Chapter 7

Multiple regression and model selection

7.1 Objectives

1. To state, fit, check, interpret, and compare regression models that have multiple explanatory variables (predictors).
2. To use R in the fitting, checking and comparison of models

7.2 Multiple Regression

In linear regression, we can have more than one predictor variable. This is known as multiple regression. The model is

$$Y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \ldots + \epsilon_i; \quad \epsilon_i \sim \mathrm{N}(0, \sigma)$$

We can write the model without the random component in terms of the expected value (or mean) of the random variable Y:

$$\mu_Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots$$

Example (Soil-sediment)
Soil and sediment absorption, the extent to which chemicals collect in a condensed form on the surface, is an important characteristic because it influences the effectiveness of pesticides and various agricultural chemicals. In an experiment, the following variables were measured on soil samples:

y = phosphate absorption index
x_1 = amount of extractable iron
x_2 = amount of extractable aluminium

y	4	18	14	18	26	26	21	30	28	36	65	62	40
x_1	61	175	111	124	130	173	169	169	160	244	257	333	199
x_2	13	21	24	23	64	38	33	61	39	71	112	88	54

```
> soil <-
+   data.frame(pai = c(4,18,14,18,26,26,21,30,28,36,65,62,40),
+              iron = c(61,175,111,124,130,173,169,169,160,244,257,333,199),
+              aluminium = c(13,21,24,23,64,38,33,61,39,71,112,88,54))
```

Problem: Fit a model of the form $\mu_Y = \alpha + \beta_1 x_1 + \beta_2 x_2$ and use it to obtain a predicted value for the phosphate adsorption index when $x_1 = 150$ and $x_2 = 40$.

Solution:

```
> soil.lm.1 <- lm(pai ~ iron + aluminium, data = soil)
> summary(soil.lm.1)
```

```
Call:
lm(formula = pai ~ iron + aluminium, data = soil)

Residuals:
    Min      1Q  Median      3Q     Max
-8.9352 -2.2182  0.4613  3.3448  6.0708

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -7.35066    3.48467  -2.109 0.061101 .
iron         0.11273    0.02969   3.797 0.003504 **
aluminium    0.34900    0.07131   4.894 0.000628 ***
---
Signif. codes:  0 ,***, 0.001 ,**, 0.01 ,*, 0.05 ,., 0.1 , , 1

Residual standard error: 4.379 on 10 degrees of freedom
Multiple R-squared: 0.9485,        Adjusted R-squared: 0.9382
F-statistic: 92.03 on 2 and 10 DF,  p-value: 3.634e-07
```

```
> predict(soil.lm.1,
+         newdata = data.frame(iron=150, aluminium=40),
+         interval = "prediction")

       fit      lwr      upr
1 23.51929 13.33372 33.70486
```

The parameter estimates have the usual interpretation, but with an additional proviso: that the other explanatory variables remain constant. For example, the estimate for iron, 0.113, means that phosphate absorption index is predicted to increase by 0.113 for each additional unit of extractable iron, *provided that the amount of extractable aluminium remains constant.*

In the model summary above, the P-values for the two predictors are very small, implying that both extractable iron and extractable aluminium contribute significantly to the phosphate absorption index. Fit each predictor on its own, and think about how beneficial it is having two predictors rather than one.

7.2.1 Polynomial regression

We can also include quadratic terms such as x_1^2 and x_2^2 to model a curved relationship between Y and an explanatory variable. In particular, when there is only one explanatory variable, a model of the form

$$\mu_Y = \beta_0 + \beta_1 x + \beta_2 x^2 + \cdots + \beta_k x^k$$

is called a **polynomial regression**.

Example (Nursery plants)

From Lab 6, Section 6.5—have a quick look at the plot.

```
> plants <- data.frame(age = rep(2:7, rep(4, 6)),
+                      height = c(
+                        5.6, 4.8, 5.3, 5.7, 6.2, 5.9, 6.4, 6.1,
+                        6.2, 6.7, 6.4, 6.7, 7.1, 7.3, 6.9, 6.9,
+                        7.2, 7.5, 7.8, 7.8, 8.9, 9.2, 8.5, 8.7))
```

We previously fitted a straight line, but the plot suggests a cubic relationship. Fit a cubic regression $\mu_Y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3$ to the data and plot the residuals. Does it look better than the simple linear regression model?

Solution:

```
> plants.lm.1 <- lm(height ~ age, data = plants)
> plants.lm.3 <- lm(height ~ age + I(age^2) + I(age^3), data = plants)
```

The `I()` function enables us to include the squared and cubic terms without having to calculate them separately.

```
> summary(plants.lm.3)

Call:
lm(formula = height ~ age + I(age^2) + I(age^3), data = plants)

Residuals:
     Min       1Q   Median       3Q      Max
-0.55377 -0.13338  0.02599  0.17758  0.38591

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.67381    1.22876   1.362  0.18828
age          2.84203    0.96302   2.951  0.00790 **
I(age^2)    -0.59732    0.22925  -2.606  0.01692 *
I(age^3)     0.04815    0.01690   2.849  0.00992 **
---
Signif. codes:  0 ,***, 0.001 ,**, 0.01 ,*, 0.05 ,., 0.1 , , 1

Residual standard error: 0.2721 on 20 degrees of freedom
Multiple R-squared: 0.9517,        Adjusted R-squared: 0.9445
F-statistic: 131.4 on 3 and 20 DF,  p-value: 2.499e-13
```

We see that the cubic term is highly significant, and R^2 has improved from 0.92 to 0.95— not a large improvement in absolute terms, but a substantial proportion of the remaining unexplained variation. Let's check the diagnostics (Figure 7.1). This is a good looking model!

```
> par(mfrow = c(2, 2), las = 1)
> plot(plants.lm.3)
```

We can make a graphical comparison of the models as follows (Figure 7.2).

```
> plot(height ~ age, data = plants, xlab="Age (y)", ylab="Height (ft)")
> curve(predict(plants.lm.1,
+               newdata=data.frame(age = x)), add=TRUE, col="red")
> curve(predict(plants.lm.3,
+               newdata=data.frame(age = x)), add=TRUE, col="blue")
```

7.3 Model Selection

7.3.1 Introduction

There is usually more than one model we can fit to a data set. It is important to be able to compare different models and decide which one is most appropriate. Some data sets may many explanatory variables, some of which may not make a significant contribution. To decide which variables to include or exclude is part of the model selection process.

There is usually a trade-off to be made between simplicity and goodness of fit. A more complex model will always give a better fit (in the sense of a smaller residual sum of squares and a larger R^2). But if we can get a fit that is almost as good with a simpler model, then we would usually prefer the simpler one. Our guiding principle then is that unless the more complicated model significantly reduces the residual SS, we will stick with the simpler model. This is called the **principle of parsimony**.

7.3.2 Extra information about F-tests

Many statistical analyses can be considered as essentially comparing two nested models. We usually make the following assumptions:

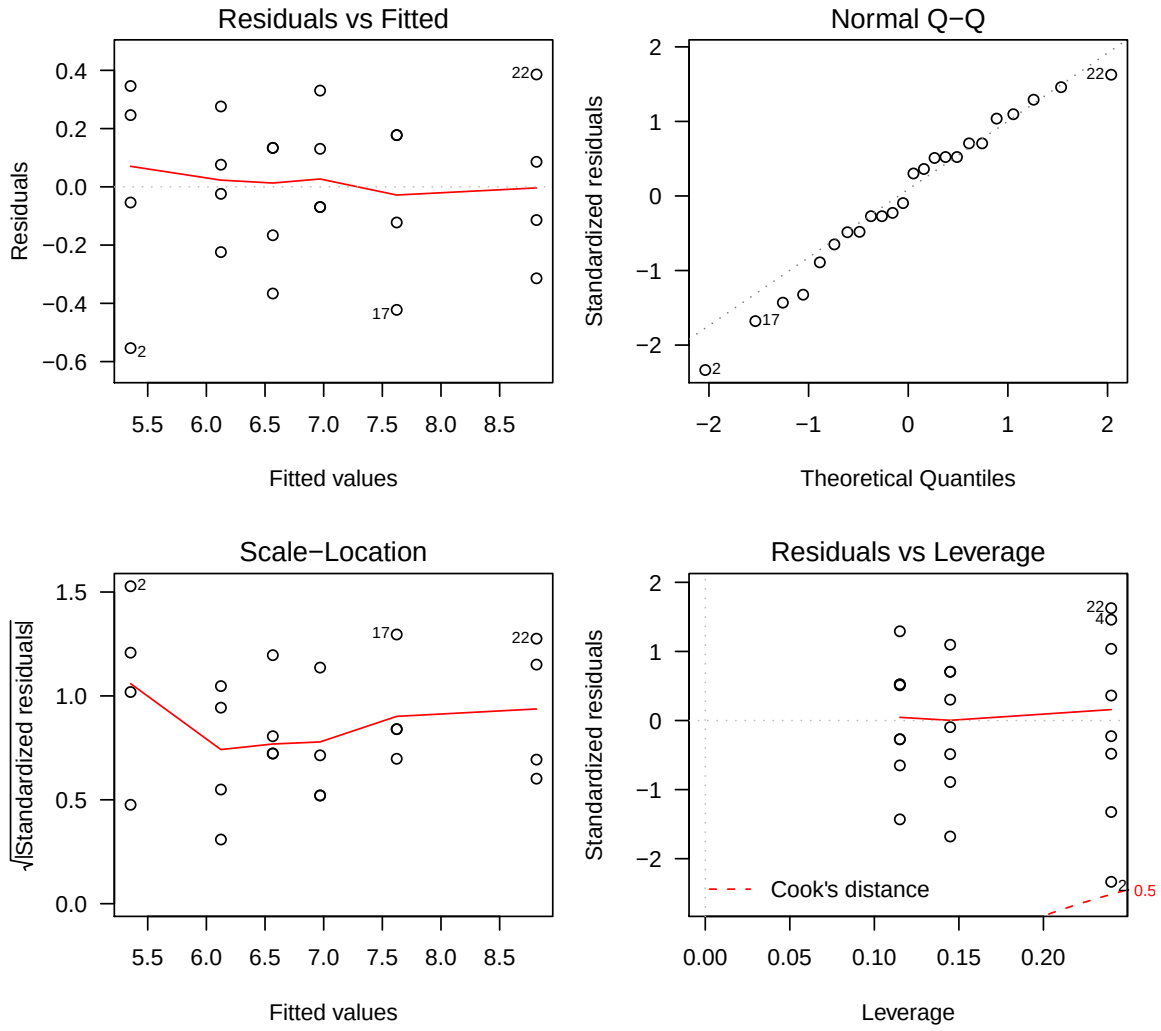

Figure 7.1: Regression diagnostic plots for cubic polynomial linear model for plant height growth data.

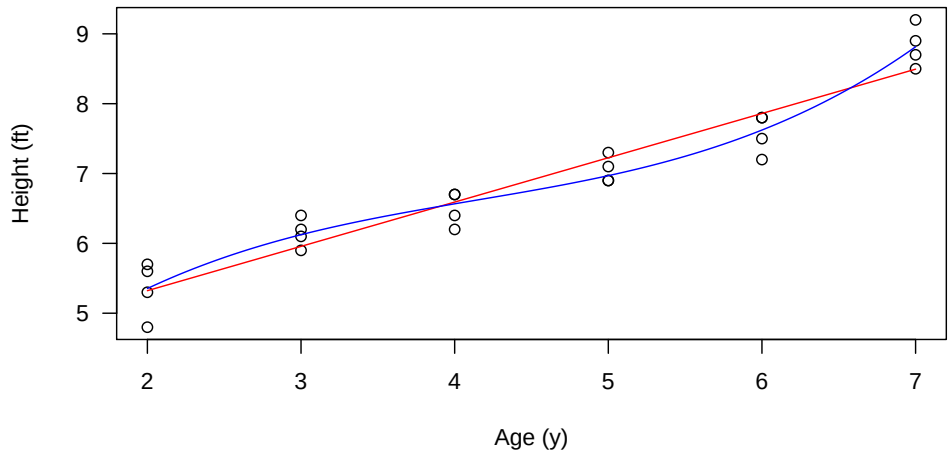

Figure 7.2: Graphical comparison of the linear and cubic models for plant height growth data.

- the errors (and therefore the response variable Y) have a normal distribution;
- the explanatory variables influence Y by influencing the mean in a linear way, e.g. $\mu_Y = \alpha + \beta x$;
- the variance of Y is not affected by the explanatory variables, i.e. $\text{var}(Y) = \sigma^2$ across the whole range of values, signifying constant variance or homoscedasticity.

Under these assumptions, we can compare two models by an F-test. Suppose the two models are M0 and M1, where M0 corresponds to our null hypothesis. Let $y_1, y_2, \ldots, y_n$ be the observed values of the response variable. The process is as follows (SS denotes sum of squares):

1. For model M0:
   (a) fit the model; that is, estimate the parameters;
   (b) for each observation y_i, calculate its predicted value $\hat{y}_i$;
   (c) calculate the residuals $y_i - \hat{y}_i, \quad i = 1, \ldots, n$;
   (d) calculate the sum of the squares of the residuals; this is the **residual sums of squares** for model M0, which we call RSS_0;
   (e) calculate the degrees of freedom associated with RSS_0, which is $df_0 = n - p_0$, p_0 being the number of parameters of M0 that were estimated.
2. Repeat the above for M1 to obtain the residual SS for M1, RSS_1, and its degree of freedom, df_1.
3. The F–statistic is given by
$$\begin{aligned} F &= \frac{\text{change in RSS/change in } df}{\text{RSS}_1/df_1} \\ &= \frac{(\text{RSS}_0 - \text{RSS}_1)/(df_0 - df_1)}{\text{RSS}_1/df_1} \end{aligned}$$
4. Under the null hypothesis (which specifies M0), the statistic F has an F distribution with degrees of freedom $(df_0 - df_1, df_1)$.
5. A large value of F indicates that the change in RSS is substantial when we change from M0 to M1, meaning that M0 does not fit as well as M1. Hence, large values of F cause us to reject H_0. This is a right-tailed test, as the F distribution takes only positive values.

The residual sum of squares (RSS) measures the total departure of the data from the model. If the difference $\text{RSS}_0 - \text{RSS}_1$ is small, then the model M0 is almost as good as M1, and we would accept M0 in the spirit of parsimony (simpler models are better). The F statistic takes into account how many extra parameters ($= df_0 - df_1$) are used to achieve the reduction in RSS and also the intrinsic variation (RSS_1/df_1) in the data.

Some information on F–distributions

An F–distribution arises from the ratio of two independent χ^2 random variables. If $U \sim \chi^2_m$ and $V \sim \chi^2_n$ and U and V are independent, then
$$F = \frac{U/m}{V/n}$$
has an F–distribution with degrees of freedom m and n, denoted by $F_{m,n}$.

7.3.3 Checking the utility of a model

When fitting a model, it is possible to check straight away whether it is significantly better than the null model. The null model assumes that none of the explanatory variables have any effect on the response variable, in which case the predicted value for each observation is simply the mean of all the y values (graphically this would be represented by a line of zero slope through the mean). Thus,
$$\text{residual SS from null model} = \text{total SS} = \sum (y - \bar{y})^2.$$
An F-test can be used to compare the model of interest with the null model. In R, this information is provided at the bottom of the summary of the linear model:

```
> summary(plants.lm.1)

Call:
lm(formula = height ~ age, data = plants)

Residuals:
      Min        1Q    Median        3Q       Max
-0.659762 -0.224762 -0.008333  0.215238  0.705952

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.05405    0.19378   20.92 5.19e-16 ***
age          0.63429    0.04026   15.76 1.82e-13 ***
---
Signif. codes:  0 ,***, 0.001 ,**, 0.01 ,*, 0.05 ,., 0.1 , , 1

Residual standard error: 0.3368 on 22 degrees of freedom
Multiple R-squared: 0.9186,        Adjusted R-squared: 0.9149
F-statistic: 248.2 on 1 and 22 DF,  p-value: 1.821e-13
```

In multiple regression, this test of "utility" is usually of limited interest, because it only answers the question "is fitting the model of interest better than nothing?". Of more interest is the proportion of the total variation that is not explained by the model,

$$\frac{\text{residual SS}}{\text{total SS}}$$

or the proportion that *is* explained by the model.

$$R^2 \;=\; 1 - \frac{\text{residual SS}}{\text{total SS}} \;=\; \frac{\text{regression SS}}{\text{total SS}}$$

R^2 is sometimes denoted `R-sq` or `multiple R-squared`, and is also called the **coefficient of determination** in some books.

7.3.4 Model selection

The overall guiding principle is the principle of parsimony: we want the simplest model that fits the data well. So, in general we do not include a variable in a model unless it significantly reduces the residual SS.

Model selection methods can be divided into two types.

1. Consider all "reasonable" models and compute one or more measures of "goodness" for each. Compare these quantities to choose the most satisfactory model.
2. Consider only a few models in a sequential approach. These methods are often referred to as automatic selection, or step-wise procedures. Such methods are best suited to when there are many possible explanatory variables.

Goodness measures

- $R^2 = \dfrac{\text{regression SS}}{\text{total SS}} = 1 - \dfrac{\text{residual SS}}{\text{total SS}}$
  R^2 cannot decrease if a new variable is added to a model, hence the biggest model always has the largest value of R^2.
- Adjusted $R^2 = R^2_{adj} = \bar{R}^2 = 1 - \dfrac{\text{residual MS}}{\text{total MS}}$.
  $\bar{R}^2$ imposes a penalty for adding variables to the model.
  $\bar{R}^2$ increases only if the added variable is 'good enough' to overcome the penalty. $\bar{R}^2$ can be negative when the model fit is very poor.

- `AIC` (Akaike's "Information Criterion")

$$\begin{aligned} \texttt{AIC} &= -2 \text{ maximised log-likelihood} + 2p \\ &= \frac{\text{residual SS}}{\sigma^2} + 2p + \text{constant} \quad \text{if } \sigma^2 \text{ known} \\ &= n \log[\text{residual SS}/n] + 2p + \text{constant} \quad \text{if } \sigma^2 \text{ unknown} \end{aligned}$$

  Good models have small values of `AIC`. There is a close relationship between `AIC` and Mallow's C_p. Other information criteria are also available; for example Schwarz's BIC (Bayesian Information Criterion) and the AICc, which is Akaike's Information Criterion corrected for small sample sizes.

Remarks

- Good models have:
  - *small* residual SS, $\hat{\sigma}$, C_p, `AIC`;
  - *large* R^2, $\bar{R}^2$.
- All measures depend on the residual SS in some way, so any unusual points or features (e.g. heteroscedasticity) may distort variable selection.
- One approach is to compute the measure(s) for all 'reasonable' models and study further (look at residuals etc.) those models with good values of the measure(s). This approach:
  - is *good* because it is comprehensive (examines all reasonable models);
  - may be *bad* if there are a very large number of models to examine – up to 2^k models for k predictors (e.g. $2^{10} = 1024$).

By the adjusted R^2, s^2 and AIC criteria, model (x_1, x_2, x_4) is 'best', whereas by the C_p criterion, model (x_1, x_2) is 'best'.

7.3.5 Stepwise (sequential) methods

Using AIC

We use the AIC for these stepwise methods, which is what R does with the following functions:

- forward with `add1()`
- backward with `drop1()`
- stepwise with `step()`

The algorithm is as follows:

- starts with a given model (`model_1`, say)
- calculates `AIC` for `model_1` as well as for all (reasonable) models with one term omitted, and for all (reasonable) models obtained by adding one term (from a given list — the *scope*) as follows:

$$AIC = n \log[\text{residual SS}/n] + 2r$$

- drops or adds the term that reduces `AIC` the most
- continues until it is not possible to reduce `AIC` by adding or dropping *one* term
- the final model will, in general, depend upon the starting model (`model_1`).

1. **Backward elimination**

   This method starts with the full model, then omits the least significant predictor, then the next least, and so on until only significant predictors remain.

   (a) Fit the full model.

   (b) Compute the AIC for omission of each variable.

   (c) Select the model corresponding to the lowest AIC.

(d) If that is the starting model, stop. Otherwise, repeat from (b).

2. **Forward selection**

   (a) Fit all one-variable models.

   (b) Compute the AIC for each variable.

   (c) Select the model corresponding to the lowest AIC and repeat from (a) with all two-variable models that include it.

   (d) Quit when AIC can't be lowered further.

Which method is best?

None of them is ideal and different methods can lead to different final models. Forward selection is generally seen as less effective than backward elimination. The algorithms have the advantage of being relatively easily applied to a range of different model-fitting tools.

Example **Heat evolved in the hardening of cement.**

An experiment was conducted to examine the effect of the composition of cement on heat evolved during hardening. Fourteen clinker compositions were formulated using oxides and carbonates, dried into bars, fired in a furnace, and cooled. Each clinker was crushed, ground, and rolled, and a sample of each was taken for chemical analysis. The data consist of 4 predictor variables and the response, as follows:

y: Heat evolved per gram of cement (in calories)
x_1: Amount of tricalcium aluminate
x_2: Amount of tricalcium silicate
x_3: Amount of tetracalcium alumino ferrite
x_4: Amount of dicalcium silicate

```
> heat <- read.csv("../data/heat.csv")
> heat
```

```
   x1 x2 x3 x4     y
1   7 26  6 60  78.5
2   1 29 15 52  74.3
3  11 56  8 20 104.3
4  11 31  8 47  87.6
5   7 52  6 33  95.9
6  11 55  9 22 109.2
7   3 71 17  6 102.7
8   1 31 22 44  72.5
9   2 54 18 22  93.1
10 21 47  4 26 115.9
11  1 40 23 34  83.8
12 11 66  9 12 113.3
13 10 68  8 12 109.4
```

(a) **Start with the 'full' model.**

```
> heat.lm1 <- lm(y~x1+x2+x3+x4, data=heat)
> coef(summary(heat.lm1))
```

```
             Estimate Std. Error    t value   Pr(>|t|)
(Intercept) 62.4053693 70.0709592  0.8906025 0.39913356
x1           1.5511026  0.7447699  2.0826603 0.07082169
x2           0.5101676  0.7237880  0.7048577 0.50090110
x3           0.1019094  0.7547090  0.1350314 0.89592269
x4          -0.1440610  0.7090521 -0.2031741 0.84407147
```

The following `step()` command starts with the full model, and adds "nothing", i.e. uses backward elimination.

```
> step(heat.lm1, ~.)

Start:  AIC=26.94
y ~ x1 + x2 + x3 + x4

       Df Sum of Sq    RSS    AIC
- x3    1    0.1091 47.973 24.974
- x4    1    0.2470 48.111 25.011
- x2    1    2.9725 50.836 25.728
<none>              47.864 26.944
- x1    1   25.9509 73.815 30.576

Step:  AIC=24.97
y ~ x1 + x2 + x4

       Df Sum of Sq    RSS    AIC
<none>              47.97 24.974
- x4    1      9.93  57.90 25.420
+ x3    1      0.11  47.86 26.944
- x2    1     26.79  74.76 28.742
- x1    1    820.91 868.88 60.629

Call:
lm(formula = y ~ x1 + x2 + x4, data = heat)

Coefficients:
(Intercept)           x1           x2           x4
    71.6483       1.4519       0.4161      -0.2365
```

Omitting x_3 reduces AIC, so the model with (x_1, x_2, x_4) is 'best'.

(b) **Start with the 'null' model.**

```
> heat.lm2 <- lm(y~1, data=heat)
```

"1" represents a null model, i.e. no predictors. The following `step()` command starts with the null model then adds any combination of x_1, x_2, x_3 and x_4.

```
> step(heat.lm2, ~. + x1 + x2 + x3 + x4)

Start:  AIC=71.44
y ~ 1

       Df Sum of Sq     RSS    AIC
+ x4    1   1831.90  883.87 58.852
+ x2    1   1809.43  906.34 59.178
+ x1    1   1450.08 1265.69 63.519
+ x3    1    776.36 1939.40 69.067
<none>              2715.76 71.444

Step:  AIC=58.85
y ~ x4

       Df Sum of Sq     RSS    AIC
+ x1    1    809.10   74.76 28.742
```

```
+ x3    1    708.13  175.74 39.853
<none>               883.87 58.852
+ x2    1     14.99  868.88 60.629
- x4    1   1831.90 2715.76 71.444

Step:  AIC=28.74
y ~ x4 + x1

       Df Sum of Sq     RSS    AIC
+ x2    1     26.79   47.97 24.974
+ x3    1     23.93   50.84 25.728
<none>                74.76 28.742
- x1    1    809.10  883.87 58.852
- x4    1   1190.92 1265.69 63.519

Step:  AIC=24.97
y ~ x4 + x1 + x2

       Df Sum of Sq    RSS    AIC
<none>               47.97 24.974
- x4    1      9.93  57.90 25.420
+ x3    1      0.11  47.86 26.944
- x2    1     26.79  74.76 28.742
- x1    1    820.91 868.88 60.629

Call:
lm(formula = y ~ x4 + x1 + x2, data = heat)

Coefficients:
(Intercept)           x4           x1           x2
    71.6483      -0.2365       1.4519       0.4161
```

Again, the model with (x_1, x_2, x_4) is 'best' according to the AIC criterion. Output from the `step` command can be abbreviated by using the `trace = FALSE` argument.

We now look more closely at the model selected by the `step` procedure.

```
> heat.lm3 <- lm(y ~ x1 + x2 + x4, data = heat)
> summary(heat.lm3)

Call:
lm(formula = y ~ x1 + x2 + x4, data = heat)

Residuals:
    Min      1Q  Median      3Q     Max
-3.0919 -1.8016  0.2562  1.2818  3.8982

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  71.6483    14.1424   5.066 0.000675 ***
x1            1.4519     0.1170  12.410 5.78e-07 ***
x2            0.4161     0.1856   2.242 0.051687 .
x4           -0.2365     0.1733  -1.365 0.205395
---
Signif. codes:  0 ,***, 0.001 ,**, 0.01 ,*, 0.05 ,., 0.1 , , 1

Residual standard error: 2.309 on 9 degrees of freedom
Multiple R-squared: 0.9823,        Adjusted R-squared: 0.9764
F-statistic: 166.8 on 3 and 9 DF,  p-value: 3.323e-08
```

This is clearly a good fitting model, but the method has included a predictor (x_4) with a final P-value of 0.205.

7.4 Transformations

Transformations of the response variable and, at times, the explanatory variables, should be considered, especially when:

- all values are positive
- the range of the data is large $\frac{\max}{\min} > 5$ (say)
- the data are skewed (univariate data)
- the scatterplot is curved (bivariate data)
- the variance of the errors is not constant (heteroscedasticity)

Taking logs is by far the most commonly used transformation, often with a 'nice' interpretation.

- $\log(y) = \alpha + \beta x \Rightarrow y = e^{\alpha+\beta x} = Ae^{\beta x}$, where $A = e^{\alpha}$ (exponential growth or decay)
- $\log(y) = \alpha + \beta \log(x) \Rightarrow y = Ax^{\beta}$, where $A = e^{\alpha}$ (power function model)
- $\log(y) = \mu_i$ in group $i \Rightarrow \mu_i - \mu_j = \log(y_i) - \log(y_j) = \log\left(\frac{y_i}{y_j}\right)$ (differences relate to ratios)

It doesn't matter if you use $\log_{10}$ or $\log_e$, provided you are consistent.

Commonly used transformations

$\log(y)$	for ratios, skewed data, or very large counts
$\sqrt{y}$	for areas, or counts
$\sqrt[3]{y}$	for volumes
$1/y$	for rates
$\left.\begin{array}{r}\arcsin\sqrt{\frac{y}{n}}\\ \text{or } \log\frac{y}{n-y}\end{array}\right\}$	for binomial data $[Y \overset{d}{=} \text{Bin}(n,p)]$ The effect of this transformation is more pronounced for p close to 0 or 1.

Transforming data can improve linearity, constant variance and normality, simultaneously.

- If the variance increases with the mean of the response, log transform the response variable.
- Transforming can affect other aspects of model fitting, such as variable selection.
- Don't transform unless it clearly helps.
- If using polynomials, include all lower order powers unless there is a good *theoretical* reason for not doing so. For example, if x^3 is included, x and x^2 should also be included.

7.5 Exercises

1. **Taste of cheese**

   As cheddar cheese matures, a variety of chemical processes take place, and the taste of matured cheese is related to the concentration of several chemicals. The data in `cheese.csv` were obtained from one cheese manufacturing process. Taste is the response variable of interest, based on combining scores of many tasters. There are also measures of three chemicals: acetic acid, hydrogen sulfide and lactic acid.

   (a) Load the data into R and produce an appropriate graphical display.

(b) Fit a separate regression of taste on each of the three explanatory variables.

(c) Fit a multiple regression of taste on the three variables and state your conclusions. In particular, are the three explanatory variables needed? If not, determine the best model.

(d) Find the correlation coefficients between the four variables and use them to help interpret your findings.

(e) For the best model, interpret the coefficient of the most significant explanatory variable.

2. **Soil Data**

Return to the soil sediment data used at the start of this lab.

(a) Construct an appropriate graph of these data. Comment on the relationships.

(b) Fit a model with the only predictor being iron, and then fit the full model (iron + aluminium).

(c) Obtain a predicted value for the phosphate absorption index when iron = 150. Compare values between models.

(d) Examine diagnostic plots for the full model, and comment.

(e) Use the AIC criterion to choose the best model. Does this match with the results from part (b)?

3. **Girth of horses**

A small laboratory experiment was performed on the girths of horses (the girth is the belt holding the saddle). Different weights were hung on the end of the girth, to simulate the tension applied during a horse race. The variable of interest was the amount that the girth stretched as a proportion of its total length (call this variable `stretch`). The results were as follows (weight is in kg):

weight	2.5	4.5	6.5	8.5	10.5	12.5	14.5	16.5	18.5	20.5
stretch	.0136	.0172	.0230	.0301	.0372	.0430	.0469	.0448	.0401	.0301

(a) Enter the data into R, and plot stretch against weight. Find the correlation coefficient, and comment on the usefulness of it in this situation.

(b) Fit a polynomial regression, increasing the power of the explanatory variable until you reach a final model. Check diagnostic plots along the way.

(c) Predict the stretch when the weight is 10.5 kg, and hence find the residual for that observation.

4. **Phosphorus content in corn.**

Where does the phosphorus in corn plants get its P from? An experiment measured the following:

```
avP  = phosphorus actually in plant (available P);
ino  = amount of inorganic P in the soil;
org1 = amount of organic P in the soil of one type;
org2 = amount of organic P in the soil of another type.
```

avP	ino	org1	org2
64	0.4	53	158
60	0.4	23	163
71	3.1	19	37
61	0.6	34	157
54	4.7	24	59
77	1.7	65	123
81	9.4	44	46
93	10.1	31	117
93	11.6	29	173
51	12.6	58	112
76	10.9	37	111
96	23.1	46	114

```
77   23.1     50    134
93   21.6     44     73
95   23.1     56    168
54    1.9     36    143
99   29.9     51    124
```

Read the data into R (`Pcorn.csv`).

(a) Construct an appropriate graph of these data, and examine the correlation between the variables. Comment on the relationships.

(b) Find the most appropriate model for predicting the available P, using both backward elimination and forward selection.

5. **Estimating timber volume**

In the lecture, several models were fitted relating volume to diameter and height. The data are in `timber.csv`. By transforming the data, see if you can produce a model with a larger R^2 than those produced in the lecture.