Regression Analysis

Regression analysis is one of the most important topics in Statistical theory. In the sequel this widely known methodology will be used with `S-Plus` by means of formulae for models.

Models in S-Plus

A formula is an `S-PLUS` expression that specifies the form of a model in terms of the variables involved. For example, to specify that Y is a linear combination of two predictors $X1$ and $X2$, you use the following formula:

```
> Y ~ X1 + X2
```

The tilde character separates the response variable from the explanatory variables. So in essence we fit the model

$$Y = \beta_0 + \beta_1 * X1 + \beta_2 * X2.$$

A formula *always* implicitly includes an intercept term (β_0) in the above formula). You can, however, remove the intercept term by specifying the model with -1 as an explicit predictor:

```
> Y ~ -1 + X1 + X2
```

Similarly, you can explicitly include an intercept with a + 1.

When you specify categorical variables (factors, ordered factors, or categories) as predictors in the formulas, the modelling functions fit a coefficient for each level of the variable. For example, to model `salary` as a linear model of `age` (continuous) and `gender` (factor) you specify it as follows:

```
> salary ~ age + gender
```

However, a different parameter is fitted for each of the two levels of gender. This is equivalent to fitting two dummy variables, one for males and one for females. Thus, you need not create and specify dummy variables in the model.

Here are the main expressions:

- Y X: Y is modeled as X

- X1+X2: Include both X1 and X2 in the model
- X1-X2: Include all of X1 except what is in X2 in the model
- X1:X2; The interaction between X1 and X2
- X1*X2 full model X1 + X2 + X1 : X2

The next sections put these applies these concepts to a multiple linear regression model.

A Multiple Regression Model

The data set consists of a sample of black cherry trees. These were cut down measurements made on the diameter (in inches) height (in feet) and volume (in cubic feet). The purpose of collecting these data was to provide a way of predicting the volume of timber in trees from their height and diameter measurements, using a regression model.

Diameter	Height	Volume
8.3	70	10.3
8.6	65	10.3
8.8	63	10.2
10.5	72	16.4
10.7	81	18.8
10.8	83	19.7

Table 1: The first six observations from the data set

Here the dependent variable is continuous and the initial model considered is the usual linear regression one, having the general form

$$y = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p + \epsilon$$

where y is the response variable, $x_1, \ldots, x_p$ a set of explanatory variables, and ϵ a residual term. We estimate β's by least squares assuming that the residual term is normally distributed with mean 0 and constant variance σ^2. For n observations on the response and explanatory variables the model may be written concisely as

$$E(\mathbf{y}) = \mathbf{X}\beta$$

A formal analysis in `S-Plus` uses the function `lm()` as outlined below:

```
trees.fit <- lm(Volume~ Diameter+Height, trees)
> trees.fit
Call: lm(formula = Volume ~ Diameter + Height, data = trees)
```

```
Coefficients:
 (Intercept) Diameter    Height
   -57.98766 4.708161 0.3392512

Degrees of freedom: 31 total; 28 residual Residual standard error:
3.881832
> trees.res <- residuals(trees.fit) #obtain the residuals
> trees.prd <- predict(trees.fit)   #obtain the predicted values
```

These results suggest that the regression coefficients for both Diameter and Height are significantly different from zero, and that about 95% of the variance (the value given by R^2) in volume can attributed to two explanatory variables.

The next stage in the analysis should be an examination of the residuals from the model, that is the differences between the observed and fitted (predicted) values of the response. The most useful plots are the following:

1. A plot of the residuals against explanatory variable in the model. The presence of a curvilinear relationship, for example, may suggest a higher order term, perhaps a quadratic should be added to the model (Figure 1).

2. A plot of the residuals against predicted values of the response variable. If the variance of the response appears to increase with predicted value, a transformation of the response may be in order (Figure 2).

3. A normal probability plot of the residuals. After all systematic variation has been removed from the data, the residuals should look like a sample from the normal distribution. That is a plot of the ordered residuals against the expected order statistics from a normal distribution (Figure 3).

We are going to work with the standardized residuals which are defined by

$$r_i = \frac{y_i - \hat{y}_i}{s\sqrt{1 - h_{ii}}}$$

where h_{ii} are the diagonal elements of the matrix:

$$\mathbf{H} = \mathbf{X}(\mathbf{X}^{'}\mathbf{X})^{-\mathbf{1}}\mathbf{X}.$$

```
> s <- summary(trees.fit)$sigma    #obtain an estimator of sigma
> h <- lm.influence(trees.fit)$hat  #obtain the diagonal elements of H
> trees.res <- trees.res/(s*sqrt(1-h))  # redefine the residuals so that
                                      # they are standardized.

> par(mfrow=c(2,1))                  # obtain the plots of regressors vs
                                     # standardized residuals
> plot(trees[,"Diameter"], trees.res, xlab="Diameter", ylab="Std. Residuals")
> abline(h=0, lty=2)
> title("Std. Residuals versus Diameter")
> plot(trees[,"Height"], trees.res, xlab="Height", ylab="Std. Residuals")
> abline(h=0, lty=2)
> title("Std. Residuals versus Height")
> par(mfrow=c(1,1))                  #plots of residuals vs predicted values
> plot(trees.prd, trees.res, xlab="Predicted Volume",
     ylab="Std. Residuals")
> abline(h=0,lty=2)
> title("Std. Residuals vs Fitted Values")
> qqnorm(trees.res,ylab="Std. Residuals")  #qqnorm of the residuals
> qqline(trees.res)
> title("Normal Plot of Std. Residuals")
```

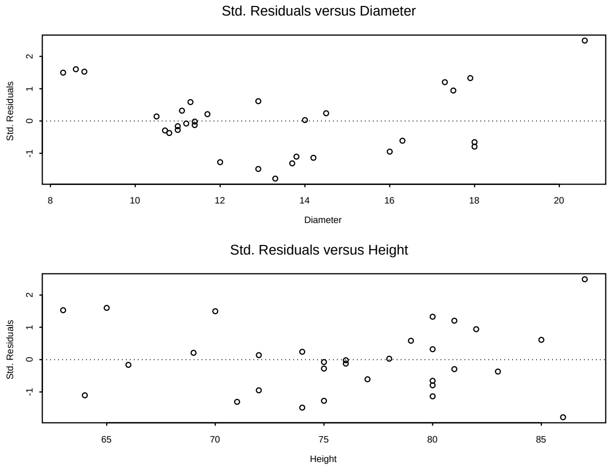

Figure 1: Standardized residuals plotted against values of explanatory variables

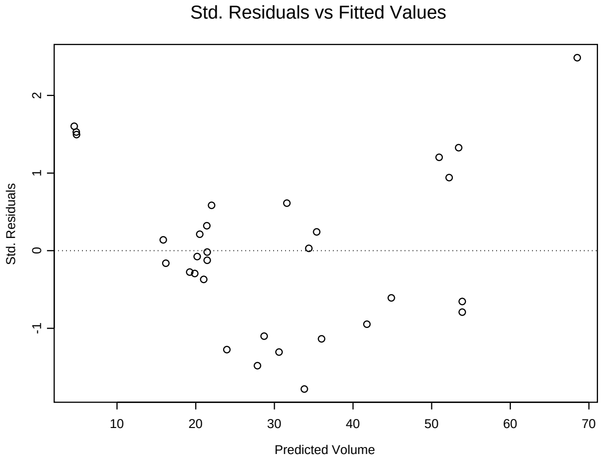

Figure 2: Standardized residuals plotted against fitted values of response variables.

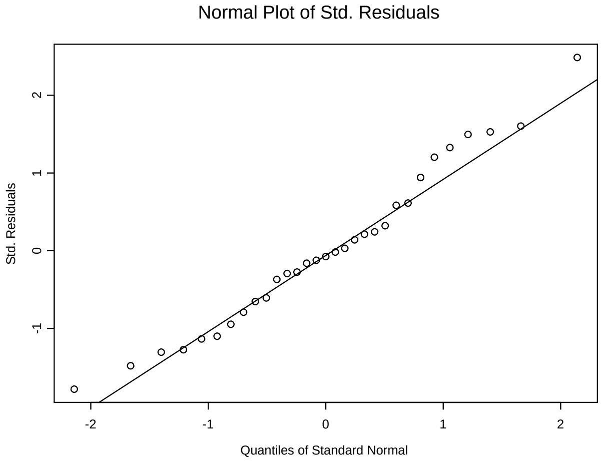

Figure 3: Normal plot of standardized residuals.

The plots involving Diameter and the fitted values both show some evidence that a quadratic term may be needed in the model. The interpretation of the normal probability plot is often unclear, particularly with small samples. Examination of the plot indicates that the residuals show little departure from normality.

The hat matrix, $\mathbf{H}$, is also helpful in identifying strange or peculiar data points, those having unusually large potential effect on the regression. Such points are indicated by relatively high values in the appropriate position in the diagonal of $\mathbf{H}$. The maximum value of any diagonal element is 1. Technically these points are referred as having high **leverage** (Figure 4).

```
> h <- lm.influence(trees.fit)$hat
> h
[1] 0.11582883 0.14720958 0.17686186 0.05919131
 [5] 0.12066468 0.15575111 0.11480262 0.05148096
 [9] 0.09200658 0.04797237 0.07382512 0.04809206
[13] 0.04809206 0.07275901 0.03764563 0.03566543
[17] 0.13130916 0.14346152 0.06665975 0.21123665
[21] 0.03580935 0.04541796 0.04994875 0.11142518
[25] 0.06930648 0.08841762 0.09603041 0.10641665
[29] 0.10982638 0.10982638 0.22705852
> plot(1:31,h, type="n", xlab="l", ylab="Diagonal of Hat Matrix")
> abline(h=mean(h))
> segments(1:31, h, 1:31, mean(h))
```

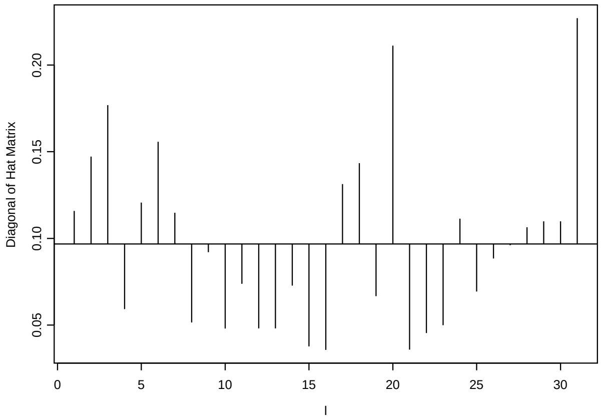

Figure 4: Leverage plot.

Here there seem to be no obvious problem points which might be unduly affecting the estimation process. The leverage values are relatively low.

Returning now to the evidence from the residual plots, a new model involving a quadratic term in Diameter might now be considered.

```
> trees1.fit <- lm(Volume~Diameter+I(Diameter*Diameter)+Height, trees)
> trees1.fit
Call:
lm(formula = Volume ~ Diameter + I(Diameter * Diameter) + Height, data = trees)

Coefficients:
 (Intercept)  Diameter I(Diameter * Diameter)    Height
   -9.920406 -2.885079              0.2686224 0.3763873

Degrees of freedom: 31 total; 27 residual
Residual standard error: 2.624753
> summary(trees1.fit)

Call: lm(formula = Volume ~ Diameter + I(Diameter * Diameter) + Height, data = trees)
Residuals:
    Min     1Q  Median    3Q   Max
 -4.293 -1.669 -0.1018 1.785 4.349

Coefficients:
                          Value Std. Error  t value Pr(>|t|)
           (Intercept)  -9.9204  10.0791    -0.9843   0.3337
              Diameter  -2.8851   1.3099    -2.2026   0.0363
I(Diameter * Diameter)   0.2686   0.0459     5.8517   0.0000
                Height   0.3764   0.0882     4.2659   0.0002

Residual standard error: 2.625 on 27 degrees of freedom
Multiple R-Squared: 0.9771
F-statistic: 383.2 on 3 and 27 degrees of freedom, the p-value is 0

Correlation of Coefficients:
                       (Intercept) Diameter I(Diameter * Diameter)
              Diameter -0.7924
I(Diameter * Diameter)  0.8150     -0.9907
                Height -0.4816     -0.1419   0.0719
```

The new residual plots are not shown but they can be obtained as before. It turns out that the plot involving diameter has now lost the structure it had previously and the other plots look satisfactory.

Although the results indicate that the regression coefficients of both height and diameter are significantly different from zero, it is often useful to explore a number of models in an attempt to find the simplest and adequately describe the data. Essentially, this involves adding or deleting terms from an existing model and assessing the effect of the change.

For example, for the models which involves diameter and height we get by deleting one variable at a time:

```
> attach(trees)
> trees.drop1 <- drop1(trees.fit)   #dropping terms sequentially
> trees.drop1
Single term deletions

Model:
Volume ~ Diameter + Height
         Df Sum of Sq      RSS       Cp
  <none>               421.921  512.333
Diameter  1  4782.974 5204.895 5265.169
  Height  1   102.381  524.303  584.577
```

Sums of squares due to the deleted terms and residuals sums of squares for the reduced model are given. Here their value indicate the great importance of diameter in the model. The opposite approach starts from a model and adds on terms.

```
> trees0.fit <- lm(Volume~1)  #model with constant term
> trees.add1 <- add1(trees0.fit,~ Height+Diameter) #add sequentially terms
> trees.add1
Single term additions

Model:
Volume ~ 1
         Df Sum of Sq      RSS       Cp
  <none>              8106.084 8646.489
  Height  1  2901.189 5204.895 6285.706
Diameter  1  7581.781  524.303 1605.114
```

The results are shown above. The terms are the same as before, but now correspond to adding particular variables. As a final note the function `predict()` can be used to obtain predicted values for a model.