8

Regression

Regression analysis is the statistical method you use when both the response variable and the explanatory variable are continuous variables (i.e. real numbers with decimal places – things like heights, weights, volumes, or temperatures). Perhaps the easiest way of knowing when regression is the appropriate analysis is to see that a scatter plot is the appropriate graphic (in contrast to analysis of variance, say, when the plot would have been a box and whisker or a bar chart). We cover four important kinds of regression analysis:

- linear regression (the simplest, and much the most frequently used),
- polynomial regression (often used to test for non-linearity in a relationship),
- non-linear regression (to fit a specified non-linear model to data), and
- non-parametric regression (used when there is no obvious functional form).

The essence of regression analysis is using sample data to estimate parameter values and their standard errors. First, however, we need to select a model which describes the relationship between the response variable and the explanatory variable(s). The simplest model of all, is the linear model:

$$y = a + bx.$$

The response variable is y, and x is a continuous explanatory variable. There are two parameters, a and b: the intercept is a (that is the value of y when $x = 0$); and the slope is b (the slope, or gradient, is the change in y divided by the change in x which brought it about). The slope is so important, that it is worth drawing a picture to make clear what is involved.

The example refers to oil drums in a store: on week 2 there were 16 drums and when the next stock-taking was carried out on week 5 there were 10 drums left. So x is time in weeks and y is the number of full oil drums. All we know is that the graph goes through the point (2,16) and the point (5,10). Remember that when specifying coordinates on a graph (Cartesian coordinates) the x value comes first, then the y value. So the two x values

Statistics: An Introduction using R M. J. Crawley
 ISBNs: 0-470-02298-1 (PBK); 0-470-02297-3 (PPC)

are 2 and 5 and the two y values are 16 and 10. We see at once that y gets smaller as x increases, so the value of the slope is going to be negative. First we plot the axes of the graph, but put nothing (yet) between the axes (this is graph `type = "n"`):

```
plot(c(2,5),c(16,10),type="n",ylab="y",xlab="x",ylim=c(0,20),xlim=c(0,6))
```

Note that in the plot function, the x values `c(2,5)` are grouped together in the first argument and the y values `c(16,10)` are grouped together in the second (i.e. the arguments of plot are **not** Cartesian coordinates even though, as here, they sometimes look as if they might be).

Let's add the two points to the graph as solid circles (this is plotting character `pch = 16`)

```
points(c(2,5),c(16,10),pch=16)
```

Now to calculate the slope, we need to know the change in y. On the graph this is a vertical line (i.e. parallel with the y axis); the line representing the change in y would be drawn like this

```
lines(c(2,2),c(16,10))
```

Do you see how this worked? The x value did not change (so both x coordinates were 2). The top of the line was $y = 16$ and the bottom of the line was 10). Let's label this line delta y:

```
text(1,13,"delta y")
```

The next thing we need to calculate the slope is the change in x. We draw a line to represent the change in x like this:

```
lines(c(2,5),c(10,10))
```

We can label this line delta x:

```
text(3.5,8,"delta x")
```

You need to work out the x and y coordinates for locating text by trial and error after you have looked at the graph. Alternatively, there is a function called `locator(1)` that enables you to get the coordinate values of one point using the mouse to locate the cursor, then left-clicking. Note that text is centred on the location you specify (not, for instance, printed from a specified lower-left corner). Now, we can calculate slope, b, as

$$b = \frac{\text{change in } y}{\text{change in } x \text{ that brought it about}}.$$

So for our example, $b = (10 - 16)/(5 - 2) = -6/3 = -2.0$. We can draw the line with slope $= -2$ between the two points like this:

```
lines(c(2,5),c(16,10))
```

but there is a very useful function in R called abline which draws a line from exactly one edge of the plotting area to another. To use this, we need to know the value of a, the intercept. Now that we know that $b = -2.0$, this is easy. Take any one of the two known coordinates (say {2,16}) and rearrange the equation to find a. To some of you, this may be second nature, but to others it may be really hard. We'll work though this example to show what's involved. Start with what we know:

$$y = a + bx.$$

Now we know y (16), we know x (2) and we know $b(-2)$. How do we get a out of this equation? We know that $a + bx$ is equal to y, so if we subtract bx from both sides of the equation, we are left with:

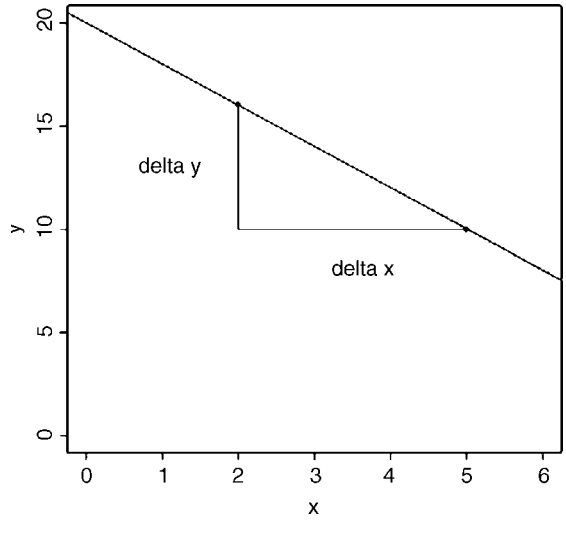
$$y - bx = a + bx - bx.$$

The $+bx$ and $-bx$ cancel out, so

$$a = y - bx.$$

We can work this out for our example: $a = 16 - (-2 \times 2)$. Remember that 'minus minus equals plus' so $a = 16 + 4 = 20$. Now we are in a position to use abline to draw a line right across the plotting area: the arguments of abline are first a, then b, like this:

```
abline(20,-2)
```



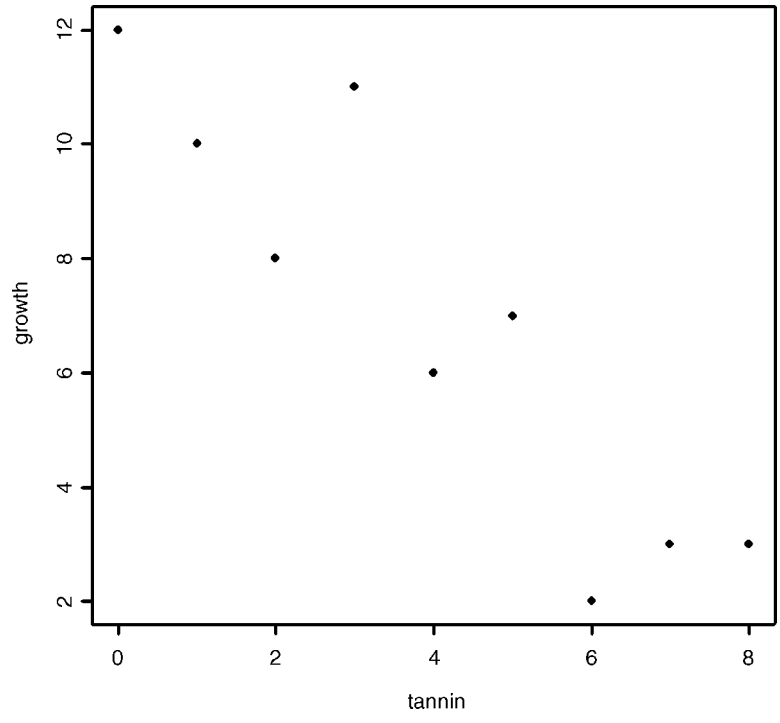
Linear Regression

Let's start with an example. The thing to understand is that there is nothing difficult, or mysterious about estimating the regression parameters. We can do a reasonable job, just by eye.

```
reg.data < -read.table("c:\\temp\\tannin.txt",header = T)
attach(reg.data)
names(reg.data)
```

```
[ 1] "growth" "tannin"
```

```
plot(tannin,growth,pch = 16)
```



This is how we do regression 'by eye'. We ask 'what has happened to the y value ?'. It decreased from about 12 to about 2, so delta y (the change in y) $= -10$ (the minus sign is important). How did the x value change? It increased from 0 to 8, so the change in x is $+8$. (tip: when working out regressions by eye, it is a good idea to take as big a range of x values as possible, so here we took the complete range of x.). What is the value of y when $x = 0$? It is about 12, so the intercept $a = 12$. Finally, what is the value of b? It is the change in y (-10) divided by the change in x which brought it about (8), so $b = -10/8 = -1.25$. So our rough guess at the regression equation is

$$y = 12.0 - 1.25x.$$

That's all there is to it. Obviously, we want to make the procedure more objective than this. We also want to estimate the unreliability of the two estimated parameters (i.e. the standard errors of the slope and intercept), but the basics are just as straightforward as that.

Linear Regression in R

How close did we get to the maximum likelihood estimates of a and b with our guesstimates of 12 and -1.25? It is easy to find out using the R function lm which stands for 'Linear Model' (the first letter is a lower case L, not a number one). All we need do, is to tell R which of the variables is the response variable (growth in this case) and which is the explanatory variable (tannin concentration in the diet). The response variable goes on the left of the tilde ~ and the explanatory variable goes on the right: growth ~ tannin. This is read 'growth is modelled as a function of tannin'. Now we write:

```
lm(growth ~ tannin)

Coefficients:
 (Intercept)          tannin
      11.756          -1.217
```

The two parameters are called 'coefficients' in R: the intercept is 11.756 (compared with out guesstimate of 12) and the slope is -1.217 (compared with our guesstimate of -1.25). Not bad at all.

So where does R get its coefficients from? We need to do some calculations to find this out. If you are more mathematically inclined, you might like to work through Box 8.1, but this is not essential to understand what is going on. Remember that what we want are the maximum likelihood estimates of the parameters. That is to say, that given the data, and having selected a linear model, we want **to find the values of the slope and intercept that make the data most likely**. Keep re-reading this sentence until you understand what it is saying.

Box 8.1. The least-squares estimate of the regression slope, *b*

The **best fit** slope is found by rotating the line until the **error sum of squares**, **SSE**, is minimized, so we want to find the minimum of $\sum (y - a - bx)^2$. We start by finding the derivative of *SSE* with respect to b

$$\frac{dSSE}{db} = -2\sum x(y - a - bx).$$

Now, multiplying through the bracketed term by x gives

$$\frac{dSSE}{db} = -2\sum xy - ax - bx^2.$$

Apply summation to each term separately, set the derivative to zero, and divide both sides by -2 to remove the unnecessary constant:

$$\sum xy - \sum ax - \sum bx^2 = 0.$$

We cannot solve the equation as it stands because there are two unknowns, a and b. However, we know the value of a is $\bar{y} - b\bar{x}$. Also, note that $\sum ax$ can be written as $a\sum x$, so, replacing a and taking both a and b outside their summations gives:

$$\sum xy - \left[\frac{\sum y}{n} - b\frac{\sum x}{n}\right]\sum x - b\sum x^2 = 0.$$

Now multiply out the central bracketed term by $\sum x$ to get

$$\sum xy - \frac{\sum x \sum y}{n} + b\frac{(\sum x)^2}{n} - b\sum x^2 = 0.$$

Finally, take the two terms containing b to the right-hand side, and note their change of sign:

$$\sum xy - \frac{\sum x \sum y}{n} = b\sum x^2 - b\frac{(\sum x)^2}{n},$$

then divide both sides by $\sum x^2 - (\sum x)^2/n$ to obtain the required estimate b:

$$b = \frac{\sum xy - \dfrac{\sum x \sum y}{n}}{\sum x^2 - \dfrac{(\sum x)^2}{n}}.$$

Thus, the value of b that minimizes the sum of squares of the departures is given simply by

$$b = \frac{SSXY}{SSX}.$$

This is **the maximum likelihood estimate of the slope** of the linear regression.

The best way to see what is going on is to do it graphically. Let's cheat a bit by fitting the best-fit straight line through our scatterplot, using abline with a linear model, like this:

```
abline(lm(growth~tannin))
```

The fit is reasonably good, but it is not perfect. The data points do not lie on the fitted line. The difference between each data point and the value predicted by the model at the

same value of x is called a **residual**. Some residuals are positive (above the line) and others are negative (below the line). Let's draw vertical lines to indicate the size of the residuals. The first x point is at tannin = 0. The y value measured at this point was growth = 12, but what is the growth predicted by the model at tannin = 0? There is a built-in function called predict to work this out:

```
fitted <-predict(lm(growth ~tannin))
fitted
```

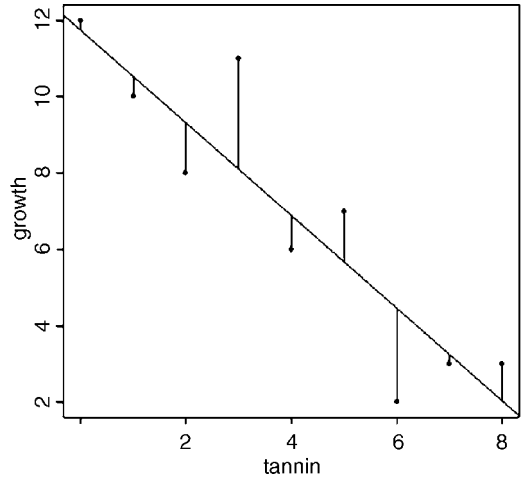
```
       1         2         3         4         5         6         7         8
11.755556 10.538889  9.322222  8.105556  6.888889 5.672222 4.455556 3.238889
       9
2.022222
```

So the first predicted value of growth is 11.75555. To draw the first residual, both x coordinates will be 0. The first y coordinate will be 12 (the observed value) and the second will be 11.7555 (the fitted (or predicted) value). We use lines, like this:

```
lines(c(0,0),c(12,11.7555555))
```

We could go through, laboriously, and draw each residual like this, but it is much quicker to automate the procedure, using a loop to deal with each residual in turn:

```
for (i in 1:9) lines (c(tannin[i],tannin[i]),c(growth[i],fitted[i]))
```



These residuals describe the goodness of fit of the regression line. Our maximum likelihood model is defined as **the model that minimizes the sum of the squares of these residuals**. It is useful, therefore, to write down exactly what any one of the residuals, d, is – it is the measured value, y, minus the fitted value called $\hat{y}$ (y 'hat'):

$$d = y - \hat{y}.$$

We can improve on this, because we know that $\hat{y}$ is on the straight line $a + bx$, so

$$d = y - (a + bx) = y - a - bx.$$

The equation includes $a - bx$ because of the minus sign outside the bracket. Now our best fit line, by definition, is given by the values of a and b that minimize the sums of the squares of the d's (see Box 8.1). Note, also, that just as $\sum (y - \bar{y}) = 0$ (Box 4.1), so $\sum d = \sum (y - a - bx) = 0$ (Box 8.2).

Box 8.2. Proof that $\sum (y - a - bx) = 0$

Take the summation through each of the terms, bearing in mind that $\sum a = na$ and $\sum bx = b \sum x$

$$\sum y - na - b \sum x.$$

We also know that the linear regression $y = a + bx$ passes through the point $(\bar{x}, \bar{y})$ defined by the mean values of x and y, so it must be the case that $\bar{y} = a + b\bar{x}$. Replacing $\bar{y}$ by $\sum y/n$ and $\bar{x}$ by $\sum x/n$ allows us to work out the value of $\sum y$

$$\frac{\sum y}{n} = a + b\frac{\sum x}{n},$$

so

$$\sum y = na + b \sum x$$

because the n's cancel. Now we substitute this value for $\sum y$:

$$\sum y - na - b \sum x = na + b \sum x - na - b \sum x = 0$$

as required for the proof that $\sum (y - a - bx) = 0$.

We want to find the minimum of $\sum d^2 = \sum (y - a - bx)^2$. To work this out we need 'the famous five' which are $\sum y^2$ and $\sum y$, $\sum x^2$ and $\sum x$ and a new quantity, $\sum xy$, the sum of products. The sum of products is worked out pointwise, so for our data, it is:

tannin

```
[1] 0 1 2 3 4 5 6 7 8
```

growth

```
[1] 12 10 8 11 6 7 2 3 3
```

tannin*growth

```
[1] 0 10 16 33 24 35 12 21 24
```

zero times 12 = 0, plus one times 10 = 10, plus two times 8 = 16, and so on:

```
sum(tannin*growth)
```

```
[1] 175
```

The next thing is to use the famous five to work out three essential 'corrected sums': the corrected sum of squares of x, the corrected sum of squares of y and the corrected sum of products, $x.y$. The corrected sums of squares of x and y should be familiar to you:

$$SSY = \sum y^2 - \frac{(\sum y)^2}{n}$$
$$SSX = \sum x^2 - \frac{(\sum x)^2}{n}$$

because if we wanted the variance in y, we would just divide SSY by its degrees of freedom (and likewise for the variance in x; see p. 137). It is only the corrected sum of products that is novel, but its structure is directly analogous. Think about the formula for SSY, above. It is 'the sum of y times y' $\sum y^2$, 'minus the sum of y times the sum of y' $(\sum y)^2$ 'divided by the sample size', n. The formula for SSX is similar. It is 'the sum of x times x' $\sum x^2$, 'minus the sum of x times the sum of x' $(\sum x)^2$ 'divided by the sample size', n:

$$SSXY = \sum xy - \frac{(\sum x)(\sum y)}{n}.$$

If you look carefully you will see that the corrected sum of products has exactly the same kind of structure. It is 'the sum of x times y' $\sum xy$, 'minus the sum of x times the sum of y' $(\sum x)(\sum y)$ 'divided by the sample size', n.

These three corrected sums of squares are absolutely central to everything that follows about regression analysis, so it is a good idea to re-read this section as often as necessary, until you are confident that you understand what SSX, SSY and $SSXY$ represent (Box 8.3).

Box 8.3. Corrected sums of squares and products in regression

The total sum of squares is SSY, the corrected sum of products is $SSXY$ and the sum of squares of x is SSX:

$$SSY = \sum y^2 - \frac{(\sum y)^2}{n}$$
$$SSX = \sum x^2 - \frac{(\sum x)^2}{n}$$
$$SSXY = \sum xy - \frac{\sum x \sum y}{n}.$$

The explained variation is the regression sum of squares, *SSR*:

$$SSR = \frac{SSXY^2}{SSX}.$$

The unexplained variation is the error sum of squares, *SSE*, can be obtained by difference

$$SSE = SSY - SSR,$$

but *SSE* is defined as **the sum of the squares of the residuals** which is

$$SSE = \sum (y - a - bx)^2.$$

The correlation coefficient, *r*, is given by

$$r = \frac{SSXY}{\sqrt{SSX \times SSY}}.$$

The next question is how we use *SSX, SSY* and *SSXY* to find the maximum likelihood estimates of the parameters and their associated standard errors. It turns out that this step is much simpler than what has gone before. The maximum likelihood estimate of the slope, b, is just:

$$b = \frac{SSXY}{SSX}$$

(the detailed derivation of this is in Box 8.1). Now that we know the value of the slope, we can use any point on the fitted straight line to work out the maximum likelihood estimate of the intercept, a. One part of the definition of the best-fit straight line is that it passes through the point $(\bar{x}, \bar{y})$ determined by the mean values of x and y. Since we know that $y = a + bx$, it must be the case that $\bar{y} = a + b\bar{x}$, and so

$$a = \bar{y} - b\bar{x} = \frac{\sum y}{n} - b.\frac{\sum x}{n}.$$

Box 8.4. The shortcut formula for the sum of products, *SSXY*

SSXY is based on the expectation of the product $(x - \bar{x})(y - \bar{y})$. Start by multiplying out the brackets:

$$(x - \bar{x})(y - \bar{y}) = xy - x\bar{y} - y\bar{x} + \bar{x}.\bar{y}.$$

Now apply the summation remembering that $\sum \bar{y} = n.\bar{y}$ and $\sum x.\bar{y} = \bar{y}\sum x$

$$\sum xy - \bar{y}\sum x - \bar{x}\sum y + n.\bar{x}.\bar{y} = \sum xy - n.\bar{y}.\bar{x} - n.\bar{x}.\bar{y} + n.\bar{x}.\bar{y}\sum xy - n.\bar{x}.\bar{y}$$

because $\sum x = n.\bar{x}$ and $\sum y = n.\bar{y}$. Now replace the product of the two means by $\sum x/n \times \sum y/n$

$$\sum xy - n\frac{\sum x}{n}.\frac{\sum y}{n}$$

which, on cancelling the n's gives the corrected sum of products as

$$SSXY = \sum xy - \frac{\sum x \sum y}{n}.$$

We can work out the parameter values for our example. To keep things as simple as possible, we can call the variables *SSX*, *SSY* and *SSXY* (note that R is 'case sensitive' so the variable *SSX* is different from *ssx*):

```
SSX = sum(tannin^2)-sum(tannin)^2/length(tannin)
SSX
```

```
[1] 60
```

```
SSY = sum(growth^2)-sum(growth)^2/length(growth)
SSY
```

```
[1] 108.8889
```

```
SSXY = sum(tannin*growth)-sum(tannin)*sum(growth)/length(tannin)
SSXY
```

```
[1] -73
```

That's all we need. So the slope is:

$$b = \frac{SSXY}{SSX} = \frac{-73}{60} = -1.2166667$$

and the intercept is given by:

$$a = \frac{\sum y}{n} - b.\frac{\sum x}{n} = \frac{62}{9} + 1.2166667\frac{36}{9} = 6.8889 + 4.86667 = 11.755556.$$

Now we can write the maximum likelihood regression equation in full:

$$y = 11.75556 - 1.216667x.$$

This, however, is only half of the story. In addition to the parameter estimates, $a = 11.756$ and $b = -1.2167$, we need to measure the unreliability associated with each of the estimated parameters. In other words, we need to calculate the standard error of the intercept and the standard error of the slope. We have already met the standard error of a mean, and we used it in calculating confidence intervals (p. 45) and in doing Student's *t*-test (p. 77). Standard errors of regression parameters are similar in so far as they are enclosed inside a big square root term (so that the units of the standard error are the same as the units of the parameter), and they have the error variance, s^2, in the numerator. There are extra components, however, which are specific to the unreliability of a slope or an intercept (see Boxes 8.6 and 8.7 for details), but we cannot work out the standard errors until we know the value of the error variance s^2 and to do this, we need to carry out an analysis of variance.

Error Variance in Regression: $SSY = SSR + SSE$

The idea is simple – we take the total variation in *y*, *SSY*, and partition it into components that tell us about the explanatory power of our model. The variation that is explained by the model is called the regression sum of squares (denoted by *SSR*), and the unexplained variation is called the error sum of squares (denoted by *SSE* that we drew on the scatterplot, earlier). Then $SSY = SSR + SSE$ (the proof is presented in Box 8.5). Now, in principle, we could compute *SSE* because we know that it is the sum of the squares of the deviations of the data points from the fitted model, $\sum d^2 = \sum (y - a - bx)^2$. Since we know the values of *a* and *b*, we are in a position to work this out. The formula is fiddly, however, because of all those subtractions, squarings and addings-up. Fortunately, there is a very simple shortcut that involves computing *SSR*, the explained variation, rather than *SSE*. This is because

$$SSR = b.SSXY$$

so we can immediately work out $SSR = -1.21667 \times -73 = 88.81667$; and since $SSY = SSR + SSE$ we can get *SSE* by subtraction:

$$SSE = SSY - SSR = 108.8889 - 88.81667 = 20.07222.$$

These components are now drawn together in what is known as the 'Anova table'. Strictly, we have analysed sums of squares, rather than variances up to this point, but you will see why it is called analysis of variance shortly. The left-most column of the Anova table lists the sources of variation: regression, error and total in our example. The next column contains the sums of squares, *SSR*, *SSE* and *SSY*. The third column is in many ways the most important to understand; it contains the degrees of freedom. There are *n* points on the graph ($n = 9$ in this example). So far, our table looks like this.

Source	Sum of squares	Degrees of freedom	Mean squares	*F* ratio
Regression	88.817			
Error	20.072			
Total	108.889			

We shall work out the degrees of freedom associated with each of the sums of squares in turn. The easiest to deal with is the total sum of squares, because it always has the same formula for its degrees of freedom. The definition is $SSY = \sum (y - \bar{y})^2$, and you can see that there is just one parameter estimated from the data: the mean value, $\bar{y}$. Because we have estimated one parameter from the data, we have $n - 1$ degrees of freedom. The next easiest to work out is the error sum of squares. Let's look at its formula to see how many parameters need to be estimated from the data before we can work out $SSE = \sum (y - a - bx)^2$. We need to know the values of both a and b before we can calculate SSE. These are estimated from the data, so the degrees of freedom for error are $n - 2$. This is important, so re-read the last sentence if you don't see it yet. The most difficult of the three is the regression degrees of freedom, because you need to think about this one in a different way. The question is this: how many extra parameters, over and above the mean value of y, did you estimate when fitting the regression model to the data? The answer is one. The extra parameter you estimated was the slope, b. So the regression degrees of freedom in this simple model, with just one explanatory variable, is 1. This will only become clear with practice. To complete the Anova table, we need to understand the fourth column, headed 'mean squares'. This column contains the variances, on which analysis of variance is based. The key point to recall is that

$$\text{variance} = \frac{\text{sum of squares}}{\text{degrees of freedom}}.$$

This is very easy to calculate in the context of the Anova table, because the relevant sums of squares and degrees of freedom are in adjacent columns. Thus the regression variance is just $SSR/1 = SSR$, and the error variance is $s^2 = SSE/(n - 2)$. Traditionally, one does not fill in the bottom box (it would be the overall variance in y, $SSY/(n - 1)$). Finally, the Anova table is completed by working out the F ratio, which is a ratio between two variances. In most simple Anova tables, you divide the treatment variance in the numerator (the regression variance in this case) by the error variance s^2 in the denominator. The null hypothesis under test in a linear regression is that the slope of the regression line is zero (i.e. no dependence of y on x). The two-tailed alternative hypothesis is that the slope is significantly different from zero (either positive or negative). In many applications it is not particularly interesting to reject the null hypothesis, because we are interested in the estimates of the slope and its standard error (we often know from the outset that the null hypothesis is false). To test whether the F ratio is sufficiently large to reject the null hypothesis, we compare the calculated value of F in the final column of the Anova table with the critical value of F, expected by chance alone (this is found from quantiles of the F distribution qf, with 1 d.f. in the numerator and $n - 2$ d.f. in the denominator, as described below). Here is the completed Anova table:

Source	Sum of squares	Degrees of freedom	Mean squares	F ratio
Regression	88.817	1	88.817	30.974
Error	20.072	7	$s^2 = 2.86746$	
Total	108.889	8		

Notice that the component degrees of freedom add up to the total degrees of freedom (this is always true, in any Anova table, and is a good check on your understanding of the design of the experiment). The last question concerns the magnitude of the F ratio $= 30.974$: is it big enough to justify rejection of the null hypothesis? The critical value of the F ratio is the value of F that would arise due to chance alone when the null hypothesis was true, given that we have 1 d.f. in the numerator and 7 d.f. in the denominator. We have to decide on the level of uncertainty that we are willing to put up with; the traditional value for work like this is 5%, so our certainty $= 0.95$. Now we can use quantiles of the F distribution, qf, to find the critical value:

```
qf(0.95,1,7)
```

```
[1] 5.591448
```

Because our calculated value of F (30.974) is much larger than the critical value (5.591), we can be confident in rejecting the null hypothesis. Perhaps a better thing to do, rather than working rigidly at the 5% uncertainty level, is to ask what is the probability of getting a value for F as big as 30.974 or larger if the null hypothesis is true. For this we use 1 – pf rather than qf:

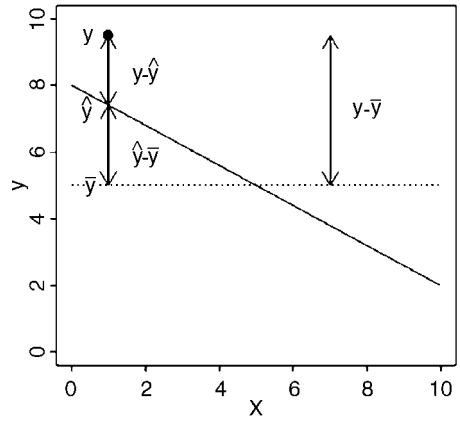
```
1-pf(30.974,1,7)
```

```
[1] 0.0008460725
```

It is very unlikely indeed ($p < 0.001$). Up to this point we have assumed that $SSY = SSR + SSE$; see Box 8.5 for the proof.

Box 8.5. Proof that $SSY = SSR + SSE$

Let's start with what we know. The difference between y and $\bar{y}$ is the sum of the differences $(y - \hat{y})$ and $(\hat{y} - \bar{y})$ as you can see from the figure:



However, it is not at all obvious that the sum of the squares of these quantities, $(y-\hat{y})^2+(\hat{y}-\bar{y})^2$, should be equal to $(y-\bar{y})^2$. We begin by squaring $(y-\hat{y})+(\hat{y}-\bar{y})$ to see where this gets us. Remember that $(a+b)^2$ is a^2+b^2+2ab so let's write out the square of the sum in full:

$$(y-\hat{y})^2+(\hat{y}-\bar{y})^2+2(y-\hat{y})(\hat{y}-\bar{y}).$$

Now we apply summation

$$\sum(y-\hat{y})^2+\sum(\hat{y}-\bar{y})^2+2\sum(y-\hat{y})(\hat{y}-\bar{y}).$$

The first term is $SSE=\sum(y-\hat{y})^2$, the second term is $SSR=\sum(\hat{y}-\bar{y})^2$ and the third term, $2\sum(y-\hat{y})(\hat{y}-\bar{y})$, needs to be equal to zero if SSY is to be equal to $SSE+SSR$ as we aim to prove. The first step is to replace $\hat{y}$ and $\bar{y}$ by their relations to x in the right-most term: $\hat{y}=a+bx$ and $\bar{y}=a+b\bar{x}$

$$2\sum(y-a-bx)(a+bx-(a+b\bar{x})).$$

The minus sign outside the inner bracket means that this becomes

$$2\sum(y-a-bx)(bx-b\bar{x})$$

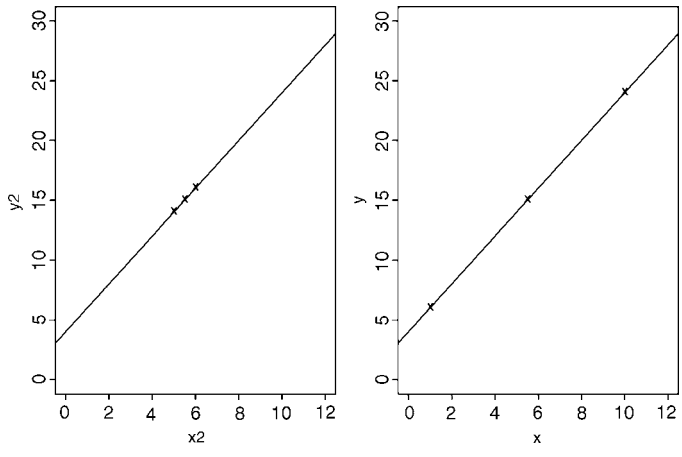
because the a's cancel out. This summation will be zero if both $\sum(y-a-bx)$ and $\sum(bx-b\bar{x})$ are zero. We have already proved that $\sum(y-a-bx)=0$ in Box 8.2 and that $\sum(x-\bar{x})=0$ in Box 4.1 (admittedly in the guise of $\sum(y-\bar{y})=0$), so all we need to note is that $\sum(bx-b\bar{x})$ can be written as $b\sum(x-\bar{x})$ to complete the proof $(b\times 0=0)$.

Next, we can use the calculated error variance $s^2=2.867$ to work out the standard errors of the slope (Box 8.6) and the intercept (Box 8.7). First the standard error of the slope:

$$\text{s.e.}_b=\sqrt{\frac{s^2}{SSX}}=\sqrt{\frac{2.867}{60}}=0.2186.$$

Box 8.6. The standard error of the regression slope, b, is given by: $s.e._b=\sqrt{\frac{s^2}{SSX}}$

The error variance s^2 comes from the Anova table and is the quantity used in calculating standard errors and confidence intervals for the parameters, and in carrying out hypothesis testing. SSX measures the spread of the x values along the x axis. Recall that standard errors are **unreliability estimates**. Unreliability increases with the error variance so it makes sense to have s^2 in the numerator (on top of the division). It is less



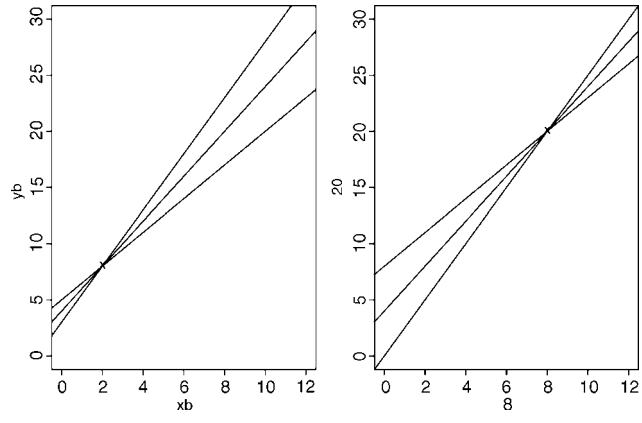
obvious why unreliability should depend on the range of x values. Look at these two graphs that have exactly the same slopes and intercepts. The difference is that the left-hand graph has all of its x values close to the mean value of x, while the graph on the right has a broad span of x values. Which of these do you think would give the most reliable estimate of the slope? It is pretty clear that it is the graph on the right, with the wider range of x values. Increasing the spread of the x values reduces unreliability of the estimated slope and hence appears in the denominator (on the bottom of the equation). What is the purpose of the big square root term? This is there to make sure that the units of the unreliability estimate are the same as the units of the parameter whose unreliability is being assessed. The error variance is in units of y **squared**, but the slope is in units of y per unit change in x.

The formula for the standard error of the intercept is a little more involved (Box 8.7)

$$\text{s.e.}_a = \sqrt{\frac{s^2 \sum x^2}{n \times SSX}} = \sqrt{\frac{2.867 \times 204}{9 \times 60}} = 1.0408.$$

Box 8.7. The standard error of the intercept, a, is given by: $s.e._a = \sqrt{\frac{s^2 \sum x^2}{n.SSX}}$

This is like the formula for the standard error of the slope, but with two additional terms. Uncertainty declines with increasing sample size n. It is less clear why uncertainty should increase with $\sum x^2$. The reason for this is that uncertainty in the estimate of the intercept increases, the further away from the intercept that the mean value of x lies. You can see this from the following graphs. On the left is a graph with a low value of $\bar{x}$ and on the right an identical graph (same slope and intercept) but estimated from a data set with a higher value of $\bar{x}$. In both cases there is a 25% variation in the slope. Compare the difference in the prediction of the intercept in the two cases.



Confidence in predictions made with linear regression declines with the square of the distance between the mean value of x and the value of x at which the prediction is to be made (i.e. with $(x - \bar{x})^2$). Thus, when the origin of the graph is a long way from the mean value of x, the standard error of the intercept will be large, and vice versa. In general, the **standard error for a predicted value** $\hat{y}$ is given by:

$$\text{s.e.}_{\hat{y}} = \sqrt{s^2\left[\frac{1}{n} + \frac{(x - \bar{x})^2}{SSX}\right]}.$$

Note that the formula for the standard error of the intercept is just the special case of this for $x = 0$ (you should check the algebra of this result as an exercise).

Now that we know where all the numbers come from, we can repeat the analysis in R and see just how straightforward it is. It is good practice to give the statistical model a name: 'model' is as good as any.

```
model <-lm(growth ~ tannin)
```

Then, you can do a variety of things with the model. The most important, perhaps, is to see the details of the estimated effects, which you get from the summary function:

```
summary(model)
```

```
Coefficients:
             Estimate   Std. Error   t value   Pr(>|t|)
(Intercept)   11.7556       1.0408    11.295   9.54e-06  ***
tannin        -1.2167       0.2186    -5.565   0.000846  ***

Residual standard error: 1.693 on 7 degrees of freedom
Multiple R-Squared: 0.8157,    Adjusted R-squared: 0.7893
F-statistic: 30.97 on 1 and 7 DF, p-value: 0.0008461
```

This shows everything you need to know about the parameters and their standard errors (compare the values for s.e.$_a$ and s.e.$_b$ with those you calculated long-hand, above). If you want to see the Anova table rather than the parameter estimates, then the appropriate function is summary.aov

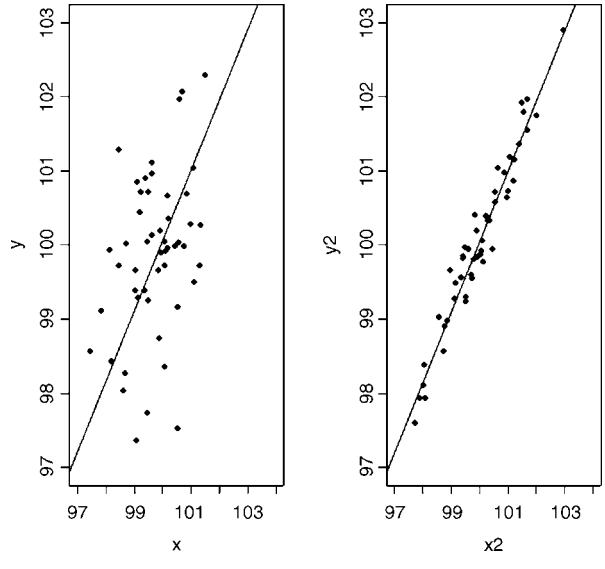
summary.aov(model)

```
             Df  Sum Sq  Mean Sq  F value    Pr(>F)
tannin        1  88.817   88.817   30.974  0.000846  ***
Residuals     7  20.072    2.867
```

This shows the error variance ($s^2 = 2.867$) along with *SSR* (88.817) and *SSE* (20.072), and the p value we just computed using 1-pf. Of the two sorts of summary table, the summary.lm is vastly the more informative, because it shows the effect sizes (in this case the slope of the graph) and their unreliability estimates (the standard error of the slope). Generally, you should resist the temptation to put Anova tables in your written work. The important information like the p-value and the error variance can be put in the text, or in figure legends, much more efficiently. Anova tables put far too much emphasis on hypothesis testing, and show nothing directly about effect sizes.

Measuring the Degree of Fit, r^2

There is a very important issue that remains to be considered. Two regression lines can have exactly the same slopes and intercepts, and yet be derived from completely different relationships as shown in the figures below. We need to be able to quantify the degree of

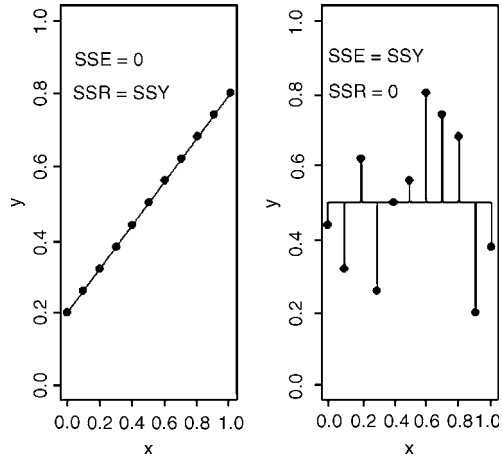


fit, which is low in the graph on the left and high in the right. In the limit, all the data points might fall exactly on the line. The degree of scatter in that case would be zero and the fit would be perfect (we might define a perfect fit as 1). At the other extreme, x might

explain none of the variation in y at all; in this case, fit would be zero and the degree of scatter would be 100%. Can we combine what we have learned about *SSY*, *SSR* and *SSE* into a measure of fit that has these properties? Our proposed metric is **the fraction of the total variation in *y* that is explained by the regression**. The total variation is *SSY* and the explained variation is *SSR*, so our measure – let's call it r^2 – is given by

$$r^2 = \frac{SSR}{SSY}.$$

This varies from 1, when the regression explains all of the variation in y ($SSR = SSY$), to 0 when the regression explains none of the variation in y ($SSE = SSY$).



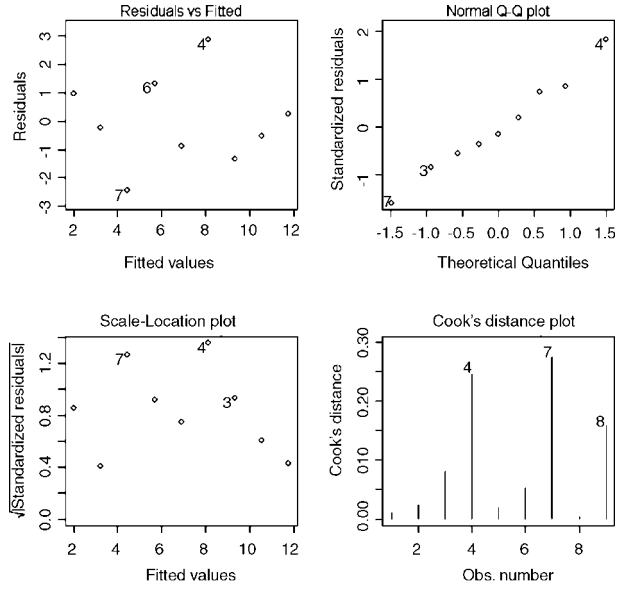
The formal name of this quantity is the coefficient of determination, but these days most people just refer to it as '*r squared*'. We have already met the square root of this quantity (r or ρ), as the correlation coefficient (p. 93).

Model Checking

The final thing you will want to do is to expose the model to critical appraisal. The assumptions we really want to be sure about are **constancy of variance** and **normality of errors**. The simples way to do this is with four model-checking plots:

```
plot(model)
```

The first graph (top left) shows residuals on the y axis against fitted values on the x axis. It takes experience to interpret these plots, but what you ***don't*** want to see is lots of structure or pattern in the plot. Ideally, as here, the points should look like the sky at night. It is a major problem if the scatter increases as the fitted values get bigger; this would show up like a wedge of cheese on its side (see p. 200). However, in our present case, everything is all right on the constancy of variance front. The next plot (top right) shows the Normal qqnorm plot (p. 64) which should be a straight line if the errors are normally distributed. Again, the present example looks fine. If the pattern were S-shaped



or banana-shaped, we would need to fit a different model to the data. The third plot (bottom left) is a repeat of the first, but on a different scale. It shows the square root of the standardized residuals (where all the values are positive) against the fitted values. If there was a problem, the points would be distributed inside a triangular shape, with the scatter of the residuals increasing as the fitted values increase; but there is no such pattern here, which is good. The fourth and final plot (lower right) shows Cook's distance for each of the observed values of the response variable (in the order in which they appear in the dataframe). The point of this plot is to highlight those y values that have the biggest effect on the parameter estimates (i.e. it shows **influence**; p. 123). You can see that point number 7 is the most influential; but which point is that? You can use 7 as a subscript (i.e. in square brackets) to find out:

```
tannin[7];growth[7]
```

```
[1] 6
[1] 2
```

The most influential point was the one with tannin = 6% and growth rate = 2. You might like to investigate how much this influential point (6,2) affected the parameter estimates and their standard errors. To do this, we repeat the statistical modelling but leave out the point in question, using subset like this (!= means 'not equal to'):

```
model2<-update(model,subset=(tannin != 6))
summary(model2)
```

```
Coefficients:
             Estimate   Std. Error   t value    Pr(>|t|)
(Intercept)   11.6892       0.8963    13.042   1.25e-05   ***
tannin        -1.1171       0.1956    -5.712    0.00125    **
```

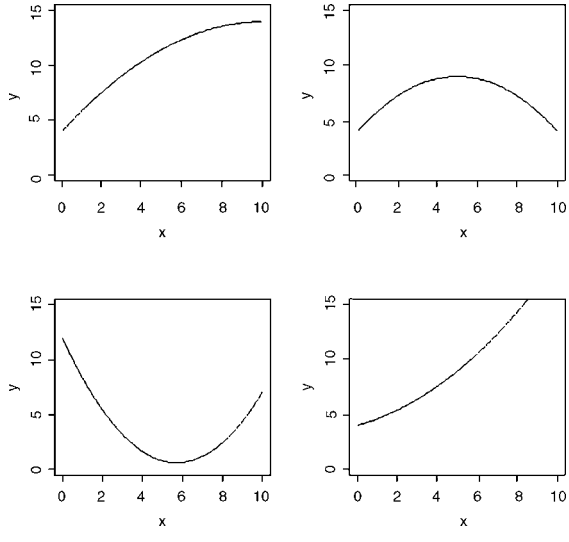
```
Residual standard error: 1.457 on 6 degrees of freedom
Multiple R-Squared: 0.8446,     Adjusted R-squared: 0.8188
F-statistic: 32.62 on 1 and 6 DF, p-value: 0.001247
```

First of all, notice that we have lost one degree of freedom, because there are now eight values of y rather than nine. The estimate of the slope has changed from -1.2167 to -1.1171 (a difference of about 9%) and the standard error of the slope has changed from 0.2186 to 0.1956 (a difference of about 12%). What you do in response to this information depends on the circumstances. Here, we would simply note that point (6,2) was influential and stick with our first model, using all the data. In other circumstances, a data point might be so influential that the structure of the model is changed completely by leaving it out. In that case, we might gather more data or, if the study was already finished, we might publish both results (with and without the influential point) so that the reader could make up their own mind about the interpretation. The important point is that we always do model-checking; the summary.lm(model) table is not the end of the process of regression analysis.

Polynomial Regression

The relationship between y and x often turns out not to be a straight line; but Occam's Razor requires that we fit a linear model unless a non-linear relationship is significantly better at describing the data. So this begs the question: how do we assess the significance of departures from linearity? One of the simplest ways is to use polynomial regression.

The idea of polynomial regression is straightforward. As before, we have just one continuous explanatory variable, x, but we can fit higher powers of x, like x squared and x cubed to the model in addition to x to explain curvature in the relationship between y and x. It is useful to experiment with the kinds of curves that can be generated with very simple models. Even if we restrict ourselves to the inclusion of a quadratic term, x^2, there are many curves we can describe, depending upon the signs of the linear and quadratic terms:



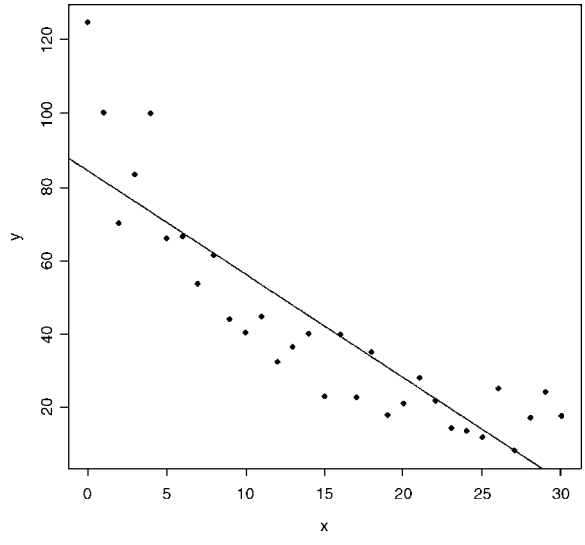
In the top left panel, there is a curve with positive but declining slope, with no hint of a hump ($y = 4 + 2x - 0.1x^2$). Top right shows a curve with a clear maximum ($y = 4 + 2x - 0.2x^2$), and bottom left shows a curve with a clear minimum ($y = 12 - 4x + 0.35x^2$). The bottom right curve shows a positive association between y and x with the slope increasing as x increases ($y = 4 + 0.5x + 0.1x^2$). So you can see that a simple quadratic model with three parameters (an intercept, a slope for x, and a slope for x^2) is capable of describing a wide range of functional relationships between y and x. It is very important to understand that the quadratic model **describes** the relationship between y and x; it does not pretend to **explain** the mechanistic (or causal) relationship between y and x.

We can see how polynomial regression works by analysing an example. Here are data showing the relationship between radioactive emissions (y) and time (x):

```
rm(x,y)
par(mfrow=c(1,1))
curve<-read.table("c:\\temp\\decay.txt",header=T)
attach(curve)
names(curve)
```

```
[1] "x" "y"
```

```
plot(x,y,pch=16)
abline(lm(y~x))
```



The fitted straight line (using abline) draws attention to the curvature. Most of the residuals for low and high values of x are positive, and most of the residuals for intermediate values of x are negative.

There are several ways of fitting polynomial regression models, but the simplest is to calculate a new explanatory variable which is x^2:

```
x2<-x^2
```

Now we do a multiple regression with two continuous explanatory variables: x and x^2

```
quadratic < -lm(y ~ x + x2)
summary(quadratic)
```

```
Coefficients:
              Estimate   Std. Error   t value   Pr(>|t|)
(Intercept)  106.38880     4.65627    22.849    < 2e-16  ***
x             -7.34485     0.71844   -10.223   5.90e-11  ***
x2             0.15059     0.02314     6.507   4.73e-07  ***

Residual standard error: 9.205 on 28 degrees of freedom
Multiple R-Squared: 0.908,    Adjusted R-squared: 0.9014
F-statistic: 138.1 on 2 and 28 DF,  p-value: 3.109e-015
```

The equation of the model is $y = 106.3888 - 7.34485x + 0.15059x^2$, and the standard errors of each of the three parameters appear in column 3. The key point here is that the quadratic term is highly significant ($p = 4.73\ 10^{-7}$), so there is strong evidence that the relationship is non-linear. Another way to come to the same conclusion is to use anova to compare linear and quadratic models, like this:

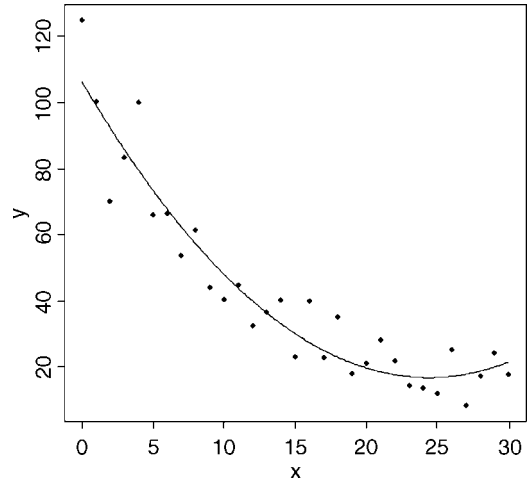
```
linear < -lm(y ~ x)
anova(quadratic,linear)
```

```
Analysis of Variance Table

Model 1: y ~ x + x2
Model 2: y ~ x
 Res.Df    RSS   Df  Sum of Sq      F     Pr(>F)
1    28 2372.6
2    29 5960.6  -1    -3588.1 42.344  4.727e-07   ***
```

Note that the significance of the difference is exactly the same as the significance of the quadratic term in the first model ($p = 4.73\ 10^{-7}$). So we can conclude that there is significant non-linearity, but is the quadratic model the best description of this non-linearity? We can get some impression by plotting the fitted values from the quadratic model on our initial scatterplot. We generate a series of x values between 0 and 30 and then use these in predict with the quadratic model to generate a smooth curve of 'y hat' against x:

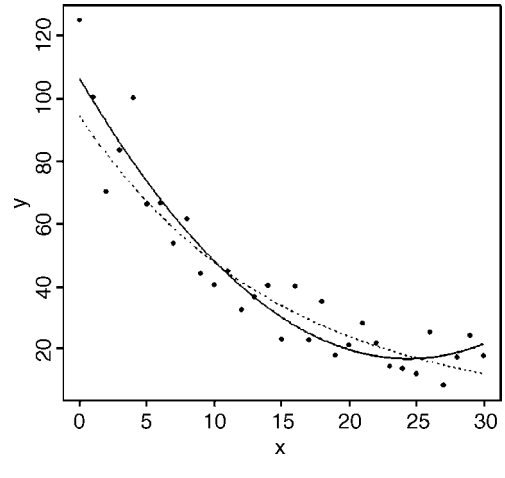
```
xv < -seq(0,30,0.1)
yv < -predict(quadratic,list(x = xv,x2 = xv^2))
lines(xv.yv)
```

The fit looks reasonably good, but model checking with plot(quadratic) suggests a degree of non-normality in the errors. The apparent increase in y at the highest values of x is also rather suspect (this problem often arises with polynomial models). Because the data relate to a decay process, it might be that an exponential function $y = ae^{-bx}$



describes the data better than a quadratic. This is a question of model comparison. We can use lm to fit an exponential curve if we make ln(y) rather than y the response variable:

```
exponential <- lm(log(y) ~ x)
yv2 <- exp(predict(exponential,list(x = xv)))
lines(xv,yv2,lty = 2)
```



Evidently, the exponential model (dotted line, lty = 2) provides a more intuitive description of the decay process, even though plot(exponential) draws attention to quite serious non-constancy of errors. This is a good example of how insights based on a mechanistic understanding of the process (e.g. a decay curve would probably not have a minimum)

need to be weighed against statistical rules of thumb (e.g. higher r^2 is better) in model comparisons.

Non-linear Regression

Sometimes we have a mechanistic model for the relationship between y and x, and we want to estimate the parameters and standard errors of the parameters of a specific non-linear equation from data. What we mean in this case by non-linear is not that the relationship is curved (it was curved in the case of polynomial regressions, but these were linear models), but that the relationship cannot be linearized by transformation of the response variable or the explanatory variable (or both). Here is an example, which shows jaw bone length as a function of age in deer. Theory indicates that the relationship is an 'asymptotic exponential' with three parameters:

$$y = a - be^{-cx}.$$

In R, the main difference between linear models and non-linear models is that we have to tell R the exact nature of the equation as part of the model formula when we use non-linear modelling. In place of lm we write nls (this stands for 'non-linear least squares'). Then we write y ~ a-b*exp(-c*x) to spell out the precise non-linear model we want R to fit to the data. The slightly tedious thing is that R requires us to specify initial guesses at the values of the parameters a, b and c (note, however, that some common non-linear models have 'self-starting' versions in R which bypass this step; see ? nls). Let's plot the data to work out sensible starting values. It always helps in cases like this to work out the equation's 'behaviour at the limits'. That is to say, the values of y when $x = 0$ and when $x =$ infinity. For $x = 0$, we have exp(−0) which is 1, and $1 \times b = b$ so $y = a - b$. For $x =$ infinity, we have exp(−infinity) which is 0, and $0 \times b = 0$ so $y = a$. That is to say, the asymptotic value of y is a, and the intercept is $a - b$.

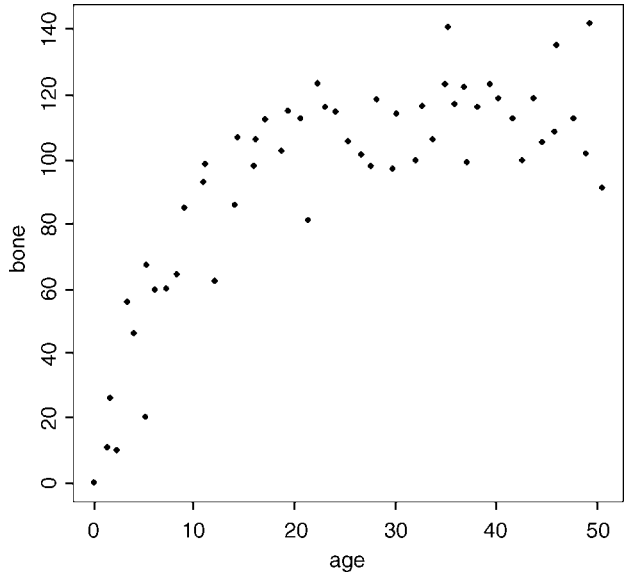
```
deer <-read.table("c:\\temp\\jaws.txt",header=T)
attach(deer)
names(deer)
```

```
[1] "age" "bone"
```

```
plot(age,bone,pch=16)
```

Inspection suggests that a reasonable estimate of the asymptote is $a \approx 120$ and intercept ≈ 10, so $b = 120 - 10 = 110$. Our guess of the value of c is slightly harder. Where the curve is rising most steeply, jaw length is about 40 where age is 5; rearranging the equation gives

$$c = -\frac{\log[(a-y)/b]}{x} = -\frac{\log[120-40)/110]}{5} = 0.06369075.$$



Now that we have the three parameter estimates, we can provide them to R as the starting conditions as part of the nls call like this: list(a = 120, b = 110, c = 0.064)

```
library(nls)
model <-nls(bone ~a-b*exp(-c*age),start=list(a=120,b=110,c= 0.064))
summary(model)

Formula: bone~a - b * exp(-c * age)

Parameters:
  Estimate     Std. Error   t value    Pr(>|t|)
a 115.2528         2.9139     39.55     < 2e-16   ***
b 118.6875         7.8925     15.04     < 2e-16   ***
c 0.1235           0.0171      7.22    2.44e-09   ***

Residual standard error: 13.21 on 51 degrees of freedom
```

All the parameters appear to be significant at $p < 0.001$, but beware. This does not necessarily mean that all the parameters need to be retained in the model. In this case, $a = 115.2528$ with s.e. $= 2.9139$ is clearly not significantly different from $b = 118.6875$ with s.e. $= 7.8925$ (they would need to differ by more than 2 s.e. to be significant). So we should try fitting the simpler two-parameter model

$$y = a(1 - e^{-cx})$$

```
model2 <- nls(bone ~ a*(1-exp(-c*age)),start = list(a = 120,c = 0.064))
anova(model,model2)
```

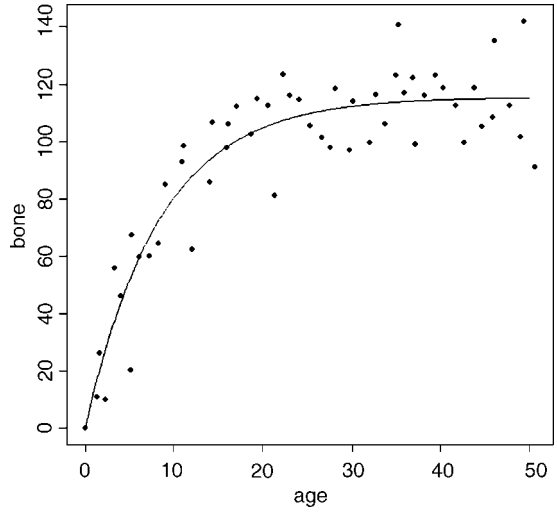
```
Analysis of Variance Table

Model 1: bone ~ a - b * exp(-c * age)
Model 2: bone ~ a * (1 - exp(-c * age))

    Res.Df    Res.Sum    Sq Df    Sum Sq    F value    Pr(>F)
1       51     8897.3
2       52     8929.1      -1     -31.8     0.1825     0.671
```

Model simplification was clearly justified ($p = 0.671$), so we accept the two-parameter version, model2, as our minimal adequate model. We finish by plotting the curve through the scatterplot. The age variable needs to go from 0 to 50:

```
av <- seq(0,50,0.1)
```

and we use predict with model2 to generate the predicted bone lengths:

```
bv <- predict(model2,list(age = av))
lines(av,bv)
```



The parameters of this curve are obtained from model2:

```
summary(model2)
```

```
Parameters:
   Estimate   Std. Error   t value    Pr(>|t|)
a 115.58056      2.84365   40.645      < 2e-16   ***
c   0.11882      0.01233    9.635    3.69e-13   ***

Residual standard error: 13.1 on 52 degrees of freedom
```

which we could write like this $y = 115.58(1 - e^{-0.1188x})$ or like this $y = 115.58\ [1 - \exp(-0.1188x)]$ according to taste or journal style. If you want to present the standard errors as well as the parameter estimates, you could write 'the model $y = a[1 - \exp(-bx)]$ had $a = 115.58 \pm 2.84$ (1 s.e.) and $b = 0.1188 \pm 0.0123$ (1 s.e., $n = 54$) and explained 84.6% of the total variation in bone length'. Note that because there are only two parameters in the minimal adequate model, we have called them a and b (rather than a and c as in the original formulation).

Testing for Humped Relationships

Proving the existence of humps in a relationship between y and x is controversial and can be difficult, but it is easy to appreciate the issues that are involved. For instance, is there good evidence for a hump in the following relationship?

```
smooth <-read.table("c:\\temp\\smoothing.txt",header=T)
attach(smooth)
names(smooth)
```

```
[1] "x" "y"
```

```
par(mfrow=c(1,2))
plot(x,y)
abline(lm(y~x))
```

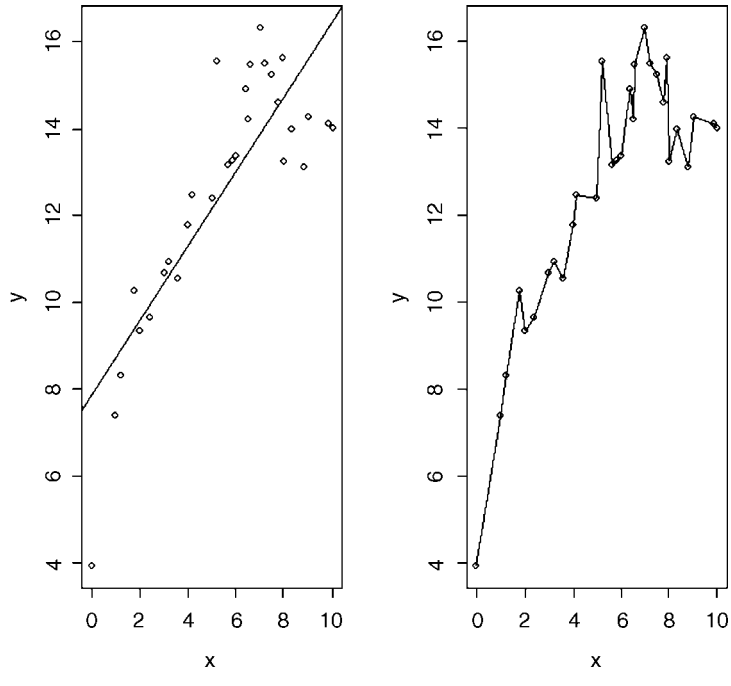
This is the most parsimonious relationship between y and x with just two parameters; the intercept and the slope of the linear regression. At the other extreme, we could produce a model which explained *all* of the variation in y. This is what it would look like:

```
sequence <-order(x)
plot(x,y)
lines(x[sequence],y[sequence])
```

The model has as many parameters as there are data points, hence it has no degrees of freedom, and exhibits no explanatory power. What we need is a model of intermediate complexity that optimizes the trade-off between the number of parameters and explanatory power. Incidentally, look what a mess you get if you try `lines(x,y)`; this illustrates the advantage of using ordered subscripts. You can carry out a quadratic polynomial regression on these data as an exercise, but does the existence of a significant quadratic term in the model prove the existence of a significant hump in the relationship?

Generalized Additive Models (gams)

Sometimes we can see that the relationship between y and x is non-linear but we don't have any theory or any mechanistic model to suggest a particular functional form (mathematical equation) to describe the relationship. In such circumstances, gams are



particularly useful, because they fit non-parametric smoothers to the data without requiring us to specify any particular mathematical model to describe the non-linearity. This will become clear with an example.

```
rm(x,y)
library(mgcv)
hump<-read.table("c:\\temp\\hump.txt",header=T)
attach(hump)
names(hump)
```

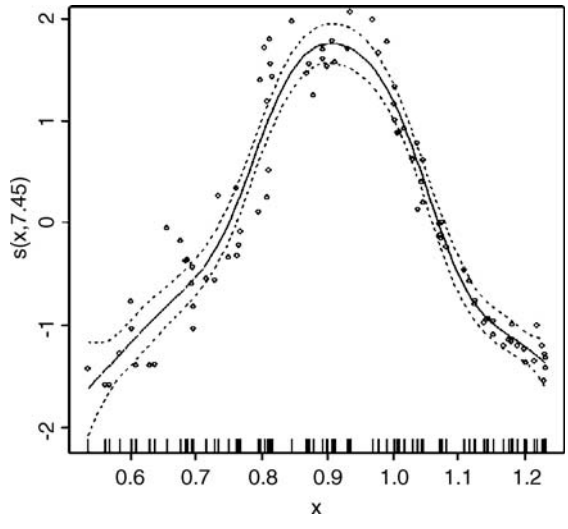
```
[1] "y" "x"
```

We start by fitting the generalized additive model as a smoothed function of *x*, s(x):

```
model<-gam(y~s(x))
```

then we plot the model, and overlay the scattergraph of data points

```
plot(model)
points(x,y-mean(y))
```



Model summary is obtained in the usual way:

```
summary(model)

Family: gaussian
Link function: identity

Formula:
y ~ s(x)

Parametric coefficients:
           Estimate   std. err.   t ratio     Pr(>|t|)
constant      1.9574     0.03446      56.8   < 2.22e-16

Approximate significance of smooth terms:
         edf    chi.sq       p-value
s(x)   7.452    982.38    < 2.22e-16

Adjusted r-sq. = 0.919    GCV score = 0.1156
Scale estimate = 0.1045           n = 88
```

This shows that the humped relationship between y and x is highly significant (see the p-value of the smooth term `s`(x) with an r^2 of 0.919). Note that because of the strong hump in the relationship, a linear model `lm(y~x)` indicates no significant relationship between the two variables ($p = 0.346$). This is an object lesson in always plotting the data before you come to conclusions from the statistical analysis; in this case, if you had started with a linear model you would have thrown out the baby with the bathwater by concluding that nothing was happening. In fact, something very significant is happening but it is producing a humped, rather than a trended relationship.