gives:

$$60 - 100 p_2(1) = 0 \Rightarrow \hat{p}_2(1) = \frac{60}{100} = 0.6$$

It follows that $\hat{p}_1(0) = 0.3$ and $\hat{p}_2(0) = 0.4$. Note that the fitted margins are equal to the observed margins (see table 10).

To complete the table of fitted probabilities, we use the model's independence assumption:

$$\hat{p}_{12}(1,1) = \hat{p}_1(1) \times \hat{p}_2(1) = 0.7 \times 0.6 = 0.42$$

fitted	$x_2 = 0$	$x_2 = 1$	Total
$x_1 = 0$	0.12	0.18	0.3
$x_1 = 1$	0.28	0.42	0.7
Total	0.4	0.6	1.0

Table 11: Maximum likelihood fitted probabilities of the independence model.

This gives the fitted probabilities as given in table 11.

It is straightforward to obtain the fitted u terms from the fitted probabilities:

$$\begin{aligned}
\hat{u}_0 &= \log \hat{p}(0,0) = \log 0.12 = -2.12 \\
\hat{u}_1 &= \log \hat{p}(1,0) - \hat{u}_0 = -1.27 + 2.12 = 0.85 \\
\hat{u}_2 &= \log \hat{p}(0,1) - \hat{u}_0 = -1.71 + 2.12 = 0.41 \\
\hat{u}_{12} &= 0 && \text{(Assumption)}
\end{aligned}$$

Hence the fitted model expressed in u terms is:

$$\log \hat{P}(x_1, x_2) = -2.12 + 0.85x_1 + 0.41x_2.$$

Luckily, we don't have to solve every estimation problem from first principles, because there is a general result we can use to simplify the problem. This general result is that the maximum likelihood estimator of graphical log-linear model M satisfies the likelihood equations

$$\hat{n}_a = N\hat{P}_a = n_a$$

whenever the subset of vertices a in the graph form a clique. This is summarized by the slogan: "Observed = Fitted" for every marginal table corresponding to a complete subgraph. We can see as follows why this has to be the case:

1. If there are no constraints to fit an observed table of counts, then the parameter estimates that yield fitted counts equal to the observed counts maximize the likelihood function. For example, the saturated model will yield fitted counts identical to the observed counts.

2. By definition, a graphical model is arbitrary (has no constraints) except for the constraints that can be read from the independence graph.

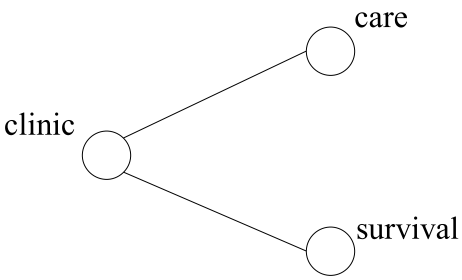
3. Suppose a forms a clique in the independence graph. Now consider the partitioning $X = (X_a, X_b)$ where b contains all variables not in a. We can write
$$P(X) = P(X_a)P(X_b \mid X_a)$$
Since $P(X_a)$ is not constrained by the model (complete graph), all model constraints apply only to $P(X_b \mid X_a)$. Therefore, the maximum likelihood estimates will yield $\hat{n}_a = n_a$.

Likewise, the maximum likelihood estimator of hierarchical log-linear model M satisfies the likelihood equations

$$\hat{n}_a = N\hat{P}_a = n_a$$

whenever a belongs to the highest order interaction terms of M.

As an example, we return to the infant survival data. We saw that the model



seemed to give a pretty good representation of the data at first sight. Let's fit this model to the data

n_{123}		survival	
clinic	care	no	yes
clinic 1	less	3	176
	more	4	293
clinic 2	less	17	197
	more	2	23

We number the variables as follows: 1=clinic, 2=care, 3=survival. Then the cliques in the graph are 12 and 13, and so the sufficient statistics are n_{12}

and n_{13}. Hence the maximum likelihood estimates satisfy the equations

$$\hat{n}_{12}(x_1, x_2) = N\hat{P}_{12}(x_1, x_2) = n_{12}(x_1, x_2) \tag{4}$$

$$\hat{n}_{13}(x_1, x_3) = N\hat{P}_{13}(x_1, x_3) = n_{13}(x_1, x_3) \tag{5}$$

The tables with the sufficient statistics are given below

n_{12}	care	
clinic	less	more
clinic 1	179	297
clinic 2	214	25

n_{13}	survival	
clinic	no	yes
clinic 1	7	469
clinic 2	19	220

Next we apply the rules of probability, and the independence constraint expressed by the model to find an expression for the maximum likelihood estimates.

$$\begin{aligned}
\hat{P}(x_1, x_2, x_3) &= \hat{P}(x_2, x_3 \mid x_1)\hat{P}(x_1) && \text{(product rule)} \\
&= \hat{P}(x_2|x_1)\hat{P}(x_3|x_1)\hat{P}(x_1) && (X_2 \perp\!\!\!\perp X_3 \mid X_1) \\
&= \hat{P}(x_2|x_1)\hat{P}(x_1, x_3) && \text{(product rule)} \\
&= \frac{\hat{P}(x_1, x_2)\hat{P}(x_1, x_3)}{\hat{P}(x_1)} && \text{(product rule)}
\end{aligned}$$

In terms of counts (writing $\hat{n}$ for $N\hat{P}$) we obtain:

$$\begin{aligned}
\hat{n}(x_1, x_2, x_3) &= \frac{\hat{n}(x_1, x_2)\hat{n}(x_1, x_3)}{\hat{n}(x_1)} \\
&= \frac{n(x_1, x_2)n(x_1, x_3)}{n(x_1)} \qquad (6)
\end{aligned}$$

In the last step we replace the fitted counts on the right hand side by the corresponding observed counts, making use of the fact that the maximum likelihood estimates satisfy the margin constraints given in equations (4) and (5). This gives the fitted values:

$\hat{n}_{123}$		survival	
clinic	care	no	yes
clinic 1	less	2.63	176.37
	more	4.37	292.63
clinic 2	less	17.01	196.99
	more	1.99	23.01

The model seems to fit very well indeed.

6.1 Iterative Proportional Fitting

Not all (hierarchical) log-linear models have closed form maximum likelihood estimates as in the previous example. There is however a simple iterative algorithm called Iterative Proportional Fitting (IPF) that will converge to those estimates. We start by giving a simple example that actually does not require IPF. Suppose we want to fit the independence model to

$n(x_1, x_2)$	$x_2 = 0$	$x_2 = 1$	$n_1(x_1)$
$x_1 = 0$	30	10	40
$x_1 = 1$	30	30	60
$n_2(x_2)$	60	40	100

The minimal sufficient statistics are row totals $n_1(x_1)$ and column totals $n_2(x_2)$. In other words, the ML estimates satisfy the equations

$$\begin{aligned} \hat{n}_1(x_1) &= n_1(x_1) \\ \hat{n}_2(x_2) &= n_2(x_2) \end{aligned}$$

This gives the closed form estimates

$$\hat{n}_{12}(x) = n_1(x_1)n_2(x_2)/N$$

Application of this formula gives the following table of fitted values

$\hat{n}(x_1, x_2)$	$x_2 = 0$	$x_2 = 1$	$\hat{n}_1(x_1)$
$x_1 = 0$	24	16	40
$x_1 = 1$	36	24	60
$\hat{n}_2(x_2)$	60	40	100

We will now show how we arrive at this solution using IPF. We usually begin with a table $\hat{n}^{(0)}$ of uniform counts

1	1	2
1	1	2

In the first step we fit to the observed row margin:

$$\hat{n}(x)^{(1)} = \hat{n}(x)^{(0)} \times \frac{n_1(x_1)}{\hat{n}_1(x_1)^{(0)}}$$

We compute

$$\hat{n}(0,0)^{(1)} = 1 \times \frac{40}{2} = 20 \qquad \hat{n}(0,1)^{(1)} = 1 \times \frac{40}{2} = 20$$

and

$$\hat{n}(1,0)^{(1)} = 1 \times \frac{60}{2} = 30 \qquad \hat{n}(1,1)^{(1)} = 1 \times \frac{60}{2} = 30$$

which yields $\hat{n}^{(1)}$:

20	20	40
30	30	60

In the second step we fit to the observed column margin:

$$\hat{n}(x)^{(2)} = \hat{n}(x)^{(1)} \times \frac{n_2(x_2)}{\hat{n}_2(x_2)^{(1)}}$$

Which gives

$$\hat{n}(0,0)^{(2)} = 20 \times \frac{60}{50} = 24 \qquad \hat{n}(0,1)^{(2)} = 20 \times \frac{40}{50} = 16$$

and

$$\hat{n}(1,0)^{(2)} = 30 \times \frac{60}{50} = 36 \qquad \hat{n}(1,1)^{(2)} = 30 \times \frac{40}{50} = 24$$

This yields $\hat{n}^{(2)}$:

24	16
36	24
60	40

Notice that the row totals are still 40 and 60, so we have simultaneously satisfied the conditions

$$\hat{n}_1(x_1) = n_1(x_1) \text{ and } \hat{n}_2(x_2) = n_2(x_2)$$

so we have converged. IPF has the nice property that if there is an explicit formula for the ML estimates, then the algorithm will reach these values within one iteration, i.e. each margin has to be fit only once (provided the margins are processed in the correct order, see section 7. In case there is no closed-form solution more iterations are required. Why did we start the procedure from a uniform table of counts? The point is we have to start with a table that satisfies all constraints imposed by the log-linear model. In our example, we were fitting the independence model

$$\log P(x_1, x_2) = u_0 + u_1 x_1 + u_2 x_2$$

The uniform table of counts satisfies this model with $u_1 = 0$, $u_2 = 0$ and $u_0 = \log 1/4$. In fact the uniform table sets all u terms to zero except for u_0 which has the value $\log 1/N$. So as long as the model does not set u_0 to zero (and no acceptable model does), the uniform table satisfies the model constraints. Now if the log-linear model constrains a particular u-term to be zero, then the steps of the IPF algorithm will not violate this constraint. For example, in the independence model we set

$$u_{12} = \log \mathrm{cpr}(X_1, X_2) = 0$$

In other words, $\mathrm{cpr}(X_1, X_2) = 1$. Now the uniform table obviously satisfies this constraint (recall the definition of the cross-product ratio). A proportional adjustment of a row or column does not change the cross-product ratio since

$$\frac{\hat{n}(0,0)\hat{n}(1,1)}{\hat{n}(0,1)\hat{n}(1,0)} = \frac{c\,\hat{n}(0,0)\hat{n}(1,1)}{c\,\hat{n}(0,1)\hat{n}(1,0)}$$

for any value of $c \neq 0$. Hence we had to start with a table with cpr = 1, to get a solution for which this is also the case.

We now consider a slightly more complicated example in 3 dimensions. The only hierarchical model with 3 variables that does not have a closed from solution is the so called homogeneous association model with highest order interaction terms: 12,13,23. IPF proportionally adjusts the estimated expected frequencies $\hat{n}_{123}(x)$ to in turn satisfy the constraints

(1) $\hat{n}_{12}(x_1, x_2) = n_{12}(x_1, x_2)$

(2) $\hat{n}_{13}(x_1, x_3) = n_{13}(x_1, x_3)$

(3) $\hat{n}_{23}(x_2, x_3) = n_{23}(x_2, x_3)$

One iteration of IPF for this model looks like this.
Fit to 12 margin:

$$\hat{n}_{123}(x)^{(t+1)} = \hat{n}_{123}(x)^{(t)} \left(\frac{n_{12}(x_1, x_2)}{\hat{n}_{12}(x_1, x_2)^{(t)}} \right)$$

Fit to 13 margin:

$$\hat{n}_{123}(x)^{(t+2)} = \hat{n}_{123}(x)^{(t+1)} \left(\frac{n_{13}(x_1, x_3)}{\hat{n}_{13}(x_1, x_3)^{(t+1)}} \right)$$

Fit to 23 margin:

$$\hat{n}_{123}(x)^{(t+3)} = \hat{n}_{123}(x)^{(t+2)} \left(\frac{n_{23}(x_2, x_3)}{\hat{n}_{23}(x_2, x_3)^{(t+2)}} \right)$$

In the first step we make sure the fitted 12 margin is equal to the observed 12 margin. In the second step we do the same for the 13 margin. This may disrupt the result op the previous step. In the third step we fit to the 13 margin. These three steps are repeated until all three fitted margins are equal to the observed margins simultaneously.

Finally we give a sketch of the general IPF algorithm. Say we have m margins $\{a_1, a_2, \ldots, a_m\}$ to be fitted ($\cup_i a_i = K$). We have to find a table $\hat{n}(x)$ that agrees with the observed table $n(x)$ on the m margins corresponding to the subsets a_i.

The algorithm cycles through the list of subsets

$$a = a_i, \qquad\qquad i = 1, 2, \ldots, m$$

fitting $\hat{n}(x)$ to each margin in turn. For each margin a we apply the IPF updating rule

$$\hat{n}_{ab}(x_a, x_b)^{(t+1)} = n_a(x_a) \left(\frac{\hat{n}_{ab}(x_a, x_b)^{(t)}}{\hat{n}_a(x_a)^{(t)}} \right)$$

where b is the complement of a. We keep cycling through the margins until convergence is reached. It can be shown that after fitting to margin a, we indeed have

$$\hat{n}_a(x_a)^{(t+1)} = n_a(x_a).$$

Proof:

$$\begin{aligned}
\hat{n}_a(x_a)^{(t+1)} &= \sum_{x_b} \hat{n}_{ab}(x_a, x_b)^{(t+1)} \\
&= \sum_{x_b} \left(\frac{\hat{n}_{ab}(x_a, x_b)^{(t)}}{\hat{n}_a(x_a)^{(t)}} \right) n_a(x_a) \\
&= \sum_{x_b} \left(\frac{\hat{n}_{ab}(x_a, x_b)^{(t)}}{\sum_{x_b} \hat{n}_{ab}(x_a, x_b)^{(t)}} \right) n_a(x_a) \\
&= n_a(x_a)
\end{aligned}$$

We finish this section with pseudo-code for the IPF algorithm:

Algorithm 1 IPF($n(x)$, $\mathcal{A}$)

```
t ← 0
for all values x of X do
    $\hat{n}(x)^{(t)} \leftarrow 1$
end for
repeat
    for all margins $a \in \mathcal{A}$ do
        for all values $x_a$ of $X_a$ do
            for all values $x_b$ of $X_b$ do
                $\hat{n}_{ab}(x_a, x_b)^{(t+1)} \leftarrow n_a(x_a) \left( \frac{\hat{n}_{ab}(x_a,x_b)^{(t)}}{\hat{n}_a(x_a)^{(t)}} \right)$
            end for
        end for
        $t \leftarrow t + 1$
    end for
until convergence
```

7 Decomposable Graphical Models

Decomposable models are graphical models that have explicit formulas for the maximum likelihood estimates. This is a convenient property from a computational viewpoint. If we only have to fit one model this is perhaps not so important, but when we have little prior knowledge we typically have to search a potentially large space of possible models.

Decomposable models are easy to characterize by their independence graphs. They have *triangulated* independence graphs: their independence graphs have no *chordless cycles* of length greater than three. A cycle is called chordless if no other than successive pairs of vertices in the cycle are adjacent in the graph. More informally, a cycle is chordless if it doesn't have a “shortcut”. The graph in the left of figure 7 is *not* decomposable because it contains the chordless 4-cycle 1-2-3-4-1. The graph in the right of figure 7 *is* decomposable because now the cycle 1-2-3-4-1 has the shortcut 3-1.

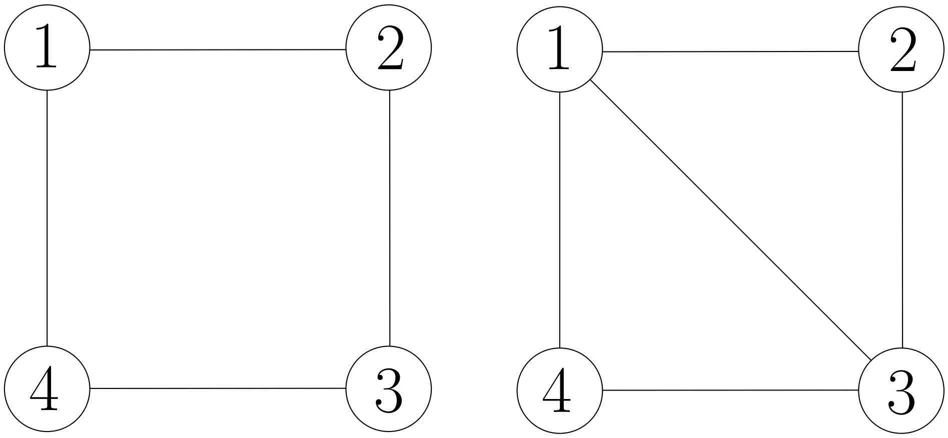


Figure 5: Graph with chordless 4-cycle (left), and its triangulated version (right).

For decomposable models, the formula for the maximum likelihood estimates can be determined as follows. First we have to find an ordering of the cliques that has the so-called running intersection property (RIP). An ordering $C_1, \ldots, C_m$ of the cliques of the graph has the running intersection property if and only if:

$$C_j \cap (C_1 \cup \ldots \cup C_{j-1}) \subseteq C_i,$$

for some $i < j$, and for $j = 2, \ldots, m$. We define the corresponding separator sets

$$S_j = C_j \cap (C_1 \cup \ldots \cup C_{j-1}),$$

with $S_1 = \emptyset$. Once we have found the clique ordering and the corresponding separator sets, the maximum likelihood fitted counts are given by:

$$\hat{n}(x) = \frac{\prod_{j=1}^{m} n(x_{C_j})}{\prod_{j=2}^{m} n(x_{S_j})}$$

where $n(x_\emptyset) = N$.

If the cliques of a decomposable model are presented in RIP order to IPF, then the algorithm will converge in one iteration (one cycle through all cliques). Otherwise IPF will converge in two iterations.

Consider the graph given in figure 7. We observe that the cliques are: AC, BC, CDE, DEF.

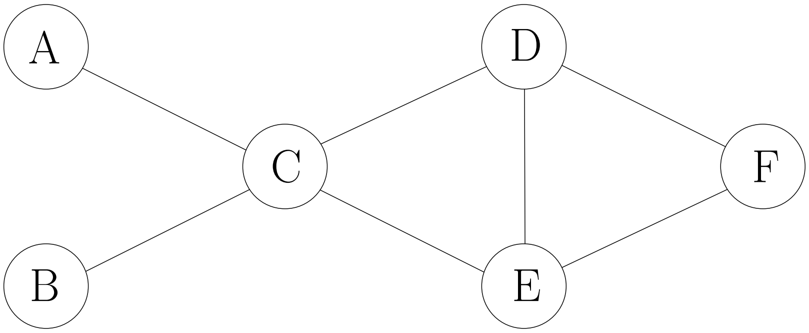


Figure 6: Example independence graph

The clique ordering $C_1 = AC, C_2 = DEF, C_3 = BC, C_4 = CDE$ does not have the running intersection property, as can been seen from the following table:

	C	S
1	AC	$\emptyset$
2	DEF	$\emptyset$
3	BC	C
4	CDE	CDE

Here $S_4 = CDE$ is not a subset of any of the preceding cliques C_1, C_2, C_3.

The clique ordering $C_1 = AC, C_2 = BC, C_3 = CDE, C_4 = DEF$ does have the running intersection property, as can been seen from the following table:

	C	S
1	AC	$\emptyset$
2	BC	C
3	CDE	C
4	DEF	DE

Each separator is a subset of (at least) one of the preceding cliques, hence the given ordering is a RIP ordering of the cliques and can serve to find the formula for the maximum likelihood estimates:

$$\hat{n}(ABCDEF) = \frac{n(AC)n(BC)n(CDE)n(DEF)}{n(C)^2 n(DE)}$$

An undirected graph is decomposable if and only if its cliques can be ordered as a RIP-ordering. The following two algorithms allow us to determine whether a graph is decomposable, and if so, to produce a RIP-ordering.

Algorithm 2 (Maximum Cardinality Search) *Let $G = (V, E)$ be an undirected graph.*

1. *Pick any vertex and label this vertex as $|V|$.*
2. *Let $\ell \leftarrow |V| - 1$.*
3. *As the next vertex to be labeled, select the unlabeled vertex adjacent to the largest number of labeled vertices. Break ties arbitrarily. Label the vertex as ℓ.*
4. *$\ell \leftarrow \ell - 1$.*
5. *Repeat from step 3 until all vertices are labeled.*

Algorithm 3 (Test decomposability) *Let $G = (V, E)$ be an undirected graph with vertices labeled by maximum cardinality search, and let $\mathcal{C}$ denote the set of cliques of G. Let m denote the number of cliques.*

1. *Let $j \leftarrow 1$, $k \leftarrow m$, and $\mathcal{R} \leftarrow \mathcal{C}$.*

2. *Consider the vertex labeled j. If $j \in C'$ and $j \in C''$ for $C' \neq C'' \in \mathcal{R}$, then stop and G is not decomposable. If $j \in C$ for $C \in \mathcal{R}$, then let $C_k \leftarrow C$, $\mathcal{R} \leftarrow \mathcal{R} \setminus C$ and $k \leftarrow k - 1$.*

3. *If $\mathcal{R} = \emptyset$ then stop: G is decomposable and $C_1, \ldots, C_m$ is a RIP ordering. Otherwise let $j \leftarrow j + 1$ and repeat from step 2.*

8 Deviance and Likelihood Ratio Test

The deviance of a fitted model compares the log-likelihood of the fitted model to the log-likelihood of the saturated model. The larger the model deviance, the poorer the fit. The likelihood score of a model M is

$$L(\hat{P}(x); n(x)) = \prod_x \hat{P}(x)^{n(x)}$$

where $\hat{P}(x)$ are the ML estimates of the cell probabilities for model M. This is of course just the probability of the data given $\hat{P}$.

Hence, the log-likelihood score of a model M is

$$\mathcal{L}(\hat{P}(x); n(x)) = \sum_x n(x) \log \hat{P}(x)$$

For example, suppose we have the following table of observed counts:

$n(x_1, x_2)$		$x_2 = 0$	$x_2 = 1$	$n_1(x_1)$
	$x_1 = 0$	30	10	40
	$x_1 = 1$	30	30	60
	$n_2(x_2)$	60	40	100

We have already seen that the independence model gives estimates

$$\hat{P}(0,0) = 0.24,\ \hat{P}(0,1) = 0.16,\ \hat{P}(1,0) = 0.36,\ \hat{P}(1,1) = 0.24$$

So the probability of the observed table for this model is

$$L = 0.24^{30} \times 0.16^{10} \times 0.36^{30} \times 0.24^{30},$$

which is a very small number. The log-likelihood is

$$\mathcal{L} = 30 \log 0.24 + 10 \log 0.16 + 30 \log 0.36 + 30 \log 0.24 \approx -134.6$$

Since for the saturated model we have

$$\hat{P}(x) = \frac{n(x)}{N},$$

the log-likelihood of the saturated model is

$$\mathcal{L}^{\text{sat}} = \sum_x n(x) \log \frac{n(x)}{N}$$

So for the saturated model the log-likelihood value is

$$\mathcal{L} = 30 \log 0.3 + 10 \log 0.1 + 30 \log 0.3 + 30 \log 0.3 \approx -131.4$$

The log-likelihood value of the saturated model is of course always higher than for any other model. The saturated model gives the best possible fit.

The deviance of M is twice the difference between the log-likelihood of the saturated model and the log-likelihood of M, i.e.

$$\begin{aligned} \text{dev}(M) &= 2 \left(\sum_x n(x) \log \frac{n(x)}{N} - \sum_x n(x) \log \hat{P}^M(x) \right) \\ &= 2 \sum_x n(x) \log \frac{n(x)}{\hat{P}^M(x) N} \end{aligned}$$

which can be summarised by the *slogan*

$$\text{dev}(M) = 2 \sum_{\text{cells}} \text{observed} \times \log \frac{\text{observed}}{\text{fitted}}$$

The deviance of the independence model in the previous example is

$$\text{dev(independence model)} = 2(-131.4 + 134.6) = 6.4$$

Let

$$\mathcal{L}^i = \mathcal{L}(\hat{P}^{M_i})$$

be the value of the log-likelihood function evaluated at $\hat{P}^{M_i}$; the ML estimates of P under M_i. Let $M_0 \subseteq M_1$; i.e. M_0 can be obtained from M_1 by imposing additional restrictions (setting additional u-terms to zero). The deviance difference between M_0 and M_1 is

$$\text{dev}(M_0) - \text{dev}(M_1) = -2\mathcal{L}^0 + 2\mathcal{L}^1 = 2(\mathcal{L}^1 - \mathcal{L}^0)$$

We state without proof that for large N

$$2(\mathcal{L}^1 - \mathcal{L}^0) \approx_{M_0} \chi^2_\nu$$

where χ^2_ν denotes the chi-square distribution with ν degrees of freedom, and the degrees of freedom ν is equal to the number of additional restrictions of M_0 compared to M_1. This result will be the basis for subsequent model testing. We reject the null hypothesis that M_0 is the true model when

$$2(\mathcal{L}^1 - \mathcal{L}^0) > \chi^2_{\nu;\alpha},$$

where $P(X > \chi^2_{\nu;\alpha}) = \alpha$ for random variable X with χ^2_ν distribution.

Remark 1 *The test is called a likelihood ratio test because we are looking at logs, and*

$$\log \frac{L^1}{L^0} = \log L^1 - \log L^0 = \mathcal{L}^1 - \mathcal{L}^0$$

We show how the likelihood ratio test can be used to test whether a model gives an adequate fit of the data. Does

$$\text{survival} \perp\!\!\!\perp \text{care} \mid \text{clinic} \tag{7}$$

give a good fit of the observed table? To test this we perform a likelihood ratio test against the saturated model. We fit the model and compute its deviance:

$$2 \sum_{\text{cells}} \text{observed} \times \log \frac{\text{observed}}{\text{fitted}} \approx 0.082$$

Now we have to determine the appropriate degrees of freedom for the test. Since (7) imposes two additional constraints (two u-terms to zero) compared to the saturated model, we compute

$$\chi^2_{2;\,0.05} \approx 6$$

Since the deviance difference is not significant at the 5% level, we accept model (7) See figure 7.

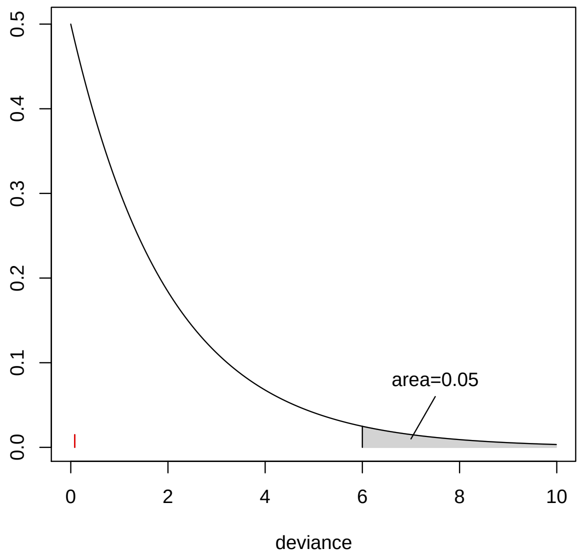


Figure 7: The χ^2_2 distribution. The critical value $\chi^2_{2;\,0.05}$ is approximately equal to 6. The observed deviance is indicated by the red bar near zero.

Does the mutual independence model give a good fit of the observed table? Compute

$$2 \sum_{\text{cells}} \text{observed} \times \log \frac{\text{observed}}{\text{fitted}} \approx 211$$

Now, since

$$\chi^2_{4;\,0.05} \approx 9.5$$

we reject the mutual independence model at the 5% level. See figure 8.

9 Fitting Hierarchical Loglinear Models in R

Our preferred data analysis system `R` contains a function called `loglin` for fitting hierarchical loglinear models. To specify the model you want to fit, you have to list the highest order interaction terms. Here's the clinic example in `R`:

```
> a <- array(c(3,17,4,2,176,197,293,23),dim=c(2,2,2),
```

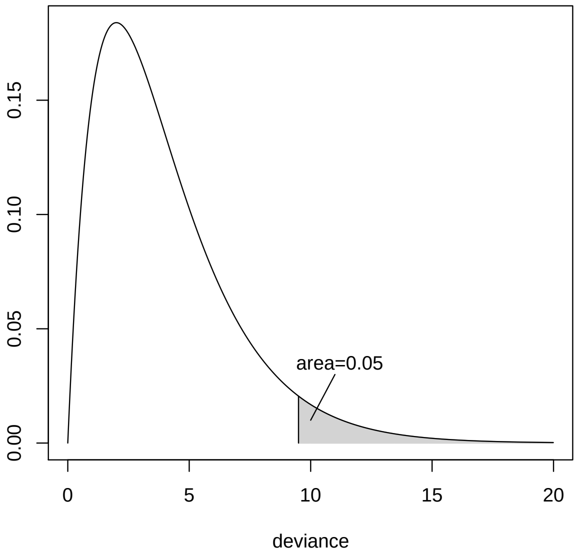


Figure 8: The χ^2_4 distribution. The critical value $\chi^2_{4;\,0.05}$ is approximately equal to 9.5. The observed deviance is somewhere off the page.

```
       dimnames=list(c("clinic 1","clinic 2"),
                     c("less","more"),c("no","yes")))

> a <- as.table(a)
> names(dimnames(a)) <- c("clinic","care","survival")

> a
, , survival = no

          care
clinic     less more
  clinic 1    3    4
  clinic 2   17    2

, , survival = yes

          care
clinic     less more
```

```
  clinic 1  176  293
  clinic 2  197   23
```

We entered the data as an array. The array data is given as a single vector, with the leftmost subscript moving fastest. Since the function `loglin` expects a table rather than an array, we convert it to a table. Finally, we add the variable names and print the data. We start by fitting the model where care and survival are independent given clinic:

```
> model.1 <- loglin(a,margin=list(c("clinic","care"),c("clinic","survival")),
                    fit=TRUE)
2 iterations: deviation 0
> model.1
$lrt
[1] 0.08228918

$pearson
[1] 0.08361853

$df
[1] 2

$margin
$margin[[1]]
[1] "clinic" "care"

$margin[[2]]
[1] "clinic"   "survival"


$fit
, , survival = no

          care
clinic           less      more
  clinic 1   2.632353  4.367647
  clinic 2  17.012552  1.987448
```

```
, , survival = yes

          care
clinic           less       more
  clinic 1 176.367647 292.632353
  clinic 2 196.987448  23.012552
```

The first argument we pass to loglin is the table with observed counts. The second argument specifies the model that has to be fitted by giving the list of highest order interaction terms. The call to loglin returns a list with a number of components. The component `lrt` gives the likelihood ratio test statistic (model deviance), and the component `df` gives the appropriate degrees of freedom (number of u-terms set to zero). Since in the call we specified `fit = TRUE`, the table with the fitted counts is also returned.

As a second example, we fit the independence model:

```
> model.2 <- loglin(a,margin=list(c("clinic"),c("care"),c("survival")),
                    fit=TRUE)
2 iterations: deviation 3.552714e-15
> model.2
$lrt
[1] 211.4820

$pearson
[1] 199.6457

$df
[1] 4

$margin
$margin[[1]]
[1] "clinic"

$margin[[2]]
[1] "care"

$margin[[3]]
```

```
[1] "survival"
```

```
$fit
, , survival = no

          care
clinic           less       more
  clinic 1   9.513948   7.795143
  clinic 2   4.776961   3.913948

, , survival = yes

          care
clinic           less       more
  clinic 1 252.119619 206.571291
  clinic 2 126.589472 103.719619
```

We observed from the output that the deviance of the independence model is 211.482. To perform the appropriate test in R, we can find the critical value for $\alpha = 0.05$ as follows:

```
> qchisq(0.05,df=4,lower.tail=F)
[1] 9.487729
> qchisq(0.95,df=4)
[1] 9.487729
```

The function qchisc gives the value of the test statistic for which $P(X^2 < c) = \alpha$ where X^2 is a random variable with chi-square distribution with df degrees of freedom. Since we actually want the value for which $P(X^2 > c) = \alpha$, we can either specify this explicitely, or pass $1 - \alpha$ instead of α to the function.

We can also fit log-linear models via so-called Poisson regression for count data. This gives us the convenient formula interface of glm, and also produces estimates of the u terms as they are described in these lecture notes (loglin uses different constraints to identify the u terms, so its estimates of the u

terms are also different). We first have to convert the contingency table to a data frame with counts:

```
> clinic.count <- as.data.frame(a)
> clinic.count
    clinic care survival Freq
1 clinic 1 less       no    3
2 clinic 2 less       no   17
3 clinic 1 more       no    4
4 clinic 2 more       no    2
5 clinic 1 less      yes  176
6 clinic 2 less      yes  197
7 clinic 1 more      yes  293
8 clinic 2 more      yes   23
```

As an example, we fit the homogeneous association model via Poisson regression:

```
> clinic.loglin <- glm(Freq ~ survival*clinic + survival*care +
            clinic*care,family="poisson",data=clinic.count)

Coefficients:
                           Estimate Std. Error z value Pr(>|z|)
(Intercept)                  1.0343     0.5053   2.047 0.040647 *
survivalyes                  4.1372     0.5077   8.149 3.66e-16 ***
clinicclinic 2               1.8098     0.5272   3.433 0.000598 ***
caremore                     0.3976     0.5606   0.709 0.478165
survivalyes:clinicclinic 2  -1.6991     0.5307  -3.202 0.001365 **
survivalyes:caremore         0.1104     0.5610   0.197 0.844028
clinicclinic 2:caremore     -2.6467     0.2339 -11.317  < 2e-16 ***
```

From the `R` output we can read, for example, that $\hat{u}_0 = 1.0343$ and $\hat{u}$(survival=yes,care=more) $= 0.1104$.

10 Model Selection

In the previous section we have shown how to fit a single hierarchical loglinear model in `R`. To get a data mining algorithm, you have to superimpose some search strategy to search the model space. You also need a way to measure model quality.

Akaike's Information Criterion assigns quality AIC(M) to model M as follows

$$\mathrm{AIC}(M) = \mathrm{dev}(M) + 2\mathrm{dim}(M)$$

where dim(M) is the number of parameters of the model. This quality measure consists of two components: the lack-of-fit of the model as measured by the deviance, and the complexity of the model as measured by the number of parameters (i.e. the number of u-terms not constrained to be equal to zero). Notice the analogy with the total cost of a tree in cost-complexity pruning. By including the penalty for complexity we try to avoid overfitting. If we did not include this penalty term the saturated model would always win. Now it is possible that we prefer a simpler model that has a worse fit, over a more complex model. We give an example of stepwise search with AIC. To begin with, we fit a loglinear model that will be used as the initial model from which the search starts. We use a frontend to `loglin` available in the library `MASS`:

```
> library(MASS)
> model.init <- loglm( ~ clinic + care + survival,data=a)
> model.init
Call:
loglm(formula = ~clinic + care + survival, data = a)

Statistics:
                      X^2 df P(> X^2)
Likelihood Ratio 211.4820  4        0
Pearson          199.6457  4        0
```

The `loglm` function actually calls the function `loglin` that we used before, but allows (or requires, depending on your preference) you to specify the model differently. The first argument is a formula where on the right hand side of the tilde, you specify the highest order interaction terms. For example, to fit the homogeneous association model, the call should be:

```
> model.6 <- loglm( ~ clinic*care+clinic*survival+care*survival,data=a)
> model.6
Call:
loglm(formula = ~clinic*care + clinic*survival + care*survival,
    data = a)

Statistics:
                        X^2 df  P(> X^2)
Likelihood Ratio 0.04334249  1 0.8350817
Pearson          0.04410757  1 0.8336536
```

The reason we use `loglm` rather than `loglin` is that the stepwise search performed by `stepAIC` requires the format returned by `loglm`. Here we use `stepAIC` to search the model space:

```
> model.step <- stepAIC(model.init,scope= ~ clinic*care*survival)
Start:  AIC=219.48
~clinic + care + survival

                  Df    AIC
+ clinic:care      1  27.83
+ clinic:survival  1 203.74
+ care:survival    1 215.87
<none>              219.48
- care             1 224.54
- clinic           1 297.55
- survival         1 985.30

Step:  AIC=27.83
~clinic + care + survival + clinic:care

                  Df    AIC
+ clinic:survival  1  12.08
+ care:survival    1  24.22
<none>               27.83
- clinic:care      1 219.48
- survival         1 793.65
```

```
Step:  AIC=12.08
~clinic + care + survival + clinic:care + clinic:survival

                  Df     AIC
<none>                12.082
+ care:survival    1  14.043
- clinic:survival  1  27.828
- clinic:care      1 203.736
```

The first argument of the call to `stepAIC` specifies the initial model, where the search process starts. The second argument specifies the scope of the search. In general, we can specify a lower and upper model here. The lower model is the least complex model considered and the upper model the most complex. In our example we only specify one model, in which case it is taken to be the upper model. The lower model is taken to be the empty model (i.e. containing only $u_\varnothing$) in that case. Here we specify the upper model to be the saturated model. What follows is a report of the search process. For example, the initial model has an AIC value of 219.48, because the deviance of this model (as we saw before) is 211.48 and we have to add to that two times the number of parameters. Since the model has four u-terms, we have to add eight to get the AIC value. Then we get a list of *neighbouring* models sorted from low AIC value to high AIC value. A neighbouring model is any hierarchical model that can be obtained by adding a term (either single variable or interaction) or removing a term from the current model. If there is a neighbouring model with a lower AIC value than the current model, we move to that neighbouring model; otherwise the search stops. We see for example that adding the interaction term `clinic:care` to the initial model produces a model with AIC score 27.83. This is better than the current model (listed as `<none>` in the table), and also better than other neighbours, so we move to this model. Note that in the second step, the search does not consider the removal of either `clinic` or `care`, since their removal would produce a non-hierarchical model. The `anova` component of the call to `stepAIC` summarizes the search process:

```
> model.step$anova
```

```
Stepwise Model Path
Analysis of Deviance Table

Initial Model:
~clinic + care + survival

Final Model:
~clinic + care + survival + clinic:care + clinic:survival


              Step Df  Deviance Resid. Df   Resid. Dev       AIC
1                                       4 211.48204459 219.48204
2     + clinic:care  1 193.65365         3  17.82839924  27.82840
3 + clinic:survival  1  17.74611         2   0.08228918  12.08229
```

Note that `stepAIC` searches the space of hierarchical models, not the space of graphical models. Using the function `loglin` it should not be too hard though to write your own graphical model search function. The elementary operations could be adding or removing an edge between two variables. When you add an edge, you have to make sure that you detect whether this forms a clique in the graph.If so, you should include the corresponding interaction term. Likewise, when removing an edge, you have to detect whether this breaks up a clique.

11 Log-linear models and logistic regression

In the beginning we have stated that log-linear models are used to model the associations between a collection of variables, where all variables are treated equally. Nevertheless, we may at some point be interested in the conditional distribution of one variable given the remaining variables. To syntactically distinguish it, we will call this variable y, and the remaining variables $x_1, \ldots, x_m$. Let's start with a simple example, with three binary variables: y, x_1, x_2. As usual the values are coded as 0 and 1. Let's study the conditional independence model $x_1 \perp\!\!\!\perp x_2 \mid y$. Recall it has the following

Algorithm 2 Hill Climbing for Hierarchical Models

```
$M \leftarrow$ initial model
max $\leftarrow$ score($M$)
**repeat**
  nb.add $\leftarrow \{a \subseteq \{1, 2, ..., k\} | a \notin M \wedge \forall a' \subset a : a' \in M\}$
  nb.del $\leftarrow \{a \subseteq \{1, 2, ..., k\} | a \in M \wedge \forall a' \supset a : a' \notin M\}$
  $M^{\text{old}} \leftarrow M$
  **for all** $a \in$ nb.add **do**
    **if** score $(M^{\text{old}} \cup \{a\}) >$ max **then**
      max $\leftarrow$ score($M^{\text{old}} \cup \{a\}$)
      $M \leftarrow M^{\text{old}} \cup \{a\}$
    **end if**
  **end for**
  **for all** $a \in$ nb.del **do**
    **if** score $(M^{\text{old}} \setminus \{a\}) >$ max **then**
      max $\leftarrow$ score($M^{\text{old}} \setminus \{a\}$)
      $M \leftarrow M^{\text{old}} \setminus \{a\}$
    **end if**
  **end for**
**until** $M = M^{\text{old}}$
**return** $M$
```

log-linear expansion:

$$\log P(y, x_1, x_2) = u_0 + u_1 x_1 + u_2 x_2 + u_y y + u_{1y} x_1 y + u_{2y} x_2 y$$

Or, exponentiating both sides:

$$\begin{aligned} P(y, x_1, x_2) &= e^{u_0 + u_1 x_1 + u_2 x_2 + u_y y + u_{1y} x_1 y + u_{2y} x_2 y} \\ &= e^{u_0} e^{u_1 x_1} e^{u_2 x_2} e^{u_y y} e^{u_{1y} x_1 y} e^{u_{2y} x_2 y} \qquad (8) \end{aligned}$$

Now

$$P(y = 1 \mid x_1, x_2) = \frac{P(y = 1, x_1, x_2)}{P(x_1, x_2)} = \frac{P(y = 1, x_1, x_2)}{P(y = 0, x_1, x_2) + P(y = 1, x_1, x_2)}$$

Filling in the appropriate expressions from equation 8, we get:

$$\begin{aligned} P(y = 1 \mid x_1, x_2) &= \frac{e^{u_0} e^{u_1 x_1} e^{u_2 x_2} e^{u_y} e^{u_{1y} x_1} e^{u_{2y} x_2}}{e^{u_0} e^{u_1 x_1} e^{u_2 x_2} + e^{u_0} e^{u_1 x_1} e^{u_2 x_2} e^{u_y} e^{u_{1y} x_1} e^{u_{2y} x_2}} \\ &= \frac{e^{u_0} e^{u_1 x_1} e^{u_2 x_2} \left(e^{u_y} e^{u_{1y} x_1} e^{u_{2y} x_2}\right)}{e^{u_0} e^{u_1 x_1} e^{u_2 x_2} \left(1 + e^{u_y} e^{u_{1y} x_1} e^{u_{2y} x_2}\right)} \qquad (9) \\ &= \frac{e^{u_y} e^{u_{1y} x_1} e^{u_{2y} x_2}}{1 + e^{u_y} e^{u_{1y} x_1} e^{u_{2y} x_2}} \qquad (10) \\ &= \frac{e^{u_y + u_{1y} x_1 + u_{2y} x_2}}{1 + e^{u_y + u_{1y} x_1 + u_{2y} x_2}} \\ &= \frac{e^{\beta_0 + \beta_1 x_1 + \beta_2 x_2}}{1 + e^{\beta_0 + \beta_1 x_1 + \beta_2 x_2}} \end{aligned}$$

Note that $e^{u_0} e^{u_1 x_1} e^{u_2 x_2}$ cancels out when going from line (9) to line (10). Convince yourself that in general any term that does not involve y will cancel out in this step. In the last line we replaced the u term notation by the notation used in most logistic regression literature, to make it even more evident that the functional form for the conditional distribution of y in the log-linear model is identical to the logistic regression assumption. Does this mean that if we estimate the log-linear model (using maximum likelihood estimation) and we also estimate the logistic regression model (using maximum likelihood), that we will find:

$$\hat{u}_y = \hat{\beta}_0, \qquad \hat{u}_{1y} = \hat{\beta}_1 \qquad \hat{u}_{2y} = \hat{\beta}_2?$$

The answer is no. The reason is that in log-linear modeling we maximize the joint likelihood (that is, the likelihood function based on the joint probability of all variables), whereas in logistic regression we maximize the conditional likelihood of y given the remaining variables. So even though the functional form is the same, the coefficient estimates will be different. There is one log-linear model however, that yields the same estimates as logistic regression. In the above example, that would be the model that also has an association between x_1 and x_2. Recall that all expressions that do not involve y cancel out, so in this (homogeneous association) model $P(y \mid x_1, x_2)$ has the same functional form as in the conditional independence model. In general it is the hierarchical model with the following highest order interaction terms:

- $[x_1 x_2 \cdots x_m]$: the saturated model on the explanatory variables.
- $[x_i y]$ for all $i = 1, \ldots, m$: one-way association between y and each explanatory variable.

This model is not graphical since its independence graph is the complete graph, but obviously it doesn't correspond to the saturated model. Note also that the above equivalence gives credence to the statement that in logistic regression we don't make any assumptions about the marginal distribution of x.

In fact, the log-linear conditional independence model corresponds to the naive Bayes model, a popular classifier that will be discussed in the context of directed graphical models.

Recall that the log-linear expansion of the probability distribution P_K is given by:

$$\log P_K(x) = \sum_{a \subseteq K} u_a(x_a)$$

where the sum is taken over all possible subsets a of $K = \{1, 2, \ldots, k\}$.
The full log-linear expansion on y, x_1 and x_2 is given by

$$\begin{aligned}\log P(y, x_1, x_2) = u_\emptyset + u_y(y) + u_1(x_1) + u_2(x_2) + u_{y1}(y, x_1) \\ + u_{y2}(y, x_2) + u_{12}(x_1, x_2) + u_{y12}(y, x_1, x_2)\end{aligned}$$

Or (take the exponent left and right):

$$P(y, x_1, x_2) = e^{u_\emptyset + u_y(y) + u_1(x_1) + u_2(x_2) + u_{y1}(y,x_1) + u_{y2}(y,x_2) + u_{12}(x_1,x_2) + u_{y12}(y,x_1,x_2)}$$

Now

$$P(y \mid x_1, x_2) = \frac{P(y, x_1, x_2)}{P(x_1, x_2)} = \frac{P(y, x_1, x_2)}{\sum_{y'} P(y', x_1, x_2)}$$

So we get:

$$P(y \mid x_1, x_2) = \frac{e^{u_\emptyset + u_y(y) + u_1(x_1) + u_2(x_2) + u_{y1}(y,x_1) + u_{y2}(y,x_2) + u_{12}(x_1,x_2) + u_{y12}(y,x_1,x_2)}}{\sum_{y'} e^{u_\emptyset + u_y(y') + u_1(x_1) + u_2(x_2) + u_{y1}(y',x_1) + u_{y2}(y',x_2) + u_{12}(x_1,x_2) + u_{y12}(y',x_1,x_2)}}$$

All terms in the denominator that do not contain y can be pulled in front of the summation, and cancel against identical terms in the numerator. So the expression simplifies to:

$$P(y \mid x_1, x_2) = \frac{e^{u_y(y) + u_{y1}(y,x_1) + u_{y2}(y,x_2) + u_{y12}(y,x_1,x_2)}}{\sum_{y'} e^{u_y(y') + u_{y1}(y',x_1) + u_{y2}(y',x_2) + u_{y12}(y',x_1,x_2)}}$$

This is almost like a first-order logistic regression model, except for the term $u_{y12}(y, x_1, x_2)$. To get rid of it, we have to remove it from the log-linear expansion, which gives us the homogeneous association model:

$$\begin{aligned}\log P(y, x_1, x_2) = u_\emptyset + u_y(y) + u_1(x_1) + u_2(x_2) + u_{y1}(y, x_1) \\ + u_{y2}(y, x_2) + u_{12}(x_1, x_2)\end{aligned}$$

Now by the same reasoning as before we get:

$$P(y \mid x_1, x_2) = \frac{e^{u_y(y) + u_{y1}(y,x_1) + u_{y2}(y,x_2)}}{\sum_{y'} e^{u_y(y') + u_{y1}(y',x_1) + u_{y2}(y',x_2)}}$$

This is indeed the first-order (multinomial) logistic regression model.

12 Conclusion

Graphical modeling has become a pretty big area in the last decade. We have looked only at a small part of it: undirected graphs for discrete data. Possible extensions are: models for continuous variables or mixed discrete and continuous variables; models represented by directed graphs (e.g. Bayesian Networks), etc. The book of Edwards [Edw00] in combination with the MIM program is a good starting point to get acquainted with the different variations of graphical models around. For graphical models in R, [HEL12] is recommended.

References

[BFH75] Y. Bishop, S.E. Fienberg, and P.W. Holland. *Discrete Multivariate Analysis.* MIT Press, Cambridge(MA), 1975.

[Chr97] R. Christensen. *Log-Linear Models and Logistic Regression (second edition).* Springer, New York, 1997.

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

[HEL12] S. Hojsgaard, D. Edwards, and S. Lauritzen. *Graphical Models with R.* Springer, 2012.

[Sch97] J.L. Schafer. *Analysis of Incomplete Multivariate Data.* Chapman & Hall, London, 1997.

[Whi90] J. Whittaker. *Graphical Models in Applied Multivariate Statistics.* Wiley, Chichester, 1990.