STAT 210
Applied Statistics and Data Analysis
Week 10 - Summary

Joaquin Ortega

King Abdullah University of Science and Technology

V33 - Multiple Regression

Multiple Linear Regression

The basic hypothesis of the model is that

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p + \epsilon$$

where

- y and ϵ are random variables,
- $x_1, \ldots, x_p$ are the regressors or explanatory or independent variables,
- ϵ includes all random factors, and
- $\beta_0, \ldots, \beta_p$ are the regression coefficients we want to estimate.

β_j represents the increase in the response y when x_j increases one unit.

Multiple Linear Regression

We can write the model in matrix notation as

$$\mathbf{Y} = \mathbf{X}\beta + \boldsymbol{\epsilon}$$

where

$$\mathbf{Y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}; \quad \mathbf{X} = \begin{pmatrix} 1 & x_{11} & \cdots & x_{1p} \\ 1 & x_{21} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & \cdots & x_{np} \end{pmatrix}; \quad \boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix}; \quad \boldsymbol{\epsilon} = \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{pmatrix}$$

and the dimensions are

$$\mathbf{Y} : (n \times 1) \quad \mathbf{X} : (n \times (p+1)) \quad \boldsymbol{\beta} : ((p+1) \times 1) \quad \boldsymbol{\epsilon} : (n \times 1)$$

Multiple Linear Regression

We make the following hypothesis on ϵ.

i) $\epsilon_i \sim N(0, \sigma^2)$.

ii) $Cov(\epsilon_i \epsilon_j) = E(\epsilon_i \epsilon_j) = 0$ for all $i \neq j$.

This means that $\boldsymbol{\epsilon} \sim N_n(0, \sigma^2 I_n)$.

We also have the following hypothesis on the regressors

iii) The sample size n is bigger than or equal to $p + 1$, i.e., we have enough data to estimate the $p + 1$ parameters.

iv) Regressors are linearly independent, i.e., none of them is entirely determined by the rest.

Multiple Linear Regression

These hypotheses imply the following for the dependent variable:

i') $E(Y|X_1, \ldots, X_p) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$.

ii') $Var(Y|X_1, \ldots, X_p) = \sigma^2$.

iii') The Y_i's are not correlated.

iv') **Y** has a normal distribution, and its components are independent.

Estimation

Estimation

We fit the model by the method of least squares that is equivalent to maximum likelihood under the hypothesis of normality.

As in the case of simple regression, the normal equation is

$$\mathbf{X}'\mathbf{X}\hat{\beta} = \mathbf{X}'\mathbf{Y}$$

with solution

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}.$$

Estimation

The unbiased estimator for the variance σ^2 of the errors is $\sum_i \hat{\epsilon}_i^2$ divided by the corresponding number of degrees of freedom.

From the normal equation, we have that

$$\mathbf{X}'(\mathbf{X}\hat{\beta} - \mathbf{Y}) = 0$$

and since $\mathbf{X}\hat{\beta}$ is the vector of fitted values, this is equivalent to

$$\mathbf{X}'\hat{\epsilon} = 0$$

so we have $p + 1$ restrictions.

Since there are n data points, the sum of squared residuals has $n - p - 1$ degrees of freedom.

Estimation

The unbiased estimator for σ^2 is then

$$MSE = \frac{SSE}{n-p-1} = \frac{1}{n-p-1}\sum_{i=1}^{n}\hat{\epsilon}_i^2.$$

We have that

$$\frac{\sum \hat{\epsilon}_i^2}{\sigma^2} \sim \chi^2_{n-p-1}.$$

Geometric Interpretation

Consider vectors $\mathbf{1}, \mathbf{X}_1, \ldots, \mathbf{X}_p$, columns of matrix $\mathbf{X}$. The objective of the estimation is to determine a linear combination of these vectors

$$\hat{\mathbf{Y}} = \beta_0 \mathbf{1} + \beta_1 \mathbf{X}_1 + \cdots + \beta_p \mathbf{X}_p$$

so that the norm of $\hat{\epsilon} = \mathbf{Y} - \hat{\mathbf{Y}}$ is minimal.

$\hat{\mathbf{Y}}$ is the projection of vector $\mathbf{Y}$ on the subspace generated by $(\mathbf{1}, \mathbf{X}_1, \ldots, \mathbf{X}_p)$ and $\hat{\epsilon} = \mathbf{Y} - \hat{\mathbf{Y}}$ must be orthogonal to any vector of this subspace, in particular to the generators, that is,

$$\mathbf{1}'\hat{\epsilon} = \mathbf{X}_1'\hat{\epsilon} = \cdots = \mathbf{X}_p'\hat{\epsilon} = 0$$

or in matrix notation

$$\mathbf{X}'\hat{\epsilon} = 0.$$

Geometric Interpretation

This is

$$\mathbf{X}'(\mathbf{Y} - \mathbf{X}\hat{\beta}) = 0 \quad \Rightarrow \quad \mathbf{X}'\mathbf{Y} = \mathbf{X}'\mathbf{X}\hat{\beta}$$

which is the normal equation.

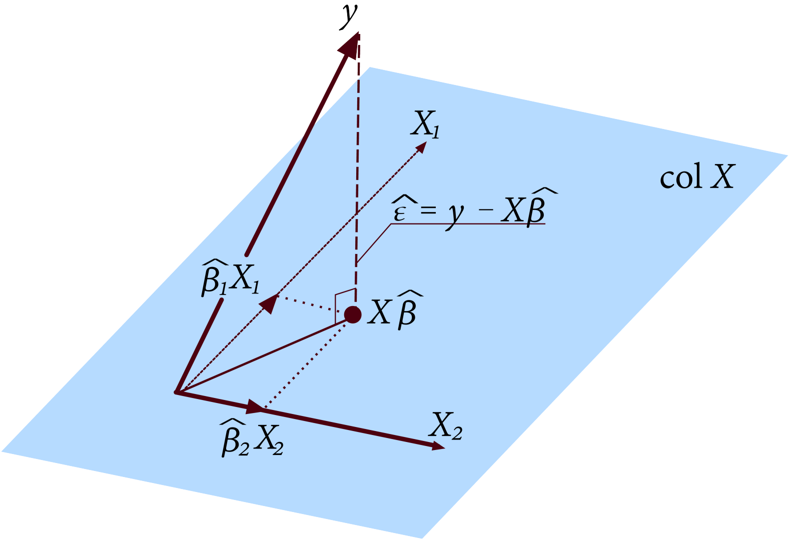
By orthogonality

$$||\mathbf{Y}||^2 = ||\hat{\mathbf{Y}}||^2 + ||\hat{\boldsymbol{\epsilon}}||^2$$

which is equivalent to

$$\sum y_i^2 = \sum \hat{y}_i^2 + \sum \hat{\epsilon}_i^2.$$

Geometric Interpretation



Maximum Likelihood

Another important estimation method is Maximum Likelihood, which in this case gives the same parameter estimates for the regression model.

Recall that Y has a normal distribution with mean

$$\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

and variance σ^2 and they are independent. In matrix notation:

$$\mathbf{Y} \sim N(\boldsymbol{\beta}_0 + \boldsymbol{\beta}_1 \mathbf{X}_1 + \cdots + \boldsymbol{\beta}_p \mathbf{X}_p, \sigma^2 \mathbf{I}_n) = N(\mathbf{X}\beta, \sigma^2 \mathbf{I}_n)$$

The density function for $\mathbf{Y}$ is

$$\prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} \exp\Big\{ - \frac{(Y_i - (\beta_0 + \beta_1 X_{1i} + \cdots + \beta_p X_{pi}))^2}{2\sigma^2} \Big\} \qquad (1)$$

Maximum Likelihood

After we observe the sample, and we replace the observed values in (1), this becomes a function of the unknown parameters β and σ^2, known as the **likelihood function**.

$$\mathcal{L}(\boldsymbol{\beta}, \sigma^2|\mathbf{X}) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{ -\frac{(Y_i - (\beta_0 + \beta_1 X_{1i} + \cdots + \beta_p X_{pi}))^2}{2\sigma^2} \right\} \tag{2}$$

In the maximum likelihood method, we obtain the estimators by maximizing the likelihood function.

Equivalently, we can obtain the parameters maximizing the natural log of the likelihood, known as the log-likelihood function

$$\ell(\boldsymbol{\beta}, \sigma^2|\mathbf{X}) = \ln \mathcal{L}(\boldsymbol{\beta}, \sigma^2|\mathbf{X})$$

Maximum Likelihood

To get the estimators, we differentiate either of these expressions with respect to the parameters and set the derivatives equal to zero.

The estimator for β is the same we obtained with least squares

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}.$$

The estimator for σ^2 is

$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}\hat{\epsilon}_i^2$$

which is biased.

We prefer to use MSE, since it is unbiased.

Example

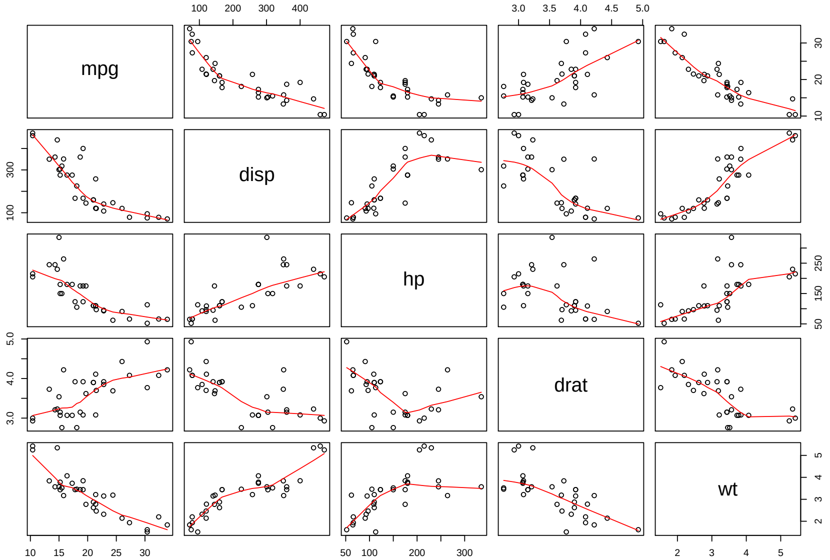
For this example, we use the `mtcars` data set, available in the base package.

For our example, we will only keep

- `mpg`, which measures fuel efficiency and will be the response variable in the model,
- `disp`, the displacement in cubic inches,
- `hp`, the gross horsepower,
- `drat`, the rear axle ratio and
- `wt`, the weight.

Example

```
pairs(data.cars, panel=panel.smooth)
```



Example

The top graphs have mpg on the y axis and show that this variable decreases with disp, hp, and wt but increases with drat.

We fit a model with the four variables using the function lm().

```
model1 <- lm(mpg ~ hp + wt + disp + drat, data = data.cars)
```

Example

```r
summary(model1)
```

```
##
## Call:
## lm(formula = mpg ~ hp + wt + disp + drat, data = data.cars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.5077 -1.9052 -0.5057  0.9821  5.6883
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 29.148738   6.293588   4.631  8.2e-05 ***
## hp          -0.034784   0.011597  -2.999  0.00576 **
## wt          -3.479668   1.078371  -3.227  0.00327 **
## disp         0.003815   0.010805   0.353  0.72675
## drat         1.768049   1.319779   1.340  0.19153
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.602 on 27 degrees of freedom
## Multiple R-squared:  0.8376, Adjusted R-squared:  0.8136
## F-statistic: 34.82 on 4 and 27 DF,  p-value: 2.704e-10
```

Example

A fundamental principle in statistical modeling is parsimony, also known as Occam's razor.

As Einstein said, 'Everything should be made as simple as possible, but not simpler'.

In this spirit, we could try a simpler model excluding the variable `disp`.

Example

```r
model2 <- lm(mpg ~ hp + wt + drat, data = data.cars)
summary(model2)
```

```
##
## Call:
## lm(formula = mpg ~ hp + wt + drat, data = data.cars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.3598 -1.8374 -0.5099  0.9681  5.7078
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 29.394934   6.156303   4.775 5.13e-05 ***
## hp          -0.032230   0.008925  -3.611 0.001178 **
## wt          -3.227954   0.796398  -4.053 0.000364 ***
## drat         1.615049   1.226983   1.316 0.198755
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.561 on 28 degrees of freedom
## Multiple R-squared:  0.8369, Adjusted R-squared:  0.8194
## F-statistic: 47.88 on 3 and 28 DF,  p-value: 3.768e-11
```

Example

We see that the R^2 has decreased marginally while the adjusted R^2, which considers the number of variables in the model, has increased.

Let us compare the coefficients

```
round(coef(model1),3); round(coef(model2),3)
```

```
## (Intercept)          hp          wt        disp        drat 
##      29.149      -0.035      -3.480       0.004       1.768

## (Intercept)          hp          wt        drat 
##      29.395      -0.032      -3.228       1.615
```

We see that the change is small.

Example

We can also try removing the variable `drat`.

```
model3 <- lm(mpg ~ hp + wt , data = data.cars)
summary(model3)
```

```
##
## Call:
## lm(formula = mpg ~ hp + wt, data = data.cars)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -3.941 -1.600 -0.182  1.050  5.854
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 37.22727    1.59879  23.285  < 2e-16 ***
## hp          -0.03177    0.00903  -3.519  0.00145 **
## wt          -3.87783    0.63273  -6.129 1.12e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.593 on 29 degrees of freedom
## Multiple R-squared:  0.8268, Adjusted R-squared:  0.8148
## F-statistic: 69.21 on 2 and 29 DF,  p-value: 9.109e-12
```

The R^2's have diminished slightly, and the estimated coefficients are similar to those we had before.

Sampling Distribution

Sampling Distribution

We have $\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$.

As in the simple regression case, the expected value is

$$E(\hat{\beta}) = \beta$$

so $\hat{\beta}$ is an unbiassed estimator for β. As for the variance,

$$Var(\hat{\beta}) = \sigma^2(\mathbf{X}'\mathbf{X})^{-1} \tag{3}$$

For each estimated parameter $\hat{\beta}_j$, we have

$$\hat{\beta}_j \sim N(\beta_j, \sigma^2 v_{jj}),$$

where the v_{jj}'s are the diagonal elements of matrix $V = (\mathbf{X}'\mathbf{X})^{-1}$.

In general, the matrix $(\mathbf{X}'\mathbf{X})^{-1}$ is not diagonal and the estimated coefficients $\hat{\beta}$ will not be independent.

The Hat Matrix

Since $\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$ and $\hat{\mathbf{Y}} = \mathbf{X}\hat{\beta}$ we have that

$$\hat{\mathbf{Y}} = \mathbf{X}\hat{\beta} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y} = \mathbf{H}\mathbf{Y}$$

where $\mathbf{H}$ is known as the ‘hat’ matrix.

$\mathbf{H}$ is idempotent ($\mathbf{H}^2 = \mathbf{H}$) and symmetric.

It is a projection matrix that sends points onto the column space of the design matrix.

We have already interpreted the elements of this matrix in terms of ‘leverage’, particularly the diagonal elements, in the simple regression case.

The Hat Matrix

Properties:

1. $h_{ii} = \mathbf{X}_i'(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}_i, \quad h_{ij} = \mathbf{X}_i'(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}_j$ only depend on $\mathbf{X}$, $\mathbf{X}_i'$ is the i-th row of $\mathbf{X}$.

2. We have

$$rank(\mathbf{H}) = rank(\mathbf{X}) = p + 1, \quad tr(\mathbf{H}) = \sum h_{ii} = p + 1$$

3. $\mathbf{HX} = \mathbf{X} \Rightarrow \mathbf{H1} = \mathbf{1} \Rightarrow \sum_{j=1}^{n} h_{ij} = 1.$

4. $\mathbf{H}^2 = \mathbf{H} \Rightarrow h_{ii} = \sum_j h_{ij}^2 \Rightarrow h_{ii} \geq h_{ii}^2 \Rightarrow |h_{ii}| \leq 1.$

5. It is also true that $h_{ij}^2 \leq h_{ii}$. Thus $nh_{ii} \geq 1 \Rightarrow h_{ii} \geq 1/n$ and we get

$$\frac{1}{n} \leq h_{ii} \leq 1$$

The Hat Matrix

6. $Cov(\hat{\mathbf{Y}}) = \sigma^2 \mathbf{H}$ and $Var(\hat{\mathbf{Y}}(\mathbf{X}_i)) = \sigma^2 h_{ii}$. Therefore
$$\frac{1}{n} \leq \frac{Var(\hat{\mathbf{Y}}(\mathbf{X}_i))}{\sigma^2} \leq 1.$$

7. If there are many observations, the h_{ij} are small, and the residuals are practically uncorrelated. On the other hand, if h_{ii} is big, $Var(\hat{\epsilon}_i)$ is small, and the i-th observation will attract the regression line or hyperplane.

The Hat Matrix

As an example we are going to obtain the hat matrix for the model with regressors `hp` and `wt` in the previous example. Remember that this is `model3`.

```r
X <-model.matrix(model3)
n <-nrow(X)
p <-ncol(X)
H <- X%*%solve(t(X)%*%X)%*%t(X)
hii <-diag(H)
```

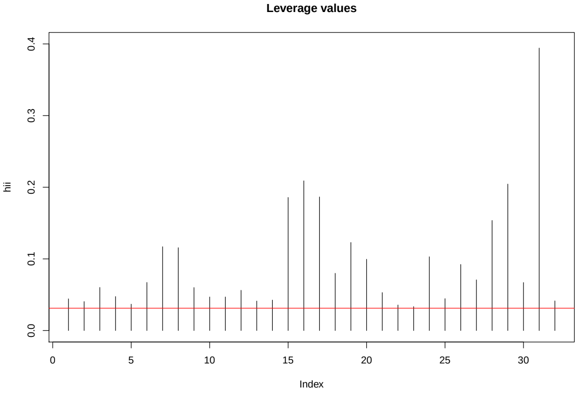
Let us verify that the sum of the diagonal elements of **H** (the trace of **H**) is p:

```r
sum(hii)
```

```
## [1] 3
```

The Hat Matrix

```r
plot(hii,type = 'h', ylim = c(0,.4), main = 'Leverage values')
abline(h=1/32, col = 'red')
```



V34 Multiple Regression 2: Anova and Hypotheses Tests

Anova with Multiple Regression

The Anova calculations for multiple regression are almost identical to those of simple linear regression, except that now the degrees of freedom have to be adjusted, taking into account the number of parameters.

Table 1: Anova table for multiple regression.

Source of	Sum of	Degrees of	Mean	
Variation	Squares	Freedom	Squares	F_{obs}
Regression	$SSR = \hat{\beta}'\mathbf{X}'\mathbf{Y} - \frac{1}{n}\mathbf{Y}'\mathbf{J}\mathbf{Y}$	p	$MSR = \frac{SSR}{p}$	$F = \frac{MSR}{MSE}$
Error	$SSE = \mathbf{Y}'\mathbf{Y} - \hat{\beta}\mathbf{X}'\mathbf{Y}$	$n - p - 1$	$MSE = \frac{SSE}{n-p-1}$	
Total	SST	$n - 1$		

Here, $\mathbf{J}$ is a matrix of 1's.

Anova with Multiple Regression

If we call for the anova table associated with the first regression in the previous video using the anova command in R, we get

```
(model1.anova <- anova(model1))
```

```
## Analysis of Variance Table
##
## Response: mpg
##           Df Sum Sq Mean Sq  F value    Pr(>F)
## hp         1 678.37  678.37 100.1767 1.393e-10 ***
## wt         1 252.63  252.63  37.3059 1.593e-06 ***
## disp       1   0.06    0.06   0.0084    0.9275
## drat       1  12.15   12.15   1.7947    0.1915
## Residuals 27 182.84    6.77
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

and this does not look like table 1.

Anova with Multiple Regression

In this example there are

- $n = 32$ observations,
- 4 independent variables or regressors,
- $p = 5$ coefficients (add one for the intercept),

so $n - p - 1 = 32 - 5 = 27$, which are the degrees of freedom corresponding to `Residuals` in the table.

The sum of squares for the residuals is $SSE = 182.84$, and the mean square error is

$$MSE = \frac{SSE}{n - p - 1} = \frac{182.84}{27} = 6.77.$$

This corresponds to the second line in table 1.

Anova with Multiple Regression

However, first line values, which correspond to the regression sum of squares, are split up into the individual regressors.

We need to add degrees of freedom and sums of squares for the first four rows in our table to get the first row in table 1:

```
(SSR = sum(model1.anova$`Sum Sq`[1:4]))
```

```
## [1] 943.2096
```

and since $p = 4$,

```
(MSR = SSR/4)
```

```
## [1] 235.8024
```

$$MSR = \frac{SSR}{p} = \frac{943.21}{4} = 235.8$$

Anova with Multiple Regression

Finally, F_{obs} is given by

```
(Fobs <- MSR/MSE)
```

```
## [1] 34.82143
```

Observe that this is the same value we get at the bottom line of the summary for the regression:

Anova with Multiple Regression

```
summary(model1)
```

```
##
## Call:
## lm(formula = mpg ~ hp + wt + disp + drat, data = data.cars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.5077 -1.9052 -0.5057  0.9821  5.6883
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 29.148738   6.293588   4.631  8.2e-05 ***
## hp          -0.034784   0.011597  -2.999  0.00576 **
## wt          -3.479668   1.078371  -3.227  0.00327 **
## disp         0.003815   0.010805   0.353  0.72675
## drat         1.768049   1.319779   1.340  0.19153
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.602 on 27 degrees of freedom
## Multiple R-squared:  0.8376, Adjusted R-squared:  0.8136
## F-statistic: 34.82 on 4 and 27 DF,  p-value: 2.704e-10
```

We can obtain the *p*-value with

```
1-pf(Fobs,4,27)
```

```
## [1] 2.70431e-10
```

Coefficient of Determination

Coefficient of Determination

As in the case of simple linear regression, the coefficient of determination R^2 is defined by

$$R^2 = \frac{SSR}{SST} = 1 - \frac{SSE}{SST}$$

and is interpreted as the proportion of the variability that is explained by the regression.

However, adding more variables to the model always increases R^2, even if the new variables do not add anything significant to the model.

Coefficient of Determination

Therefore, we need to adjust this coefficient to consider the number of variables in the model.

The new coefficient is known as the **adjusted coefficient of determination** or **adjusted R squared** and is defined as

$$R_a^2 = 1 - \frac{SSE/(n-p-1)}{SST/(n-1)} = 1 - \frac{MSE}{MST}.$$

As we stated when we introduced R^2 in the case of simple linear regression, caution must be exercised when using this coefficient in the context of model fitting.

Extra Sums of Squares

Extra Sums of Squares

If we add regressors to a model, the sums of squares will change. To assess the contribution of a new variable on an existing model, we look at the change in the sum of squares.

For example, the marginal increase when variable x_2 is added to a model based on variable x_1 is given by

$$SSR(\beta_2|\beta_1, \beta_0) = SSR(\beta_1, \beta_2|\beta_0) - SSR(\beta_1|\beta_0) \tag{4}$$

where

- $SSR(\beta_1, \beta_2|\beta_0)$ is the regression sum of squares for the (complete) model including β_1 and β_2 (β_0 is always included) and
- $SSR(\beta_1|\beta_0)$ is the sum for the model with only x_1 as a regressor.

Extra Sum of Squares

We can turn (4) around and get

$$SSR(\beta_1, \beta_2|\beta_0) = SSR(\beta_1|\beta_0) + SSR(\beta_2|\beta_1, \beta_0) \qquad (5)$$

which can be seen as a decomposition of the regression sum of squares into two parts, the contribution of the simple model with only x_1 as a regressor plus the increase when a second regressor x_2 is added to the model.

Observe, however, that we could also have done this decomposition in a different order

$$SSR(\beta_1, \beta_2|\beta_0) = SSR(\beta_2|\beta_0) + SSR(\beta_1|\beta_2, \beta_0) \qquad (6)$$

and the results may be different.

These are known as **incremental** or **type I** sums of squares.

Extra Sum of Squares

The sums of squares that appear in an anova table obtained with the command `anova` acting on an `lm` object are incremental sums of squares that follow the order set in the model's defining equation.

Thus, if the formula for the model is

$$\texttt{y} \sim \texttt{x1 + x2},$$

the first sum in the table corresponds to $SSR(\beta_1|\beta_0)$ and the second to $SSR(\beta_2|\beta_1, \beta_0)$.

As an example, let us look at the third model fitted to the `mtcars` data. There we had `mpg ~ hp + wt` and we will produce the anova tables for the two possible orders for the regressors:

Extra Sum of Squares

```
## Analysis of Variance Table
##
## Response: mpg
##           Df Sum Sq Mean Sq F value    Pr(>F)
## hp         1 678.37  678.37 100.862 5.987e-11 ***
## wt         1 252.63  252.63  37.561 1.120e-06 ***
## Residuals 29 195.05    6.73
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## Analysis of Variance Table
##
## Response: mpg
##           Df Sum Sq Mean Sq F value    Pr(>F)
## wt         1 847.73  847.73 126.041 4.488e-12 ***
## hp         1  83.27   83.27  12.381  0.001451 **
## Residuals 29 195.05    6.73
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Extra Sum of Squares

The sum of squares for the residuals are the same, as are the degrees of freedom appearing in the table. However, the sums of squares for the regressors are different. From these tables we have

$$\begin{aligned} SSR(hp|\beta_0) &= 678.37 \qquad & SSR(wt|\beta_0) &= 847.73 \\ SSR(wt|hp,\beta_0) &= 252.63 \qquad & SSR(hp|wt,\beta_0) &= 83.27 \end{aligned}$$

Observe that the column sums are equal:

$$678.37 + 252.63 = 847.73 + 83.27 = 931$$

which corresponds to the sum of squares for the regression SSR.

Confidence Intervals

Regression Coefficients

We have assumed the errors to have a standard normal distribution, therefore $\hat{\beta}_j \sim N(\beta_j, \sigma^2 v_{jj})$ and

$$\frac{\hat{\beta}_j - \beta_j}{\sigma\sqrt{v_{jj}}} \sim N(0, 1).$$

σ is unknown but $(n - p - 1)MSE/\sigma^2 \sim \chi^2_{n-p-1}$, and therefore

$$\frac{\hat{\beta}_j - \beta_j}{\sqrt{MSEv_{jj}}} \sim t_{n-p-1}.$$

Hence, the α-level confidence interval is

$$\hat{\beta}_j \pm t_{n-p-1,1-\frac{\alpha}{2}}(MSEv_{jj})^{1/2}. \tag{7}$$

Confidence Intervals

Mean value at a point

The confidence interval at level α is

$$\hat{y}(\mathbf{z}) \pm t_{n-p-1,1-\frac{\alpha}{2}}(MSE(\mathbf{z}'(\mathbf{X}'\mathbf{X})^{-1}\mathbf{z})^{1/2}. \tag{8}$$

Predicted value at a point

The prediction interval at level α is

$$\hat{y}(\mathbf{z}) \pm t_{n-p-1,1-\frac{\alpha}{2}}\big(MSE(1+\mathbf{z}'(\mathbf{X}'\mathbf{X})^{-1}\mathbf{z})\big)^{1/2}. \tag{9}$$

Hypotheses Tests

Significance Test

This test is done to determine if the regression is significant, i.e., if the regressor variables contribute at all to explain Y:

$$H_0: \ \beta_1 = \beta_2 = \cdots = \beta_p = 0 \quad vs. \quad H_A: \ \text{at least one } \beta_i \neq 0$$

For this test, we divide the total variability of the problem

$$SST = \sum_{i=1}^{n} (y_i - \bar{y})^2$$

into the part that is explained by the model and the part that remains unexplained:

$$SST = SSE + SSR.$$

Significance Test

We know

$$\frac{SSE}{\sigma^2} \sim \chi^2_{n-p-1}.$$

It is also possible to prove that

$$\frac{SSR}{\sigma^2} \sim \chi^2_p \quad \text{under } H_0.$$

Since *SSE* and *SSR* turn out to be independent, the test statistic

$$F_0 = \frac{SSR/p}{SSE/n-p-1} \sim F_{p,n-p-1} \qquad \text{under } H_0.$$

Test on Individual Parameters

Test on Individual Parameters

The hypotheses for the test of significance of the parameter β_j is

$$H_0: \ \beta_j = 0 \qquad vs \qquad H_1: \ \beta_j \neq 0.$$

We know that $\hat{\beta}_j \sim N(\beta_j, \sigma^2 v_{jj})$. Under H_0

$$\frac{\hat{\beta}_j}{\sqrt{MSEv_{jj}}} \sim t_{n-p-1}.$$

Since this is a bilateral test, the rejection region is

$$\Big\{ \frac{|\hat{\beta}_j|}{\sqrt{MSEv_{jj}}} > t_{n-p-1,1-\frac{\alpha}{2}} \Big\}.$$

Tests on Subsets of Parameters

Tests on Subsets of Parameters

These contrasts look at the contribution of variables $x_{r+1}, \ldots, x_p$ to the model based on $x_1, \ldots, x_r$.
β_0 is not included in the hypothesis since it will always be in the model:

$$H_0: \ \beta_{r+1} = \cdots = \beta_p = 0 \quad vs. \quad H_1: \text{ at least one } \beta_j \neq 0, \ r+1 \leq j \leq p.$$

Fit the model $Y = \beta_0 + \beta_1 x_1 + \cdots + \beta_r x_r + \epsilon$ with which we measure the contribution of $x_1, \ldots, x_r$ when they are alone, and then fit the complete model and compare to see the contribution of all the variables.

Tests on Subsets of Parameters

In the first model $SSR(\beta_1, \ldots, \beta_r|\beta_0)$ has r degrees of freedom and in the second $SSR(\beta_1, \ldots, \beta_p|\beta_0)$ has p degrees of freedom.

The difference between these two quantities is the contribution of $x_{r+1}, \ldots, x_p$ when $x_1, \ldots, x_r$, are in the model, which is written as

$$SSR(\beta_{r+1}, \ldots, \beta_p|\beta_0) = SSR(\beta_1, \ldots, \beta_p|\beta_0) - SSR(\beta_1, \ldots, \beta_r|\beta_0)$$

and has $p - r$ degrees of freedom. We use the statistic

$$F_0 = \frac{SSR(\beta_{r+1}, \ldots, \beta_p|\beta_0)/p - r}{SSR(\beta_1, \ldots, \beta_p|\beta_0)/n - p - 1} \sim F_{p-r,n-p-1}.$$

for this test.

Hypotheses Tests: Example

As an example, let us consider the `mtcars` data again.

Our previous analysis saw that variables `disp` and `drat` did not seem to contribute much to the model. Let us test whether these two variables can be dropped from the model.

`model1` is the full model, and variables `hp` and `wt` were introduced as the first and second term in the regression equation.

The anova table for this model was calculated previously and is in `model1.anova`.

Hypotheses Tests: Example

```
model1.anova
```

```
## Analysis of Variance Table
##
## Response: mpg
##           Df Sum Sq Mean Sq  F value    Pr(>F)
## hp         1 678.37  678.37 100.1767 1.393e-10 ***
## wt         1 252.63  252.63  37.3059 1.593e-06 ***
## disp       1   0.06    0.06   0.0084    0.9275
## drat       1  12.15   12.15   1.7947    0.1915
## Residuals 27 182.84    6.77
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' '
```

Hypotheses Tests: Example

We calculate now the anova table for the reduced model

```
(model3.anova <- anova(model3))
```

```
## Analysis of Variance Table
##
## Response: mpg
##           Df Sum Sq Mean Sq F value    Pr(>F)
## hp         1 678.37  678.37 100.862 5.987e-11 ***
## wt         1 252.63  252.63  37.561 1.120e-06 ***
## Residuals 29 195.05    6.73
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' '
```

Hypotheses Tests: Example

This is what we get when we do an anova to compare the two models:

```
anova(model3,model1)
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ hp + wt
## Model 2: mpg ~ hp + wt + disp + drat
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     29 195.05
## 2     27 182.84  2     12.21 0.9016 0.4178
```

Residual Analysis

Residual Analysis

1.- To verify the hypothesis of normality of the residuals, quantile plots of normality are made. If normality does not hold, transformations can be made to achieve it, but the linearity or homogeneity of the variance can be lost. The hypothesis of normality may not be too critical.

2.- If time is a regressor or measurements are taken periodically, a graph of the residuals against time can be made. In this graph, we can see if there is any tendency that can make us reject the hypothesis of independence.

3.-It is also useful to graph the residuals against the regressors or against the fitted values, to discover possible trends that indicate the need to introduce higher-order polynomial terms or make transformations of the variables. Usually, graphs are made of the residuals divided by their (sample) standard deviation to normalize them.

Properties of Residuals

1. $\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$.
2. $\hat{\mathbf{Y}} = \mathbf{H}\mathbf{Y}$ with $\mathbf{H} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$.
3. $\mathbf{H}$ is the projection matrix on the subspace generated by the columns of the design matrix $\mathbf{X}$: $\mathbf{1}, \mathbf{X}_1, \ldots, \mathbf{X}_p$.
4. $\hat{\epsilon} = \mathbf{Y} - \hat{\mathbf{Y}} = (\mathbf{I} - \mathbf{H})\mathbf{Y}$.
5. $\mathbf{I} - \mathbf{H}$ is the projection matrix on the subspace orthogonal to $\langle \mathbf{1}, \mathbf{X}_1, \ldots, \mathbf{X}_p \rangle$.

Properties of Residuals

6. $\mathbf{H}$ and $\mathbf{I}-\mathbf{H}$ are idempotent and symmetrical: $\mathbf{H}^2=\mathbf{H}$, $(\mathbf{I}-\mathbf{H})^2=\mathbf{I}-\mathbf{H}$.
7. $E(\hat{\epsilon})=(\mathbf{I}-\mathbf{H})E(\mathbf{Y})=(\mathbf{I}-\mathbf{H})\mathbf{X}\beta=\mathbf{X}\beta-\mathbf{X}\beta=\mathbf{0}$.
8. $Cov(\hat{\boldsymbol{\epsilon}})=(\mathbf{I}-\mathbf{H})\sigma^2\mathbf{I}(\mathbf{I}-\mathbf{H})=\sigma^2(\mathbf{I}-\mathbf{H})$. Therefore

$$\begin{gathered} E(\hat{\epsilon}_i)=0 \\ Var(\hat{\epsilon}_i)=\sigma^2(1-h_{ii}), \\ Cov(\hat{\epsilon}_i,\hat{\epsilon}_j)=-\sigma^2 h_{ij} \end{gathered}$$

Properties of Residuals

Unlike the ϵ_i, the $\hat{\epsilon}_i$ do not have constant variance and are correlated.

We have

$$E\left(\frac{\hat{\epsilon}_i}{\sigma\sqrt{1-h_{ii}}}\right)=0, \qquad Var\left(\frac{\hat{\epsilon}_i}{\sigma\sqrt{1-h_{ii}}}\right)=1$$

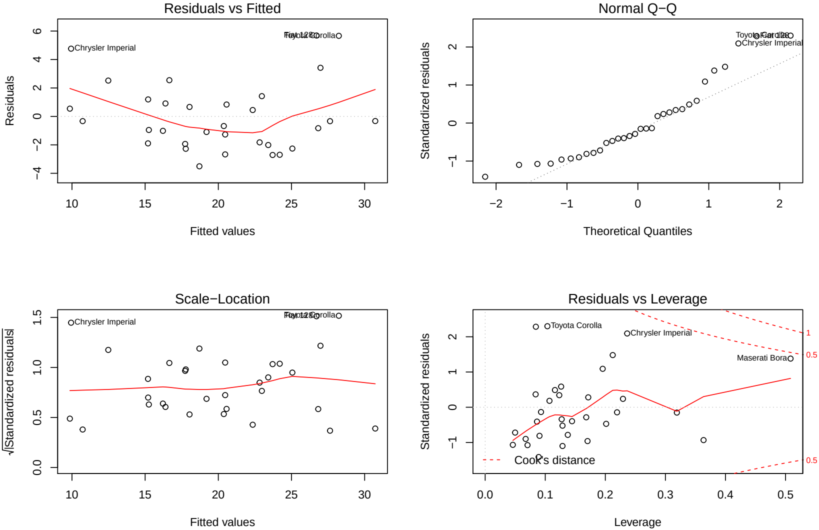
Since σ^2 is unknown, to standardize the residuals they are divided by the empirical standard deviation

$$r_i=\frac{\hat{\epsilon}_i}{\sqrt{MSE(1-h_{ii})}}. \tag{10}$$

Standardized residuals are also known as internally studentized residuals.

Properties of Residuals

Residual graphs for some of the models fitted for the `mtcars` data set. We start with the complete model



Residuals vs Fitted

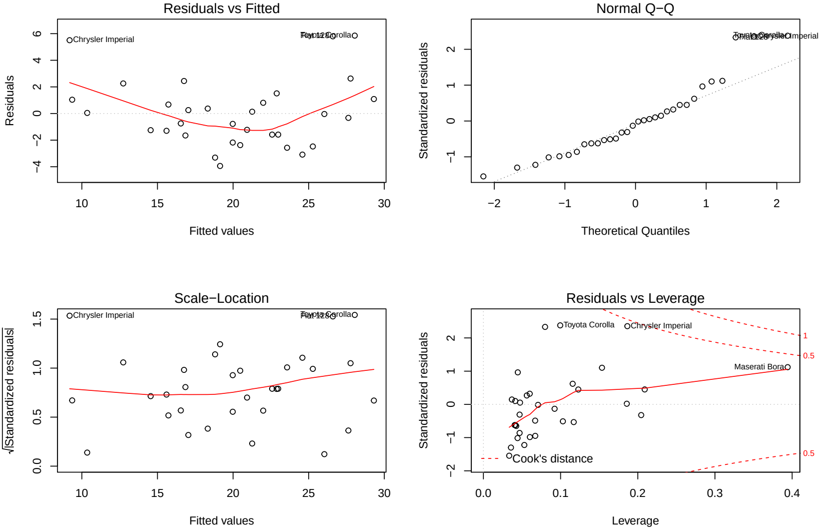
Normal Q−Q

Scale−Location

Residuals vs Leverage

Properties of Residuals

and then plot the reduced model with only two variables: `model3`.



Properties of Residuals

We see that all graphs are very similar and reasonably good.

The graphs of residuals vs. fitted values have some curvature that suggests exploring a higher-order model. Still, the graphs of standardized residuals vs. fitted values do not show important tendencies, so that the curvature may be due to differences in variance.

The right tail in the normal quantile plot shows some points distant from the reference line.

The `R` function `rstandard()` computes standardized residuals according to (10). The function `stdres()` in the MASS package also computes standardized residuals.

V35 Multiple Regression 3: Model Selection

Model Selection

(This section follows closely Ugarte, Militino and Arnholt, Probability and Statistics with R, Chapman and Hall, 2008)

In this section, we discuss several general methods for model selection.

We consider two approaches for selecting variables:

1. a stepwise testing strategy that compares successive models
2. a criterion approach that attempts to maximize some measure of goodness-of-fit.

Procedures based on testing

Backward Elimination

Backward elimination begins with a model containing all potential regressors and identifies the one with the largest *p*-value.

This can be done by looking at the *p*-values for the *t* tests of the $\hat{\beta}_i, i = 1, \ldots, p$ using the function `summary()` or using the *p*-values from the `R` function `drop1()`.

If the variable with the largest *p*-value is above a predetermined value, α_{crit}, that regressor is dropped.

A model with the remaining *x*-variables is then fitted, and the procedure continues until all the *p*-values for the remaining variables in the model are below the predetermined α_{crit}.

α_{crit} is sometimes referred to as the '*p*-to-remove' and is typically set to 15 or 20%.

Forward Selection

Forward selection starts with no variables in the model and then adds the regressor that produces the smallest *p*-value below α_{crit} when included in the model.

This procedure is continued until no new predictors can be added.

The user can determine the variable that produces the smallest *p*-value by regressing the response variable on the x's one at a time using `lm()` and `summary()` or by using the function `add1()`.

Stepwise Regression

Stepwise regression is a combination of backward elimination and forward selection.

This technique allows variables that were either removed or added early to reenter or exit the model later in the process.

At each stage, a variable may be added or removed.

Criterion-Based Procedures

Criterion-Based Procedures

There are several well-defined optimality criteria used in model building including

- R_a^2 (adjusted R^2),
- Mallows' C_p,
- Bayes Information Criterion (BIC),
- Akaike Information Criterion (AIC).

Adjusted R^2

R_a^2 is used instead of R^2 since R^2 will always increase with the addition of new variables to the model. Recall that

$$R_a^2 = 1 - \frac{SSE/(n-p)}{SST/(n-1)}.$$

Mallows C_p

Mallows C_p statistic is a measure of the total mean square error for the model. Consider a model with p parameters and define

$$
\begin{aligned}
\Gamma_p &= \frac{1}{\sigma^2}\sum_{i=1}^{n} E(\hat{y}_i - E(y_i))^2 \\
&= \frac{1}{\sigma^2}\left(\sum_{i=1}^{n}\left(E(y_i) - E(\hat{y}_i)\right)^2 + \sum_{i=1}^{n} Var(\hat{y}_i)\right) \\
&= \frac{1}{\sigma^2}\left((bias)^2 + variance\right).
\end{aligned}
$$

This is the mean square prediction error. It will not necessarily get smaller if more terms are added.

Mallows C_p

Using the MSE for the complete model to estimate σ^2, $\hat{\sigma}^2 = MSE$, we get as estimate of Γ_p

$$C_p = \frac{SSE(p)}{\hat{\sigma}^2} - n + 2p$$

which is the C_p statistic. If the model with p terms has a small bias, it can be shown that

$$E(C_p | null\ bias) \simeq p$$

When all p parameters are used in the model, $C_p = p$. A model with a bad fit will produce a C_p much bigger than p. Desirable models have small p and C_p less than or equal to p. It is common practice to plot C_p against p.

AIC and BIC

Recall that $\ln L(\beta, \sigma^2|\mathbf{X})$ is the log-likelihood function. The *BIC* for linear regression models is defined as

$$\begin{aligned} BIC &= -2\max(\ln L(\beta, \sigma^2|\mathbf{X})) + p\ln(n) \\ &= n\ln(SSE/n) + p\ln(n) + constant \end{aligned}$$

while the *AIC* for linear regression models is defined as

$$\begin{aligned} AIC &= -2\max(\ln L(\beta, \sigma^2|\mathbf{X})) + 2p \\ &= n\ln(SSE/n) + 2p + constant \end{aligned}$$

Since the constant is the same for a given data set and error distribution, it can be ignored when comparing models based on the same data. This is what the function `stepAIC()` does.

AIC and BIC

The goal when using BIC or AIC is to create a model that minimizes either BIC or AIC. Both AIC and BIC search for models that have small SSE.

However, BIC penalizes larger models more so than does AIC (assuming $n > e^2 = 7.39$). Consequently, BIC will favor smaller models than will AIC.

When building a model to be used for predictive purposes, AIC will generally be favored over BIC.

In `R`, the package `leaps` contains the function `regsubsets()`, which is very useful for computing R_a^2 and Mallows's C_p.

Example

Example

The data frame HSwrestler contains information on nine variables for a group of 78 high school wrestlers that was collected by the human performance lab at Appalachian State University. The variables are

- ‘AGE‘ (in years),
- ‘HT‘ (height in inches),
- ‘WT‘ (weight in pounds),
- ‘ABS‘ (abdominal skinfold measure),
- ‘TRICEPS‘ (tricep skinfold measure),
- ‘SUBSCAP‘ (subscapular skinfold measure),
- ‘HWFAT‘ (hydrostatic determination of fat),
- ‘TANFAT‘ (Tanita determination of fat), and
- ‘SKFAT‘ (skinfold determination of fat).

Example

In this example we want to create a model for predicting wrestlers' hydrostatic fat (`HWFAT`).

(a) Use backward elimination with the predictors `AGE`, `HT`, `WT`, `ABS`, `TRICEPS`, and `SUBSCAP` and an α_{crit} of 0.20.

(b) Use forward selection with an α_{crit} of 0.20.

(c) Use the function `regsubsets` in the `R` package `leaps` to select a model using R_a^2 as the criterion.

(d) Use the function `regsubsets` in the `R` package `leaps` to select a model using Mallows's C_p as the criterion.

(e) Use *AIC* as the criterion for selecting a model.

(f) Use *BIC* as the criterion for selecting a model.

Example: Backward elimination

- Backward elimination starts with all the variables in the model and eliminates variables with the largest (least significant) *p*-values:

```r
library(PASWR)
attach(HSwrestler)
str(HSwrestler)
```

```
## 'data.frame':    78 obs. of  9 variables:
##  $ AGE    : int  18 15 17 17 17 14 14 17 15 14 ...
##  $ HT     : num  65.8 65.5 64 72 69.5 ...
##  $ WT     : num  134 129 121 145 299 ...
##  $ ABS    : num  8 10 6 11 54 40 6 11 9 19 ...
##  $ TRICEPS: num  6 8 6 10 42 25 8 7 6 13 ...
##  $ SUBSCAP: num  10.5 9 8 10 37 26 7 8 8 11.5 ...
##  $ HWFAT  : num  10.71 8.53 6.78 9.32 41.89 ...
##  $ TANFAT : num  11.9 10 8.3 8.2 41.6 29.9 12.4 11.1 10.1 15.5 ...
##  $ SKFAT  : num  9.8 10.56 8.43 11.77 41.09 ...
```

Example: Backward elimination

The functions `summary()` or `drop1()` can be used in R:

```
round(drop1(lm(HWFAT ~ AGE + HT +WT + ABS +TRICEPS +SUBSCAP),
      test="F"),3)
```

```
## Single term deletions
##
## Model:
## HWFAT ~ AGE + HT + WT + ABS + TRICEPS + SUBSCAP
##         Df Sum of Sq    RSS    AIC F value Pr(>F)
## <none>                651.05 179.51
## AGE      1     9.594 660.64 178.65   1.046  0.310
## HT       1     1.613 652.66 177.70   0.176  0.676
## WT       1     2.546 653.60 177.81   0.278  0.600
## ABS      1   162.000 813.05 194.84  17.667 <2e-16 ***
## TRICEPS  1    72.683 723.73 185.76   7.926  0.006 **
## SUBSCAP  1     5.921 656.97 178.21   0.646  0.424
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Example: Backward elimination

Note that HT has the largest *p*-value of 0.676, so it is eliminated from the model:

```r
round(drop1(lm(HWFAT ~ AGE + WT + ABS +TRICEPS +SUBSCAP),
      test="F"),3)
```

```
## Single term deletions
##
## Model:
## HWFAT ~ AGE + WT + ABS + TRICEPS + SUBSCAP
##         Df Sum of Sq    RSS    AIC F value Pr(>F)
## <none>               652.66 177.70
## AGE      1     9.875 662.54 176.87   1.089  0.300
## WT       1    10.554 663.22 176.95   1.164  0.284
## ABS      1   189.072 841.74 195.54  20.858 <2e-16 ***
## TRICEPS  1    78.809 731.47 184.59   8.694  0.004 **
## SUBSCAP  1     5.693 658.36 176.38   0.628  0.431
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Example: Backward elimination

Note that SUBSCAP has the largest *p*-value of 0.431, so it is eliminated from the model:

```r
round(drop1(lm(HWFAT ~ AGE + WT + ABS +TRICEPS),
      test="F"),3)
```

```
## Single term deletions
##
## Model:
## HWFAT ~ AGE + WT + ABS + TRICEPS
##         Df Sum of Sq    RSS    AIC F value Pr(>F)
## <none>                658.36 176.38
## AGE      1    13.615 671.97 175.97   1.510  0.223
## WT       1     6.833 665.19 175.18   0.758  0.387
## ABS      1   220.994 879.35 196.95  24.504 <2e-16 ***
## TRICEPS  1   201.768 860.12 195.23  22.373 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Example: Backward elimination

Note that `WT` has the largest *p*-value of `0.387`, so it is eliminated from the model:

```r
round(drop1(lm(HWFAT ~ AGE + ABS +TRICEPS),
      test="F"),3)
```

```
## Single term deletions
##
## Model:
## HWFAT ~ AGE + ABS + TRICEPS
##         Df Sum of Sq    RSS    AIC F value Pr(>F)
## <none>                665.19 175.18
## AGE      1    37.595 702.78 177.47   4.182  0.044 *
## ABS      1   282.896 948.09 200.82  31.471 <2e-16 ***
## TRICEPS  1   198.891 864.08 193.59  22.126 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Example: Backward elimination

The resulting model uses `AGE`, `ABS`, and `TRICEPS` to predict `HWFAT`.

Example: Forward selection

Forward selection assumes a model with an intercept only and adds the most significant (smallest *p*-values) variables one at a time.

The function `add1()` in `R` is used as the *p*-values at each stage are shown:

```
round(add1(lm(HWFAT~1), scope=(~.+ AGE + HT + WT + ABS +
                            TRICEPS + SUBSCAP), test="F"),3)
```

```
## Single term additions
##
## Model:
## HWFAT ~ 1
##         Df Sum of Sq    RSS    AIC F value Pr(>F)
## <none>               6017.8 340.97
## AGE      1     175.0 5842.8 340.67   2.277  0.135
## HT       1     117.8 5900.0 341.43   1.517  0.222
## WT       1    3237.6 2780.2 282.74  88.505 <2e-16 ***
## ABS      1    5072.8  945.0 198.57 407.993 <2e-16 ***
## TRICEPS  1    5056.3  961.5 199.92 399.646 <2e-16 ***
## SUBSCAP  1    4939.0 1078.8 208.90 347.946 <2e-16 ***
```

Example: Forward selection

The variable `ABS` has the most significant (smallest) p-value = `2.2e-16` with the largest `F` value = `407.993`, so it is added to the model:

```r
round(add1(lm(HWFAT~ABS),
           scope=(~.+AGE +HT +WT +TRICEPS +SUBSCAP), test="F"),3)
```

```
## Single term additions
##
## Model:
## HWFAT ~ ABS
##         Df Sum of Sq    RSS    AIC F value Pr(>F)
## <none>               944.96 198.57
## AGE      1    80.876 864.08 193.59   7.020  0.010 **
## HT       1    61.598 883.36 195.31   5.230  0.025 *
## WT       1    43.734 901.22 196.87   3.640  0.060 .
## TRICEPS  1   242.173 702.78 177.47  25.844 <2e-16 ***
## SUBSCAP  1   132.580 812.38 188.77  12.240  0.001 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Example: Forward selection

The variable TRICEPS has the most significant (smallest) p-value = 2.639e-06 with the largest F value = 25.844, so it is added to the model:

```r
round(add1(lm(HWFAT~ABS+TRICEPS),
           scope=(~.+ AGE + HT + WT + SUBSCAP), test="F"),3)
```

```
## Single term additions
##
## Model:
## HWFAT ~ ABS + TRICEPS
##         Df Sum of Sq    RSS    AIC F value Pr(>F)
## <none>               702.78 177.47
## AGE      1    37.595 665.19 175.18   4.182  0.044 *
## HT       1    25.246 677.54 176.62   2.757  0.101
## WT       1    30.812 671.97 175.97   3.393  0.069 .
## SUBSCAP  1     2.244 700.54 179.22   0.237  0.628
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Example: Forward selection

The variable AGE has the most significant (smallest) p-value $=$ 0.044 with the largest F value$=$ 4.182, so it is added to the model:

```r
round(add1(lm(HWFAT~ABS+TRICEPS+AGE),
           scope=(~.+ HT + WT + SUBSCAP), test="F"),3)
```

```
## Single term additions
##
## Model:
## HWFAT ~ ABS + TRICEPS + AGE
##         Df Sum of Sq    RSS    AIC F value Pr(>F)
## <none>                665.19 175.18
## HT       1     7.029 658.16 176.35   0.780  0.380
## WT       1     6.833 658.36 176.38   0.758  0.387
## SUBSCAP  1     1.972 663.22 176.95   0.217  0.643
```

Example: Forward selection

None of the p-values now meet the α_{crit} level of 0.20, so the model is complete with `ABS`, `TRICEPS`, and `AGE` being used to predict `HWFAT`.

Note that the same model results in both the forward and backward selection procedures: (`HWFAT` $\sim$ `ABS + TRICEPS + AGE`). This is not always the case.

Example: Adjusted R^2

The R package leaps is needed for the function regsubsets(). The arguments have predictors as a matrix first, then the response as a vector. The first six variables of HSwrestler are the predictors, while the response, HWFAT, is in column 7.

```r
str(HSwrestler[,-c(8,9)])
library(leaps)
a <- regsubsets(as.matrix(HSwrestler[,-c(7,8,9)]), HSwrestler[,7])
```

```
## 'data.frame':    78 obs. of  7 variables:
##  $ AGE    : int  18 15 17 17 17 14 14 17 15 14 ...
##  $ HT     : num  65.8 65.5 64 72 69.5 ...
##  $ WT     : num  134 129 121 145 299 ...
##  $ ABS    : num  8 10 6 11 54 40 6 11 9 19 ...
##  $ TRICEPS: num  6 8 6 10 42 25 8 7 6 13 ...
##  $ SUBSCAP: num  10.5 9 8 10 37 26 7 8 8 11.5 ...
##  $ HWFAT  : num  10.71 8.53 6.78 9.32 41.89 ...
```

Example: Adjusted R^2

```r
summary(a)
```

```
## Subset selection object
## 6 Variables  (and intercept)
##         Forced in Forced out
## AGE         FALSE      FALSE
## HT          FALSE      FALSE
## WT          FALSE      FALSE
## ABS         FALSE      FALSE
## TRICEPS     FALSE      FALSE
## SUBSCAP     FALSE      FALSE
## 1 subsets of each size up to 6
## Selection Algorithm: exhaustive
##          AGE HT  WT  ABS TRICEPS SUBSCAP
## 1  ( 1 ) " " " " " " "*" " "     " "
## 2  ( 1 ) " " " " " " "*" "*"     " "
## 3  ( 1 ) "*" " " " " "*" "*"     " "
## 4  ( 1 ) "*" "*" " " "*" "*"     " "
## 5  ( 1 ) "*" " " "*" "*" "*"     "*"
## 6  ( 1 ) "*" "*" "*" "*" "*"     "*"
```

Example: Adjusted R^2

```r
summary(a)$adjr2
```

```
## [1] 0.8409068 0.8801014 0.8849817 0.8846381 0.8840129
## [6] 0.8826699
```

```r
max(summary(a)$adjr2)
```

```
## [1] 0.8849817
```

```r
which.max(summary(a)$adjr2)
```

```
## [1] 3
```

The largest R_a^2 value is 0.8849817, which corresponds to the model with three predictors.

The row beside the 3 shows "*" symbols for `AGE`, `ABS`, and `TRICEPS`, so these are the appropriate predictor variables.

Example: Mallows's C_p

When using Mallows's C_p, the idea is to select the smallest C_p value less than or equal to p.

In this case, the `R` package `leaps` and the output from `regsubsets()` gives the optimal value $C_4 = 2.541953$, so the three-predictor (plus an intercept) model using `AGE`, `ABS`, and `TRICEPS` is again selected:

Example: Mallows's C_p

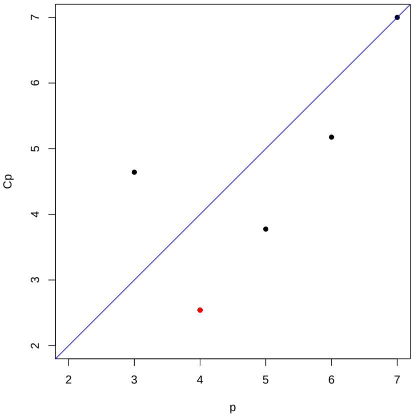
```
## Subset selection object
## 6 Variables  (and intercept)
##         Forced in Forced out
## AGE         FALSE      FALSE
## HT          FALSE      FALSE
## WT          FALSE      FALSE
## ABS         FALSE      FALSE
## TRICEPS     FALSE      FALSE
## SUBSCAP     FALSE      FALSE
## 1 subsets of each size up to 6
## Selection Algorithm: exhaustive
##          AGE HT  WT  ABS TRICEPS SUBSCAP
## 1  ( 1 ) " " " " " " "*" " "     " "
## 2  ( 1 ) " " " " " " "*" "*"     " "
## 3  ( 1 ) "*" " " " " "*" "*"     " "
## 4  ( 1 ) "*" "*" " " "*" "*"     " "
## 5  ( 1 ) "*" " " "*" "*" "*"     "*"
## 6  ( 1 ) "*" "*" "*" "*" "*"     "*"
```

```
summary(a)$cp
```

```
## [1] 29.051861  4.641808  2.541953  3.775400  5.175856
## [6]  7.000000
```

Example: Mallows's C_p

```
par(pty="s")
plot(2:7, summary(a)$cp, ylim=c(2,7), xlab="p",
     ylab="Cp", pch=16); abline(a=0, b=1, col='blue');
points(4,summary(a)$cp[3], col='red', pch=16)
```



Example: AIC

The function `stepAIC()` in the `MASS` package will compute models based on both `AIC` and `BIC` statistics.

The argument `k` of this function will be set equal to 2 for the AIC statistic and `ln(n)` for the BIC statistic.

The user needs to specify the scope of the model with the argument `scope=`. In this case, the scope of the model includes any of the six predictors `AGE`, `HT`, `WT`, `ABS`, `TRICEPS`, and `SUBSCAP`. For further details, see the `stepAIC()` help file.

The starting AIC value is 179.51. The `stepAIC()` function adds or removes variables until it finds the smallest AIC value.

A – before a variable indicates that the variable will be removed to produce the given AIC, while a ‘$+$ indicates the variable will be added to produce the given AIC.

Example: AIC

```r
reg.all <- lm(HWFAT ~ AGE + HT + WT + ABS + TRICEPS + SUBSCAP)
mod.aic <- stepAIC(reg.all, direction="both",
                   scope=(~.+SUBSCAP+TRICEPS+ABS+WT+HT+AGE), k=2)
```

```
## Start:  AIC=179.51
## HWFAT ~ AGE + HT + WT + ABS + TRICEPS + SUBSCAP
##
##           Df Sum of Sq    RSS    AIC
## - HT       1     1.613 652.66 177.70
## - WT       1     2.546 653.60 177.81
## - SUBSCAP  1     5.921 656.97 178.21
## - AGE      1     9.594 660.64 178.65
## <none>                 651.05 179.51
## - TRICEPS  1    72.683 723.73 185.76
## - ABS      1   162.000 813.05 194.84
##
## Step:  AIC=177.7
## HWFAT ~ AGE + WT + ABS + TRICEPS + SUBSCAP
##
##           Df Sum of Sq    RSS    AIC
## - SUBSCAP  1     5.693 658.36 176.38
## - AGE      1     9.875 662.54 176.87
```

Example: AIC

```
mod.aic
```

```
##
## Call:
## lm(formula = HWFAT ~ AGE + ABS + TRICEPS)
##
## Coefficients:
## (Intercept)          AGE          ABS      TRICEPS
##     10.6161      -0.5331       0.3564       0.4656
```

Example: AIC

The final model uses `AGE, ABS`, and `TRICEPS` as predictors.

Example: BIC

When BIC is the criterion, the model selected is `HWFAT ~ ABS + TRICEPS`.

```
mod.bic <- stepAIC(reg.all, direction="both",
              scope=(~.+SUBSCAP+TRICEPS+ABS+WT+HT+AGE),
              k=log(length(HWFAT)))
```

```
## Start:  AIC=196
## HWFAT ~ AGE + HT + WT + ABS + TRICEPS + SUBSCAP
##
##           Df Sum of Sq    RSS    AIC
## - HT       1     1.613 652.66 191.84
## - WT       1     2.546 653.60 191.95
## - SUBSCAP  1     5.921 656.97 192.35
## - AGE      1     9.594 660.64 192.79
## <none>                 651.05 196.00
## - TRICEPS  1    72.683 723.73 199.90
## - ABS      1   162.000 813.05 208.98
##
## Step:  AIC=191.84
## HWFAT ~ AGE + WT + ABS + TRICEPS + SUBSCAP
```

Example: BIC

```
mod.bic
```

```
##
## Call:
## lm(formula = HWFAT ~ ABS + TRICEPS)
##
## Coefficients:
## (Intercept)          ABS      TRICEPS
##      2.0590       0.3371       0.5043
```