Chapter 2

Estimation

2.1 Example

Let's start with an example. Suppose that Y is the fuel consumption of a particular model of car in m.p.g. Suppose that the predictors are

1. X_1 — the weight of the car
2. X_2 — the horse power
3. X_3 — the no. of cylinders.

X_3 is discrete but that's OK. Using country of origin, say, as a predictor would not be possible within the current development (we will see how to do this later in the course). Typically the data will be available in the form of an array like this

$$\begin{array}{cccc} y_1 & x_{11} & x_{12} & x_{13} \\ y_2 & x_{21} & x_{22} & x_{23} \\ \cdots & & \cdots & \\ y_n & x_{n1} & x_{n2} & x_{n3} \end{array}$$

where n is the number of observations or *cases* in the dataset.

2.2 Linear Model

One very general form for the model would be

$$Y = f(X_1, X_2, X_3) + \varepsilon$$

where f is some unknown function and ε is the error in this representation which is additive in this instance. Since we usually don't have enough data to try to estimate f directly, we usually have to assume that it has some more restricted form, perhaps linear as in

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \varepsilon$$

where β_i, $i = 0, 1, 2, 3$ are unknown *parameters*. β_0 is called the *intercept* term. Thus the problem is reduced to the estimation of four values rather than the complicated infinite dimensional f.

In a linear model the *parameters enter linearly* — the predictors do not have to be linear. For example

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 \log X_2 + \varepsilon$$

is linear but

$$Y = \beta_0 + \beta_1 X_1^{\beta_2} + \varepsilon$$

is not. Some relationships can be transformed to linearity — for example $y = \beta_0 x_1^{\beta} \varepsilon$ can be linearized by taking logs. Linear models seem rather restrictive but because the predictors can transformed and combined in any way, they are actually very flexible. Truly non-linear models are rarely absolutely necessary and most often arise from a theory about the relationships between the variables rather than an empirical investigation.

2.3 Matrix Representation

Given the actual data, we may write

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \beta_3 x_{3i} + \varepsilon_i \quad i = 1, \ldots, n$$

but the use of subscripts becomes inconvenient and conceptually obscure. We will find it simpler both notationally and theoretically to use a matrix/vector representation. The regression equation is written as

$$y = X\beta + \varepsilon$$

where $y = (y_1 \ldots y_n)^T$, $\varepsilon = (\varepsilon_1 \ldots \varepsilon_n)^T$, $\beta = (\beta_0 \ldots \beta_3)^T$ and

$$X = \begin{pmatrix} 1 & x_{11} & x_{12} & x_{13} \\ 1 & x_{21} & x_{22} & x_{23} \\ \ldots & & \ldots & \\ 1 & x_{n1} & x_{n2} & x_{n3} \end{pmatrix}$$

The column of ones incorporates the intercept term. A couple of examples of using this notation are the simple no predictor, mean only model $y = \mu + \varepsilon$

$$\begin{pmatrix} y_1 \\ \ldots \\ y_n \end{pmatrix} = \begin{pmatrix} 1 \\ \ldots \\ 1 \end{pmatrix} \mu + \begin{pmatrix} \varepsilon_1 \\ \ldots \\ \varepsilon_n \end{pmatrix}$$

We can assume that $E\varepsilon = 0$ since if this were not so, we could simply absorb the non-zero expectation for the error into the mean μ to get a zero expectation. For the two sample problem with a treatment group having the response $y_1, \ldots, y_m$ with mean μ_y and control group having response $z_1, \ldots, z_n$ with mean μ_z we have

$$\begin{pmatrix} y_1 \\ \ldots \\ y_m \\ z_1 \\ \ldots \\ z_n \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ \ldots & \\ 1 & 0 \\ 0 & 1 \\ . & . \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \mu_y \\ \mu_z \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \ldots \\ \ldots \\ \ldots \\ \varepsilon_{m+n} \end{pmatrix}$$

2.4 Estimating β

We have the regression equation $y = X\beta + \varepsilon$ - what estimate of β would best separate the systematic component $X\beta$ from the random component ε. Geometrically speaking, $y \in \mathbb{R}^n$ while $\beta \in \mathbb{R}^p$ where p is the number of parameters (if we include the intercept then p is the number of predictors plus one).

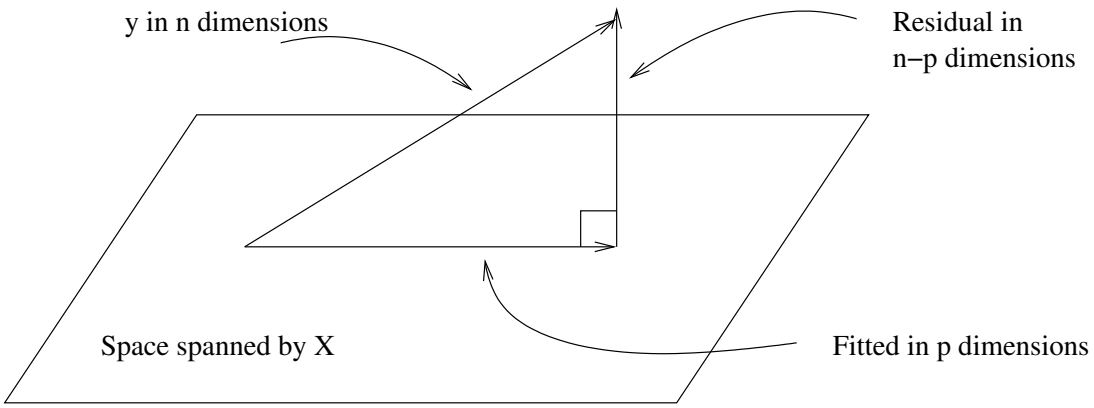

Figure 2.1: Geometric representation of the estimation β. The data vector Y is projected orthogonally onto the model space spanned by X. The fit is represented by projection $\hat{y} = X\hat{\beta}$ with the difference between the fit and the data represented by the residual vector $\hat{\varepsilon}$.

The problem is to find β such that $X\beta$ is close to Y. The best choice of $\hat{\beta}$ is apparent in the geometrical representation shown in Figure 2.4.

$\hat{\beta}$ is in some sense the best estimate of β within the model space. The response predicted by the model is $\hat{y} = X\hat{\beta}$ or Hy where H is an orthogonal projection matrix. The difference between the actual response and the predicted response is denoted by $\hat{\varepsilon}$ — the residuals.

The conceptual purpose of the model is to represent, as accurately as possible, something complex — y which is n-dimensional — in terms of something much simpler — the model which is p-dimensional. Thus if our model is successful, the structure in the data should be captured in those p dimensions, leaving just random variation in the residuals which lie in an $n-p$ dimensional space. We have

$$\begin{aligned} \text{Data} &= \text{Systematic Structure} + \text{Random Variation} \\ n \text{ dimensions} &= p \text{ dimensions} + (n-p) \text{ dimensions} \end{aligned}$$

2.5 Least squares estimation

The estimation of β can be considered from a non-geometric point of view. We might define the best estimate of β as that which minimizes the sum of the squared errors, $\varepsilon^T\varepsilon$. That is to say that the least squares estimate of β, called $\hat{\beta}$ minimizes

$$\sum \varepsilon_i^2 = \varepsilon^T\varepsilon = (y - X\beta)^T(y - X\beta)$$

Expanding this out, we get

$$y^Ty - 2\beta X^Ty + \beta^T X^T X\beta$$

Differentiating with respect to β and setting to zero, we find that $\hat{\beta}$ satisfies

$$X^TX\hat{\beta} = X^Ty$$

These are called the normal equations. We can derive the same result using the geometric approach. Now provided X^TX is invertible

$$\begin{aligned} \hat{\beta} &= (X^TX)^{-1}X^Ty \\ X\hat{\beta} &= X(X^TX)^{-1}X^Ty \\ &= Hy \end{aligned}$$

$H = X(X^TX)^{-1}X^T$ is called the "hat-matrix" and is the orthogonal projection of y onto the space spanned by X. H is useful for theoretical manipulations but you usually don't want to compute it explicitly as it is an $n \times n$ matrix.

- Predicted values: $\hat{y} = Hy = X\hat{\beta}$.
- Residuals: $\hat{\varepsilon} = y - X\hat{\beta} = y - \hat{y} = (I - H)y$
- Residual sum of squares: $\hat{\varepsilon}^T\hat{\varepsilon} = y^T(I-H)(I-H)y = y^T(I-H)y$

Later we will show that the least squares estimate is the best possible estimate of β when the errors ε are uncorrelated and have equal variance - i.e. var $\varepsilon = \sigma^2 I$.

2.6 Examples of calculating $\hat{\beta}$

1. When $y = \mu + \varepsilon$, $X = \mathbf{1}$ and $\beta = \mu$ so $X^TX = \mathbf{1}^T\mathbf{1} = n$ so
$$\hat{\beta} = (X^TX)^{-1}X^Ty = \frac{1}{n}\mathbf{1}^Ty = \bar{y}$$
2. Simple linear regression (one predictor)
$$y_i = \alpha + \beta x_i + \varepsilon_i$$
$$\begin{pmatrix} y_1 \\ \dots \\ y_n \end{pmatrix} = \begin{pmatrix} 1 & x_1 \\ \dots & \\ 1 & x_n \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \dots \\ \varepsilon_n \end{pmatrix}$$
We can now apply the formula but a simpler approach is to rewrite the equation as
$$y_i = \overbrace{\alpha + \beta\bar{x}}^{\alpha'} + \beta(x_i - \bar{x}) + \varepsilon_i$$
so now
$$X = \begin{pmatrix} 1 & x_1 - \bar{x} \\ \dots & \\ 1 & x_n - \bar{x} \end{pmatrix} \qquad X^TX = \begin{pmatrix} n & 0 \\ 0 & \sum_{i=1}^n (x_i - \bar{x})^2 \end{pmatrix}$$

Now work through the rest of the calculation to reconstruct the familiar estimates, i.e.
$$\hat{\beta} = \frac{\sum(x_i - \bar{x})y_i}{\sum(x_i - \bar{x})^2}$$

In higher dimensions, it is usually not possible to find such explicit formulae for the parameter estimates unless X^TX happens to be a simple form.

2.7 Why is $\hat{\beta}$ a good estimate?

1. It results from an orthogonal projection onto the model space. It makes sense geometrically.
2. If the errors are independent and identically normally distributed, it is the maximum likelihood estimator. Loosely put, the maximum likelihood estimate is the value of β that maximizes the probability of the data that was observed.
3. The Gauss-Markov theorem states that it is best linear unbiased estimate. (BLUE).

2.8 Gauss-Markov Theorem

First we need to understand the concept of an *estimable function*. A linear combination of the parameters $\psi = c^T\beta$ is estimable if and only if there exists a linear combination $a^T y$ such that

$$Ea^T y = c^T\beta \qquad \forall\beta$$

Estimable functions include predictions of future observations which explains why they are worth considering. If X is of full rank (which it usually is for observational data), then all linear combinations are estimable.

Gauss-Markov theorem

Suppose $E\varepsilon = 0$ and var $\varepsilon = \sigma^2 I$. Suppose also that the structural part of the model, $EY = X\beta$ is correct. Let $\psi = c^T\beta$ be an estimable function, then in the class of all unbiased linear estimates of ψ, $\hat{\psi} = c^T\hat{\beta}$ has the minimum variance and is unique.

Proof:

We start with a preliminary calculation:

Suppose $a^T y$ is some unbiased estimate of $c^T\beta$ so that

$$\begin{aligned} Ea^T y &= c^T\beta \qquad \forall\beta \\ a^T X\beta &= c^T\beta \qquad \forall\beta \end{aligned}$$

which means that $a^T X = c^T$. This implies that c must be in the range space of X^T which in turn implies that c is also in the range space of $X^T X$ which means there exists a λ such that

$$\begin{aligned} c &= X^T X\lambda \\ c^T\hat{\beta} &= \lambda^T X^T X\hat{\beta} = \lambda^T X^T y \end{aligned}$$

Now we can show that the least squares estimator has the minimum variance — pick an arbitrary estimable function $a^T y$ and compute its variance:

$$\begin{aligned} \text{var}\,(a^T y) &= \text{var}\,(a^T y - c^T\hat{\beta} + c^T\hat{\beta}) \\ &= \text{var}\,(a^T y - \lambda^T X^T y + c^T\hat{\beta}) \\ &= \text{var}\,(a^T y - \lambda^T X^T y) + \text{var}\,(c^T\hat{\beta}) + 2\text{cov}(a^T y - \lambda^T X^T y, \lambda^T X^T y) \end{aligned}$$

but

$$\begin{aligned} \text{cov}(a^T y - \lambda^T X^T y, \lambda^T X^T y) &= (a^T - \lambda^T X^T)\sigma^2 I X\lambda \\ &= (a^T X - \lambda X^T X)\sigma^2 I\lambda \\ &= (c^T - c^T)\sigma^2 I\lambda = 0 \end{aligned}$$

so

$$\text{var}\,(a^T y) = \text{var}\,(a^T y - \lambda^T X^T y) + \text{var}\,(c^T\hat{\beta})$$

Now since variances cannot be negative, we see that

$$\text{var}\,(a^T y) \geq \text{var}\,(c^T\hat{\beta})$$

In other words $c^T\hat{\beta}$ has minimum variance. It now remains to show that it is unique. There will be equality in above relation if var $(a^T y - \lambda^T X^T y) = 0$ which would require that $a^T - \lambda^T X^T = 0$ which means that $a^T y = \lambda^T X^T y = c^T\hat{\beta}$ so equality occurs only if $a^T y = c^T\hat{\beta}$ so the estimator is unique.

Implications

The Gauss-Markov theorem shows that the least squares estimate $\hat{\beta}$ is a good choice, but if the errors are correlated or have unequal variance, there will be better estimators. Even if the errors behave but are non-normal then non-linear or biased estimates may work better in some sense. So this theorem does not tell one to use least squares all the time, it just strongly suggests it unless there is some strong reason to do otherwise.

Situations where estimators other than ordinary least squares should be considered are

1. When the errors are correlated or have unequal variance, generalized least squares should be used.
2. When the error distribution is long-tailed, then robust estimates might be used. Robust estimates are typically not linear in y.
3. When the predictors are highly correlated (collinear), then biased estimators such as ridge regression might be preferable.

2.9 Mean and Variance of $\hat{\beta}$

Now $\hat{\beta} = (X^T X)^{-1} X^T y$ so

- Mean $E\hat{\beta} = (X^T X)^{-1} X^T X\beta = \beta$ (unbiased)
- var $\hat{\beta} = (X^T X)^{-1} X^T \sigma^2 I X (X^T X)^{-1} = (X^T X)^{-1}\sigma^2$

Note that since $\hat{\beta}$ is a vector, $(X^T X)^{-1}\sigma^2$ is a variance-covariance matrix. Sometimes you want the standard error for a particular component which can be picked out as in $se(\hat{\beta}_i) = \sqrt{(X^T X)^{-1}_{ii}}\hat{\sigma}$.

2.10 Estimating σ^2

Recall that the residual sum of squares was $\hat{\varepsilon}^T\hat{\varepsilon} = y^T(I-H)y$. Now after some calculation, one can show that $E\hat{\varepsilon}^T\hat{\varepsilon} = \sigma^2(n-p)$ which shows that

$$\hat{\sigma}^2 = \frac{\hat{\varepsilon}^T\hat{\varepsilon}}{n-p}$$

is an unbiased estimate of σ^2. $n-p$ is the *degrees of freedom* of the model. Actually a theorem parallel to the Gauss-Markov theorem shows that it has the minimum variance among all quadratic unbiased estimators of σ^2.

2.11 Goodness of Fit

How well does the model fit the data? One measure is R^2, the so-called *coefficient of determination* or *percentage of variance explained*

$$R^2 = 1 - \frac{\sum(\hat{y}_i - y_i)^2}{\sum(y_i - \bar{y})^2} = 1 - \frac{\text{RSS}}{\text{Total SS(corrected for mean)}}$$

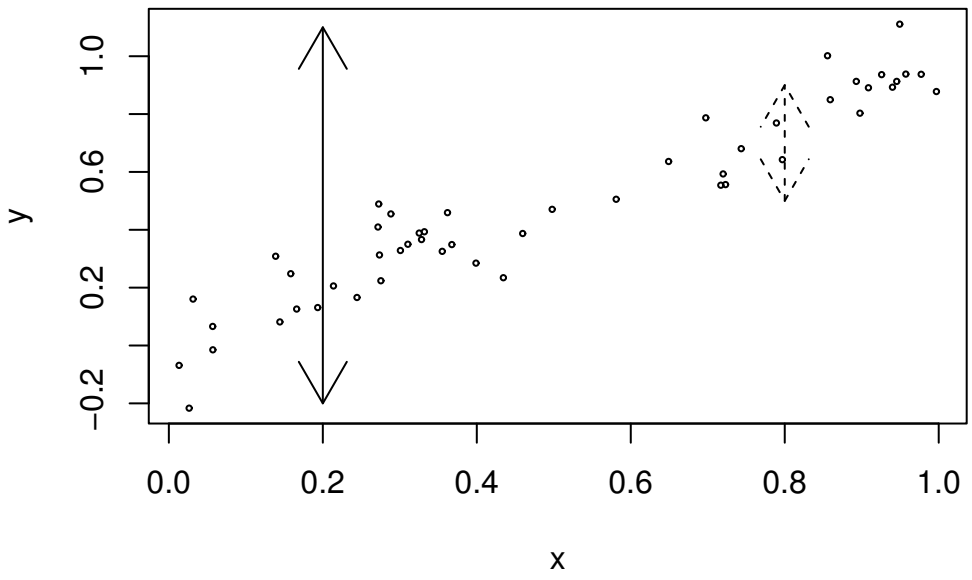

Figure 2.2: Variation in the response y when x is known is denoted by dotted arrows while variation in y when x is unknown is shown with the solid arrows

The range is $0 \leq R^2 \leq 1$ - values closer to 1 indicating better fits. For simple linear regression $R^2 = r^2$ where r is the correlation between x and y. An equivalent definition is

$$R^2 = \frac{\sum(\hat{y}_i - \bar{y})^2}{\sum(y_i - \bar{y})^2}$$

The graphical intuition behind R^2 is shown in Figure 2.2. Suppose you want to predict y. If you don't know x, then your best prediction is $\bar{y}$ but the variability in this prediction is high. If you do know x, then your prediction will be given by the regression fit. This prediction will be less variable provided there is some relationship between x and y. R^2 is one minus the ratio of the sum of squares for these two predictions. Thus for perfect predictions the ratio will be zero and R^2 will be one.

Warning: R^2 as defined here doesn't make any sense if you do not have an intercept in your model. This is because the denominator in the definition of R^2 has a null model with an intercept in mind when the sum of squares is calculated. Alternative definitions of R^2 are possible when there is no intercept but the same graphical intuition is not available and the R^2's obtained should not be compared to those for models with an intercept. Beware of high R^2's reported from models without an intercept.

What is a good value of R^2? It depends on the area of application. In the biological and social sciences, variables tend to be more weakly correlated and there is a lot of noise. We'd expect lower values for R^2 in these areas — a value of 0.6 might be considered good. In physics and engineering, where most data comes from closely controlled experiments, we expect to get much higher R^2's and a value of 0.6 would be considered low. Of course, I generalize excessively here so some experience with the particular area is necessary for you to judge your R^2's well.

An alternative measure of fit is $\hat{\sigma}$. This quantity is directly related to the standard errors of estimates of β and predictions. The advantage is that $\hat{\sigma}$ is measured in the units of the response and so may be directly interpreted in the context of the particular dataset. This may also be a disadvantage in that one

must understand whether the practical significance of this measure whereas R^2, being unitless, is easy to understand.

2.12 Example

Now let's look at an example concerning the number of species of tortoise on the various Galapagos Islands. There are 30 cases (Islands) and 7 variables in the dataset. We start by reading the data into R and examining it

```
> data(gala)
> gala
             Species Endemics    Area Elevation Nearest Scruz Adjacent
      Baltra      58        23   25.09       346     0.6   0.6     1.84
   Bartolome      31        21    1.24       109     0.6  26.3   572.33

--- cases deleted ---

     Tortuga      16         8    1.24       186     6.8  50.9    17.95
        Wolf      21        12    2.85       253    34.1 254.7     2.33
```

The variables are

Species The number of species of tortoise found on the island

Endemics The number of endemic species

Elevation The highest elevation of the island (m)

Nearest The distance from the nearest island (km)

Scruz The distance from Santa Cruz island (km)

Adjacent The area of the adjacent island (km^2)

The data were presented by Johnson and Raven (1973) and also appear in Weisberg (1985). I have filled in some missing values for simplicity (see Chapter 14 for how this can be done). Fitting a linear model in R is done using the `lm()` command. Notice the syntax for specifying the predictors in the model. This is the so-called *Wilkinson-Rogers* notation. In this case, since all the variables are in the gala data frame, we must use the `data=` argument:

```
> gfit <- lm(Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
     data=gala)
> summary(gfit)
Call:
lm(formula = Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
    data = gala)

Residuals:
    Min      1Q  Median      3Q     Max
-111.68  -34.90   -7.86   33.46  182.58
```

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  7.06822   19.15420    0.37   0.7154
Area        -0.02394    0.02242   -1.07   0.2963
Elevation    0.31946    0.05366    5.95  3.8e-06
Nearest      0.00914    1.05414    0.01   0.9932
Scruz       -0.24052    0.21540   -1.12   0.2752
Adjacent    -0.07480    0.01770   -4.23   0.0003

Residual standard error: 61 on 24 degrees of freedom
Multiple R-Squared: 0.766,      Adjusted R-squared: 0.717
F-statistic: 15.7 on 5 and 24 degrees of freedom,       p-value: 6.84e-07
```

We can identify several useful quantities in this output. Other statistical packages tend to produce output quite similar to this. One useful feature of R is that it is possible to directly calculate quantities of interest. Of course, it is not necessary here because the `lm()` function does the job but it is very useful when the statistic you want is not part of the pre-packaged functions.

First we make the X-matrix

```
> x <- cbind(1,gala[,-c(1,2)])
```

and here's the response y:

```
> y <- gala$Species
```

Now let's construct X^TX: `t()` does transpose and `%*%` does matrix multiplication:

```
> t(x) %*% x
Error: %*% requires numeric matrix/vector arguments
```

Gives a somewhat cryptic error. The problem is that matrix arithmetic can only be done with numeric values but `x` here derives from the data frame type. Data frames are allowed to contain character variables which would disallow matrix arithmetic. We need to force x into the matrix form:

```
> x <- as.matrix(x)
> t(x) %*% x
```

Inverses can be taken using the `solve()` command:

```
> xtxi <- solve(t(x) %*% x)
> xtxi
```

A somewhat more direct way to get $(X^TX)^{-1}$ is as follows:

```
> gfit <- lm(Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
                       data=gala)
> gs <- summary(gfit)
> gs$cov.unscaled
```

The `names()` command is the way to see the components of an Splus object - you can see that there are other useful quantities that are directly available:

```
> names(gs)
> names(gfit)
```

In particular, the fitted (or predicted) values and residuals are

```
> gfit$fit
> gfit$res
```

We can get $\hat{\beta}$ directly:

```
> xtxi %*% t(x) %*% y
          [,1]
[1,]  7.068221
[2,] -0.023938
[3,]  0.319465
[4,]  0.009144
[5,] -0.240524
[6,] -0.074805
```

or in a computationally efficient and stable manner:

```
> solve(t(x) %*% x, t(x) %*% y)
          [,1]
[1,]  7.068221
[2,] -0.023938
[3,]  0.319465
[4,]  0.009144
[5,] -0.240524
[6,] -0.074805
```

We can estimate σ using the estimator in the text:

```
> sqrt(sum(gfit$res^2)/(30-6))
[1] 60.975
```

Compare this to the results above.

We may also obtain the standard errors for the coefficients. Also `diag()` returns the diagonal of a matrix):

```
> sqrt(diag(xtxi))*60.975
[1] 19.154139  0.022422  0.053663  1.054133  0.215402  0.017700
```

Finally we may compute R^2:

```
> 1-sum(gfit$res^2)/sum((y-mean(y))^2)
[1] 0.76585
```