Data Mining 2013
Graphical Models for Discrete Data
Part 1: Undirected Graphs (2)

Ad Feelders

Universiteit Utrecht

October 1, 2013

Overview

- Conditional Independence
- Graphical Representation
- Log-linear Models
  - Hierarchical
  - Graphical
  - Decomposable
- Maximum Likelihood Estimation
- Model Testing/Selection

Bernoulli random variable

A Bernoulli random variable X with probability of success p, has probability density 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 clever way of writing the probability density in one formula; check that indeed $P(1) = p$ and $P(0) = 1 - p$ as required.

2×2 Table

The density function P_{12} of bivariate Bernoulli random vector (X_1, X_2) is determined by

$$P(x_1, x_2) = p(x_1, x_2)$$

where $p(x_1, x_2)$ is 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

Density function for 2×2 Table

We can write this as one function:

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

Taking logarithms and collecting terms in x_1 and x_2 gives

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

Verify this using elementary properties of logarithms:

1. $\log a^b = b \log a$,
2. $\log \frac{a}{b} = \log a - \log b$, and
3. $\log ab = \log a + \log b$.

Log-linear expansion

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

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

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

For example, the coefficient of the product $x_1 x_2$

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

is the logarithm of the cross product ratio of X_1 and X_2.

Independence and u-terms

Claim:

$$X_1 \perp\!\!\!\perp X_2 \Leftrightarrow u_{12} = 0$$

Proof: the factorisation criterion states that $X_1 \perp\!\!\!\perp X_2$ iff 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 $u_{12} = 0$, we get

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

so

$$g(x_1) = u_\emptyset + x_1 u_1 \qquad h(x_2) = x_2 u_2$$

suffices. If $u_{12} \neq 0$, no such decomposition is possible.

Three Dimensional Bernoulli

The joint distribution of three binary variables can be written:

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

Log-linear expansion

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

With

$$\begin{aligned} u_{123} &= \log \frac{p(1,1,1)p(1,0,0)}{p(1,1,0)p(1,0,1)} \cdot \frac{p(0,1,0)p(0,0,1)}{p(0,0,0)p(0,1,1)} \\ &= \log \frac{\text{cpr}(X_2, X_3 | X_1 = 1)}{\text{cpr}(X_2, X_3 | X_1 = 0)} \end{aligned}$$

Independence and the u-terms

Observation:

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

Proof: use factorisation criterion.

$X_2 \perp\!\!\!\perp X_3 | X_1 \Leftrightarrow$ there are functions $g(x_1, x_2)$ and $h(x_1, x_3)$ such that

$$\log P(x_1, x_2, x_3) = g(x_1, x_2) + h(x_1, x_3)$$

This is only possible when $u_{23} = 0$ (so the term $x_2 x_3$ drops out), and $u_{123} = 0$ (so the term $x_1 x_2 x_3$ drops out).

Log-linear expansion: non-binary variables

For a 2×2 table the log-linear expansion is given by:

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

for $x \in \{0, 1\}^2$.

What if the x_i have more than two levels? In that case the u terms become functions of x rather than constants:

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

Log-linear expansion: non-binary variables

Suppose $x \in \{0, 1, 2\}$. We can write

$$P(x) = p(1)^{\delta_{x=1}} p(2)^{\delta_{x=2}} p(0)^{(1-\delta_{x=1}-\delta_{x=2})},$$

where δ_A is the indicator function, that is,

$$\delta_A = \begin{cases} 1 & \text{if } A \text{ is true} \\ 0 & \text{otherwise} \end{cases}$$

Taking logarithms left and right, we get

$$\begin{aligned} \log P(x) &= \delta_{x=1} \log p(1) + \delta_{x=2} \log p(2) + (1 - \delta_{x=1} - \delta_{x=2}) \log p(0) \\ &= \delta_{x=1} \log p(1) + \delta_{x=2} \log p(2) + \log p(0) - \delta_{x=1} \log p(0) - \delta_{x=2} \log p(0) \\ &= \log p(0) + \log \frac{p(1)}{p(0)} \delta_{x=1} + \log \frac{p(2)}{p(0)} \delta_{x=2} \\ &= u_\emptyset + u(x) \end{aligned}$$

Log-linear expansion: non-binary variables

Where

$$u(x) = \begin{cases} \log \frac{p(1)}{p(0)} & \text{if } x = 1 \\ \log \frac{p(2)}{p(0)} & \text{if } x = 2 \\ 0 & \text{if } x = 0 \end{cases}$$

Similar rules apply to the case of multiple non-binary variables.

Log-linear expansion: general

The log-linear expansion of the probability 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 $K = \{1, 2, \ldots, k\}$.

- To avoid getting too many parameters, we set $u_a(x_a) = 0$ whenever $x_i = 0$ and $i \in a$.
- This is analogous to the case where x is binary.

Independence and the u-terms: general

If (X_a, X_b, X_c) is a partitioned random vector $(a \cup b \cup c = \{1, 2, \ldots, k\})$ then $X_b \perp\!\!\!\perp X_c | X_a$ if and only if all u-terms in the log-linear expansion with coordinates in both b and c, are zero.

Example: $X = (X_1, \ldots, X_5)$, $a = \{1, 3\}, b = \{4\}, c = \{2, 5\}$, so $X_b \perp\!\!\!\perp X_c | X_a$ means $X_4 \perp\!\!\!\perp (X_2, X_5) | (X_1, X_3)$. This corresponds to setting u-terms that contain elements from both the sets $\{4\}$ and $\{2, 5\}$ to zero. So set $u_{24}, u_{45}, u_{124}, u_{145}, \ldots, u_{12345}$ to zero.

Otherwise we cannot write

$$\log P(x_1, \ldots, x_5) = g(x_1, x_3, x_4) + h(x_1, x_2, x_3, x_5)$$

Independence and the u-terms: proof

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

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

But this function is of the form $g(x_a, x_b) + h(x_a, x_c)$ and hence $X_b \perp\!\!\!\perp X_c | X_a$ by the factorisation criterion.

Hierarchical Models

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 all 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.

Hence, a hierarchical model is identified by listing its highest order interaction terms.

Hierarchical Models for three dimensions

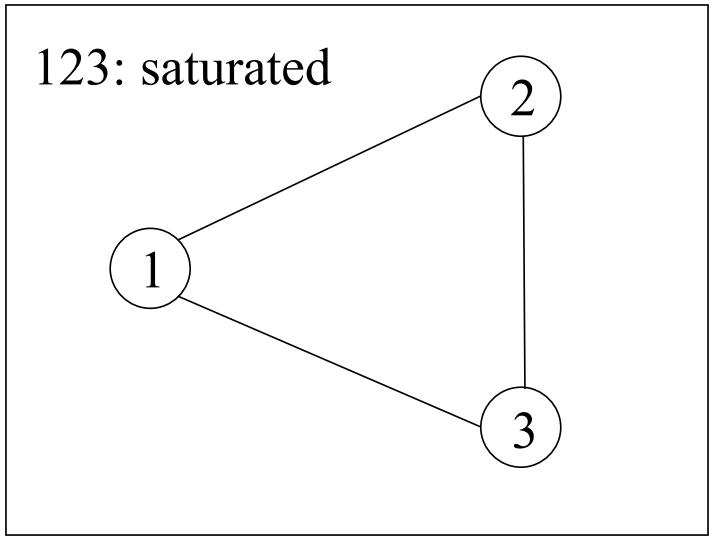
Model	Omitted	Interpretation
123	none	saturated
12,13,23	u_{123}	homogeneous association
12,13	u_{123}, u_{23}	$X_2 \perp\!\!\!\perp X_3 \mid X_1$
12,23	u_{123}, u_{13}	$X_1 \perp\!\!\!\perp X_3 \mid X_2$
13,23	u_{123}, u_{12}	$X_1 \perp\!\!\!\perp X_2 \mid X_3$
12,3	u_{123}, u_{13}, u_{23}	$(X_1, X_2)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

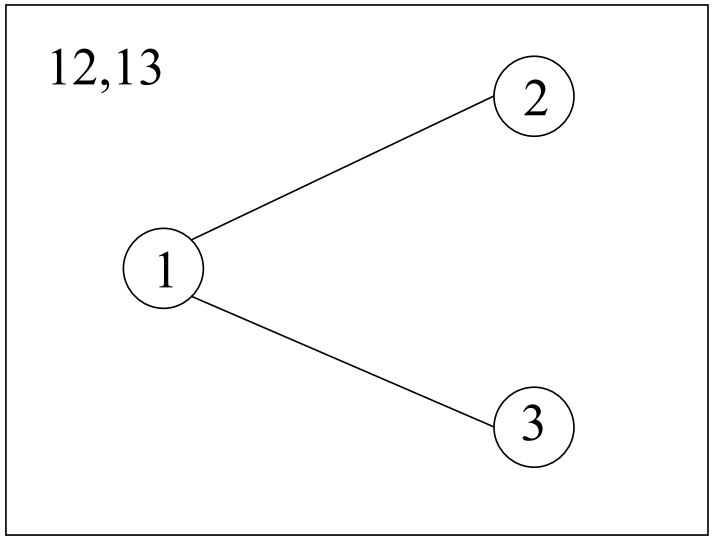
Graphical Log-linear Model

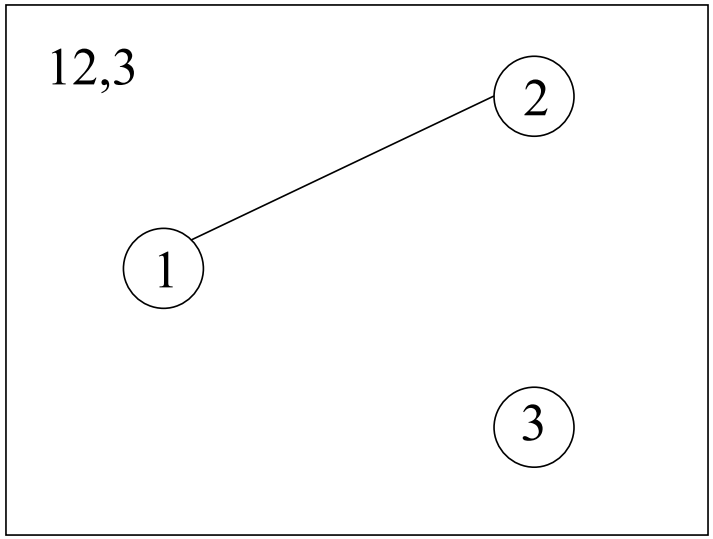
Given its independence graph $G = (K, E)$, the log-linear model for the random vector X is a *graphical model* for X if the distribution of X is *arbitrary* apart from constraints of the form that for all pairs of coordinates not in the edge set E, the u-terms containing the selected coordinates are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

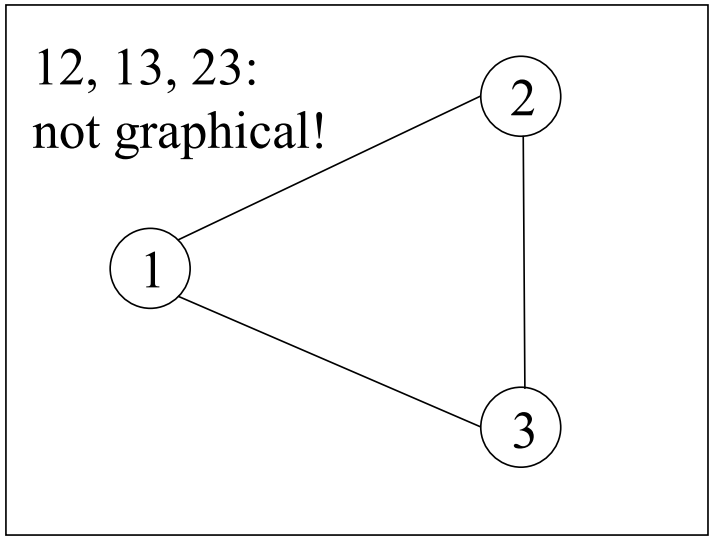
All constraints can be read from the independence graph.

Hierarchical models and their independence graphs









Maximum Likelihood Estimation

- ML estimator of graphical log-linear model M returns estimates of the cell probabilities that maximize the probability of the observed data, subject to the constraint that the conditional independencies of M are satisfied by the estimates.
- ML estimator of graphical log-linear model M satisfies the likelihood equations

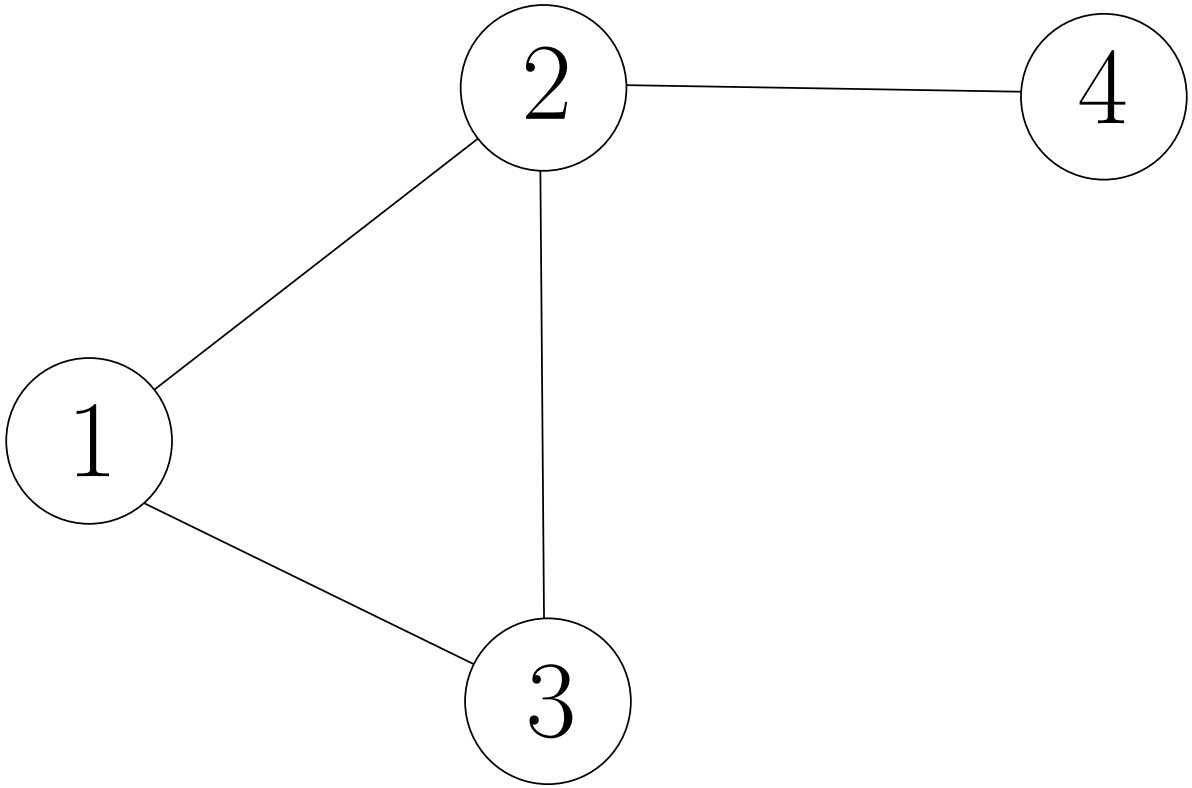
$$\hat{n}_a^M = N\hat{P}_a^M = n_a$$

whenever the subset of vertices a in the graph form a clique.

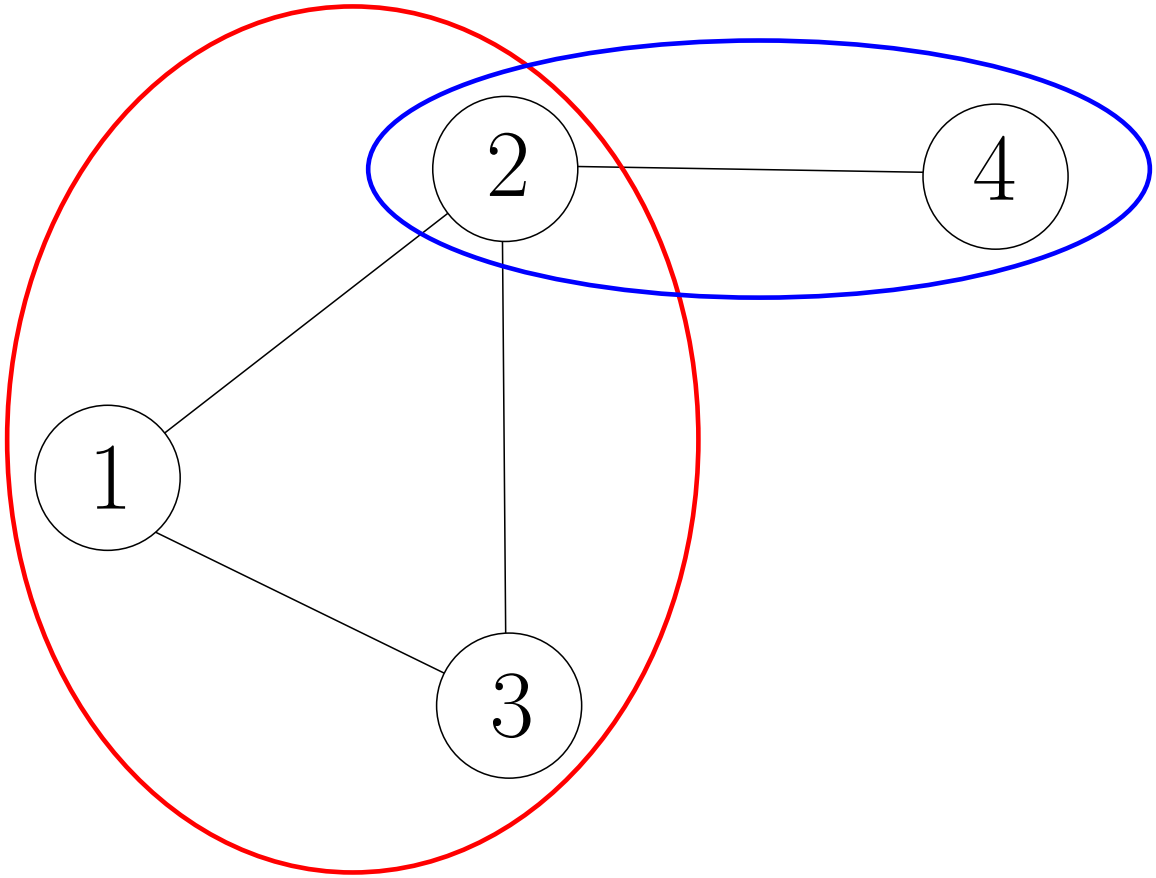
Maximum Likelihood Estimation

- Slogan: Observed = Fitted for every marginal table corresponding to a complete subgraph.
- The same likelihood equations hold for all hierarchical models, where the margins *a* correspond to the highest order interaction terms in the model.

ML: Determine the cliques



ML: Observed=Fitted for margins corresponding to cliques



$$\hat{n}(x_1, x_2, x_3) = n(x_1, x_2, x_3)$$
$$\hat{n}(x_2, x_4) = n(x_2, x_4)$$

ML: Example

$$
\begin{aligned}
\hat{P}(x_1, x_2, x_3, x_4) &= \hat{P}(x_1, x_3, x_4 | x_2)\hat{P}(x_2) && \text{(product rule)} \\
&= \hat{P}(x_1, x_3 | x_2)\hat{P}(x_4 | x_2)\hat{P}(x_2) && (X_4 \perp\!\!\!\perp (X_1, X_3) | X_2) \\
&= \hat{P}(x_1, x_3 | x_2)\hat{P}(x_2, x_4) && \text{(product rule)} \\
&= \frac{\hat{P}(x_1, x_2, x_3)\hat{P}(x_2, x_4)}{\hat{P}(x_2)} && \text{(product rule)}
\end{aligned}
$$

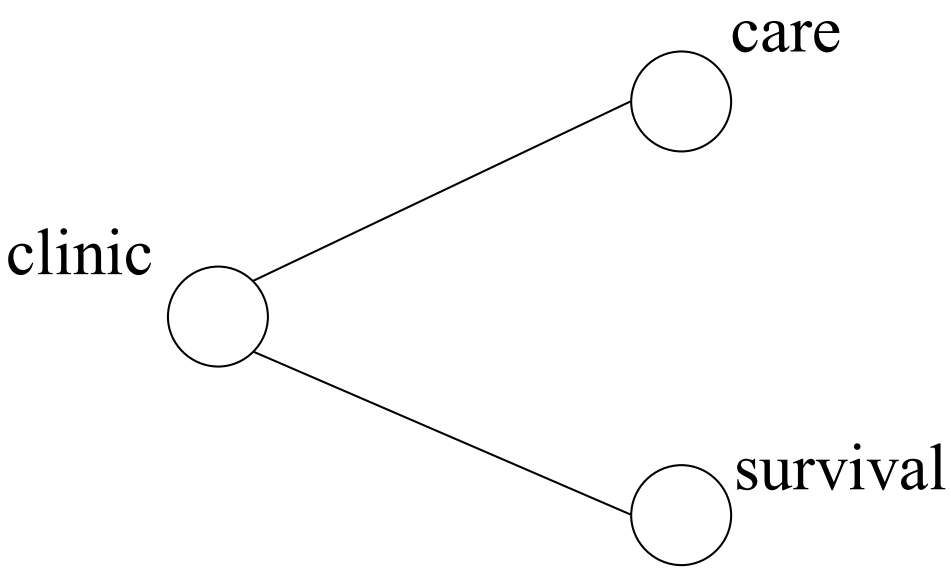
In terms of counts we have:

$$
\begin{aligned}
\hat{n}(x_1, x_2, x_3, x_4) &= \frac{\hat{n}(x_1, x_2, x_3)\hat{n}(x_2, x_4)}{\hat{n}(x_2)} \\
&= \frac{n(x_1, x_2, x_3)n(x_2, x_4)}{n(x_2)} && \text{(fitted = observed for complete subgraph)}
\end{aligned}
$$

In this case we have a closed form solution for the maximum likelihood fitted counts.

ML Estimation: Example

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



Sufficient Statistics

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

Fitted values

$$\hat{n}_{123}(x) = \frac{n_{12}(x_1, x_2)n_{13}(x_1, x_3)}{n_1(x_1)}$$

$\hat{n}_{123}$		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

Model seems to fit very well!

Iterative Proportional Fitting (IPF)

IPF is an algorithm to compute the maximum likelihood fitted counts for hierarchical log-linear models.

Fit 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

Sufficient statistics are row totals $n_1(x_1)$ and column totals $n_2(x_2)$.

Iterative Proportional Fitting

We begin with a table $\hat{n}^{(0)}$ of uniform counts

	0	1	
0	1	1	2
1	1	1	2
	2	2	

First step: fit to row margin

$$\hat{n}(x_1, x_2)^{(1)} = n_1(x_1) \times \frac{\hat{n}(x_1, x_2)^{(0)}}{\hat{n}_1(x_1)^{(0)}}$$

We compute (row 1):

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

Iterative Proportional Fitting

First step continued (row 2):

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

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

	0	1	
0	20	20	40
1	30	30	60
	50	50	

Iterative Proportional Fitting

Second step: fit to column margin

$$\hat{n}(x_1, x_2)^{(2)} = n_2(x_2) \times \frac{\hat{n}(x_1, x_2)^{(1)}}{\hat{n}_2(x_2)^{(1)}}$$

Which gives (first column):

$$\hat{n}(0,0)^{(2)} = 60 \times \frac{20}{50} = 24 \qquad \hat{n}(1,0)^{(2)} = 60 \times \frac{30}{50} = 36$$

and (second column):

$$\hat{n}(0,1)^{(2)} = 40 \times \frac{20}{50} = 16 \qquad \hat{n}(1,1)^{(2)} = 40 \times \frac{30}{50} = 24$$

IPF

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

	0	1	
0	24	16	40
1	36	24	60
	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: Homogeneous association

Fit the model: 12,13,23

IPF proportionally adjusts the estimated expected frequencies $\hat{n}_{123}(x)$ to 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. $\hat{n}_{23}(x_2, x_3) = n_{23}(x_2, x_3)$

IPF: One iteration

Fit to 12 margin:

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

Fit to 13 margin:

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

Fit to 23 margin:

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

IPF: General Algorithm Sketch

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

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

fitting $\hat{n}(x)$ to each margin in turn.

IPF updating rule

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, until convergence is reached.

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

IPF updating rule

To fit to the margin a, the observed count $n_a(x_a)$ on x_a is distributed over $\hat{n}_{ab}(x_a, x_b)^{(t+1)}$ according to

$$\hat{P}(x_b|x_a)^{(t)} = \frac{\hat{n}_{ab}(x_a, x_b)^{(t)}}{\hat{n}_a(x_a)^{(t)}},$$

i.e., the current estimate of $P(X_b = x_b|X_a = x_a)$.

IPF updating rule

Proof:

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

IPF Pseudocode

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

$t \leftarrow 0$
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

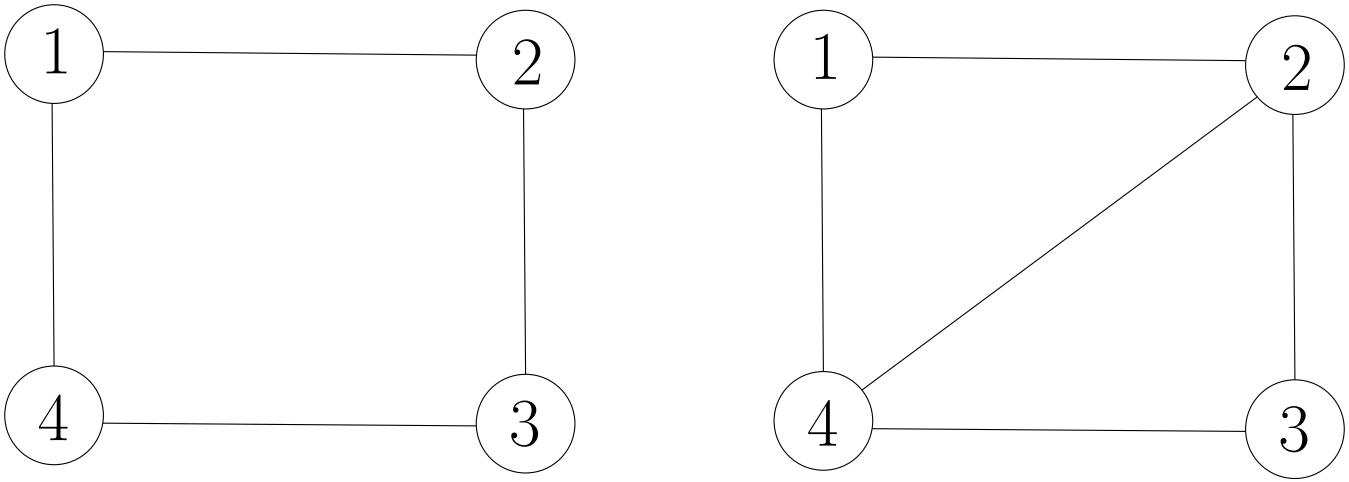
Decomposable Graphical Models

Decomposable models have explicit formulas for the MLE's.

Decomposable models have *triangulated* independence graphs, i.e. have no *chordless cycles* of length greater than three.

A cycle is chordless if only the *successive* pairs of vertices in the cycle are adjacent in the graph (i.e. connected by an edge).

Example



- The left graph is *not* decomposable because it contains the chordless 4-cycle $1-2-3-4-1$.
- The graph on the right *is* decomposable.
  The cycle $1-2-3-4-1$ is no longer chordless because 2 and 4 are adjacent in the graph but not successive in the cycle.

Likelihood and log-likelihood

The likelihood of a model M is

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

where $\hat{P}^M(x)$ is the fitted probability of cell x according to model M.

Hence, the likelihood of model M is the probability of the observed data using the fitted cell probabilities according to model M.

The log-likelihood of a model M is

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

Model Deviance

Since for the saturated model

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

the log-likelihood of the saturated model is

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

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.

Example

Suppose we have data

$n(x)$	$x_2 = 0$	$x_2 = 1$	
$x_1 = 0$	30	10	40
$x_1 = 1$	30	30	60
	60	40	100

The independence model gives probability estimates: $\hat{P}(0,0) = 0.24$, $\hat{P}(0,1) = 0.16$, $\hat{P}(1,0) = 0.36$, $\hat{P}(1,1) = 0.24$.

The probability of the observed data according to this model is

$$0.24^{30} \times 0.16^{10} \times 0.36^{30} \times 0.24^{30}$$

This is the likelihood of the model given the data. The log-likelihood is

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

Example (continued)

Suppose we have data

$n(x)$	$x_2 = 0$	$x_2 = 1$	
$x_1 = 0$	30	10	40
$x_1 = 1$	30	30	60
	60	40	100

The saturated model gives probability estimates: $\hat{P}(0,0) = 0.3$, $\hat{P}(0,1) = 0.1$, $\hat{P}(1,0) = 0.3$, $\hat{P}(1,1) = 0.3$.

The probability of the observed data according to this model is

$$0.3^{30} \times 0.1^{10} \times 0.3^{30} \times 0.3^{30}$$

This is the likelihood of the model given the data. The log-likelihood is

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

Of course this is better than the independence model.

Model Deviance

Deviance of M is 2 (log-likelihood of the saturated model $-$ log-likelihood of M), i.e.

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

which can be summarised by the *slogan*

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

Deviance difference

Let $M_0 \subseteq M_1$, that is M_0 is the simpler model (the u-terms present in M_0 are a subset of the u-terms present in M_1).

The *deviance difference* between M_0 and M_1 is

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

For large N

$$2(\mathcal{L}^{M_1} - \mathcal{L}^{M_0}) \approx_{M_0} \chi^2_\nu$$

ν: number of *additional* restrictions (zero u-terms) of M_0 compared to M_1. (ν is called the degrees of freedom)

Likelihood Ratio Test

We reject the null hypothesis that M_0 is the true model when

$$2(\mathcal{L}^{M_1} - \mathcal{L}^{M_0}) > \chi^2_{\nu;\alpha},$$

where α is the significance level of the test.

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

$$\log \frac{L^{M_1}}{L^{M_0}} = \log L^{M_1} - \log L^{M_0} = \mathcal{L}^{M_1} - \mathcal{L}^{M_0}$$

Model Testing: example

Does

$$\text{survival} \perp\!\!\!\perp \text{care} | \text{clinic}$$

give a good fit of the observed table? Test against the saturated model.

Compute the deviance

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

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

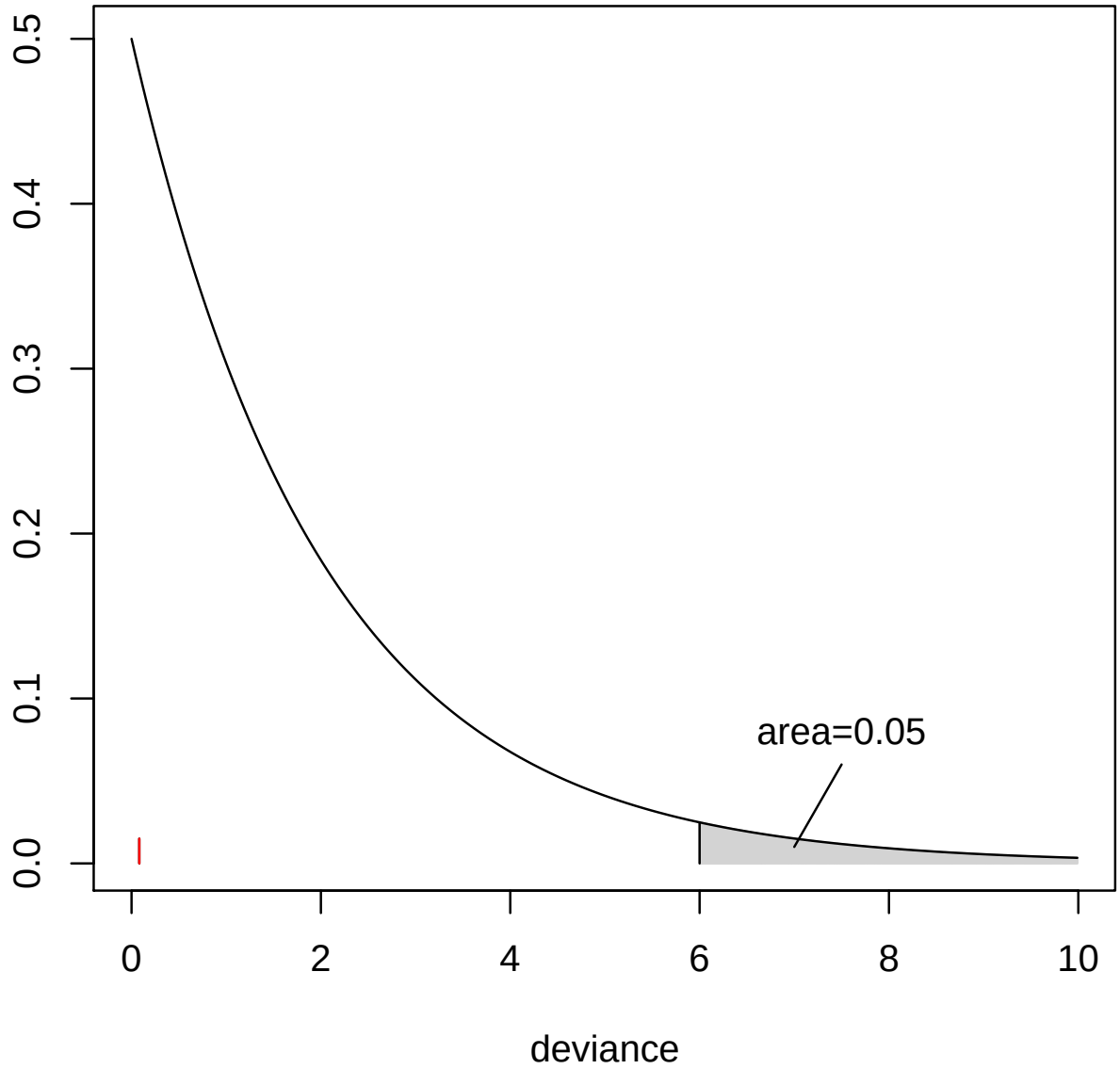
So we "accept" the model.

Fitted Counts and Observed Counts

$\hat{n}$(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

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

Test of survival $\perp\!\!\!\perp$ care|clinic; χ^2_2 distribution.



Model Testing: example

Does the mutual independence model give a good fit of the observed table? Test against the saturated model.

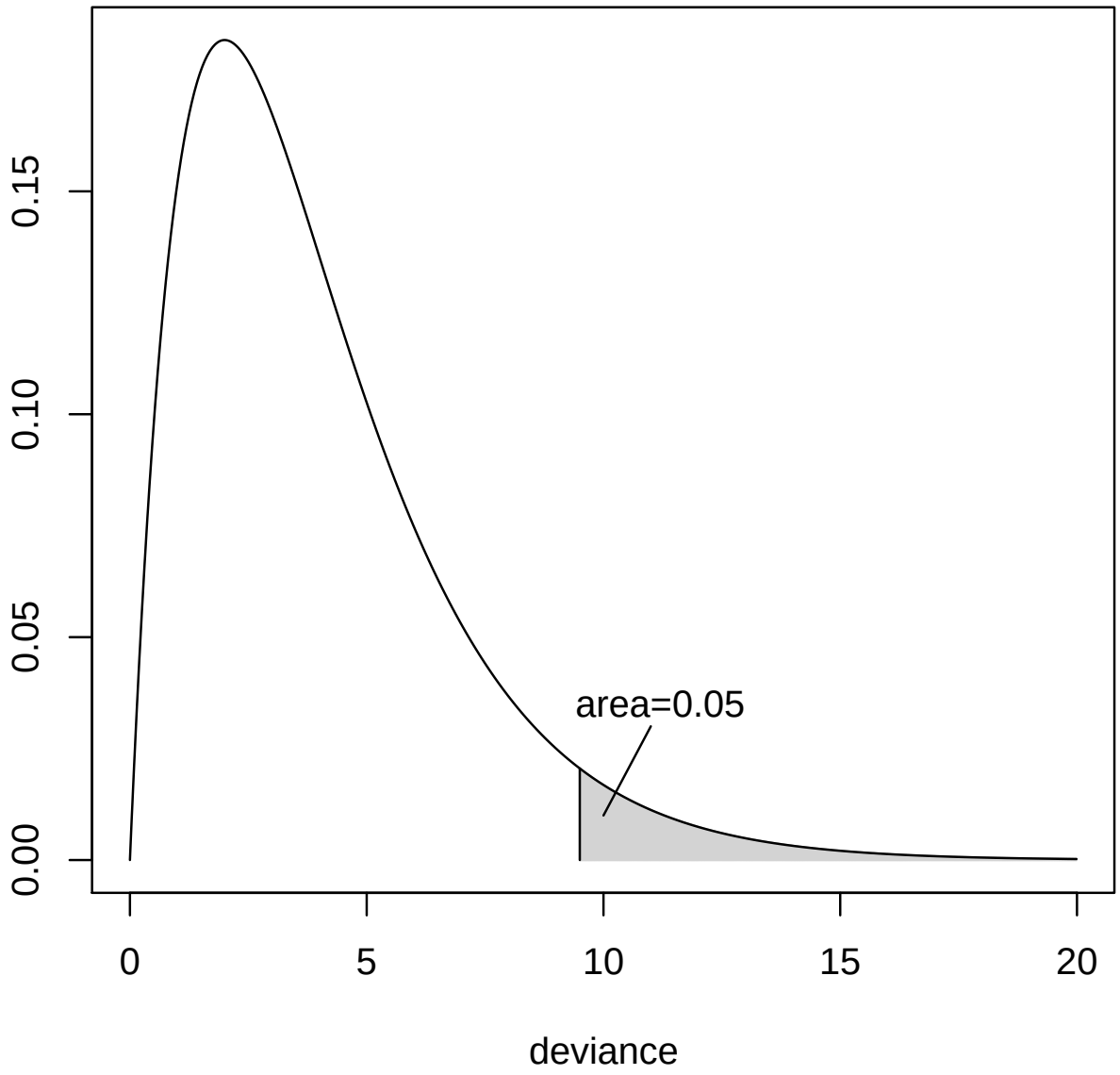
Compute the deviance

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

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

So we reject the mutual independence model.

Test of Independence Model; χ^2_4 distribution.



Fitting Hierarchical Loglinear Models in R

Here's the clinic example in R:

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

Fitting a model

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

$df
[1] 2

$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
```

Model Selection

The Problem: find a good model for a high-dimensional table when little prior knowledge is available.

Solution: Search the space of possible models.
Two approaches:

- Use significance testing
- Use a quality function

Quality Function: Akaike's Information Criterion

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

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

where dim(M) is the number of parameters of the model.

Two components:

- the lack-of-fit of the model
- complexity of the model

Search

Exhaustive search is usually not feasible. A straightforward approach is hill climbing:

1. pick some initial model
2. consider the quality of all neighbors of the current model
3. if they all have lower quality, stop and return the current model.
4. otherwise move to the neighbor with highest quality and return to 2.

Example: Hierarchical Models

1. pick a hierarchical model, e.g. the loglinear model containing just the constant term.
2. neighbors
   - add a term whose lower order terms are all present
   - delete a term whose higher order terms are all absent
3. if all neighbors have higher AIC, stop and return the current model.
4. otherwise move to the neighbor with lowest AIC and return to 2.

Fitting an initial model

`loglm` calls `loglin`, just syntactic sugar

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

Hill climbing with stepAIC

Scope now specifies the upperbound of the search space, that is, the most complex model considered. Here we specified the saturated model.

```
> model.step <- stepAIC(m.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
```

Hill climbing with stepAIC (continued)

```
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
```

Hill climbing with stepAIC

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
```

Decomposable Graphical Models

1. pick an initial model, e.g. the empty graph
2. neighbors
   - add an edge that does not create a chordless cycle of length > 3.
   - delete an edge without creating a chordless cycle of length > 3.
3. if all neighbors have higher AIC, stop and return the current model.
4. otherwise move to the neighbor with lowest AIC and return to 2.

Literature

- J. Whittaker, Graphical Models in Applied Multivariate Statistics, Wiley, 1990.
- D. Edwards, Introduction to Graphical Modelling (2nd edition), Springer, 2000.
- Y. Bishop, S.E. Fienberg, P.W. Holland, Discrete Multivariate Analysis, MIT Press, 1975.