A Handbook of Statistical Analyses Using R

Brian S. Everitt and Torsten Hothorn

CHAPTER 5

Multiple Linear Regression: Cloud Seeding

5.1 Introduction

5.2 Multiple Linear Regression

5.3 Analysis Using R

Both the boxplots (Figure 5.1) and the scatterplots (Figure 5.2) show some evidence of outliers. The row names of the extreme observations in the `clouds` *data.frame* can be identified via

```
R> rownames(clouds)[clouds$rainfall %in% c(bxpseeding$out,
+     bxpecho$out)]
```

```
[1] "1"  "15"
```

where `bxpseeding` and `bxpecho` are variables created by `boxplot` in Figure 5.1. For the time being we shall not remove these observations but bear in mind during the modelling process that they may cause problems.

5.3.1 Fitting a Linear Model

In this example it is sensible to assume that the effect that some of the other explanatory variables is modified by seeding and therefore consider a model that allows interaction terms for `seeding` with each of the covariates except `time`. This model can be described by the *formula*

```
R> clouds_formula <- rainfall ~ seeding * (sne + cloudcover +
+     prewetness + echomotion) + time
```

and the design matrix $\mathbf{X}^\star$ can be computed via

```
R> Xstar <- model.matrix(clouds_formula, data = clouds)
```

By default, treatment contrasts have been applied to the dummy codings of the factors `seeding` and `echomotion` as can be seen from the inspection of the `contrasts` attribute of the model matrix

```
R> attr(Xstar, "contrasts")
```

```
$seeding
[1] "contr.treatment"

$echomotion
[1] "contr.treatment"
```

```
R> data("clouds", package = "HSAUR")
R> layout(matrix(1:2, nrow = 2))
R> bxpseeding <- boxplot(rainfall ~ seeding, data = clouds,
+     ylab = "Rainfall", xlab = "Seeding")
R> bxpecho <- boxplot(rainfall ~ echomotion, data = clouds,
+     ylab = "Rainfall", xlab = "Echo Motion")
```

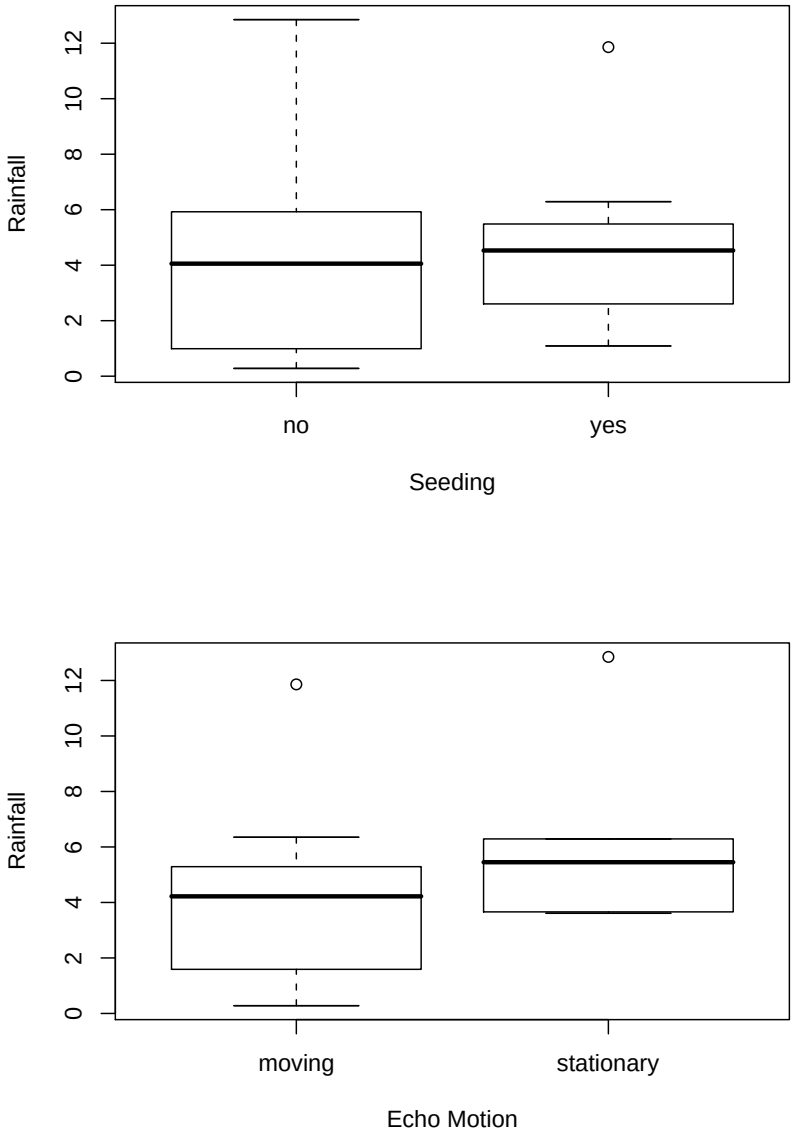

Figure 5.1 Boxplots of `rainfall`.

```
R> layout(matrix(1:4, nrow = 2))
R> plot(rainfall ~ time, data = clouds)
R> plot(rainfall ~ sne, data = clouds, xlab = "S-NE criterion")
R> plot(rainfall ~ cloudcover, data = clouds)
R> plot(rainfall ~ prewetness, data = clouds)
```

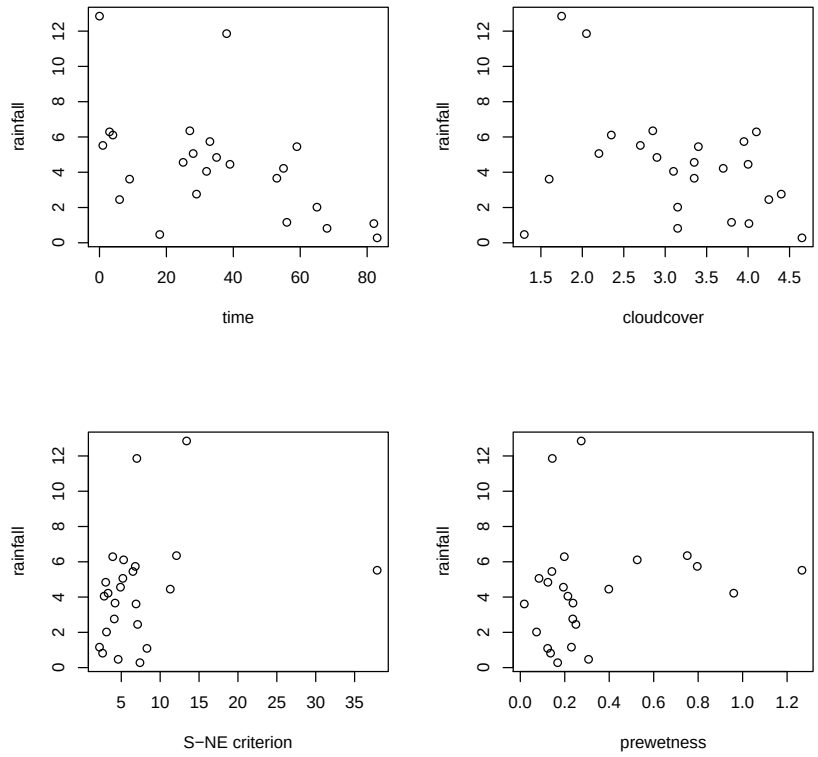

Figure 5.2 Scatterplots of `rainfall` against the continuous covariables.

The default contrasts can be changed via the `contrasts.arg` argument to `model.matrix` or the `contrasts` argument to the fitting function, for example `lm` or `aov` as shown in Chapter **??**. However, such internals are hidden and performed by high-level model fitting functions such as `lm` which will be used to fit the linear model defined by the *formula* `clouds_formula`:

```
R> clouds_lm <- lm(clouds_formula, data = clouds)
R> class(clouds_lm)
```

```
[1] "lm"
```

The results of the model fitting is an object of class *lm* for which a `summary` method showing the conventional regression analysis output is available. The output in Figure 5.3 shows the estimates $\hat{\beta}^\star$ with corresponding standard errors and t-statistics as well as the F-statistic with associated p-value.

```
R> summary(clouds_lm)

Call:
lm(formula = clouds_formula, data = clouds)

Residuals:
    Min      1Q  Median      3Q     Max
-2.5259 -1.1486 -0.2704  1.0401  4.3913

Coefficients:
                               Estimate Std. Error t value
(Intercept)                    -0.34624    2.78773  -0.124
seedingyes                     15.68293    4.44627   3.527
sne                             0.38786    0.21786   1.780
cloudcover                      0.41981    0.84453   0.497
prewetness                      4.10834    3.60101   1.141
echomotionstationary            3.15281    1.93253   1.631
time                           -0.04497    0.02505  -1.795
seedingyes:sne                 -0.48625    0.24106  -2.017
seedingyes:cloudcover          -3.19719    1.26707  -2.523
seedingyes:prewetness          -2.55707    4.48090  -0.571
seedingyes:echomotionstationary -0.56222    2.64430  -0.213
                               Pr(>|t|)
(Intercept)                     0.90306
seedingyes                      0.00372 **
sne                             0.09839 .
cloudcover                      0.62742
prewetness                      0.27450
echomotionstationary            0.12677
time                            0.09590 .
seedingyes:sne                  0.06482 .
seedingyes:cloudcover           0.02545 *
seedingyes:prewetness           0.57796
seedingyes:echomotionstationary 0.83492
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.205 on 13 degrees of freedom
Multiple R-Squared: 0.7158,        Adjusted R-squared: 0.4972
F-statistic: 3.274 on 10 and 13 DF,  p-value: 0.02431
```

Figure 5.3 R output of the linear model fit for the `clouds` data.

Many methods are available for extracting components of the fitted model. The estimates $\hat{\beta}^\star$ can be assessed via

```
R> betastar <- coef(clouds_lm)
R> betastar
```

```
                    (Intercept)
                    -0.34624093
                     seedingyes
                    15.68293481
                            sne
                     0.38786207
                     cloudcover
                     0.41981393
                     prewetness
                     4.10834188
            echomotionstationary
                     3.15281358
                           time
                    -0.04497427
                  seedingyes:sne
                    -0.48625492
           seedingyes:cloudcover
                    -3.19719006
           seedingyes:prewetness
                    -2.55706696
seedingyes:echomotionstationary
                    -0.56221845
```

and the corresponding covariance matrix $\mathsf{Cov}(\hat{\beta}^\star)$ is available from the `vcov` method

```
R> Vbetastar <- vcov(clouds_lm)
```

where the square roots of the diagonal elements are the standard errors as shown in Figure 5.3

```
R> sqrt(diag(Vbetastar))
```

```
                    (Intercept)
                     2.78773403
                     seedingyes
                     4.44626606
                            sne
                     0.21785501
                     cloudcover
                     0.84452994
                     prewetness
                     3.60100694
            echomotionstationary
                     1.93252592
                           time
                     0.02505286
```

```
                  seedingyes:sne
                       0.24106012
           seedingyes:cloudcover
                       1.26707204
           seedingyes:prewetness
                       4.48089584
seedingyes:echomotionstationary
                       2.64429975
```

5.3.2 Regression Diagnostics

In order to investigate the quality of the model fit, we need access to the residuals and the fitted values. The residuals can be found by the `residuals` method and the fitted values of the response from the `fitted` (or `predict`) method

```
R> clouds_resid <- residuals(clouds_lm)
R> clouds_fitted <- fitted(clouds_lm)
```

Now the residuals and the fitted values can be used to construct diagnostic plots; for example the residual plot in Figure 5.5 where each observation is labelled by its number. Observations 1 and 15 give rather large residual values and the data should perhaps be reanalysed after these two observations are removed. The normal probability plot of the residuals shown in Figure 5.6 shows a reasonable agreement between theoretical and sample quantiles, however, observations 1 and 15 are extreme again. An index plot of the Cook's distances for each observation (and many other plots including those constructed above from using the basic functions) can be found from applying the `plot` method to the object that results from the application of the `lm` function. Figure 5.7 suggests that observations 2 and 18 have undue influence on the estimated regression coefficients, but the two outliers identified previously do not. Again it may be useful to look at the results after these two observations have been removed (see Exercise 5.2).

```
R> psymb <- as.numeric(clouds$seeding)
R> plot(rainfall ~ cloudcover, data = clouds, pch = psymb)
R> abline(lm(rainfall ~ cloudcover, data = clouds,
+     subset = seeding == "no"))
R> abline(lm(rainfall ~ cloudcover, data = clouds,
+     subset = seeding == "yes"), lty = 2)
R> legend("topright", legend = c("No seeding", "Seeding"),
+     pch = 1:2, lty = 1:2, bty = "n")
```

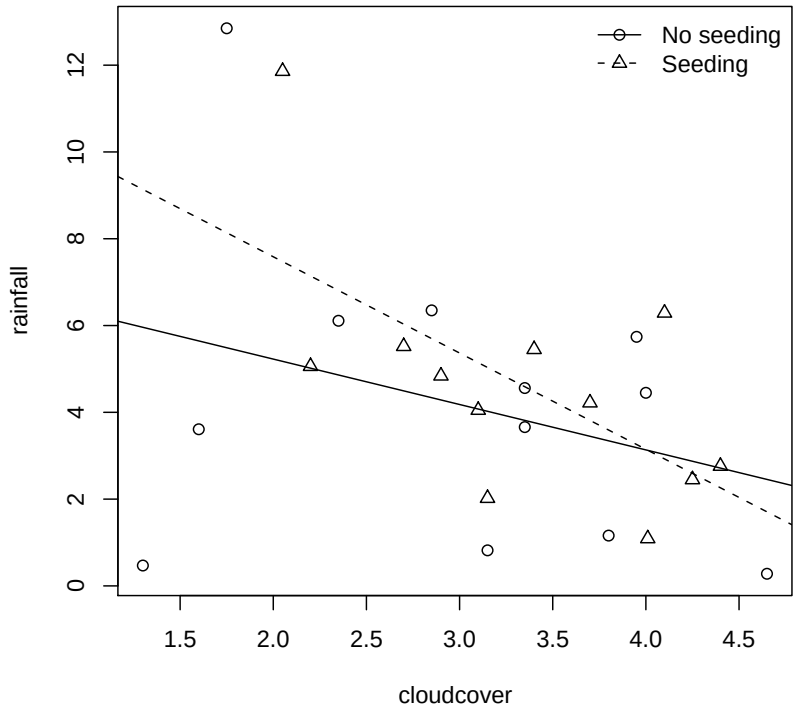

Figure 5.4 Regression relationship between cloud coverage and rainfall with and without seeding.

```
R> plot(clouds_fitted, clouds_resid, xlab = "Fitted values",
+     ylab = "Residuals", ylim = max(abs(clouds_resid)) *
+         c(-1, 1), type = "n")
R> abline(h = 0, lty = 2)
R> text(clouds_fitted, clouds_resid, labels = rownames(clouds))
```

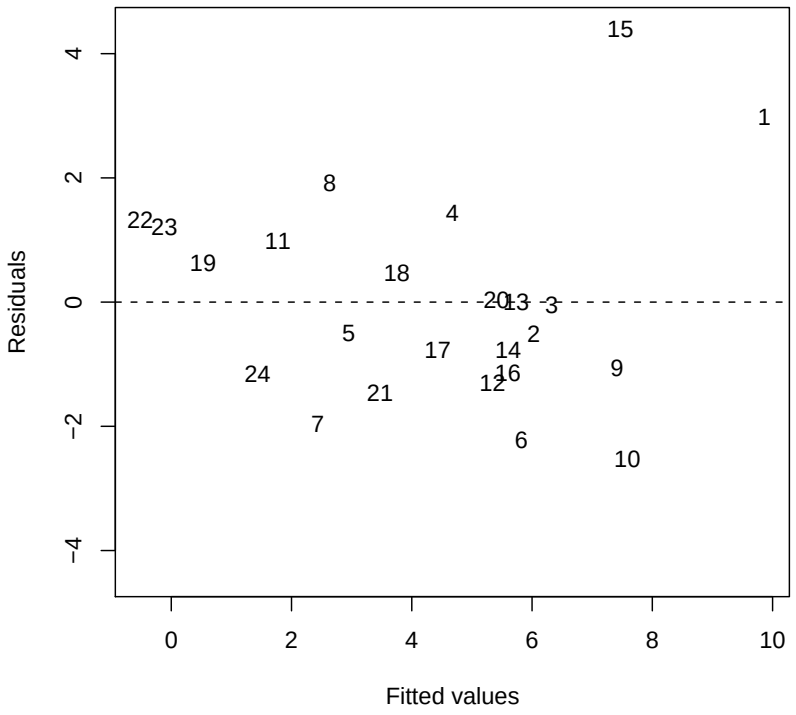

Figure 5.5 Plot of residuals against fitted values for `clouds` seeding data.

```
R> qqnorm(clouds_resid, ylab = "Residuals")
R> qqline(clouds_resid)
```

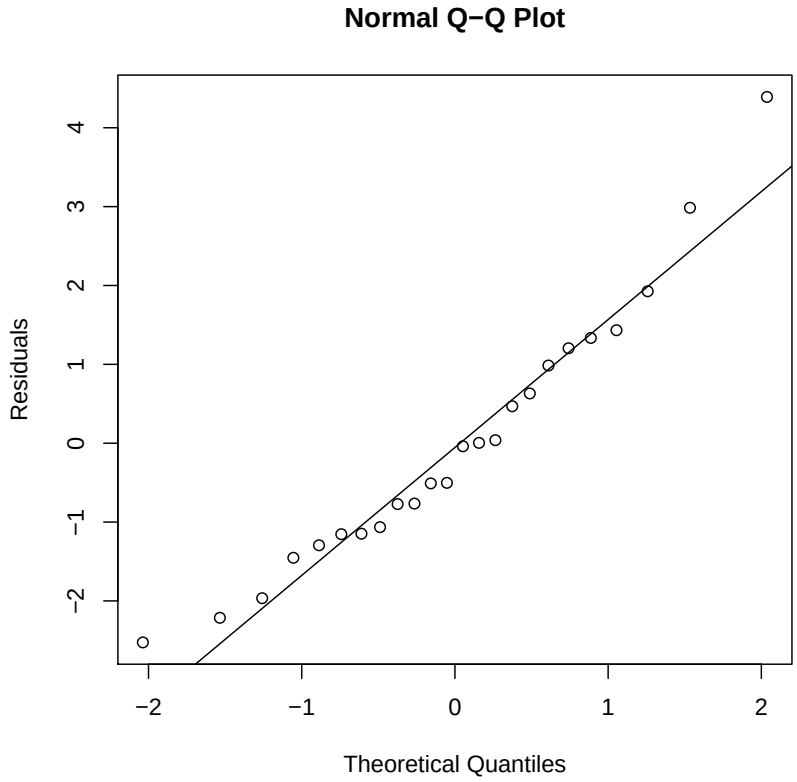

Figure 5.6 Normal probability plot of residuals from cloud seeding model `clouds_lm`.

```
R> plot(clouds_lm)
```

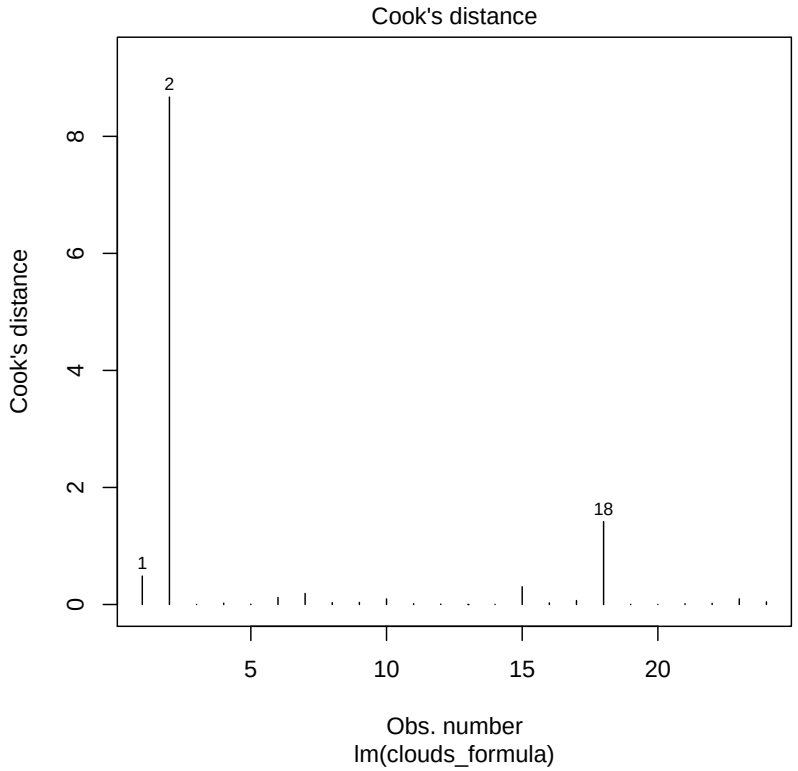

Figure 5.7 Index plot of Cook's distances for cloud seeding data.