

# Linear Models Part II

# Orthogonality and Polynomials

Two terms in a linear model are said to be *orthogonal* when all the cross-products between the one column from term one and one from term two are zero.

Orthogonality make least-square fits (*only*) easier to interpret. If all the terms are orthogonal to each other (and to the implied intercept, if there is one),

- the coefficients for each term do not depend on which other terms have been fitted,
- the sum-of-squares for each term does not depend on which other terms have been fitted (type I vs type III?), and
- the coefficients for each term are statistically independent.

The ability to piece together the fit from fitting a term at a time was crucial in the days of hand calculation.

This is why the coding matrix for ordered factors is based on orthogonal polynomials. If each level occurs the same number of times, the columns in the design matrix are orthogonal.

How do we get such polynomials? One can look them up in tables, and there is a recursive formula to calculate them. Or we can just put the polynomial terms into a design matrix and orthogonalize them. To do so regress each column in turn on the earlier columns, and replace it by its residuals (taking care to do the calculation accurately).

Function `poly` in **S-PLUS** calculates orthogonal polynomials of one or more numeric variables.

For example, `contr.poly(n)` generates orthogonal polynomials of order  $n$  in **S-PLUS**,

```
> contr.poly(4)
      .L      .Q      .C
[1,] -0.6708204  0.5 -0.2236068
[2,] -0.2236068 -0.5  0.6708204
[3,]  0.2236068 -0.5 -0.6708204
[4,]  0.6708204  0.5  0.2236068
```

or you could define your own contrast matrix of orthogonal polynomials

```
> contrasts(data) <- matrix(c(1,1,-1,-1,1,-1,-1,1,-.5,.5,-.5,.5), nr=4, nc=3)
> contrasts(data)
      [,1] [,2] [,3]
[1,]    1    1 -0.5
[2,]    1   -1  0.5
[3,]   -1   -1 -0.5
[4,]   -1    1  0.5
```

# Predictions

Given the values of the regressors for a new case, how do we predict the response? Easy: use the linear model with those regressors and the estimated coefficients  $\hat{\beta}$ . Note that the answer will not depend on the coding used.

How accurate are our predictions? We need to distinguish between

- predictions of  $E y$ . where the only uncertainty comes from that in  $\hat{\beta}$ , and
- predictions of  $y$ , where we need to account for the error distribution.

We normally talk about giving confidence intervals in the first case and *tolerance intervals* in the second.

Suppose want to predict gas consumption at average temperature 5°, both before and after insulation is added. We need to set up a data frame with the new data. R gives the 95% intervals by

```
> whiteside.lm <- lm(Gas ~ Insul/Temp - 1, data = whiteside)
> ## we might match order of the factor levels here but this is OK
> wsnew <- data.frame(Temp=5, Insul=factor(c("Before", "After")))
> predict(whiteside.lm, wsnew, interval="confidence")
      fit    lwr    upr
1 4.8876 4.7595 5.0157
2 3.3342 3.2133 3.4551
> predict(whiteside.lm, wsnew, interval="prediction")
      fit    lwr    upr
1 4.8876 4.2269 5.5483
2 3.3342 2.6748 3.9935
```

In **S-PLUS** one needs to work harder, as all one gets are the standard errors:

```
> (pr <- predict(whiteside.lm, wsnew, se=T))
$fit
4.8876 3.3342
$se.fit
0.063833 0.060242
$residual.scale:
[1] 0.323
$df:
[1] 52
> mc <- qt(0.975, df=pr$df)*pr$se.fit
> cbind(fit=pr$fit, lower=pr$fit - mc, upper=pr$fit + mc)
      fit lower upper
1 4.8876 4.7595 5.0157
2 3.3342 3.2133 3.4551
> mt <- qt(0.975, df=pr$df)*sqrt(pr$se.fit^2 + pr$residual.scale^2)
> cbind(fit=pr$fit, lower=pr$fit - mt, upper=pr$fit + mt)
      fit lower upper
1 4.8876 4.2269 5.5483
2 3.3342 2.6748 3.9935
```

Now suppose we want the difference insulation makes at 5°. Again, the estimate is easy:

```
> diff(pr$fit)
-1.5535
```

but a confidence interval is harder. The easiest way is to re-parametrize the problem so this becomes a parameter:

```
> options(contrasts=c("contr.treatment", "contr.poly"))
> summary(lm(Gas ~ Insul/I(Temp-5), data = whiteside), cor=F)
Coefficients:
                Value Std. Error t value Pr(>|t|)
(Intercept)    4.888    0.064    76.569  0.000
          Insul  -1.553    0.088   -17.699  0.000
InsulBeforeI(Temp - 5) -0.393    0.022   -17.487  0.000
  InsulAfterI(Temp - 5) -0.278    0.023   -12.124  0.000
> c(-1.553 - qt(0.975, df=52)*0.088, -1.553 + qt(0.975, df=52)*0.088)
[1] -1.7296 -1.3764
```

# Box–Cox Transformations

Suppose the normal linear model really applied to  $z = g(y)$ , for example  $\log(y)$ ,  $\sqrt{y}$  or  $1/y$ . Then

$$p(Y; \beta, \sigma^2, g()) \propto \prod \frac{1}{\sigma} \exp -\frac{1}{2\sigma^2}(z_i - x_i\beta)^2 \cdot g'(y_i)$$

and so the log-likelihood is

$$L(\beta, \sigma^2, g(); Y) = \text{const} - \frac{n}{2} \log \sigma^2 - \frac{1}{2\sigma^2} \|Z - X\beta\|^2 + \sum_i \log g'(y_i)$$

Maximizing over  $\beta$  gives  $\hat{\beta}$  as the least-squares estimate for  $Z = g(Y)$ , and  $\hat{\sigma}^2 = RSS(g)/n$ .

Now consider the Box–Cox family of transformations  $g_\lambda(y) = (y^\lambda - 1)/\lambda$ .

$$\begin{aligned} L(\hat{\beta}, \hat{\sigma}^2, \lambda; Y) &= \text{const} - \frac{n}{2} \log \frac{RSS(\lambda)}{n} + \sum_i \log y_i^{\lambda-1} \\ &= \text{const} - \frac{n}{2} \log RSS(\lambda) + n(\lambda - 1) \log \dot{y} \\ &= \text{const} - \frac{n}{2} \log \left[ \frac{RSS(\lambda)}{(\dot{y}^{\lambda-1})^2} \right] \end{aligned}$$

where  $\dot{y} = \sqrt[n]{\prod y_i}$  is the geometric mean of the data. Thus the MLE of  $\lambda$  is chosen to minimize the RSS for the data on the scale

$$z(y) = \frac{y^\lambda - 1}{\lambda \dot{y}^{\lambda-1}}$$

In practice we find a confidence interval for  $\hat{\lambda}$  from asymptotic theory for the profile likelihood

$$2L(\hat{\beta}, \hat{\sigma}^2, \hat{\lambda}; Y) - 2L(\hat{\beta}, \hat{\sigma}^2, \lambda; Y) \sim \chi_1^2$$

to choose an interpretable value of  $\lambda$ .

Note that transforming the response  $y$  can have up to three benefits

1. To make the error variances more nearly equal,
2. To make the distribution of the errors closer to normal,
3. To give a simpler explanation (e.g. no interactions),

but possibly not all for the same transformation.

## Example

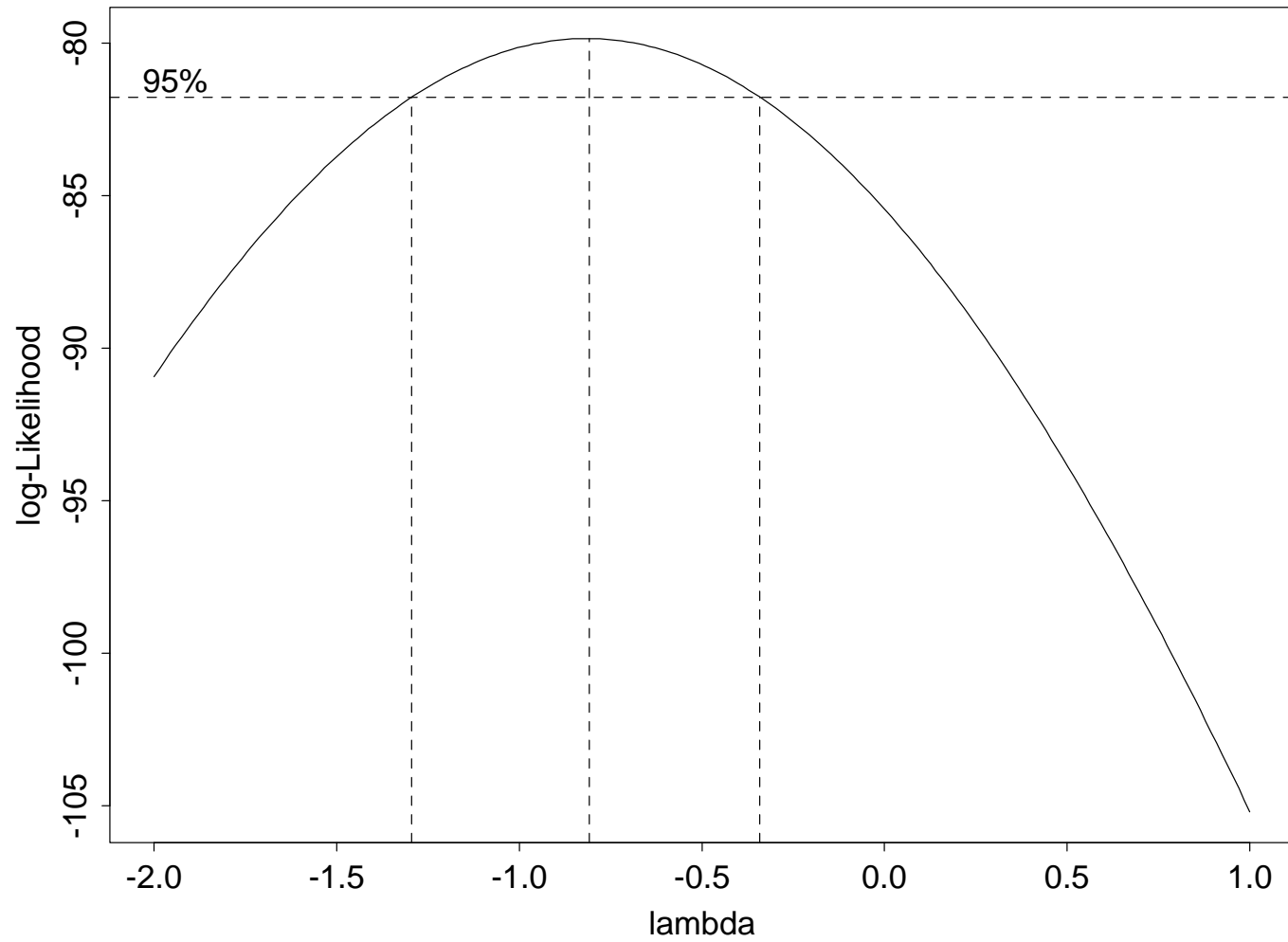
The ANOVA table for the Box–Cox poisons data is

	df	SS	MS	F	SS( $y^{-1}$ )
poisons	2	103.30	51.65		0.3488
treatments	3	92.24	30.75		0.2041
interaction	6	25.01	4.17	1.88	0.0157
residual	36	80.07	2.22		0.0864
‘total’	47	300.62			0.6550

Fitting the model to the survival times shows some evidence for an interaction ( $P \approx 11\%$ ). For survival rates the  $F$  ratio is 1.09.

The log-likelihood function for a Box–Cox transformation is shown in the figure.

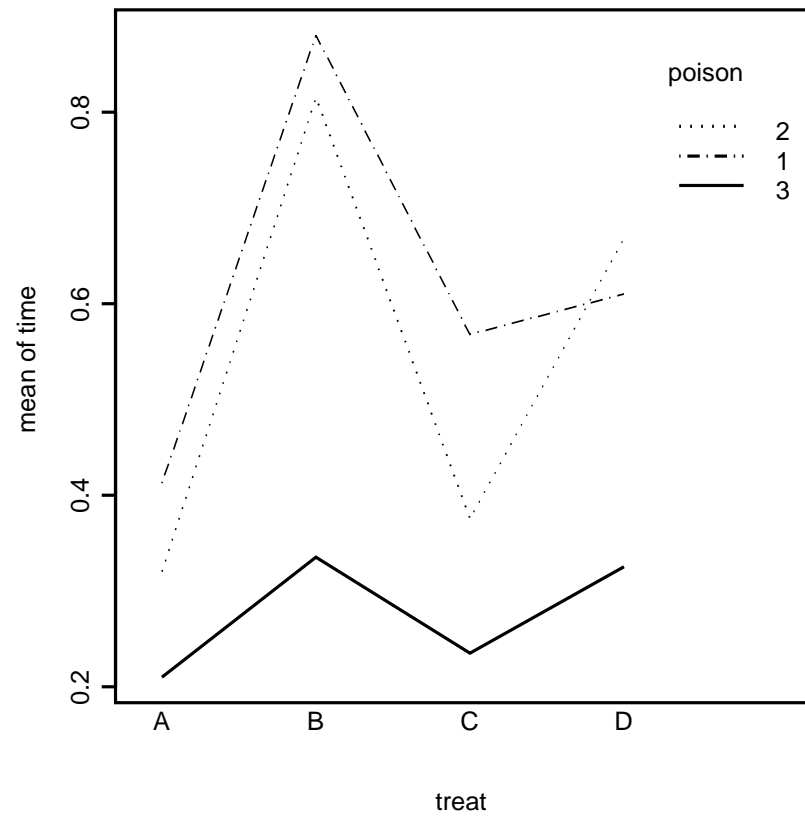
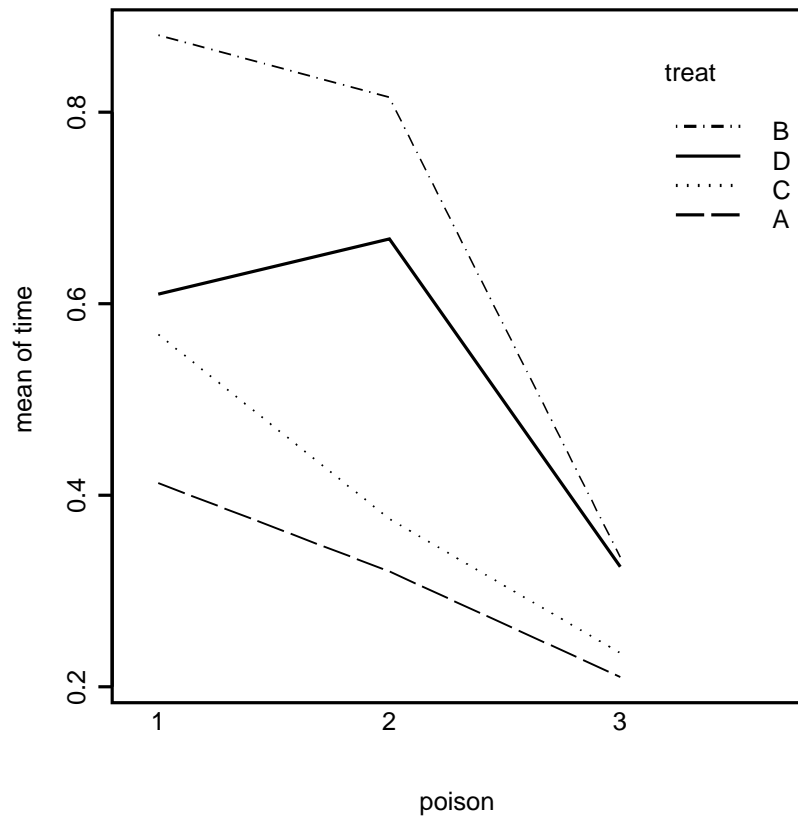
For ease of interpretation we take  $\lambda = -1$ , that is analyze the survival *rates*. On that scale the additive model fits well.



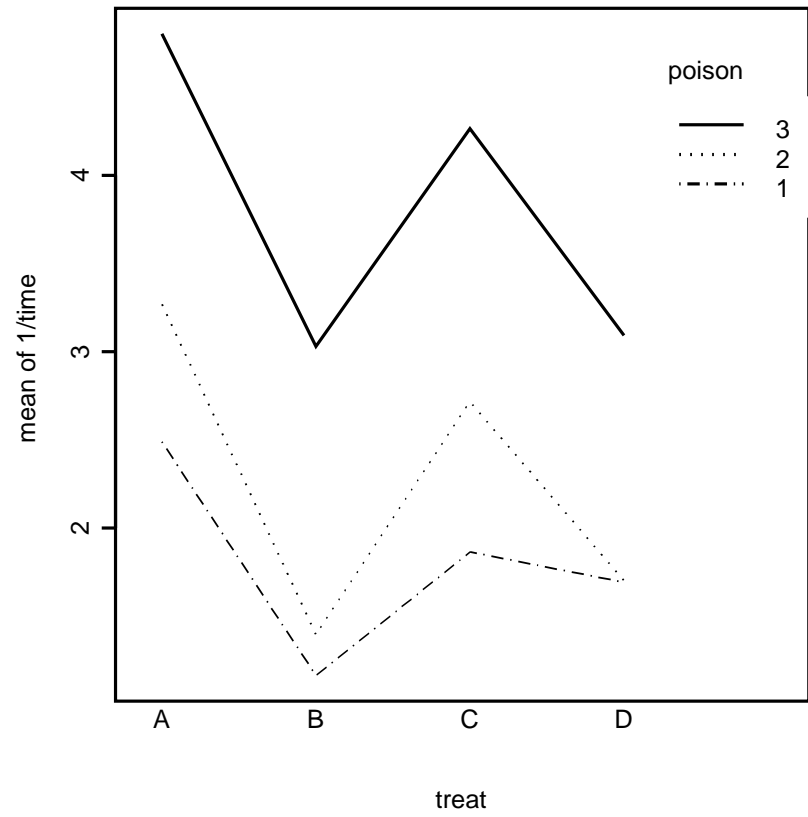
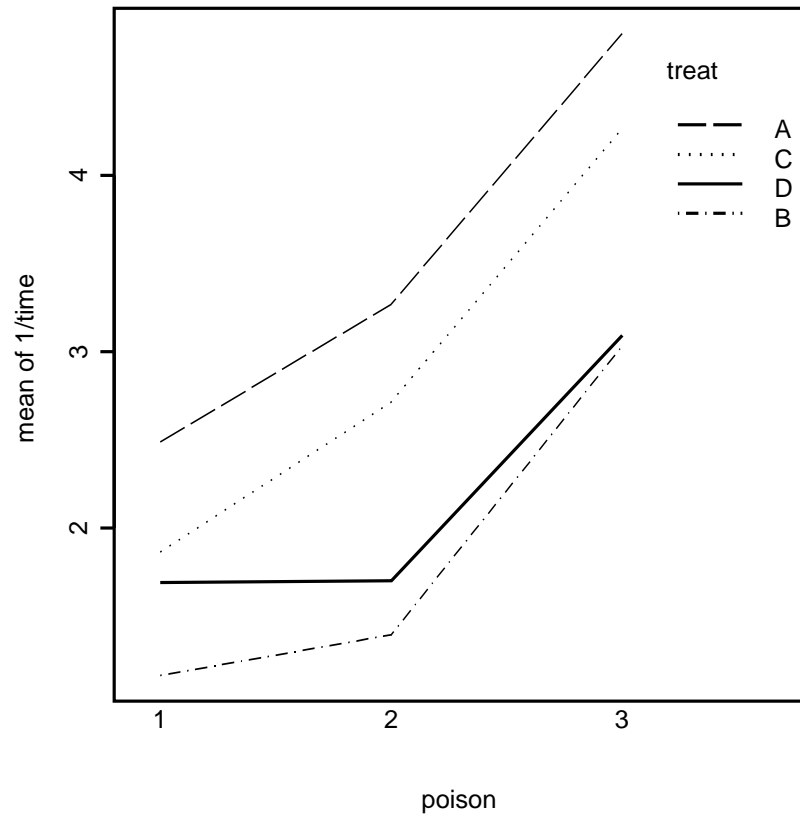
The log-likelihood for the Box-Cox transformation for the poisons data.

# Interaction Plots

For a two-way layout we can plot the cells means using function `interaction.plot` as in



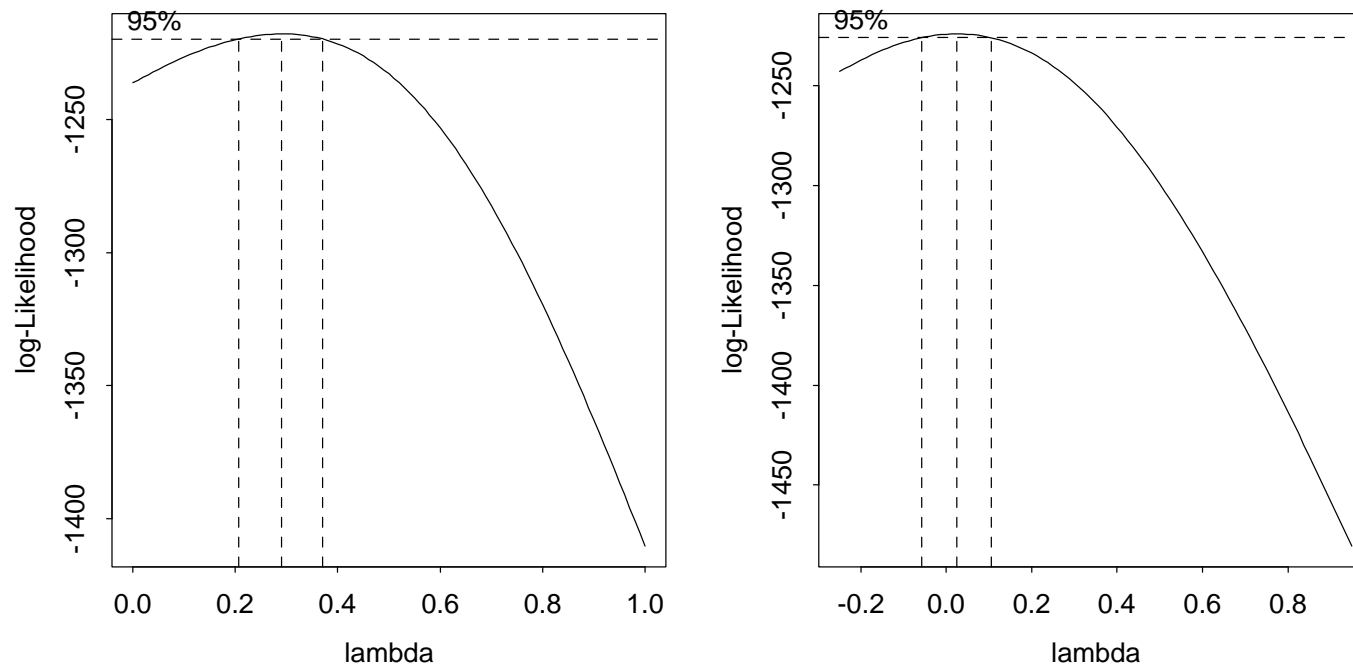
The lines are more nearly parallel on reciprocal scale:



# Transforming Both Sides

If there are numeric regressors we may need to transform both regressors and responses simultaneously. That can be done, but a simple and often effective idea is to discretize the numeric regressors.

Consider the CPUs performance dataset you saw in *Case Studies*. For the right plot we split the numeric variables at their quartiles.



# Regression Diagnostics

Having fitted our model, we want to check whether the fit is reasonable. We do this by looking at various types of residuals. The (ordinary) residuals ( $e_i$ ) fail to be a fair measure of the errors for two reasons:

1. The variance of  $e_i$  varies over the space of regressors, being greatest at  $(\bar{x}_1, \dots, \bar{x}_p)$ .

$$e = Y - Xb = (I - X(X^T X)^{-1} X^T)Y = (I - H)\epsilon$$

The matrix  $H$  is often known as the *hat* matrix as  $HY = \hat{Y}$ . We have

$$\text{var}(e_i) = \sigma^2(1 - h_{ii})$$

The **standardized residual** is formed by normalizing to unit variance then replacing  $\sigma^2$  by  $s^2$ , so

$$e'_i = \frac{e_i}{s\sqrt{1 - h_{ii}}}$$

Another way of looking at this reduced variance is to say that data point  $i$  has high **leverage** and is able to pull the fitted surface toward the point: the average leverage is  $p/n$ . We will generally take note of points with leverages more than two or three times this average.

2. If one residual is very large, the variance estimate  $s^2$  will be too large, and this deflates the standardized residuals. Suppose we fit the model without observation  $i$ . We get a prediction  $\hat{y}_{(i)}$  of  $y_i$ . The **studentized (or deletion or jackknife) residual** is

$$e_i^* = \frac{y_i - \hat{y}_{(i)}}{\sqrt{\text{var}(y_i - \hat{y}_{(i)})}}$$

where  $\sigma$  (used in the variance estimate) is replaced by its estimate in this fit,  $s_{(i)}$ .

Fortunately, it is not necessary to re-fit the model each time deleting an observation. Let  $e_{(i)} = y_i - \hat{y}_{(i)}$ .

$$e_i^* = \frac{e_{(i)}}{s_{(i)}/\sqrt{(1-h_{ii})}} = \frac{e_i}{s_{(i)}\sqrt{(1-h_{ii})}} = \frac{se'_i}{s_{(i)}} = \frac{e'_i}{\sqrt{\frac{n-p-e_i'^2}{n-p-1}}}$$

This shows explicitly that where the standardized residual  $e'_i$  is larger than one (in modulus), the studentized residual  $e_i^*$  is larger. In fact, the maximum value of the standardized residual is  $\sqrt{n-p}$ .

# Cook's statistic

The studentized residuals tell us whether a point has been explained well by the model, but if it has not, they do not tell us what the size of the effect on the fitted coefficients of omitting the point might be. A badly-fitted point in the middle of the design space will have much less effect on the predictions than one at the edge of the design space.

Cook (1977) proposed a measure that combines both the effect of leverage and that of being badly fitted. His statistic is

$$D_i = \frac{(b_{(i)} - b)^T X^T X (b_{(i)} - b)}{ps^2} = \frac{\|\hat{Y}_{(i)} - \hat{Y}\|^2}{ps^2} = \frac{(e'_i)^2 h_{ii}}{p(1 - h_{ii})}$$

Several small modifications have been proposed. One (Atkinson, 1985, p. 25) is to use the signed square root, taking the same sign as that of the residuals, and to drop the  $p$ . If in this we replace  $s$  by  $s_{(i)}$  we get

$$\text{DFITS}_i = \sqrt{\frac{h_{ii}}{(1 - h_{ii})}} e_i^*$$

As Sen & Srivastava (1990, p. 161) point out

‘Actually, the number of measures available in the literature for identifying outliers and influential points verges on being mind-boggling.’

So use what tools your computer package makes available, and remember that deleting one or more points and re-fitting is much less onerous than when most of these measures were designed (in the days of punch cards and batch computing).

# How To Use Diagnostic Plots

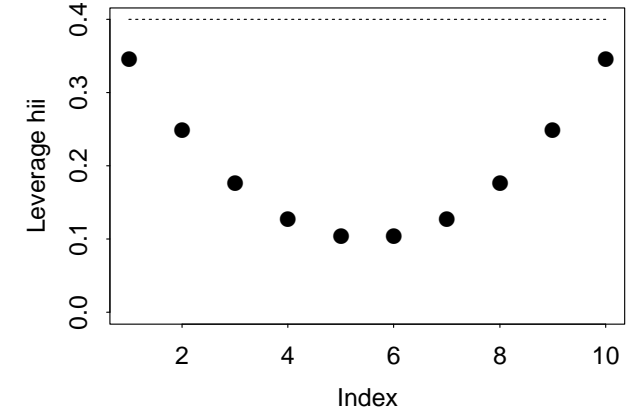
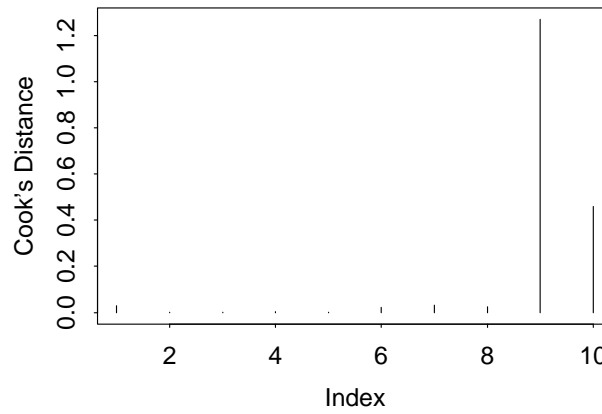
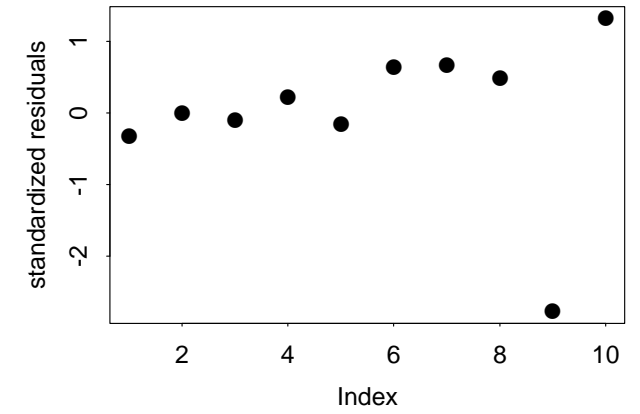
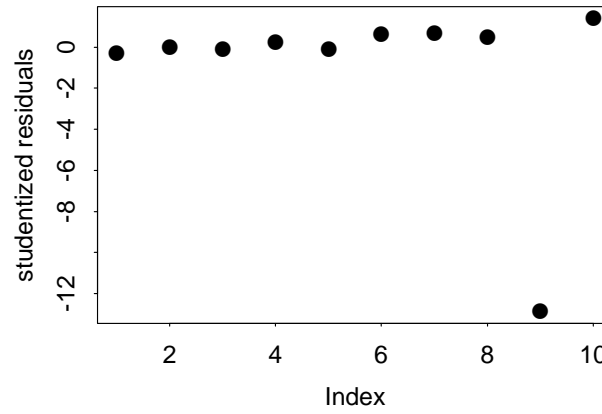
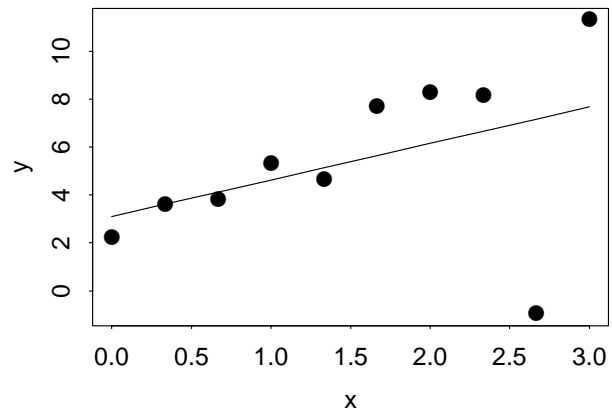
Various plots of the different types of residuals, leverage and Cook's distance are useful:

- plot residuals against the index of the dataset. This will show up observations with large residuals: possible outliers. It can also show effects from the time ordering of the measurements.
- plot residuals against  $x$ , if one-dimensional, any single column of  $X$ , or any possible extra regressor. This can show up patterns in the residuals which indicate non-linearity: for example, that the relationship is with  $x^2$  rather than with  $x$ . It can also demonstrate that a potential extra regressor will be useful.
- plot residuals against the fitted values of the  $y$ . This can show up heteroscedasticity, where the variance is not constant over the whole range. This plot is done against  $\hat{y}$  rather than  $y$  as the residuals are correlated with  $y$  but not with  $\hat{y}$ .

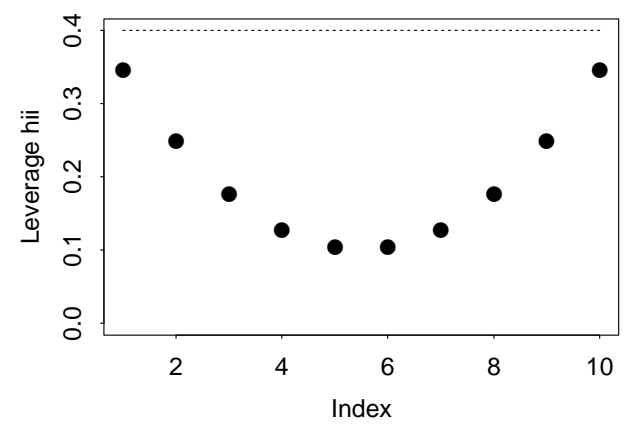
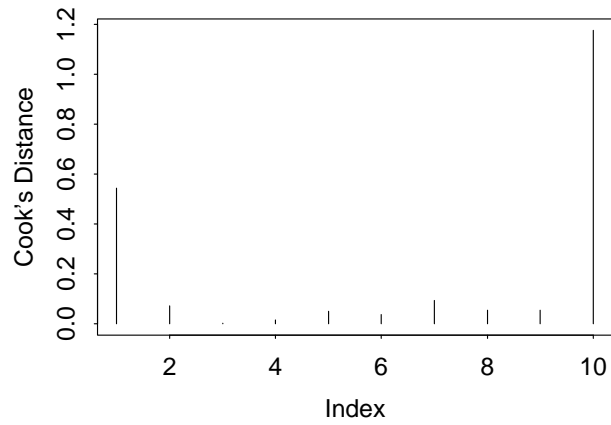
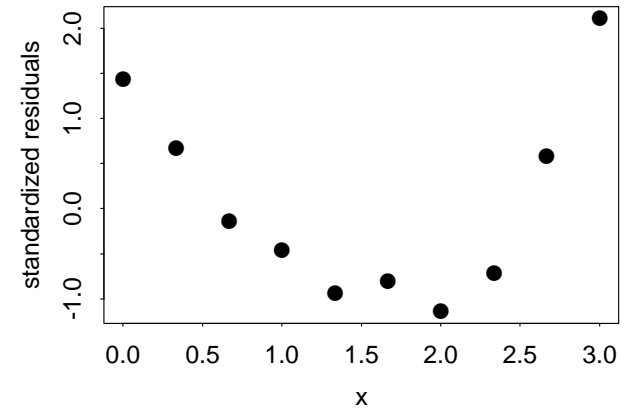
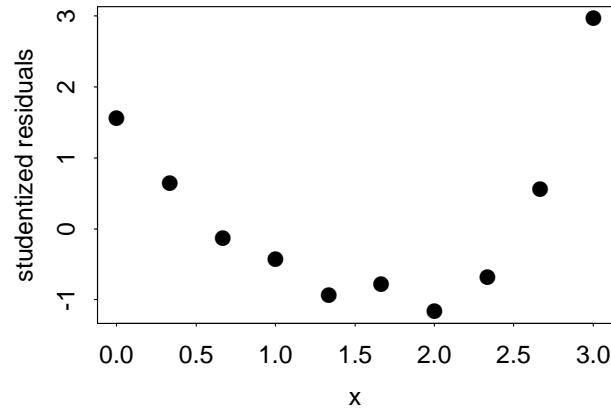
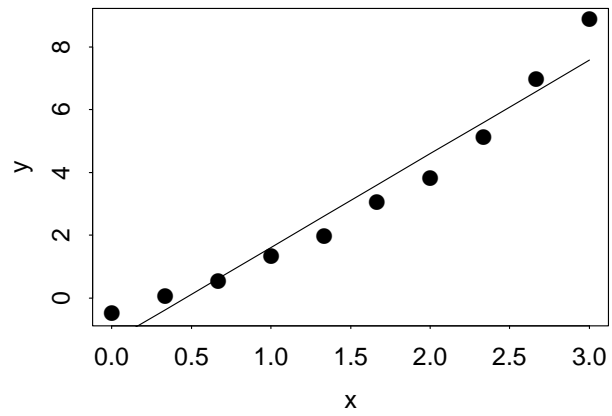
- Leverage plots against index will show which points *may* have large influence: such points may or may not be outliers and plotting the Cook's statistic will draw attention to points which seem to be influential.

**Caution:** if there is more than one outlier these methods may fail to show any of them, as we only consider the effect of omitting one point at a time.

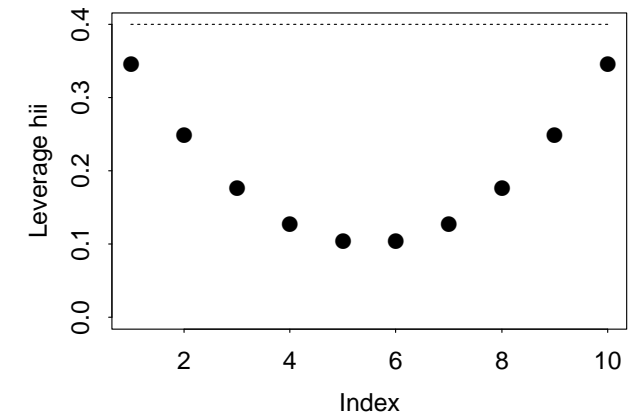
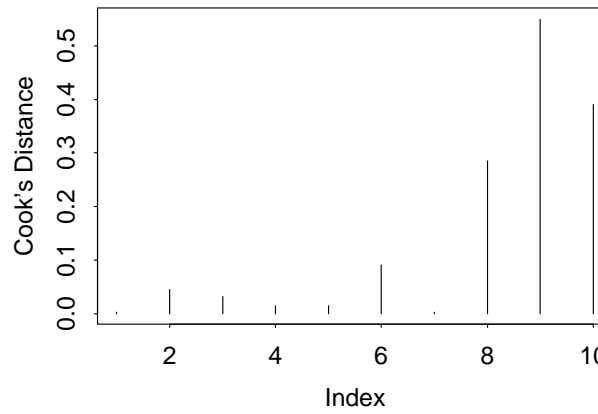
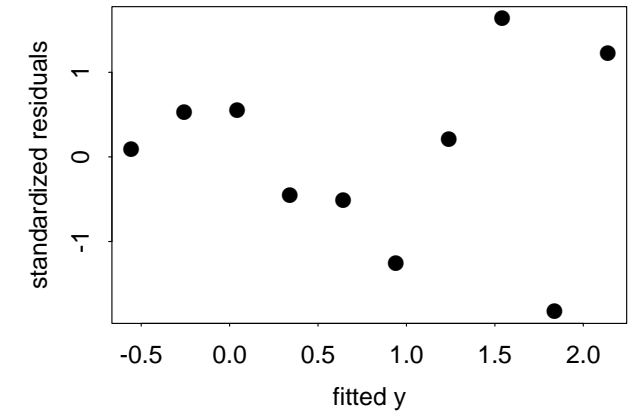
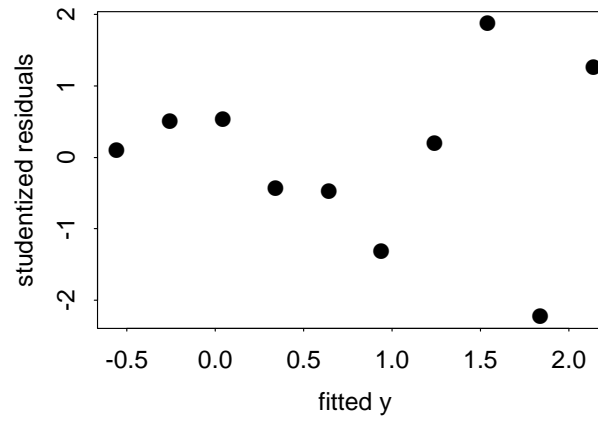
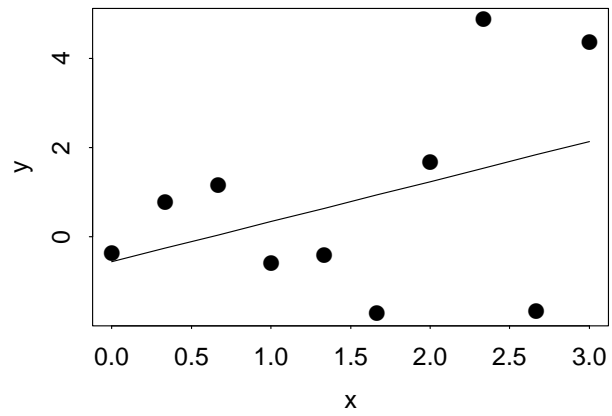
The next few figures show some simple (synthetic) examples of the effect of deviations from the assumed linear model on the diagnostic plots. The top middle plot is of studentized residuals, the top right of standardized residuals. The bottom two plots are of Cook's statistic and leverage against index. The dotted line on the leverage plot is at  $2p/n$ .



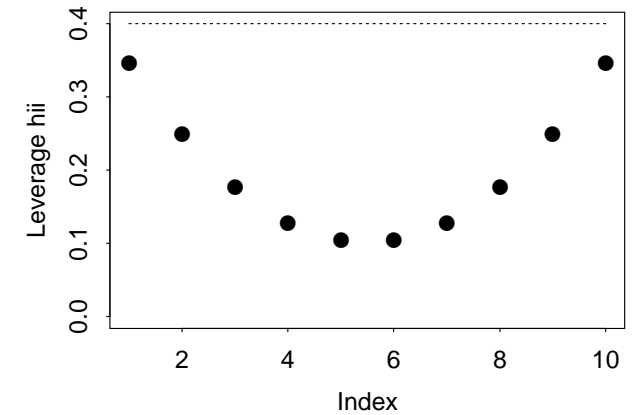
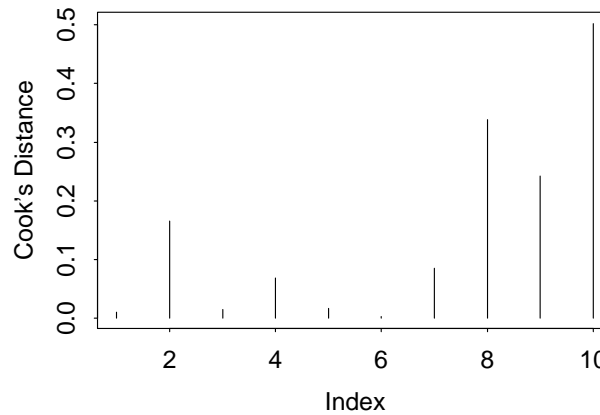
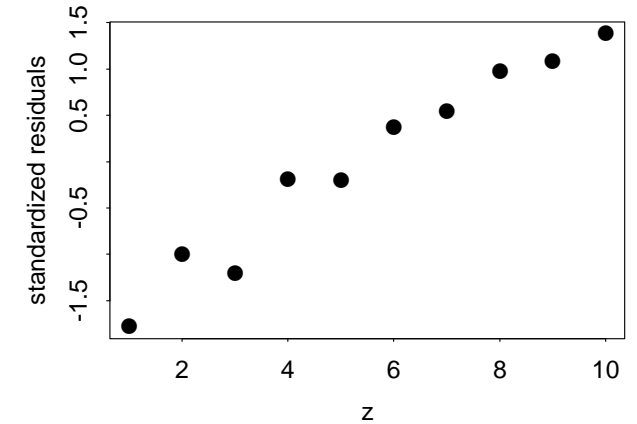
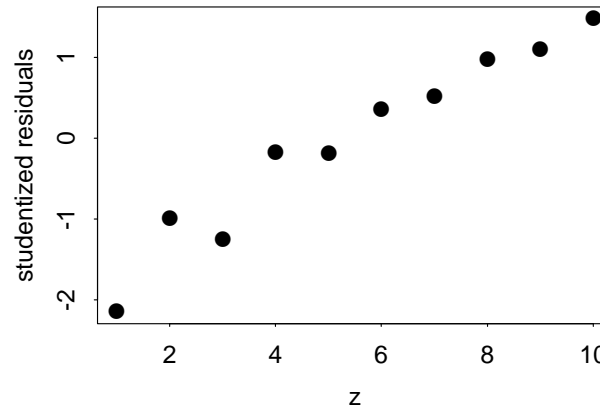
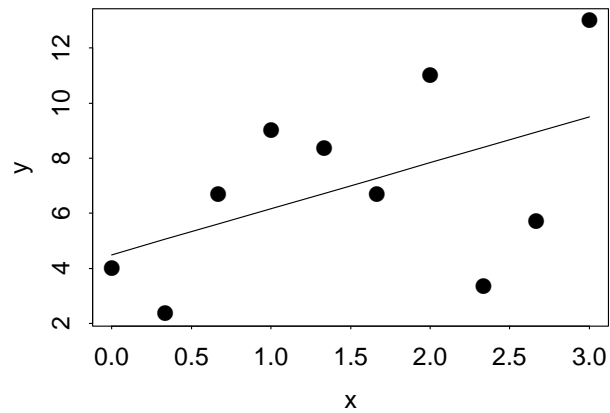
One large residual. Residuals are plotted against index.



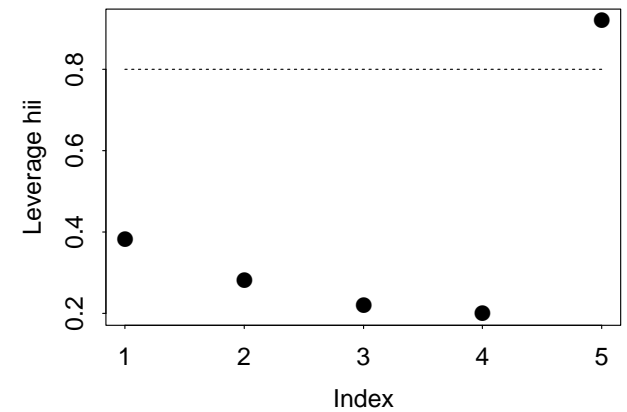
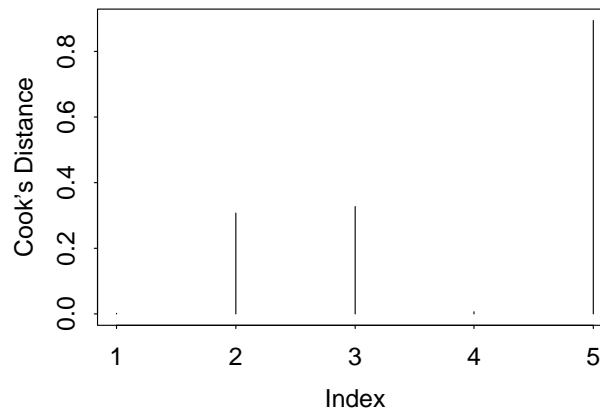
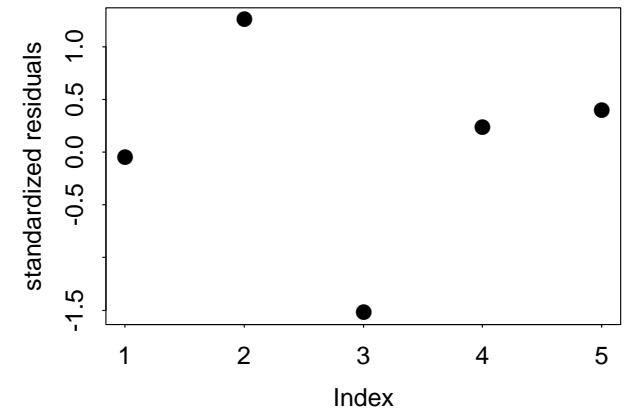
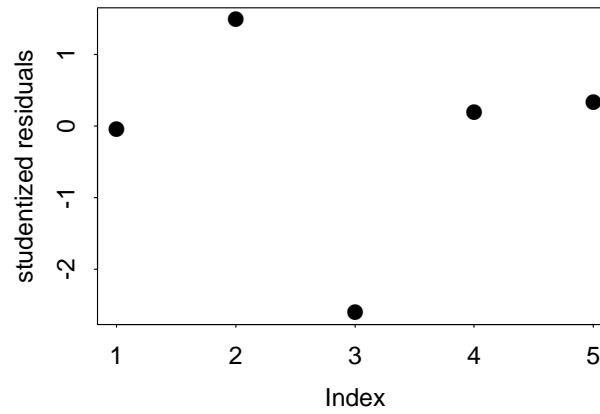
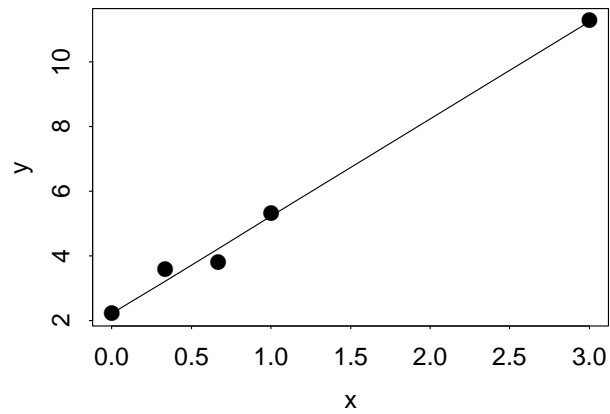
Pattern in the residuals. Residuals plotted against  $x$ .



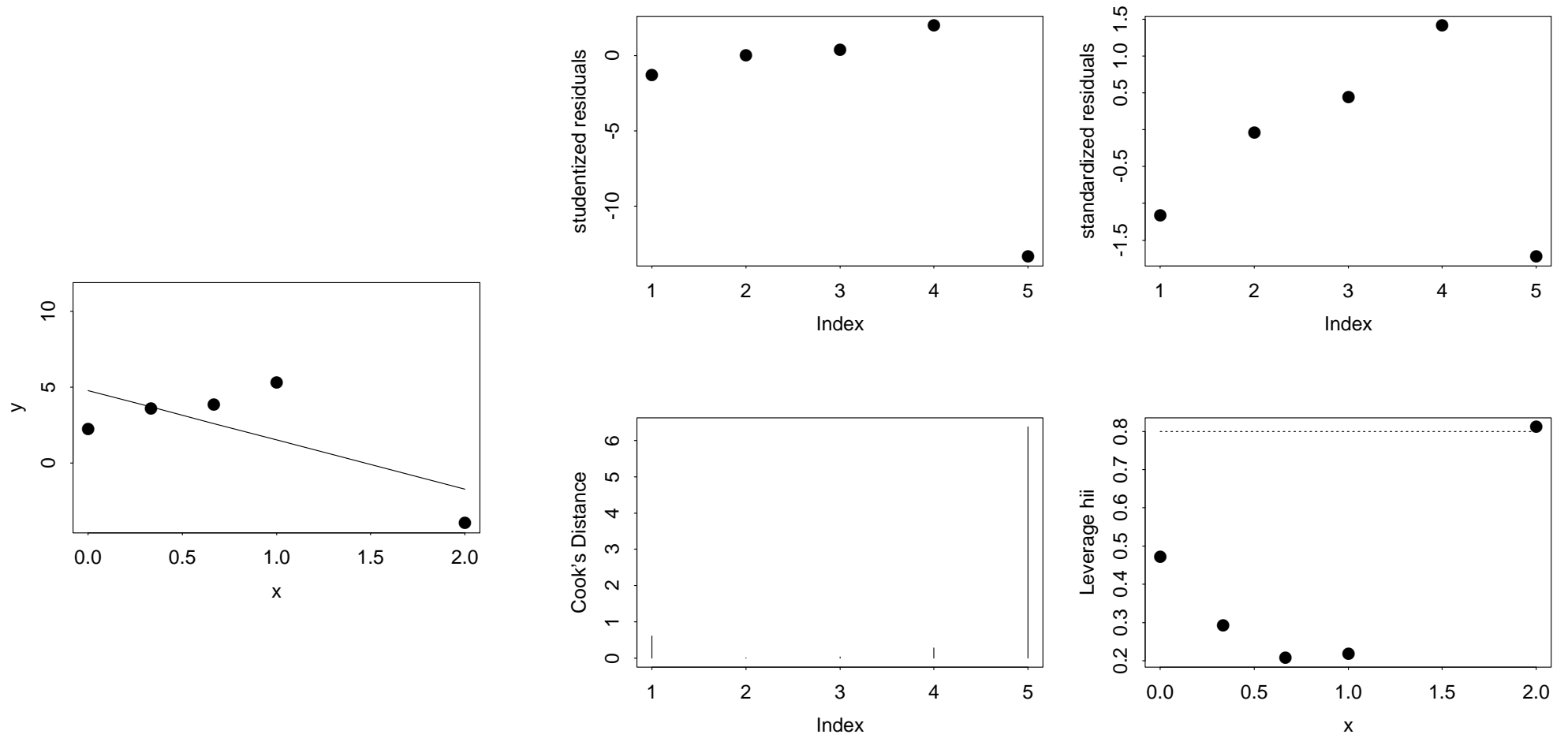
Heteroscedasticity. Residuals plotted against fitted values.



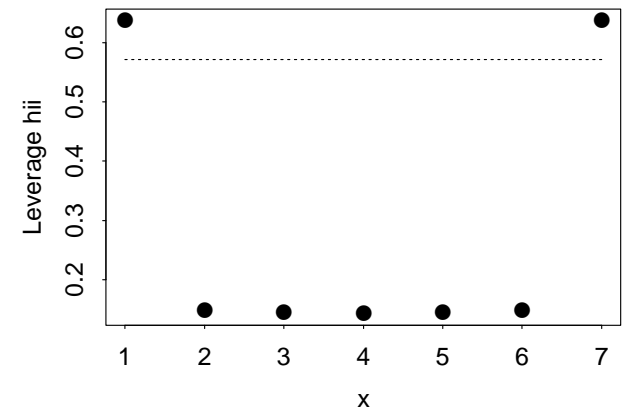
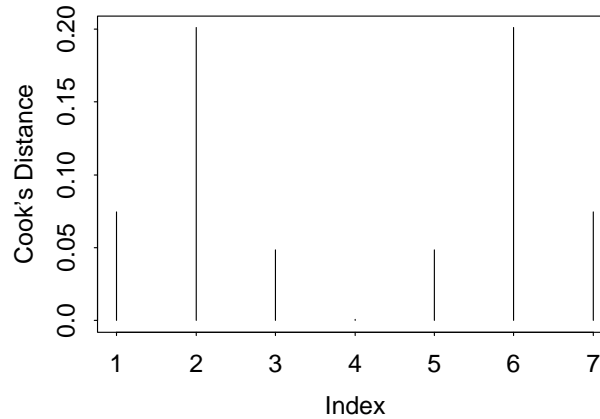
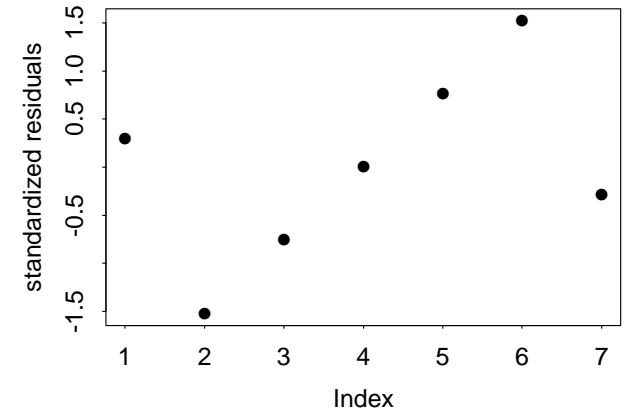
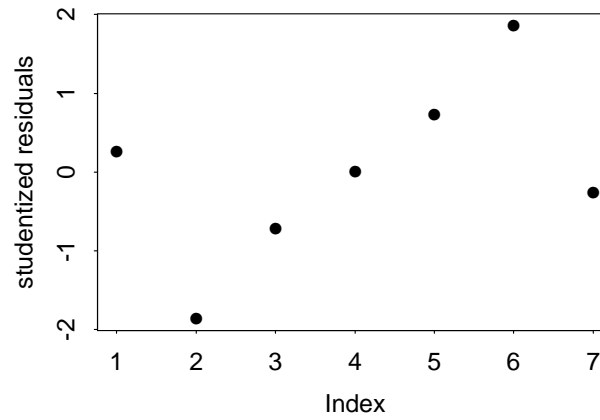
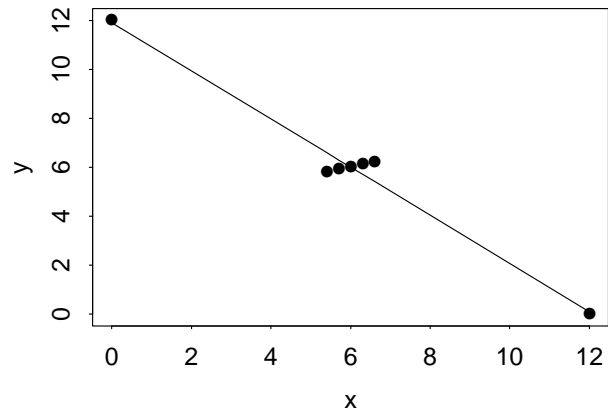
Missing regressor  $z$ . Residuals plotted against  $z$ . (It would be even better to plot against the residuals from regressing  $z$  on  $x$ , a so-called *added variable* plot.)



Non-outlier has large influence. Residuals plotted against index. The effect of the non-outlier to make predictions much more precise.



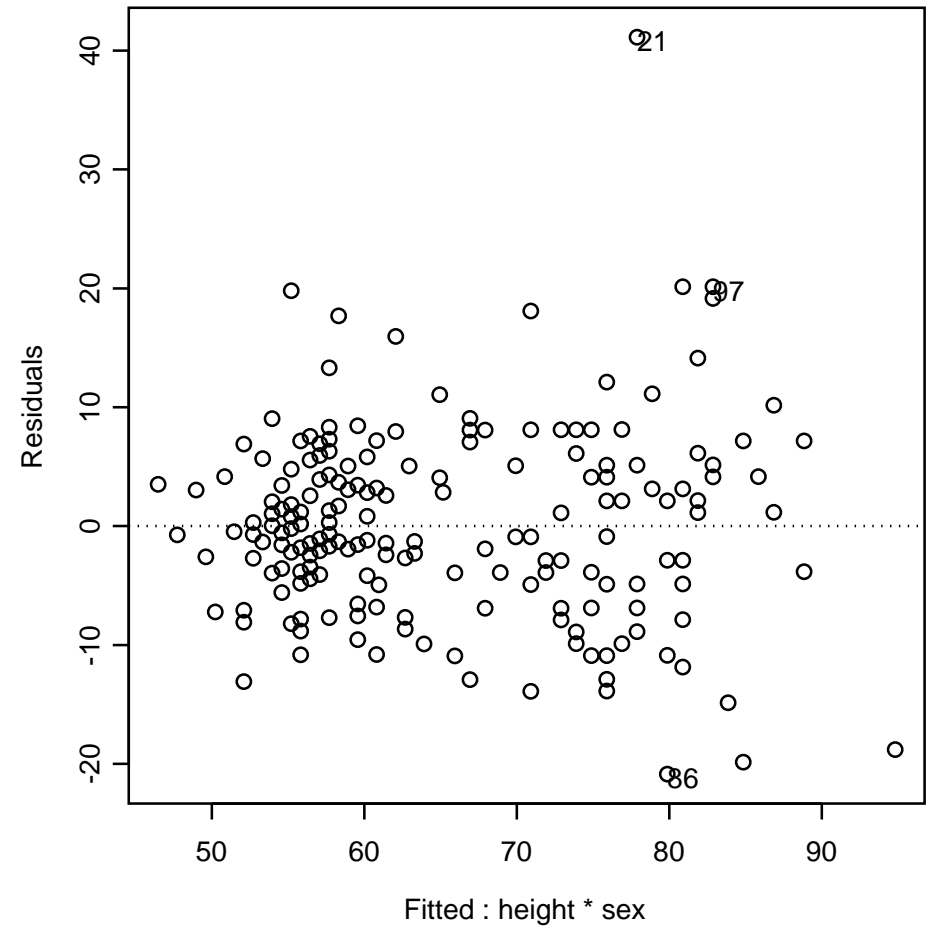
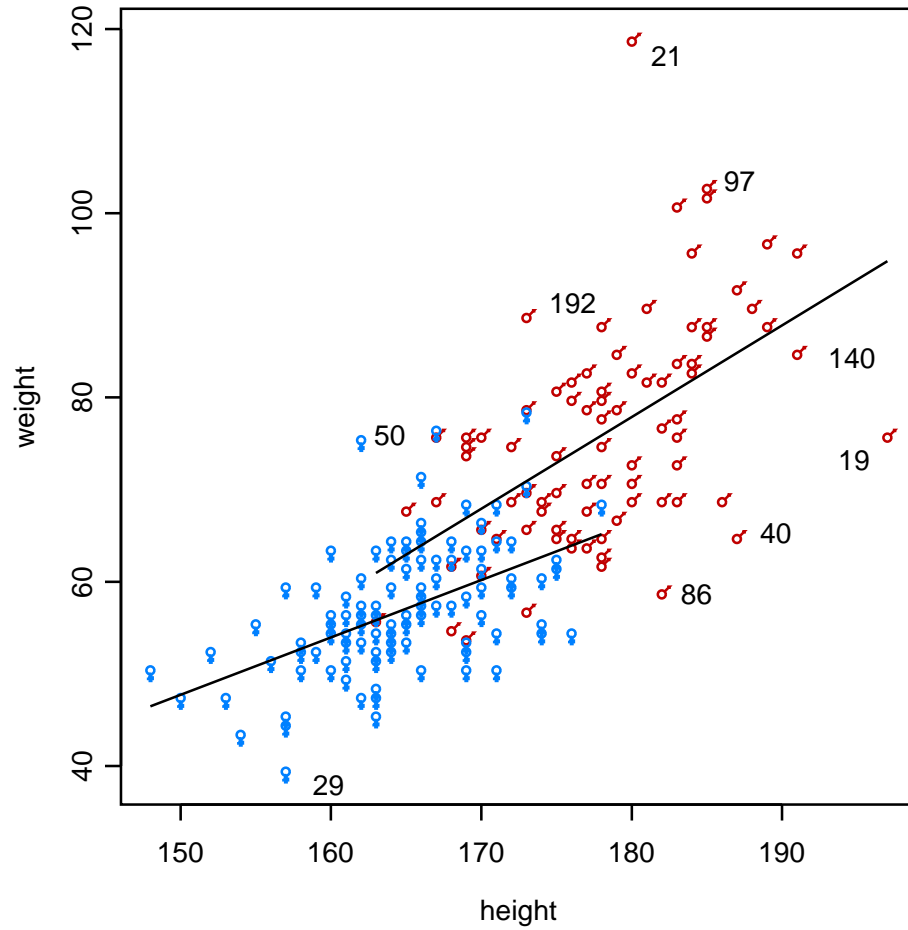
One outlier has large influence. Residuals plotted against index.

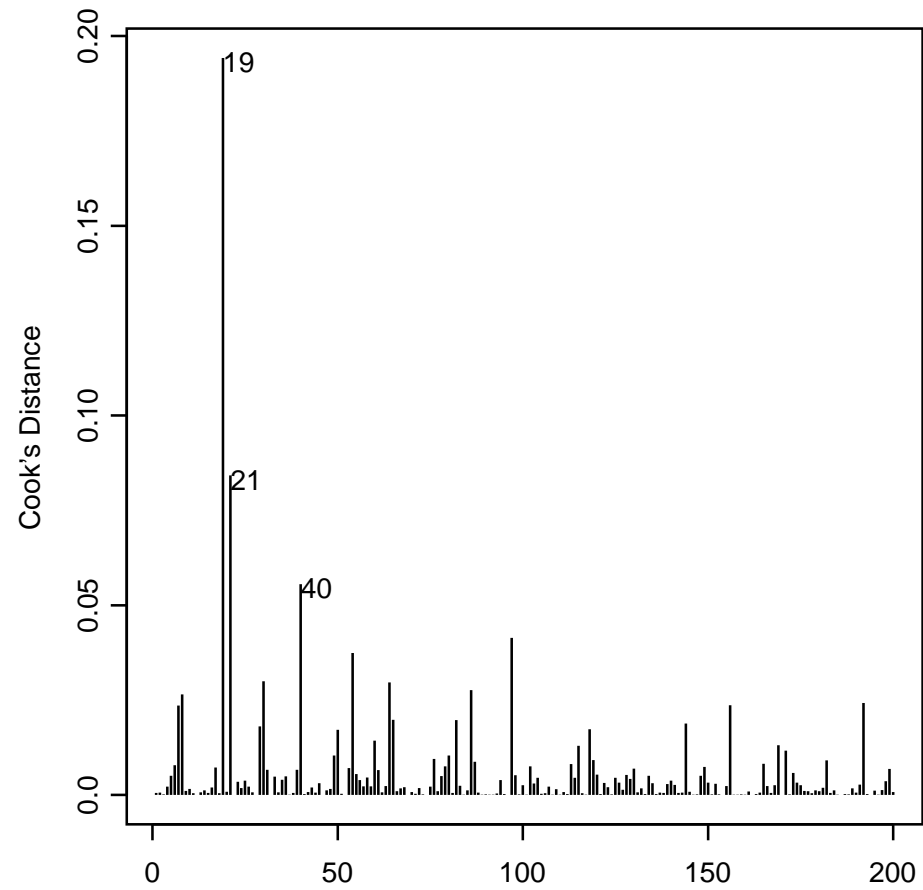
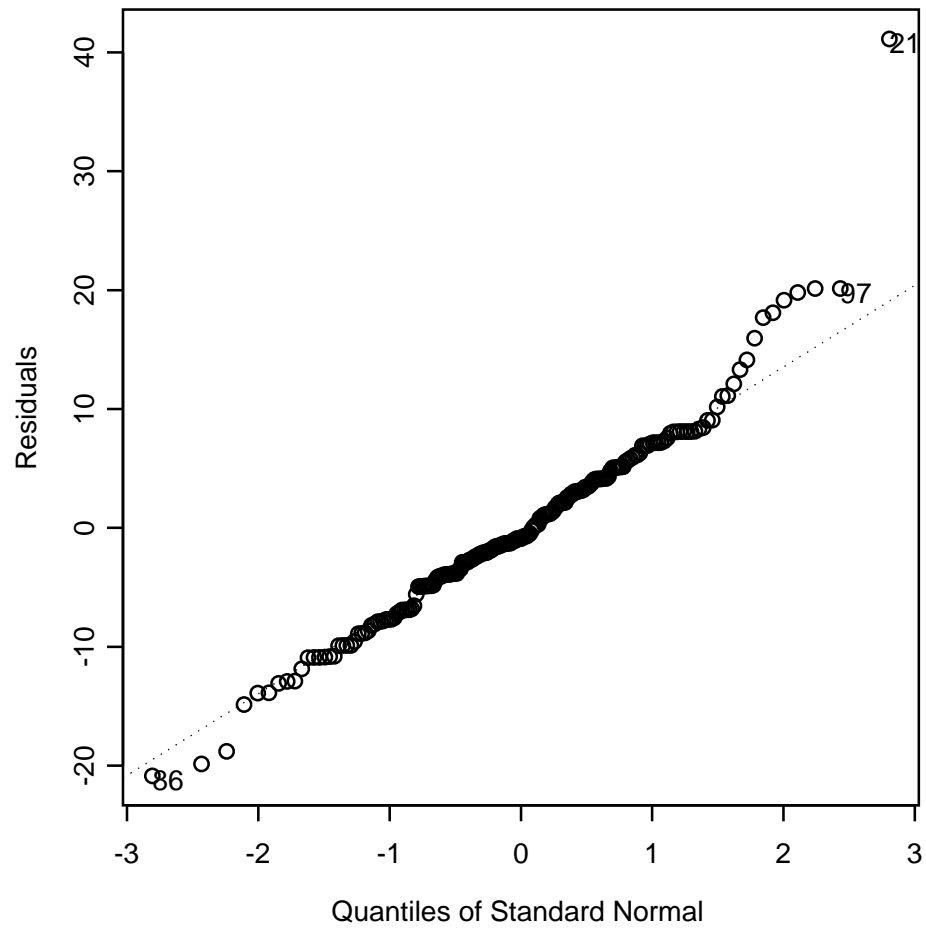


Two outliers have jointly large influence. Residuals plotted against index. Cook's statistic does not give enough of the story here.

# Davis' Height–Weight Data

from the practical class in weeks 2 and 3.





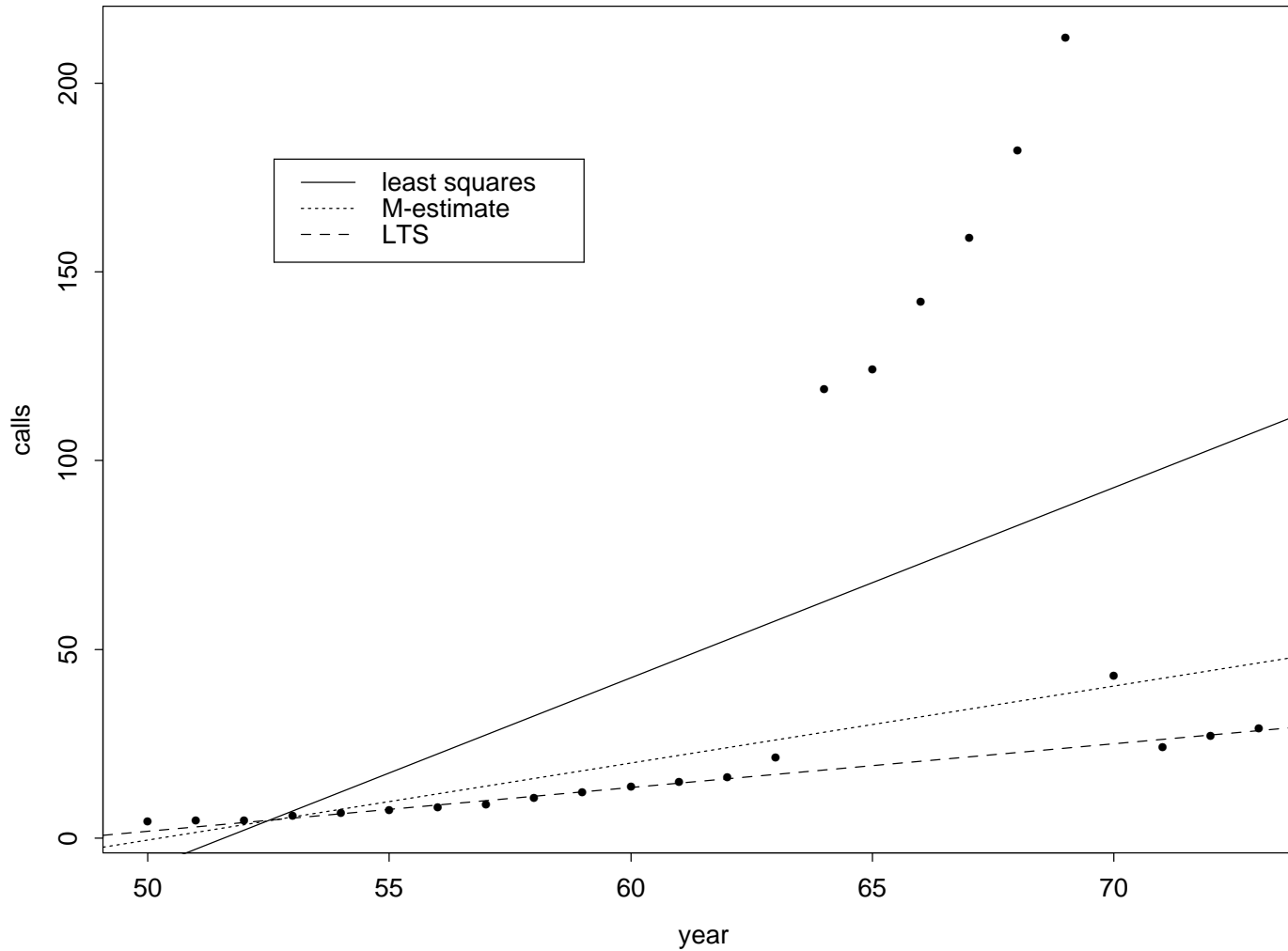
# Resistant and Robust Regression

Resistant regression has a high breakdown point, approaching 50%. We consider replacing least-squares by one of

**LMS** Least median of squares: minimize the median of the squared residuals. More generally, for LQS, minimize some quantile (say 80%) of the squared residuals.

**LTS** Least trimmed squares: minimize the sum of squares for the smallest  $q$  of the residuals. Originally  $q$  included just over 50%, but **S-PLUS** has switched to 90%.

However, either involves very much more computing than least squares. Both do show up the effect of multiple outliers, as they concentrate on fitting just over 50% of the data well. In doing so they are less efficient when there are no outliers (LMS more so than LTS).



Millions of phone calls in Belgium, 1950–73, from Rousseeuw & Leroy (1987), with three fitted lines.

# M-estimators

Solve the non-linear equations

$$\sum_{i=1}^n \mathbf{x}_i \psi \left( \frac{y_i - \mathbf{x}_i \boldsymbol{\beta}}{s} \right) = 0$$

for a scale estimate  $s$ . A common way to solve these is by iterated re-weighted least squares, with weights

$$w_i = \psi \left( \frac{y_i - \mathbf{x}_i b}{s} \right) / \left( \frac{y_i - \mathbf{x}_i b}{s} \right)$$

The iteration is only guaranteed to converge for *convex*  $\rho$  functions, and for redescending functions the equation may have multiple roots. In such cases it is usual to choose a good starting point and iterate carefully.

Of course, in practice the scale  $s$  is not known. A simple and very resistant scale estimator is the MAD about some centre. This is applied to the residuals about zero, either to the current residuals within the loop or to the residuals from a very resistant fit.

# Model Selection

How do we choose a subset of  $q$  of the  $p$  regressors in such a way that the fit is nearly as good as with all  $p$ ? There is no universal answer.

- (a) **best subsets** is in principle simple: compute all subsets of size  $q$  and choose that with the smallest RSS. The problem is that there are  $\binom{p}{q}$  such subsets, and if  $p$  is large this is daunting. There are some iterative procedures which reduce the computation somewhat, but still this is only practicable with  $p \leq 50$  or so, depending on  $q$ .
- (b) **forward selection** works by fitting one extra regressor at a time, and choosing that with the largest reduction in RSS, equivalently with the largest  $F$ -ratio. We stop if the  $F$ -ratio is not significant enough. We use a different denominator at each test, which is not very desirable.
- (c) **backward elimination** first fits all  $p$  regressors, and deletes the one with the smallest  $t$ -ratio (in absolute value). This is repeated until all remaining variables have significant enough  $t$ -ratios.

(d) **stepwise selection** alternates between adding and deleting. First the regressor with the largest  $F$ -ratio to add is considered, and entered if significant enough. Then the regressor with the smallest  $t$ -ratio is dropped unless significant enough. This is repeated until no progress is possible.

There is no guarantee that any of the last three procedures will even find the best subset of the size on which they settle, but alarming examples are rare.

## Mallows' $C_p$ and Akaike's AIC

To search for models it would be appealing to have a criterion for goodness of fit that took account of model complexity. Clearly models with larger  $p$  should fit better.

Many criteria of the form

$$\text{RSS}/\sigma^2 + \alpha p \quad (\text{with mean } n + (\alpha - 1)p)$$

have been proposed. The most famous of these is Mallows'  $C_p$ , with  $\alpha = 2$ , and (according to some authors including Mallows, 1973) subtracting  $n$ . Of course  $\sigma^2$  is normally known, but as for  $F$ -tests, it is estimated from the largest model under consideration in the whole set of models.

Akaike's **An Information Criterion** applies to maximum-likelihood fitting of sets of models, and minimizes

$$\text{AIC} = -2 \text{ maximized log likelihood} + 2 \# \text{ parameters}$$

Since the log-likelihood is defined only up to a constant depending on the data, this is also true of AIC.

For a regression model with  $n$  observations,  $p$  parameters and normally-distributed errors and unknown  $\sigma^2$  we have

$$\text{AIC} = \frac{\text{RSS}}{\sigma^2} + 2p$$

(which is Mallows'  $C_p$ ) whereas for unknown  $\sigma^2$

$$\text{AIC} = n \log(\text{RSS}/n) + 2p$$

We can search stepwise for model(s) with small values of these criteria.

## Added Variable Plots

Suppose we are contemplating adding a column  $z$  to the design matrix (possibly for many different  $z$ 's). We can avoid re-fitting the model by finding the residuals  $e$  and the residuals  $e_z$  from the regression of  $z$  on  $X$  (which can be done rapidly for many  $z$ 's).

Then the coefficient of  $z$  when added to the regression is the regression coefficient of  $e$  on  $e_z$ , and the reduction in sum of squares in the main regression is the regression SSq in this regression (by orthogonality).

That's cute and can save time. What is really useful is that by plotting the simple linear regression on its scatterplot we can see if a small number of points dominate the need for that extra term. Such a plot is called an *added variable plot*.

## Example—The Effects of Pollution on Mortality

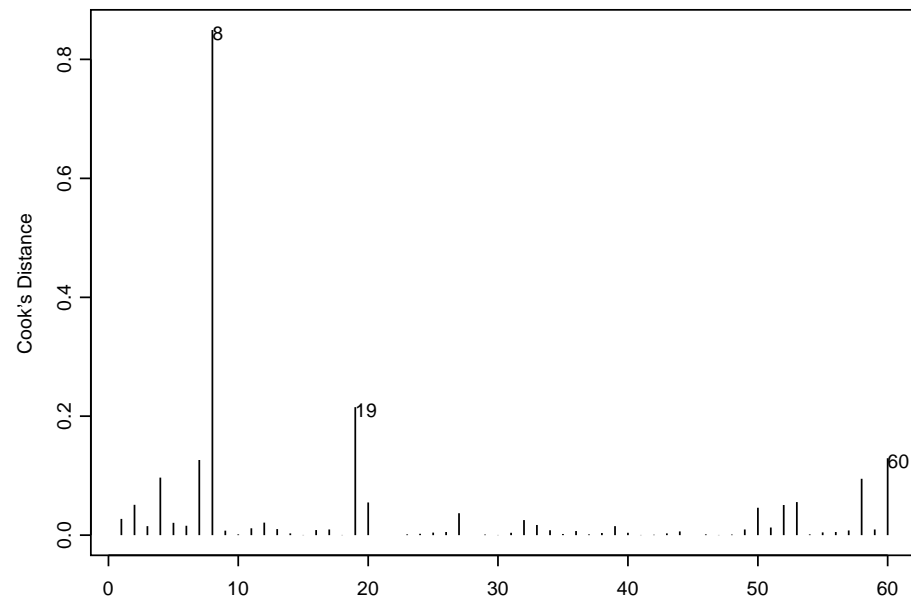
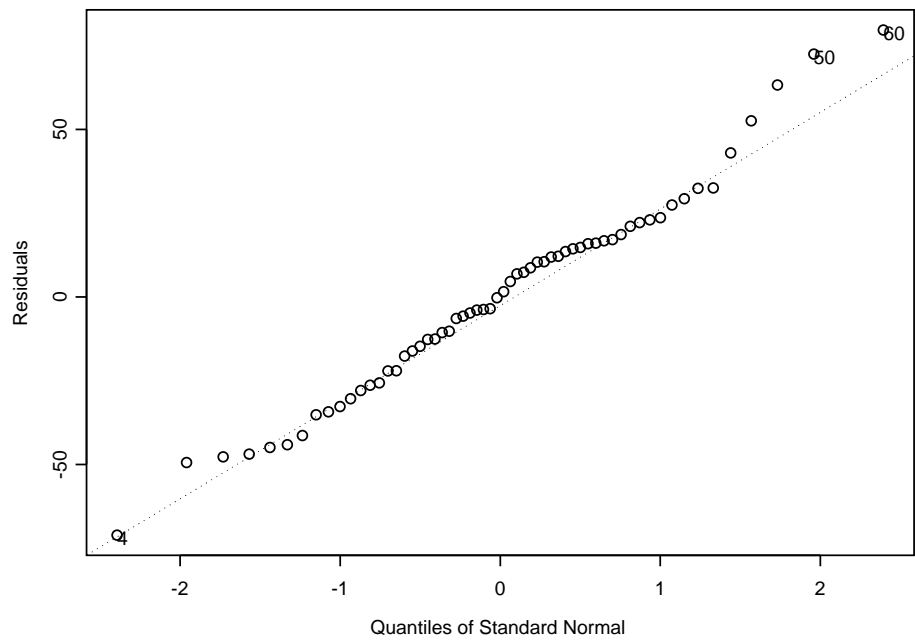
Data on the mortality rates in 60 US cities, with possible explanatory variables 4 weather variables, 8 census variables and pollution levels of hydrocarbons,  $NO_x$  and  $SO_2$ .

```
> fit <- lm(MORTALITY ~ ., data=mortality)
> fit2 <- stepAIC(fit, direction = "both", trace=F)
> fit2$anova
```

	Step	Df	Deviance	Resid. Df	Resid. Dev	AIC
1				44	53631	440
2	- POOR	1	3	45	53634	438
3	- HUMIDITY	1	13	46	53647	436
4	- WHITECOL	1	18	47	53664	434
5	- SOUND	1	215	48	53879	432
6	- S02	1	295	49	54175	430
7	- DENSITY	1	1168	50	55342	430

```
> dropterm(fit2, test="F")
```

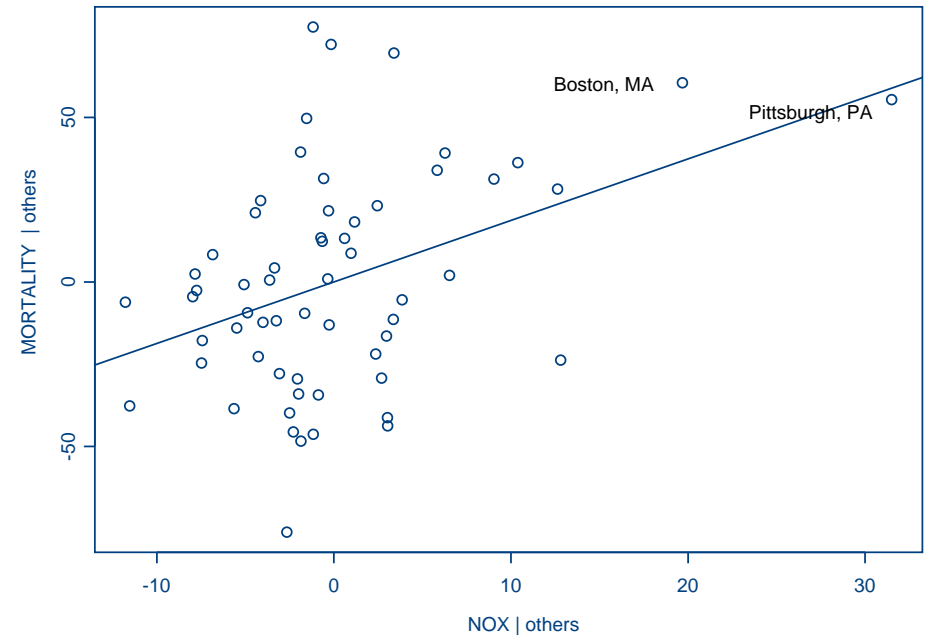
	Df	Sum of Sq	RSS	AIC	F Value	Pr(F)
<none>			55342	430		
PRECIP	1	5443	60785	433	4.9	0.031
JANTEMP	1	11709	67052	439	10.6	0.002
JULYTEMP	1	6251	61593	434	5.6	0.021
OVER65	1	2590	57932	430	2.3	0.132
HOUSE	1	5849	61191	434	5.3	0.026
EDUC	1	12175	67518	440	11.0	0.002
NONWHITE	1	25631	80973	450	23.2	0.000
HC	1	9238	64580	437	8.3	0.006
NOX	1	10433	65776	438	9.4	0.003



Added-Variable Plot



Added-Variable Plot



# Random Effects

So far all our analyses have been for what are termed *fixed effects*: we really are interested in the treatment groups. However, suppose a group of people had measured the effect of insulation in their houses. We would most likely not be interested in those houses (they have already been insulated) but in the population of similar houses, and in how much variability there is between houses (and households).

In a *random effects* model the groups are taken as a sample from a population, and the interest is in aspects of the distribution (mean and especially variance) of the group effects.

*Mixed* models have both random and fixed effects. They are covered in *Further Statistical Methods*.

## Example—Classic Nested Designs

A cooperative trial in analytical chemistry is designed to quantify sources of uncertainty.

Seven specimens were sent to six laboratories, each three times a month apart for duplicate analysis. The response is the concentration of (unspecified) analyte in g/kg. The data from Specimen 1 were

Batch	Laboratory					
	1	2	3	4	5	6
1	0.29	0.40	0.40	0.9	0.44	0.38
	0.33	0.40	0.35	1.3	0.44	0.39
2	0.33	0.43	0.38	0.9	0.45	0.40
	0.32	0.36	0.32	1.1	0.45	0.46
3	0.34	0.42	0.38	0.9	0.42	0.72
	0.31	0.40	0.33	0.9	0.46	0.79

The laboratories and batches are regarded as random effects. A model for the response for laboratory  $i$ , batch  $j$  and duplicate  $k$  is

$$y_{ijk} = \mu + \xi_i + \beta_{ij} + \epsilon_{ijk}$$

where  $\xi$ ,  $\beta$  and  $\epsilon$  are independent random variables with zero means and variances  $\sigma_L^2$ ,  $\sigma_B^2$  and  $\sigma_e^2$  respectively. For  $l$  laboratories,  $b$  batches and  $r$  duplicates a nested analysis of variance gives:

Source of variation	Degrees of freedom	Sum of squares	Mean square	E(MS)
Between laboratories	$l - 1$	$br \sum_i (\bar{y}_i - \bar{y})^2$	$MS_L$	$br\sigma_L^2 + r\sigma_B^2 + \sigma_e^2$
Batches within laboratories	$l(b - 1)$	$r \sum_{ij} (\bar{y}_{ij} - \bar{y}_i)^2$	$MS_B$	$r\sigma_B^2 + \sigma_e^2$
Replicates within batches	$lb(r - 1)$	$\sum_{ijk} (y_{ijk} - \bar{y}_{ij})^2$	$MS_e$	$\sigma_e^2$

So the unbiased estimators of the variance components are

$$\hat{\sigma}_L^2 = (br)^{-1}(MS_L - MS_B), \quad \hat{\sigma}_B^2 = r^{-1}(MS_B - MS_e), \quad \hat{\sigma}_e^2 = MS_e$$

Note that some of the variance estimators can be negative.

The model is fitted in the same way as an analysis of variance model with `raov`, or via `varcomp`:

```
> summary(raov(Conc ~ Lab/Bat, data = coop, subset = Spc=="S1"))
```

	Df	Sum of Sq	Mean Sq	Est. Var.
Lab	5	1.8902	0.37804	0.060168
Bat %in% Lab	12	0.2044	0.01703	0.005368
Residuals	18	0.1134	0.00630	0.006297

```
> coop <- coop # make a local copy
```

```
> is.random(coop) <- T
```

```
> varcomp(Conc ~ Lab/Bat, data = coop, subset = Spc=="S1")
```

Variances:

Lab	Bat %in% Lab	Residuals
0.060168	0.0053681	0.0062972

# Multiple Comparisons

In fitting, selecting an interpreting a regression model we will typically look at a lot of test statistics and/or confidence intervals. Remember that the size of a test of the coverage of a confidence interval refers to what might happen by chance *on that one test/CI*. Even then, it is for *pre-planned* procedures, and does not allow for **data snooping**.

Data snooping includes making transformations and selecting explanatory variables (or even selecting responses), and it is hard to account rigorously for selection effects in the whole data analysis procedure.

Nevertheless, some attempts have been made to compensate for data snooping, most often when looking at comparisons between different levels of a single factor. Thus the area is often known as *multiple comparisons*.

# Bonferroni Corrections

This is the simplest and most general idea. Suppose we perform  $M$  tests at sizes  $\alpha_i$ . Then the probability that any test shows rejects under the null hypothesis that all null hypotheses hold is

$$P(\text{any test rejects}) \leq \sum_i P(\text{test } i \text{ rejects}) \leq \sum_i \alpha_i$$

So if we take  $\sum_i \alpha_i \leq \alpha$  we will have an overall test of significance level  $\alpha$ .

If  $\alpha$  is small and the tests are independent, the realized size will be close to  $\alpha$ .

In our disabilities example there are  $5 \times 4/2 = 10$  possible pairwise comparisons, so taking a size of 0.5% would give an overall test of level not exceeding 5%. With 65 df this corresponds to a  $t$  multiplier of 2.90, whereas the uncorrected multiplier would be 2.00.

If we were only interested in the 4 comparisons of disabilities with the control the multiplier would be 2.57.

# Tukey HSD

Now consider just pairwise comparisons  $|\overline{Y}_j - \overline{Y}_i|$ . These all have the same standard error (since we have equal replication), and the largest one is  $\overline{Y}_{\max} - \overline{Y}_{\min}$ .

Thus all comparisons will be insignificant at level  $\alpha$  if  $T = (\overline{Y}_{\max} - \overline{Y}_{\min})/s < M$  for a suitably chosen  $M$ . The distribution of  $T$  is known: it is called the *studentized range* and we can look up the appropriate multiplier in **S-PLUS** using `qtukey(0.95, 5, 65)/sqrt(2) = 2.81`.

This method is known as Tukey's "honest significant difference".

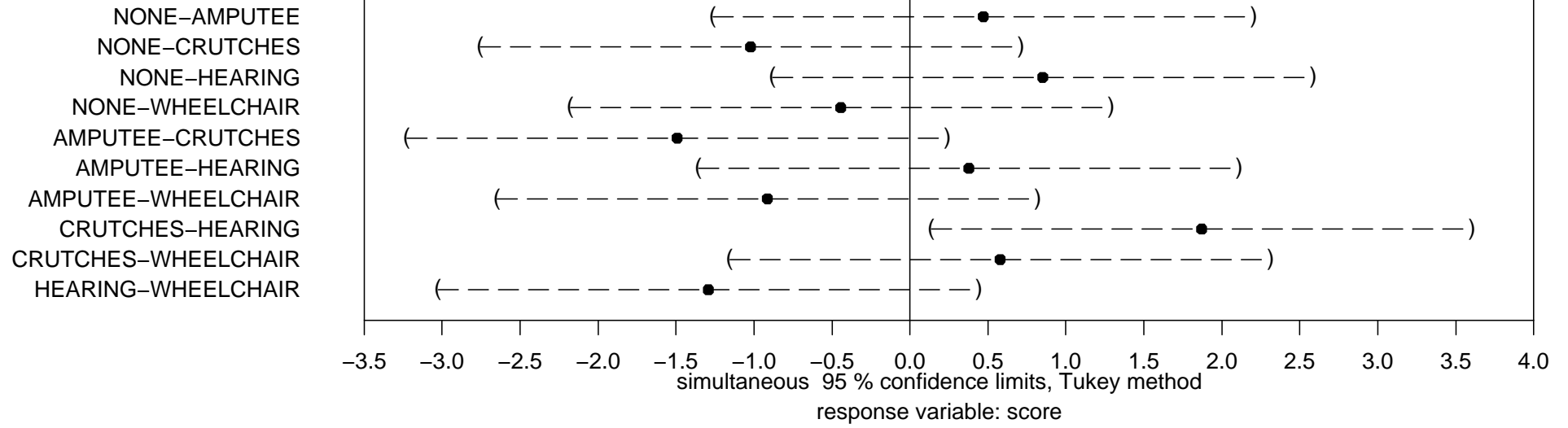
```
> multicomf(fit)
```

95 % simultaneous confidence intervals for specified  
linear combinations, by the Tukey method

critical point: 2.8058

intervals excluding 0 are flagged by '\*\*\*\*'

	Estimate	Std.Error	Lower Bound	Upper Bound	
NONE-AMPUTEE	0.471	0.617	-1.260	2.200	
NONE-CRUTCHES	-1.020	0.617	-2.750	0.710	
NONE-HEARING	0.850	0.617	-0.882	2.580	
NONE-WHEELCHAIR	-0.443	0.617	-2.170	1.290	
AMPUTEE-CRUTCHES	-1.490	0.617	-3.220	0.239	
AMPUTEE-HEARING	0.379	0.617	-1.350	2.110	
AMPUTEE-WHEELCHAIR	-0.914	0.617	-2.650	0.817	
CRUTCHES-HEARING	1.870	0.617	0.140	3.600	****
CRUTCHES-WHEELCHAIR	0.579	0.617	-1.150	2.310	
HEARING-WHEELCHAIR	-1.290	0.617	-3.020	0.439	



# Scheffé's Correction

Scheffé considered testing all possible contrasts (zero-sum linear combinations) of the group means. This leads to multiplier

$$\sqrt{(K - 1)F_{K-1,df}(1 - \alpha)}$$

for  $K$  groups and a scale estimate on  $df$  degrees of freedom. In our example this gives multiplier 3.17.

Scheffé's correction is for a much larger class of tests, and for just looking at pairwise differences it tends to over-correct.

# Comparisons with a Control

There are other methods which specialize in comparing all other treatments with a control: Dunnett's is the most popular. Here the multiplier is only just less than that given by Bonferroni.

```
> multcomp(fit, comparisons = "mcc", control = 1)
```

```
95 % simultaneous confidence intervals for specified  
linear combinations, by the Dunnett method
```

```
critical point: 2.5032
```

```
intervals excluding 0 are flagged by '****'
```

	Estimate	Std.Error	Lower Bound	Upper Bound
AMPUTEE-NONE	-0.471	0.617	-2.020	1.070
CRUTCHES-NONE	1.020	0.617	-0.524	2.570
HEARING-NONE	-0.850	0.617	-2.390	0.695
WHEELCHAIR-NONE	0.443	0.617	-1.100	1.990

## Additional Example on Factorial Designs

Consider an experiment on noise of a washing-machine bearing. There were three factors, IR (inner ring smooth or rough), OR (outer ring smooth or rough) and the type of ball (G5 or G10, the former being rounder). The results are given in the table, small meaning low noise (preferable).

IR	OR	G5	G10
s	s	25	38
s	r	26	43
r	s	87	76
r	r	81	67

One way to enter the data in S-PLUS is

```
IR <- c("s", "s", "r", "r"); IR <- c(IR, IR)
OR <- c("s", "r", "s", "r"); OR <- c(OR, OR)
ball <- rep(c("G5", "G10"), each=4)
noise <- scan()
25 26 87 81 38 43 76 67
bearings <- data.frame(IR, OR, ball, noise)
rm(IR, OR, ball, noise)
```

Then a basic analysis is given by

```
> summary(fm <- aov(noise ~ .^2, bearings))
```

	Df	Sum of Sq	Mean Sq	F Value	Pr(F)
IR	1	4005.125	4005.125	653.8980	0.0248831
OR	1	10.125	10.125	1.6531	0.4208332
ball	1	3.125	3.125	0.5102	0.6051369
IR:OR	1	55.125	55.125	9.0000	0.2048328
IR:ball	1	378.125	378.125	61.7347	0.0805911
OR:ball	1	0.125	0.125	0.0204	0.9096655
Residuals	1	6.125	6.125		

The model can be refined by dropping unneeded interactions and so freeing up more degrees of freedom to estimate the error variance. This can be achieved by the commands `dropterm`, `update` or `stepAIC` from `library(MASS)`.

```
> summary(stepAIC(fm, trace = F))
      Df Sum of Sq Mean Sq F Value Pr(F)
IR     1  4005.125 4005.125 1281.64 0.0007793
OR     1    10.125  10.125   3.24 0.2136663
ball   1     3.125   3.125   1.00 0.4226497
IR:OR  1    55.125  55.125  17.64 0.0522833
IR:ball 1   378.125 378.125 121.00 0.0081634
Residuals 2     6.250   3.125
```

This yields the model

$$\text{Noise} = 55.4 - 22.4 \text{ IR} + 1.1 \text{ OR} - 0.6 \text{ ball} - 2.6 \text{ IR:OR} - 6.9 \text{ IR:ball}$$

It is always important to check the contrasts, as the explanatory variables may mean differently under different contrasts. The above model is obtained using Helmert contrasts. Using treatment contrasts we get

```
> options(contrasts=c("contr.treatment","contr.poly"))
> stepAIC(aov(noise~.^2,bearings), trace = F)$coefficients
(Intercept)      IR  OR ball IR:OR IR:ball
      67.75 -25.75  7.5 12.5 -10.5   -27.5
```

and this yields the model

```
Noise = 67.8 - 25.8 IR + 7.5 OR + 12.5 ball - 10.5 IR:OR - 27.5 IR:ball
```

This highlights the importance of checking the contrasts used, although you should also check that both models produce the same fitted values.