Following the book's notation

- Write the frequencies as $x_1, \ldots, x_k$.

$$x_j = \sum_{i=1}^{N} x_{i,j} \qquad L(\mathbf{p}) = p_1^{x_1} \cdots p_k^{x_k}$$

- Later, *x* values with multiple subscripts will refer to frequencies in a multi-dimensional table, like $x_{i,j,k}$ will be the frequency in row *i* and column *j* of sub-table *k*.
- Write likelihood function as

$$L(\mathbf{p}) = p_1^{x_1} \cdots p_k^{x_k} = p_1^{x_1} \cdots p_{k-1}^{x_{k-1}} (1 - \sum_{j=1}^{k-1} p_j)^{N - \sum_{j=1}^{k-1} x_j}$$

Log likelihood: *p-1* parameters

$$L(\mathbf{p}) = p_1^{x_1} \cdots p_{k-1}^{x_{k-1}} (1 - \sum_{j=1}^{k-1} p_j)^{N - \sum_{j=1}^{k-1} x_j}$$

$$\begin{aligned} \ell(\mathbf{p}) &= \ln L(\mathbf{p}) \\ &= \sum_{i=1}^{k-1} x_i \ln(p_i) + (N - \sum_{i=1}^{k-1} x_i) \ln(1 - \sum_{i=1}^{k-1} p_i) \end{aligned}$$

$$\frac{\partial \ell}{\partial p_j} = \frac{x_j}{p_j} - \frac{N - \sum_{i=1}^{k-1} x_i}{1 - \sum_{i=1}^{k-1} p_i}, \text{ for } j = 1, \ldots, k-1$$

Set all k-1 derivatives to zero and solve for $p_1, \ldots, p_k$. Verify that $p_i = x_i/N$ for $i = 1, \ldots k-1$ works: MLE is the sample mean.

Likelihood Ratio Tests

$$X_1, \ldots, X_N \overset{i.i.d.}{\sim} F_\theta,\ \theta \in \Theta,$$
$$H_0 : \theta \in \Theta_0 \text{ v.s. } H_A : \theta \in \Theta \cap \Theta_0^c,$$

$$G^2 = -2 \ln \left(\frac{\max_{\theta \in \Theta_0} L(\theta)}{\max_{\theta \in \Theta} L(\theta)} \right)$$

Under H_0, G^2 has an approximate chi-square distribution for large N. Degrees of freedom = number of (non-redundant, linear) equalities specified by H_0. Reject when G^2 is large.

Degrees of Freedom

Express H_0 as a set of linear combinations of the parameters, set equal to constants (usually zeros).

Degrees of freedom = number of *non-redundant* linear combinations.

Suppose $\boldsymbol{\theta} = (\theta_1, \ldots \theta_7)$, with

$$H_0 : \theta_1 = \theta_2,\ \theta_6 = \theta_7,\ \frac{1}{3}(\theta_1 + \theta_2 + \theta_3) = \frac{1}{3}(\theta_4 + \theta_5 + \theta_6)$$

df=3

Example

University administrators recognize that the percentage of students who are unemployed after graduation will vary depending upon economic conditions, but they claim that still, about twice as many students will be employed in a job related to their field of study, compared to those who get an unrelated job. To test this hypothesis, they select a random sample of 200 students from the most recent class, and observe 106 employed in a job related to their field of study, 74 employed in a job unrelated to their field of study, and 20 unemployed. Test the hypothesis using a large-sample likelihood ratio test and significance level $\alpha = 0.05$. State your conclusions in symbols and words.

- What is the model?

$$X_1, \ldots, X_N \overset{i.i.d.}{\sim} M(1, (p_1, p_2, p_3))$$

- What is the null hypothesis, in symbols?

$$H_0 : p_1 = 2p_2$$

- What are the degrees of freedom for this test?

$$1$$

What is the restricted MLE? Your answer is a symbolic expression. It's a vector. Show your work.

$$\begin{aligned}
& \frac{\partial}{\partial p}\left(x_1 \ln(2p) + x_2 \ln p + x_3 \ln(1-3p)\right) \\
= \quad & \frac{x_1}{p} + \frac{x_2}{p} + \frac{x_3}{1-3p}(-3) \stackrel{\text{set}}{=} 0 \\
\Rightarrow \quad & \frac{x_1 + x_2}{p} = \frac{3x_3}{1-3p} \\
\Rightarrow \quad & (x_1 + x_2)(1-3p) = 3px_3 \\
\Rightarrow \quad & x_1 + x_2 = 3p(x_1 + x_2 + x_3) = 3pN \\
\Rightarrow \quad & p = \frac{x_1 + x_2}{3N}
\end{aligned}$$

$$\text{So } \widehat{\mathbf{p}} = \left(\tfrac{2(x_1+x_2)}{3N}, \tfrac{x_1+x_2}{3N}, \tfrac{x_3}{N}\right).$$

- What is the unrestricted MLE? Your answer is a numeric vector: 3 numbers.

$$\left(\frac{106}{200}, \frac{74}{200}, \frac{20}{200}\right) = (0.53, 0.37, 0.10)$$

- What is the restricted MLE? Your answer is a numeric vector: 3 numbers.

$$\left(\frac{2(106+74)}{600}, \frac{106+74}{600}, \frac{20}{200}\right) = (0.6, 0.3, 0.1)$$

- What are the estimated expected frequencies under the null hypothesis? Your answer is a numeric vector: 3 numbers.

`(200*0.6,200*0.3,200*0.10) = (120,60,20)`, because

$$\widehat{\mathbf{m}} = (\widehat{m}_1, \widehat{m}_2, \widehat{m}_3) = (\widehat{Np_1}, \widehat{Np_2}, \widehat{Np_3}) = (N\widehat{p}_1, N\widehat{p}_2, N\widehat{p}_3)$$

Calculate G^2. Show your work.

$$
\begin{aligned}
G^2 &= -2 \ln \frac{\widehat{p}_1^{\,x_1} \widehat{p}_2^{\,x_2} \overline{x}_3^{\,x_3}}{\overline{x}_1^{\,x_1} \overline{x}_2^{\,x_2} \overline{x}_3^{\,x_3}} \\
&= -2 \left(\ln \left[\frac{\widehat{p}_1}{\overline{x}_1} \right]^{x_1} + \ln \left[\frac{\widehat{p}_2}{\overline{x}_2} \right]^{x_2} \right) \\
&= -2 \left(x_1 \ln \frac{\widehat{p}_1}{\overline{x}_1} + x_2 \ln \frac{\widehat{p}_2}{\overline{x}_2} \right) \\
&= -2 \left(106 \ln \frac{0.60}{0.53} + 74 \ln \frac{0.30}{0.37} \right) \\
&= 4.739
\end{aligned}
$$

State your conclusions

- **In symbols**: Reject H_0: $p_1=2p_2$ at alpha = 0.05

- **In words**: More graduates appear to be employed in jobs unrelated to their fields of study than expected.

Statement in words is justified because

Observed	106	74	20
Expected	120	60	20
Obs-Exp	-14	14	0

For a general hypothesis about a multinomial

$$
\begin{aligned}
G^2 &= -2\ln\left(\frac{\max_{\theta\in\Theta_0} L(\theta)}{\max_{\theta\in\Theta} L(\theta)}\right) \\
&= -2\ln\left(\frac{\prod_{j=1}^{k} \widehat{p}_j^{\,x_j}}{\prod_{j=1}^{k} \overline{x}_j^{\,x_j}}\right) \\
&= -2\ln\prod_{j=1}^{k}\left(\frac{\widehat{p}_j}{\overline{x}_j}\right)^{x_j} = 2\sum_{j=1}^{k} -\ln\left(\frac{\widehat{p}_j}{\overline{x}_j}\right)^{x_j} \\
&= 2\sum_{j=1}^{k} x_j \ln\left(\frac{\widehat{p}_j}{\overline{x}_j}\right)^{-1} = 2\sum_{j=1}^{k} x_j \ln\left(\frac{\overline{x}_j}{\widehat{p}_j}\right) \\
&= 2\sum_{j=1}^{k} x_j \ln\left(\frac{x_j}{N\widehat{p}_j}\right) = 2\sum_{j=1}^{k} x_j \ln\left(\frac{x_j}{\widehat{m}_j}\right)
\end{aligned}
$$

$$
\text{Book calls it} \quad G^2 = 2\sum(\text{Observed})\log\left(\frac{\text{Observed}}{\text{Expected}}\right)
$$

Two chi-square formulas

- Likelihood Ratio

$$G^2 = 2\sum(\text{Observed})\log\left(\frac{\text{Observed}}{\text{Expected}}\right)$$

- Pearson

$$X^2 = \sum\frac{(\text{Observed-Expected})^2}{\text{Expected}}$$

- Summation is over all cells
- By expected frequency, we mean estimated expected frequency.
- Asymptotically equivalent
- Same degrees of freedom
- Book's formula for *df* applies only to log-linear models. Use the approach given here, for now.

Pearson Chi-square on the jobs data

Observed	106	74	20
Expected	120	60	20

$$X^2 = \sum \frac{(\text{Observed-Expected})^2}{\text{Expected}}$$

$$= \frac{(106-120)^2}{120} + \frac{(74-60)^2}{60} + 0$$

$$= 4.9 \quad (\text{Compare } G^2 = 4.74)$$

Computing the Pearson chisquare test of independence

- Calculate (estimated) expected frequencies

$$\widehat{m}_{ij} = \frac{x_{i+}x_{+j}}{N}$$

- Calculate $$X^2 = \sum_{\text{Cells}} \frac{(\text{Observed-Expected})^2}{\text{Expected}}$$
- For large samples, has an approximate Chisquare distribution if H_0 is true
- Degrees of freedom (I-1)(J-1)

Numerical example of Pearson chisquare

	White Victim	Black Victim	Total
White Prisoner	151 (105)	9 (55)	160
Black Prisoner	63 (109)	103 (57)	166
Total	214	112	326

$$X^2 = \sum_{\text{Cells}} \frac{(\text{Observed-Expected})^2}{\text{Expected}} = 20.2 \ + \ 38.3 \ + \ 19.4 \ + \ 37.1 \ = \ 115$$

Conclusions

- $X^2 = 115$, df = (2-1) (2-1) = 1
- Critical value at alpha = 0.05 is 3.84
- Reject H_0
- Conclude race of prisoner and race of victim are not independent.
- **That's not good enough!** Murder victims and the persons convicted of murdering them tend to be of the same race. (Say what happened!)

Two treatments for Kidney Stones

	Treatment A	Treatment B
Effective	273	289
Ineffective	77	61

$X^2 = 2.3106$, $df = 1$, $p = 0.1285$

These results are consistent with no difference in effectiveness between treatments.

Single categorical variable, *k* categories

$$\mu = \frac{1}{k}\sum_{j=1}^{k} \log m_j$$

$$\log m_j = \mu + \mu_{(j)} \text{ where } \sum_{j=1}^{k} \mu_{(j)} = 0$$

Linear model for log of expected frequencies

No probability can equal zero!

This is a Re-Parameterization

$$p_j = \frac{m_j}{N} = \frac{1}{N} e^{\log m_j} = \frac{1}{N} e^{\mu + \mu_{(j)}}$$

Substitute into likelihood function and do maximum likelihood

$$L(\mathbf{p}) = p_1^{x_1} \cdots p_k^{x_k} = p_1^{x_1} \cdots p_{k-1}^{x_{k-1}} (1 - \sum_{j=1}^{k-1} p_j)^{N - \sum_{j=1}^{k-1} x_j}$$

How many parameters, *k* or *k-1*?

There are still *k-1* parameters

- $$\mu = \log\left(\frac{N}{\sum_{i=1}^{k} e^{\mu_{(i)}}}\right)$$

- $$p_j = \frac{e^{\mu_{(j)}}}{\sum_{i=1}^{k} e^{\mu_{(i)}}}$$

- All “effects” zero corresponds to equal probabilities

Maximum Likelihood

$$
\begin{aligned}
L(\boldsymbol{\mu}) &= \prod_{j=1}^{k} p_j^{x_j} \\
&= \prod_{j=1}^{k} \left(\frac{e^{\mu_{(j)}}}{\sum_{i=1}^{k} e^{\mu_{(i)}}} \right)^{x_j} \\
&= \frac{\prod_{j=1}^{k} e^{\mu_{(j)} x_j}}{\prod_{j=1}^{k} \left(\sum_{i=1}^{k} e^{\mu_{(i)}} \right)^{x_j}} \\
&= \frac{e^{\sum_{j=1}^{k} \mu_{(j)} x_j}}{\left(\sum_{i=1}^{k} e^{\mu_{(i)}} \right)^{\sum_{j=1}^{k} x_j}} = \frac{e^{\sum_{j=1}^{k} \mu_{(j)} x_j}}{\left(\sum_{i=1}^{k} e^{\mu_{(i)}} \right)^{N}}
\end{aligned}
$$

$$
\ell(\boldsymbol{\mu}) = \sum_{j=1}^{k} \mu_{(j)} x_j - N \log \sum_{i=1}^{k} e^{\mu_{(i)}}
$$

For a table with *I* rows and *J* columns

$$\mu = \frac{1}{IJ}\sum_{i=1}^{I}\sum_{j=1}^{J}\log m_{ij}$$

$$\log m_{ij} = \mu + \mu_{1(i)} + \mu_{2(j)} + \mu_{12(ij)}$$

Compare

$$\mu_{ij} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij}$$

$$\mu_{1(i)}$$

Variable Level

Linear Model for the Log Expected Frequency

$$\log m_{ij} = \mu + \mu_{1(i)} + \mu_{2(j)} + \mu_{12(ij)}$$

$$\mu = \frac{1}{IJ}\sum_{i=1}^{I}\sum_{j=1}^{J}\log m_{ij}$$

$$\mu_{1(i)} = \frac{1}{J}\sum_{j=1}^{J}\log m_{ij} - \mu \qquad \mu_{2(j)} = \frac{1}{I}\sum_{i=1}^{I}\log m_{ij} - \mu$$

Main effects are deviations of marginal mean log expected frequency from the grand mean of the log expected frequencies.

$$\sum_{i=1}^{I}\mu_{1(i)} = \sum_{j=1}^{J}\mu_{2(j)} = 0 \qquad \sum_{i=1}^{I}\mu_{12(ij)} = \sum_{j=1}^{J}\mu_{12(ij)} = 0$$

Interaction terms Represent **Relationship** between Variables

- $\log m_{ij} = \mu + \mu_{1(i)} + \mu_{2(i)} + \mu_{12(ij)}$
- Interaction means the pattern of probabilities for one variable depends on the value of the other variable. This means they are related.
- See how it works for a 2x2 table
- Start with the cross-product ratio alpha (not the same as the significance level, and not a main effect).

The cross-product ratio is an index of relationship

m_{11}	m_{12}
m_{21}	m_{22}

p_{11}	p_{12}
p_{21}	p_{22}

$$\alpha = \frac{m_{11}m_{22}}{m_{12}m_{21}} = \frac{p_{11}p_{22}}{p_{12}p_{21}}$$

$$\alpha = 1 \text{ means no relationship.}$$

alpha=1 means no relationship

a b	a (1-b)	a
(1-a) b	(1-a) (1-b)	1-a
b	1-b	1

$$\alpha = \frac{p_{11}p_{22}}{p_{12}p_{21}} = \frac{ab(1-a)(1-b)}{a(1-b)(1-a)b} = 1$$

Independence =>
alpha=1 <=> interaction = 0

$$\mu_{12(11)} = \frac{1}{4} \log \alpha$$

$$\widehat{\alpha} = \frac{x_{11} x_{22}}{x_{12} x_{21}}$$

Representing the probability of an event by *p*
(Could be conditional)

$$\text{Odds} = \frac{p}{1-p}$$

- If *p*=1/2, odds = .5/(1-.5) = 1 (to 1)
- If *p*=2/3, odds = 2 (to 1)
- If *p*=3/5, odds = (3/5)/(2/5) = 1.5 (to 1)
- If *p*=1/5, odds = .25 (to 1)

Odds Ratio

p_{11}	p_{12}
p_{21}	p_{22}

- Conditional odds of being in Col One given in Row One $= \dfrac{p_{11}/(p_{11}+p_{12})}{1-p_{11}/(p_{11}+p_{12})} = p_{11}/p_{12}$
- Conditional odds of being in Col One given in Row Two = p_{21}/p_{22}
- Ratio of these two quantities is

$$\text{Odds Ratio} = \frac{p_{11}/p_{12}}{p_{21}/p_{22}} = \frac{p_{11}p_{22}}{p_{12}p_{21}} = \alpha$$

	Admitted	Not Admitted
Dept. A	601	332
Dept. B	370	215
Dept. C	322	596
Dept. D	269	523
Dept. E	147	437
Dept. F	46	668

The (estimated) odds of being accepted are

$$\alpha = \frac{(601)(668)}{(332)(46)} = 26.3$$

times as great in Department A, compared to Department F.

Some things to notice

- The cross-product (odds) ratio is meaningful for large tables; apply it to 2x2 sub-tables.
- Re-arrange rows and columns as desired to get the cell you want in the upper left position.
- Combining rows or columns (especially columns) by adding the frequencies is natural and valid.
- If you hear something like “Chances of death before age 50 are four times as great for smokers,” most likely they are talking about an odds ratio.

No relationship means parallel slopes in the log scale

p_{11}	p_{12}
p_{21}	p_{22}

$$\alpha = \frac{p_{11}/p_{12}}{p_{21}/p_{22}} = 1$$

$$\Leftrightarrow$$

$$\log p_{11} - \log p_{12} = \log p_{21} - \log p_{22}$$

Also applies to expected frequencies

The loglin Command

```
> # Playing with how to do it in R -- loglin command
> # Got X2 = 115 by hand
> # help(loglin)
> racetable1 = rbind(c(151,9),
+                    c(63,103))
> try1 = loglin(racetable1,margin=list(1,2)); try1
2 iterations: deviation 0
$lrt
[1] 129.7977

$pearson
[1] 115.0083

$df
[1] 1

$margin
$margin[[1]]
[1] 1

$margin[[2]]
[1] 2
```

```
> # Look at estimated expected frequencies and parameter
> # estimates under H0
> try2 = loglin(racetable1,margin=list(1,2),fit=T,param=T); try2
2 iterations: deviation 0
$lrt
[1] 129.7977

$pearson
[1] 115.0083

$df
[1] 1

$margin
$margin[[1]]
[1] 1

$margin[[2]]
[1] 2
```

```
$fit
         [,1]     [,2]
[1,] 105.0307 54.96933
[2,] 108.9693 57.03067
```

$\widehat{m}_{ij}$

```
$param
$param$`(Intercept)`
[1] 4.348921
```

$\widehat{\mu}$

```
$param$`1`
[1] -0.01840699  0.01840699
```

$\widehat{\mu}_{1(1)}, \widehat{\mu}_{1(2)}$

```
$param$`2`
[1]  0.3237386 -0.3237386
```

$\widehat{\mu}_{2(1)}, \widehat{\mu}_{2(2)}$

```
> # try2$fit are the usual expected frequencies
```

```
> sum(racetable1); sum(try2$fit) # Both = N
[1] 326
[1] 326

> # Remember the LR test formula from the Multinomial lecture?
```

$$G^2 = 2\sum(\text{Observed})\log\left(\frac{\text{Observed}}{\text{Expected}}\right)$$

```
> G2 = 2 * sum(racetable1 * log(racetable1/try2$fit)) ; G2
[1] 129.7977
> try2$lrt
[1] 129.7977
```

A General Rule

- For any 2-dimensional table, maximum likelihood under the null hypothesis of independence yields the same estimated expected frequencies used by the Pearson chisquare test.
- So you can always use $$\widehat{m}_{ij} = \frac{x_{i+}x_{+j}}{N}$$
- And calculate either test statistic with *df = (I-1)(J-1)*

$$G^2 = 2\sum(\text{Observed})\log\left(\frac{\text{Observed}}{\text{Expected}}\right)$$

$$X^2 = \sum\frac{(\text{Observed-Expected})^2}{\text{Expected}}$$

- Trust Pearson statistic more for smaller samples.

```
> # Try a saturated model. Recall last command:
> # try2 = loglin(racetable1,margin=list(1,2),fit=T,param=T)
> try3 = loglin(racetable1,margin=list(c(1,2)),fit=T,param=T)
> try3
2 iterations: deviation 0
$lrt
[1] 0

$pearson
[1] 0

$df
[1] 0

$margin
$margin[[1]]
[1] 1 2
```

```
$fit
     [,1] [,2]
[1,]  151    9
[2,]   63  103

$param
$param$`(Intercept)`
[1] 3.998092

$param$`1`
[1] -0.3908398  0.3908398

$param$`2`
[1]  0.5821152 -0.5821152

$param$`1.2`
           [,1]       [,2]
[1,]  0.8279124 -0.8279124
[2,] -0.8279124  0.8279124
```

$\widehat{m}_{ij} = x_{ij}$

$\widehat{\mu}$

$\widehat{\mu}_{1(1)}, \widehat{\mu}_{1(2)}$

$\widehat{\mu}_{2(1)}, \widehat{\mu}_{2(2)}$

$\widehat{\mu}_{12(ij)}$

$$\log m_{ij} = \mu + \mu_{1(i)} + \mu_{2(i)} + \mu_{12(ij)}$$

```
> log(151)
[1] 5.01728
> 3.998092  -0.3908398 + 0.5821152 + 0.8279124
[1] 5.01728
>
> alpha = (151*103)/(9*63); log(alpha)/4
[1] 0.8279124
>
```

MLEs and parameters obey the same relationships.

Log-linear model for a *k*-dimensional table

- Model for log of expected frequencies
- Looks like model for a *k*-factor ANOVA, with log expected frequency playing the role of the cell mean.
- Main effects represent departure from equal marginal probabilities
- Two-factor interactions represent relationship (association, lack of independence) between variables in two-dimensional marginal tables.
- Three-factor interaction means the nature of the relationship **depends** on the value of the 3d variable.
- Etc.

Log-linear model for a 3-dimensional table

$$\begin{aligned}\log m_{ijk} = \mu \quad &+ \quad \mu_{1(i)} + \mu_{2(j)} + \mu_{3(k)} \\ &+ \quad \mu_{12(ij)} + \mu_{13(ik)} + \mu_{23(jk)} \\ &+ \quad \mu_{123(ijk)}\end{aligned}$$

- μ is the mean of all log expected frequencies.
- Main effects are deviations of the marginal means from the grand mean, etc.
- Effects add to zero over any subscript in parentheses.

We will stick to **hierarchical** models

- If a higher-order term is in the model, all lower-order terms involving those variables must be in the model too.
- Non-hierarchical models are useful at times, but interpretation can be very tricky.

$$\begin{aligned}\log m_{ijk} = \mu \quad &+ \quad \mu_{1(i)} + \mu_{2(j)} + \mu_{3(k)} \\ &+ \quad \mu_{12(ij)} + \mu_{13(ik)} + \mu_{23(jk)} \\ &+ \quad \mu_{123(ijk)}\end{aligned}$$

Florida Prison Data

1. Prisoner's Race (B-W)
2. Victim's Race (B-W)
3. Death Penalty (Y-N)

$$\begin{aligned}\log m_{ijk} = \mu \quad &+ \quad \mu_{1(i)} + \mu_{2(j)} + \mu_{3(k)} \\ &+ \quad \mu_{12(ij)} + \mu_{13(ik)} + \mu_{23(jk)} \\ &+ \quad \mu_{123(ijk)}\end{aligned}$$

Bracket Notation

- Represent variables by numbers, or maybe letters, like VR, PR, DP
- For each variable, enclose vars involving highest order interaction in brackets
- Main effects and lower order interactions are implied, because the models are hierarchical.
- For example, [PR VR] [VR DP] means Prisoner's race and Victim's race are related, and Victim's race and Death penalty are related, but any relationship between Prisoner's race and Death penalty comes from the other 2 relationships. This is a model of *conditional independence*.

[PR VR] [VR DP] = [1 2] [2 3]

1. Prisoner's Race (B-W)
2. Victim's Race (B-W)
3. Death Penalty (Y-N)

$$\log m_{ijk} = \mu + \mu_{1(i)} + \mu_{2(j)} + \mu_{3(k)} + \mu_{12(ij)} + \mu_{23(jk)}$$

Obtain estimated expected frequencies by maximum likelihood, test goodness of fit with X^2 or G^2, approximately chisquare if the model is true.

Table 3-4
Degrees of Freedom Associated with Various Loglinear Models for Three-Dimensional Tables

Model	Abbreviation	# parameters fitted*	d.f.*
$u + u_1 + u_2 + u_3$	[1][2][3]	4 $[1 + (I-1) + (J-1) + (K-1)]$	4 $[IJK - I - J - K + 2]$
$u + u_1 + u_2 + u_3 + u_{12}$	[12][3]	5 $[1 + (I-1) + (J-1) + (K-1) + (I-1)(J-1)]$	3 $[(K-1)(IJ-1)]$
$u + u_1 + u_2 + u_3 + u_{12} + u_{23}$	[12][23]	6 $[1 + (I-1) + (J-1) + (K-1) + (I-1)(J-1) + (J-1)(K-1)]$	2 $[J(I-1)(K-1)]$
$u + u_1 + u_2 + u_3 + u_{12} + u_{23} + u_{13}$	[12][23][13]	7 $[1 + (I-1) + (J-1) + (K-1) + (I-1)(J-1) + (J-1)(K-1) + (I-1)(K-1)]$	1 $[(I-1)(J-1)(K-1)]$
$u + u_1 + u_2 + u_3 + u_{12} + u_{23} + u_{13} + u_{123}$	[123]	8 IJK	0

*The first entry pertains to the $2 \times 2 \times 2$ table. The second entry pertains to the $I \times J \times K$ table.

Conditional independence is Important!

- [1 2] [2 3] means that variables 1 and 2 are related and variables 2 and 3 are related, but any connection between 1 and 3 appears only because they are both related to 2.
- Given (that is, conditionally upon) the value of variable 2, Variables 1 and 3 are independent.
- Controlling for (allowing for) variable 2, there is no relationship between variables 1 and 3.
- Simpson's paradox: Vars 1 and 3 seem to be related but looking at it separately for each level of Var 2, the relationship disappears or even reverses direction.
- Kidney stones: V1 = Treatment, V3=Effectiveness, V2=Size of stones.

Fitting and testing models with the loglin function

- Hierarchical models only
- Very close to bracket notation
- Give it a table and a list of vectors
- Vectors are vars in a bracket, like c(1,2,4) means [1 2 4]
- Iterative proportional model fitting
- Returns estimated expected frequencies as an option

loglin(table,margin,fit=F,param=F)

```
> lizards
, , Species = Sagrei

        Diameter
Height   le 2.5 gt 2.5
  gt 5.0     15     18
  le 5.0     48     84

, , Species = Angusticeps

        Diameter
Height   le 2.5 gt 2.5
  gt 5.0     21      1
  le 5.0      3      2

> lizmodel1 <- loglin(lizards,list(1,c(2,3))) # [1] [23]
2 iterations: deviation 0
```

```
> lizmodel1
$lrt
[1] 43.87073

$pearson
[1] 47.46099

$df
[1] 3

$margin
$margin[[1]]
[1] "Height"

$margin[[2]]
[1] "Diameter" "Species"

> 1-pchisq(43.87073,df=3)
[1] 1.607684e-09
> 1-pchisq(lizmodel1$lrt,df=lizmodel1$df)
[1] 1.607688e-09
```

G^2

X^2

$$\log m = \mu + \mu_1 + \mu_2 + \mu_3 + \mu_{23}$$

$$8 - 5 = 3$$

Some options

```
> lizmodel1b <- loglin(lizards,list('Height',c('Diameter','Species')),
+                      fit=T,param=T)
2 iterations: deviation 0
> lizmodel1b$lrt
[1] 43.87073
>  # Same as before, of course
> lizmodel1b$fit # Estimated expected values

, , Species = Sagrei

        Diameter
Height     le 2.5   gt 2.5
  gt 5.0 18.04688 29.21875
  le 5.0 44.95312 72.78125

, , Species = Angusticeps

        Diameter
Height   le 2.5   gt 2.5
  gt 5.0  6.875 0.859375
  le 5.0 17.125 2.140625
```

Parameter estimates

```
> lizmodel1b$param
$`(Intercept)`
[1] 2.467355

$Height
    gt 5.0     le 5.0
-0.4563239  0.4563239

$Diameter
    le 2.5     gt 2.5
 0.3994009 -0.3994009

$Species
     Sagrei Angusticeps
   1.122860   -1.122860

$Diameter.Species
        Species
Diameter     Sagrei Angusticeps
  le 2.5 -0.6403199   0.6403199
  gt 2.5  0.6403199  -0.6403199
```

μ

$\mu_{1(1)}$ $\mu_{1(2)}$

$\mu_{2(1)}$ $\mu_{2(2)}$

$\mu_{3(1)}$ $\mu_{3(2)}$

$\mu_{23(11)}$ $\mu_{23(12)}$

$\mu_{23(21)}$ $\mu_{23(22)}$

Likelihood Ratio Test for nested models

- Compare “Full” (unrestricted) & “Reduced” (restricted) models.

- Model 1, usually one in which you really believe. This is the full model. If it has all the terms (saturated), it’s equivalent to an unrestricted multinomial model.
- Model 2: A hierarchical log-linear model whose terms are a *subset* of the ones in Model 1. This is the reduced model. It is Model 1, but with some thing(s) missing.

- Test Model 1 versus 2. Model 2 is null, Model 1 is alternative.

Now let Θ_1 be the parameter space under Model 1 and Θ_2 be the parameter space under Model 2: $\Theta_2 \subset \Theta_1 \subset \Theta$.

$$
\begin{aligned}
G^2 &= -2 \ln \left(\frac{\max_{\theta \in \Theta_2} L(\theta)}{\max_{\theta \in \Theta_1} L(\theta)} \right) \\
&= -2 \ln \left(\frac{\max_{\theta \in \Theta_2} L(\theta) / \max_{\theta \in \Theta} L(\theta)}{\max_{\theta \in \Theta_1} L(\theta) / \max_{\theta \in \Theta} L(\theta)} \right) \\
&= G_2^2 - G_1^2 \\
&= 2 \sum (\text{Observed}) \log \left(\frac{\text{Observed}}{\text{Expected}_2} \right) - 2 \sum (\text{Observed}) \log \left(\frac{\text{Observed}}{\text{Expected}_1} \right) \\
&= 2 \sum (\text{Observed}) \log \left(\frac{\text{Expected}_1}{\text{Expected}_2} \right)
\end{aligned}
$$

That's Equation (4.2) in the textbook.

Testing two nested models

- Model 2 is a restricted version of Model 1
- Likelihood ratio test statistic is the difference between the two likelihood ratio tests for goodness of fit: $G^2 = G^2_2 – G^2_1$
- G^2_2 is always bigger because the model is more restricted.
- Asymptotically chisquare, $df = df_2 – df_1$

Florida Prison Data

```
> Prace <- factor(florida$Prace, labels=c('White','Black')) # In order 1,2
> Vrace <- factor(florida$Vrace, labels=c('White','Black'))
> DeathPen <- factor(florida$DeathPen, labels=c('Yes','No'))
> PR_by_DP = table(Prace, DeathPen); PR_by_DP
       DeathPen
Prace   Yes  No
  White  19 141
  Black  17 149

> prop.table(PR_by_DP,1) # Row proportions
       DeathPen
Prace         Yes        No
  White 0.1187500 0.8812500
  Black 0.1024096 0.8975904
> round(100*prop.table(PR_by_DP,1),2) # Row percentages
       DeathPen
Prace     Yes    No
  White 11.88 88.12
  Black 10.24 89.76
```

```
> chisq.test(PR_by_DP,correct=F)

Pearson's Chi-squared test

data:  PR_by_DP
X-squared = 0.2214, df = 1, p-value = 0.638

> dp <- table(Prace, DeathPen, Vrace); dp
, , Vrace = White

       DeathPen
Prace   Yes  No
  White  19 132
  Black  11  52

, , Vrace = Black

       DeathPen
Prace   Yes  No
  White   0   9
  Black   6  97
```

Something interesting may be going on

```
> # Row percents
> round(100*prop.table(dp[,,1],1),2)
        DeathPen
Prace       Yes    No
  White   12.58 87.42
  Black   17.46 82.54
> round(100*prop.table(dp[,,2],1),2)
        DeathPen
Prace      Yes     No
  White   0.00 100.00
  Black   5.83  94.17
```

Prace and Deathpen CONTROLLING for (conditional upon) Vrace

Chisquare tests on sub-tables

```
> # Pearson
> chisq.test(dp[,,1],correct=F)

Pearson's Chi-squared test

data:  dp[, , 1]
X-squared = 0.8774, df = 1, p-value = 0.3489

> chisq.test(dp[,,2],correct=F)

Pearson's Chi-squared test

data:  dp[, , 2]
X-squared = 0.5539, df = 1, p-value = 0.4567

Warning message:
Chi-squared approximation may be incorrect in:
chisq.test(dp[, , 2], correct = F)
```

What's the problem? Look at expected frequencies.

```
> loglin(dp[,,2],margin=list(1,2),fit=T)$fit
2 iterations: deviation 1.421085e-14
       DeathPen
Prace          Yes        No
  White 0.4821429  8.517857
  Black 5.5178571 97.482143
```

Low expected frequencies tend to inflate chisquare.

No problem here.

Complete Independence

```
> ind <- loglin(dp,list(1,2,3)); ind
2 iterations: deviation 2.842171e-14
$lrt
[1] 137.9294

$pearson
[1] 122.3975

$df
[1] 4

$margin
$margin[[1]]
[1] "Prace"

$margin[[2]]
[1] "DeathPen"

$margin[[3]]
[1] "Vrace"
```

Model with all 2-factor relationships

```
> twoways <- loglin(dp,list(c(1,2),c(1,3),c(2,3))); twoways
5 iterations: deviation 0.05215771
$lrt
[1] 0.7007595

$pearson
[1] 0.3750283

$df
[1] 1

$margin
$margin[[1]]
[1] "Prace"    "DeathPen"

$margin[[2]]
[1] "Prace" "Vrace"

$margin[[3]]
[1] "DeathPen" "Vrace"
```

How is G^2 being calculated?!

```
, , Vrace = White

       DeathPen
Prace   Yes  No
  White  19 132
  Black  11  52

, , Vrace = Black

       DeathPen
Prace   Yes  No
  White   0   9
  Black   6  97
```

$$G^2 = 2\sum(\text{Observed})\log\left(\frac{\text{Observed}}{\text{Expected}}\right)$$

Zero cell is being dropped

- Conservative, for a test of fit. Chisquare is smaller, so it's less likely to force you to a more complicated model.
- Add a small constant to the observed frequency of zero, just for computing G^2, not for computing the expected frequencies. How small? The smaller the better.

$$\lim_{x \to 0} \left(x \log \frac{x}{\text{Expected}} \right) = 0$$

- No effect on LR tests of nested models.

$$G^2_{1,2} = 2 \sum (\text{Observed}) \log \left(\frac{\text{Expected}_1}{\text{Expected}_2} \right)$$

Look at 2-factor marginal tables

- Prisoner's race by death penalty: Consistent with no relationship.
- Prisoner's race by victim's race: Strong, we think.
- Victim's race by death penalty: Need to check it.

Prisoner’s Race and Victim's Race

```
> PR_by_VR = table(Prace, Vrace); PR_by_VR
       Vrace
Prace   White Black
  White   151     9
  Black    63   103
> round(100*prop.table(PR_by_VR,1),2) # Row percentages
       Vrace
Prace   White Black
  White 94.38  5.62
  Black 37.95 62.05
> chisq.test(PR_by_VR,correct=F)

Pearson’s Chi-squared test

data:  PR_by_VR
X-squared = 115.0083, df = 1, p-value < 2.2e-16
```

People tend to be in jail for killing someone of their own race.
Anything else interesting?

Victim's Race and Death Penalty

```
> VR_by_DP = table(Vrace, DeathPen); VR_by_DP
       DeathPen
Vrace   Yes  No
  White  30 184
  Black   6 106
> round(100*prop.table(VR_by_DP,1),2) # Row percentages
       DeathPen
Vrace     Yes    No
  White 14.02 85.98
  Black  5.36 94.64
> chisq.test(VR_by_DP,correct=F)

Pearson's Chi-squared test

data:  VR_by_DP
X-squared = 5.6149, df = 1, p-value = 0.01781
```

Suggests death penalty more likely if victim is White

It look like we want to add [PR, VR], but marginal tables can be misleading – See Section 3.8. Choose model with smallest G^2 (best fit)

```
> # 1=Prace, 2=DeathPen, 3=Vrace)
> loglin(dp,list(2,c(1,3)))$lrt # [DP] [PR, VR]
2 iterations: deviation 0
[1] 8.131611
> loglin(dp,list(1,c(2,3)))$lrt # [PR] [VR, DP]
2 iterations: deviation 0
[1] 131.6796
> loglin(dp,list(3,c(1,2)))$lrt # [VR] [PR, DP]
2 iterations: deviation 0
[1] 137.7079
```

[DP] [PR, VR] is the best choice, by far

- Is it an improvement?
- Does it fit?

```
> ModelA = ind
> ModelB <- loglin(dp,list(2,c(1,3)))
2 iterations: deviation 0
> #   Is it an improvement?
> G2Change = ModelA$lrt-ModelB$lrt; G2Change
[1] 129.7977
> dfChange = ModelA$df-ModelB$df; dfChange
[1] 1
> pvalChange = 1-pchisq(G2Change, df=dfChange)
> pvalChange
[1] 0
```

Does it fit?

```
> #   Does it fit?
> G2B = ModelB$lrt; G2B
[1] 8.131611
> dfB = ModelB$df; dfB
[1] 3
> pvalB = 1-pchisq(G2B, df=dfB); pvalB
[1] 0.04336859
> ModelB$pearson; 1-pchisq(ModelB$pearson,df=ModelB$df)
[1] 6.977343
[1] 0.07262343
```

I say we proceed, but there could be argument.

Add another association
Either [PR,VR][PR,DP] or [PR,VR][VR,DP]

```
> # 1=Prace, 2=DeathPen, 3=Vrace
> loglin(dp,list(c(1,3),c(1,2)))$lrt # [PR,VR][PR,DP]
2 iterations: deviation 0
[1] 7.91016
> loglin(dp,list(c(1,3),c(2,3)))$lrt # [PR,VR][VR,DP]
2 iterations: deviation 1.421085e-14
[1] 1.881895
```

Choose [PR,VR][VR,DP]

```
> ModelC <- loglin(dp,list(c(1,3),c(2,3)))
2 iterations: deviation 1.421085e-14
> #   Is it an improvement?
> G2Change = ModelB$lrt-ModelC$lrt; G2Change
[1] 6.249715
> dfChange = ModelB$df-ModelC$df; dfChange
[1] 1
> pvalChange = 1-pchisq(G2Change, df=dfChange)
> pvalChange
[1] 0.01242133
> #   Does it fit?
> G2C = ModelC$lrt; G2C
[1] 1.881895
> dfC = ModelC$df; dfC
[1] 2
> pvalC = 1-pchisq(G2C, df=dfC); pvalC
[1] 0.3902578
```

Does it help to add [PR,DP]?

```
> ModelD <- twoways
> G2Change = ModelC$lrt-ModelD$lrt; G2Change
[1] 1.181136
> dfChange = ModelC$df-ModelD$df; dfChange
[1] 1
> pvalChange = 1-pchisq(G2Change, df=dfChange)
> pvalChange
[1] 0.2771249
```

Hierarchy: Not planned in advance

	Fit			Change		
Model	Chisq	df	p	Chisq	df	p
[VR] [PR] [DP]	137.93	4	0.00			
[DP] [VR,PR]	8.13	3	0.04	129.80	1	0.00
[VR,PR] [VR,DP]	1.88	2	0.39	6.25	1	0.01
[VR,PR] [VR,DP] [PR,DP]	0.70	1	0.40	1.18	1	0.28

Model is [VR,PR] [VR,DP]

- Hierarchy of models was the result of exploring the data
- Kind of forward stepwise method, could be automated
- Guided by hypothesis tests, but please don't take them completely at face value. We did quite a few tests, and the theory applies to single tests performed in isolation.

Describe the findings in words

- Prisoners in jail for murder in Florida tended to be convicted of killing people of the same race.
- The death penalty was less likely when the victim was Black.

(These conclusions are based on looking at the marginal 2-way tables. Let's check the parameter estimates too.)

Checking the parameter estimates

Just part of the output

```
> loglin(dp,list(c(1,3),c(2,3)),param=T)$param
$Prace.Vrace
       Vrace
Prace        White      Black
  White  0.8279124 -0.8279124
  Black -0.8279124  0.8279124

$DeathPen.Vrace
        Vrace
DeathPen      White      Black
     Yes  0.2644853 -0.2644853
     No  -0.2644853  0.2644853
```

- Prace.Vrace interaction says increased chance of White-White and Black-Black
- DeathPen.Vrace interaction says increased chance of Yes-White and No-Black

A little more about the interpretation of [VR,PR] [VR,DP]

- It's a model of conditional independence
- Allowing (controlling) for Victim's Race, Prisoner's Race is unrelated to Death Penalty
- Model says that in each sub-table (VR=Black, VR=White), Prisoner's Race is independent of Death Penalty.
- So the test of model fit should be like a combined test of independence for both 2-way tables.

$$H_0 : \mu_{12} = \mu_{123} = 0$$

Had $G^2 = 1.88$, $df=2$, $p = 0.39$

$$H_0 : \mu_{12} = \mu_{123} = 0$$

```
> dp
, , Vrace = White

       DeathPen
Prace    Yes  No
  White   19 132
  Black   11  52

, , Vrace = Black

       DeathPen
Prace    Yes  No
  White    0   9
  Black    6  97
```

```
> a = loglin(dp[,,1],margin=list(1,2))$lrt; a
2 iterations: deviation 0
[1] 0.847478
> b = loglin(dp[,,2],margin=list(1,2))$lrt; b
2 iterations: deviation 1.421085e-14
[1] 1.034417
> a+b
[1] 1.881895
```

Control by sub-division: Very natural.

Works for Pearson X^2 too.

The lesson

- Want to examine the relationship between *A* and *B*, but *A* might be related to *C* and *B* might be related to *C*.
- So look at the relationship between *A* and *B* controlling for *C*.
- Examine (test) A by B separately for each level of C: Sub-division.
- Pool (combine) the tests by adding chi-squares and adding degrees of freedom.
- *Identical* to the chi-square test for fit of a log-linear model of conditional independence!

Marginal Tables with *R*

- Data frame: Use **xtabs**
  - UCB <- xtabs(Freq ~ Dept + Gender + Admit, data = berkeley)
  - GenderAdmit <- xtabs(Freq ~ Gender + Admit, data = berkeley)
  - xtabs(Freq ~ Dept + Admit + Gender, data = berkeley)
- Factors: Use **table**
  - deathrow <- table(Prace, DeathPen, Vrace)
  - PR_by_DP = table(Prace, DeathPen)
  - table(Vrace, DeathPen, Prace)
- Data already in a table: Use **margin.table**

margin.table

```
> lizards
, , Species = Sagrei

        Diameter
Height   le 2.5 gt 2.5
  gt 5.0     15     18
  le 5.0     48     84

, , Species = Angusticeps

        Diameter
Height   le 2.5 gt 2.5
  gt 5.0     21      1
  le 5.0      3      2

> species_by_height = margin.table(lizards,margin=c(3,1))
> species_by_height
             Height
Species       gt 5.0 le 5.0
  Sagrei          33    132
  Angusticeps     22      5
> # spec_by_height_by_diam = margin.table(lizards,margin=c(3,1,2))
```

The Berkeley Graduate Admissions Data

```
> UCB
, , Admit = Admitted

    Gender
Dept Female Male
   A     89  512
   B     17  353
   C    202  120
   D    131  138
   E     94   53
   F     24   22

, , Admit = Rejected

    Gender
Dept Female Male
   A     19  313
   B      8  207
   C    391  205
   D    244  279
   E    299  138
   F    317  351

> is.table(UCB) # T
[1] TRUE
> summary(UCB)  # X2 for complete independence = 2000.3, df=16
Call: xtabs(formula = Freq ~ Dept + Gender + Admit, data = berkeley)
Number of cases in table: 4526
Number of factors: 3
Test for independence of all factors:
 Chisq = 2000.3, df = 16, p-value = 0
```

```
> all2ways <- loglin(UCB,margin=list(c(1,2),c(1,3),c(2,3))); all2ways
7 iterations: deviation 0.04308377
$lrt
[1] 20.20428

$pearson
[1] 18.82298

$df
[1] 5

$margin
$margin[[1]]
[1] "Dept"   "Gender"

$margin[[2]]
[1] "Dept"  "Admit"

$margin[[3]]
[1] "Gender" "Admit"


> 1-pchisq(all2ways$lrt,df=all2ways$df) # p-value for H0: mu123=0
[1] 0.001144076
> # So the relationship between gender and admission DEPENDS on department
>
```

Let's look at some 2-dimensional marginal tables

```
> sex_by_admit = xtabs(Freq ~ Gender + Admit, data = berkeley)
> sex_by_admit
        Admit
Gender   Admitted Rejected
  Female      557     1278
  Male       1198     1493
> round(100*prop.table(sex_by_admit,1),2) # Row percentages
        Admit
Gender   Admitted Rejected
  Female    30.35    69.65
  Male      44.52    55.48
> summary(sex_by_admit)
Call: xtabs(formula = Freq ~ Gender + Admit, data = berkeley)
Number of cases in table: 4526
Number of factors: 2
Test for independence of all factors:
 Chisq = 92.21, df = 1, p-value = 7.814e-22

>
> sex_by_dept = xtabs(Freq ~ Gender + Dept, data = berkeley)
> sex_by_dept
        Dept
Gender     A   B   C   D   E   F
  Female 108  25 593 375 393 341
  Male   825 560 325 417 191 373
> round(100*prop.table(sex_by_dept,1),2) # Row percentages
        Dept
Gender       A     B     C     D     E     F
  Female  5.89  1.36 32.32 20.44 21.42 18.58
  Male   30.66 20.81 12.08 15.50  7.10 13.86
> summary(sex_by_dept)
Call: xtabs(formula = Freq ~ Gender + Dept, data = berkeley)
Number of cases in table: 4526
Number of factors: 2
Test for independence of all factors:
 Chisq = 1068.4, df = 5, p-value = 9.444e-229
>
> dept_by_admit = xtabs(Freq ~ Dept + Admit, data = berkeley)
> dept_by_admit
    Admit
Dept Admitted Rejected
   A      601      332
   B      370      215
   C      322      596
   D      269      523
   E      147      437
   F       46      668
```

```
> round(100*prop.table(dept_by_admit,1),2) # Row percentages
    Admit
Dept Admitted Rejected
   A    64.42    35.58
   B    63.25    36.75
   C    35.08    64.92
   D    33.96    66.04
   E    25.17    74.83
   F     6.44    93.56
> summary(dept_by_admit)
Call: xtabs(formula = Freq ~ Dept + Admit, data = berkeley)
Number of cases in table: 4526
Number of factors: 2
Test for independence of all factors:
 Chisq = 778.9, df = 5, p-value = 4.23e-166
>
> # What is going on here? Assemble a good table.
> admitper <- round(100*prop.table(dept_by_admit,1),2)
> genderper <- round(100*prop.table(sex_by_dept,1),2)
> cbind(admitper[,1],t(genderper))
        Female  Male
A 64.42   5.89 30.66
B 63.25   1.36 20.81
C 35.08  32.32 12.08
D 33.96  20.44 15.50
E 25.17  21.42  7.10
F  6.44  18.58 13.86
>
```

```
> # Look at gender by admit controlling for department
> ucb <- xtabs(Freq ~ Gender + Admit + Dept, data = berkeley)
> # That's 6 2x2 tables -- hard to look at
> dept <- dimnames(ucb)$Dept; dept
[1] "A" "B" "C" "D" "E" "F"
> totalgsq <- 0
> for(k in 1:6)
+     {
+     cat("\n", "   Department ",dept[k],"\n")
+     cat("    ------------- \n\n")
+     freq <- ucb[,,k]
+     rowper <- round(100*prop.table(freq,1),2)
+     llm <- loglin(freq,margin=list(1,2),print=F) # Don't print iterations
+     g2 <- llm$lrt; df = llm$df; pval = 1-pchisq(g2,df)
+     cat("   Observed Frequencies \n\n")
+     print(freq)
+     cat("\n    Row Percentages \n\n")
+     print(rowper)
+     cat("\n G-squared = ",g2,", df = ",df,", p = ",pval,"\n")
+     totalgsq = totalgsq + g2
+     }

    Department  A
    -------------

   Observed Frequencies

        Admit
Gender   Admitted Rejected
  Female       89       19
  Male        512      313

    Row Percentages

        Admit
Gender   Admitted Rejected
  Female    82.41    17.59
  Male      62.06    37.94

 G-squared =  19.05401 , df =  1 , p =  1.270705e-05
```

Department B

Observed Frequencies

Gender	Admit: Admitted	Admit: Rejected
Female	17	8
Male	353	207

Row Percentages

Gender	Admit: Admitted	Admit: Rejected
Female	68.00	32.00
Male	63.04	36.96

G-squared = 0.2586429 , df = 1 , p = 0.611054

Department C

Observed Frequencies

Gender	Admit: Admitted	Admit: Rejected
Female	202	391
Male	120	205

Row Percentages

Gender	Admit: Admitted	Admit: Rejected
Female	34.06	65.94
Male	36.92	63.08

G-squared = 0.7509844 , df = 1 , p = 0.3861648

Department D

Observed Frequencies

Gender	Admit: Admitted	Admit: Rejected
Female	131	244
Male	138	279

Row Percentages

Gender	Admit: Admitted	Admit: Rejected
Female	34.93	65.07
Male	33.09	66.91

G-squared = 0.2978665 , df = 1 , p = 0.585223

Department E

Observed Frequencies

Gender	Admit: Admitted	Admit: Rejected
Female	94	299
Male	53	138

Row Percentages

Gender	Admit: Admitted	Admit: Rejected
Female	23.92	76.08
Male	27.75	72.25

G-squared = 0.9903864 , df = 1 , p = 0.3196480

```
    Department  F
    -------------

   Observed Frequencies

        Admit
Gender   Admitted Rejected
  Female       24      317
  Male         22      351

    Row Percentages

        Admit
Gender   Admitted Rejected
  Female     7.04    92.96
  Male       5.90    94.10

 G-squared =  0.3836167 , df =  1 , p =  0.535674
>
> # Model of conditional independence should not fit, with
> # G-squared = totalgsq
> loglin(ucb,margin=list(c("Gender","Dept"),c("Dept","Admit")))$lrt
2 iterations: deviation 5.684342e-14
[1] 21.73551
> totalgsq
[1] 21.73551
> 1-pchisq(totalgsq,6)
[1] 0.001351993
```

Detergent Data (Table 5-1)

```
> # Navigate to the location of the data using the Misc menu
> soapdata <- read.table("DetergentFrame.txt"); soapdata
   Softness Prev_Use   Temp Pref Freq
1    1=Soft    1=Yes 1=High  1=X   19
2    1=Soft    1=Yes 1=High  2=M   29
3    1=Soft    1=Yes  2=Low  1=X   57
4    1=Soft    1=Yes  2=Low  2=M   49
5    1=Soft     2=No 1=High  1=X   29
6    1=Soft     2=No 1=High  2=M   27
7    1=Soft     2=No  2=Low  1=X   63
8    1=Soft     2=No  2=Low  2=M   53
9    2=Medm    1=Yes 1=High  1=X   23
10   2=Medm    1=Yes 1=High  2=M   47
11   2=Medm    1=Yes  2=Low  1=X   47
12   2=Medm    1=Yes  2=Low  2=M   55
13   2=Medm     2=No 1=High  1=X   33
14   2=Medm     2=No 1=High  2=M   23
15   2=Medm     2=No  2=Low  1=X   66
16   2=Medm     2=No  2=Low  2=M   50
17   3=Hard    1=Yes 1=High  1=X   24
18   3=Hard    1=Yes 1=High  2=M   43
19   3=Hard    1=Yes  2=Low  1=X   37
20   3=Hard    1=Yes  2=Low  2=M   52
21   3=Hard     2=No 1=High  1=X   42
22   3=Hard     2=No 1=High  2=M   30
23   3=Hard     2=No  2=Low  1=X   68
24   3=Hard     2=No  2=Low  2=M   42
> soap <- xtabs(Freq ~ Softness+Prev_Use+Temp+Pref, data=soapdata)
> summary(soap)
Call: xtabs(formula = Freq ~ Softness + Prev_Use + Temp + Pref, data = soapdata)
Number of cases in table: 1008
Number of factors: 4
Test for independence of all factors:
      Chisq = 43.9, df = 18, p-value = 0.0005957
> loglin(soap,list(1,2,3,4))$lrt # Matches text, p. 76
2 iterations: deviation 1.136868e-13
[1] 42.92866
```

```
> # Strategy: Find a model for the explanatory variables, using a
> # marginal table. Then check links of explanatory to response.
> soapex = xtabs(Freq ~ Softness+Prev_Use+Temp, data=soapdata); soapex
, , Temp = 1=High

        Prev_Use
Softness 1=Yes 2=No
  1=Soft    48   56
  2=Medm    70   56
  3=Hard    67   72

, , Temp = 2=Low

        Prev_Use
Softness 1=Yes 2=No
  1=Soft   106  116
  2=Medm   102  116
  3=Hard    89  110

> summary(soapex)
Call: xtabs(formula = Freq ~ Softness + Prev_Use + Temp, data = soapdata)
Number of cases in table: 1008
Number of factors: 3
Test for independence of all factors:
      Chisq = 10.019, df = 7, p-value = 0.1875
> soapexA = loglin(soapex,list(1,2,3)) # Complete independence
2 iterations: deviation 1.136868e-13
> soapexA$lrt
[1] 10.10304
>
> # Check 2-d marginal tables anyway
> softemp = xtabs(Freq ~ Softness+Temp, data=soapdata); softemp
        Temp
Softness 1=High 2=Low
  1=Soft    104   222
  2=Medm    126   218
  3=Hard    139   199
> round(100*prop.table(softemp,1),2) # Row percents
        Temp
Softness 1=High 2=Low
  1=Soft  31.90 68.10
  2=Medm  36.63 63.37
  3=Hard  41.12 58.88
> summary(softemp)
Call: xtabs(formula = Freq ~ Softness + Temp, data = soapdata)
Number of cases in table: 1008
Number of factors: 2
Test for independence of all factors:
      Chisq = 6.082, df = 2, p-value = 0.04778
> # Harder water goes with higher temp, sort of
```

```
> softprev = xtabs(Freq ~ Softness+Prev_Use, data=soapdata); softprev
        Prev_Use
Softness 1=Yes 2=No
  1=Soft   154  172
  2=Medm   172  172
  3=Hard   156  182
> round(100*prop.table(softprev,1),2) # Row percents
        Prev_Use
Softness 1=Yes  2=No
  1=Soft 47.24 52.76
  2=Medm 50.00 50.00
  3=Hard 46.15 53.85
> summary(softprev)
Call: xtabs(formula = Freq ~ Softness + Prev_Use, data = soapdata)
Number of cases in table: 1008
Number of factors: 2
Test for independence of all factors:
      Chisq = 1.0753, df = 2, p-value = 0.5841
> # Not much

> prevtemp = xtabs(Freq ~ Prev_Use+Temp, data=soapdata); prevtemp
        Temp
Prev_Use 1=High 2=Low
   1=Yes    185   297
   2=No     184   342
> summary(prevtemp)
Call: xtabs(formula = Freq ~ Prev_Use + Temp, data = soapdata)
Number of cases in table: 1008
Number of factors: 2
Test for independence of all factors:
      Chisq = 1.2535, df = 1, p-value = 0.2629
> # Not much
>
> JustSoftemp = loglin(soapex,list(2,c(1,3)))
2 iterations: deviation 0
> JustSoftemp$lrt; JustSoftemp$df
[1] 4.003931
[1] 5
> 1-pchisq(JustSoftemp$lrt, JustSoftemp$df)
[1] 0.5488501
> # Fits fine. Any better than complete independence?
> G2Change = soapexA$lrt-JustSoftemp$lrt; G2Change
[1] 6.099104
> dfChange = soapexA$df-JustSoftemp$df; dfChange
[1] 2
> pvalChange = 1-pchisq(G2Change, df=dfChange)
> pvalChange
[1] 0.04738014
> # Okay, keep [Softness Temperature]
>
```

```
> # Any IV, DV link at all?
> ModelA = loglin(soap,list(2,4,c(1,3))); ModelA
2 iterations: deviation 5.684342e-14
$lrt
[1] 36.82955

$pearson
[1] 37.76417

$df
[1] 16

$margin
$margin[[1]]
[1] "Prev_Use"

$margin[[2]]
[1] "Pref"

$margin[[3]]
[1] "Softness" "Temp"


> 1-pchisq(ModelA$lrt,ModelA$df)
[1] 0.002216038
> # Something is going on. Try model with all 2-way links
> # between explanatory and response variables.
> link2 = loglin(soap,list(c(1,3),c(1,4),c(2,4),c(3,4))); link2
3 iterations: deviation 0.06630545
$lrt
[1] 11.54287

$pearson
[1] 11.45839

$df
[1] 12

$margin
$margin[[1]]
[1] "Softness" "Temp"

$margin[[2]]
[1] "Softness" "Pref"

$margin[[3]]
[1] "Prev_Use" "Pref"

$margin[[4]]
[1] "Temp" "Pref"


> # Fits well. Try adding each link separately, and compare
```

```
> loglin(soap,list(2,c(1,3),c(1,4)))$lrt
2 iterations: deviation 1.136868e-13
[1] 36.43426
> loglin(soap,list(c(1,3),c(2,4)))$lrt
2 iterations: deviation 5.684342e-14
[1] 16.24809
> loglin(soap,list(2,c(1,3),c(3,4)))$lrt
2 iterations: deviation 5.684342e-14
[1] 32.46795

> ModelB = loglin(soap,list(c(1,3),c(2,4))) # [Soft Temp] [PrevUse Pref]
2 iterations: deviation 5.684342e-14
> # Does it fit?
> ModelB$lrt; ModelB$df
[1] 16.24809
[1] 15
> 1-pchisq(ModelB$lrt, ModelB$df)
[1] 0.365758
> # Improvement?
> G2Change = ModelA$lrt-ModelB$lrt; G2Change
[1] 20.58147
> dfChange = ModelA$df-ModelB$df; dfChange
[1] 1
> pvalChange = 1-pchisq(G2Change, df=dfChange); pvalChange
[1] 5.71467e-06
> # I like this one. But just check to see if another link is justified.
>
> loglin(soap,list(c(1,3),c(2,4),c(1,4)))$lrt # Add [Soft Pref]?
2 iterations: deviation 2.842171e-14
[1] 15.85279
> loglin(soap,list(c(1,3),c(2,4),c(3,4)))$lrt # Add [Temp Pref]?
2 iterations: deviation 5.684342e-14
[1] 11.88649
> ModelC = loglin(soap,list(c(1,3),c(2,4),c(3,4))) # Adding [Temp Pref]
2 iterations: deviation 5.684342e-14
> G2Change = ModelB$lrt-ModelC$lrt; G2Change
[1] 4.361601
> dfChange = ModelB$df-ModelC$df; dfChange
[1] 1
> pvalChange = 1-pchisq(G2Change, df=dfChange); pvalChange
[1] 0.03675775
> # I have to take it. Is link2 an improvement over this?
>
> ModelD = link2
> G2Change = ModelC$lrt-ModelD$lrt; G2Change
[1] 0.3436218
> dfChange = ModelC$df-ModelD$df; dfChange
[1] 2
> pvalChange = 1-pchisq(G2Change, df=dfChange); pvalChange
[1] 0.8421384
> # Okay, Model C looks like the choice.
> # [1 3] [2 4] [3 4] = [Soft Temp] [PrevUse Pref] [Temp Pref]
```

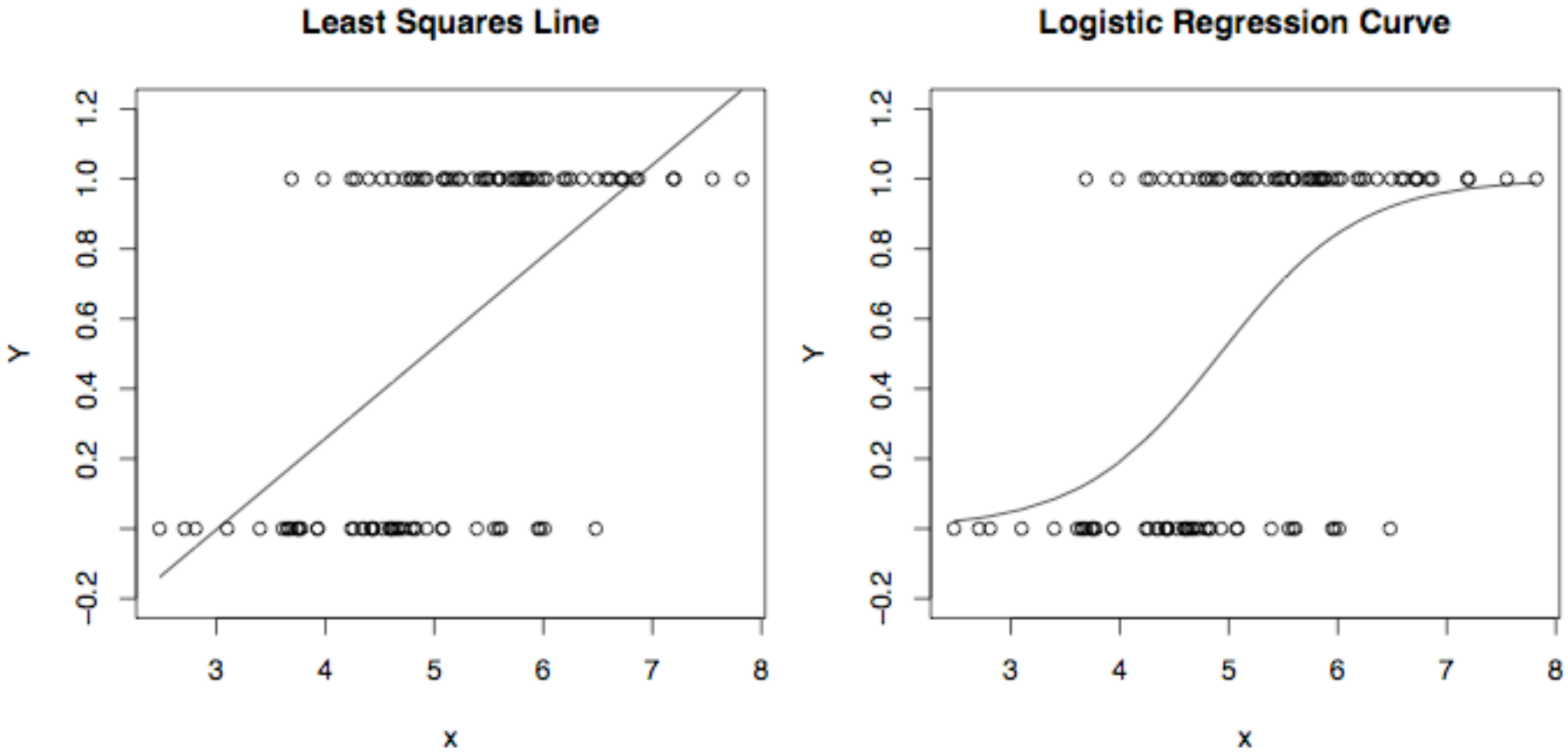
```
>
> # Look at marginal tables and parameter estimates to see what's happening
> PrevusePref = xtabs(Freq ~ Prev_Use+Pref, data=soapdata); PrevusePref
        Pref
Prev_Use 1=X 2=M
   1=Yes 207 275
   2=No  301 225
> round(100*prop.table(PrevusePref,1),2) # Row percents
        Pref
Prev_Use   1=X   2=M
   1=Yes 42.95 57.05
   2=No  57.22 42.78
> summary(PrevusePref)
Call: xtabs(formula = Freq ~ Prev_Use + Pref, data = soapdata)
Number of cases in table: 1008
Number of factors: 2
Test for independence of all factors:
      Chisq = 20.512, df = 1, p-value = 5.925e-06
> # Those who used M before tend to prefer it
> TempPref = xtabs(Freq ~ Temp+Pref, data=soapdata); TempPref
        Pref
Temp     1=X 2=M
  1=High 170 199
  2=Low  338 301
> round(100*prop.table(TempPref,1),2) # Row percents
        Pref
Temp       1=X   2=M
  1=High 46.07 53.93
  2=Low  52.90 47.10
> summary(TempPref)
Call: xtabs(formula = Freq ~ Temp + Pref, data = soapdata)
Number of cases in table: 1008
Number of factors: 2
Test for independence of all factors:
      Chisq = 4.358, df = 1, p-value = 0.03683
> # High temp goes with pref for M
```

```
> # Parameter estimates for Model C
> loglin(soap,list(c(1,3),c(2,4),c(3,4)),param=T)$param
2 iterations: deviation 5.684342e-14

$Softness.Temp
        Temp
Softness       1=High        2=Low
  1=Soft -0.101588153  0.101588153
  2=Medm  0.003448510 -0.003448510
  3=Hard  0.098139643 -0.098139643

$Prev_Use.Pref
        Pref
Prev_Use        1=X        2=M
   1=Yes -0.1437655  0.1437655
   2=No   0.1437655 -0.1437655

$Temp.Pref
        Pref
Temp            1=X        2=M
  1=High -0.0683605  0.0683605
  2=Low   0.0683605 -0.0683605

> #
> # Conclusions
> #
> #   Consumers with harder water tend to use higher temperature
> #   Those who used Brand M before tend to prefer it
> #   Use of High temperature water goes with preference for M
> #
> #   Book arrives at the same model
> #   But if the conclusions are actually stated in the book, I missed it.
```

Logistic Regression

For a binary dependent variable:
1=Yes, 0=No

Least Squares vs. Logistic Regression



Linear regression model for the log odds of the event Y=1

$$\ln\left(\frac{P(Y=1|\mathbf{X}=\mathbf{x})}{P(Y=0|\mathbf{X}=\mathbf{x})}\right) = \beta_0 + \beta_1 x_1 + \ldots + \beta_{p-1} x_{p-1}$$

Equivalent Statements

$$\ln\left(\frac{P(Y=1|\mathbf{X}=\mathbf{x})}{P(Y=0|\mathbf{X}=\mathbf{x})}\right) = \beta_0 + \beta_1 x_1 + \ldots + \beta_{p-1} x_{p-1}$$

$$\begin{aligned}\frac{P(Y=1|\mathbf{X}=\mathbf{x})}{P(Y=0|\mathbf{X}=\mathbf{x})} &= e^{\beta_0+\beta_1 x_1+\ldots+\beta_{p-1}x_{p-1}} \\ &= e^{\beta_0} e^{\beta_1 x_1} \ldots e^{\beta_{p-1}x_{p-1}}\end{aligned}$$

$$P(Y=1|x_1,\ldots,x_{p-1}) = \frac{e^{\beta_0+\beta_1 x_1+\ldots+\beta_{p-1}x_{p-1}}}{1+e^{\beta_0+\beta_1 x_1+\ldots+\beta_{p-1}x_{p-1}}}$$

$F(x) = \frac{e^x}{1+e^x}$ is called the *logistic distribution.*

- Could use any cumulative distribution function:

$$P(Y = 1|x_1, \ldots, x_{p-1}) = F(\beta_0 + \beta_1 x_1 + \ldots + \beta_{p-1} x_{p-1})$$

- CDF of the standard normal used to be popular
- Called probit analysis
- Can be closely approximated with a logistic regression.

In terms of log odds, logistic regression is like regular regression

$$\ln\left(\frac{P(Y=1|\mathbf{X}=\mathbf{x})}{P(Y=0|\mathbf{X}=\mathbf{x})}\right) = \beta_0 + \beta_1 x_1 + \ldots + \beta_{p-1} x_{p-1}$$

In terms of plain odds,

- Logistic regression coefficients represent *odds ratios*
- For example, “Among 50 year old men, the odds of being dead before age 60 are three times as great for smokers.”

$$\frac{\text{Odds of death given smoker}}{\text{Odds of death given nonsmoker}} = 3$$

Logistic regression

- X=1 means smoker, X=0 means non-smoker
- Y=1 means dead, Y=0 means alive

- Log odds of death = $\beta_0 + \beta_1 x$

- Odds of death = $e^{\beta_0} e^{\beta_1 x}$

$$\text{Odds of Death} = e^{\beta_0} e^{\beta_1 x}$$

Group	x	**Odds of Death**
Smokers	1	$e^{\beta_0} e^{\beta_1}$
Non-smokers	0	e^{β_0}

$$\frac{\text{Odds of death given smoker}}{\text{Odds of death given nonsmoker}} = \frac{e^{\beta_0} e^{\beta_1}}{e^{\beta_0}} = e^{\beta_1}$$

Cancer Therapy Example

$$\text{Log Survival Odds} = \beta_0 + \beta_1 d_1 + \beta_2 d_2 + \beta_3 x$$

Treatment	d_1	d_2	**Odds of Survival** $= e^{\beta_0} e^{\beta_1 d_1} e^{\beta_2 d_2} e^{\beta_3 x}$
Chemotherapy	1	0	$e^{\beta_0} e^{\beta_1} e^{\beta_3 x}$
Radiation	0	1	$e^{\beta_0} e^{\beta_2} e^{\beta_3 x}$
Both	0	0	$e^{\beta_0} e^{\beta_3 x}$

For any given disease severity x,

$$\frac{\text{Survival odds with Chemo}}{\text{Survival odds with Both}} = \frac{e^{\beta_0} e^{\beta_1} e^{\beta_3 x}}{e^{\beta_0} e^{\beta_3 x}} = e^{\beta_1}$$

In general,

- When x_k is increased by one unit and all other independent variables are held constant, the odds of Y=1 are multiplied by e^{β_k}
- That is, e^{β_k} is an **odds ratio** --- the ratio of the odds of Y=1 when x_k is increased by one unit, to the odds of Y=1 when everything is left alone.
- As in ordinary regression, we speak of "controlling" for the other variables.

The conditional probability of Y=1

$$P(Y=1|x_1,\ldots,x_{p-1}) = \frac{e^{\beta_0+\beta_1 x_1+\ldots+\beta_{p-1}x_{p-1}}}{1+e^{\beta_0+\beta_1 x_1+\ldots+\beta_{p-1}x_{p-1}}}$$

This formula can be used to calculate a predicted P(Y=1)
Just replace betas by their estimates

It can also be used to calculate the probability of getting The sample data values we actually did observe, as a function of the betas.

Maximum likelihood estimation

- Likelihood = Conditional probability of getting the data values we did observe,
- As a function of the betas
- Maximize the (log) likelihood with respect to betas.
- Maximize numerically (“Iteratively re-weighted least squares”)
- Likelihood ratio tests as usual

Wald tests

- MLEs have an approximate multivariate normal sampling distribution for large samples (Thanks Mr. Wald.)
- Approximate mean vector = the true parameter values for large samples
- Asymptotic variance-covariance matrix is easy to estimate
- H_0: $\boldsymbol{C\theta} = \boldsymbol{h}$ (Linear hypothesis)
- For logistic regression, $\boldsymbol{\theta} = \boldsymbol{\beta}$

$$H_0 : \mathbf{C}\boldsymbol{\theta} = \mathbf{h}$$

$$\mathbf{C}\widehat{\boldsymbol{\theta}} - \mathbf{h} \text{ is multivariate normal as } n \to \infty$$

Leads to a straightforward chisquare test

- Called a Wald test
- Based on the full (maybe even saturated) model
- Asymptotically equivalent to the LR test
- Not as good as LR for smaller samples
- Very convenient, especially with SAS

$$Z = \frac{\hat{\theta}_k}{\sqrt{\widehat{Var}(\hat{\theta}_k)}}$$

- Approximately standard normal for large samples if $\theta_k=0$.
- Can use to form large-sample confidence intervals
- Denominator is the square root of a diagonal element of the asymptotic variance-covariance matrix of $\hat{\theta}$
- Square it to get a Wald test with 1 df.

Wald statistics and asymptotic standard errors

- Exist for the classical (non-conditional) log-linear models
- This is what the text is talking about in Section 5.4
- Not easy to get from R

- For logistic regression, straightforward with R as well as SAS

Detergent Data (Table 5-1)

```
> # Navigate to the location of the data using the Misc menu
> soapdata <- read.table("DetergentFrame.txt"); soapdata
   Softness Prev_Use   Temp Pref Freq
1    1=Soft    1=Yes 1=High  1=X   19
2    1=Soft    1=Yes 1=High  2=M   29
3    1=Soft    1=Yes  2=Low  1=X   57
4    1=Soft    1=Yes  2=Low  2=M   49
5    1=Soft     2=No 1=High  1=X   29
6    1=Soft     2=No 1=High  2=M   27
7    1=Soft     2=No  2=Low  1=X   63
8    1=Soft     2=No  2=Low  2=M   53
9    2=Medm    1=Yes 1=High  1=X   23
10   2=Medm    1=Yes 1=High  2=M   47
11   2=Medm    1=Yes  2=Low  1=X   47
12   2=Medm    1=Yes  2=Low  2=M   55
13   2=Medm     2=No 1=High  1=X   33
14   2=Medm     2=No 1=High  2=M   23
15   2=Medm     2=No  2=Low  1=X   66
16   2=Medm     2=No  2=Low  2=M   50
17   3=Hard    1=Yes 1=High  1=X   24
18   3=Hard    1=Yes 1=High  2=M   43
19   3=Hard    1=Yes  2=Low  1=X   37
20   3=Hard    1=Yes  2=Low  2=M   52
21   3=Hard     2=No 1=High  1=X   42
22   3=Hard     2=No 1=High  2=M   30
23   3=Hard     2=No  2=Low  1=X   68
24   3=Hard     2=No  2=Low  2=M   42
> soap <- xtabs(Freq ~ Softness+Prev_Use+Temp+Pref, data=soapdata)
> summary(soap)
Call: xtabs(formula = Freq ~ Softness + Prev_Use + Temp + Pref, data = soapdata)
Number of cases in table: 1008
Number of factors: 4
Test for independence of all factors:
      Chisq = 43.9, df = 18, p-value = 0.0005957
> loglin(soap,list(1,2,3,4))$lrt # Matches text, p. 76
2 iterations: deviation 1.136868e-13
[1] 42.92866
```

```
> # Strategy: Find a model for the explanatory variables, using a
> # marginal table. Then check links of explanatory to response.
> soapex = xtabs(Freq ~ Softness+Prev_Use+Temp, data=soapdata); soapex
, , Temp = 1=High

        Prev_Use
Softness 1=Yes 2=No
  1=Soft    48   56
  2=Medm    70   56
  3=Hard    67   72

, , Temp = 2=Low

        Prev_Use
Softness 1=Yes 2=No
  1=Soft   106  116
  2=Medm   102  116
  3=Hard    89  110

> summary(soapex)
Call: xtabs(formula = Freq ~ Softness + Prev_Use + Temp, data = soapdata)
Number of cases in table: 1008
Number of factors: 3
Test for independence of all factors:
      Chisq = 10.019, df = 7, p-value = 0.1875
> soapexA = loglin(soapex,list(1,2,3)) # Complete independence
2 iterations: deviation 1.136868e-13
> soapexA$lrt
[1] 10.10304
>
> # Check 2-d marginal tables anyway
> softemp = xtabs(Freq ~ Softness+Temp, data=soapdata); softemp
        Temp
Softness 1=High 2=Low
  1=Soft    104   222
  2=Medm    126   218
  3=Hard    139   199
> round(100*prop.table(softemp,1),2) # Row percents
        Temp
Softness 1=High 2=Low
  1=Soft  31.90 68.10
  2=Medm  36.63 63.37
  3=Hard  41.12 58.88
> summary(softemp)
Call: xtabs(formula = Freq ~ Softness + Temp, data = soapdata)
Number of cases in table: 1008
Number of factors: 2
Test for independence of all factors:
      Chisq = 6.082, df = 2, p-value = 0.04778
> # Harder water goes with higher temp, sort of
```

```
> softprev = xtabs(Freq ~ Softness+Prev_Use, data=soapdata); softprev
        Prev_Use
Softness 1=Yes 2=No
  1=Soft   154  172
  2=Medm   172  172
  3=Hard   156  182
> round(100*prop.table(softprev,1),2) # Row percents
        Prev_Use
Softness 1=Yes  2=No
  1=Soft 47.24 52.76
  2=Medm 50.00 50.00
  3=Hard 46.15 53.85
> summary(softprev)
Call: xtabs(formula = Freq ~ Softness + Prev_Use, data = soapdata)
Number of cases in table: 1008
Number of factors: 2
Test for independence of all factors:
      Chisq = 1.0753, df = 2, p-value = 0.5841
> # Not much

> prevtemp = xtabs(Freq ~ Prev_Use+Temp, data=soapdata); prevtemp
        Temp
Prev_Use 1=High 2=Low
   1=Yes    185   297
   2=No     184   342
> summary(prevtemp)
Call: xtabs(formula = Freq ~ Prev_Use + Temp, data = soapdata)
Number of cases in table: 1008
Number of factors: 2
Test for independence of all factors:
      Chisq = 1.2535, df = 1, p-value = 0.2629
> # Not much
>
> JustSoftemp = loglin(soapex,list(2,c(1,3)))
2 iterations: deviation 0
> JustSoftemp$lrt; JustSoftemp$df
[1] 4.003931
[1] 5
> 1-pchisq(JustSoftemp$lrt, JustSoftemp$df)
[1] 0.5488501
> # Fits fine. Any better than complete independence?
> G2Change = soapexA$lrt-JustSoftemp$lrt; G2Change
[1] 6.099104
> dfChange = soapexA$df-JustSoftemp$df; dfChange
[1] 2
> pvalChange = 1-pchisq(G2Change, df=dfChange)
> pvalChange
[1] 0.04738014
> # Okay, keep [Softness Temperature]
>
```

```
> # Any IV, DV link at all?
> ModelA = loglin(soap,list(2,4,c(1,3))); ModelA
2 iterations: deviation 5.684342e-14
$lrt
[1] 36.82955

$pearson
[1] 37.76417

$df
[1] 16

$margin
$margin[[1]]
[1] "Prev_Use"

$margin[[2]]
[1] "Pref"

$margin[[3]]
[1] "Softness" "Temp"


> 1-pchisq(ModelA$lrt,ModelA$df)
[1] 0.002216038
> # Something is going on. Try model with all 2-way links
> # between explanatory and response variables.
> link2 = loglin(soap,list(c(1,3),c(1,4),c(2,4),c(3,4))); link2
3 iterations: deviation 0.06630545
$lrt
[1] 11.54287

$pearson
[1] 11.45839

$df
[1] 12

$margin
$margin[[1]]
[1] "Softness" "Temp"

$margin[[2]]
[1] "Softness" "Pref"

$margin[[3]]
[1] "Prev_Use" "Pref"

$margin[[4]]
[1] "Temp" "Pref"


> # Fits well. Try adding each link separately, and compare
```

```
> loglin(soap,list(2,c(1,3),c(1,4)))$lrt
2 iterations: deviation 1.136868e-13
[1] 36.43426
> loglin(soap,list(c(1,3),c(2,4)))$lrt
2 iterations: deviation 5.684342e-14
[1] 16.24809
> loglin(soap,list(2,c(1,3),c(3,4)))$lrt
2 iterations: deviation 5.684342e-14
[1] 32.46795

> ModelB = loglin(soap,list(c(1,3),c(2,4))) # [Soft Temp] [PrevUse Pref]
2 iterations: deviation 5.684342e-14
> # Does it fit?
> ModelB$lrt; ModelB$df
[1] 16.24809
[1] 15
> 1-pchisq(ModelB$lrt, ModelB$df)
[1] 0.365758
> # Improvement?
> G2Change = ModelA$lrt-ModelB$lrt; G2Change
[1] 20.58147
> dfChange = ModelA$df-ModelB$df; dfChange
[1] 1
> pvalChange = 1-pchisq(G2Change, df=dfChange); pvalChange
[1] 5.71467e-06
> # I like this one. But just check to see if another link is justified.
>
> loglin(soap,list(c(1,3),c(2,4),c(1,4)))$lrt # Add [Soft Pref]?
2 iterations: deviation 2.842171e-14
[1] 15.85279
> loglin(soap,list(c(1,3),c(2,4),c(3,4)))$lrt # Add [Temp Pref]?
2 iterations: deviation 5.684342e-14
[1] 11.88649
> ModelC = loglin(soap,list(c(1,3),c(2,4),c(3,4))) # Adding [Temp Pref]
2 iterations: deviation 5.684342e-14
> G2Change = ModelB$lrt-ModelC$lrt; G2Change
[1] 4.361601
> dfChange = ModelB$df-ModelC$df; dfChange
[1] 1
> pvalChange = 1-pchisq(G2Change, df=dfChange); pvalChange
[1] 0.03675775
> # I have to take it. Is link2 an improvement over this?
>
> ModelD = link2
> G2Change = ModelC$lrt-ModelD$lrt; G2Change
[1] 0.3436218
> dfChange = ModelC$df-ModelD$df; dfChange
[1] 2
> pvalChange = 1-pchisq(G2Change, df=dfChange); pvalChange
[1] 0.8421384
> # Okay, Model C looks like the choice.
> # [1 3] [2 4] [3 4] = [Soft Temp] [PrevUse Pref] [Temp Pref]
```

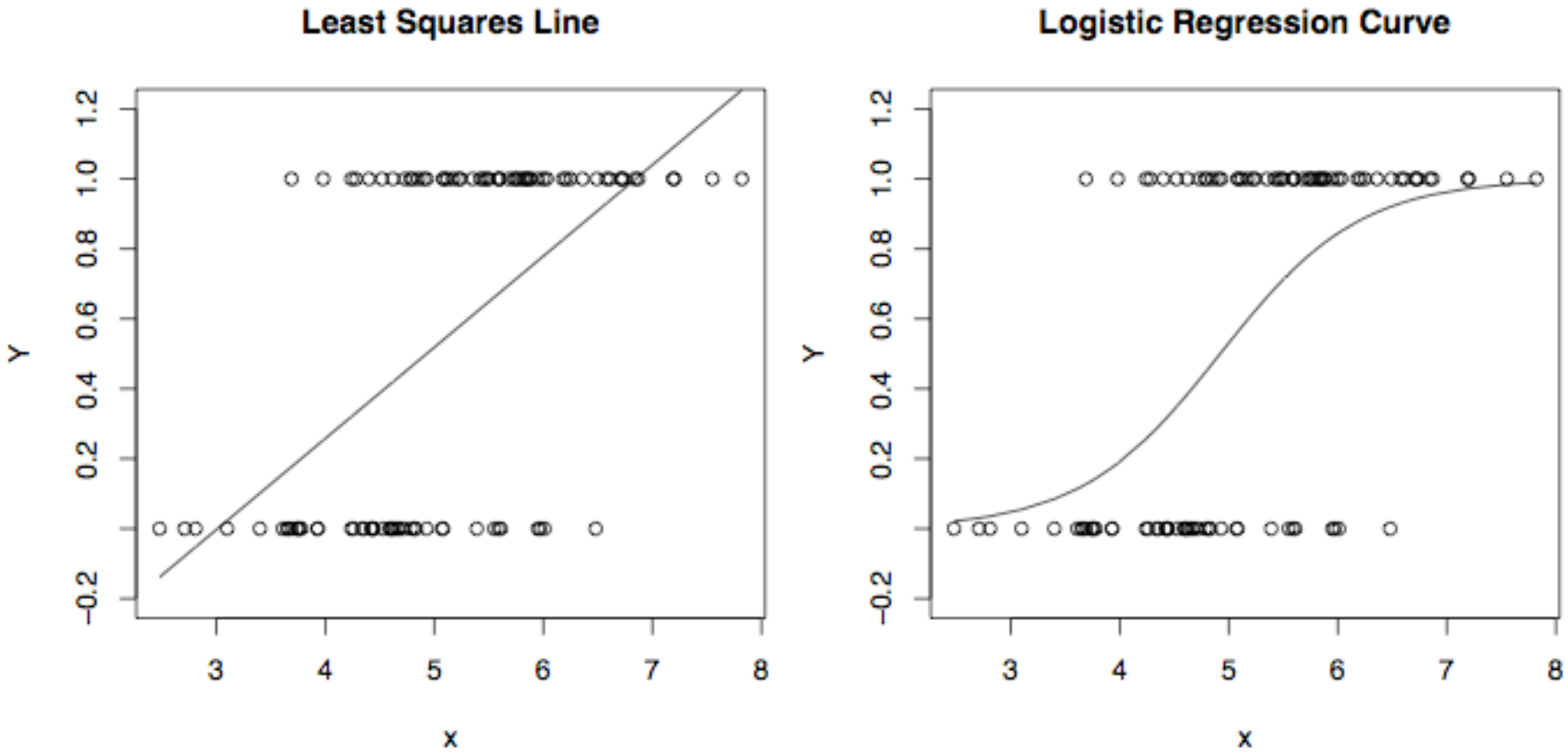
```
>
> # Look at marginal tables and parameter estimates to see what's happening
> PrevusePref = xtabs(Freq ~ Prev_Use+Pref, data=soapdata); PrevusePref
        Pref
Prev_Use 1=X 2=M
   1=Yes 207 275
   2=No  301 225
> round(100*prop.table(PrevusePref,1),2) # Row percents
        Pref
Prev_Use   1=X   2=M
   1=Yes 42.95 57.05
   2=No  57.22 42.78
> summary(PrevusePref)
Call: xtabs(formula = Freq ~ Prev_Use + Pref, data = soapdata)
Number of cases in table: 1008
Number of factors: 2
Test for independence of all factors:
      Chisq = 20.512, df = 1, p-value = 5.925e-06
> # Those who used M before tend to prefer it
> TempPref = xtabs(Freq ~ Temp+Pref, data=soapdata); TempPref
        Pref
Temp     1=X 2=M
  1=High 170 199
  2=Low  338 301
> round(100*prop.table(TempPref,1),2) # Row percents
        Pref
Temp       1=X   2=M
  1=High 46.07 53.93
  2=Low  52.90 47.10
> summary(TempPref)
Call: xtabs(formula = Freq ~ Temp + Pref, data = soapdata)
Number of cases in table: 1008
Number of factors: 2
Test for independence of all factors:
      Chisq = 4.358, df = 1, p-value = 0.03683
> # High temp goes with pref for M
```

```
> # Parameter estimates for Model C
> loglin(soap,list(c(1,3),c(2,4),c(3,4)),param=T)$param
2 iterations: deviation 5.684342e-14

$Softness.Temp
        Temp
Softness       1=High        2=Low
  1=Soft -0.101588153  0.101588153
  2=Medm  0.003448510 -0.003448510
  3=Hard  0.098139643 -0.098139643

$Prev_Use.Pref
        Pref
Prev_Use        1=X        2=M
   1=Yes -0.1437655  0.1437655
   2=No   0.1437655 -0.1437655

$Temp.Pref
        Pref
Temp            1=X        2=M
  1=High -0.0683605  0.0683605
  2=Low   0.0683605 -0.0683605

> #
> # Conclusions
> #
> #   Consumers with harder water tend to use higher temperature
> #   Those who used Brand M before tend to prefer it
> #   Use of High temperature water goes with preference for M
> #
> #   Book arrives at the same model
> #   But if the conclusions are actually stated in the book, I missed it.
```

Logistic Regression

For a binary dependent variable:
1=Yes, 0=No

Least Squares vs. Logistic Regression



Linear regression model for the log odds of the event Y=1

$$\ln\left(\frac{P(Y=1|\mathbf{X}=\mathbf{x})}{P(Y=0|\mathbf{X}=\mathbf{x})}\right) = \beta_0 + \beta_1 x_1 + \ldots + \beta_{p-1} x_{p-1}$$

Equivalent Statements

$$\ln\left(\frac{P(Y=1|\mathbf{X}=\mathbf{x})}{P(Y=0|\mathbf{X}=\mathbf{x})}\right) = \beta_0 + \beta_1 x_1 + \ldots + \beta_{p-1} x_{p-1}$$

$$\begin{aligned}\frac{P(Y=1|\mathbf{X}=\mathbf{x})}{P(Y=0|\mathbf{X}=\mathbf{x})} &= e^{\beta_0+\beta_1 x_1+\ldots+\beta_{p-1}x_{p-1}} \\ &= e^{\beta_0} e^{\beta_1 x_1} \ldots e^{\beta_{p-1}x_{p-1}}\end{aligned}$$

$$P(Y=1|x_1,\ldots,x_{p-1}) = \frac{e^{\beta_0+\beta_1 x_1+\ldots+\beta_{p-1}x_{p-1}}}{1+e^{\beta_0+\beta_1 x_1+\ldots+\beta_{p-1}x_{p-1}}}$$

$F(x) = \frac{e^x}{1+e^x}$ is called the *logistic distribution.*

- Could use any cumulative distribution function:

$$P(Y = 1|x_1, \ldots, x_{p-1}) = F(\beta_0 + \beta_1 x_1 + \ldots + \beta_{p-1} x_{p-1})$$

- CDF of the standard normal used to be popular
- Called probit analysis
- Can be closely approximated with a logistic regression.

In terms of log odds, logistic regression is like regular regression

$$\ln\left(\frac{P(Y=1|\mathbf{X}=\mathbf{x})}{P(Y=0|\mathbf{X}=\mathbf{x})}\right) = \beta_0 + \beta_1 x_1 + \ldots + \beta_{p-1} x_{p-1}$$

In terms of plain odds,

- Logistic regression coefficients represent *odds ratios*
- For example, “Among 50 year old men, the odds of being dead before age 60 are three times as great for smokers.”

$$\frac{\text{Odds of death given smoker}}{\text{Odds of death given nonsmoker}} = 3$$

Logistic regression

- X=1 means smoker, X=0 means non-smoker
- Y=1 means dead, Y=0 means alive

- Log odds of death = $\beta_0 + \beta_1 x$

- Odds of death = $e^{\beta_0} e^{\beta_1 x}$

$$\text{Odds of Death} = e^{\beta_0} e^{\beta_1 x}$$

Group	x	**Odds of Death**
Smokers	1	$e^{\beta_0} e^{\beta_1}$
Non-smokers	0	e^{β_0}

$$\frac{\text{Odds of death given smoker}}{\text{Odds of death given nonsmoker}} = \frac{e^{\beta_0} e^{\beta_1}}{e^{\beta_0}} = e^{\beta_1}$$

Cancer Therapy Example

$$\text{Log Survival Odds} = \beta_0 + \beta_1 d_1 + \beta_2 d_2 + \beta_3 x$$

Treatment	d_1	d_2	**Odds of Survival** $= e^{\beta_0} e^{\beta_1 d_1} e^{\beta_2 d_2} e^{\beta_3 x}$
Chemotherapy	1	0	$e^{\beta_0} e^{\beta_1} e^{\beta_3 x}$
Radiation	0	1	$e^{\beta_0} e^{\beta_2} e^{\beta_3 x}$
Both	0	0	$e^{\beta_0} e^{\beta_3 x}$

For any given disease severity x,

$$\frac{\text{Survival odds with Chemo}}{\text{Survival odds with Both}} = \frac{e^{\beta_0} e^{\beta_1} e^{\beta_3 x}}{e^{\beta_0} e^{\beta_3 x}} = e^{\beta_1}$$

In general,

- When x_k is increased by one unit and all other independent variables are held constant, the odds of Y=1 are multiplied by e^{β_k}
- That is, e^{β_k} is an **odds ratio** --- the ratio of the odds of Y=1 when x_k is increased by one unit, to the odds of Y=1 when everything is left alone.
- As in ordinary regression, we speak of "controlling" for the other variables.

The conditional probability of Y=1

$$P(Y=1|x_1,\ldots,x_{p-1}) = \frac{e^{\beta_0+\beta_1 x_1+\ldots+\beta_{p-1}x_{p-1}}}{1+e^{\beta_0+\beta_1 x_1+\ldots+\beta_{p-1}x_{p-1}}}$$

This formula can be used to calculate a predicted P(Y=1)
Just replace betas by their estimates

It can also be used to calculate the probability of getting The sample data values we actually did observe, as a function of the betas.

Maximum likelihood estimation

- Likelihood = Conditional probability of getting the data values we did observe,
- As a function of the betas
- Maximize the (log) likelihood with respect to betas.
- Maximize numerically (“Iteratively re-weighted least squares”)
- Likelihood ratio tests as usual

Wald tests

- MLEs have an approximate multivariate normal sampling distribution for large samples (Thanks Mr. Wald.)
- Approximate mean vector = the true parameter values for large samples
- Asymptotic variance-covariance matrix is easy to estimate
- H_0: $\boldsymbol{C\theta} = \boldsymbol{h}$ (Linear hypothesis)
- For logistic regression, $\boldsymbol{\theta} = \boldsymbol{\beta}$

$$H_0 : \mathbf{C}\boldsymbol{\theta} = \mathbf{h}$$

$$\mathbf{C}\widehat{\boldsymbol{\theta}} - \mathbf{h} \text{ is multivariate normal as } n \to \infty$$

Leads to a straightforward chisquare test

- Called a Wald test
- Based on the full (maybe even saturated) model
- Asymptotically equivalent to the LR test
- Not as good as LR for smaller samples
- Very convenient, especially with SAS

$$Z = \frac{\hat{\theta}_k}{\sqrt{\widehat{Var}(\hat{\theta}_k)}}$$

- Approximately standard normal for large samples if $\theta_k=0$.
- Can use to form large-sample confidence intervals
- Denominator is the square root of a diagonal element of the asymptotic variance-covariance matrix of $\hat{\theta}$
- Square it to get a Wald test with 1 df.

Wald statistics and asymptotic standard errors

- Exist for the classical (non-conditional) log-linear models
- This is what the text is talking about in Section 5.4
- Not easy to get from R

- For logistic regression, straightforward with R as well as SAS

Low Birth Weight Study

bweight.data

Col 1 = Identification Code
Col 2 = Low Birth Weight Baby (1=Yes under 2500g, 0=No)
Col 3 = Mother's age in years
Col 4 = Weight at Last Period
Col 5 = Race (1=White, 2=Black, 3=Other)
Col 6 = Smoke during Pregnancy (1=Yes, 0=No)
Col 7 = History of Premature Labour (# of times)
Col 8 = History of Hypertension (1=Yes, 0=No)
Col 9 = Presence of Uterine Irritability (1=Yes, 0=No)
Col 10 = Visits to Doctor During 1st trimester
Col 11 = Baby's birth Weight in Grams

```
> bweight = read.table("http://www.utstat.toronto.edu/~brunner/312f10/code_n_data/
bweight.data")
> bweight[1:5,]
   low age lwt race smoke ptl ht ui ftv  bwt
85   0  19 182    2     0   0  0  1   0 2523
86   0  33 155    3     0   0  0  0   3 2551
87   0  20 105    1     1   0  0  0   1 2557
88   0  21 108    1     1   0  0  1   2 2594
89   0  18 107    1     1   0  0  1   0 2600
> # The following is just to save some typing
> low <- bweight$low ; age <- bweight$age ; lwt <- bweight$lwt
> race <- bweight$race ; smoke <- bweight$smoke; ptl <- bweight$ptl
> ht <- bweight$ht; ui <- bweight$ui; ftv <- bweight$ftv
> # Crude descriptive stats
> table(low)
low
  0   1
130  59
> summary(age)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  14.00   19.00   23.00   23.24   26.00   45.00
> summary(lwt)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   80.0   110.0   121.0   129.8   140.0   250.0
> table(race) # 1=White, 2=Black, 3=Other
race
 1  2  3
96 26 67
> table(smoke)
smoke
  0   1
115  74
> table(ptl)
ptl
  0   1   2   3
159  24   5   1
> ptl[ptl>1]=1 # Collapsing categories
> table(ptl)
ptl
  0   1
159  30
```

```
> table(ht)
ht
  0   1
177  12
> table(ui)
ui
  0   1
161  28
> table(ftv)
ftv
  0   1   2   3   4   6
100  47  30   7   4   1
> # Don't collapse ftv for now

> # First, some simple examples to illustrate the methods
> # Two continuous independent variables
> model1 <- glm(low ~ age + lwt, family=binomial)
> summary(model1)

Call:
glm(formula = low ~ age + lwt, family = binomial)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.1352  -0.9088  -0.7480   1.3392   2.0595

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.748773   0.997097   1.754   0.0795 .
age         -0.039788   0.032287  -1.232   0.2178
lwt         -0.012775   0.006211  -2.057   0.0397 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 234.67  on 188  degrees of freedom
Residual deviance: 227.12  on 186  degrees of freedom
AIC: 233.12

Number of Fisher Scoring iterations: 4
```

$$\text{Deviance} = \sum_{i=1}^{n}(-2\log P\{Y_i = y_i|x_i, \widehat{\beta}\}) = \sum_{i=1}^{n} d_i$$

$$\text{Deviance Residual: } r_i^D = \text{sign}\left(y_i - P\{Y_i = y_i|x_i, \widehat{\beta}\}\right)\sqrt{d_i}$$

Null deviance is the deviance of a model with just the intercept.

```
> summary(model1)

Call:
glm(formula = low ~ age + lwt, family = binomial)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.1352  -0.9088  -0.7480   1.3392   2.0595

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.748773   0.997097   1.754   0.0795 .
age         -0.039788   0.032287  -1.232   0.2178
lwt         -0.012775   0.006211  -2.057   0.0397 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 234.67  on 188  degrees of freedom
Residual deviance: 227.12  on 186  degrees of freedom
AIC: 233.12

Number of Fisher Scoring iterations: 4

> model1$coefficients
(Intercept)         age         lwt
 1.74877349 -0.03978793 -0.01277541
> model1$deviance
[1] 227.1234
> model1$null.deviance
[1] 234.672
> # G-squared = Deviance(Reduced)-Deviance(Full)
> # df = difference in number of betas
> G2 = model1$null.deviance-model1$deviance; G2
[1] 7.548608
> 1-pchisq(G2,df=1)
[1] 0.006005646
> anova(model1)
Analysis of Deviance Table

Model: binomial, link: logit

Response: low

Terms added sequentially (first to last)


     Df Deviance Resid. Df Resid. Dev
NULL                   188    234.672
age   1    2.760       187    231.912
lwt   1    4.789       186    227.123

>
> 1-pchisq(4.789,1) # LR test of weight controlling for age
[1] 0.02864205
> 1-pchisq(2.057^2,1) # Wald test of weight controlling for age
[1] 0.03968623
>
> # Estimate probability of low birth weight for a 19 year old
> # mother weighing 120 pounds
> x = c(1,19,120); xb = sum(x*model1$coefficients)
> phat = exp(xb)/(1+exp(xb)); phat
[1] 0.3681301
```

```
> # For constant age, increase of weight by one pound multiplies
> # odds of low birth weight baby by ...
> exp(model1$coefficients[3])
      lwt
0.9873058

> # Represent race with 2 indicator dummy variables. First the hard way:
> n = length(race); n
[1] 189
> r1=numeric(n); r2 = numeric(n)
> r1[race==2]=1; r2[race==3]=1
> table(r1,race)
   race
r1   1  2  3
  0 96  0 67
  1  0 26  0
> table(r2,race)
   race
r2   1  2  3
  0 96 26  0
  1  0  0 67
>
> model2a = glm(low ~ r1 + r2, family=binomial); summary(model2a)

Call:
glm(formula = low ~ r1 + r2, family = binomial)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.0489  -0.9665  -0.7401   1.4041   1.6905

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -1.1550     0.2391  -4.830 1.36e-06 ***
r1            0.8448     0.4634   1.823   0.0683 .
r2            0.6362     0.3478   1.829   0.0674 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 234.67  on 188  degrees of freedom
Residual deviance: 229.66  on 186  degrees of freedom
AIC: 235.66

Number of Fisher Scoring iterations: 4

>
> G2a = model2a$null.deviance-model2a$deviance; G2a
[1] 5.010366
> 1-pchisq(G2a,2)
[1] 0.08166065
> racelow = table(race,low); racelow
    low
race  0  1
   1 73 23
   2 15 11
   3 42 25
> loglin(racelow,margin=list(1,2))$lrt
2 iterations: deviation 0
[1] 5.010366
```

```
>
> racefac <- factor(bweight$race,label=c("White","Black","Other"))
> contrasts(racefac)
      Black Other
White     0     0
Black     1     0
Other     0     1
> # So the default is indicator dummy variable coding
> model2b = glm(low ~ racefac, family=binomial)
> # summary(model2b) is 100% identical to summary(model2a)

> # Estimated odds of low birth weight baby are ___ times as
> # great for Blacks as Whites: Do it 2 ways
> # First directly with alpha
> racelow
    low
race  0  1
   1 73 23
   2 15 11
   3 42 25
> 73*11/(23*15)
[1] 2.327536
> # Now with logistic regression concepts
> exp(model2b$coefficients[2])
racefacBlack
    2.327536

>
> # Control for a continuous variable
> model3 = glm(low ~ lwt + racefac, family=binomial); summary(model3)

Call:
glm(formula = low ~ lwt + racefac, family = binomial)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.3491  -0.8919  -0.7196   1.2526   2.0993

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)   0.805753   0.845167   0.953   0.3404
lwt          -0.015223   0.006439  -2.364   0.0181 *
racefacBlack  1.081066   0.488052   2.215   0.0268 *
racefacOther  0.480603   0.356674   1.347   0.1778
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 234.67  on 188  degrees of freedom
Residual deviance: 223.26  on 185  degrees of freedom
AIC: 231.26

Number of Fisher Scoring iterations: 4
```

```
> G2change = model2b$deviance-model3$deviance; G2change
[1] 6.40254
> # What is H0?
> 1-pchisq(G2change,1)
[1] 0.01139572

> # Another way, using anova to compare 2 models
> anova(model2b,model3)
Analysis of Deviance Table

Model 1: low ~ racefac
Model 2: low ~ lwt + racefac
  Resid. Df Resid. Dev  Df Deviance
1       186    229.662
2       185    223.259   1    6.403
>

> # What about race controlling for weight?
> # Could fit a reduced model with just weight, but ...
> anova(model3)
Analysis of Deviance Table

Model: binomial, link: logit

Response: low

Terms added sequentially (first to last)


         Df Deviance Resid. Df Resid. Dev
NULL                       188    234.672
lwt       1    5.981       187    228.691
racefac   2    5.432       185    223.259
> 1-pchisq(5.432,2)
[1] 0.06613878


> # Still not statistically significant. It's time to get serious
> # about model building.
> bweight[1:5,]
   low age lwt race smoke ptl ht ui ftv  bwt
85   0  19 182    2     0   0  0  1   0 2523
86   0  33 155    3     0   0  0  0   3 2551
87   0  20 105    1     1   0  0  0   1 2557
88   0  21 108    1     1   0  0  1   2 2594
89   0  18 107    1     1   0  0  1   0 2600
> fullmod = glm(low ~ age+lwt+racefac+smoke+ptl+ht+ui+ftv,family=binomial)
```

```
> summary(fullmod)

Call:
glm(formula = low ~ age + lwt + racefac + smoke + ptl + ht +
    ui + ftv, family = binomial)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.6305  -0.7894  -0.5094   0.9119   2.2257

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)   0.644476   1.223889   0.527  0.59849
age          -0.039548   0.038305  -1.032  0.30186
lwt          -0.015078   0.007034  -2.143  0.03207 *
racefacBlack  1.218791   0.533168   2.286  0.02226 *
racefacOther  0.819439   0.450466   1.819  0.06890 .
smoke         0.859459   0.409836   2.097  0.03599 *
ptl           1.218512   0.463015   2.632  0.00850 **
ht            1.860429   0.708161   2.627  0.00861 **
ui            0.719299   0.463419   1.552  0.12062
ftv           0.050900   0.175456   0.290  0.77174
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 234.67  on 188  degrees of freedom
Residual deviance: 196.75  on 179  degrees of freedom
AIC: 216.75

Number of Fisher Scoring iterations: 4

> # Try dropping age, ui, ftv: Test simultaneously
> # What is H0?
> redmod1 = glm(low ~ lwt+racefac+smoke+ptl+ht,family=binomial)
> G2change1 = redmod1$deviance-fullmod$deviance; G2change1
[1] 3.732170
> 1-pchisq(G2change1,3)
[1] 0.2918750
> # No problem discarding these.
> # Controlling for the other vars, they do nothing.
> summary(redmod1)

Call:
glm(formula = low ~ lwt + racefac + smoke + ptl + ht, family = binomial)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.8188  -0.8035  -0.5457   0.9667   2.1530

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)   0.09462    0.95704   0.099  0.92124
lwt          -0.01673    0.00695  -2.407  0.01608 *
racefacBlack  1.26372    0.52933   2.387  0.01697 *
racefacOther  0.86418    0.43509   1.986  0.04701 *
smoke         0.87611    0.40071   2.186  0.02879 *
ptl           1.23144    0.44625   2.760  0.00579 **
ht            1.76744    0.70841   2.495  0.01260 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)
```

```
    Null deviance: 234.67  on 188  degrees of freedom
Residual deviance: 200.48  on 182  degrees of freedom
AIC: 214.48

Number of Fisher Scoring iterations: 4

> # Test all the variables at once.
> G2 = redmod1$null.deviance-redmod1$deviance; G2
[1] 34.18974
> 1-pchisq(G2,6)
[1] 6.182967e-06
> # What about race controlling for the other variables?
> redmod2 = glm(low ~ lwt+smoke+ptl+ht,family=binomial)
> G2race = redmod2$deviance-redmod1$deviance; G2race
[1] 7.47308
> 1-pchisq(G2race,2)
[1] 0.02383643
>
> # Controlling for other variables, the estimated odds
> # of a low birth weight baby are ___ times as great
> # for a Black mother as compared to a White mother.
> redmod1$coefficients
 (Intercept)          lwt racefacBlack racefacOther        smoke
  0.09461948  -0.01672867   1.26372441   0.86417633   0.87610630
         ptl           ht
  1.23143674   1.76744247
> exp(redmod1$coefficients[3])
racefacBlack
    3.538576
>
> # Controlling for other variables, the estimated odds
> # of a low birth weight baby are ___ times as great
> # for an Other mother as compared to a White mother.
> exp(redmod1$coefficients[4])
racefacOther
    2.373051

> # Controlling for other variables, are the odds of
> # a low birth weight baby different for Other and Black mothers?
```

$$\begin{aligned}\log \text{odds} &= \beta_0 + \beta_1 \text{ltw} + \beta_2 \text{r1} + \beta_2 \text{r2} + \beta_4 \text{smoke} + \beta_5 \text{ptl} + \beta_6 \text{ht} \\ &= \beta_0 + \beta_1 \text{ltw} + \beta_2 (\text{r1}+\text{r2}) + \beta_4 \text{smoke} + \beta_5 \text{ptl} + \beta_6 \text{ht}\end{aligned}$$

```
> r = r1+r2
> redmod3 = glm(low ~ lwt+r+smoke+ptl+ht,family=binomial)
> G2change = redmod3$deviance-redmod1$deviance; G2change
[1] 0.5313281
> 1-pchisq(G2change,1)
[1] 0.4660491
> # Consistent with no difference.
```

Bweight2: Comparing log-linear models and logistic regression

```
> bweight = read.table("http://www.utstat.toronto.edu/~brunner/312f10/code_n_data/
bweight.data")
> bweight[1:5,]
   low age lwt race smoke ptl ht ui ftv  bwt
85   0  19 182    2     0   0  0  1   0 2523
86   0  33 155    3     0   0  0  0   3 2551
87   0  20 105    1     1   0  0  0   1 2557
88   0  21 108    1     1   0  0  1   2 2594
89   0  18 107    1     1   0  0  1   0 2600
>
> # Confine attention to smoking, race, low birth weight
> race <- factor(bweight$race,label=c("White","Black","Other"))
> contrasts(race)
      Black Other
White     0     0
Black     1     0
Other     0     1
> contrasts(race) <- contr.sum # Effect coding
> contrasts(race)
      [,1] [,2]
White    1    0
Black    0    1
Other   -1   -1
> smoke <- factor(bweight$smoke,label=c("No","Yes"))
> contrasts(smoke) <- contr.sum
> contrasts(smoke)
    [,1]
No     1
Yes   -1
> low <- factor(bweight$low,label=c("No","Yes"))
> contrasts(low) <- contr.sum
>
> threeD = table(smoke,race,low)
> margin.table(threeD,c(1,3,2))
, , race = White

     low
smoke No Yes
  No  40   4
  Yes 33  19

, , race = Black

     low
smoke No Yes
  No  11   5
  Yes  4   6

, , race = Other

     low
smoke No Yes
  No  35  20
  Yes  7   5
```

```
> # Conditional log-linear model with no association between
> # explanatory and response variables
> loglin1 = loglin(threeD,list(c(1,2),3))
2 iterations: deviation 2.842171e-14
> G2 = loglin1$lrt; df = loglin1$df
> G2; df; 1-pchisq(G2,df)
[1] 17.85422
[1] 5
[1] 0.003134764
> # The equivalent logistic regression model is the null model
> logregfull = glm(low ~ smoke + race + smoke:race, family=binomial)
> # low ~ smoke*race is equivalent
> summary(logregfull)

Call:
glm(formula = low ~ smoke + race + smoke:race, family = binomial)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.3537  -0.9508  -0.4366   1.4190   2.1899

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  -0.68896    0.20323  -3.390 0.000699 ***
smoke1       -0.52793    0.20323  -2.598 0.009384 **
race1        -0.73837    0.26668  -2.769 0.005627 **
race2         0.49746    0.31665   1.571 0.116178
smoke1:race1 -0.34733    0.26668  -1.302 0.192778
smoke1:race2 -0.06903    0.31665  -0.218 0.827425
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 234.67  on 188  degrees of freedom
Residual deviance: 216.82  on 183  degrees of freedom
AIC: 228.82

Number of Fisher Scoring iterations: 4

> anova(logregfull)
Analysis of Deviance Table

Model: binomial, link: logit

Response: low

Terms added sequentially (first to last)


            Df Deviance Resid. Df Resid. Dev
NULL                          188    234.672
smoke        1    4.867       187    229.805
race         2    9.830       185    219.975
smoke:race   2    3.157       183    216.818

> G2b = logregfull$null.deviance-logregfull$deviance
> G2b; G2
[1] 17.85422
[1] 17.85422
>
> # Connection between MLEs for the 2 kinds of model:
> # Messy for 3 and higher-D tables
>
```

```
> # Z-tests for loglinfull suggest a logistic regression model
> # without the smoke by race interaction. This is equivalent to a
> # log-linear model without the smoke by race by low interaction.
> # In general, a main effect in logistic regression corresponds to
> # an interaction between that variable and the response variable
> # in a log-linear model -- provided, of course, that the log-linear
> # model also has all interactions among explanatory variables.
> # A k-factor interaction in logistic regression corresponds to a
> # k+1-factor interaction in a log-linear model, The k+1-factor interaction
> # has all the explanatory variables in the k-factor interaction, plus
> # the response variable. Again, this is assuming  that the log-linear
> # model has all interactions among explanatory variables.
>
> # Conduct this two-df test both ways, using LR tests.
> # First with logistic regression:
>
> logregreduced = glm(low ~ smoke + race, family=binomial)
> anodev = anova(logregreduced,logregfull); anodev

Analysis of Deviance Table

Model 1: low ~ smoke + race
Model 2: low ~ smoke + race + smoke:race
  Resid. Df Resid. Dev  Df Deviance
1       185    219.975
2       183    216.818   2    3.157
> anodev[2,3]; anodev[2,4]
[1] 2
[1] 3.156937

> # Now a log-linear model. Only mu123 is missing
> loglin2 = loglin(threeD,list(c(1,2),c(1,3),c(2,3)))
5 iterations: deviation 0.08003072
> loglin2
$lrt
[1] 3.157074

$pearson
[1] 3.113864

$df
[1] 2

$margin
$margin[[1]]
[1] "smoke" "race"

$margin[[2]]
[1] "smoke" "low"

$margin[[3]]
[1] "race" "low"


> 1-pchisq(loglin2$lrt,loglin2$df)
[1] 0.2062767
> 1-pchisq(anodev[2,4],anodev[2,3])
[1] 0.2062908
> # The no-interaction logistic regression model is fine
> # [smoke race] [smoke low] [race low]
```

```
> summary(logregreduced)

Call:
glm(formula = low ~ smoke + race, family = binomial)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.3442  -0.8862  -0.5428   1.4964   1.9939

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -0.5517     0.1833  -3.009  0.00262 **
smoke1       -0.5580     0.1846  -3.023  0.00251 **
race1        -0.7309     0.2490  -2.936  0.00333 **
race2         0.3532     0.2992   1.181  0.23776
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 234.67  on 188  degrees of freedom
Residual deviance: 219.97  on 185  degrees of freedom
AIC: 227.97

Number of Fisher Scoring iterations: 4

> # Why is the coefficient for smoke negative?
>
> # Test race controlling for smoke
> anodev2 = anova(logregreduced); anodev2
Analysis of Deviance Table

Model: binomial, link: logit

Response: low

Terms added sequentially (first to last)


       Df Deviance Resid. Df Resid. Dev
NULL                     188    234.672
smoke   1    4.867       187    229.805
race    2    9.830       185    219.975
> 1-pchisq(anodev2[3,2],anodev2[3,1])
[1] 0.007336125
> # Or,
> loglin3 = loglin(threeD,list(c(1,2),c(1,3)))
2 iterations: deviation 0
> G2change = loglin3$lrt-loglin2$lrt; G2change
[1] 9.829752
> dfchange = loglin3$df-loglin2$df; dfchange
[1] 2
> 1-pchisq(G2change,dfchange)
[1] 0.007336629
```

```
> # For ease of interpretation, prefer indicator dummy vars
> # when there are no interactions.
> race <- factor(bweight$race,label=c("White","Black","Other"))
> contrasts(race)
      Black Other
White     0     0
Black     1     0
Other     0     1
> smoke <- factor(bweight$smoke,label=c("No","Yes"))
> contrasts(smoke)
    Yes
No    0
Yes   1
> # Could have done: contrasts(smoke) <- contr.treatment
> # But labels were lost when we moved to effect coding
> logregreduced = glm(low ~ smoke + race, family=binomial)
> summary(logregreduced)

Call:
glm(formula = low ~ smoke + race, family = binomial)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.3442  -0.8862  -0.5428   1.4964   1.9939

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -1.8405     0.3529  -5.216 1.83e-07 ***
smokeYes      1.1160     0.3692   3.023  0.00251 **
raceBlack     1.0841     0.4900   2.212  0.02693 *
raceOther     1.1086     0.4003   2.769  0.00562 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 234.67  on 188  degrees of freedom
Residual deviance: 219.97  on 185  degrees of freedom
AIC: 227.97

Number of Fisher Scoring iterations: 4
```

Using the `glm` function on data that come in table format

```
> # Help says:
> # "For binomial and quasibinomial families the response can also be
> # specified as ... a two-column matrix with the columns giving the
> # numbers of successes and failures."
>
> margin.table(threeD,c(2,3,1))
, , smoke = No

       low
race    No Yes
  White 40   4
  Black 11   5
  Other 35  20

, , smoke = Yes

       low
race    No Yes
  White 33  19
  Black  4   6
  Other  7   5

> # Make a data frame from the output, and ...
> testdata <- read.table("TestFrame.txt"); testdata
  smoke  race No Yes
1    No White 40   4
2    No Black 11   5
3    No Other 35  20
4   Yes White 33  19
5   Yes Black  4   6
6   Yes Other  7   5

> Smoke <- factor(testdata$smoke); contrasts(Smoke)
    Yes
No    0
Yes   1
> Race <- factor(testdata$race,levels=c("White","Black","Other"))
> # Otherwise, alphabetical order makes Black the reference category
> contrasts(Race)
      Black Other
White     0     0
Black     1     0
Other     0     1
> # Recall we had trouble earlier controlling order of categories
> # in tables. The levels parameter will do the trick.
> LowBW <- cbind(testdata$Yes,testdata$No); LowBW
     [,1] [,2]
[1,]    4   40
[2,]    5   11
[3,]   20   35
[4,]   19   33
[5,]    6    4
[6,]    5    7
>
> # Notice order must be Yes, No!
>
> summary(glm(LowBW ~ Smoke + Race, family=binomial))
```

```
> summary(glm(LowBW ~ Smoke + Race, family=binomial))

Call:
glm(formula = LowBW ~ Smoke + Race, family = binomial)

Deviance Residuals:
       1         2         3         4         5         6
-0.93864  -0.05946   0.60978   0.59394   0.07123  -1.24205

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -1.8405     0.3529  -5.216 1.83e-07 ***
SmokeYes      1.1160     0.3692   3.023  0.00251 **
RaceBlack     1.0841     0.4900   2.212  0.02693 *
RaceOther     1.1086     0.4003   2.769  0.00562 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 17.8542  on 5  degrees of freedom
Residual deviance:  3.1569  on 2  degrees of freedom
AIC: 31.886

Number of Fisher Scoring iterations: 4

> # Compare:
> summary(logregreduced)

Call:
glm(formula = low ~ smoke + race, family = binomial)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.3442  -0.8862  -0.5428   1.4964   1.9939

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -1.8405     0.3529  -5.216 1.83e-07 ***
smokeYes      1.1160     0.3692   3.023  0.00251 **
raceBlack     1.0841     0.4900   2.212  0.02693 *
raceOther     1.1086     0.4003   2.769  0.00562 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 234.67  on 188  degrees of freedom
Residual deviance: 219.97  on 185  degrees of freedom
AIC: 227.97

Number of Fisher Scoring iterations: 4

> 17.8542-3.1569
[1] 14.6973
> logregreduced$null.deviance-logregreduced$deviance
[1] 14.69729
>
```