A Handbook of Statistical Analyses Using R — 3rd Edition

Torsten Hothorn and Brian S. Everitt
CHAPTER 6

Simple and Multiple Linear Regression: How Old is the Universe and Cloud Seeding

6.1 Introduction

? give the relative velocity and the distance of 24 galaxies, according to measurements made using the Hubble Space Telescope – the data are contained in the gamair package accompanying ?, see Table 6.1. Velocities are assessed by measuring the Doppler red shift in the spectrum of light observed from the galaxies concerned, although some correction for ‘local’ velocity components is required. Distances are measured using the known relationship between the period of Cepheid variable stars and their luminosity. How can these data be used to estimate the age of the universe? Here we shall show how this can be done using simple linear regression.

Table 6.1: hubble data. Distance and velocity for 24 galaxies.

<table>
<thead>
<tr>
<th>galaxy</th>
<th>velocity</th>
<th>distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>NGC0300</td>
<td>133</td>
<td>2.00</td>
</tr>
<tr>
<td>NGC0925</td>
<td>664</td>
<td>9.16</td>
</tr>
<tr>
<td>NGC1326A</td>
<td>1794</td>
<td>16.14</td>
</tr>
<tr>
<td>NGC1365</td>
<td>1594</td>
<td>17.95</td>
</tr>
<tr>
<td>NGC1425</td>
<td>1473</td>
<td>21.88</td>
</tr>
<tr>
<td>NGC2403</td>
<td>278</td>
<td>3.22</td>
</tr>
<tr>
<td>NGC2541</td>
<td>714</td>
<td>11.22</td>
</tr>
<tr>
<td>NGC2090</td>
<td>882</td>
<td>11.75</td>
</tr>
<tr>
<td>NGC3031</td>
<td>80</td>
<td>3.63</td>
</tr>
<tr>
<td>NGC3198</td>
<td>772</td>
<td>13.80</td>
</tr>
<tr>
<td>NGC3351</td>
<td>642</td>
<td>10.00</td>
</tr>
<tr>
<td>NGC3368</td>
<td>768</td>
<td>10.52</td>
</tr>
</tbody>
</table>

Weather modification, or cloud seeding, is the treatment of individual clouds or storm systems with various inorganic and organic materials in the hope of achieving an increase in rainfall. Introduction of such material into a cloud that contains supercooled water, that is, liquid water colder than zero degrees Celsius, has the aim of inducing freezing, with the consequent ice particles growing at the expense of liquid droplets and becoming heavy enough to fall as rain from clouds that otherwise would produce none.

The data shown in Table 6.2 were collected in the summer of 1975 from an experiment to investigate the use of massive amounts of silver iodide (100 to 1000 grams per cloud) in cloud seeding to increase rainfall. In the experiment, which was conducted in an area of Florida, 24 days were judged suitable for seeding on the basis that a measured suitability criterion, denoted $S-Nc$, was not less than 1.5. Here $S$ is the ‘seedability’, the difference between the maximum height of a cloud if seeded and the same cloud if not seeded.
predicted by a suitable cloud model, and $Ne$ is the number of hours between 1300 and 1600 G.M.T. with 10 centimeter echoes in the target; this quantity biases the decision for experimentation against naturally rainy days. Consequently, optimal days for seeding are those on which seedability is large and the natural rainfall early in the day is small.

On suitable days, a decision was taken at random as to whether to seed or not. For each day the following variables were measured:

- **seeding** a factor indicating whether seeding action occurred (yes or no),
- **time** number of days after the first day of the experiment,
- **cloudc** the percentage cloud cover in the experimental area, measured using radar,
- **prewet** the total rainfall in the target area one hour before seeding (in cubic meters $\times 10^7$),
- **EM** a factor showing whether the radar echo was moving or stationary,
- **rain** the amount of rain in cubic meters $\times 10^7$,
- **sne** suitability criterion, see above.

The objective in analyzing these data is to see how rainfall is related to the explanatory variables and, in particular, to determine the effectiveness of seeding. The method to be used is *multiple linear regression*.

### 6.2 Simple Linear Regression

### 6.3 Multiple Linear Regression

#### 6.3.1 Regression Diagnostics

### 6.4 Analysis Using R

#### 6.4.1 Estimating the Age of the Universe

Prior to applying a simple regression to the data it will be useful to look at a plot to assess their major features. The R code given in Figure 6.1 produces a scatterplot of velocity and distance. The diagram shows a clear, strong relationship between velocity and distance. The next step is to fit a simple linear regression model to the data, but in this case the nature of the data requires a model without intercept because if distance is zero so is relative speed. So the model to be fitted to these data is

$$\text{velocity} = \beta_1 \text{distance} + \varepsilon.$$  

This is essentially what astronomers call Hubble’s Law and $\beta_1$ is known as Hubble’s constant; $\beta_1^{-1}$ gives an approximate age of the universe.

To fit this model we are estimating $\beta_1$ using formula (??). Although this operation is rather easy

```r
R> sum(hubble$distance * hubble$velocity) / + sum(hubble$distance^2)
```
it is more convenient to apply R’s linear modeling function

\begin{verbatim}
R> hmod <- lm(velocity ~ distance - 1, data = hubble)
\end{verbatim}

Note that the model formula specifies a model without intercept. We can now extract the estimated model coefficients via

\begin{verbatim}
R> coef(hmod)
\end{verbatim}

\begin{verbatim}
distance
  76.6
\end{verbatim}

and add this estimated regression line to the scatterplot; the result is shown in Figure 6.2. In addition, we produce a scatterplot of the residuals $y_i - \hat{y}_i$ against fitted values $\hat{y}_i$ to assess the quality of the model fit. It seems that for higher distance values the variance of velocity increases; however, we
ANALYSIS USING R

R> layout(matrix(1:2, ncol = 2))
R> plot(velocity ~ distance, data = hubble)
R> abline(hmod)
R> plot(hmod, which = 1)

![Scatterplot of velocity and distance with estimated regression line (left) and plot of residuals against fitted values (right).](image)

**Figure 6.2** Scatterplot of velocity and distance with estimated regression line (left) and plot of residuals against fitted values (right).

are interested in only the estimated parameter $\hat{\beta}_1$ which remains valid under variance heterogeneity (in contrast to $t$-tests and associated $p$-values).

Now we can use the estimated value of $\hat{\beta}_1$ to find an approximate value for the age of the universe. The Hubble constant itself has units of $\text{km} \times \text{sec}^{-1} \times \text{Mpc}^{-1}$. A mega-parsec (Mpc) is $3.09 \times 10^{19}\text{km}$, so we need to divide the estimated value of $\beta_1$ by this amount in order to obtain Hubble’s constant with units of $\text{sec}^{-1}$. The approximate age of the universe in seconds will then be the inverse of this calculation. Carrying out the necessary computations

R> Mpc <- 3.09 * 10^-19
R> ysec <- 60^2 * 24 * 365.25
R> Mpcyear <- Mpc / ysec
R> 1 / (coef(hmod) / Mpcyear)

gives an estimated age of roughly 12.8 billion years.
6.4.2 Cloud Seeding

Again, a graphical display highlighting the most important aspects of the data will be helpful. Here we will construct boxplots of the rainfall in each category of the dichotomous explanatory variables and scatterplots of rainfall against each of the continuous explanatory variables.

Both the boxplots (Figure 6.3) and the scatterplots (Figure 6.4) 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$rain %in% c(bxpseeding$out, bxpecho$out)]
```

where bxpseeding and bxpecho are variables created by boxplot in Figure 6.3. Now we shall not remove these observations but bear in mind during the modeling process that they may cause problems.

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

```R
clouds_formula <- rain ~ seeding + 
  seeding:(sne + cloudc + prewet + EM) +
  time
```

and the design matrix $X^*$ 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 EM as can be seen from the inspection of the contrasts attribute of the model matrix

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

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 5.

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"
```
R> data("clouds", package = "HSAUR3")
R> layout(matrix(1:2, nrow = 2))
R> bxpseeding <- boxplot(rain ~ seeding, data = clouds,
+ ylab = "Rainfall", xlab = "Seeding")
R> bxpecho <- boxplot(rain ~ EM, data = clouds,
+ ylab = "Rainfall", xlab = "Echo Motion")
```

Figure 6.3 Boxplots of rain.
R> layout(matrix(1:4, nrow = 2))
R> plot(rain ~ time, data = clouds)
R> plot(rain ~ cloudc, data = clouds)
R> plot(rain ~ sne, data = clouds, xlab="S-Ne criterion")
R> plot(rain ~ prewet, data = clouds)

![Scatterplots of rain against the continuous covariates.](image)

**Figure 6.4** Scatterplots of rain against the continuous covariates.

The result 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 6.5 shows the estimates $\hat{\beta}$ with corresponding standard errors and $t$-statistics as well as the $F$-statistic with associated $p$-value.

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

```r
R> summary(clouds_lm)
Call:
  lm(formula = clouds_formula, data = clouds)
Residuals:
          Min     1Q Median      3Q     Max
-2.532  -1.149  -0.267   1.047   4.396
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  -0.3462     2.7877  -0.123   0.9031
seedingyes   15.6829     4.4463   3.530  0.0037*
time         -0.0450     0.0251  -1.800   0.0959 .
seedingno:sne  0.4198     0.8445   0.500   0.6274
seedingyes:sne -2.7774     0.9284  -2.960  0.0104 *
seedingno:cloudc  0.3879     0.2179   1.763   0.0984 .
seedingyes:cloudc -0.0984     0.1103  -0.899   0.3885
seedingno:prewet  4.1083     3.6010   1.149   0.2745
seedingyes:prewet  1.5513     2.6929   0.580   0.5744
seedingno:EMstationary  3.1528     1.8325   1.632   0.1268
seedingyes:EMstationary  2.5906     1.8173   1.434   0.1776

Residual standard error: 2.20 on 13 degrees of freedom
Multiple R-squared: 0.716,     Adjusted R-squared: 0.497
F-statistic: 3.27 on 10 and 13 DF,  p-value: 0.0243
```

Figure 6.5  R output of the linear model fit for the clouds data.

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

(Intercept)      seedingyes
   -0.3462         15.6829
     time      seedingno:sne
      -0.0450         0.4198
seedingyes:sne   seedingno:cloudc
     -2.7774         0.3879
seedingyes:cloudc   seedingno:prewet
       -0.0984        4.1083
seedingyes:prewet   seedingno:EMstationary
      1.5513         3.1528
seedingyes:EMstationary
             2.5906

and the corresponding covariance matrix \( \text{Cov}(\hat{\beta}^\ast) \) is available from the \texttt{vcov} method

```r
R> Vbetastar <- vcov(clouds_lm)
```n
where the square roots of the diagonal elements are the standard errors as shown in Figure 6.5

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

(Intercept)      seedingyes
    2.7877         4.4463
     time      seedingno:sne
```n
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
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 6.7 where each observation is labelled by its number (using `textplot` from package `wordclouds`). 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 6.8 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 6.9 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 6.2).
ANALYSIS USING R

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

Figure 6.6 Regression relationship between S-Ne criterion and rainfall with and without seeding.
Figure 6.7  Plot of residuals against fitted values for clouds seeding data.
R > qqnorm(clouds_resid, ylab = "Residuals")
R > qqline(clouds_resid)

Figure 6.8 Normal probability plot of residuals from cloud seeding model clouds_lm.
R> plot(clouds_lm)

Figure 6.9  Index plot of Cook’s distances for cloud seeding data.