

# Extending the Linear Model



G.Janacek  
School of Computing Sciences,  
UEA

**Part I**  
**generalized linear models**

# Contents

<b>I</b>	<b>generalized linear models</b>	<b>1</b>
0.1	Introduction . . . . .	4
0.1.1	Strange data . . . . .	10
0.2	The exponential family . . . . .	11
0.3	The link . . . . .	12
0.4	The generalized linear model . . . . .	13
0.4.1	Estimation in models . . . . .	15
0.4.2	Model comparison . . . . .	15
0.4.3	Single parameter distributions . . . . .	15
0.4.4	Two parameter distributions . . . . .	16
0.4.5	Likelihood based parameter estimates . . . . .	17
0.4.6	Summary . . . . .	17
0.4.7	Model Notation . . . . .	18
0.5	Aliasing . . . . .	19
0.6	Offsets and weights . . . . .	21
0.7	Some Examples of glms . . . . .	22
0.7.1	Poisson Regression . . . . .	22
0.7.2	A Gamma example . . . . .	25
0.8	Toxicity of the Tobacco budworm: Moths revisited . . . . .	27
0.8.1	Yet another example : Clotting from Nelder . . . . .	37
0.9	Diagnostic methods . . . . .	38
0.10	Overdispersion . . . . .	42
0.11	Residuals . . . . .	46
0.12	More complex analysis . . . . .	48
0.12.1	Added Variable plots . . . . .	48
<b>II</b>	<b>Categorical Data</b>	<b>50</b>
0.12.2	Binomial problems . . . . .	51
0.12.3	Senility and WAIS . . . . .	55
0.12.4	The Hosmer-Lemeshow Goodness-of-Fit Test . . . . .	56
0.12.5	An agricultural example . . . . .	57
0.12.6	Ante-Natal Clinic . . . . .	60
0.12.7	Tetanus example . . . . .	60

---

0.13	Survival of breast cancer patients . . . . .	63
0.14	Coma Patients . . . . .	67
0.15	contingency tables . . . . .	68
0.16	The loglinear model . . . . .	69
0.16.1	Zero frequencies . . . . .	80
0.16.2	Sampling zeros . . . . .	80
0.16.3	Structural zeros . . . . .	81
0.16.4	Fitting distributions . . . . .	83
<b>III</b>	<b>Counting Processes</b>	<b>87</b>
0.17	Introduction . . . . .	88
0.17.1	Point Processes . . . . .	91
0.18	Introduction . . . . .	97
0.18.1	Manipulating data . . . . .	98
0.19	Help . . . . .	101
0.19.1	data input . . . . .	101
0.19.2	Smoothing . . . . .	104
0.19.3	Description . . . . .	107
0.20	Fitting generalized linear models: glm stats R Documentation . . . . .	115
0.21	Publications related to R . . . . .	120

## 0.1 Introduction

As you have seen in the previous lectures the linear model gives a large set of useful methods that we can apply to our datasets. Of course in reality many of the problems we are interested in are concerned with non-normal data or violate some regression assumption.

### Example 1

We take a simple data set in table 1 as an example. The table below gives the amount of plasma by age, see the plot supplied in figure 1. It can be seen from the plot that while

age	plasma	age	plasma	age	plasma	age	plasma	age	plasma
0	13.44	1	10.11	2	9.83	3	7.94	4	4.86
0	12.84	1	11.38	2	9.00	3	6.01	4	5.10
0	11.91	1	10.28	2	8.65	3	5.14	4	5.67
0	20.09	1	8.96	2	7.85	3	6.90	4	5.75
0	15.60	1	8.59	2	8.88	3	6.77	4	6.23

Table 1: Plasma data

there is a pretty linear looking trend the variation in the plasma readings is decreasing with age. This is a problem since, as you will recall when we use regression we assume that the variability is constant over the explanatory variables. To be quite explicit in this case a simple linear regression model might be

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i$$

where  $y$  and  $x$  denote plasma and age respectively while the  $\epsilon_i$  are independent zero mean random variables with a **common variance**. This model is flawed.

In this case we can argue that the (ordinary) least squares are unbiased, since we may argue for zero mean and independence of the error terms but our inferences will be wrong as we do not have constant variances. **In this sort of case we typically overestimate the precision of our estimates.**

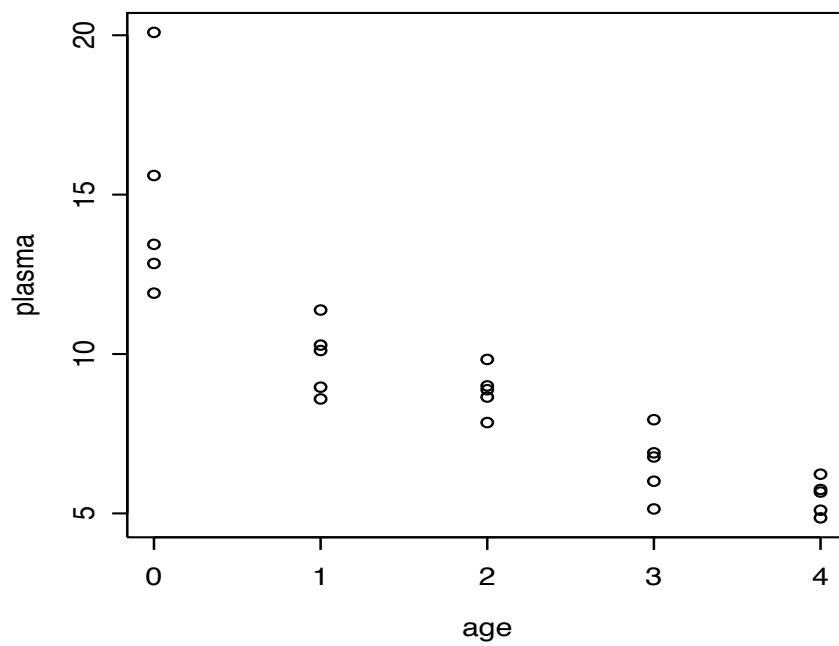


Figure 1: Plasma plot

	Variety			
	A	B	C	D
	0.8	4.0	9.8	6.0
	3.8	1.9	56.2	79.8
	0.0	0.7	66.0	7.0
	6.0	3.5	10.3	84.6
	1.7	3.2	9.2	2.8
mean	2.46	2.66	30.30	36.04
st.devn	2.435	1.343	28.332	42.201

Table 2: Bunt data

**Example 2**

The percentage of bunt infection in four varieties of wheat, sampled five times is shown in table 2. We fit a regression model with a factor to account for the four varieties. The outcome is

```
lm(formula = y ~ var)
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	2.46	11.38	0.216	0.8316
var2	0.20	16.10	0.012	0.9902
var3	27.84	16.10	1.729	0.1030
var4	33.58	16.10	2.086	0.0533 .

Residual standard error: 25.45 on 16 degrees of freedom

Multiple R-Squared: 0.315, Adjusted R-squared: 0.1866

F-statistic: 2.453 on 3 and 16 DF, p-value: 0.1008

```
anova(r1, test="F")
```

Analysis of Variance Table

Response: y

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
var	3	4767.3	1589.1	2.453	0.1008
Residuals	16	10365.3	647.8		

which is clearly rubbish

We use a

$$z = \sin^{-1} \left( \sqrt{\frac{y}{n}} \right)$$

which gives the values in table 3

Call:

```
lm(formula = yy ~ var)
```

Residuals:

```
      Min       1Q   Median       3Q      Max
-0.42298 -0.22518 -0.01102  0.07622  0.57642
```

Coefficients:

```
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.13280    0.13475   0.986   0.3390
var2         0.02549    0.19056   0.134   0.8952
var3         0.41703    0.19056   2.188   0.0438 *
var4         0.45831    0.19056   2.405   0.0286 *
---
```

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3013 on 16 degrees of freedom

Multiple R-Squared: 0.3848, Adjusted R-squared: 0.2695

F-statistic: 3.336 on 3 and 16 DF, p-value: 0.04599

```
> anova(r2, test="F")
```

Analysis of Variance Table

Response: yy

```
      Df Sum Sq Mean Sq F value Pr(>F)
var      3  0.90868  0.30289   3.3365 0.04599 *
Residuals 16  1.45252  0.09078
```

	Variety			
	A	B	C	D
	0.0896	0.2014	0.3184	0.2475
	0.1962	0.1383	0.8476	1.1047
	0.0000	0.0838	0.9483	0.2678
	0.2475	0.1882	0.3267	1.1675
	0.1308	0.1799	0.3082	0.1681
mean	0.1328	0.1583	0.5498	0.5911
sd	0.0958	0.0479	0.3198	0.4994

Table 3: Transformed Bunt data

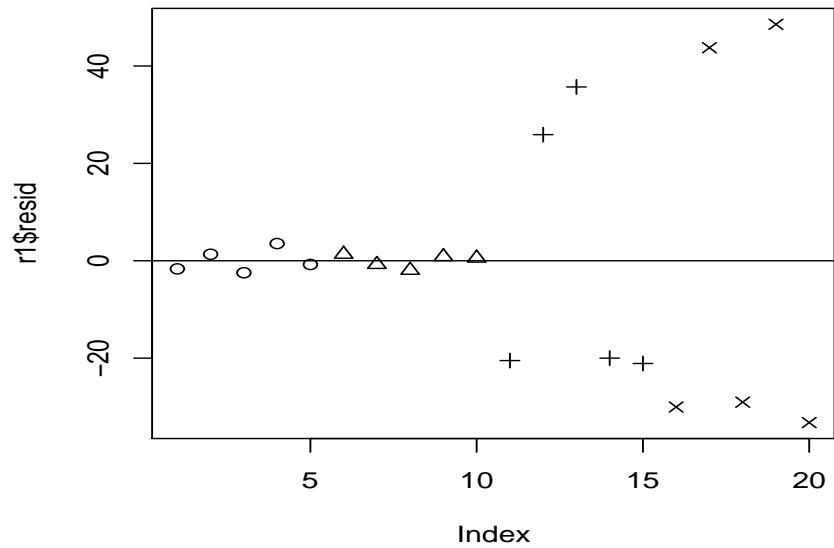


Table 4: Residuals in order

The reason for the odd results is that *one of our regression assumptions is wrong*. We assume that the error term is constant which is *not* the case here. The transformation does help but notice that the pattern remains. The original has residuals shown in figure 4 while the transformed residuals are shown in figure 5

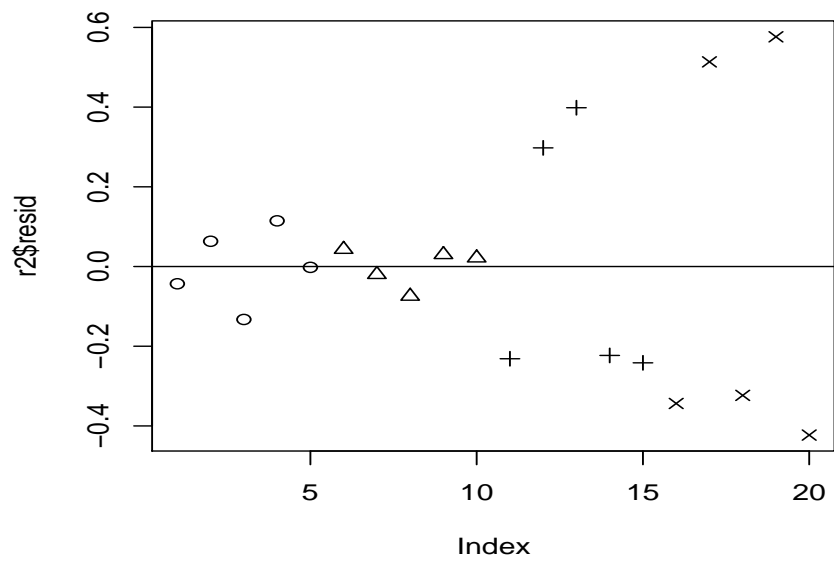


Table 5: Residuals in order

### 0.1.1 Strange data

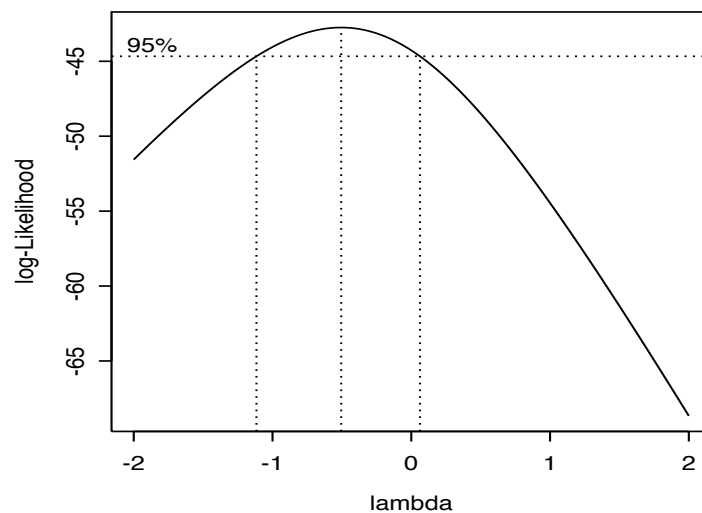
Strange data of this form is not at all uncommon. Atkinson points out

1. There may be gross errors in the response or the explanatory variables. These could arise from, amongst other things incorrect measurements and transcription errors.
2. A linear model may be inadequate to describe the systematic structure of the data.
3. It may be that the data would be better analyzed in another scale, for example by taking logarithms.
4. The systematic part of the model and the scale may be correct but the error distribution of the response is not normal, for example it may have long tails.
5. The error variance may not be constant.

One approach to the strange data problem is to transform the data. For example one might try a square root or log transformation. The aim is usually to *stabilize the variance* i.e. make it constant. For example the arcsin transform  $\arcsin \sqrt{x}$  can be shown to make the variance of Binomial variables constant. There are rather more systematic approaches such as the Box-Cox transform

$$y^{(\lambda)} = \begin{cases} (y_i^\lambda - 1)/\lambda & \lambda \neq 0 \\ \log(y_i) & \lambda = 0 \end{cases}$$

Here one chooses the optimal value for  $\lambda$  as in the plot below ( using the plasma data)



I want to discuss the extension to the linear model first proposed by Nelder in the mid 70's known as generalized linear modelling.

## 0.2 The exponential family

We cannot deal with all possible distributions for the error so we restrict ourselves to the exponential family. This is all distributions whose density or probability functions are of the form

$$f(y : \theta, \phi) = \exp\{(y\theta - b(\theta))/a(\phi) + c(y, \phi)\} \quad (1)$$

here  $\theta$  and  $\phi$  are parameters while  $a()$ ,  $b()$  and  $c()$  are known functions. Many if not most of the distributions we encounter belong to this family. Our interest is the canonical parameter  $\theta$  or, more usually the mean  $\mu$ , and we regard  $\phi$  as a nuisance parameter. If we use standard distributional results it is easy to show that

$$E[y] = \mu = b'(\theta) \quad (2)$$

and

$$\text{var}(y) = b''(\theta)a(\phi) \quad (3)$$

So we see that the variance is the product of two functions  $b''(\theta)$  and  $a(\phi)$  which depends only on  $\phi$ . We can write the variance function as a function of the mean  $\mu$ , say  $V(\mu)$ .

As an example take the Binomial distribution with mean  $\mu = n\pi$  and variance  $\sigma^2 = n\pi(1 - \pi)$ . Clearly we can write  $V(\mu) = \mu(1 - \frac{\mu}{n})$

It is common for the function  $a(\phi)$  to take the form

$$a(\phi) = \frac{\phi}{w}$$

Here  $\phi$ , called the dispersion parameter is constant over the data set and  $w$  is a known prior weight. The characteristics of some of the more important members of the exponential family are given below.

### Normal

For the normal

$$f(y) = \frac{1}{\sqrt{(2\pi\sigma^2)}} \exp\left[-\frac{(y - \mu)^2}{2\sigma^2}\right]$$

so in this parameterization  $a(\sigma) = \sigma^2$  and the nuisance parameter is  $\sigma$

### Gamma

For the Gamma we have

$$f(y) = \frac{1}{\Gamma(\nu)} \left(\frac{\nu}{\mu}\right)^\nu y^{\nu-1} \exp\left(-\frac{\nu y}{\mu}\right)$$

This parameterization gives  $E[Y] = \mu$  and  $\text{var}[Y] = \mu^2/\nu$  It seems clear that  $a(\nu) = \frac{1}{\nu}$

## Binomial

$$f(y) = \binom{n}{y} \pi^y (1 - \pi)^{n-y} \quad y = 0, 1, \dots, n$$

Here we have only one parameter  $\pi$  and the dispersion is one, that is  $a(\phi) = 1$ . Some authors, in particular Dobson, use a one parameter variant on the exponential family. They write the probability function as

$$f(y) = \exp\{a(\theta)b(y) + c(\theta) + d(y)\} \quad (4)$$

This has the advantage of simplicity when we deal with one parameter distributions but it can be confusing in the two parameter case. As we wish to deal with two parameter distributions like the normal or the gamma we shall stick to the explicit form.

## 0.3 The link

For the linear model we just equate the value of the predictor variables  $\eta$  to the mean  $\mu$ . When we have non-normal error distribution we try and be more flexible. Suppose

$$\eta_j = \mathbf{x}_j' \beta$$

is the linear combination of predictors  $\mathbf{x}$  for the  $j$ th response  $y_j$ . Here  $\beta$  is just the coefficient matrix.

For a generalized linear model we connect the mean and the predictor by a link function  $g(\cdot)$

$$g(\mu_j) = \eta_j = \mathbf{x}_j' \beta$$

The link is a monotone function and some possible choices are

1. Logit  $\eta = \log\{\mu/(1 - \mu)\}$
2. probit  $\eta = \Phi^{-1}(\mu)$  or  $\Phi(\mu) = \eta$
3. Complementary log-log  $\eta = \log\{-\log(1 - \mu)\}$

All the distributions that we commonly use have special canonical link functions where we have sensible sufficient statistics. The ones of interest to us are set out in table 6.

Distribution	Link
Normal	$\eta = \mu$
Poisson	$\eta = \log(\mu)$
Binomial	$\eta = \log\{\pi/(1 - \pi)\}$
Gamma	$\eta = \mu^{-1}$
Inverse Gaussian	$\eta = \mu^{-2}$

Table 6: canonical links for common distributions

While from the technical statistical view there is much to be said for the canonical link **there is no a priori reason why this link is appropriate for a particular data set.**

*The link function is chosen so that we have additivity and linearity of the explanatory variables. This choice is part of the modelling process.* Fortunately the canonical links work pretty well!

### Toxicity of the Tobacco budworm

Collett reports an experiment in which batches of 20 moths were exposed for 3 days to a pyrethroid. The number in each batch which were killed or knocked down are recorded in table 0.3 below. If we plot the numbers killed by dose we see in figure 2 that the

		dose					
		1	2	4	8	16	32
Sex	male	1	4	9	13	18	20
	female	0	2	6	10	12	16

Table 7: Numbers of moths killed

relation between the predictor variable (dose) and the response (deaths) is not linear. However since the response is binomial, deaths per 20, there is no real reason to suppose that it should be. One possibility is to use the logit link ( which is the canonical one for the binomial) and if the model is not satisfactory to try variants.

If you are not sure then you can try the empirical variant Suppose  $y_j$  is the binomial response from  $n$ , we can plot the empirical logistic

$$z_j = \log \left( \frac{y_j + 0.5}{n - y_j + 0.5} \right)$$

against the explanatory variable dose.

## 0.4 The generalized linear model

Suppose we have

- The error distribution
- the link function  $g()$
- A set of explanatory variables  $\mathbf{x}$  such that

$$E[y_j] = \mathbf{x}_j' \boldsymbol{\beta}$$

These three components make up a **generalized linear model** sometimes referred to as a *glm* or a *glim*.

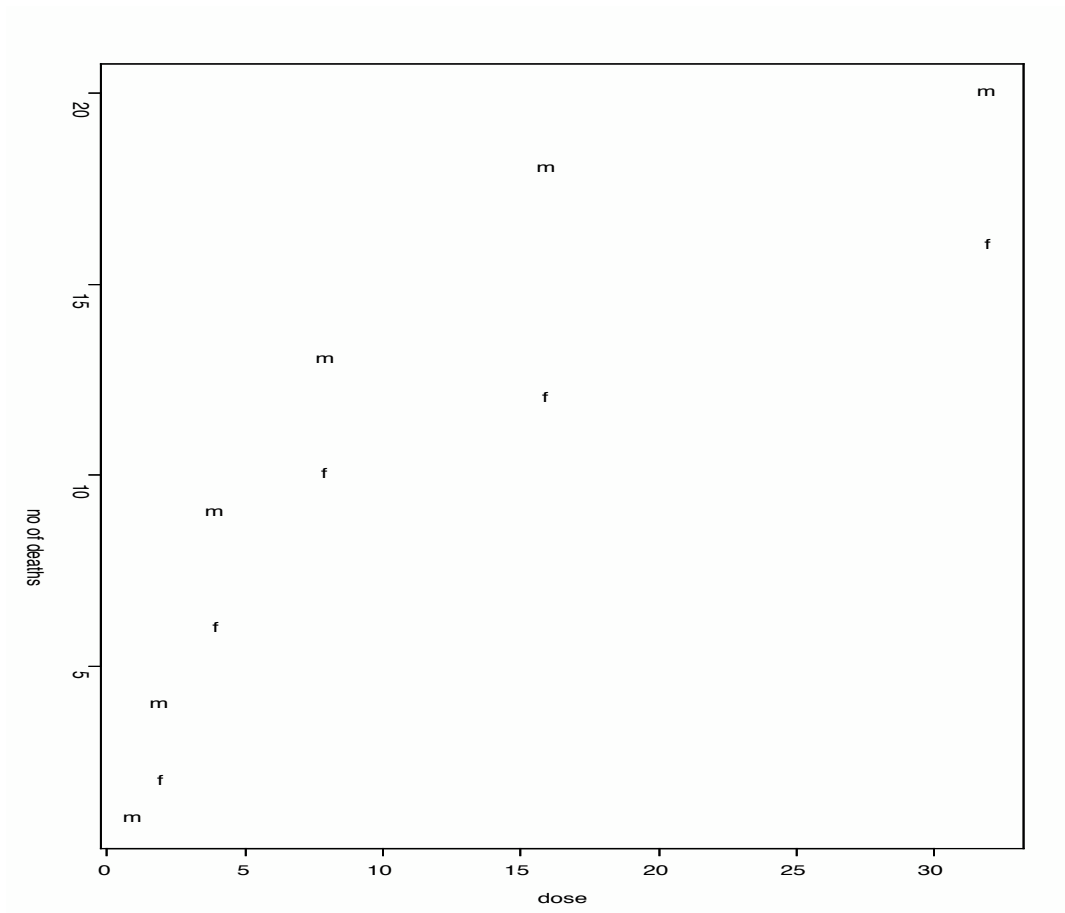


Figure 2: Moths deaths by dose

### 0.4.1 Estimation in models

For generalized linear models the natural approach to estimation is via the likelihood function. This will enable us to derive estimates of the various parameters in our model.

With these components of our model, see above, we can obtain the maximum likelihood estimates of the vector  $\beta$ . There is also the opportunity of using asymptotic likelihood theory to find confidence intervals for the parameters. Of course we still need some general way of comparing models.

### 0.4.2 Model comparison

The most complex model we can fit to the data provides an estimate for every mean value of every response. The maximum likelihood estimate of  $\mu_j$  based on a single observation  $y_j$  is  $y_j$  so this can be regarded as the *maximal, or saturated model*.

If we have a  $y_1, y_2, \dots, y_n$  which we assume are normally distributed, then the saturated model would assume a different mean  $\mu_j$  for each  $y_j$ . If you think about it you will realize that the estimates are  $\hat{\mu}_j = y_j \quad j = 1, 2, \dots, n$

If we take any less complex model with some  $\beta_{\mathbf{p}}$  with  $p \leq N$  parameters we can compare the log likelihoods

$$\ell(\beta_{\mathbf{max}}) - \ell(\beta_{\mathbf{p}})$$

In fact we use

$$D = 2 [\ell(\beta_{\mathbf{max}}) - \ell(\beta_{\mathbf{p}})] \quad (5)$$

a quantity known as *the deviance*. The deviance is preferable because we know that under fairly straightforward regularity conditions it has a chi-squared distribution with  $N - p$  degrees of freedom. Here  $N$  is the number of observations and  $p$  is the number of parameters in the beta vector under the simple model.

If we have two non maximal models say one  $M_1$  with  $\beta_{\mathbf{p}}$  and  $p$  parameters and a simpler variant,  $M_2$ , with  $q \leq p$  parameters then the change in deviance in moving from the simpler to the complex is just the difference in the model deviances

$$\Delta D = 2 [\ell(\beta_{\mathbf{p}}) - \ell(\beta_{\mathbf{q}})]$$

This is chi-squared with  $p - q$  degrees of freedom if the dispersion is known. We can use the analysis of deviance as an aid in choosing the appropriate model. To compare two models  $M_1$  and  $M_2$  ( provided  $M_2$  is a sub-model of  $M_1$ ) We look at the change in deviance  $\Delta D$  as we move from  $M_1$  to  $M_2$ .

### 0.4.3 Single parameter distributions

If there is no dispersion parameter in the distribution  $\Delta D$  is chi-squared with  $p - q$  degrees of freedom. It follows that if  $\Delta D$ , is not significantly different from zero ( it is less than the upper percentage point of the Chi-squared distribution) then the two models are equivalent and *we choose the simpler one*. On the other hand a significant  $\Delta D$  means that the more complex model is preferable. Of course the deviance gives a

way of assessing the overall fit of the model, at least in comparison with the saturated one. One should however bear in mind that these results are asymptotic and need to be treated with appropriate caution.

Some authors choose the model with minimum AIC, a criterion due to Akaike . We can define this as

$$\text{AIC} = -2 \times \text{maximising log likelihood} + 2 \times \text{the number of parameters}$$

The strategy in this case is to select the model which minimises this criterion.

#### 0.4.4 Two parameter distributions

For our two dimensional exponential family we are often given the scaled deviance by our software. To understand why we need to look at the likelihood. The two parameter exponential form is

$$f(y : \theta, \phi) = \exp[(y\theta - b(\theta))/a(\phi) + c(y, \phi)]$$

so the likelihood is

$$\ell(\theta, \phi) = \sum_{j=1}^N [(y_j \theta_j - b(\theta_j))/a(\phi) + c(y_j, \phi)]$$

If we have two models then the change in deviance ( or even the deviance if one of the models is saturated ) has the form

$$\Delta D = \frac{1}{a(\phi)} \sum_{j=1}^N \{y_j [\hat{\theta}_j - \tilde{\theta}_j] - [b(\hat{\theta}_j) - b(\tilde{\theta}_j)]\}$$

Now when

$$a(\phi) = \frac{\phi}{w}$$

$$\Delta D = \frac{w}{\phi} \sum_{j=1}^N \{y_j [\hat{\theta}_j - \tilde{\theta}_j] - [b(\hat{\theta}_j) - b(\tilde{\theta}_j)]\}$$

When the scale factor  $\frac{\phi}{w}$  is known then there is no problem. This may well be true for example the exponential distribution is a Gamma with scale 1.

In many cases the Scale factor is not known and must be estimated- this is clearly the case with the normal distribution. Provided a reasonable model has been fitted to the data we may estimate  $\psi$  by the mean deviance, that is

$$\hat{\phi} = D/(N - p)$$

Note this is not in general a consistent estimator! **Many programs provide  $\phi D$  the scaled deviance** As we see this is sensible since we need to estimate the parameters

and testing is do by considering ratios! The argument is that as  $\Delta D$  is the ratio of two chi squared variates we should be thinking in terms of an approximate F distribution rather than chi-squared. If we have deviances  $D_1$  and  $D_2$  for models  $M_1$  and  $M_2$  we replace the chi-squared tests by F tests. If  $M_2$  is nested in  $M_1$  then

$$F = \frac{(D_2 - D_1)/(p - q)}{D_1/(N - p)} \quad (6)$$

where  $q \leq p$  is the number of parameters in  $M_2$  while  $p$  is the number in  $M_1$ . As always  $N$  is the number of observations. F has, as you might expect an F distribution with  $(p - q)$  and  $(N - p)$  degrees of freedom, that is we reject the hypothesis that the models are equivalent if the value of F exceed the critical values in the F tables.

#### 0.4.5 Likelihood based parameter estimates

One great advantage of using likelihood based estimates is that we know a great deal about their distributional properties. This for a vector  $\hat{\beta}$  we know that the parameter estimates are jointly normal and that

$$E[\hat{\beta}] = \beta \text{ and } var(\hat{\beta}) = -E[\mathbf{H}^{-1}]$$

where

$$\mathbf{H} = \left[ \frac{\partial^2 \ell}{\partial \beta_i \partial \beta_j} \right]$$

In most cases we concentrate on the variances of the parameter estimates and ignore the off-diagonal terms of the variance-covariance matrix. The reason is simple for if we have  $\hat{\beta}_j$  and  $var[\hat{\beta}_j]$  then we can look at

$$z = \frac{\hat{\beta}_j}{std[\hat{\beta}_j]}$$

If  $\beta_j$  is zero the we can, for large samples, regard  $z$  as standard normal. This gives us a simple way of checking that the estimates are zero.

#### 0.4.6 Summary

1. Parameter estimates are asymptotically normal but be careful in Binomial cases.
2. For single parameter models deviance changes are distributed as  $\chi^2$
3. For two parameter distributions we cannot use  $\chi^2$  for deviance changes as the dispersion has to be estimated so we use the F distribution, see equation 6.

### 0.4.7 Model Notation

It is convenient to have a way of writing down the models we aim to fit. Most programs have a variant of the Wilkinson and Rogers notation used by GLIM and GENSTAT. The response is defined as ( for a linear model)

$$response \sim \pm term_1 \pm term_2 \pm term_3 \dots$$

Suppose the response is  $\mathbf{y}$  while X is a covariate i.e a continuous variable while A and B are factors.

- 1 implies a constant is fitted. This is often implicit in the model.  
The algebraic form is  $y_i = \beta + \epsilon_i$
- X implies X effects are included.  
The algebraic form is  $y_i = \beta_0 + \beta_1 x_i + \epsilon_i$
- A implies A effects are included.  
The algebraic form is  $y_{ij} = \beta_0 + \alpha_j + \epsilon_{ij}$
- A+B implies A and B effects are included.  
The algebraic form is  $y_{ij} = \beta_0 + \alpha_i + \beta_j + \epsilon_{ij}$
- A - B implies the effects of A less the effects of B
- A.B is the A.B interaction for factors and A.X means fitting different slopes for each factor to the covariate X.
- A\*B is equivalent to A+B+A.B  
The algebraic form is  $y_{ijk} = \beta_0 + \alpha_i + \beta_j + (\alpha\beta)_{ij} + \epsilon_{ijk}$  where the  $(\alpha\beta)_{ij}$  are the interaction terms.

The notation generalizes in a natural way.

## 0.5 Aliasing

As you have seen from the examples earlier in the course we have a mix of continuous explanatory variables and discrete ones. The latter, the factors, have levels and you will see some of these appear to have missing estimates. This is an *aliasing* effect and is, in this case an artifact of the way we have parameterized the model. To see how this happens we look at a simple case.

Suppose we have 3 treatments which are applied 4 times each to members of a population chosen at random. The responses are  $y_{jk}$  where  $j$  is the level of the factor (treatment) and  $k$  is the number of the observation on this treatment. A simple model is

$$y_{jk} = \mu + \alpha_j + u_{jk}$$

where the  $u_{jk}$  are independent errors,  $\mu$  is the global mean and the  $\alpha_j$  are the treatment effects. When we consider the estimation problem we have 3 samples and 4 level parameters to estimate which we cannot do. We say that the parameters are *aliased*. To get around this problem we add a further condition, the usual ones are

- Corner-point:

Here we set one of the alphas to zero, say  $\alpha_1 = 0$  and estimate the rest. The constant term in the first sample is set to  $\mu$  while the others become

$$\mu + \alpha_2, \mu + \alpha_3$$

We are in effect making comparisons between the sample using the first as the baseline.

- Sum-to-zero:

Here we set the sum of the  $\alpha$  to zero viz.  $\sum_{i=2}^3 \alpha_i = 0$ . We are in some sense measuring the deviation of the samples from the mean level.

There are others but they are rather obscure.

### An example

We take some plant yield data

type	yield	type	yield	type	yield
1	4.17	2	4.81	3	6.31
1	5.58	2	4.17	3	5.12
1	5.18	2	4.41	3	5.54
1	6.11	2	3.59	3	5.50
1	4.50	2	5.87	3	5.37
1	4.61	2	3.83	3	5.29
1	5.17	2	6.03	3	4.92
1	4.53	2	4.89	3	6.15
1	5.33	2	4.32	3	5.80
1	5.14	2	4.69	3	5.26

```

> g1<-glm(yield~type)
> summary(g1)
Call:
glm(formula = yield ~ type)
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  5.0320     0.1971  25.527  <2e-16 ***
type2       -0.3710     0.2788  -1.331   0.1944
type3        0.4940     0.2788   1.772   0.0877 .
Null \index{deviance} deviance: 14.258  on 29  degrees of freedom
Residual \index{deviance} deviance: 10.492  on 27  degrees of freedom
AIC: 61.619
Number of Fisher Scoring iterations: 2
> g2<-glm(yield~type-1)
> summary(g2)

Call:
glm(formula = yield ~ type - 1)
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
type1     5.0320     0.1971  25.53  <2e-16 ***
type2     4.6610     0.1971  23.64  <2e-16 ***
type3     5.5260     0.1971  28.03  <2e-16 ***
(\index{dispersion parameter} dispersion parameter for gaussian family taken to be 0.388)

Null \index{deviance} deviance: 786.318  on 30  degrees of freedom
Residual \index{deviance} deviance:  10.492  on 27  degrees of freedom
AIC: 61.619

```

You can see that the residual deviance is the same for models `g1` and `g2`. They give different estimates because they are made with different parameterisations. They do however give the same predictions. Of course this problem may arise in much more complex models with many factors and to interactions. One can have a different kind of aliasing where a parameter cannot be estimated because the data is not collected in such a way as to make this possible. Usually the effect is confounded with another. Thus for example suppose we have two kinds of drug which we wish to compare for the relief of Asthma. We also have two kinds of inhaler to supply the drug. If we run each drug in a different inhaler then we cannot distinguish the inhaler effect from the drug effect- the inhaler is aliased.

## 0.6 Offsets and weights

GLIM, R and some other packages allow the user to define *offsets* to weight the data. The weights given to each point during fitting depend upon the error distribution and link but may be modified by multiplying by prior weights contained in a special weight variate, so that the variances of individual points are divided by these prior weights.

This is useful if you are allowed to set zero weight but this is not always possible. The weighting function is an oddly Bayesian idea. It is clearly useful in setting up data sets where we wish to omit one or two observations. A more interesting application is changing the perceived precision of a data point. We may be uneasy about a point and to hence inflate its variance or, and this may be more likely to deflate the variance of a point if we have some a priori knowledge, perhaps it is a sum of  $n$  values and we can discount by  $\sqrt{n}$ .

### Known Parameter Values

It is sometimes required to fit a model where some of the  $\beta_j$ , in the linear predictor  $\eta_i = \sum x_{ij}\beta_j$  are fixed in advance.

Thus in a simple dilution assay the proportion of fertile tubes  $\pi$  is related to the dilution  $u$  by

$$\pi = 1 - \exp(-\lambda u_i)$$

so that the complementary log-log transformation gives

$$\eta_i = \log(-\log(1 - \pi_i)) = \log\lambda + \log u_i$$

If we write  $x_i = \log u_i$  as the covariate and  $\alpha = \log\lambda$  as the intercept we have a GLM with

$$\eta_i = \alpha + x_i$$

i.e. the slope is fixed at 1.

More generally if a subset of the  $\beta_j$  are fixed the sum of their contributions to  $\eta_i$  is called an *offset* so that

$$\eta_i = \text{offset} + \sum x_{ij}\beta_j$$

where the summation is over the terms for which the  $\beta_j$ , are not fixed. In fitting such a model the offset is first subtracted from the linear predictor and the result can then be regressed on the remaining covariates.

This gives some interesting possibilities. Declaring an offset can in fact reduce the number of parameters. Thus if we have as the link  $g(\pi_j) = \alpha + \beta x_j$  and we know that  $x_0$  and  $\pi_0$  then  $g(\pi_0) = \alpha + \beta x_0$  so we can eliminate one of the coefficients.

## 0.7 Some Examples of glms

### 0.7.1 Poisson Regression

In this example, the number of maintenance repairs on a complex system are modeled as realizations of Poisson random variables. The system under investigation has a large number of components, which occasionally break down and are replaced or repaired. During a four-year period, the system was thought to be in a state of steady operation, meaning that the rate of operation remained approximately constant. A monthly maintenance record is available for that period, which tracks the number of components removed for maintenance each month. The data are listed in the following data set, table 8. For the plot see figure 3. We might look for a model with a Poisson

year	month	defects	year	month	defects	year	month	defects
1987	1	2	1987	2	4	1987	3	3
1987	4	3	1987	5	3	1987	6	8
1987	7	2	1987	8	6	1987	9	3
1987	10	9	1987	11	4	1987	12	10
1988	1	4	1988	2	6	1988	3	4
1988	4	4	1988	5	3	1988	6	5
1988	7	3	1988	8	4	1988	9	5
1988	10	3	1988	11	6	1988	12	3
1989	1	2	1989	2	6	1989	3	1
1989	4	5	1989	5	5	1989	6	4
1989	7	2	1989	8	2	1989	9	2
1989	10	5	1989	11	1	1989	12	10
1990	1	3	1990	2	8	1990	3	12
1990	4	7	1990	5	3	1990	6	2
1990	7	4	1990	8	3	1990	9	0
1990	10	6	1990	11	6	1990	12	6

Table 8: Maintenance Data

error and with explanatory variates like time and month. We include a factor for month as there appears to be a monthly effect. We did look at years but the years did not seem to contribute anything to our model. Thus we have model with a factor `month` and a covariate `time.`, that is

```
y~month+time
```

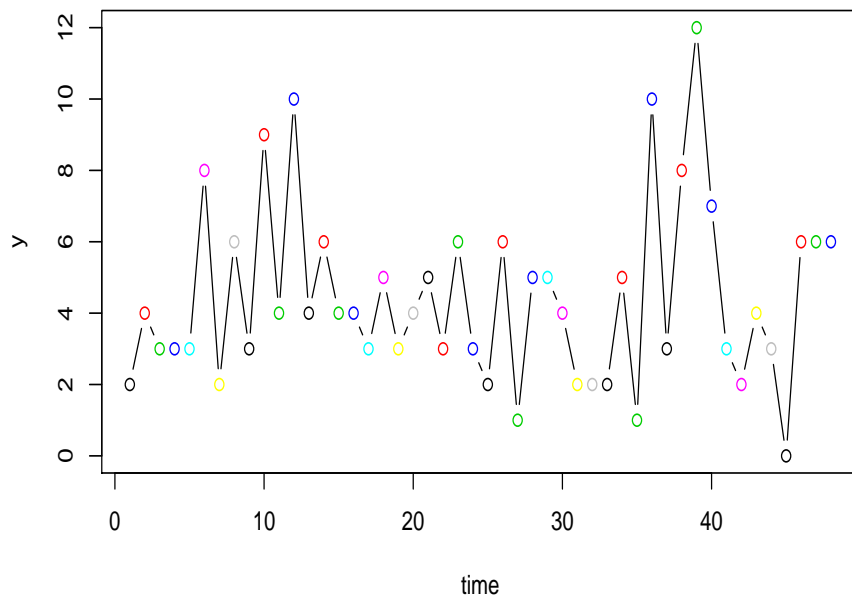


Figure 3: Plot of Poisson data

```
> anova(p2)
Analysis of \index{deviance} deviance Table
Model: poisson, link: log
Response: y
Terms added sequentially (first to last)
      Df \index{deviance} deviance Resid. Df Resid. Dev P(>|Chi|)
NULL                                47      66.892
time  1      0.146                46      66.746      0.703
month 11    20.990                35      45.756      0.033
```

```
> summary(p2)
glm(formula = y ~ time + month, family = poisson)
Coefficients:
      Estimate Std. Error z value Pr(>|z|)
(Intercept)  0.9996154  0.3169902   3.153  0.00161 **
time          0.0006289  0.0051192   0.123  0.90222
month2        0.7795296  0.3641455   2.141  0.03230 *
month3        0.5965791  0.3755182   1.589  0.11213
month4        0.5446569  0.3791788   1.436  0.15089
month5        0.2386463  0.4034315   0.592  0.55416
month6        0.5433990  0.3797313   1.431  0.15243
month7       -0.0037736  0.4275063  -0.009  0.99296
month8        0.3057523  0.3985723   0.767  0.44301
month9       -0.1003417  0.4388465  -0.229  0.81914
month10       0.7319385  0.3694727   1.981  0.04759 *
month11       0.4290287  0.3903246   1.099  0.27170
month12       0.9624822  0.3585566   2.684  0.00727 **
(\index{dispersion parameter} dispersion parameter for poisson family taken to be 1)
Null \index{deviance} deviance: 66.892 on 47 degrees of freedom
Residual \index{deviance} deviance: 45.756 on 35 degrees of freedom
AIC: 224.48
```

```
glm(formula = y ~ month + time - 1, family = poisson)
Coefficients:
      Estimate Std. Error z value Pr(>|z|)
month1  0.9996154  0.3169902   3.153  0.00161 **
month2  1.7791450  0.2286227   7.782  7.14e-15 ***
month3  1.5961946  0.2483588   6.427  1.30e-10 ***
month4  1.5442723  0.2558251   6.036  1.58e-09 ***
month5  1.2382617  0.2922819   4.237  2.27e-05 ***
month6  1.5430144  0.2605175   5.923  3.16e-09 ***
month7  0.9958418  0.3277758   3.038  0.00238 **
month8  1.3053678  0.2907525   4.490  7.14e-06 ***
```

```

month9 0.8992737 0.3453479 2.604 0.00922 **
month10 1.7315539 0.2533590 6.834 8.24e-12 ***
month11 1.4286441 0.2846679 5.019 5.20e-07 ***
month12 1.9620976 0.2413445 8.130 4.30e-16 ***
time 0.0006289 0.0051192 0.123 0.90222

```

```

Null \index{deviance} deviance: 368.695 on 48 degrees of freedom
Residual \index{deviance} deviance: 45.756 on 35 degrees of freedom
AIC: 224.48
Number of Fisher Scoring iterations: 5

```

## 0.7.2 A Gamma example

We now look at a problem where the error is Gamma. What we have to remember is that *the Gamma is a two parameter distribution*. This means that when we look at the deviances we *must use F distributions*. Otherwise our approach is must the same.

The table gives the time in minutes taken by four chimpanzees to learn each of 10 words.

Chimpanzee	word									
	1	2	3	4	5	6	7	8	9	10
1	178	60	177	36	225	345	40	2	287	14
2	78	14	80	15	10	115	10	12	129	80
3	99	18	20	25	15	54	25	10	476	55
4	297	20	195	18	24	420	40	15	372	190

The experimenters felt that a linear model was inappropriate and fitted a Gamma error model as shown below with a log link function. Their results were

```
glm(formula = y ~ word + chimp, family = Gamma(link = "log"))
```

Coefficients:

```

          Estimate Std. Error t value Pr(>|t|)
(Intercept)  5.4585     0.3754  14.540 2.73e-14 ***
word2        -1.7707     0.4657  -3.803 0.000744 ***
word3        -0.3649     0.4657  -0.784 0.440070
word4        -1.8214     0.4657  -3.911 0.000559 ***
word5        -1.1018     0.4657  -2.366 0.025409 *
word6         0.2786     0.4657   0.598 0.554607
word7        -1.7249     0.4657  -3.704 0.000963 ***
word8        -2.5546     0.4657  -5.486 8.28e-06 ***
word9         0.8109     0.4657   1.741 0.093009 .
word10       -0.5137     0.4657  -1.103 0.279722
chimp2       -0.9734     0.2945  -3.305 0.002686 **
chimp3       -0.8078     0.2945  -2.743 0.010679 *
chimp4       -0.1128     0.2945  -0.383 0.704786

```

```
(\index{dispersion parameter} dispersion parameter for Gamma family taken to be 0.433666)
```

```
Analysis of \index{deviance} deviance Table
```

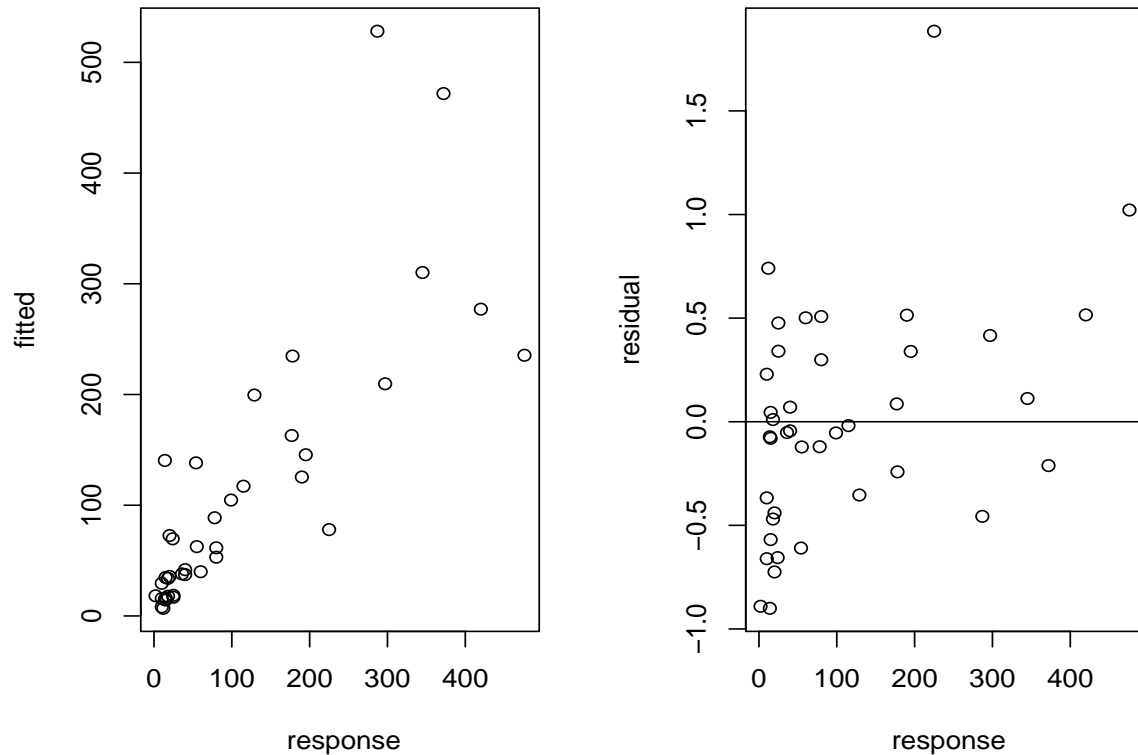


Figure 4: Gamma model with log link

	Df	deviance	Resid. Df	Resid. Dev
NULL			39	60.378
word	9	39.186	30	21.192
chimp	3	6.220	27	14.972

Ask yourself whether the chimp and word factors should both be included in the model. The plots in figure 4 give some idea of the adequacy of the model.

We can run the same model estimation program but using the canonical link. This gives

```
glm(formula = y ~ word + chimp, family = Gamma)
Coefficients:
          Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.0047989  0.0023093   2.078  0.0473 *
word2        0.0287224  0.0131458   2.185  0.0378 *
```

word3	0.0021110	0.0036311	0.581	0.5658
word4	0.0355228	0.0156153	2.275	0.0311 *
word5	0.0079215	0.0056423	1.404	0.1717
word6	-0.0015755	0.0025285	-0.623	0.5385
word7	0.0277971	0.0128101	2.170	0.0390 *
word8	0.0954116	0.0374056	2.551	0.0167 *
word9	-0.0024574	0.0023351	-1.052	0.3019
word10	0.0052334	0.0046961	1.114	0.2749
chimp2	0.0072057	0.0038898	1.852	0.0749 .
chimp3	0.0030877	0.0026527	1.164	0.2546
chimp4	-0.0005166	0.0015701	-0.329	0.7447

Analysis of `\index{deviance}` deviance Table

Model: Gamma, link: inverse

Response: time

Terms added sequentially (first to last)

	Df	deviance	Resid. Df	Resid. Dev
NULL		39	60.378	
word	9	39.186	30	21.192
chimp	3	4.111	27	17.081

The corresponding figure to 4 is figure 5. It is, at least in my view, quite difficult to choose between the models. One possibility is to choose the model with the smallest AIC. This is the Akaike Information Criterion and is commonly used for non-nested models. It is the maximum of the likelihood penalized by the number of parameters. The idea being that if one uses more parameters you always get a better fit even if you are fitting redundant terms. Using AIC as the only criterion is of course dangerous but sometimes we have little if any option.

## 0.8 Toxicity of the Tobacco budworm: Moths revisited

Collett reports an experiment in which batches of 20 moths were exposed for 3 days to a pyrethroid. The number in each batch which were killed or knocked down are recorded below.

		dose					
		1	2	4	8	16	32
Sex	male	1	4	9	13	18	20
	female	0	2	6	10	12	16

Table 9: Numbers of moths killed

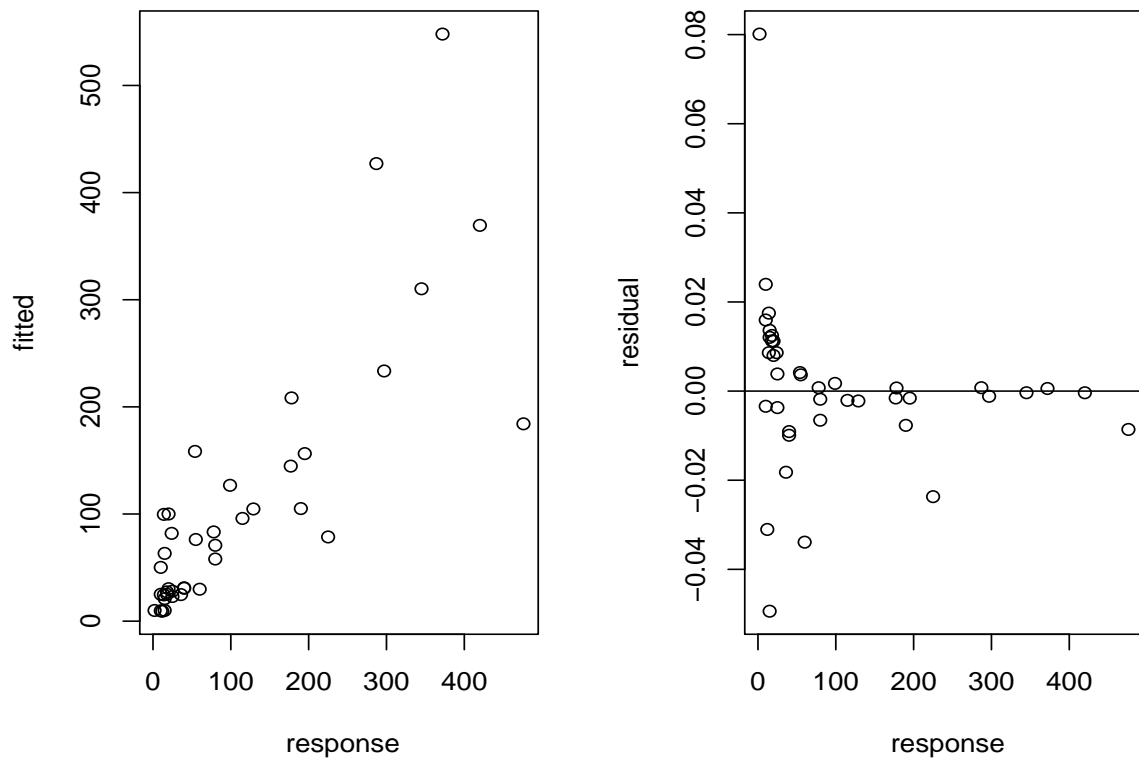


Figure 5: Gamma model with canonical link

- We have a Binomial error
- We will choose the default link function
- We have two explanatory variables,  $\log(\text{dose})$  and  $\text{sex}$ . Note we use  $\log$  dose as we did some exploratory analysis.

We get the data into R as follows: I have a text file `moth.txt` which looks like

```

deaths live dose sex
1    19    1    1
4    16    2    1
9    11    4    1
13   7     8    1
18   2    16    1
20   0    32    1
0    20    1    2
2    18    2    2
6    14    4    2
10   10    8    2
12   8     16   2
16   4     32   2

```

I read the data into R

```
moth<-read.table(file.choose(),header=T)
```

Using the resulting dialog box I get my table in R.

```

> moth
  deaths live dose sex
1      1  19    1   1
2      4  16    2   1
3      9  11    4   1
4     13   7    8   1
5     18   2   16   1
6     20   0   32   1
7      0  20    1   2
8      2  18    2   2
9      6  14    4   2
10     10  10    8   2
11     12   8   16   2
12     16   4   32   2

```

To make the names available I type

```
attach(moth)
```

Then `sex<-factor(sex)` - we have coded sex as a 1, 2 variable a *factor*.

We can now fit models using the `glm` command. To do so we need the log of the dose, say by `ldose<-log(dose)`. We also have a binomial error and this adds a small complication.

If we use a Binomial error we need to tell our generalized linear model program about both the successes and the failures - here deaths and living. In R we do this by using an `n` by 2 matrix as the response. The first column are the successes and the second the failures. So here we have

```
> yy<-cbind(deaths, live)
> yy
      deaths live
[1,]      1  19
[2,]      4  16
[3,]      9  11
[4,]     13   7
[5,]     18   2
[6,]     20   0
[7,]      0  20
[8,]      2  18
[9,]      6  14
[10,]     10  10
[11,]     12   8
[12,]     16   4
```

The most complex model ( that we chose to fit) is fitted with the default logistic link

```
> g0<-glm(yy~sex*log(dose),family=binomial)
> summary(g0)
```

Call:

```
glm(formula = yy ~ sex * log(dose), family = binomial)
```

\index{deviance} deviance Residuals:

	Min	1Q	Median	3Q	Max
	-1.39849	-0.32094	-0.07592	0.38220	1.10375

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	-2.8186	0.5480	-5.143	2.70e-07 ***
sex2	-0.1750	0.7783	-0.225	0.822
log(dose)	1.8163	0.3059	5.937	2.91e-09 ***
sex2:log(dose)	-0.5091	0.3895	-1.307	0.191

(\index{dispersion parameter} dispersion parameter for binomial family taken to be 1)

```
Null \index{deviance} deviance: 124.8756 on 11 degrees of freedom
Residual \index{deviance} deviance: 4.9937 on 8 degrees of freedom
AIC: 43.104
```

```
> anova(g0)
```

```
Analysis of \index{deviance} deviance Table
```

```
Model: binomial, link: logit
```

```
Response: yy
```

```
Terms added sequentially (first to last)
```

	Df	\index{deviance} deviance	Resid. Df	Resid. Dev
NULL			11	124.876
sex	1	6.077	10	118.799
log(dose)	1	112.042	9	6.757
sex:log(dose)	1	1.763	8	4.994

We can of course choose lesser models as below

```
> g1<-glm(yy~sex+log(dose),family=binomial)
> summary(g1)
```

Call:

```
glm(formula = yy ~ sex + log(dose), family = binomial)
```

\index{deviance} deviance Residuals:

Min	1Q	Median	3Q	Max
-1.10540	-0.65343	-0.02225	0.48471	1.42944

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	-2.3724	0.3855	-6.154	7.56e-10 ***
sex2	-1.1007	0.3558	-3.093	0.00198 **
log(dose)	1.5353	0.1891	8.119	4.70e-16 ***

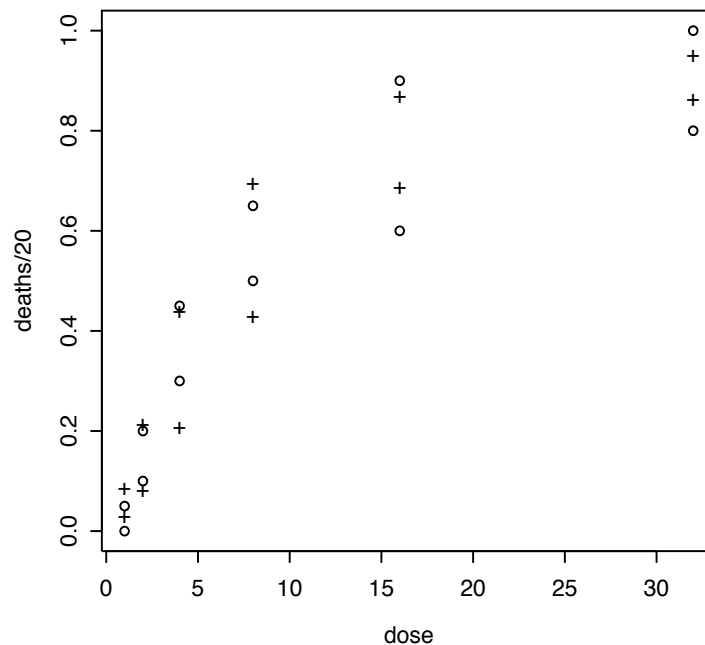
(\index{dispersion parameter} dispersion parameter for binomial family taken to be 1)

```
Null \index{deviance} deviance: 124.876 on 11 degrees of freedom
Residual \index{deviance} deviance: 6.757 on 9 degrees of freedom
AIC: 42.867
```

```
> cbind(deaths/20,g1$fitted,g1$resid)
      [,1]      [,2]      [,3]
[1,] 0.05 0.08530076 -0.45243124
[2,] 0.20 0.21278854 -0.07634514
[3,] 0.45 0.43930479  0.04342067
[4,] 0.65 0.69428515 -0.20864293
[5,] 0.90 0.86812073  0.27845269
[6,] 1.00 0.95020002  1.05240885
[7,] 0.00 0.03008577 -1.03101772
[8,] 0.10 0.08249341  0.23129846
[9,] 0.30 0.20673372  0.56871437
[10,] 0.50 0.43032791  0.28420672
[11,] 0.60 0.68647712 -0.40179626
[12,] 0.80 0.86388206 -0.54326147
```

I have of course produced the fitted values of the proportion of deaths to compare with the actual and the residuals. *Model critique is vital* as are plots

```
> plot(dose,deaths/20)
> points(dose,g1$fitted,pch='+')
```



**A normal case**

The table 10 gives the output of thermocouples made of 3 different materials at 3 different temperatures. Possible models are

	material					
Temperature	M1		M2		M3	
T1	130	155	34	40	20	70
	74	80	80	75	82	58
T2	150	188	136	122	25	70
	159	126	108	115	58	45
T3	138	110	174	120	96	104
	168	160	150	139	82	60

Table 10: Output voltages

$$y_{ijk} = \mu + \alpha_i + \beta_j + \epsilon_{ijk} \tag{7}$$

and

$$y_{ijk} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij} + \epsilon_{ijk} \tag{8}$$

where  $(\alpha\beta)_{ij}$  denotes the interaction between the main effects of material and temperature. As before we have the corner point constraint

$$(\alpha\beta)_{1j} = 0 \quad i = 1, 2, 3 \text{ and } (\alpha\beta)_{i1} = 0 \quad j = 1, 2, 3$$

The table of means, see table 11 is and these are plotted in figure 6. We see that the response over temperature is different for the different materials - hence the interaction.

R analysis gives

```
> y
[1] 130 155 34 40 20 70 74 80 80 75 82 58 150 188 136 122 25 70
[19] 159 126 108 115 58 45 138 110 174 120 96 104 168 160 150 139 82 60
> temp
[1] 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 3 3 3 3
Levels: 1 2 3
> mat
```

	M1	M2	M3
T1	109.75	155.75	144.00
T2	57.25	120.25	145.75
T3	57.50	49.50	85.50

htb

Table 11: Means of voltages

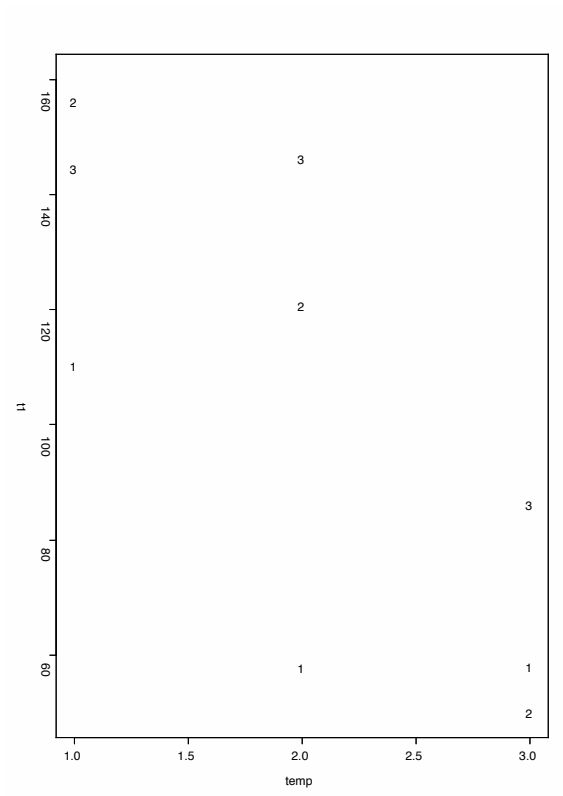


Figure 6: interaction example

```

[1] 1 1 2 2 3 3 1 1 2 2 3 3 1 1 2 2 3 3 1 1 2 2 3 3 1 1 2 2 3 3 1 1 2 2 3 3
Levels: 1 2 3
> v1<-lm(y~mat*temp)
> summary(v1)
lm(formula = y ~ mat * temp)
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   109.75      12.41    8.845 1.85e-09 ***
mat2          -52.50      17.55   -2.992 0.00586 **
mat3          -52.25      17.55   -2.978 0.00607 **
temp2         46.00      17.55    2.621 0.01421 *
temp3         34.25      17.55    1.952 0.06141 .
mat2:temp2     17.00      24.82    0.685 0.49917
mat3:temp2    -54.00      24.82   -2.176 0.03848 *
mat2:temp3     54.25      24.82    2.186 0.03766 *
mat3:temp3     -6.25      24.82   -0.252 0.80307
Residual standard error: 24.82 on 27 degrees of freedom
Multiple R-Squared: 0.7706, Adjusted R-squared: 0.7026
F-statistic: 11.34 on 8 and 27 DF, p-value: 7.018e-07

> anova(v1)
Analysis of Variance Table
Response: y
      Df Sum Sq Mean Sq F value    Pr(>F)
mat     2  31833   15916 25.8433 5.355e-07 ***
temp    2  15734    7867 12.7736 0.0001247 ***
mat:temp 4   8296    2074  3.3676 0.0232758 *
Residuals 27 16629     616
---

```

Notice we have several observations per cell in the data table. We can thus estimate the interaction terms. If there was only one observation per cell then the interaction terms would be aliased. They would be inseparable from the main effect terms!

Notice we have several observations per cell in the data table. We can thus estimate the interaction terms. If there was only one observation per cell then the interaction terms would be aliased. They would be inseparable from the main effect terms!

Of course this problem may arise in much more complex models with many factors and to interactions.

```
glm(formula = ROT ~ BACTERIA * TEMP * OXYGEN, family = gaussian)
```

```
\index{deviance} deviance Residuals:
```

Min	1Q	Median	3Q	Max
-9.333e+00	-2.917e+00	8.993e-15	2.667e+00	1.200e+01

```
Coefficients:
```

	Value	Std.error	t value	P(> t )
(Intercept)	9.4074	0.6586	14.2830	0.0000
BACTERIA1	-4.1296	0.9315	-4.4335	0.0001
BACTERIA2	-0.2407	0.9315	-0.2585	0.7975
TEMP1	-3.9630	0.6586	-6.0168	0.0000
OXYGEN1	1.8704	0.9315	2.0080	0.0522
OXYGEN2	-1.2407	0.9315	-1.3320	0.1912
BACTERIA1.TEMP1	2.2407	0.9315	2.4056	0.0214
BACTERIA2.TEMP1	-0.4259	0.9315	-0.4573	0.6502
BACTERIA1.OXYGEN1	1.0185	1.3173	0.7732	0.4445
BACTERIA2.OXYGEN1	-1.3704	1.3173	-1.0403	0.3051
BACTERIA1.OXYGEN2	-0.8704	1.3173	-0.6607	0.5130
BACTERIA2.OXYGEN2	0.7407	1.3173	0.5623	0.5774
TEMP1.OXYGEN1	-0.0926	0.9315	-0.0994	0.9214
TEMP1.OXYGEN2	0.2407	0.9315	0.2585	0.7975
BACTERIA1.TEMP1.OXYGEN1	1.3148	1.3173	0.9981	0.3249
BACTERIA2.TEMP1.OXYGEN1	-0.1852	1.3173	-0.1406	0.8890
BACTERIA1.TEMP1.OXYGEN2	-1.6852	1.3173	-1.2793	0.2090
BACTERIA2.TEMP1.OXYGEN2	1.8148	1.3173	1.3777	0.1768

```
(\index{dispersion parameter} dispersion parameter for gaussian family taken to be 23.42)
```

```
Null \index{deviance} deviance: 2707.037 on 53 degrees of freedom
```

```
Residual \index{deviance} deviance: 843.3333 on 36 degrees of freedom
```

The data is set out below

B	T	O	R	B	T	O	R	B	T	O	R
1	1	1	7	2	1	1	2	3	1	1	13
1	1	1	7	2	1	1	4	3	1	1	11
1	1	1	9	2	1	1	9	3	1	1	3
1	1	2	0	2	1	2	4	3	1	2	10
1	1	2	0	2	1	2	5	3	1	2	4
1	1	2	0	2	1	2	10	3	1	2	7
1	1	3	9	2	1	3	4	3	1	3	15
1	1	3	0	2	1	3	5	3	1	3	2
1	1	3	0	2	1	3	0	3	1	3	7
1	2	1	10	2	2	1	17	3	2	1	26
1	2	1	6	2	2	1	18	3	2	1	19
1	2	1	10	2	2	1	8	3	2	1	24
1	2	2	4	2	2	2	3	3	2	2	15
1	2	2	10	2	2	2	23	3	2	2	22
1	2	2	5	2	2	2	7	3	2	2	18
1	2	3	8	2	2	3	15	3	2	3	20
1	2	3	0	2	2	3	14	3	2	3	24
1	2	3	10	2	2	3	17	3	2	3	8

It is often useful to parameterize a model in such a way as to make the whole thing most sensible for the user.

### 0.8.1 Yet another example : Clotting from Nelder

The table gives mean clotting time in seconds of blood for nine concentrations (u) of normal plasma and two lots of clotting blood. A standard analysis gives

```

response is time
error distribution is gamma
link is reciprocal
scale is estimated using mean \index{deviance} deviance
offset is none
weight is none
maximum iterations 10
convergence tolerance 1.000e-4
alias tolerance 1.000e-5
residual method \index{deviance} deviance

```

```

Model is ln(u)+lot+ln(u).lot
\index{deviance} deviance is 2.964e-2 df 14

```

	estimate	se(est)	t ratio	Prob> t
1 Constant	-1.654e-2	8.625e-4	-19.18	<0.0001

u	ln(u)	lot	time
5	1.6090000	1	118
10	2.3030000	1	58
15	2.7079999	1	42
20	2.9960001	1	35
30	3.4010000	1	27
40	3.6889999	1	25
60	4.0939999	1	21
80	4.3820000	1	19
100	4.6050000	1	18
5	1.6090000	2	69
10	2.3030000	2	35
15	2.7079999	2	26
20	2.9960001	2	21
30	3.4010000	2	18
40	3.6889999	2	16
60	4.0939999	2	13
80	4.3820000	2	12
100	4.6050000	2	12

```

2 ln(u)          1.534e-2   3.859e-4   39.75   <0.0001
3 lot(2)         -7.349e-3   1.672e-3   -4.395   0.0006
4 ln(u).lot(2)  8.255e-3   7.328e-4   11.26   <0.0001

```

Scale is estimated at 2.117e-3 using mean \index{deviance} deviance

Model is ln(u)+lot

\index{deviance} deviance is 0.3007 df 15

```

          estimate se(est)   t ratio   Prob>|t|
1 Constant -2.144e-2  2.211e-3   -9.697   <0.0001
2 ln(u)    1.775e-2  1.034e-3   17.17   <0.0001
3 lot(2)   1.087e-2  1.971e-3    5.513   <0.0001

```

Scale is estimated at 2.005e-2 using mean \index{deviance} deviance

## 0.9 Diagnostic methods

While the fitting of models is fairly routine the selection of the best or even of an appropriate one is not straightforward. We turn our attention to the methods we can use to check how good our fitted models are at describing the data. We start with the assumption that the selected model has been carefully chosen and the type and structure of the data have been taken into account. In the first instance the experimenter will

have examined the distribution in order to select the appropriate error and will also have checked the link  $g(\cdot)$  by plotting the  $g(y_j)$  against the explanatory variables.

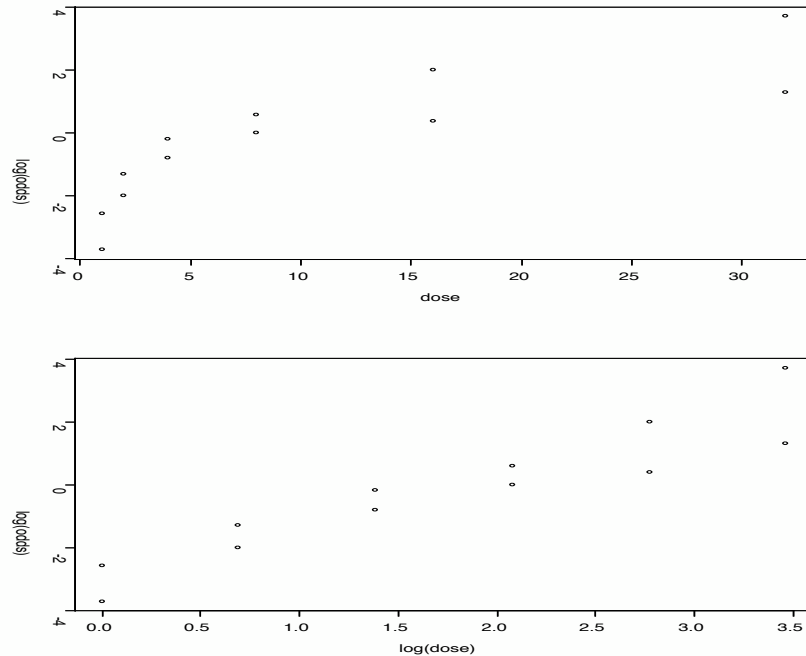


Figure 7: Moths deaths log(odds) by dose

As an example we have considered the moth data. If  $y_j$  is the binomial response from  $n$  we plot the empirical logistic

$$z_j = \log \left( \frac{y_j + 0.5}{n - y_j + 0.5} \right)$$

against the explanatory variable. As we can see this leads us to consider the log of the dose rather than the dose as an explanatory variable.

It is clear that the model is not adequate—we should have taken log dose as is conventional. Curvature here does imply either the wrong link, wrong choice of scale or the omission of a power term in the covariate. Sadly this list is not comprehensive!

For the clotting data we plot the reciprocal since this is the link to give an interesting plot.

*You might like to think how you might distinguish between a gamma and a log normal as the error.*

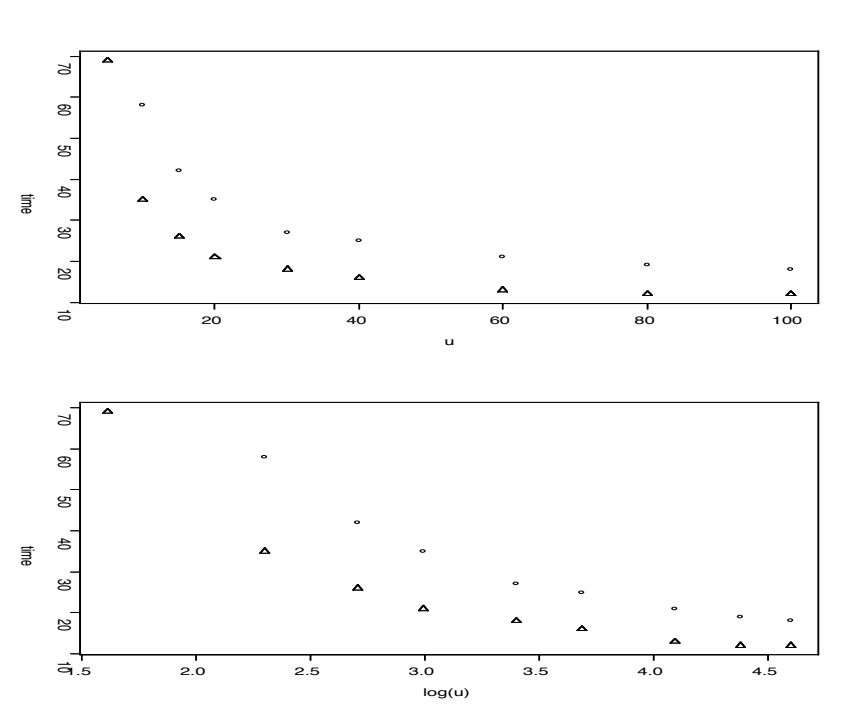


Figure 8: Clotting plot

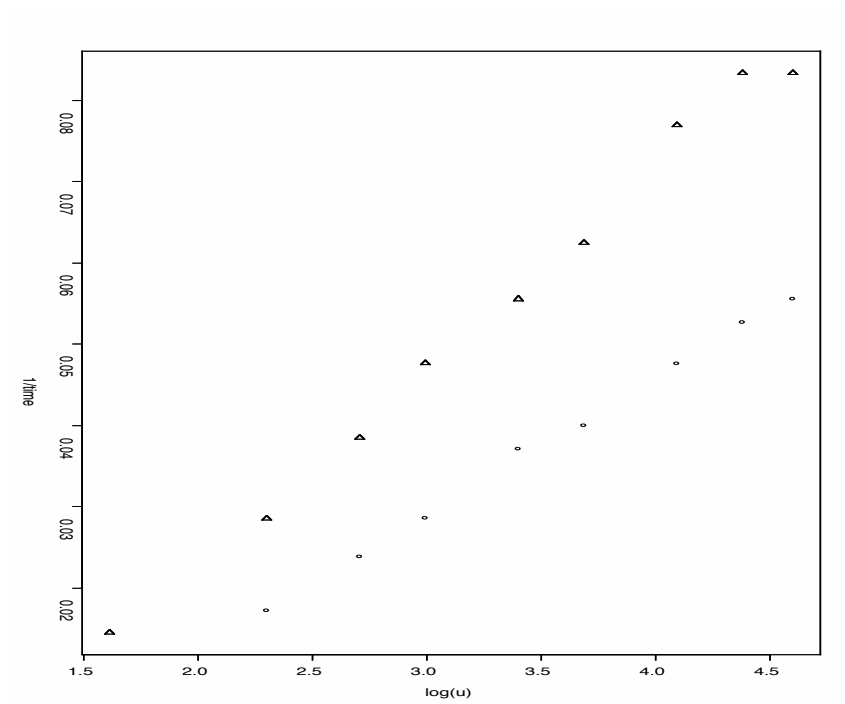


Figure 9: Clotting plot

## 0.10 Overdispersion

### Germination of seeds from Orobanche.

Two varieties of seeds from Orobanche (a parasitic plant growing on the roots of flowering plants)

1. *O. aegyptiaca* 75 and
2. *O. aegyptiaca* 73

were tested for germination if covered with two different root extracts

1. bean and cucumber.

The goal of the experiment was to determine the effect of the root extract on inhibiting the growth of the parasite plants. The data is given in table 12 A simple glm with

O. aegyptiaco 75				O. aegyptiaco 73			
bean		cucumber		bean		cucumber	
germ	total	germ	total	germ	total	germ	total
10	39	5	6	8	16	3	12
23	62	53	74	10	30	22	41
23	81	55	72	8	28	15	30
26	51	32	51	23	45	32	51
17	39	46	79	0	4	3	7
		10	13				

Table 12: Germination of seeds from Orobanche

Binomial error gives

```
> r1<-glm(yy~type+plant,family=binomial)
> anova(r1)
Analysis of \index{deviance} deviance Table
Model: binomial, link: logit
Response: yy
Terms added sequentially (first to last)
      Df \index{deviance} deviance Resid. Df Resid. Dev
NULL                20      98.719
type      1       2.544         19      96.175
plant     1     56.489         18      39.686

> summary(r1)
Call:
glm
Coefficients:
```

```

                Estimate Std. Error z value Pr(>|z|)
(Intercept)  -0.4300      0.1137  -3.781 0.000156 ***
type2        -0.2705      0.1547  -1.748 0.080435 .
plant2       1.0647      0.1442   7.383 1.55e-13 ***
(\index{dispersion parameter} dispersion parameter for binomial family taken to be 1)
Null \index{deviance} deviance: 98.719 on 20 degrees of freedom
Residual \index{deviance} deviance: 39.686 on 18 degrees of freedom
AIC: 122.28

```

but the residual deviance is 39.686 and our estimate of the dispersion is  $39.686/18 = 2.2$ . We look at the more accurate estimate using the Pearson residuals

```

> sigma<-sum(residuals(r1,type="pearson")^2)/18 # use the sum of squares of the Pearson r
> sigma
[1] 2.128368

```

Now

- The plot of the standardized deviance residuals against the factor combinations does not indicate a systematic deficiency of the model.
- In the Q-Q plot, the negative quantiles for the residuals are smaller than the quantiles from the standard normal distribution, but the positive quantiles for the residuals are larger. This indicates that the standardized residuals have a variance larger than 1.

Hence, the observations in the experiment have a larger variance than expected from the binomial model assumption. We can no longer claim, that the observations are binomially distributed. Instead, we will say that we have binomial observations, which expectation is described by the model 1 and which variance is proportional to the variance for binomially distributed data:

$$Var(y) = \phi n \pi (1 - \pi)$$

The dispersion  $\phi$  is the unknown proportionality constant. The estimation of the parameters for the model are independent on the knowledge of  $\phi$  but for the evaluation of *variability* of the estimates we have to estimate  $\phi$ . Using F tests on the predictors we have

```

> drop1(r1,scale=sigma,test="F")
Single term deletions

```

```

Model:
yy ~ type + plant

```

```

scale: 2.128368

```

```

          Df \index{deviance} deviance      AIC F value      Pr(F)
<none>      39.686 122.282
type      1   42.751 121.722  1.3902    0.2537
plant     1   96.175 146.823 25.6214 8.124e-05 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1  1
Warning message:
F test assumes 'quasi-binomial' family in: drop1.glm(r1, scale = sigma, test = "F")

```

As you can see we are using a quasi-binomial family.

Of course we could have done this directly

```

> r2<-glm(yy~type+plant,family=quasi-binomial)
> anova(r2)
Analysis of \index{deviance} deviance Table
Model: quasi-binomial, link: logit
Response: yy
Terms added sequentially (first to last)
          Df \index{deviance} deviance Resid. Df Resid. Dev
NULL              20      98.719
type      1      2.544          19      96.175
plant     1     56.489          18      39.686

> summary(r2)
Call:
glm(formula = yy ~ type + plant, family = quasibinomial)
Coefficients:
          Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.4300    0.1659  -2.592  0.0184 *
type2       -0.2705    0.2257  -1.198  0.2463
plant2      1.0647    0.2104   5.061 8.14e-05 ***

```

```

(\index{dispersion parameter} dispersion parameter for quasibinomial family taken to be 2
Null \index{deviance} deviance: 98.719 on 20 degrees of freedom
Residual \index{deviance} deviance: 39.686 on 18 degrees of freedom
AIC: NA

```

### Animal propulsion

An experimenter is investigating the reaction of a particular organism when exposed to a flow of fluid. He counts the number of reactions made by the organism per successive five second windows to a flow of fluid. There are two kinds of fluid, sea water (S) and sea water with nutrient (D).

S	1	8	21	15	8	4	2	2	3	2
D	4	3	2	2	0	3	3	4	0	0
t	1	2	3	4	5	6	7	8	9	10

He is about to fit a model when he realizes that his timings are out and the S values are in scored 7 second windows while the D values are scored in an 11 second window. To overcome this problem he uses an offset and obtains

```
> r1<-glm(y~fluid+time,family=poisson,offset=-log(off))
>
> summary(r1)
glm(formula = y ~ fluid + time, family = poisson, offset = -log(off))
Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  4.55821    0.21005  21.701 < 2e-16 ***
fluid2       -0.69315    0.25054  -2.767 0.005664 **
time         -0.14799    0.03938  -3.758 0.000171 ***
(\index{dispersion parameter} dispersion parameter for poisson family taken to be 1)
Null \index{deviance} deviance: 77.410 on 19 degrees of freedom
Residual \index{deviance} deviance: 54.001 on 17 degrees of freedom
AIC: 114.16
> anova(r1)
Analysis of \index{deviance} deviance Table
Model: poisson, link: log
Response: y
Terms added sequentially (first to last)
      Df \index{deviance} deviance Resid. Df Resid. Dev
NULL                19      77.410
fluid  1      8.511          18      68.900
time   1     14.899          17      54.001
```

Clearly the residual deviance is large,  $54.001/17 = 3.18$ . Using Pearson we get 3.0 from

```
sigma<-sum(residuals(r1,type="pearson")^2)/17
```

We have assuming overdispersion

```
> drop1(r1,scale=sigma,test="F")
Single term deletions
Model:
y ~ fluid + time
scale:  3.003505
      Df \index{deviance} deviance      AIC F value   Pr(F)
<none>      54.001 114.158
```

```

fluid  1   62.512 114.992  2.6792 0.12004
time   1   68.900 117.118  4.6902 0.04484 *
----
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1  1
Warning message:
F test assumes 'quasipoisson' family in: drop1.glm(r1, scale = sigma, test = "F")

or

> r2 <-glm(y~fluid+time,family=quasipoisson,offset=-log(off))
> summary(r2)
glm(formula = y ~ fluid + time, family = quasipoisson, offset = -log(off))
Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)   4.55821     0.36403  12.522 5.23e-10 ***
fluid2        -0.69315     0.43420  -1.596  0.1288
time          -0.14799     0.06825  -2.168  0.0446 *
(\index{dispersion parameter} dispersion parameter for quasipoisson family taken to be 3
Null \index{deviance} deviance: 77.410  on 19  degrees of freedom
Residual \index{deviance} deviance: 54.001  on 17  degrees of freedom
AIC: NA
> anova(r2)
Analysis of \index{deviance} deviance Table
Model: quasipoisson, link: log
Response: y
Terms added sequentially (first to last)

      Df \index{deviance} deviance Resid. Df Resid. Dev
NULL    19      77.410
fluid  1    8.511      18      68.900
time   1   14.899      17      54.001

```

## 0.11 Residuals

Model checking methods can be either formal or exploratory. The exploratory methods typically use the investigators eye to detect patterns or the lack of pattern. A simple check of overall adequacy of the model is the plot of fitted values against responses. Most other plots are based on residual plots of one kind or another. Glm residual analysis is, as one might expect, a direct extension of normal regression methods. The complication is the *type* of residual. Two kinds of residual are used together with just the raw difference between response  $y$  and predicted  $\hat{y}$

1. The raw residual  $r_j = y_j - \hat{y}_j$

2. The Pearson residual

$$r_j^P = \frac{y_j - \hat{y}_j}{\sqrt{\hat{v}\hat{\sigma}^2[\hat{y}]}}$$

3. The (scaled) deviance residual

$$r_j^D = \text{sgn}[y_j - \hat{y}_j]/\sqrt{S_j}$$

where  $S_j$  is the contribution to the scaled deviance made by the  $j$ th observation with the scale estimated. This is usually standardized rather like the regression residual to give

$$r_j^{DS} = \frac{r_j^D}{\sqrt{1 - h_j}}$$

where the  $h$  term comes from the analogue of the regression  $\mathbf{H}$  matrix.

Note: While we use  $\hat{y}_j$  it has become standard to use  $\hat{\mu}_j$

Nelder points out that no analysis can be considered complete unless the residuals have been plotted against some function of the fitted values. We can either plot the standardized deviance residuals against the fitted or against fitted values on a constant information scale. We digress to explain:

If we transform a random variable  $X$  to get  $Y = H(X)$  a Taylor expansion gives

$$Y = H(X) = H(\mu_x) + (X - \mu_x)H'(\mu_x) + \frac{(X - \mu_x)^2}{2}H''(\mu_x) + \dots$$

Then we can argue that

$$E[Y] = H(\mu_x) + \frac{\sigma_x^2}{2}H''(\mu_x)$$

and

$$\text{var}(Y) = \sigma_x^2[H'(x)]^2$$

If we select a transformation of  $X$  to give a constant variance then  $H'(x) = \text{const}/\sigma_x$  so  $H(\mu) = \int \text{const}/\sigma_x d\mu$ . For example instead of the plotting the standardized residuals against fitted we might use

- Normal -  $\hat{\mu}$
- Poisson -  $2\sqrt{\hat{\mu}}$
- Binomial -  $2\sin^{-1}(\sqrt{\hat{\mu}})$
- Gamma -  $2\hat{\mu}$

Once again we use the moth data as an example.

Plotting the residual against covariates can be informative in much the same way as the above. The other useful plot is the probability plot of the residuals. We would usually seek normality in the residuals and this is a good way of using the brains happy knack of finding patterns.

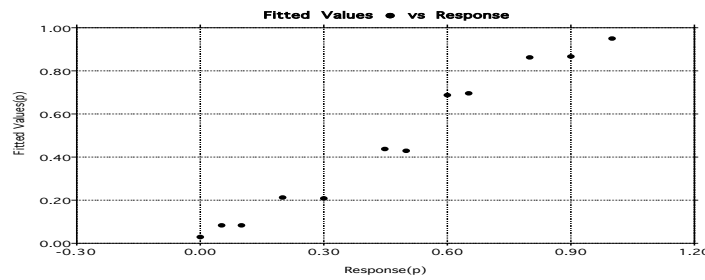


Figure 10: Moths deaths residuals by transformed fitted

If you have very large data sets then of course it is more difficult to get reasonable pictures which are of some use. Of course if you have more data you expect more work but one simple possibility is to sample your data set. This can be arranged to give you reasonably sized subsamples and indeed can give you a family of such subsamples allowing you to look at an envelope of plots!

## 0.12 More complex analysis

We can of course take our regression diagnostics and extend these.

### 0.12.1 Added Variable plots

As in regression the added variable plot gives a check on whether a covariate  $z$  say should be included in the linear predictor. The procedure is as in normal regression first we get the unstandardized residuals for  $z$  as response variable using the same linear predictor (and weights) as for the response  $y$ . The unstandardized residuals for the model fitting to  $y$  are then plotted against the corresponding residuals for  $z$ . If  $z$  is not to be included no trend will present and if you see a trend then include  $z$ . I must confess to some doubts about this as a technique. While I do believe that we need all the help that we can get

we do have other and simpler ways of checking whether a variable is to be included. Nelder mentions added variable plots but I have not seen many examples of their use in glms. I suspect they may be of more use with collections of suspect covariates but the whole procedure becomes less intuitive.

### Leverage and the Cook statistic

In regression the leverage value  $h_j$  is a measure of the remoteness (in the space of explanatory variables) of the  $j$ th observation of the remaining  $N - 1$  responses. They are thus a useful diagnostic when looking for observations which might have undue influence on the fit of the model. Nelder suggests working with a standardized leverage measure but in practice we work with whatever the program gives us, for definitions see Nelder.

Cook developed a statistic for measuring the influence of a data point by considering the deletion residuals which are the residuals obtained by fitting a model with the point in question deleted. If we write  $r_j$  as the residual when the  $j$ th point is deleted we can write the statistic as

$$DC_j = \frac{r_j^2 h_j}{q(1 - h_j)}$$

where  $h_j$  is the leverage and  $q$  the number of parameters in the model. There are variants, for example a modified Cook statistic

$$C_j = |r_j^*| \sqrt{\left(\frac{N - q}{q}\right) \left(\frac{h_j}{1 - h_j}\right)}$$

where

$$r_j^* = r_j / \sqrt{\frac{N - q - r_j^2}{N - q - 1}}$$

Whatever we use as the definition we can plot the Cook's distance for each point against the rank of the ordered observations. The model fits the observations with large distances least well. Values less than one can be accepted it is only the larger ones that might cause concern.

While plotting one of our statistics above against case number can be illuminating Nelder suggests making normal or half normal plots of the statistic. While it would be too much to expect linearity it is a good way of finding the extremes.

**Part II**

**Categorical Data**

We have discussed categorical data in previous sessions, in Binomial error distributions - see the moth example. Here we continue our examination of generalized modelling by looking specifically at categorical variables. We begin with the simplest case where the outcome has only two outcomes, in other words back to the Binomial

### 0.12.2 Binomial problems

Binomial errors structures are common and are often imposed on the data to make examination simpler. For example the Bernoulli distribution

$$P[Z = 1] = \pi \quad P[Z = 0] = 1 - \pi$$

is commonly used as a counting variable. Thus for example we might check a batch of items for defectives. Let  $Z_j = 1$  if the  $j$ th item is defective and zero otherwise. Then a quantity of interest is  $Y = \sum_{j=1}^N Z_j$ . The likelihood for the  $Z$ 's is just

$$\prod_{j=1}^N \pi^{z_j} (1 - \pi)^{1-z_j} \text{ or } \pi^{\sum z_j} (1 - \pi)^{N - \sum z_j}$$

As this belongs to the exponential family, as does the version with variable probabilities we can use generalized linear modelling ideas. You will of course notice that the sum  $Y$  is Binomial

$$P[Y = y] = \binom{N}{y} \pi^y (1 - \pi)^{N-y}$$

a distribution that also belongs to the exponential family. We often encounter problems where the response is two valued suggesting a binomial error. The typical Binomial data set takes the form

sample	1	2	...	N
successes	$y_1$	$y_2$	...	$y_N$
fails	$n_1 - y_1$	$n_2 - y_2$	...	$n_N - y_N$
Totals	$n_1$	$n_2$	...	$n_N$

with perhaps some explanatory variables. In this case we have  $N$  binomials and the likelihood is

$$\prod_{j=1}^N \binom{n_j}{y_j} \pi_j^{y_j} (1 - \pi_j)^{n_j - y_j}$$

The deviance between two models is then

$$D = 2 \sum_{j=1}^N \left[ y_j \log\left(\frac{\hat{\pi}_j}{\tilde{\pi}_j}\right) - (n_j - y_j) \log\left(\frac{n_j - \hat{\pi}_j}{n_j - \tilde{\pi}_j}\right) \right]$$

where  $\hat{\pi}_j$  and  $\tilde{\pi}_j$  are the maximum likelihood estimates. For the saturated model each binomial has a different parameter and so the individual maximum likelihood estimates

are then  $y_j/n_j$ . This gives a simple expression for the deviance which we can think of as

$$D = 2 \sum_{j=1}^N O_j \log \left( \frac{O_j}{e_j} \right)$$

where  $O_j$  are the observed values in the tables while the  $e_j$  are the expected values under the model. This is equivalent to

$$D = \sum_{j=1}^N \frac{(O_j - e_j)^2}{e_j}$$

the expression commonly seen in texts.

In the case of the Binomial we have the logit or logistic as the canonical link

$$\log \left( \frac{\pi_j}{1 - \pi_j} \right) = \mathbf{x}^T \boldsymbol{\beta} \quad (9)$$

You will notice that here the natural parameter is now  $\frac{\pi_j}{1 - \pi_j}$  the *odds ratio*. Other possibilities are, as we have mentioned the probit

$$\Phi^{-1}(\pi_j) = \mathbf{x}^T \boldsymbol{\beta} \quad (10)$$

and the complementary log-log

$$\log(-\log(1 - \pi)) = \mathbf{x}^T \boldsymbol{\beta} \quad (11)$$

Much of the interest in these Binomial regression models was driven by people working in Bioassay. They wanted to study survival rates, so for example they might wish to predict the number of deaths for some given doses of insecticide. You will come across phrases like the *median lethal dose LD(50)*.

To put these ideas into context we consider some simple examples

### Aircraft fasteners

The data set gives the number of fasteners in a sample that have failed in service. This failure rate is probably related to load ( in psi)

LOAD	NUM	FAILS	ratio
2500	50	10	0.20
2700	70	17	0.24
2900	100	30	0.30
3100	60	21	0.35
3300	40	18	0.45
3500	85	43	0.51
3700	90	54	0.60
3900	50	33	0.66
4100	80	60	0.75
4300	65	51	0.78

Fitting a model with Binomial error (logit link) in R gives the following

```
Call: glm(formula = yy ~ load, family = binomial)
```

Coefficients:

	Value	Std. Error	t value
(Intercept)	-3.3791612558	0.4521455185	-7.473614
load	0.0007545691	0.0001252397	6.024997

(\index{dispersion parameter} dispersion parameter for Binomial family taken to be 1 )

Null \index{deviance} deviance: 39.03219 on 9 degrees of freedom

Residual \index{deviance} deviance: 0.9283085 on 8 degrees of freedom

Clearly pretty convincing evidence of a load Load effect.

*What do you think?*

order	fraction	Fitted fraction	number	fitted number
1	0.2000000	0.1871513	10	9.357566
2	0.2428571	0.2388598	17	16.720187
3	0.3000000	0.2995894	30	29.958938
4	0.3500000	0.3682883	21	22.097295
5	0.4500000	0.4427816	18	17.711265
6	0.5058824	0.5199410	43	44.194982
7	0.6000000	0.5961606	54	53.654456
8	0.6600000	0.6680059	33	33.400293
9	0.7500000	0.7327982	60	58.623859
10	0.7846154	0.7889409	51	51.281157

It is important to remember you may well have few observations and the deviance statistics can be quite unreliable. It is therefore very important to study the residuals and to plot at the very least the residuals against the fitted values. It may well be useful to plot against the transformed fitted values using the  $2\sin^{-1}(\sqrt{y})$  transform introduced earlier. Here the question is the relation or otherwise between the classification variables. In this case it is fairly easy to see that there probably is a relation but there can be several classification variables giving very complex tables. The following is a three way example.

### Tetanus example

Patients suffering from tetanus are classified as having a severe or less severe illness. Some patients were treated with an antitoxin while others were not. The response is live or die. The results are given below

	Severe		Less Severe	
Anti-Toxin	No	Yes	No	Yes
Deaths	22	15	7	5
Survivors	4	6	5	15
Odds	5.5	2.5	1.4	0.33

Again a Binomial model seems appropriate we have

Model is anti+severe

Scaled \index{deviance} deviance is 0.3677 df 1

	estimate	se(est)	z ratio	Prob> z
1 Constant	1.884	0.4868	3.869	0.0001
2 anti(2)	-1.096	0.5340	-2.053	0.0401
3 severe(2)	-1.739	0.5258	-3.308	0.0009

Scale is fixed at 1.000

	odds ratio	95% confidence interval
1 Constant	6.577	(2.533,17.07)
2 anti(2)	0.3341	(0.1173,0.9514)
3 severe(2)	0.1756	(6.266e-2,0.4922)

	actual	estimate	residual
1	22.00	22.57	-0.3222
2	15.00	14.43	0.2698
3	7.000	6.432	0.3301
4	5.000	5.568	-0.2867

**Simpsons Paradox example**

You may recall a similar table we discussed when reviewing Simpson’s paradox Here is the more detailed data set.

	Good Condition			Bad Condition		
	Hospital A	Hospital B	Total	Hospital A	Hospital B	Total
Died	6	8	14	577	8	65
Survived	594	592	1186	1443	192	1635
Total	600	600	1200	1500	200	1700
Death Rate	0.010	0.013	0.016	0.038	0.040	0.038

An analysis gives

Model is hospital+condition

Scaled \index{deviance} deviance is 20.94 df 1

	estimate	se(est)	z ratio	Prob> z
1 Constant	-3.858	0.2710	-14.24	<0.0001

```
2 hospital(2)  -2.068      0.2644      -7.824      <0.0001
3 condition(2)  3.370      0.2750       12.25      <0.0001
Scale is fixed at 1.000
```

```

                odds ratio      95% confidence interval
1 Constant      2.111e-2      (1.241e-2,3.590e-2)
2 hospital(2)   0.1264      (7.529e-2,0.2122)
3 condition(2)  29.08      (16.96,49.86)
  actual      estimate  residual
1      6.000      12.40    -2.040
2      8.000       1.597     3.612
3     577.0      570.6     0.3403
4      8.000      14.40    -1.902
```

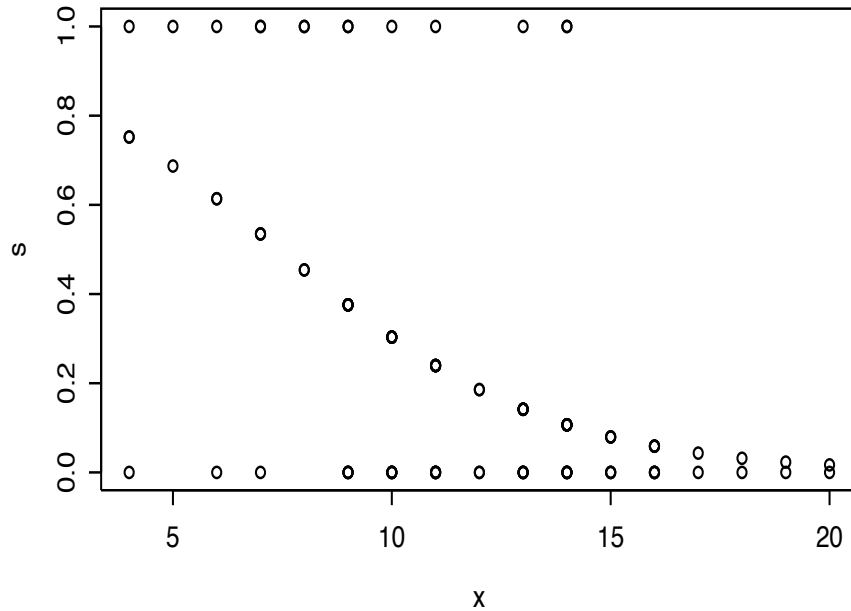
### 0.12.3 Senility and WAIS

We consider the following data set from Dobson. A sample of elderly people we examined for signs of senility  $s$ . the response  $s$  was coded 1 if symptoms were present and zero otherwise. At the same time the score on the WAIS IQ scale was recorded. The data is given below, we fit the basic logistic model

x	s	x	s	x	s	x	s	x	s	x	s
9	1	7	1	11	0	13	0	9	0	13	0
13	1	9	1	14	0	15	0	11	0	9	0
6	1	7	1	15	0	13	0	14	0	15	0
8	1	5	1	18	0	10	0	10	0	10	0
10	1	14	1	7	0	11	0	16	0	11	0
4	1	13	0	16	0	6	0	10	0	12	0
14	1	16	0	9	0	17	0	16	0	4	0
8	1	10	0	9	0	14	0	14	0	14	0
11	1	12	0	11	0	19	0	13	0	20	0

```
> y<-cbind(s,1-s)
> b1<-glm(y~x,family=binomial)
> summary(b1)
glm(formula = y ~ x, family = binomial)
Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)    2.4040     1.1918   2.017  0.04369 *
x              -0.3235     0.1140  -2.838  0.00453 **
(\index{dispersion parameter} dispersion parameter for binomial family taken to be 1)
Null \index{deviance} deviance: 61.806 on 53 degrees of freedom
Residual \index{deviance} deviance: 51.017 on 52 degrees of freedom
AIC: 55.017
```

Straightforward plots may not be informative, as you can see from the plot below



#### 0.12.4 The Hosmer-Lemeshow Goodness-of-Fit Test

We need enough replication within subpopulations to make the Pearson and deviance goodness-of-fit tests valid. When there are one or more continuous predictors in the model, the data are often too sparse to use these statistics as in the case above. Hosmer and Lemeshow proposed a statistic that they show, through simulation, is distributed as chi-square when there is no replication in any of the subpopulations. This test is only available for binary response models.

1. The observations are sorted in increasing order of their estimated event probability.
2. The observations are then divided into approximately ten groups according to the following scheme. Suppose  $N$  is the total number of subjects and let  $M$  be the target number of subjects where  $M = (0.1N+0.5)$  rounded to the nearest smaller integer.
3. We aim to have the largest  $M$  values in the first group, the remaining largest in the next group and so on. We are happy to conflate tiny groups .

The Hosmer-Lemeshow goodness-of-fit statistic is obtained by calculating the Pearson chi-square statistic from the  $2 \times g$  table of observed and expected frequencies, where

$g$  is the number of groups. We do not have to have ten groups, it is nice if we can be all is not lost otherwise.

The ordered predictions are given below for the senility data.

$y$	$\hat{\pi}$	$y$	$\hat{\pi}$	$y$	$\hat{\pi}$
0	0.01684746	1	0.14162581	0	0.3033786
0	0.02313427	0	0.14162581	0	0.3033786
0	0.03169146	0	0.14162581	1	0.37572578
0	0.04327366	0	0.14162581	1	0.37572578
0	0.05883161	0	0.14162581	0	0.37572578
0	0.05883161	0	0.14162581	0	0.37572578
0	0.05883161	0	0.18568111	0	0.37572578
0	0.05883161	0	0.18568111	0	0.37572578
0	0.07951811	1	0.23961509	1	0.45407982
0	0.07951811	0	0.23961509	1	0.45407982
0	0.07951811	0	0.23961509	1	0.53477641
1	0.10665418	0	0.23961509	1	0.53477641
1	0.10665418	0	0.23961509	0	0.53477641
0	0.10665418	0	0.23961509	1	0.61369267
0	0.10665418	1	0.3033786	0	0.61369267
0	0.10665418	0	0.3033786	1	0.68705597
0	0.10665418	0	0.3033786	1	0.75211453
0	0.10665418	0	0.3033786	0	0.75211453

Taking the range as approximately 0-0.8 we have 10 groups with upper bound at 0.08,0.16, etc with, we expect 4 per group. I will leave you to sort out the details!

### 0.12.5 An agricultural example

Phelps helped design an experiment to predict the proportions of damaged carrots in an insecticide field experiment. The carrots were planted in 3 Blocks and varying levels of insecticide we used as can be seen below.

Dose		Block		
level $j$	dose $x_j$	1	2	2
1	1.52	10/35	7/38	10/34
2	1.64	16/42	10/40	10/38
3	1.72	8/50	8/33	5/36
4	1.88	6/42	8/37	3/35
5	2.00	9/35	5/47	2/49
6	2.12	9/42	17/42	1/40
7	2.24	1/32	6/35	3/22
8	2.36	2/28	4/35	2/31

When we fit the most complex model we have:

```
> p1<-glm(y~dose*block,family=binomial)
> summary(p1)
```

Call:

```
glm(formula = y ~ dose * block, family = binomial)
```

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	2.0099	1.0834	1.855	0.0636 .
dose	-1.8181	0.5855	-3.105	0.0019 **
block2	-2.8462	1.4711	-1.935	0.0530 .
block3	1.3764	1.7848	0.771	0.4406
dose:block2	1.5702	0.7780	2.018	0.0436 *
dose:block3	-1.0621	0.9829	-1.081	0.2799

(\index{dispersion parameter} dispersion parameter for binomial family taken to be 1)

Null \index{deviance} deviance: 69.860 on 23 degrees of freedom  
 Residual \index{deviance} deviance: 34.902 on 18 degrees of freedom  
 AIC: 127.04

Analysis of \index{deviance} deviance Table

	Df	\index{deviance} deviance	Resid. Df	Resid. Dev
NULL		69.860	23	69.860
dose	1	16.446	22	53.414
block	2	9.186	20	44.229
dose:block	2	9.327	18	34.902

I can of course change the parameterization as in

Call:

```
glm(formula = y ~ block + dose:block - 1, family = binomial)
```

Coefficients:

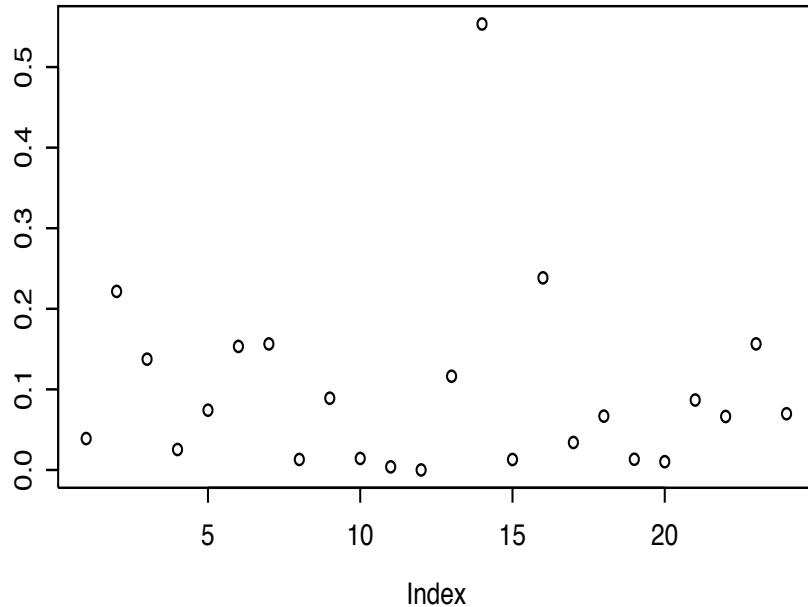
	Estimate	Std. Error	z value	Pr(> z )
block1	2.0099	1.0834	1.855	0.063567 .
block2	-0.8363	0.9951	-0.840	0.400695
block3	3.3863	1.4184	2.387	0.016965 *
block1:dose	-1.8181	0.5855	-3.105	0.001900 **
block2:dose	-0.2480	0.5123	-0.484	0.628348
block3:dose	-2.8802	0.7895	-3.648	0.000264 ***

(\index{dispersion parameter} dispersion parameter for binomial family taken to be 1)

Null \index{deviance} deviance: 467.039 on 24 degrees of freedom  
 Residual \index{deviance} deviance: 34.902 on 18 degrees of freedom  
 AIC: 127.04

We can use the same sorts of diagnostics as we used in regression.

Cooks distances



Here is a reparameterized model

Call:

```
glm(formula = y ~ block + block:dose, family = binomial)
```

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )	
(Intercept)	2.0099	1.0834	1.855	0.063567	.
block2	-2.8462	1.4711	-1.935	0.053016	.
block3	1.3764	1.7848	0.771	0.440592	
block1:dose	-1.8181	0.5855	-3.105	0.001900	**
block2:dose	-0.2480	0.5123	-0.484	0.628348	
block3:dose	-2.8802	0.7895	-3.648	0.000264	***

(\index{dispersion parameter} dispersion parameter for binomial family taken to be 1)

Null \index{deviance} deviance: 69.860 on 23 degrees of freedom  
 Residual \index{deviance} deviance: 34.902 on 18 degrees of freedom  
 AIC: 127.04

Number of Fisher Scoring iterations: 5

### 0.12.6 Ante-Natal Clinic

Consider the following data on the mortality of babies whose mothers attended ante-natal clinics

Attendance length	less than 1 month	more than 1 month	total
clinic A	3/179	4/297	7/476
clinic B	17/214	4/25	21/239
total	20/393	8/222	

The odds of dying are

Attendance length	less than 1 month	more than 1 month	total
clinic A	1.705	1.365	1.492
clinic B	8.629	8.626	9.633
total	5.362	3.3738	

Analysis shows

Model is clinic+attend

Scaled \index{deviance} deviance is 4.326e-2(0.000) df 1(0)

	estimate	se(est)	z ratio	Prob> z
1 Constant	-4.137	0.5077	-8.149	<0.0001
2 clinic(2)	1.699	0.5307	3.202	0.0014
3 attend(2)	-0.1104	0.5610	-0.1967	0.8440

Scale is fixed at 1.000

	odds ratio	95% confidence interval
1 Constant	1.597e-2	(5.903e-3,4.319e-2)
2 clinic(2)	5.469	(1.933,15.47)
3 attend(2)	0.8955	(0.2982,2.689)

	actual	estimate	residual
1	3.000	2.813	0.1123
2	4.000	4.187	-9.194e-2
3	17.00	17.19	-4.698e-2
4	2.000	1.813	0.1440

### 0.12.7 Tetanus example

Patients suffering from tetanus are classified as having a severe or less severe illness. Some patients were treated with an antitoxin while others were not. The response is live or die. The results are given below

	Severe		Less Severe	
Anti-Toxin	No	Yes	No	Yes
Deaths	22	15	7	5
Survivors	4	6	5	15
Odds	5.5	2.5	1.4	0.33

Here we have two factors `severity` and `anti`. The best non-saturated model is

`severity+anti`

whose deviance is 0.36773 on 1 degrees of freedom

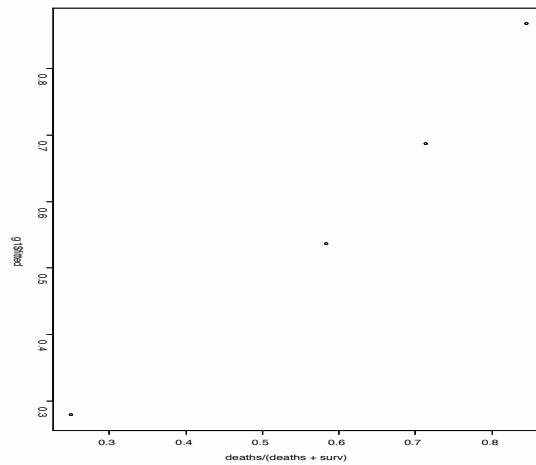
This clearly implies that there is no interaction effect but that severity is related to death as is the administration of an antidote. See below

`severity+anti`

Analysis gives :

```
> deaths<-c(22,15,7,5) # Input deaths
> surv<-c(4,6,5,15) # survivors
> odds<-deaths/surv
> odds
[1] 5.500000 2.500000 1.400000 0.333333
# odds ratios
> severity<-factor(c(1,1,2,2))
# set up severity and declare it a factor
> anti<-factor(c(1,2,1,2))
# set up anti and declare it a factor
> yy<-cbind(deaths,surv)
#WARNING need to give both the binomial outcomes
> g1<-glm(yy~severity+anti,family=binomial)
# fit the most complex non-saturated model
> summary(g1)
Call:
glm(formula = yy ~ severity + anti, family = binomial)
\index{deviance} deviance Residuals:
[1] -0.3222 0.2698 0.3301 -0.2867
Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.8835     0.4868   3.869 0.000109 ***
severity2    -1.7394     0.5258  -3.308 0.000940 ***
anti2        -1.0964     0.5340  -2.053 0.040048 *
(\index{dispersion parameter} dispersion parameter for binomial family taken to be 1)
Null \index{deviance} deviance: 18.65636 on 3 degrees of freedom
Residual \index{deviance} deviance: 0.36773 on 1 degrees of freedom
AIC: 18.939
```

```
Number of Fisher Scoring iterations: 3
> g1$resid
[1] -0.1908264  0.1259299  0.1904673 -0.1414706
> g1$fitted
[1] 0.8680172 0.6872168 0.5359627 0.2784224
> deaths/(deaths+surv)
[1] 0.8461538 0.7142857 0.5833333 0.2500000
> plot(deaths/(deaths+surv),g1$fitted)
```



Complex tables result, as you might expect, in complex analysis - there really is no such thing as a free lunch.

### 0.13 Survival of breast cancer patients

The data in the table are from a study of the survival of breast cancer patients (see Morrison et al. [1973]). The variables and categories are

- Variable 1. Degree of chronic inflammatory reaction
  1. Minimal
  2. Moderate severe
- Variable 2. Nuclear grade
  1. Relatively malignant appearance
  2. Relatively benign appearance
- Variable 3. Survival for three years
  1. No
  2. Yes
- Variable 4. Age of diagnosis
  1. Under 50 years
  2. 50-69 years
  3. 70 or older
- Variable 5. Center where patient was diagnosed
  1. Tokyo
  2. Boston
  3. Glamorgan

Center	Age	Survived	Minimal Inflammation		Greater Inflammation	
			Malignant	Benign	Malignant	Benign
Tokyo	Under 50	No	9	7	4	3
		Yes	26	68	25	9
	50-69	No	9	9	11	2
		Yes	20	46	18	5
	70 or over	No	2	3	1	0
		Yes	1	6	5	1
Boston	Under 50	No	6	7	6	0
		Yes	11	24	4	0
	50-69	No	8	20	3	2
		Yes	18	58	10	3
	70 or over	No	9	19	3	0
		Yes	15	26	1	1
Glamorgan	Under 50	No	16	7	3	0
		Yes	16	20	8	1
	50-69	No	14	12	3	0
		Yes	27	39	to	4
	70 or over	No	3	7	3	0
		Yes	12	11	4	1

Of course the output (shown in part below) is complex but almost tractable

```

Model is age+center+malig+inflam+age.center+age.malig
+age.inflam+center.malig+center.inflam+inflam.malig
+age.center.malig+age.center.inflam+age.inflam.malig
+center.inflam.malig
Scaled \index{deviance} deviance is 2.509e-4 df 3
Failure to converge in specified iterations

```

		estimate	se(est)	z ratio	Prob
1	Constant	1.061	0.3867	2.743	0.0061
2	age(2)	-0.2624	0.5574	-0.4707	0.6379
3	age(3)	-1.754	1.284	-1.366	0.1721
4	center(2)	-0.4547	0.6381	-0.7127	0.4761
5	center(3)	-1.061	0.5240	-2.025	0.0429
6	malig(2)	1.213	0.5542	2.188	0.0287
7	inflam(2)	0.7717	0.6630	1.164	0.2445
8	age(2).center(2)	0.4672	0.8653	0.5399	0.5893
9	age(2).center(3)	0.9191	0.7377	1.246	0.2128
10	age(3).center(2)	1.659	1.444	1.149	0.2507
11	age(3).center(3)	3.140	1.480	2.121	0.0339
12	age(2).malig(2)	-0.3798	0.7753	-0.4899	0.6242
13	age(3).malig(2)	0.1736	1.519	0.1143	0.9090
14	age(2).inflam(2)	-1.078	0.8644	-1.247	0.2125
15	age(3).inflam(2)	1.531	1.772	0.8640	0.3876
16	center(2).malig(2)	-0.5867	0.8656	-0.6778	0.4979
17	center(3).malig(2)	-0.1629	0.7906	-0.2061	0.8368
18	center(2).inflam(2)	-1.783	1.055	-1.690	0.0911
19	center(3).inflam(2)	0.2091	1.011	0.2068	0.8362
20	inflam(2).malig(2)	-1.947	1.021	-1.907	0.0565
21	age(2).center(2).malig(2)	7.588e-3	1.136	6.679e-3	0.9947
22	age(2).center(3).malig(2)	-0.1481	1.066	-0.1390	0.8895
23	age(3).center(2).malig(2)	-0.9427	1.738	-0.5424	0.5876
24	age(3).center(3).malig(2)	-2.158	1.810	-1.192	0.2332
25	age(2).center(2).inflam(2)	2.482	1.427	1.740	0.0819
26	age(2).center(3).inflam(2)	0.6441	1.368	0.4707	0.6378
27	age(3).center(2).inflam(2)	-2.129	2.308	-0.9225	0.3563
28	age(3).center(3).inflam(2)	-3.610	2.173	-1.661	0.0967
29	age(2).inflam(2).malig(2)	1.538	1.477	1.041	0.2979
30	age(3).inflam(2).malig(2)	13.49	75.78	0.1781	0.8587
31	center(2).inflam(2).malig(2)	-0.6432	1.629	-0.3948	0.6930
32	center(3).inflam(2).malig	10.41	78.43	0.1328	0.8944

Kill the 3 way interactions to get

```
Model is age+center+malig+inflam+age.center+age.malig
+age.inflam+center.malig+center.inflam+inflam.malig
Scaled \index{deviance} deviance is 18.56(+18.56) df 15(+12)
Scale is fixed at 1.000
```

		estimate	se(est)	z ratio	Prob —z—
1	Constant	1.219	0.3408	3.577	0.0003
2	age(2)	-0.4909	0.4190	-1.172	0.2414
3	age(3)	-0.5059	0.6940	-0.7290	0.4660
4	center(2)	-0.7576	0.4663	-1.624	0.1043
5	center(3)	-1.101	0.4242	-2.594	0.0095
6	malig(2)	1.007	0.4085	2.464	0.0137
7	inflam(2)	6.338e-2	0.4453	0.1423	0.8868
8	age(2).center(2)	0.8810	0.4789	1.840	0.0658
9	age(2).center(3)	1.037	0.4686	2.212	0.0269
10	age(3).center(2)	0.5396	0.6894	0.7827	0.4338
11	age(3).center(3)	1.031	0.7105	1.451	0.1468
12	age(2).malig(2)	-0.3171	0.4142	-0.7657	0.4439
13	age(3).malig(2)	-0.7925	0.5248	-1.510	0.1310
14	age(2).inflam(2)	-2.374e-2	0.4958	-4.787e-2	0.9618
15	age(3).inflam(2)	-0.1567	0.7001	-0.2237	0.8230
16	center(2).malig(2)	-0.3715	0.4676	-0.7946	0.4269
17	center(3).malig(2)	-0.1361	0.4605	-0.2955	0.7676
18	center(2).inflam(2)	-0.3910	0.5474	-0.7142	0.4751
19	center(3).inflam(2)	0.6105	0.5566	1.097	0.2727
20	inflam(2).malig(2)	-0.2727	0.5413	-0.5037	0.6145

Can do better still

Model is age+center+malig+inflam

Scaled \index{deviance} deviance is 33.10(+14.53) df 28(+13)

		estimate	se(est)	z ratio	Prob
1	Constant	1.123	0.2103	5.339	0.0001
2	age(2)	-4.854e-2	0.1900	-0.2555	0.7984
3	age(3)	-0.4240	0.2406	-1.762	0.0781
4	center(2)	-0.5568	0.2115	-2.632	0.0085
5	center(3)	-0.4363	0.2136	-2.042	0.0411
6	malig(2)	0.5224	0.1772	2.948	0.0032
7	inflam(2)	1.776e-2	0.2199	8.077e-2	0.9356

Scale is fixed at 1.000

## 0.14 Coma Patients

The following four-way table displays data arising from a study to determine indication of survival following a severe head injury. The factors are

- A: outcome after six months
  1. dead
  2. alive
- B Age:
  1. 10-25 years
  2. 26-45 years
  3. over 45 years
- C Coma severity (first 24 hours of injury)
  1. deep
  2. moderate
  3. light
- D Intercranial haematoma:
  1. absent
  2. present

		D					
		1			2		
		C			C		
B	A	1	2	3	1	2	3
1	1	48	29	7	25	23	8
2	2	25	98	58	7	30	27
3	1	27	22	4	29	33	8
1	2	3	36	32	8	26	25
2	1	34	27	12	59	73	19
3	2	4	4	17	8	23	30

*Analyze the head injury data set as a Binomial model*

Of course we may have categorical responses with more than two outcomes. These are traditionally regarded as multinomial problems and this is our next topic.

## 0.15 contingency tables

One very useful application of generalized linear models is to contingency tables. These are tables of counts which are classified by variables. As an illustration, a data set used by Bishop, Fienberg, and Holland examines the relationship between occupational classifications of fathers and sons. The classes are

Label	Description
A	Professional, High Administrative
S	Managerial, Executive, High Supervisory
I	Low Inspectional, Supervisory
N	Routine Nonmanual, Skilled Manual
U	Semi- and Unskilled Manual

The counts are given by

Father	Son				
	A	S	I	N	U
A	50	45	8	18	8
S	28	174	84	154	55
I	11	78	110	223	96
N	14	150	185	714	447
U	3	42	72	320	411

The following four-way table displays data arising from a study to determine indication of survival following a severe head injury.

The factors are

A Outcome after six months: 1 dead - 2 alive

B Age: 1 10-25 years - 2 26-45 years - 3 over 45 years

C Coma severity (first 24 hours of injury): 1 deep - 2 moderate - 3 light

D Intercranial haematoma: 1 absent - 2 present

		D					
		1			2		
		C			C		
		A	1	2	3	1	2
1	1	48	29	7	25	23	8
2	2	25	98	58	7	30	27
3	1	27	22	4	29	33	8
1	2	3	36	32	8	26	25
2	1	34	27	12	59	73	19
3	2	4	4	17	8	23	30

## 0.16 The loglinear model

We consider ( for simplicity) the 2 way table. Suppose we have an  $r \times c$  table and the value in the  $ij$ th cell of the table is  $y_{ij}$ . One possibility is to assume that the cells are independent and the number of items in a cell has a Poisson distribution

$$f(y_{ij}) = \frac{\lambda_{ij}^{y_{ij}}}{y_{ij}!} e^{-\lambda_{ij}}$$

This would be simple but the cell entries are usually not independent as the total number of observations  $N$  is fixed a priori. This makes like a little more difficult as we have to compute the likelihood *conditional* on a given  $N$ . If we work out the likelihood conditional on the total we end up with a ( multinomial ) likelihood for the table of the form

$$\ell(\theta) = N! \prod_{ij} \frac{\theta_{ij}^{y_{ij}}}{y_{ij}!}$$

where  $\theta_{ij}$  is the probability of an observation falling in the in the  $ij$ th cell of the table.

There is another possible constraint on such tables. We may fix either the row or column totals in advance. For example

	non-user	user	total
A	56	54	100
B	67	33	100

The constraint in this case gives another form of multinomial, the product multinomial

$$\ell(\theta) = \prod_i y_i! \prod_j \frac{\theta_{ij}^{y_{ij}}}{y_{ij}!}$$

Neither of these distributions belong to the exponential family however Birch showed that we can get maximum likelihood estimates for the parameters by using a Poisson distribution for the counts and a (canonical) log link.

For the two way table the saturated models is

$$\log(E[y_{ij}]) = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij}$$

If we assume independence,

$$\begin{aligned} P[\text{an observation in cell (ij)}] &= \theta_{ij} = \theta_i \cdot \theta_j \\ &= P[\text{an observation in row i}]P[\text{an observation in column j}] \end{aligned}$$

or

$$\log(N\theta_{ij}) = \log N + \log\theta_i + \log\theta_j$$

this implies the  $(\alpha\beta)_{ij}$  term in our Poisson model is zero.

This idea generalizes to multiway tables, the interactions corresponding to dependence between marginals. In consequence we are able to analyze such tables.

### Two way table

For a two way table with *free margins* we have as above

$$P[\text{fall in } jk] = \theta_{jk}$$

so the hypothesis of independence is

$$\theta_{jk} = \theta_{j.}\theta_{.k}$$

and for the loglinear model

$$\log E[y_{jk}] = \lambda_{jk} = \mu + \alpha_j + \beta_k + (\alpha\beta)_{jk}$$

this implies no interaction. We fit the most complex non-saturated Poisson model to the Haberman data

$$\log E[y_{jk}] = \lambda_{jk} = \mu + \alpha_j + \beta_k$$

and obtain a scaled deviance is 792.2 with 16 degrees of freedom. This is very large compared with the saturated model and we conclude that an interaction term is required. This means that we do not have independence between the row and column classifications.

If a margin is fixed, say the row sums  $y_{j.}$  are set a priori, then in terms of the multinomial probabilities we might think that

$$\theta_{jk} = y_{j.}\theta_{.k}$$

. Again in log-linear terms we are investigating the presence or absence of an interaction since

$$\log \theta_{jk} = \log y_{j.} + \log \theta_{.k}$$

### A fixed margin example

Suppose we take samples of 90 from 12 batches and examine the samples for defectives. The results obtained are given below

Sample	1	2	3	4	5	6	7	8	9	10	11	12
Defectives	19	6	9	18	15	13	14	15	16	20	22	14
Non-defective	71	84	91	72	75	77	76	75	74	70	68	76
Total	90	90	90	90	90	90	90	90	90	90	90	90
ratio	0.21	0.07	0.10	0.20	0.17	0.14	0.16	0.17	0.18	0.22	0.24	0.16

While this is clearly a set of Binomial experiments we can also regard it as a contingency table with one fixed margin giving.

Call:

```
glm(formula = yy ~ 1, family = binomial)
```

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	1.61385	0.08139	19.83	<2e-16 ***

(\index{dispersion parameter} dispersion parameter for binomial family taken to be 1)

Null \index{deviance} deviance: 20.801 on 11 degrees of freedom  
 Residual \index{deviance} deviance: 20.801 on 11 degrees of freedom  
 AIC: 74.742

Fitting log-linear models we have

Call:

```
glm(formula = y ~ row + col, family = poisson)
```

\index{deviance} deviance Residuals:

	Min	1Q	Median	3Q	Max
	-2.634110	-0.384397	0.003938	0.393695	1.703917

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	4.318e+00	1.063e-01	40.634	<2e-16 ***
row2	-1.614e+00	8.139e-02	-19.828	<2e-16 ***
col2	4.665e-13	1.491e-01	3.13e-12	1.000
col3	1.054e-01	1.453e-01	0.725	0.468
col4	3.348e-15	1.491e-01	2.25e-14	1.000
col5	3.614e-15	1.491e-01	2.42e-14	1.000
col6	2.180e-15	1.491e-01	1.46e-14	1.000
col7	3.415e-15	1.491e-01	2.29e-14	1.000
col8	3.598e-15	1.491e-01	2.41e-14	1.000
col9	3.114e-15	1.491e-01	2.09e-14	1.000
col10	3.565e-15	1.491e-01	2.39e-14	1.000
col11	7.859e-15	1.491e-01	5.27e-14	1.000
col12	3.456e-15	1.491e-01	2.32e-14	1.000

Null \index{deviance} deviance: 552.767 on 23 degrees of freedom  
 Residual \index{deviance} deviance: 20.801 on 11 degrees of freedom  
 AIC: 174.92

We can consider the tetanus data set

---

	Severe		Less Severe	
Anti-Toxin	No	Yes	No	Yes
Deaths	22	15	7	5
Survivors	4	6	5	15
Odds	5.5	2.5	1.4	0.33

using a Poisson model.

Here we have three factors and the best non-saturated model is Model is

`severe+death+severe.death+anti.death`

with a scaled deviance of 0.36773 with 1 degree of freedom. This clearly implies that there is no three way factor effect but that severity is related to death as is the administration of an antidote. See below

```
> summary(p2)
```

```
Call:
```

```
glm(formula = y ~ sev * anti * mort - sev:anti:mort, family = poisson)
```

```
Coefficients:
```

```

      Estimate Std. Error z value Pr(>|z|)
(Intercept)  3.1166      0.2064  15.102 < 2e-16 ***
severe2      -1.2553      0.4060  -3.092 0.001987 **
anti2       -0.4471      0.3198  -1.398 0.162105
death2      -1.8835      0.4868  -3.869 0.000109 ***
severe2:anti2  0.3030      0.5218   0.581 0.561406
severe2:death2 1.7394      0.5258   3.308 0.000940 ***
anti2:death2  1.0964      0.5340   2.053 0.040051 *
(\index{dispersion parameter} dispersion parameter for poisson family taken to be 1)
Null \index{deviance} deviance: 28.69085 on 7 degrees of freedom
Residual \index{deviance} deviance: 0.36773 on 1 degrees of freedom
AIC: 46.11
```

```
> anova(p2)
```

```
Analysis of \index{deviance} deviance Table
```

```
Model: poisson, link: log
```

```
Response: y
```

```
Terms added sequentially (first to last)
```

	Df	\index{deviance} deviance	Resid. Df	Resid. Dev
NULL			7	28.6909
severe	1	2.8655	6	25.8254
anti	1	0.1140	5	25.7114
death	1	4.6147	4	21.0967
severe:anti	1	2.4403	3	18.6564
severe:death	1	13.9082	2	4.7481
anti:death	1	4.3804	1	0.3677

*What do you think this means?*

Complex tables result, as you might expect, in complex analysis - there really is no such thing as a free lunch. The generalized linear modelling ideas follow though and one tries to reduce the model to a tractable and understandable form. This may not be easy and you do read to remember that you cannot remove some terms-there are rows and columns!

### A complex table

If we run the head injuries ( see below)

		D					
		1			2		
		C			C		
B	A	1	2	3	1	2	3
1	1	48	29	7	25	23	8
2	2	25	98	58	7	30	27
3	1	27	22	4	29	33	8
1	2	3	36	32	8	26	25
2	1	34	27	12	59	73	19
3	2	4	4	17	8	23	30

though a generalized linear modelling package we get

Call:

```
glm(formula = y ~ a * b * c * d - a:b:c:d, family = poisson)
```

Coefficients:

	Value	Std.error	z value	P(> z )
(Intercept)	3.8199	0.1458	26.2027	0.0000
a2	-2.1335	0.3828	-5.5741	0.0000
b2	-0.2498	0.2163	-1.1552	0.2480
b3	-0.4920	0.2345	-2.0982	0.0359
c2	-0.4243	0.2258	-1.8789	0.0603
c3	-1.6722	0.3385	-4.9396	0.0000
d2	-0.5094	0.2316	-2.1991	0.0279
a2.b2	1.7196	0.4234	4.0617	0.0000
a2.b3	-0.0566	0.5434	-0.1041	0.9171
a2.c2	2.2979	0.4190	5.4845	0.0000
a2.c3	3.4014	0.4848	7.0156	0.0000
a2.d2	0.5458	0.4159	1.3122	0.1895
b2.c2	0.2137	0.3160	0.6764	0.4988
b3.c2	0.0611	0.3552	0.1719	0.8635
b2.c3	0.2658	0.4417	0.6017	0.5474
b3.c3	0.0901	0.5061	0.1781	0.8586
b2.d2	0.9907	0.3003	3.2988	0.0010
b3.d2	0.5179	0.3402	1.5223	0.1279
c2.d2	0.2123	0.3322	0.6391	0.5228
c3.d2	0.2233	0.4439	0.5030	0.6150
a2.b2.c2	-0.6769	0.4607	-1.4693	0.1417
a2.b3.c2	-1.1838	0.5568	-2.1262	0.0335
a2.b2.c3	-1.0354	0.5424	-1.9089	0.0563
a2.b3.c3	-0.2313	0.6399	-0.3615	0.7177

a2.b2.d2	-2.0409	0.3617	-5.6418	0.0000
a2.b3.d2	0.4916	0.4581	1.0731	0.2832
a2.c2.d2	-0.5191	0.3942	-1.3167	0.1879
a2.c3.d2	-0.3957	0.4588	-0.8625	0.3884
b2.c2.d2	0.2126	0.4041	0.5262	0.5988
b3.c2.d2	0.3871	0.4704	0.8229	0.4106
b2.c3.d2	0.2360	0.4832	0.4885	0.6252
b3.c3.d2	-0.1421	0.5739	-0.2476	0.8044

(\index{dispersion parameter} dispersion parameter for poisson family taken to be 1)

Null \index{deviance} deviance: 502.4225 on 35 degrees of freedom

Residual \index{deviance} deviance: 10.22099 on 4 degrees of freedom

Number of Fisher Scoring iterations: 2

Analysis of \index{deviance} deviance Table

Model: poisson, link: log

Response: y

Terms added sequentially (first to last)

	Df	\index{deviance} deviance	Resid. Df	Resid. Dev
NULL		35	502.42	
a	1	0.7132	34	501.71
b	2	112.6217	32	389.09
c	2	54.6303	30	334.46
d	1	0.7132	29	333.74
a:b	2	7.1828	27	326.56
a:c	2	180.5846	25	145.98
a:d	1	27.4246	24	118.55
b:c	4	6.4986	20	112.05
b:d	2	19.3393	18	92.71
c:d	2	4.1595	16	88.55
a:b:c	4	13.0262	12	75.53
a:b:d	2	61.4765	10	14.05
a:c:d	2	1.9768	8	12.08
b:c:d	4	1.8544	4	10.22

**Survival of breast cancer patients**

The data in the table are from a study of the survival of breast cancer patients (see Morrison et al. [1973]). The variables and categories are

- Variable 1. Degree of chronic inflammatory reaction
  1. Minimal
  2. Moderate severe
- Variable 2. Nuclear grade
  1. Relatively malignant appearance
  2. Relatively benign appearance
- Variable 3. Survival for three years
  1. No
  2. Yes
- Variable 4. Age of diagnosis
  1. Under 50 years
  2. 50-69 years
  3. 70 or older
- Variable 5. Center where patient was diagnosed
  1. Tokyo
  2. Boston
  3. Glamorgan

Center	Age	Survived	appearance			
			min. inf.		greater. inf.	
			Malignant	Benign	Malignant	Benign
Tokyo	Under 50	No	9	7	4	3
		Yes	26	68	25	9
	50-69	No	9	9	11	2
		Yes	20	46	18	5
	70 or over	No	2	3	1	0
		Yes	1	6	5	1
Boston	Under 50	No	6	7	6	0
		Yes	11	24	4	0
	50-69	No	8	20	3	2
		Yes	18	58	10	3
	70 or over	No	9	19	3	0
		Yes	15	26	1	1
Glamorgan	Under 50	No	16	7	3	0
		Yes	16	20	8	1
	50-69	No	14	12	3	0
		Yes	27	39	to	4
	70 or over	No	3	7	3	0
		Yes	12	11	4	1

Of

course the output (shown in part below) is complex but almost tractable

Coefficients:

	Value	Std. Error	t value
(Intercept)	0.531560489	6.47880371	0.08204609
center1	-0.983757819	6.47616415	-0.15190440
center2	-0.128403764	3.95271388	-0.03248496
age1	-1.336441195	7.24634411	-0.18442972
age2	0.236788506	2.41612183	0.09800355
survive	1.273145482	5.56883056	0.22861990
appear	-1.390959011	6.47880908	-0.21469363
inflam	-1.948156947	6.47879645	-0.30069735
center1age1	-0.341504015	0.12953220	-2.63644111
center2age1	-0.504666525	4.96670785	-0.10160987
center1age2	0.008557805	0.09616504	0.08899081
center2age2	0.225833937	1.65592137	0.13637963
center1survive	0.720583581	6.47598740	0.11127007
center2survive	-0.211159775	2.15927585	-0.09779194
age1survive	0.702540896	5.27683201	0.13313687
age2survive	-0.494503262	1.75987058	-0.28098843
center1appear	-0.746984218	6.47606148	-0.11534545
center2appear	-0.093350494	3.95268973	-0.02361695
age1appear	-1.066283129	7.24634223	-0.14714777

```
age2appear 0.359377146 2.41585510 0.14875774
survive:appear 0.956679063 5.56884119 0.17179141
center1inflam -0.649830921 6.47616999 -0.10034186
center2inflam -0.105971264 3.95272551 -0.02680967
age1inflam -1.143845921 7.24636698 -0.15785095
age2inflam 0.345145726 2.41599914 0.14285838
survive:inflam 0.853583324 5.56887500 0.15327752
appear:inflam -1.620623374 6.47878057 -0.25014327
center1age1survive -0.025692484 0.10745908 -0.23909086
center2age1survive -0.061877674 0.05781880 -1.07019989
```

...

```
center2age1appearinflam -0.464265416 4.96655408 -0.09347838
center1age2appearinflam -0.051069422 0.09296057 -0.54936649
center2age2appearinflam 0.198731662 1.65588842 0.12001513
center1surviveappearinflam 0.637181621 6.47583563 0.09839373
center2surviveappearinflam -0.122495849 2.15919658 -0.05673214
age1surviveappearinflam 0.700967234 5.27659140 0.13284471
age2surviveappearinflam -0.200923030 1.75979744 -0.11417395
```

(\index{dispersion parameter} dispersion parameter for Poisson family taken to be 1 )

Null \index{deviance} deviance: 860.0076 on 71 degrees of freedom

Residual \index{deviance} deviance: 4.037807 on 4 degrees of freedom

Terms added sequentially (first to last)

	Df	\index{deviance}	deviance	Resid. Df	Resid. Dev
NULL			71	860.0076	
center	2	44.9179	69	815.0897	
age	2	45.0417	67	770.0480	
survive	1	160.6009	66	609.4470	
appear	1	7.5727	65	601.8743	
inflam	1	291.1986	64	310.6757	
center:age	4	46.0279	60	264.6478	
center:survive	2	3.0611	58	261.5867	
age:survive	2	66.6453	56	194.9414	
center:appear	2	6.1072	54	188.8342	
age:appear	2	1.2279	52	187.6063	
survive:appear	1	7.1981	51	180.4082	
center:inflam	2	1.4018	49	179.0064	
age:inflam	2	0.0732	47	178.9332	
survive:inflam	1	0.2341	46	178.6991	
appear:inflam	1	95.5458	45	83.1533	
center:age:survive	4	9.5517	41	73.6016	
center:age:appear	4	4.0168	37	69.5848	
center:survive:appear	2	4.0583	35	65.5264	
age:survive:appear	2	5.9560	33	59.5705	
center:age:inflam	4	2.0685	29	57.5020	
center:survive:inflam	2	15.6643	27	41.8376	
age:survive:inflam	2	5.4138	25	36.4239	
center:appear:inflam	2	0.7656	23	35.6582	
age:appear:inflam	2	1.9883	21	33.6699	
survive:appear:inflam	1	0.1750	20	33.4950	
center:age:survive:appear	4	6.2824	16	27.2126	
center:age:survive:inflam	4	14.4032	12	12.8094	
center:age:appear:inflam	4	3.9422	8	8.8672	
center:survive:appear:inflam	2	1.9270	6	6.9402	
age:survive:appear:inflam	2	2.9024	4	4.0378	

*Analyze this data set as a Binomial model*

### Exercise

In an experiment students were sent a series of the text messages. They were asked to reply giving the category most closely associated with ( but not the same as) the received message. The results are tabulated below. Model this data.

sent	received					
	love	status	information	money	goods	services
love	-	73	11	0	2	25
status	69	-	22	11	3	6
information	19	18	-	12	27	15
money	0	18	9	-	66	18
goods	7	6	23	61	-	14
services	46	20	8	18	19	-

### 0.16.1 Zero frequencies

When we have cross tabulations in several dimensions some cells often contain zeros. The zero may have occurred because the sample size was too small and there is the possibility that the cell will fill if there are further observations. Of course it may be that the combination of categories is impossible, either a structural zero or a model zero, the latter made by excluding the combination from the model for some reason. We start by looking at the case where the zeros are a sampling problem.

### 0.16.2 Sampling zeros

If we try and fit a saturated model even one zero will cause estimation problems. Most software will only estimate as many parameters as there are non-zero entries in the table. For an unsaturated model all the parameters can be estimated as long as we have at least as many non-zero frequencies as parameters. You should be aware that there have been unexpected exceptions to this rule. One indicator is unexpectedly large standard errors of parameter estimates. We take an example due to Lindsey on the survival of melanoma patients.

		remission status					
		relapsed		still in		never in	
Sex	f.history	alive	dead	alive	dead	alive	dead
Male	no	10	4	0	3	2	5
	yes	0	0	0	0	0	0
	unknown	4	0	1	3	6	16
Female	yes	7	0	2	4	5	4
	no	1	0	1	0	0	1
	unknown	6	0	2	1	6	5

The model for independence of survival from the three explanatory variables is

$$\text{surv} + \text{sex} * \text{hist} * \text{remis}$$

with deviance of 42.4 and 17 degrees of freedom. If we inspect the standard errors we see some rather large values., all where  $\text{hist}(2)$  is present. One possibility is to take out the males with a family history of melanoma. All zeros. This gives a scaled deviance of 42.36 with 14 degrees of freedom

	original data		zeros removed	
	estimate	se(est)	estimate	se(est)
Constant	2.014	0.2832	2.014	0.2831
surv(2)	-0.1417	0.2015	-0.1416	0.2013
sex(2)	-0.6931	0.4629	-0.6931	0.461
hist(2)	-10.73	34.71	0	aliased
hist(3)	-1.253	0.5669	-1.253	0.565
remis(2)	-1.54	0.6362	-1.54	0.635
remis(3)	-0.6931	0.4629	-0.6931	0.4629
sex(2).hist(2)	8.787	34.73	-1.946	1.068
sex(2).hist(3)	1.099	0.7943	1.099	0.7906
sex(2).remis(2)	1.386	0.8452	1.386	0.8432
sex(2).remis(3)	0.9445	0.6843	0.9444	0.6829
hist(2).remis(2)	1.54	49.09	0.1542	1.518
hist(2).remis(3)	0.6931	49.09	-0.2512	1.5
hist(3).remis(2)	1.54	0.9512	1.54	0.9492
hist(3).remis(3)	2.398	0.714	2.398	0.7124
sex(2).hist(2).remis(2)	-1.386	49.11	0	aliased
sex(2).hist(2).remis(3)	-0.9445	49.11	0	aliased
sex(2).hist(3).remis(2)	-2.079	1.309	-2.079	1.306
sex(2).hist(3).remis(3)	-2.043	1.011	-2.043	1.008

We see that the new parameter estimates no longer have inflated standard errors. If we go on we find survival depends on sex but I shall leave that to you.

### 0.16.3 Structural zeros

When a table has structural zeros we simply exclude them from the model and continue. Note in cases such as this when we have structural zeros and we therefore cannot have the usual independence formulation we talk of *quasi-independence*.

Consider the following table

		Health Problem			
sex	age	sex	menstruation	general	nothing
male	12-15	4	0	42	57
male	16-17	2	0	7	20
female	12-15	9	4	19	71
female	16-17	7	8	10	31

The minimal independence model is  
health+sex\*age

Model is health+sex+age+sex.age

Scaled  $\backslash\text{index}\{\text{deviance}\}$  deviance is 36.96(+27.53) df 9(+3)

So we seek something better. If we introduce a sex effect  
`health+sex*age+sex.health`  
 the model has a deviance of 9.43 and fits reasonably well.

	estimate	se(est)	z ratio	Prob> z
1 Constant	1.544	0.4109	3.757	0.0002
2 health(2)	-10.71	52.51	-0.2040	0.8384
3 health(3)	2.100	0.4325	4.855	<0.0001
4 health(4)	2.552	0.4239	6.021	<0.0001
5 sex(2)	0.7947	0.4845	1.640	0.1009
6 age(2)	-1.267	0.2102	-6.029	<0.0001
7 sex(2).age(2)	0.6581	0.2679	2.457	0.0140
8 sex(2).health(2)	10.43	52.51	0.1985	0.8426
9 sex(2).health(3)	-1.505	0.5330	-2.824	0.0047
10 sex(2).health(4)	-0.6997	0.5020	-1.394	0.1634

residuals after the fit			
obsn.	actual	estimate	residual
1	4.000	4.682	-0.3151
2	2.000	1.318	0.5939
3	9.000	10.36	-0.4239
4	7.000	5.635	0.5749
5	0.000	1.041e-4	-1.020e-2
6	0.000	2.931e-5	-5.414e-3
7	4.000	7.774	-1.353
8	8.000	4.226	1.836
9	42.00	38.23	0.6089
10	7.000	10.77	-1.148
11	19.00	18.79	4.934e-2
12	10.00	10.21	-6.691e-2
13	57.00	60.08	-0.3978
14	20.00	16.92	0.7497
15	71.00	66.08	0.6058
16	31.00	35.92	-0.8216

You will of course have noted that not all interaction effects are possible and are therefore not in the model.

### 0.16.4 Fitting distributions

Often we examine distributions by looking at the number of observations in classes or bins. This approach allows us to use logliner models. Suppose we have a distribution from the exponential family

$$f(y, \theta) = \exp[\theta b(y) + c(\theta) + d(y)]$$

then the probability of falling in any cell is

$$\pi = P[y - \Delta/2 \leq Y \leq y + \Delta/2] = \int_{y-\Delta/2}^{y+\Delta/2} f(x) dx \simeq f(y)\Delta$$

Thus

$$\log(\pi N) = \theta b(y) + c(\theta) + d(y) + \log(N\Delta)$$

which is linear in  $b(y)$ . Now as  $d(y) + \log(N\Delta)$  contain no unknown parameters we can use this term as an offset and we have Poisson regression

$$\log(\pi_j N) = \beta_0 + \beta^T \mathbf{b}(y_j)$$

-after some algebra. The upshot is that when we have frequency data any distribution from the exponential family can be represented as a Poisson log-linear regression. We can have similar representations of other distributions but the regression is no longer linear. In this formulation the explanatory variable is  $b(y)$ , the sufficient statistic. The table below gives a useful summary.

Distribution	Sufficient statistics	Offset
Geometric	$y_j$	
Poisson	$y_j$	$-\log(y_j!)$
Binomial	$y_j$	$-\log\binom{n_j}{y_j}$
Normal	$y_j, y_j^2$	
Log Normal	$\log(y_j), \log^2(y_j)$	
Exponential	$y_j$	
Pareto	$y_j$	
Gamma	$y_j, \log(y_j)$	

#### Belgian drivers

As an example consider the table below showing the number of accidents in a year for 9461 Belgian drivers.

Accidents	Drivers
0	7840
1	1317
2	239
3	42
4	14
5	4
6	4
7	1

Fitting a model with Poisson error and log link we have a deviance of 30.27 ( a change of 28640) with 6 df. The summary results are

```

              estimate    se(est)   z ratio   Prob>|z|
1 Constant    8.961      1.115e-2   803.5    <0.0001
2 accidents  -1.734      2.014e-2  -86.10   <0.0001
Scale is fixed at 1.000

```

```

      actual estimate   residual
1    7840      7791      0.5559
2    1317     1375     -1.571
3    239.0    242.8    -0.2417
4     42.00    42.85   -0.1304
5     14.00     7.565    2.340
6     4.000     1.335    2.306
7     4.000     0.2357   7.753
8     1.000     4.161e-2  4.698

```

Not a very good fit in the tail of the distribution!

### Post Office Staff

A more complex data set is by Burridge. This gives the employment durations of Post Office staff

Months	grade 1	grade 2	months	grade 1	grade 2
1	22	30	13	0	1
2	18	28	14	0	0
3	19	31	15	0	0
4	13	14	16	1	1
5	5	10	17	1	1
6	6	6	18	1	0
7	3	5	19	3	2
8	2	2	20	1	0
9	2	3	21	1	3
10	1	0	22	0	1
11	0	0	23	0	1
12	1	1	24	0	0

Fitting grade+month+loge (month) gives a deviance of 76.98 with 44 df. This is a Gamma and is rather a poor fit see figure 11

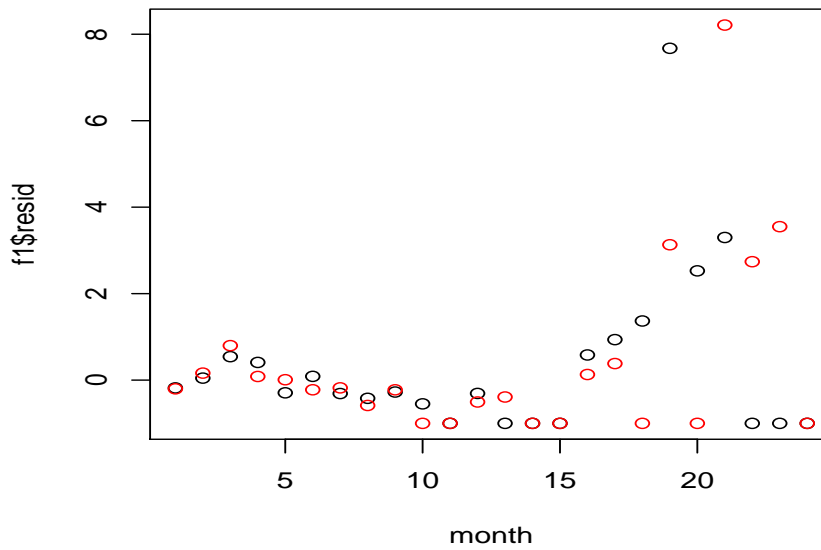


Figure 11: Post office data fit

We leave this topic with two comments. Of course we can fit other functions of  $y$  to the data and end up with distributions with complex normalizing constants. For example in the case above

$$\text{grade+month+loge(month)+reciprocal(month)+sqr(reciprocal(month))}$$

has a deviance of 39.79 with 42 df and fits very much better. All you have to do is find the analytic expression.

The following is an interesting case

Occupants	houses
1	436
2	133
3	19
4	2
5	1
6	0
7	1

### Postal Survey

These are the numbers of occupants in houses who replied to a postal survey. Obviously houses with zero occupants cannot reply so we need a truncated distribution. Try a truncated Poisson

$$P[y] = \frac{\lambda^y e^{-\lambda}}{y!(1 - e^{-\lambda})}$$

Fitting a Poisson gives

	actual	estimate	residual
1	436.0	449.8	-0.6507
2	133.0	108.1	2.399
3	19.00	25.96	-1.366
4	2.000	6.237	-1.696
5	1.000	1.498	-0.4071
6	0.000	0.3599	-0.5999
7	1.000	8.647e-2	3.107

and

	estimate	se(est)	z ratio	Prob> z
1 Constant	7.535	9.357e-2	80.53	<0.0001
2 occs	-1.426	6.388e-2	-22.32	<0.0001

Scale is fixed at 1.000

Find the parameter  $\lambda$

**Part III**

**Counting Processes**

## 0.17 Introduction

It is time we looked at dependent events. As in our first lectures we will look at the probability, called the risk or the intensity, of an event in a small time interval. We may also recall that one can also look at the accumulation of events. This counting process  $N(t)$  is a random variable over time which just count the number of events up to time  $T$ . The intensity is just the (local) rate at which the counting process changes over time.

As an illustration we consider the Prussian horse kicks data which gives the number of deaths due to horse kicks in the Prussian army from 1875 to 1894.

We can consider the totals as a random process over time and we may be interested in the death rate or the number of deaths over time. Examination of the data will show that the rate appears to be different for different army corps.

year	Corps														total
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	
1875	0	0	0	0	0	0	0	1	1	0	0	0	1	0	3
1876	2	0	0	0	1	0	0	0	0	0	0	0	1	1	5
1877	2	0	0	0	0	0	1	1	0	0	1	0	2	0	7
1878	1	2	2	1	1	0	0	0	0	0	1	0	1	0	9
1879	0	0	0	1	1	2	2	0	1	0	0	2	1	0	10
1880	0	3	2	1	1	1	0	0	0	2	1	4	3	0	18
1881	1	0	0	2	1	0	0	1	0	1	0	0	0	0	6
1882	1	2	0	0	0	0	1	0	1	1	2	1	4	1	14
1883	0	0	1	2	0	1	2	1	0	1	0	3	0	0	11
1884	3	0	1	0	0	0	0	1	0	0	2	0	1	1	9
1885	0	0	0	0	0	0	1	0	0	2	0	1	0	1	5
1886	2	1	0	0	1	1	1	0	0	1	0	1	3	0	11
1887	1	1	2	1	0	0	3	2	1	1	0	1	2	0	15
1888	0	1	1	0	0	1	1	0	0	0	0	1	1	0	6
1889	0	0	1	1	0	1	1	0	0	1	2	2	0	2	11
1890	1	2	0	2	0	1	1	2	0	2	1	1	2	2	17
1891	0	0	0	1	1	1	0	1	1	0	3	3	1	0	12
1892	1	3	2	0	1	1	3	0	1	1	0	1	1	0	15
1893	0	1	0	0	0	1	0	2	0	0	1	3	0	0	8
1894	1	0	0	0	0	0	0	0	1	0	1	1	0	0	4

As an illustration we consider the table above which gives the number of death due to horse kicks in the Prussian army from 1875 to 1894, classified by army corps. We may be interested in the death rate or the number of deaths over time. Examination of the data will show that the death rate appears to be different for different army corps. It is known that corps 1,2,4 and 9 had a different internal organization to the remainder of the army.

Fitting a Poisson error log-linear model with just the mean gives a scaled deviance of 323.2 with 279 df. Adding the corps as an explanatory variable gives a scaled deviance of 297.1 (-26.14) with 266(-13) degrees of freedom. If however we look at the parameters it is tempting to see differences in the rates between corps. As few if any of the rates differ significantly from zero this is not very sensible.

Fitting the year as well as the corps gives a deviance of 294.8(-2.287) with 265(-1) df. Hardly a huge change.

		estimate	se(est)	z ratio	Prob $\geq  z $
1	Constant	-0.2231	0.2497	-0.8937	0.3715
2	corps(2)	2.231e-5	0.3527	6.325e-5	0.9999
3	corps(3)	-0.2877	0.3810	-0.7550	0.4503
4	corps(4)	-0.2877	0.3813	-0.7545	0.4506
5	corps(5)	-0.6931	0.4326	-1.602	0.1091
6	corps(6)	-0.3747	0.3913	-0.9575	0.3383
7	corps(7)	6.063e-2	0.3478	0.1743	0.8616
8	corps(8)	-0.2877	0.3813	-0.7545	0.4506
9	corps(9)	-0.8267	0.4526	-1.826	0.0678
10	corps(10)	-0.2076	0.3729	-0.5568	0.5777
11	corps(11)	-6.453e-2	0.3588	-0.1799	0.8573
12	corps(12)	0.4463	0.3198	1.396	0.1629
13	corps(13)	0.4055	0.3224	1.258	0.2085
14	corps(14)	-0.6931	0.4317	-1.606	0.1084

In many cases the rates need to be weighed by another factor. This often happens in epidemiology and it is from here that we take the next example. This is the number of lung cancer cases in four Danish cities between 1968 and 1971.

Age	cases	popn.	cases	popn.	cases	popn.	cases	popn.
40-54	11	3059	13	2879	4	3142	5	2520
55-59	11	800	6	1083	8	1050	7	878
60-64	11	710	15	923	7	895	10	839
65-69	10	581	10	834	11	702	14	631
70-74	11	509	12	634	9	535	8	539
75+	10	605	2	782	12	659	7	619

The rates can be and incline us to believe that the rate depends on age and on the city. The additive model

city+age

-with the log of the population as an *offset*- has a deviance of 23.45 with 15 degrees of freedom. If we use the age as a covariate (midpoint of age class) we have a poorer model. This also gives a clue as to how to proceed, a quadratic model using the mid-point of the age class. A quadratic model is also prompted by the decline in rate at the end of the age range. The deviance is 23.35 with 18 degrees of freedom. The deviance is 25.10(+1.654) on 18(+3) df, for the model city+age+sqr(age). This is hardly a massive improvement but it is simpler to handle. The estimates are

		estimate	se(est)	z ratio	Prob $\geq  z $
1	Constant	-21.21	2.780	-7.628	0.0001
2	city(2)	-0.3329	0.1815	-1.835	0.0666
3	city(3)	-0.3757	0.1878	-2.001	0.0454
4	city(4)	-0.2746	0.1878	-1.462	0.1438
5	age	0.4999	9.062e-2	5.517	$\leq 0.0001$
6	sqr(age)	-3.608e-3	7.263e-4	-4.967	$\leq 0.0001$

Can we say that examination shows that one city is rather different to the others? If so which is it?

### 0.17.1 Point Processes

If the counts of a Poisson process can be disaggregated so that the point in time for each event is known then we have a point process. For example the data below gives the June days with measurable precipitation (1) at Madison, Wisconsin, 1961-1971. Because there are only two alternatives a Bernoulli (Binomial) error seems appropriate. One simple model is the Markov one. In this we suppose that the probability of any zero or one depends only on the previous state.

1961	10000	01101	01100	00010	01010	00000
1962	00110	00101	10000	01100	01000	00000
1963	00001	01110	00100	00010	00000	11000
1964	01000	00000	11011	01000	11000	00000
1965	10001	10000	00000	00001	01100	01000
1966	01100	11010	11001	00001	00000	11100
1967	00000	11011	11101	11010	00010	00110
1968	10000	00011	10011	00100	10111	11011
1969	11010	11000	11000	01100	00001	11010
1970	11000	00000	01000	11001	00000	10000
1971	10000	01000	10000	00111	01010	00000

If we fit a constant probability model we have a deviance of 418.7 with 329 df. If we add year as a factor we have a deviance of 405.5 with 319 df. Hardly a big improvement. If we examine the parameter estimates it is tempting to speculate on some cyclic effect!

		estimate	se(est)	z ratio	Prob $\geq  z $
1	Constant	-0.8473	0.3984	-2.127	0.0334
2	year(2)	-0.1643	0.5736	-0.2864	0.7746
3	year(3)	-0.1643	0.5736	-0.2864	0.7746
4	year(4)	-0.1643	0.5736	-0.2864	0.7746
5	year(5)	-0.3423	0.5871	-0.5830	0.5599
6	year(6)	0.4418	0.5455	0.8099	0.4180
7	year(7)	0.7138	0.5410	1.319	0.1870
8	year(8)	0.8473	0.5404	1.568	0.1169
9	year(9)	0.5790	0.5426	1.067	0.2860
10	year(10)	-0.3423	0.5871	-0.5830	0.5599
11	year(11)	-0.1643	0.5736	-0.2864	0.7746

One interesting question is whether rainy days are linked. To do this we can fit a Bernoulli model and use the previous days outcome as a covariate. To illustrate the process we look at the 1971 data. The y column is the original data while y1 y2 y3 etc. are the lagged columns.

y	total	year(f)	year	y1	y1	y3	y4
0	1	11	1971	0	0	0	1
0	1	11	1971	0	0	0	0
1	1	11	1971	0	0	0	0
0	1	11	1971	1	0	0	0
0	1	11	1971	0	1	0	0
0	1	11	1971	0	0	1	0
1	1	11	1971	0	0	0	1
0	1	11	1971	1	0	0	0
0	1	11	1971	0	1	0	0
0	1	11	1971	0	0	1	0
0	1	11	1971	0	0	0	1
0	1	11	1971	0	0	0	0
0	1	11	1971	0	0	0	0
1	1	11	1971	0	0	0	0
1	1	11	1971	1	0	0	0
1	1	11	1971	1	1	0	0
0	1	11	1971	1	1	1	0
1	1	11	1971	0	1	1	1
0	1	11	1971	1	0	1	1
1	1	11	1971	0	1	0	1
0	1	11	1971	1	0	1	0
0	1	11	1971	0	1	0	1
0	1	11	1971	0	0	1	0
0	1	11	1971	0	0	0	1
0	1	11	1971	0	0	0	0
0	1	11	1971	0	0	0	0

The year(f) column is the factor version of the year covariate - there are 11 years  
 Fitting 3 lagged terms using a Bernoulli model gives

		estimate	se(est)	z ratio	Prob $\geq  z $
1	Constant	-1.175	0.6484	-1.812	0.0700
2	y1	0.3461	1.064	0.3254	0.7449
3	y2	1.101	0.9776	1.126	0.2601
4	y3	-1.187	1.258	-0.9436	0.3454
		odds ratio	95% confidence interval		
1	Constant	0.3088	(8.663e-2,1.101)		
2	y1	1.413	(0.1758,11.37)		
3	y2	3.007	(0.4426,20.43)		
4	y3	0.3051	(2.592e-2,3.592)		

From these results there seems to be no dependence between days and we can probably assume a purely random process.

The table below gives a series indicating if patients were arriving (1) at an intensive care unit each day from February 1963 to March 1964 from Cox and Lewis, 1966, pp. 254-255, read across rows). Analyze this data set. Is the series random?

```
00010 00100 10000 10101 10001 00110 10001 01000
00111 00101 01000 10100 10001 00111 00011 00000
01000 01100 00101 10001 01101 01110 11110 01010
10101 00001 01100 10100 11011 11011 01000 00111
01100 00001 10110 01010 01110 00100 01010 00001
01001 00000 01010 01011 01101 01101 00101 10011
00111 00101 00011 00000 11011 00100 01110 01111
11011 00111 11001 11011 01111 10101 11011 11111
00111 11100 10010 11011 10011 10110 10111 00110
00111 00001 11000 11000 01111 00111 10001 01010
00110 00000 1
```

One point of interest in the processes such as the ones above are the transition probabilities. These are the probability of moving to a particular state given that one is in a given state. For example, what is the probability that it will rain tomorrow given that it did not rain today? Tied in with this question is the problem of the series memory. One popular, but very strong assumption is that the series is Markov and that the probability of being in a state at time  $t$  is only dependent on the state at the previous time point. The system has no memory further back. This we can check in a fairly simple way. Given our Bernoulli sequence  $y$  we use the same series, lagged by one as a covariate. Thus we use  $y$  at  $t-1$  as an explanatory variable for  $y$  at  $t$ . This can clearly be extended to use series lagged by one, two etc. This gives a plausible method of measuring dependence.

If we have a stationary markov process, that is one where the probabilities are not time dependent we can call on lots of nice theory so we are also interested in the time dependence or otherwise of the probabilities in the system. We digress for a moment to spell out a simple case.

Suppose we have  $k$  States  $S_1, S_2 \dots S_k$  that the system can reach. Then we can have the unconditional probabilities

$$P[\text{in } S_j \text{ at time } t] = p_t(j)$$

and the conditional transition probabilities

$$P[\text{in } S_i \text{ at time } t + 1 | \text{in } S_j \text{ at time } t] = p_t(i, j)$$

Note the subscripts!

We can then use simple probability theory to get

$$p_{t+1}(i) = \sum_{m=1}^k p_t(i, m)p_t(m)$$

This Markov transition equation is usually written as a vector equation

$$\mathbf{p}_{t+1} = \mathbf{P}_t \mathbf{p}_t$$

Stationarity, which means that we can drop the  $t$  for the transition matrix  $P_t$  is obviously a huge simplification. As we can examine the dependence and the stationarity via a generalized linear model we can see this is a fruitful approach. Notice the transition probabilities can be found from the parameter estimates using lagged variables

### Multiplicative Intensities

We now look briefly at some continuous time models. An event history follows a phenomenon over time, recording events. These may be repeats or different events. An event may signal a change of state so we need to allow the intensity to change. Now apart from the exponential distribution the intensity does change. Thus for the Weibull distribution the intensity function is

$$\lambda(t, \beta, \kappa) = t^{\kappa-1} h(\beta, x)$$

where  $h$  is some function of the explanatory variables. The mathematics gets a bit deep around here.

Suppose the intensity of the process is  $\lambda(t|F_{t-}, \beta)$  where  $F_{t-}$  is the filtration or history up to (but not including)  $t$ , so

$$\lambda(t|F_{t-}, \beta) = P[dN(t) = 1|F_{t-}, \beta]$$

Then the (kernel) of the log likelihood is

$$\int_0^T \log[\lambda(t|F_{t-}, \beta)] dN(t) - \int_0^T \lambda(t|F_{t-}, \beta) I(t) dt$$

where  $I(t)$  is an indicator function which is 1 when the process is observed.

When this is translated to a real series observed at intervals we may show that we get the likelihood of a censored Poisson process for  $N(t)$  with a mean  $\lambda(t, \beta, \kappa)$ . Conditional on the filtration it is the likelihood of a local Poisson process. This gives us opportunities for analysis of risky processes for as we have seen we can fit models using Poisson processes but with different models dictated by the covariates and the intensity function

### AIDS reports in England and Wales

In the study of the incidence of acquired immune deficiency syndrome (AIDS), one major problem involves delays in reporting diagnosed cases. Cases which are diagnosed locally at a certain time may only reach the collection centre some months later. Such data on incidence of AIDS, with reporting delays, take the form of a rectangular contingency table with observations in one triangular corner missing. The two dimensions of such a

table are the diagnosis period, describing the process of growing AIDS incidence, and the process causing reporting delay. The margin for diagnosis period, in which we are particularly interested, gives the total incidence over time, but with the most recent values too small because of the missing triangle of values not yet reported. The table gives the number of AIDS reports in England and Wales to the end of 1992 by quarter, with reporting delays. (de Angelis and Gilks, 1994)

Year	Delay period in quarters														
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14+
1983	2	6	0	1	1	0	0	1	0	0	0	0	0	0	1
	2	7	1	1	1	0	0	0	0	0	0	0	0	0	0
1984	4	4	0	1	0	2	0	0	0	0	2	1	0	0	0
	0	10	0	1	1	0	0	0	1	1	1	0	0	0	0
	6	17	3	1	1	0	0	0	0	0	0	1	0	0	1
	5	22	1	5	2	1	0	2	1	0	0	0	0	0	0
1985	4	23	4	5	2	1	3	0	1	2	0	0	0	0	2
	11	11	6	1	1	5	0	1	1	1	1	0	0	0	1
	9	22	6	2	4	3	3	4	7	1	2	0	0	0	0
	2	28	8	8	5	2	2	4	3	0	1	1	0	0	1
1986	5	26	14	6	9	2	5	5	5	1	2	0	0	0	2
	7	49	17	11	4	7	5	7	3	1	2	2	0	1	4
	13	37	21	9	3	5	7	3	1	3	1	0	0	0	6
	12	53	16	21	2	7	0	7	0	0	0	0	0	1	1
1987	21	44	29	11	6	4	2	2	1	0	2	0	2	2	8
	17	74	13	13	3	5	3	1	2	2	0	0	0	3	5
	36	58	23	14	7	4	1	2	1	3	0	0	0	3	1
	28	74	23	11	8	3	3	6	2	5	4	1	1	1	3
1988	31	80	16	9	3	2	8	3	1	4	6	2	1	2	6
	26	99	27	9	8	11	3	4	6	3	5	5	1	1	3
	31	95	35	13	18	4	6	4	4	3	3	2	0	3	3
	36	77	20	26	11	3	8	4	8	7	1	0	0	2	2
1989	32	92	32	10	12	19	12	4	3	2	0	2	2	0	2
	15	92	14	27	22	21	12	5	3	0	3	3	0	1	1
	34	104	29	31	18	8	6	7	3	8	0	2	1	2	-
	38	101	34	18	9	15	6	1	2	2	2	3	2	-	-
1990	31	124	47	24	11	15	8	6	5	3	3	4	-	-	-
	32	132	36	10	9	7	6	4	4	5	0	-	-	-	-
	49	107	51	17	15	8	9	2	1	1	-	-	-	-	-
	44	153	41	16	11	6	5	7	2	-	-	-	-	-	-
1991	41	137	29	33	7	11	6	4	-	-	-	-	-	-	-
	56	124	39	14	12	7	10	-	-	-	-	-	-	-	-
	53	175	35	17	13	11	-	-	-	-	-	-	-	-	-
	63	135	24	23	12	-	-	-	-	-	-	-	-	-	-
1992	71	161	48	25	-	-	-	-	-	-	-	-	-	-	-
	95	178	39	-	-	-	-	-	-	-	-	-	-	-	-
	76	181	-	-	-	-	-	-	-	-	-	-	-	-	-
	67	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Suppose we write a simple model for the intensity function of a bivariate Poisson process, with the reporting delay process stationary over time ( i.e. no time trend)

$$\lambda(t, D) = \lambda_D \lambda_t$$

We are assuming that the rate of cases falling in a cell can be written as a function of the diagnosis time ( $t$ ) with intensity  $\lambda_t$  and the reporting delay ( $D$ ) with intensity,  $\lambda_D$ . This is just the independence log-linear model, which can be fitted as

DELAY + QUARTER where DELAY and QUARTER are appropriate factor variables.

*Note there are missing cells and the triangle of missing data is weighted out in order to be able to obtain predictions of total incidence for the recent quarters. A poor fit of this model would indicate non-stationarity of the delay intensities over time.*

- i Fit such a model, comment on the fit and plot the estimates Aids incidence against time.
- ii Because of the missing triangle, a complete non-stationary model, the saturated model of Equation (DELAY\*QUARTER), cannot be fitted . Why?
- iii One possible non-stationary model is  
 DELAY + QUARTER + DELAYL QUARTER + QUARTL DELAY  
 where DELAYL and QUARTL are linear, instead of factor, variables. DELAYL uses centres of three month quarterly periods, but with 0.1 for the no reporting delay (to allow for logarithms below). How does this model compare with the one fitted in (i)? Add the predictions to your previous plot.

## 0.18 Introduction

We will need to use a statistics program during your course and these notes are designed to get you up and running with one called R. There are three main kinds of statistics program available at UEA. They are

- SPSS - a program supported by the ITSC. It was designed to support the kinds of data analysis that you might need in the social sciences. It is fine for routine analysis.
- Splus - a program which implements the S language, see Chambers and Hastie(1992). This is used by statistics departments and researchers around the world. It allows the user to be much more flexible in their analysis and is much more up to date than SPSS
- R is a free clone of Splus and is available on most machines at UEA. You can download your own copy to run on a personal machine and I would urge you to do so. R has extensive and powerful graphics abilities, that are tightly linked with its analytic abilities. The R system is also developing rapidly and new features and abilities appear every few months.

R is available from the CRAN website as is the comprehensive documentation, Web Pages and Email Lists. For official and contributed documentation, for copies of various versions of R, and for other information, go to

<http://cran.r-project.org>

Details of the R-help list, and of other lists that serve the R community, are available from the web site for the R project at

<http://www.R-project.org/>

R is a functional language, there is a language core that uses standard forms of algebraic notation, allowing the calculations such as  $2+3$ , or  $3^{11}$ . Beyond this, most computation is handled using functions.

### Startup

Start up R, either by clicking on the ikon (PC and Mac) or typing R on a UNIX box, and type `data()` This will give you a list of available data-sets. Suppose we look at the Formaldehyde data. Type `data(Formaldehyde)`

Formaldehyde

This will give you some data. In fact two sets of data! It should look something like

```
> data(Formaldehyde)
> Formaldehyde
  carb optden
1  0.1  0.086
2  0.3  0.269
3  0.5  0.446
4  0.6  0.538
5  0.7  0.626
6  0.9  0.782
```

### Quitting R

The action of quitting from an R session uses the function call `q()`.

#### 0.18.1 Manipulating data

Most of the data sets in `data` are structured in some way so for the moment we will try something simpler. R has lots of ways of simulating data from given distributions, for example

1. `runif(n, min=0, max=1)`

    Gives a vector of length `n` whose elements are random and have a uniform distribution between `min` and `max`.

2. `rnorm(n, mean=0, sd=1)`  
Gives a vector of length `n` whose elements are random and have a Normal distribution mean zero and standard deviation 1.
3. `rgamma(n, shape, scale=1)`  
Gives a vector of length `n` whose elements are random and have a Gamma distribution with scale =1 .
4. `rbinom(n, size, prob)`  
Gives a vector of length `n` whose elements are random and have a Binomial distribution with probability `prob` and size `size`.
5. `rpois(n, lambda)`  
Gives a vector of length `n` whose elements are random and have a Poisson distribution mean `1/lambda`.
6. `rcauchy(n, location = 0, scale = 1)`  
Gives a vector of length `n` whose elements are random and have a Cauchy distribution parameters `location` and `scale`

Thus `runif` generates `n` random numbers from a uniform distribution. If you specify the `max` and a `min` then the range will be `max-min`, otherwise the values of `max` and `min` default to 1 and 0. So to create two `x` and `y` vectors of length 100

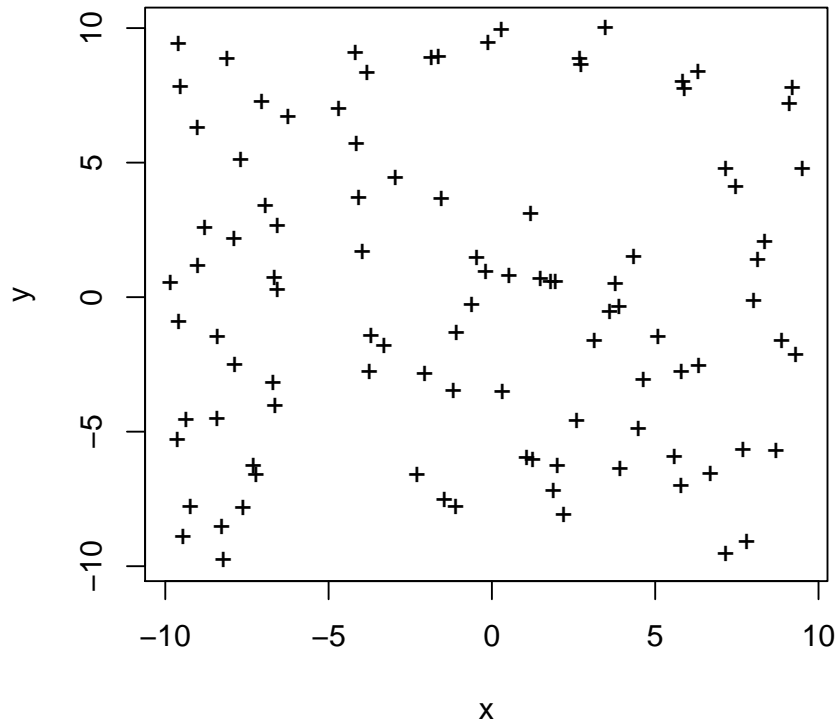
```
x<-runif(100,min=-10,max=10)
y<-runif(100,min=-10,max=10)
```

if you type `x` you can examine the numbers in `x` .

**Note we use a left pointing arrow as the assignment operator.** For versions after 1.4 you can if you prefer use `=`. I prefer to use `←` for compatibility.

If 100 is too many numbers try `x[20:50]` which gives a slice of the vector `x`.

To check the random numbers we could plot `x` and `y`. Try `?plot` or `help(plot)` This will explain the plotting command. You could try `plot(x,y,pch="+")` or `plot(x,y,pch="a")`



The obvious data summaries are the mean, median, standard deviation etc. try these functions

```
mean(x)
sd(x)
median(x)
summary(x)
print() # Prints a single R object
cat() # Prints multiple objects, one after the other
length() # Number of elements in a vector or of a list
unique() # Gives the vector of distinct values
diff() # Replace a vector by the vector of first differences N. B. diff(x) has one less element
order() # x[order(x)] orders elements of x, with NAs last
cumsum()
cumprod()
rev() # reverse the order of vector elements
```

## 0.19 Help

- The `apropos()` command is convenient when you are not sure that you know the name of a function, for example if you were after a stem and leaf function but were not sure if the name was `stem` or `stemandleaf`.
- The `help` command will help us find help on the given function or data-set *once we know the name*. For example `help(stem)` or the abbreviated `?stem` will display the documentation on the `stem` function.

```

stem {graphics} R Documentation
Stem-and-Leaf Plots
Description
stem produces a stem-and-leaf plot of the values in x. The parameter scale can be us
Usage
stem(x, scale = 1, width = 80, atom = 1e-08)
Arguments
x a numeric vector.
scale This controls the plot length.
width The desired width of plot.
atom a tolerance.
References
Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988) The New S Language. Wadsworth
Examples
stem(islands)
stem(log10(islands))

```

- More help is available via a web browser, the command is `help.start()`

### 0.19.1 data input

Getting your own data into R is reasonably easy. For a small amount of data try `c`

```
mydata<-c(12,34,23,39,47)
```

This function combines, or concatenates terms together. As an example, suppose we have the following count of the number of typos per page of these notes:

```
2 3 0 3 1 0 0 1
```

To enter this into an R session we do so with

```
typos<-c(2,3,0,3,1,0,0,1)
```

```
typos
```

```
[1] 2 3 0 3 1 0 0 1
```

We can also concatenate vectors

```
> a<-c(1,2,3)
> b<-c(4,3,2,1,0)
> c(a,b)
[1] 1 2 3 4 3 2 1 0
> d<-c(a,b)
> d
[1] 1 2 3 4 3 2 1 0
```

Notice a few things: We assigned the values to a variable called `typos`. The assignment operator is a `=`. This is valid as of R version 1.4.0, previously it was (and still can be) a

```
<-
```

If you are unfortunate enough to have a missing value you can put a `NA` in its place. R will understand that it denotes a missing value.

### **scan**

if you have more data then you might prefer ( as I do ) to type in data from the keyboard, for example

```
mydata <-scan()
12
34
23 39 47
```

A blank line tells `scan` that input has ended. `?scan` will tell you how to read from a file.

I like to cut data and past into R using `scan`. This can be a little tricky at times since cut and past from different applications may behave oddly - or not at all. If all else fails paste into a word processor like MSWORD. This can be fairly convenient when entering in a few data points (10-40 say), but you might want to use a data file if you have more. You might, as mentioned above, cut and paste from a file to R or you can use the `scan` options

If we have our numbers stored in a text file, then `scan` can be used to read them. You just need to tell `scan` to open the file and read from it . Here are two examples

- Suppose the file `ReadWithScan.txt` has contents 1 2 3 4 Then the command

```
> x <- scan(file = "ReadWithScan.txt")
```

will read the contents into your R session.

- If you had some formatting between the numbers you want to get rid of, say your file `ReadWithScan.txt` is  
1,2,3, 4  
then

```
> x=scan(file = "ReadWithScan.txt",sep=",")
```

is what you need.

- If you are like me and forget where the file is

```
> x <- scan(file =file.choose())
```

will open a dialogue box to help you find your file.

### Data tables - the dataframe

If you want to enter multivariate sets of data, you can do any of the above for each variable. However, it may be more convenient to read in tables of data at once. Suppose you data is in tabular form such as this file `ReadWithReadTable.txt`.

```
Age Weight Height Gender
18 150 65 F
21 160 68 M
45 180 65 M
54 205 69 M
```

Notice the first row supplies column names, the second and following rows the data. The command `read.table` will read this in and store the results in a *data frame*. A data frame is a special matrix where all the variables are stored as columns and each has the same length. Notice we need to specify that the headers are there in this case so we write `header=T` otherwise it is assume that the columns do not have names.

```
> x <-read.table(file="ReadWithReadTable.txt",header=T)

> x[['Gender']] # a factor, it prints the levels [1]
 F M M M
Levels: F M
> x[['Age']] # a numeric vector
[1] 18 21 45 54
> x # default print out for a data.frame
Age Weight Height Gender
1 18 150 65 F
2 21 160 68 M
3 45 180 65 M
4 54 205 69 M
```

Personally I would refer to the variables in a table using `$` as follows

```
> x =read.table(file="ReadWithReadTable.txt",header=T)
> x$Gender # a factor, it prints the levels [1]
```

```

F M M M
Levels: F M
> x$Age # a numeric vector
[1] 18 21 45 54
> x # default print out for a data.frame
Age Weight Height Gender
1 18 150 65 F
2 21 160 68 M
3 45 180 65 M
4 54 205 69 M

```

`read.table` treats the variables as numeric or as factors. A *factor* is special class to R and has a special print method. You can think of a factor as a vector of *categories*. For example

```
(red,blue,blue,red)
```

or

```
(1,2,1,2,2,2,1)
```

The "levels" of the factor are displayed after the values are printed.

If you have Excel files you can always save them as `csv` files. Then they can be read into R using `x<-read.csv(file="fred".header=T)`

### 0.19.2 Smoothing

Try `data()`

This gives a list of available data sets. Load the `cars` data set using the command `data(cars)` If you type `cars` you will see observations on two variables.

You can plot the data using

```
plot(cars$speed,cars$dist) or plot(cars$speed,cars$dist,type="l").
```

If you try `?plot` you can see what plotting options are available.

#### **attach**

Of course the drawback of this is you have to refer to variables by reference to the data set `cars`. Instead of `data$cars` we can tell R about the variables in the data set using `attach(cars)`

This allows you to do without the `cars` prefix so `cars$speed` can be written `speed`.

However my version of R does not overwrite a pre-existing variable so if `speed` already exists it does not become `cars$speed`

#### **lowess**

Often the scatter in a scatter plot tends to obscure the broad pattern. To overcome this statisticians will smooth the data. There are many methods for smoothing - we will cheat and use `lowess`, a well known technique as a black box. To overplot a smoothed

version on the original plot we use

```
plot(cars$speed,cars$dist) # this plots the points
lines(lowess(cars$speed,cars$dist)) # overplots a smooth line
```

The `modreg` library, which can be loaded by `library(modreg)` contains several suggestions for smoothers, for example

```
library(modreg)
data(cars)
attach(cars)
scatter.smooth(speed, dist)
detach()
```

We can have multiple plots on the one page. The parameter `mfrow` can be used to configure the graphics sheet so that subsequent plots appear row by row, one after the other in a rectangular layout, on the one page. For a column by column layout, use `mfc` instead. In the example below we present four different transformations of the primates data, in a two by two layout:

```
par(mfrow=c(2,2), pch=16)
data(Animals) # Needed if Animals (MASS package) is not already loaded
attach(Animals)
plot(body, brain)
plot(sqrt(body), sqrt(brain))
plot((body)^0.1, (brain)^0.1)
plot(log(body),log(brain))
detach(Animals)
par(mfrow=c(1,1),pch=1) # Restore to 1 figure per page
```

## Some Exercises

1. Generate a sample of size 10 from a normal distribution and find the order statistics.

Commands : `rnorm`, `sort`

2. The minimum from a sample of size  $k$  from a standard normal distribution can be obtained by `min(rnorm(12))`. If you define `t` to be a vector of 500 then you can generate 500 minimums from a normal distribution by

```
t<-seq(500)
for (i in (1:500)){t[i]<-min(rnorm(12))}
```

Does the distribution of the minimum look normal? You might find `plot(density(t))` and `density` useful.

Repeat the exercise for the maximum of a uniform distribution.

Commands `runif`, `max`

3. Generate the order statistics from a random sample of size  $N$  taken from a uniform distribution. Plot  $j/N$  vs  $x_j$ . Now generate the order statistics from a random sample of size  $N$  from a normal distribution. Plot  $j/N$  vs  $x_j$ .

Now let  $y_j = \Phi^{-1}(x_j)$  where  $\Phi(x)$  is the cumulative distribution function of a standard normal variate.

Plot  $j/N$  vs  $y_j$ .

Repeat the above when  $y_j = \Phi^{-1}(u_j)$  where  $u_j$  are order statistics from a uniform distribution which the same mean and variance as the original  $X$  distribution.

Commands `rnorm`, `pnorm`, `qnorm`

4. Which has the smaller variance for an exponential distribution. The sample mean or the sample median?

Commands `rexp`

5. The table below gives the total catch of anchovies off Chile and the price ( per tonne) of the catch in 1965 US dollars.

Year	1965	1966	1967	1968	1969	1970	1971
Price	190	160	134	129	172	197	167
Catch	7.23	8.53	9.82	10.96	8.96	12.27	10.28
Year	1972	1973	1974	1975	1976	1977	1978
Price	239	542	372	245	376	454	410
Catch	4.45	1.78	4	3.3	4.3	.08	.5

Compute the correlations between the variables - try `?cor`. Can one say that the correlation between price and catch is non-significant?

6. The command

```
x<-seq(30)
```

generates a vector containing the numbers 1 to 30. Suppose we now try

```
y<-2+11*x+rnorm(30,0,34)
```

What do you expect to be the relationship between (x) and (y)?

The command

```
r1<-lm(y~x)
```

produces a regression the results of which are held in `r1`. Try `summary(r1)` and compare the results with what you expect. Try

```
plot(x,y)
lines(x,r1$fitted)
```

Examine the residuals `r1$resid`

7. The data at <http://www.uea.ac.uk/~gj/nunion/wool.dat> gives the number of cycles to failure of wool yarn. Find a linear regression model for this data. How can you examine the fit of your model?

sectionFitting Linear Models: lm R Documentation

### 0.19.3 Description

`lm` is used to fit linear models. It can be used to carry out regression, single stratum analysis of variance and analysis of covariance (although `aov` may provide a more convenient interface for these).

#### Usage

```
lm(formula, data, subset, weights, na.action,
    method = "qr", model = TRUE, x = FALSE, y = FALSE, qr = TRUE,
    singular.ok = TRUE, contrasts = NULL, offset, ...)
```

## Arguments

- **formula** - a symbolic description of the model to be fit. The details of model specification are given below.
- **data** - an optional data frame containing the variables in the model. If not found in data, the variables are taken from environment(formula), typically the environment from which lm is called.
- **subset** - an optional vector specifying a subset of observations to be used in the fitting process.
- **weights** - an optional vector of weights to be used in the fitting process. If specified, weighted least squares is used with weights weights (that is, minimizing  $\sum(w * e^2)$ ); otherwise ordinary least squares is used.
- **na.action** - a function which indicates what should happen when the data contain NAs. The default is set by the na.action setting of options, and is na.fail if that is unset. The factory-fresh default is `na.omit`. Another possible value is NULL, no action.
- **method** - the method to be used; for fitting, currently only method = "qr" is supported; method = "model.frame" returns the model frame (the same as with model = TRUE, see below). model, x, y, qr logicals. If TRUE the corresponding components of the fit (the model frame, the model matrix, the response, the QR decomposition) are returned.
- **singular** - .ok logical. If FALSE (the default in S but not in R) a singular fit is an error. contrasts an optional list. See the contrasts.arg of model.matrix.default.
- **offset** - this can be used to specify an a priori known component to be included in the linear predictor during fitting. An offset term can be included in the formula instead or as well, and if both are specified their sum is used.
- ... additional arguments to be passed to the low level regression fitting functions (see below).

## Details

Models for `lm` are specified symbolically. A typical model has the form `response ~ terms` where `response` is the (numeric) response vector and `terms` is a series of terms which specifies a linear predictor for response. A terms specification of the form

`first + second`

indicates all the terms in `first` together with all the terms in `second` with duplicates removed. A specification of the form

`first:second`

indicates the set of terms obtained by taking the interactions of all terms in `first` with

all terms in second. The specification  
`first*second`

indicates the cross of first and second. This is the same as  
`first + second + first:second`.

If response is a matrix a linear model is fitted to each column of the matrix. See `model.matrix` for some further details. The terms in the formula will be re-ordered so that main effects come first, followed by the interactions, all second-order, all third-order and so on: to avoid this pass a `terms` object as the formula.

A formula has an implied intercept term. To remove this use either  $y \sim x - 1$  or  $y \sim 0 + x$ . See `formula` for more details of allowed formulae.

`lm` calls the lower level functions `lm.fit`, etc, see below, for the actual numerical computations. For programming only, you may consider doing likewise.

All of `weights`, `subset` and `offset` are evaluated in the same way as variables in formula, that is first in data and then in the environment of formula.

## Value

`lm` returns an object of class "lm" or for multiple responses of class `c("mlm", "lm")`. The functions `summary` and `anova` are used to obtain and print a summary and analysis of variance table of the results. The generic accessor functions `coefficients`, `effects`, `fitted.values` and `residuals` extract various useful features of the value returned by `lm`. An object of class "lm" is a list containing at least the following components:

- `coefficients` a named vector of coefficients
- `residuals` the residuals, that is response minus fitted values.
- `fitted.values` the fitted mean values.
- `rank` the numeric rank of the fitted linear model.
- `weights` (only for weighted fits) the specified weights.
- `df.residual` the residual degrees of freedom.
- `call` the matched call.
- `terms` the terms object used.
- `contrasts` (only where relevant) the contrasts used.
- `xlevels` (only where relevant) a record of the levels of the factors used in fitting.
- `y` if requested, the response used.
- `x` if requested, the model matrix used.
- `model` if requested (the default), the model frame used.

In addition, non-null fits will have components `assign`, `effects` and (unless not requested) `qr` relating to the linear fit, for use by extractor functions such as `summary` and `effects`.

### Using time series

Considerable care is needed when using `lm` with time series.

Unless `na.action = NULL`, the time series attributes are stripped from the variables before the regression is done. (This is necessary as omitting NAs would invalidate the time series attributes, and if NAs are omitted in the middle of the series the result would no longer be a regular time series.)

Even if the time series attributes are retained, they are not used to line up series, so that the time shift of a lagged or differenced regressor would be ignored. It is good practice to prepare a data argument by `ts.intersect(..., dframe = TRUE)`, then apply a suitable `na.action` to that data frame and call `lm` with `na.action = NULL` so that residuals and fitted values are time series.

### Note

Offsets specified by `offset` will not be included in predictions by `predict.lm`, whereas those specified by an `offset` term in the formula will be.

### Author(s)

The design was inspired by the `S` function of the same name described in Chambers (1992). The implementation of model formula by Ross Ihaka was based on Wilkinson and Rogers (1973).

### Examples

```
## Annette Dobson (1990) "An Introduction to \index{generalized linear model} generalized
## Page 9: Plant Weight Data.
ctl <- c(4.17,5.58,5.18,6.11,4.50,4.61,5.17,4.53,5.33,5.14)
trt <- c(4.81,4.17,4.41,3.59,5.87,3.83,6.03,4.89,4.32,4.69)
group <- gl(2,10,20, labels=c("Ctl","Trt"))
weight <- c(ctl, trt)
anova(lm.D9 <- lm(weight ~ group))
summary(lm.D90 <- lm(weight ~ group - 1))# omitting intercept
summary(resid(lm.D9) - resid(lm.D90)) #- residuals almost identical

opar <- par(mfrow = c(2,2), oma = c(0, 0, 1.1, 0))
plot(lm.D9, las = 1)      # Residuals, Fitted, ...
par(opar)

## model frame :
```

```
stopifnot(identical(lm(weight ~ group, method = "model.frame"),
                    model.frame(lm.D9)))
```

## Beer and Babies

Take the following example where I have a file with beer and deaths data

```
> beer<-read.table(file.choose(),header=T,sep="&") # read in data table
> beer
```

	Year	Beer	Deaths
1	1936	62	23
2	1937	61	25
3	1938	55	25
4	1939	53	26
5	1940	60	26
6	1941	63	29
7	1942	53	30
8	1943	52	30
9	1944	48	32
10	1945	49	33
11	1946	43	31

```
> plot(beer$Beer,beer$Deaths,xlab="beer",ylab="deaths")
> attach(beer)
> r1<-lm(Deaths~Beer)
> summary(r1)
```

Call:

```
lm(formula = Deaths ~ Beer)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-2.9804	-1.9037	-0.1337	1.3577	3.9743

Coefficients:

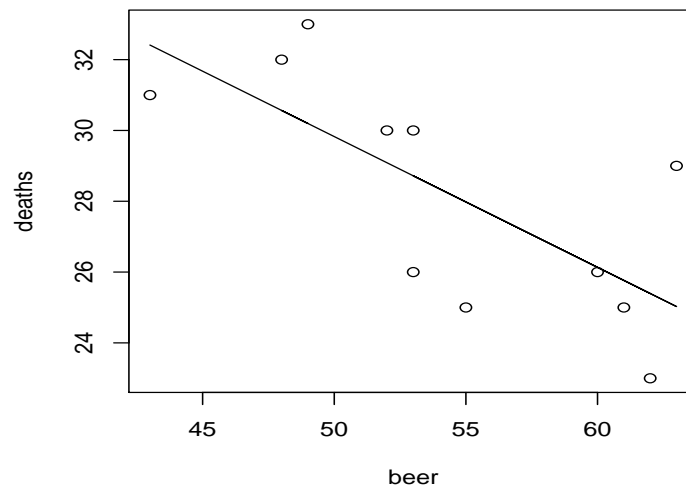
	Estimate	Std. Error	t value	Pr(> t )	
(Intercept)	48.2934	6.5018	7.428	3.98e-05	***
Beer	-0.3693	0.1186	-3.113	0.0125	*

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.422 on 9 degrees of freedom  
Multiple R-Squared: 0.5185, Adjusted R-squared: 0.465  
F-statistic: 9.69 on 1 and 9 DF, p-value: 0.01246

```
> lines(Beer,r1$fitted)
> plot(Beer,r1$resid)
> abline(h=0)
```



### More babies

```
> babies<-read.table(file.choose(),header=T)
> babies
  age weight sex
1  40  2968  b
2  38  2795  b
3  40  3163  b
4  35  2925  b
5  36  2625  b
6  37  2847  b
7  41  3292  b
8  40  3473  b
9  37  2628  b
10 38  3176  b
11 40  3421  b
12 38  2975  b
13 40  3317  g
14 36  2729  g
15 40  2935  g
16 38  2754  g
17 42  3210  g
18 39  2817  g
19 40  3126  g
20 37  2539  g
21 36  2412  g
22 38  2991  g
23 39  2875  g
24 40  3231  g
> attach(babies)
> sex<-factor(sex)
> sex
[1] b b b b b b b b b b b b b g g g g g g g g g g g g
Levels: b g
> lab<-as.character(sex)
> lab
[1] "b" "b" "b" "b" "b" "b" "b" "b" "b" "b" "b" "b" "b" "g" "g" "g" "g" "g" "g" "g" "g" "g"
[23] "g" "g"
> plot(age,weight,pch=lab)
> r1<-lm(weight~age+sex)
> summary(r1)
```

Call:

```
lm(formula = weight ~ age + sex)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-257.49	-125.28	-58.44	169.00	303.98

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	-1610.28	786.08	-2.049	0.0532 .
age	120.89	20.46	5.908	7.28e-06 ***
sexg	-163.04	72.81	-2.239	0.0361 *

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 177.1 on 21 degrees of freedom

Multiple R-Squared: 0.64, Adjusted R-squared: 0.6057

F-statistic: 18.67 on 2 and 21 DF, p-value: 2.194e-05

> anova(r1)

Analysis of Variance Table

Response: weight

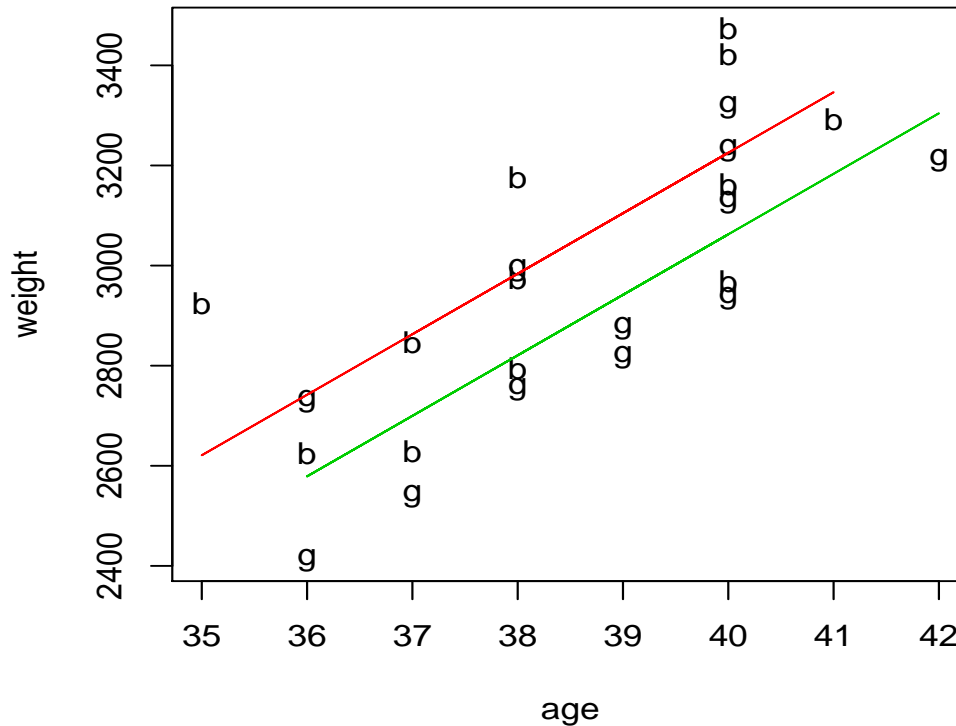
	Df	Sum Sq	Mean Sq	F value	Pr(>F)
age	1	1013799	1013799	32.3174	1.213e-05 ***
sex	1	157304	157304	5.0145	0.03609 *
Residuals	21	658771	31370		

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

> lines(age[1:12],r1\$fitted[1:12],col=2)

> lines(age[13:24],r1\$fitted[13:24],col=3)



## 0.20 Fitting generalized linear models: glm stats R Documentation

### Description

`glm` is used to fit generalized linear models, specified by giving a symbolic description of the linear predictor and a description of the error distribution.

### Usage

```
glm(formula, family = gaussian, data, weights, subset,
    na.action, start = NULL, etastart, mustart,
    offset, control = glm.control(...), model = TRUE,
    method = "glm.fit", x = FALSE, y = TRUE, contrasts = NULL, ...)
```

```
glm.fit(x, y, weights = rep(1, nobs),
```

```
start = NULL, etastart = NULL, mustart = NULL,  
offset = rep(0, nobs), family = gaussian(),  
control = glm.control(), intercept = TRUE)  
  
## S3 method for class 'glm':  
weights(object, type = c("prior", "working"), ...)
```

### Arguments

- **formula** - a symbolic description of the model to be fit. The details of model specification are given below.
- **family** - a description of the error distribution and link function to be used in the model. This can be a character string naming a family function, a family function or the result of a call to a family function. (See family for details of family functions.)
- **data** - an optional data frame containing the variables in the model. If not found in data, the variables are taken from environment(formula), typically the environment from which glm is called.
- **weights** - an optional vector of weights to be used in the fitting process.
- **subse** -t an optional vector specifying a subset of observations to be used in the fitting process.
- **na.action** - a function which indicates what should happen when the data contain NAs. The default is set by the na.action setting of options, and is na.fail if that is unset. The factory-fresh default is na.omit.
- **start** -starting values for the parameters in the linear predictor.
- **etastart** - starting values for the linear predictor.
- **mustart** - starting values for the vector of means.
- **offset** - this can be used to specify an a priori known component to be included in the linear predictor during fitting.
- **control** - a list of parameters for controlling the fitting process. See the documentation for glm.control for details.
- **model** - a logical value indicating whether model frame should be included as a component of the returned value.

- **method** - the method to be used in fitting the model. The default method "glm.fit" uses iteratively reweighted least squares (IWLS). The only current alternative is "model.frame" which returns the model frame and does no fitting.
- **x, y** - For glm: logical values indicating whether the response vector and model matrix used in the fitting process should be returned as components of the returned value. For glm.fit: x is a design matrix of dimension n \* p, and y is a vector of observations of length n.
- **contrasts** -an optional list. See the contrasts.arg of model.matrix.default.
- **object** - an object inheriting from class "glm".
- **type** - character, partial matching allowed. Type of weights to extract from the fitted model object.
- **intercept** - logical. Should an intercept be included?
- ... - further arguments passed to or from other methods.

### Details

A typical predictor has the form  $\text{response} \sim \text{terms}$  where response is the (numeric) response vector and terms is a series of terms which specifies a linear predictor for response. For binomial models the response can also be specified as a factor (when the first level denotes failure and all others success) or as a two-column matrix with the columns giving the numbers of successes and failures. A terms specification of the form  $\text{first} + \text{second}$

indicates all the terms in first together with all the terms in second with duplicates removed. The terms in the formula will be re-ordered so that main effects come first, followed by the interactions, all second-order, all third-order and so on: to avoid this pass a terms object as the formula.

A specification of the form

$\text{first}:\text{second}$

indicates the the set of terms obtained by taking the interactions of all terms in first with all terms in second. The specification

$\text{first}*\text{second}$

indicates the cross of first and second. This is the same as  $\text{first} + \text{second} + \text{first}:\text{second}$ .

`glm.fit` and `glm.fit.null` are the workhorse functions: the former calls the latter for a null model (with no intercept).

If more than one of `etastart`, `start` and `mustart` is specified, the first in the list will be used.

All of `weights`, `subset`, `offset`, `etastart` and `mustart` are evaluated in the same way as variables in formula, that is first in data and then in the environment of formula.

## Value

`glm` returns an object of class inheriting from "glm" which inherits from the class "lm". See later in this section. The function `summary` (i.e., `summary.glm`) can be used to obtain or print a summary of the results and the function `anova` (i.e., `anova.glm`) to produce an analysis of variance table. The generic accessor functions `coefficients`, `effects`, `fitted.values` and `residuals` can be used to extract various useful features of the value returned by `glm`. `weights` extracts a vector of weights, one for each case in the fit (after subsetting and `na.action`). An object of class "glm" is a list containing at least the following components:

- `coefficients` - a named vector of coefficients
- `residuals` - the working residuals, that is the residuals in the final iteration of the IWLS fit.
- `fitted.values` - the fitted mean values, obtained by transforming the linear predictors by the inverse of the link function.
- `rank` - the numeric rank of the fitted linear model.
- `family` - the family object used.
- `linear.predictors` - the linear fit on link scale.
- `deviance` - up to a constant, minus twice the maximized log-likelihood. Where sensible, the constant is chosen so that a saturated model has deviance zero.
- `aic` - Akaike's An Information Criterion, minus twice the maximized log-likelihood plus twice the number of coefficients (so assuming that the dispersion is known).
- `null.deviance` - The deviance for the null model, comparable with deviance. The null model will include the offset, and an intercept if there is one in the model
- `iter` - the number of iterations of IWLS used.
- `weights` - the working weights, that is the weights in the final iteration of the IWLS fit.
- `prior.weights` - the case weights initially supplied.
- `df.residual` - the residual degrees of freedom.
- `df.null` - the residual degrees of freedom for the null model.
- `y` - the y vector used. (It is a vector even for a binomial model.)
- `converged` - logical. Was the IWLS algorithm judged to have converged?
- `boundary` - logical. Is the fitted value on the boundary of the attainable values?

- `call` - the matched call.
- `formula` - the formula supplied.
- `terms` - the terms object used.
- `data` - the data argument.
- `offset` - the offset vector used.
- `control` - the value of the control argument used.
- `method` - the name of the fitter function used, in R always "glm.fit". contrasts (where relevant) the contrasts used.
- `xlevels` - (where relevant) a record of the levels of the factors used in fitting.

In addition, non-empty fits will have components `qr`, `R` and effects relating to the final weighted linear fit. Objects of class "glm" are normally of class `c("glm", "lm")`, that is inherit from class "lm", and well-designed methods for class "lm" will be applied to the weighted linear model at the final iteration of IWLS. However, care is needed, as extractor functions for class "glm" such as residuals and weights do not just pick out the component of the fit with the same name. If a binomial glm model is specified by giving a two-column response, the weights returned by `prior.weights` are the total numbers of cases (factored by the supplied case weights) and the component `y` of the result is the proportion of successes.

### Author(s)

The original R implementation of `glm` was written by Simon Davies working for Ross Ihaka at the University of Auckland, but has since been extensively re-written by members of the R Core team.

The design was inspired by the S function of the same name described in Hastie and Pregibon (1992).

### Examples

```
## Dobson (1990) Page 93: Randomized Controlled Trial :
counts <- c(18,17,15,20,10,20,25,13,12)
outcome <- gl(3,1,9)
treatment <- gl(3,3)
print(d.AD <- data.frame(treatment, outcome, counts))
glm.D93 <- glm(counts ~ outcome + treatment, family=poisson())
anova(glm.D93)
summary(glm.D93)
```

```
## an example with offsets from Venables & Ripley (2002, p.189)

## Not run:
## Need the anorexia data from a recent version of the package 'MASS':
library(MASS)
## End(Not run)
anorex.1 <- glm(Postwt ~ Prewt + Treat + offset(Prewt),
               family = gaussian, data = anorexia)
summary(anorex.1)

# A Gamma example, from McCullagh & Nelder (1989, pp. 300-2)
clotting <- data.frame(
  u = c(5,10,15,20,30,40,60,80,100),
  lot1 = c(118,58,42,35,27,25,21,19,18),
  lot2 = c(69,35,26,21,18,16,13,12,12))
summary(glm(lot1 ~ log(u), data=clotting, family=Gamma))
summary(glm(lot2 ~ log(u), data=clotting, family=Gamma))

## Not run:
## for an example of the use of a terms object as a formula
demo(glm.vr)
## End(Not run)
[Package stats version 2.0.1 Index]
```

## 0.21 Publications related to R

This page gives a partially annotated list of books and other publications that are related to S or R and may be useful - not this is taken from CRAN.

1. RichardA. Becker, JohnM. Chambers, and AllanR. Wilks. The New S Language. Chapman and Hall, London, 1988.  
This book is often called the Blue Book, and introduced what is now known as S version 2.
2. JohnM. Chambers and TrevorJ. Hastie. Statistical Models in S. Chapman and Hall, London, 1992.  
This is also called the White Book, and introduced S version 3, which added structures to facilitate statistical modeling in S.
3. JohnM. Chambers. Programming with Data. Springer, New York, 1998. ISBN 0-387-98503-4.  
This Green Book describes version 4 of S, a major revision of S designed by John Chambers to improve its usefulness at every stage of the programming process.

4. WilliamN. Venables and BrianD. Ripley. *Modern Applied Statistics with S*. Fourth Edition. Springer, New York, 2002. ISBN 0-387-95457-0.  
A highly recommended book on how to do statistical data analysis using R or S-Plus. In the first chapters it gives an introduction to the S language. Then it covers a wide range of statistical methodology, including linear and generalized linear models, non-linear and smooth regression, tree-based methods, random and mixed effects, exploratory multivariate analysis, classification, survival analysis, time series analysis, spatial statistics, and optimization. The ‘on-line complements’ available at the books homepage provide updates of the book, as well as further details of technical material.
5. WilliamN. Venables and BrianD. Ripley. *S Programming*. Springer, New York, 2000. ISBN 0-387-98966-8.  
This provides an in-depth guide to writing software in the S language which forms the basis of both the commercial S-Plus and the Open Source R data analysis software systems.
6. Deborah Nolan and Terry Speed. *Stat Labs: Mathematical Statistics Through Applications*. Springer Texts in Statistics. Springer, 2000. ISBN 0-387-98974-9.  
Integrates theory of statistics with the practice of statistics through a collection of case studies (labs), and uses R to analyze the data.
7. JoseC. Pinheiro and DouglasM. Bates. *Mixed-Effects Models in S and S-Plus*. Springer, 2000. ISBN 0-387-98957-0.  
A comprehensive guide to the use of the ‘nlme’ package for linear and nonlinear mixed-effects models.
8. FrankE. Harrell. *Regression Modeling Strategies, with Applications to Linear Models, Survival Analysis and logistic Regression*. Springer, 2001. ISBN 0-387-95232-2.  
There are many books that are excellent sources of knowledge about individual statistical tools (survival models, general linear models, etc.), but the art of data analysis is about choosing and using multiple tools. In the words of Chatfield ...students typically know the technical details of regression for example, but not necessarily when and how to apply it. This argues the need for a better balance in the literature and in statistical teaching between techniques and problem solving strategies. Whether analyzing risk factors, adjusting for biases in observational studies, or developing predictive models, there are common problems that few regression texts address. For example, there are missing data in the majority of datasets one is likely to encounter (other than those used in textbooks!) but most regression texts do not include methods for dealing with such data effectively, and texts on missing data do not cover regression modeling.
9. ManuelCastejn Limas, JoaquinOrdieres Mer, Fco.Javier deCosJuez, and Fco. JavierMartnez dePisnAscacibar. *Control de Calidad. Metodologia para el analisis*

previo a la modelizacin de datos en procesos industriales. Fundamentos tericos y aplicaciones con R. Servicio de Publicaciones de la Universidad de La Rioja, 2001. ISBN 84-95301-48-2.

This book, written in Spanish, is oriented to researchers interested in applying multivariate analysis techniques to real processes. It combines the theoretical basis with applied examples coded in R.

10. John Fox. An R and S-Plus Companion to Applied Regression. Sage Publications, Thousand Oaks, CA, USA, 2002. ISBN 0-761-92279-2.  
A companion book to a text or course on applied regression (such as Applied Regression, Linear Models, and Related Methods by the same author). It introduces S, and concentrates on how to use linear and generalized-linear models in S while assuming familiarity with the statistical methodology.
11. Peter Dalgaard. Introductory Statistics with R. Springer, 2002. ISBN 0-387-95475-9.
12. Stefano Iacus and Guido Masarotto. Laboratorio di statistica con R. McGraw-Hill, Milano, 2003. ISBN 88-386-6084-0.
13. John Maindonald and John Braun. Data Analysis and Graphics Using R. Cambridge University Press, Cambridge, 2003. ISBN 0-521-81336-0.
14. Giovanni Parmigiani, ElizabethS. Garrett, RafaelA. Irizarry, and ScottL. Zeger. The Analysis of Gene Expression Data. Springer, New York, 2003. ISBN 0-387-95577-1
15. Sylvie Huet, Annie Bouvier, Marie-Anne Gruet, and Emmanuel Jolivet. Statistical Tools for Nonlinear Regression. Springer, New York, 2003. ISBN 0-387-40081-8
16. S.Mase, T.Kamakura, M.Jimbo, and K.Kanefuji. Introduction to Data Science for engineers- Data analysis using free statistical software R (in Japanese). Suuri-Kogaku-sha, Tokyo, April 2004. ISBN 4901683128
17. JulianJ. Faraway. Linear Models with R. Chapman and Hall/CRC, Boca Raton, FL, 2004. ISBN 1-584-88425-8  
The book focuses on the practice of regression and analysis of variance. It clearly demonstrates the different methods available and in which situations each one applies. It covers all of the standard topics, from the basics of estimation to missing data, factorial designs, and block designs, but it also includes discussion of topics, such as model uncertainty, rarely addressed in books of this type. The presentation incorporates an abundance of examples that clarify both the use of each technique and the conclusions one can draw from the results.
18. RichardM. Heiberger and Burt Holland. Statistical Analysis and Data Display: An Intermediate Course with Examples in S-Plus, R, and SAS. Springer Texts in Statistics. Springer, 2004. ISBN 0-387-40270-5

A contemporary presentation of statistical methods featuring 200 graphical displays for exploring data and displaying analyses. Many of the displays appear here for the first time. Discusses construction and interpretation of graphs, principles of graphical design, and relation between graphs and traditional tabular results. Can serve as a graduate-level standalone statistics text and as a reference book for researchers. In-depth discussions of regression analysis, analysis of variance, and design of experiments are followed by introductions to analysis of discrete bivariate data, nonparametrics, logistic regression, and ARIMA time series modeling. Concepts and techniques are illustrated with a variety of case studies. S-Plus, R, and SAS executable functions are provided and discussed. S functions are provided for each new graphical display format. All code, transcript and figure files are provided for readers to use as templates for their own analyses.

19. John Verzani. *Using R for Introductory Statistics*. Chapman and Hall/CRC, Boca Raton, FL, 2005. ISBN 1-584-88450-9  
There are few books covering introductory statistics using R, and this book fills a gap as a true beginner book. With emphasis on data analysis and practical examples, ‘Using R for Introductory Statistics’ encourages understanding rather than focusing on learning the underlying theory. It includes a large collection of exercises and numerous practical examples from a broad range of scientific disciplines. It comes complete with an online resource containing datasets, R functions, selected solutions to exercises, and updates to the latest features. A full solutions manual is available from Chapman and Hall/CRC.
20. Fionn Murtagh. *Correspondence Analysis and Data Coding with JAVA and R*. Chapman and Hall/CRC, Boca Raton, FL, 2005. ISBN 1-584-88528-9  
This book provides an introduction to methods and applications of correspondence analysis, with an emphasis on data coding - the first step in correspondence analysis. It features a practical presentation of the theory with a range of applications from data mining, financial engineering, and the biosciences. Implementation of the methods is presented using JAVA and R software.
21. Paul Murrell. *R Graphics*. Chapman and Hall/CRC, Boca Raton, FL, 2005. ISBN 1-584-88486-X  
A description of the core graphics features of R including: a brief introduction to R; an introduction to general R graphics features. The base graphics system of R: traditional S graphics. The power and flexibility of grid graphics. Building on top of the base or grid graphics: Trellis graphics and developing new graphics functions.
22. Michael J. Crawley. *Statistics: An Introduction using R*. Wiley, 2005. ISBN 0-470-02297-3  
The book is primarily aimed at undergraduate students in medicine, engineering, economics and biology - but will also appeal to postgraduates who have not previously covered this area, or wish to switch to using R.

23. Brian S. Everitt. *An R and S-Plus Companion to Multivariate Analysis*. Springer, 2005. ISBN 1-85233-882-2  
In this book the core multivariate methodology is covered along with some basic theory for each method described. The necessary R and S-Plus code is given for each analysis in the book, with any differences between the two highlighted.
24. Richard C. Deonier, Simon Tavar, and Michael S. Waterman. *Computational Genome Analysis: An Introduction*. Springer, 2005. ISBN: 0-387-98785-1  
*Computational Genome Analysis: An Introduction* presents the foundations of key problems in computational molecular biology and bioinformatics. It focuses on computational and statistical principles applied to genomes, and introduces the mathematics and statistics that are crucial for understanding these applications. All computations are done with R.
25. Robert Gentleman, Vince Carey, Wolfgang Huber, Rafael Irizarry, and Sandrine Dudoit, editors. *Bioinformatics and Computational Biology Solutions Using R and Bioconductor*. *Statistics for Biology and Health*. Springer, 2005. ISBN: 0-387-25146-4  
This volume's coverage is broad and ranges across most of the key capabilities of the Bioconductor project, including importation and preprocessing of high-throughput data from microarray, proteomic, and flow cytometry platforms.
26. Terry M. Therneau and Patricia M. Grambsch. *Modeling Survival Data: Extending the Cox Model*. *Statistics for Biology and Health*. Springer, 2000. ISBN: 0-387-98784-3  
This is a book for statistical practitioners, particularly those who design and analyze studies for survival and event history data. Its goal is to extend the toolkit beyond the basic triad provided by most statistical packages: the Kaplan-Meier estimator, log-rank test, and Cox regression model.
27. Brian Everitt and Torsten Hothorn. *A Handbook of Statistical Analyses Using R*. Chapman and Hall/CRC, Boca Raton, FL, 2006. ISBN 1-584-88539-4  
With emphasis on the use of R and the interpretation of results rather than the theory behind the methods, this book addresses particular statistical techniques and demonstrates how they can be applied to one or more data sets using R. The authors provide a concise introduction to R, including a summary of its most important features. They cover a variety of topics, such as simple inference, generalized linear models, multilevel models, longitudinal data, cluster analysis, principal components analysis, and discriminant analysis. With numerous figures and exercises, *A Handbook of Statistical Analysis using R* provides useful information for students as well as statisticians and data analysts.
28. Julian J. Faraway. *Extending Linear Models with R: Generalized Linear, Mixed Effects and Nonparametric Regression Models*. Chapman and Hall/CRC, Boca Raton, FL, 2006. ISBN 1-584-88424-X

This book surveys the techniques that grow from the regression model, presenting three extensions to that framework: generalized linear models (GLMs), mixed effect models, and nonparametric regression models. The author's treatment is thoroughly modern and covers topics that include GLM diagnostics, generalized linear mixed models, trees, and even the use of neural networks in statistics. To demonstrate the interplay of theory and practice, throughout the book the author weaves the use of the R software environment to analyze the data of real examples, providing all of the R commands necessary to reproduce the analyses.

29. Jana Jureckova and Jan Picek. *Robust Statistical Methods with R*. Chapman and Hall/CRC, Boca Raton, FL, 2006. ISBN 1-584-88454-1

This book provides a systematic treatment of robust procedures with an emphasis on practical application. The authors work from underlying mathematical tools to implementation, paying special attention to the computational aspects. They cover the whole range of robust methods, including differentiable statistical functions, distance of measures, influence functions, and asymptotic distributions, in a rigorous yet approachable manner. Highlighting hands-on problem solving, many examples and computational algorithms using the R software supplement the discussion. The book examines the characteristics of robustness, estimators of real parameter, large sample properties, and goodness-of-fit tests. It also includes a brief overview of R in an appendix for those with little experience using the software.

30. Simon N. Wood. *Generalized Additive Models: An Introduction with R*. Chapman and Hall/CRC, Boca Raton, FL, 2006. ISBN 1-584-88474-6

This book imparts a thorough understanding of the theory and practical applications of GAMs and related advanced models, enabling informed use of these very flexible tools. The author bases his approach on a framework of penalized regression splines, and builds a well-grounded foundation through motivating chapters on linear and generalized linear models. While firmly focused on the practical aspects of GAMs, discussions include fairly full explanations of the theory underlying the methods. The treatment is rich with practical examples, and it includes an entire chapter on the analysis of real data sets using R and the author's add-on package `mgcv`. Each chapter includes exercises, for which complete solutions are provided in an appendix.

31. Bernhard Pfaff. *Analysis of Integrated and Cointegrated Time Series with R*. Use R. Springer, 2006. ISBN 0-387-98784-3

The book encompasses seasonal unit roots, fractional integration, coping with structural breaks, and inference in cointegrated vector autoregressive models.

32. Nhu D. Le and James V. Zidek. *Statistical Analysis of Environmental Space-Time Processes*. Springer, 2006. ISBN 0-387-26209-1

This book provides a broad introduction to the subject of environmental space-time processes, addressing the role of uncertainty. It covers a spectrum of technical

matters from measurement to environmental epidemiology to risk assessment. It showcases non-stationary vector-valued processes, while treating stationarity as a special case. In particular, with members of their research group the authors developed within a hierarchical Bayesian framework, the new statistical approaches presented in the book for analyzing, modeling, and monitoring environmental spatio-temporal processes. Furthermore they indicate new directions for development.

33. Peter J. Diggle and Paulo Justiniano Ribeiro. *Model-based Geostatistics*. Springer, 2006. ISBN 0-387-32907-2  
Geostatistics is concerned with estimation and prediction problems for spatially continuous phenomena, using data obtained at a limited number of spatial locations. The name reflects its origins in mineral exploration, but the methods are now used in a wide range of settings including public health and the physical and environmental sciences. Model-based geostatistics refers to the application of general statistical principles of modeling and inference to geostatistical problems. This volume is the first book-length treatment of model-based geostatistics.
34. Emmanuel Paradis. *Analysis of Phylogenetics and Evolution with R. Use R*. Springer, New York, 2006. ISBN 0-387-32914-5  
This book integrates a wide variety of data analysis methods into a single and flexible interface: the R language, an open source language is available for a wide range of computer systems and has been adopted as a computational environment by many authors of statistical software. Adopting R as a main tool for phylogenetic analyses sease the workflow in biologists' data analyses, ensure greater scientific repeatability, and enhance the exchange of ideas and methodological developments.
35. Sandrine Dudoit and Mark J. van der Laan. *Multiple Testing Procedures and Applications to Genomics*. Springer Series in Statistics. Springer, 2007. ISBN: 978-0-387-49316-9  
This book provides a detailed account of the theoretical foundations of proposed multiple testing methods and illustrates their application to a range of testing problems in genomics.
36. Uwe Ligges. *Programmieren mit R*. Springer-Verlag, Heidelberg, 2nd edition, 2007. ISBN 3-540-36332-7, in German  
R ist eine objekt-orientierte und interpretierte Sprache und Programmierumgebung für Datenanalyse und Grafik - frei erhältlich unter der GPL. Das Buch führt in die Grundlagen der Sprache R ein und vermittelt ein umfassendes Verständnis der Sprachstruktur. Die enormen Grafikfähigkeiten von R werden detailliert beschrieben. Der Leser kann leicht eigene Methoden umsetzen, Objektklassen definieren und ganze Pakete aus Funktionen und zugehöriger Dokumentation zusammenstellen. Ob Diplomarbeit, Forschungsprojekte oder Wirtschaftsdaten, das Buch unterstützt alle, die R als flexibles Werkzeug zur Datenanalyse und -visualisierung einsetzen möchten.

37. Dubravko Dolic. Statistik mit R. Einfhruung fr Wirtschafts- und Sozialwissenschaftler. R. Oldenbourg, Mnchen, Wien, 2004. ISBN 3-486-27537-2, in German
38. Andreas Behr. Einfhruung in die Statistik mit R. WiSo Kurzlehrbcher. Vahlen, Mnchen, 2005. ISBN 3-8006-3219-5, in German
39. ScottM. Lynch. Introduction to Applied Bayesian Statistics and Estimation for Social Scientists. Springer, New York, 2007. ISBN 978-0-387-71264-2  
Introduction to Bayesian Statistics and Estimation for Social Scientists covers the complete process of Bayesian statistical analysis in great detail from the development of a model through the process of making statistical inference. The key feature of this book is that it covers models that are most commonly used in social science research-including the linear regression model, generalized linear models, hierarchical models, and multivariate regression models-and it thoroughly develops each real-data example in painstaking detail.
40. Jim Albert. Bayesian Computation with R. Springer, New York, 2007. ISBN 978-0-387-71384-7  
Bayesian Computation with R introduces Bayesian modeling by the use of computation using the R language. The early chapters present the basic tenets of Bayesian thinking by use of familiar one and two-parameter inferential problems. Bayesian computational methods such as Laplace's method, rejection sampling, and the SIR algorithm are illustrated in the context of a random effects model. The construction and implementation of Markov Chain Monte Carlo (MCMC) methods is introduced. These simulation-based algorithms are implemented for a variety of Bayesian applications such as normal and binary response regression, hierarchical modeling, order-restricted inference, and robust modeling. Algorithms written in R are used to develop Bayesian tests and assess Bayesian models by use of the posterior predictive distribution. The use of R to interface with WinBUGS, a popular MCMC computing language, is described with several illustrative examples.
41. Jean-Michel Marin and ChristianP. Robert. Bayesian Core: A Practical Approach to Computational Bayesian Statistics. Springer, New York, 2007. ISBN 978-0-387-38979-0  
This Bayesian modeling book is intended for practitioners and applied statisticians looking for a self-contained entry to computational Bayesian statistics. Focusing on standard statistical models and backed up by discussed real datasets available from the book website, it provides an operational methodology for conducting Bayesian inference, rather than focusing on its theoretical justifications. Special attention is paid to the derivation of prior distributions in each case and specific reference solutions are given for each of the models. Similarly, computational details are worked out to lead the reader towards an effective programming of the methods given in the book. While R programs are provided on the book website and R hints are given in the computational sections of the book, The Bayesian Core

requires no knowledge of the R language and it can be read and used with any other programming language.

42. Dianne Cook and Deborah F. Swayne. *Interactive and Dynamic Graphics for Data Analysis*. Springer, New York, 2007. ISBN 978-0-387-71761-6

This richly illustrated book describes the use of interactive and dynamic graphics as part of multidimensional data analysis. Chapters include clustering, supervised classification, and working with missing values. A variety of plots and interaction methods are used in each analysis, often starting with brushing linked low-dimensional views and working up to manual manipulation of tours of several variables. The role of graphical methods is shown at each step of the analysis, not only in the early exploratory phase, but in the later stages, too, when comparing and evaluating models. All examples are based on freely available software: GGobi for interactive graphics and R for static graphics, modeling, and programming. The printed book is augmented by a wealth of material on the web, encouraging readers follow the examples themselves. The web site has all the data and code necessary to reproduce the analyses in the book, along with movies demonstrating the examples.

43. David Siegmund and Benjamin Yakir. *The Statistics of Gene Mapping*. Springer, New York, 2007. ISBN 978-0-387-49684-9

This book details the statistical concepts used in gene mapping, first in the experimental context of crosses of inbred lines and then in outbred populations, primarily humans. It presents elementary principles of probability and statistics, which are implemented by computational tools based on the R programming language to simulate genetic experiments and evaluate statistical analyses. Each chapter contains exercises, both theoretical and computational, some routine and others that are more challenging. The R programming language is developed in the text.

44. Lothar Sachs and Jrgen Hedderich. *Angewandte Statistik. Methodensammlung mit R*. Springer, Berlin, Heidelberg, 12th (completely revised) edition, 2006. ISBN 978-3-540-32160-6

Die Anwendung statistischer Methoden wird heute in der Regel durch den Einsatz von Computern untersttzt. Das Programm R ist dabei ein leicht erlernbares und flexibel einzusetzendes Werkzeug, mit dem der Prozess der Datenanalyse nachvollziehbar verstanden und gestaltet werden kann. Diese 12., vollstndig neu bearbeitete Auflage veranschaulicht Anwendung und Nutzen des Programms anhand zahlreicher mit R durchgerechneter Beispiele. Sie erlutert statistische Anstze und gibt leicht fasslich, anschaulich und praxisnah Studenten, Dozenten und Praktikern mit unterschiedlichen Vorkenntnissen die notwendigen Details, um Daten zu gewinnen, zu beschreiben und zu beurteilen. Neben Hinweisen zur Planung und Auswertung von Studien ermöglichen viele Beispiele, Querverweise und ein ausführliches Sachverzeichnis einen gezielten Zugang zur Statistik, insbesondere für Mediziner, Ingenieure und Naturwissenschaftler.

45. Ross Ihaka and Robert Gentleman. R: A language for data analysis and graphics. *Journal of Computational and Graphical Statistics*, 5(3):299-314, 1996
46. Francisco Cribari-Neto and SpyrosG. Zarkos. R: Yet another econometric programming environment. *Journal of Applied Econometrics*, 14:319-329, 1999
47. Robert Gentleman and Ross Ihaka. Lexical scope and statistical computing. *Journal of Computational and Graphical Statistics*, 9:491-508, 2000
48. Paul Murrell and Ross Ihaka. An approach to providing mathematical annotation in plots. *Journal of Computational and Graphical Statistics*, 9:582-599, 2000
49. StephenP. Ellner. Review of R, version 1.1.1. *Bulletin of the Ecological Society of America*, 82(2):127-128, April 2001
50. BrianD. Ripley. The R project in statistical computing. *MSOR Connections*. The newsletter of the LTSN Maths, Stats and OR Network., 1(1):23-25, February 2001
51. Kurt Hornik and Friedrich Leisch, editors. *Proceedings of the 2nd International Workshop on Distributed Statistical Computing (DSC 2001)*, Technische Universitt Wien, Vienna, Austria, 2001. ISSN 1609-395X
52. PauloJ. Ribeiro, Jr. and PatrickE. Brown. Some words on the R project. *The ISBA Bulletin*, 8(1):12-16, March 2001
53. Diego Kuonen. Introduction au data mining avec R : vers la reconqute du ‘knowledge discovery in databases’ par les statisticiens. *Bulletin of the Swiss Statistical Society*, 40:3-7, 2001
54. Diego Kuonen and Reinhard Furrer. Data mining avec R dans un monde libre. *Flash Informatique Spcial t*, pages 45-50, sep 2001
55. Reinhard Furrer and Diego Kuonen. GRASS GIS et R: main dans la main dans un monde libre. *Flash Informatique Spcial t*, pages 51-56, sep 2001
56. Diego Kuonen and Valerie Chavez. R - un exemple du succs des modles libres. *Flash Informatique*, 2:3-7, 2001
57. Vito Ricci. R : un ambiente opensource per l’analisi statistica dei dati. *Economia e Commercio*, 1:69-82, 2004

This paper would be a short introduction and overview about the language and environment for statistical analysis R, without entering in specific details too much computational. I give a look about this opensource software pointing out its main features, its functionalities, its pros and cons describing some libraries and the kind of analysis they support. I supply a summary, with a short description, about many resources concerning R that can be found in the Web: the most are in English language, but there are also some in the Italian language. The aim of this work is to contribute in increasing of the use of the R environment in Italy

among statistical researchers trying to advertise this software and its opensource philosophy.

58. Vito Ricci. Rappresentazione analitica delle distribuzioni statistiche con R (prima parte). *Economia e Commercio*, 1/2:47-60, 2005

This paper deals with distribution fitting using R environment for statistical computing. It treats briefly some theoretical issues and it points out especially practical ones proposing some examples of R statements for data graphical exploration and presentation, parameters' estimates of patterns and tests for goodness of fit.

# Index

Box-Cox, 10

canonical link, 12, 13, 26, 28, 52

deviance, 15–17, 20, 25–27, 43, 45, 47, 51–  
53, 56, 61, 70, 72, 80, 82, 84–86,  
89–91, 118

dispersion parameter, 11, 15

exponential family, 11, 12, 16, 51, 69, 83

factor, 6

generalized linear model, 1, 10, 12, 13, 15,  
30, 51, 73, 74, 94, 115, 121, 124,  
125, 127

log-linear, 70, 71, 83, 89, 97

logistic, 13, 30, 39, 52, 55, 121, 123

probit, 12, 52

standard transformation for percentages, 6