Notebook 2 One Way ANOVA Example (Corn Yields)

```r
# The following R analysis of the corn data described on pages 6 and 7 of the brick, is an
# expanded version of the code shown on pages 7 to 11 of the brick.

# There are some differences between this and the S-Plus version shown in the brick, which I
# will note and discuss below.

# Read and attach the data and examine the contents. The advantage of attaching the data is
# that we can refer to the elements of the data as if they were vectors, without having to
# specify the data= option in functions such as lm():

corn <- read.table("Corn.txt",header=T)
corn
```

```
##    yield     fert
## 1     99  Control
## 2     40  Control
## 3     61  Control
## 4     72  Control
## 5     76  Control
## 6     84  Control
## 7     96    K20+N
## 8     84    K20+N
## 9     82    K20+N
## 10   104    K20+N
## 11    99    K20+N
## 12   105    K20+N
## 13    63 K20+P205
## 14    57 K20+P205
## 15    81 K20+P205
## 16    59 K20+P205
## 17    64 K20+P205
## 18    72 K20+P205
## 19    79   N+P205
## 20    92   N+P205
## 21    91   N+P205
## 22    87   N+P205
## 23    78   N+P205
## 24    71   N+P205
```

```r
attach(corn)
names(corn)
```

```
## [1] "yield" "fert"
```

```r
yield
```

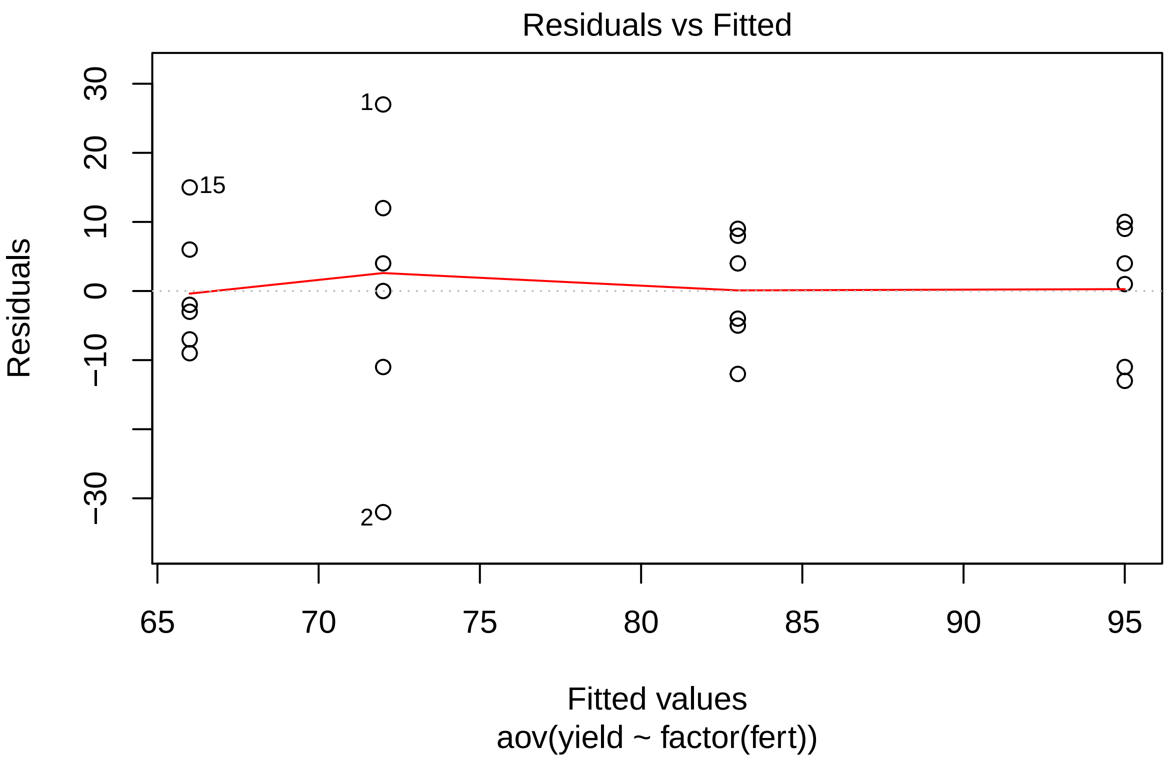
```
##  [1]  99  40  61  72  76  84  96  84  82 104  99 105  63  57  81  59  64
## [18]  72  79  92  91  87  78  71
```

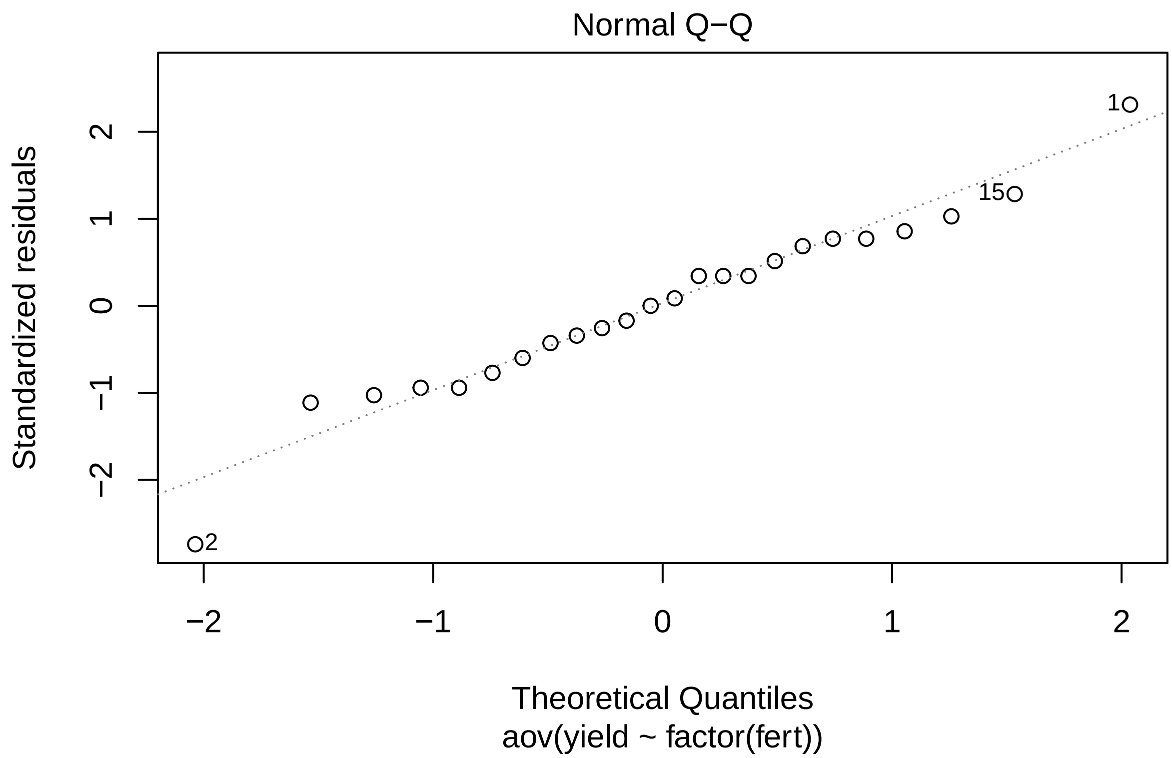
```r
fert
```

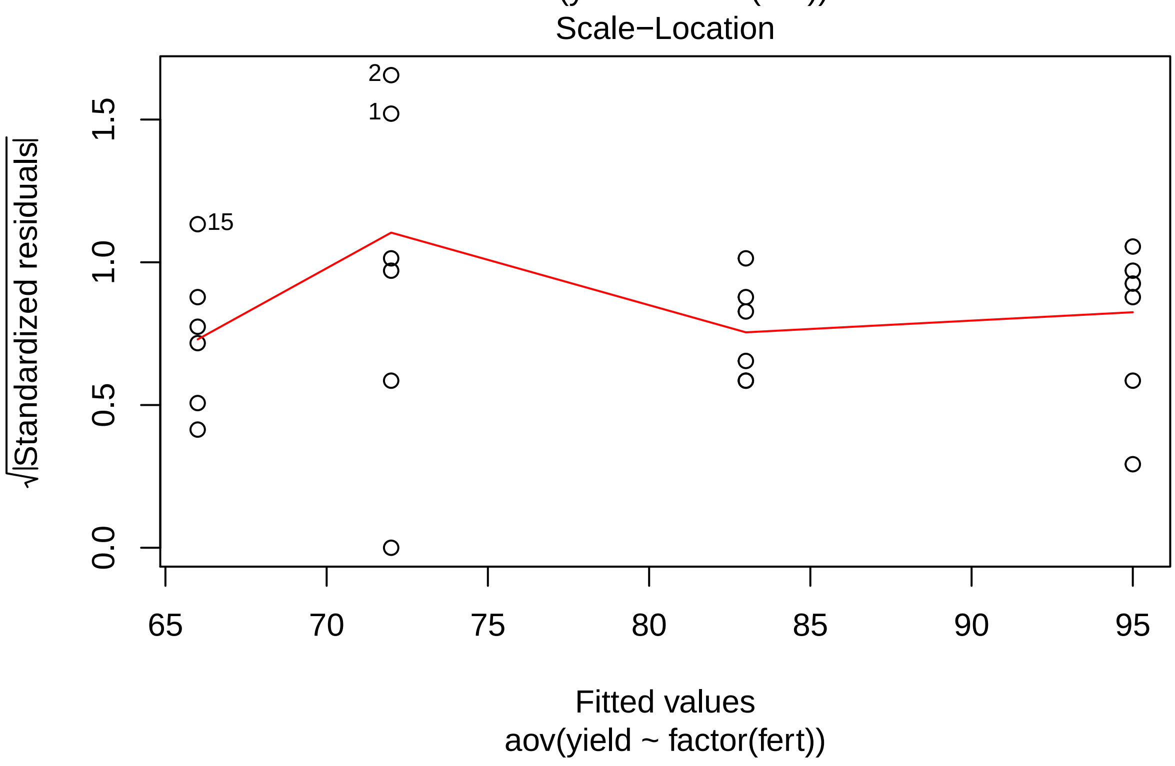
```
##  [1] Control  Control  Control  Control  Control  Control  K20+N
##  [8] K20+N    K20+N    K20+N    K20+N    K20+N    K20+P205 K20+P205
```

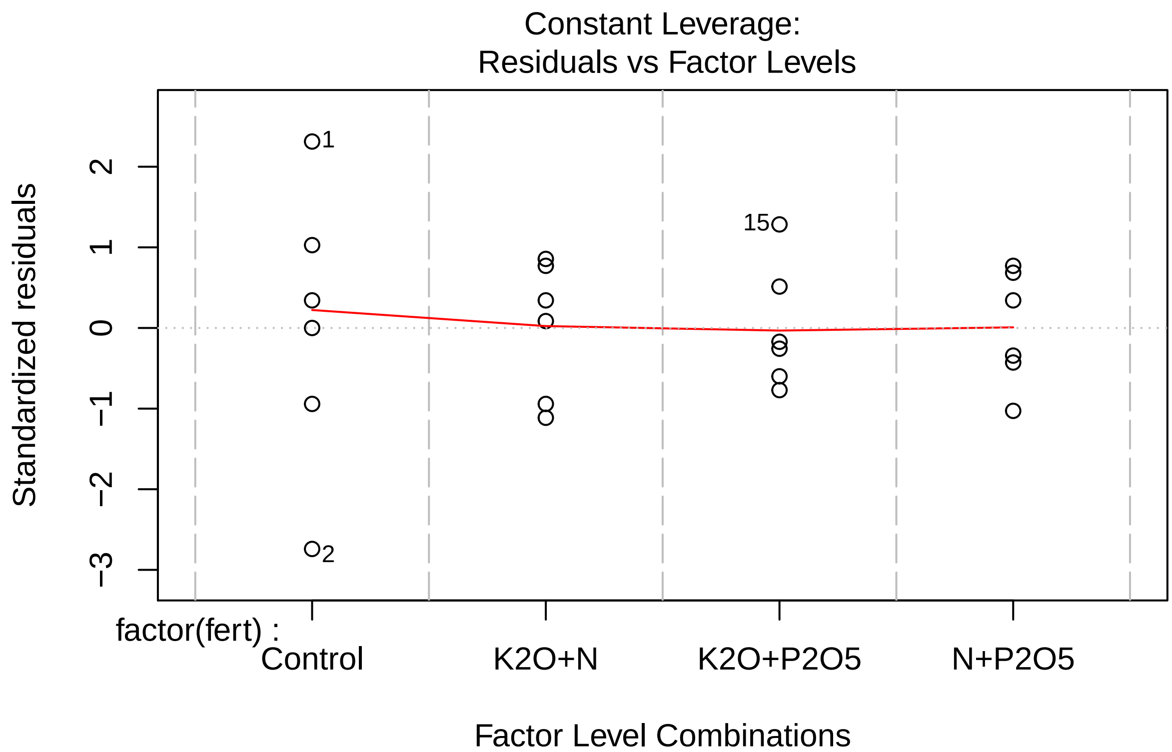
```
## [15] K20+P205 K20+P205 K20+P205 K20+P205 N+P205   N+P205   N+P205
## [22] N+P205   N+P205   N+P205
## Levels: Control K20+N K20+P205 N+P205
```

```r
# The aov command in R (or S-Plus) is used to fit the one-way ANOVA model described in the brick.
# The factor command is used to indicate that fert is a factor variable:

corn.aov <- aov(yield ~ factor(fert))

# We can use the plot command to asssess whether this is an appropriate model for the data.

plot(corn.aov)
```




Normal Q−Q
Standardized residuals
Theoretical Quantiles
aov(yield ~ factor(fert))
−2
−1
0
1
2
1
15
2

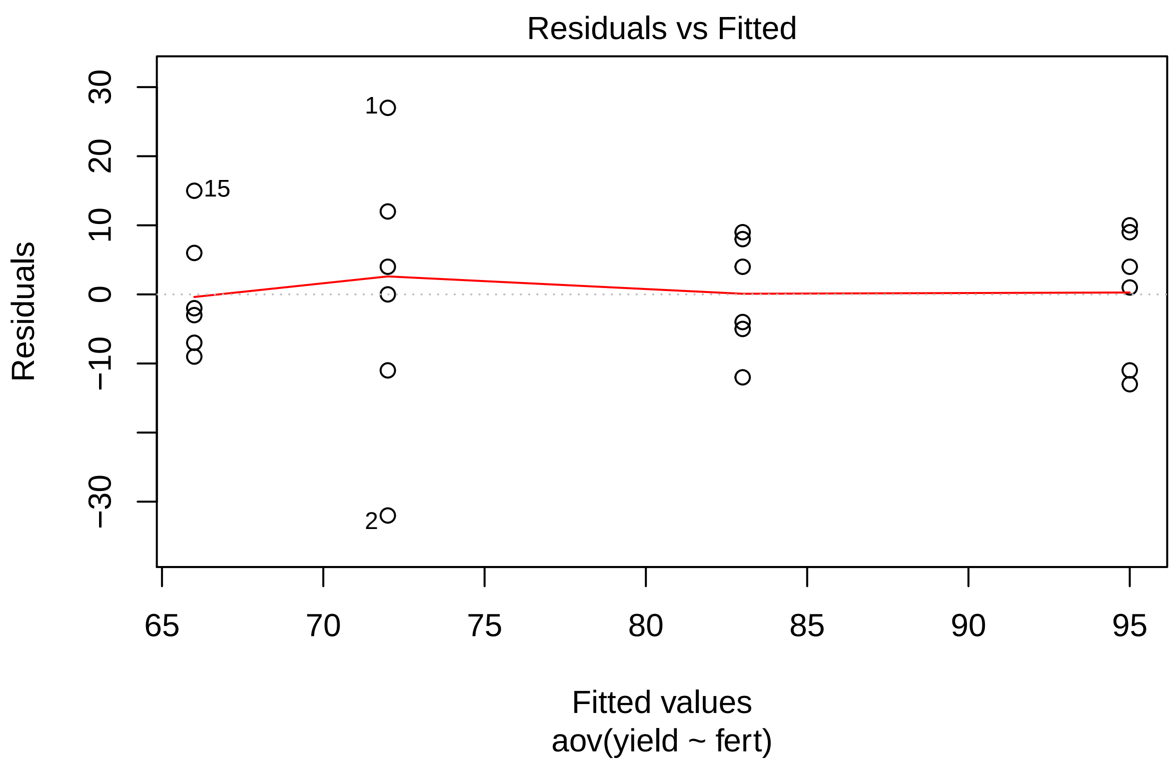

Scale−Location
√|Standardized residuals|
Fitted values
aov(yield ~ factor(fert))
0.0
0.5
1.0
1.5
65
70
75
80
85
90
95
2
1
15

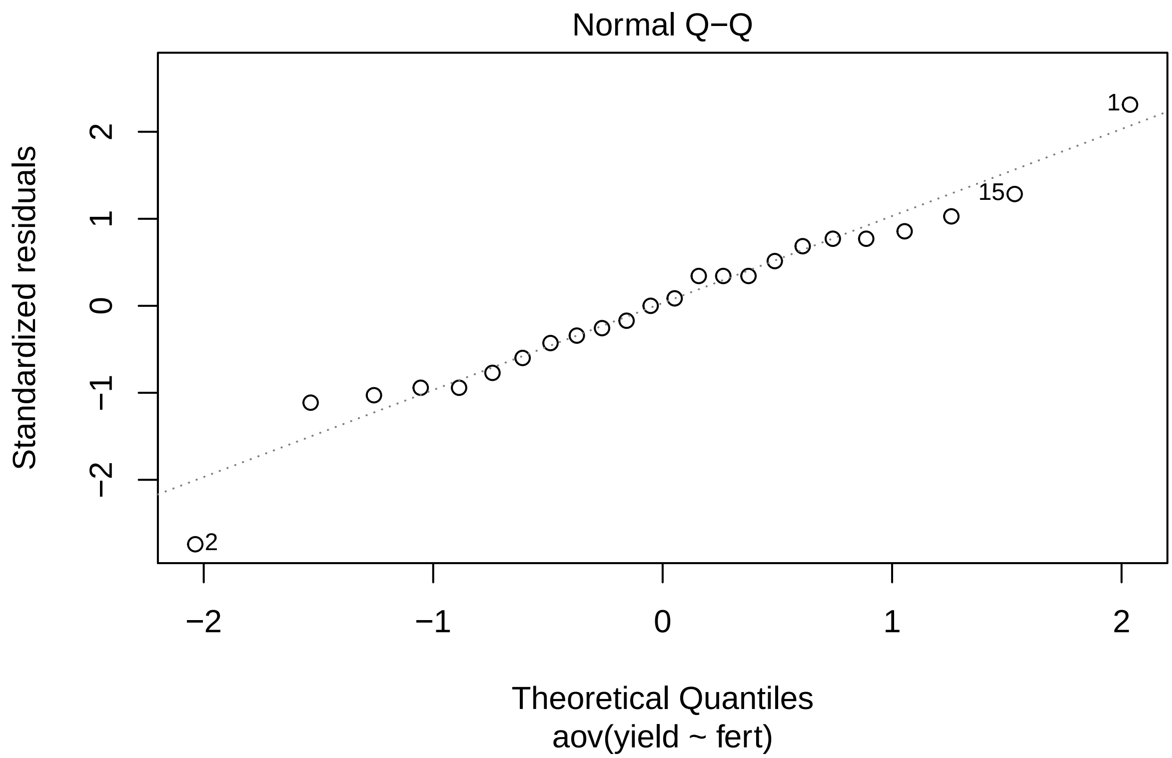


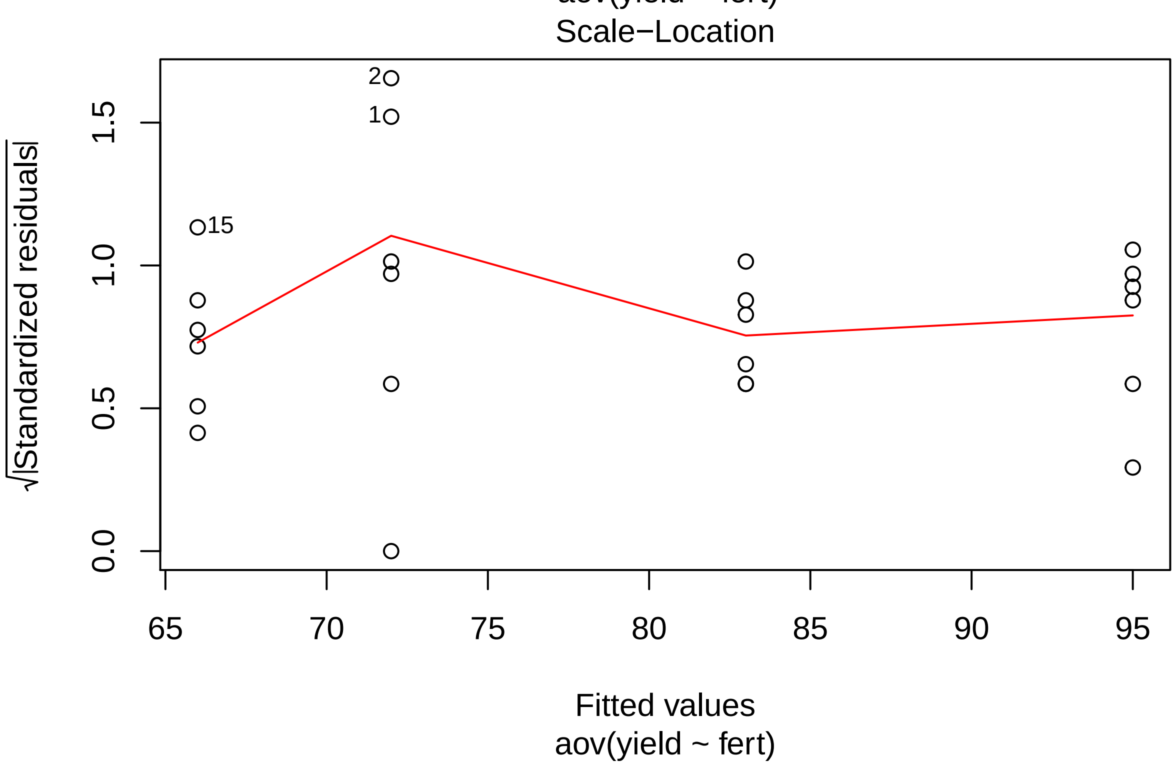

```r
# Here is a quick description of what the generic plot() function produces in R for a
# lm() or aov() object. For more details, see:

help(plot.lm)

# The four plots produced by default for a stored linear model are:

# Plot 1 - a plot of the residuals vs fitted values which is very important for assessing the
# underlying assumptions of independence and constant variance

# Plot 2 - a normal quantile plot of the rstandard() residuals

# Plot 3 - another plot of the (transformed) residuals vs the fitted values, this is an another
# attempt to graphically examine the assumption of constant variance

# Plot 4 - a leverage plot - for a lm() object, this will typically be a plot of the standardised
# residuals against the leverage values. On such a plot, arbitrary limits can be drawn for the
# Cook's D values for each of the data points, as Cook's D is a function of the
# standardised residuals and the leverage values.
# For an aov() object, especially for a balanced experimental design where all the observations
# have the same (constant leverage), plot 4 becomes a plot of the standardised residuals against
# the treatments (the factor level combinations).

# These plots in R suggest a problem - there appear to be two possible outliers in one of the
# groups - observations 1 and 2 - more about this problem later.

# We can examine the ANOVA table for an object created using the aov() function by applying the
# summary function:

summary(corn.aov)
```

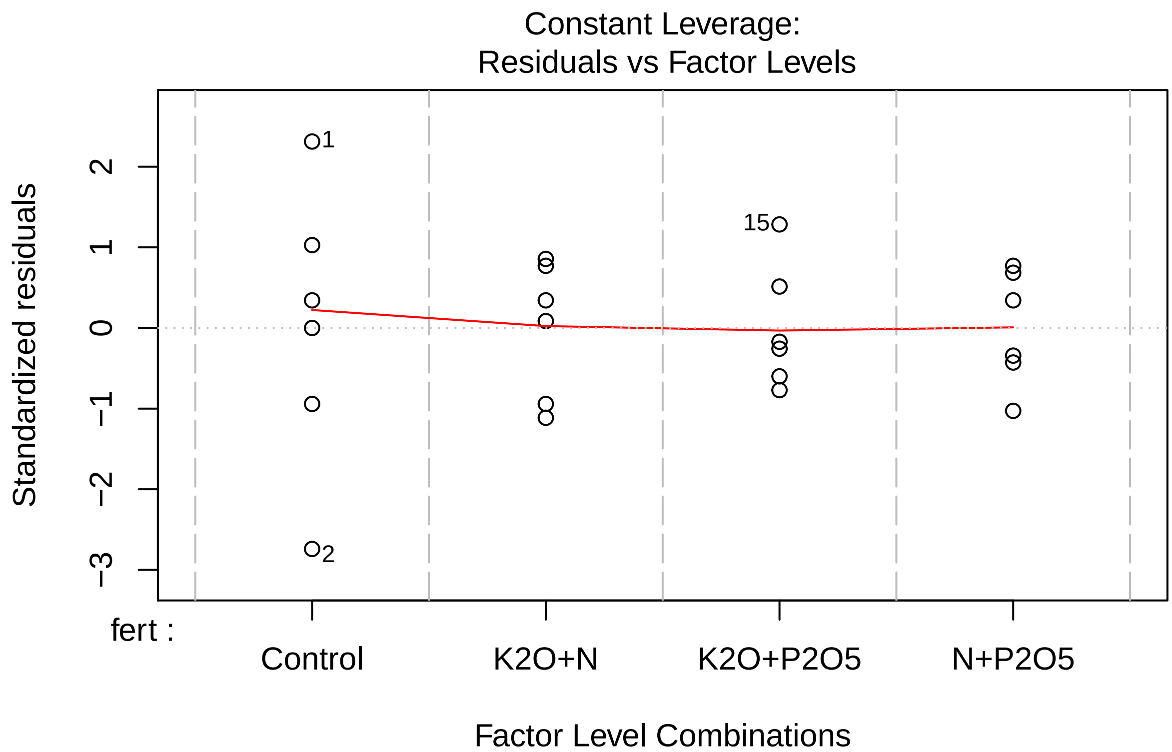
```
##               Df Sum Sq Mean Sq F value  Pr(>F)
## factor(fert)   3   2940   980.0    5.99 0.00439 **
## Residuals     20   3272   163.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Note that R adopts the dubious practice of starring the results at different level of
# significance - dubious because the choice of significance level should be chosen a priori
# (before the experiment), rather than adjusted to suit, after the analysis (a posteriori).

# Note the factor command is only really necessary when the categorical variable is coded using
# numbers such as 1,2,3,4. Here fert is a series of qualitative labels, which R would have
# correctly interpreted as a factor, not a continuous variable -  as can be seen if we refit
# the model:

corn.aov2 <- aov(yield ~ fert)
plot(corn.aov2)
```




Normal Q−Q
Standardized residuals
Theoretical Quantiles
aov(yield ~ fert)
−2
−1
0
1
2
1
15
2

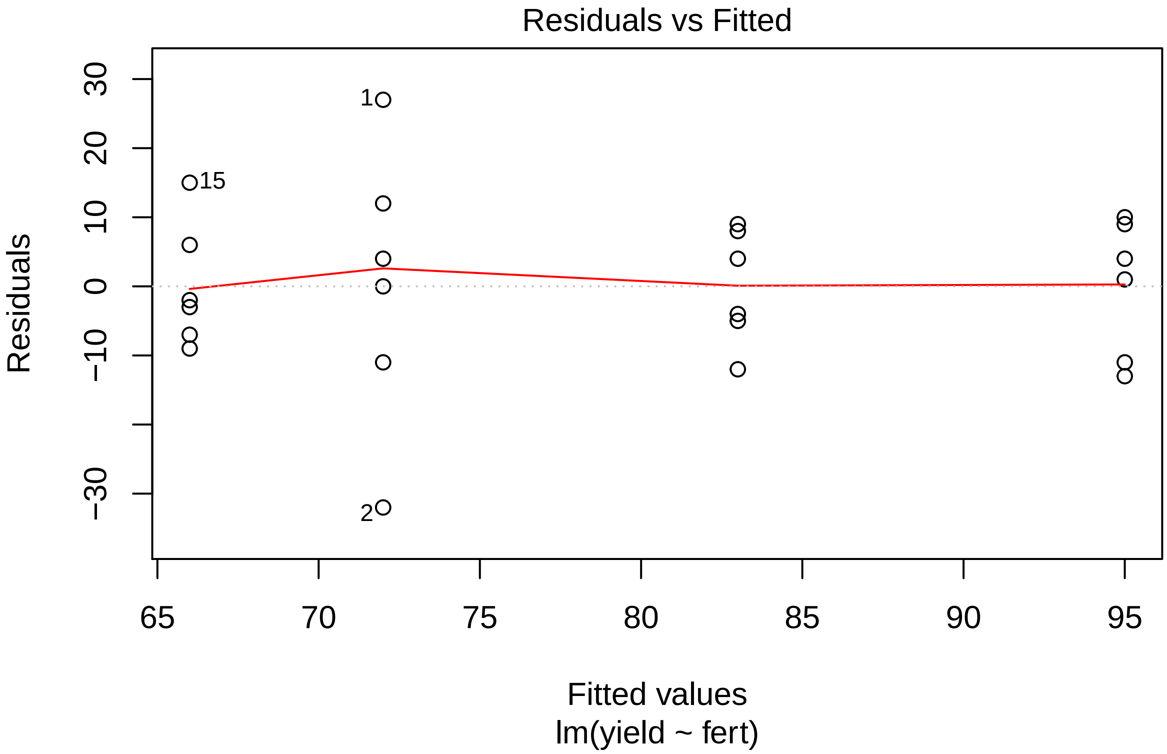

Scale−Location
√|Standardized residuals|
Fitted values
aov(yield ~ fert)
0.0
0.5
1.0
1.5
65
70
75
80
85
90
95
2
1
15

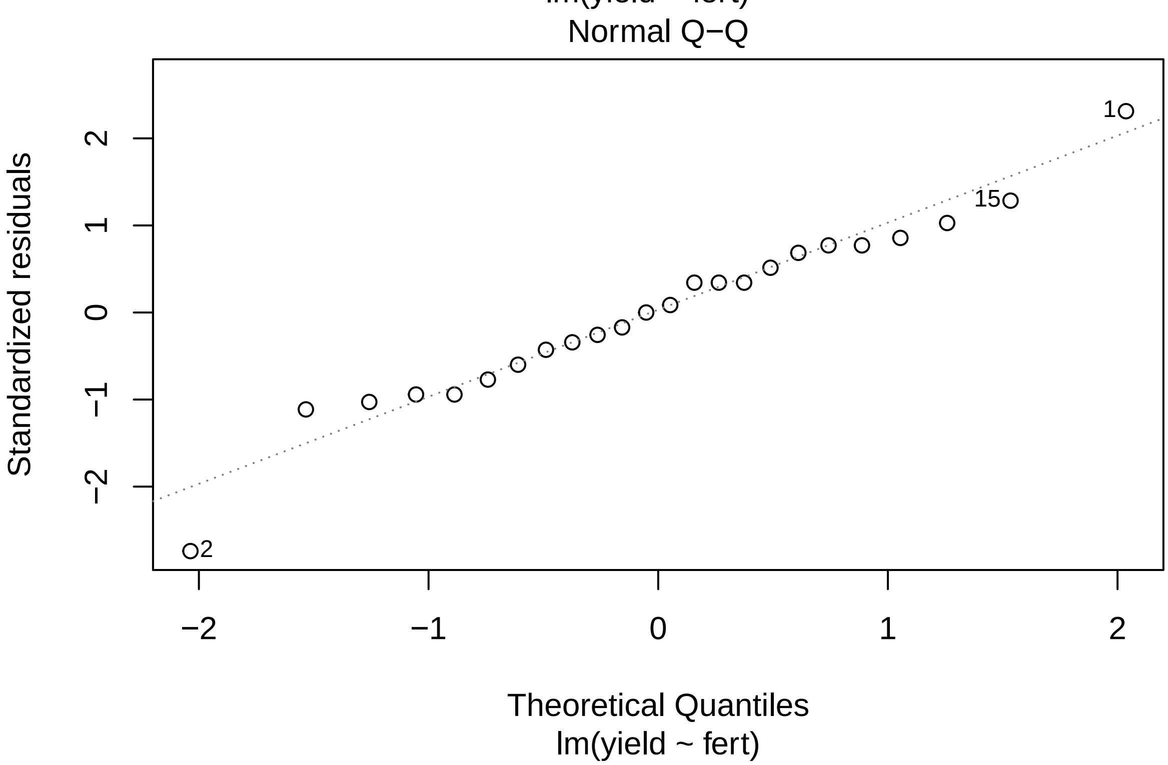


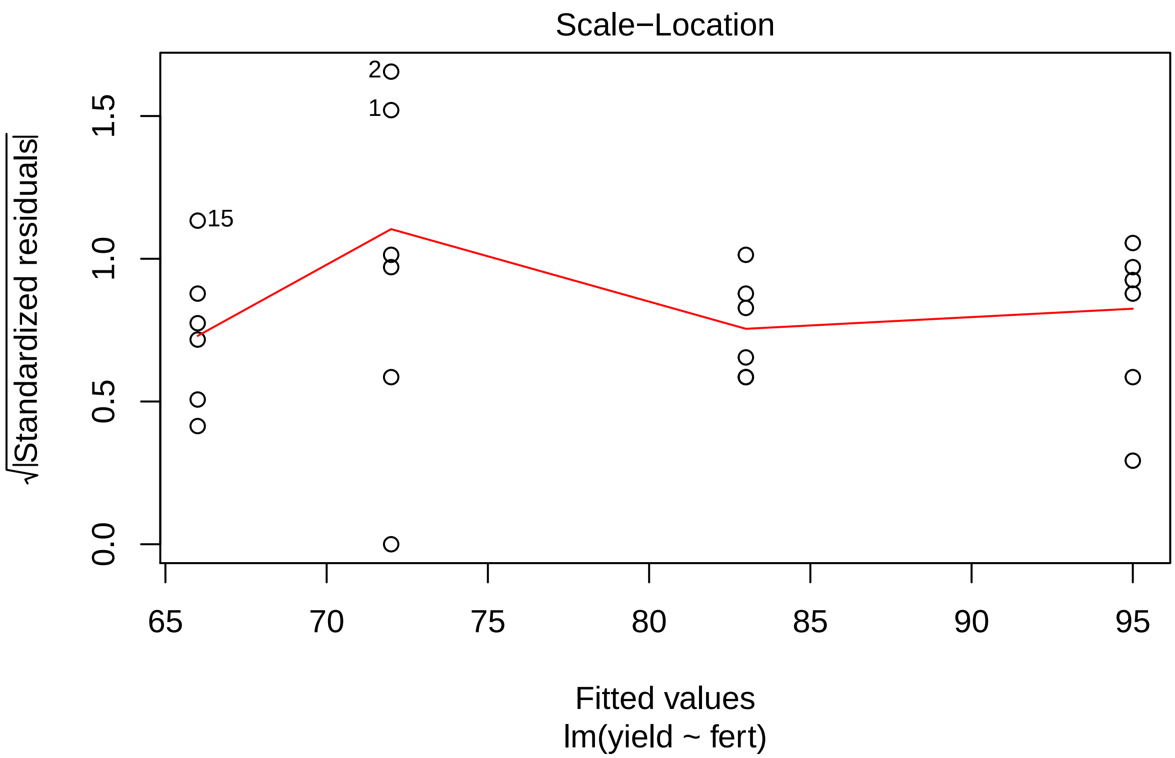

```r
summary(corn.aov2)
```

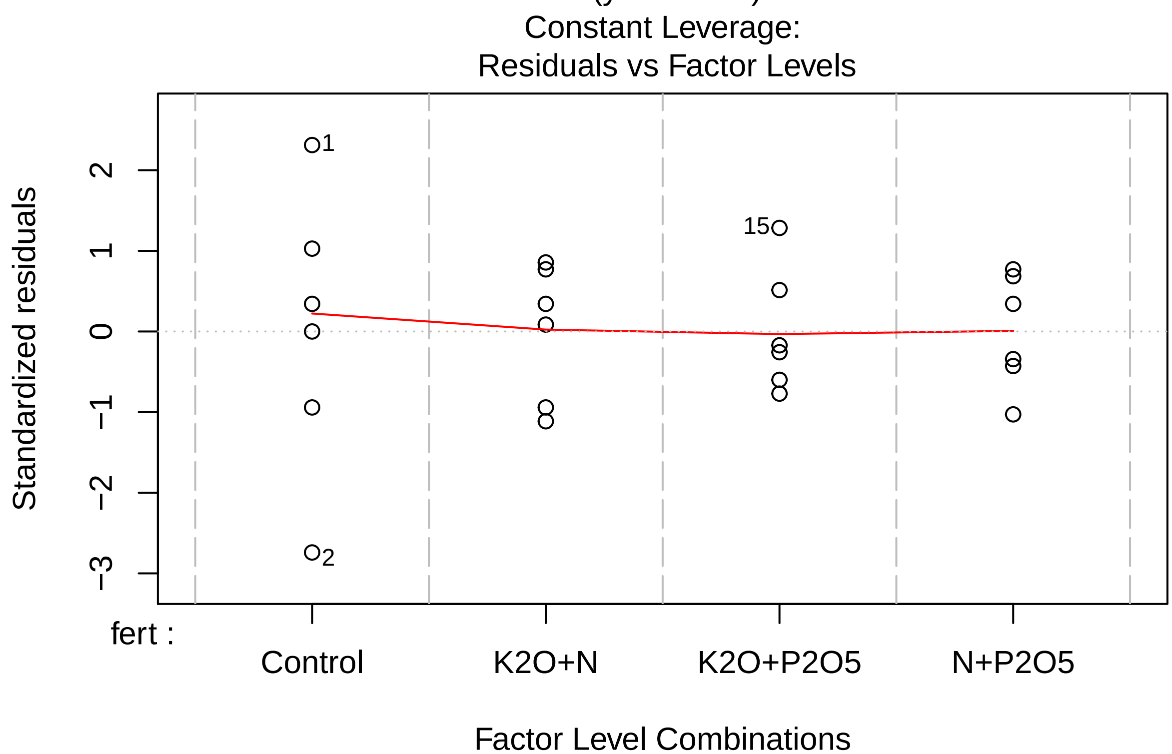
```
##             Df Sum Sq Mean Sq F value  Pr(>F)
## fert         3   2940   980.0    5.99 0.00439 **
## Residuals   20   3272   163.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# We could also use the lm() function used to fit regression models in STAT2008 to fit the same
# model:

corn.lm <- lm(yield ~ fert)
plot(corn.lm)
```


Residuals vs Fitted
Residuals
Fitted values
lm(yield ~ fert)
15
1
2


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(yield ~ fert)
1
15
2






```
anova(corn.lm)
```

```
## Analysis of Variance Table
##
## Response: yield
##           Df Sum Sq Mean Sq F value   Pr(>F)
## fert       3   2940   980.0  5.9902 0.004387 **
```

```
## Residuals 20   3272   163.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(corn.lm)
```

```
##
## Call:
## lm(formula = yield ~ fert)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -32.00  -7.50   0.50   8.25  27.00
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    72.000      5.222  13.788 1.13e-11 ***
## fertK20+N      23.000      7.385   3.115  0.00546 **
## fertK20+P205   -6.000      7.385  -0.812  0.42607
## fertN+P205     11.000      7.385   1.490  0.15194
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.79 on 20 degrees of freedom
## Multiple R-squared:  0.4733, Adjusted R-squared:  0.3943
## F-statistic:  5.99 on 3 and 20 DF,  p-value: 0.004387
```

```r
# The main difference between an lm() and an aov() object is that summary gives the table of
# coefficients for the lm() object and you need to use anova() to see the ANOVA table.
# To get the coefficients for an aov() object you need to use:

summary.lm(corn.aov2)
```

```
##
## Call:
## aov(formula = yield ~ fert)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -32.00  -7.50   0.50   8.25  27.00
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    72.000      5.222  13.788 1.13e-11 ***
## fertK20+N      23.000      7.385   3.115  0.00546 **
## fertK20+P205   -6.000      7.385  -0.812  0.42607
## fertN+P205     11.000      7.385   1.490  0.15194
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.79 on 20 degrees of freedom
## Multiple R-squared:  0.4733, Adjusted R-squared:  0.3943
## F-statistic:  5.99 on 3 and 20 DF,  p-value: 0.004387
```

```r
# How do we interpret these coefficients?  Unlike version 7 and earlier versions of S-Plus,
# (where the rather strange choice of defaults were Helmert contrasts, which are an attempt
```

```
# to deal with highly unbalanced or non-orthogonal experimental designs) the default
# parameterisation for factor variables in R and in S-Plus from version 8 onwards are treatment
# contrasts:

contrasts(fert)
```

```
##          K2O+N K2O+P2O5 N+P2O5
## Control      0        0      0
## K2O+N        1        0      0
## K2O+P2O5     0        1      0
## N+P2O5       0        0      1
```

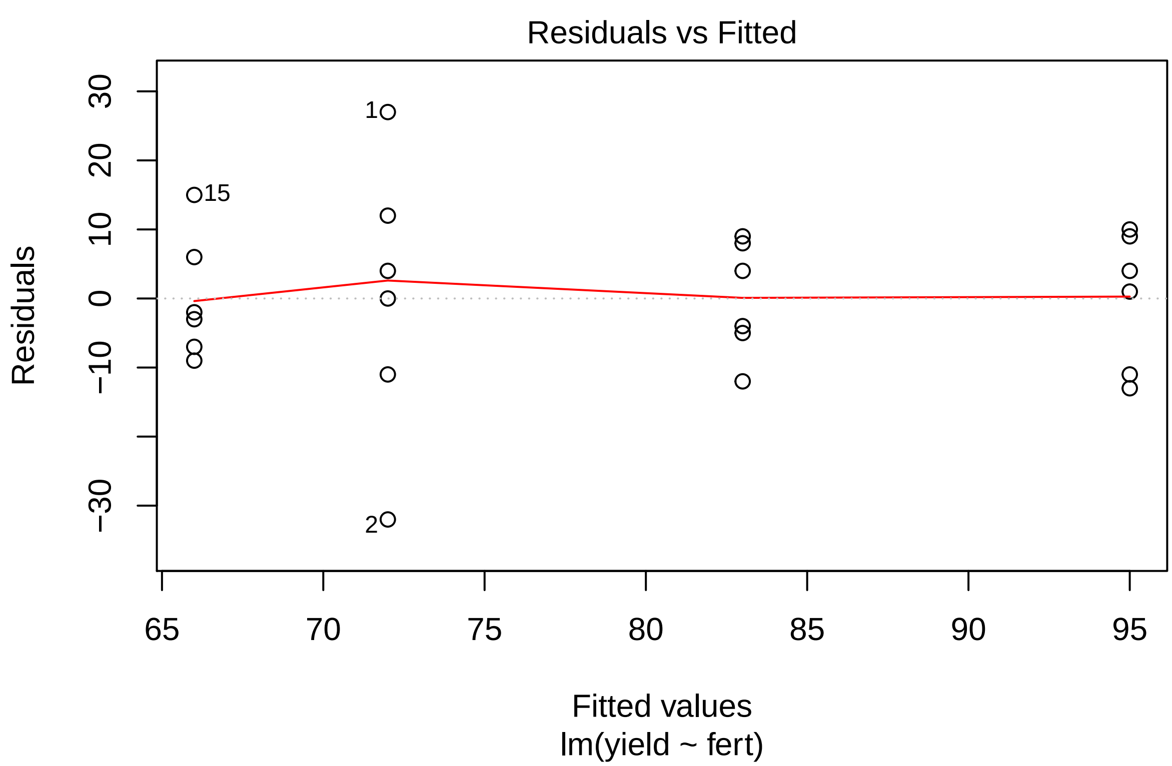
```
# Under treatment contrasts, the model parameters ARE closely related to the mean yields
# for the four fertilizers, which can be calculated as follows:

mean(yield)
```

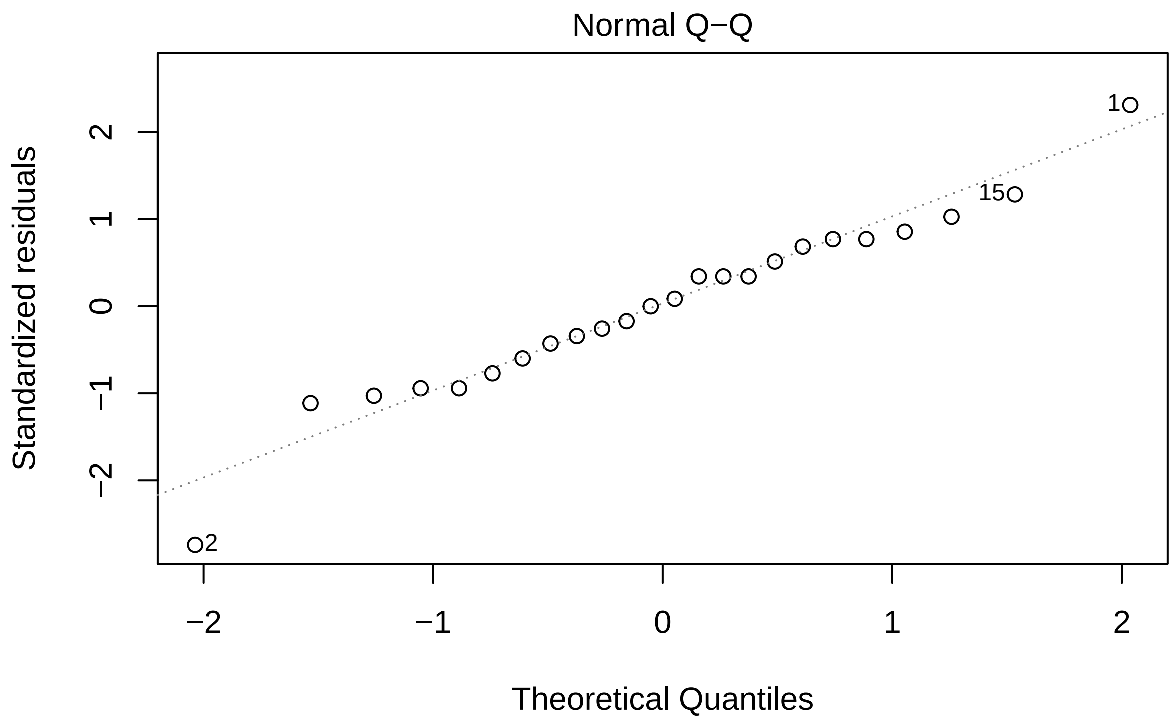
```
## [1] 79
```

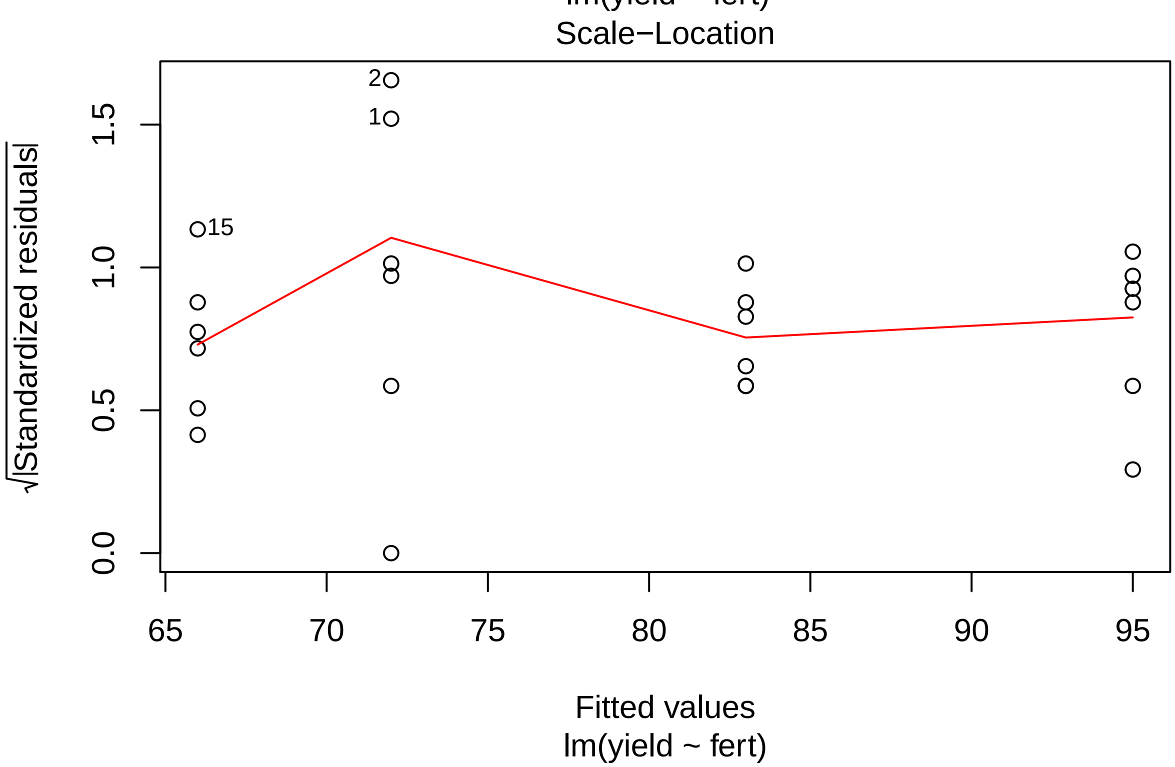
```
lvl.mns <- tapply(yield, fert, mean)
lvl.mns
```

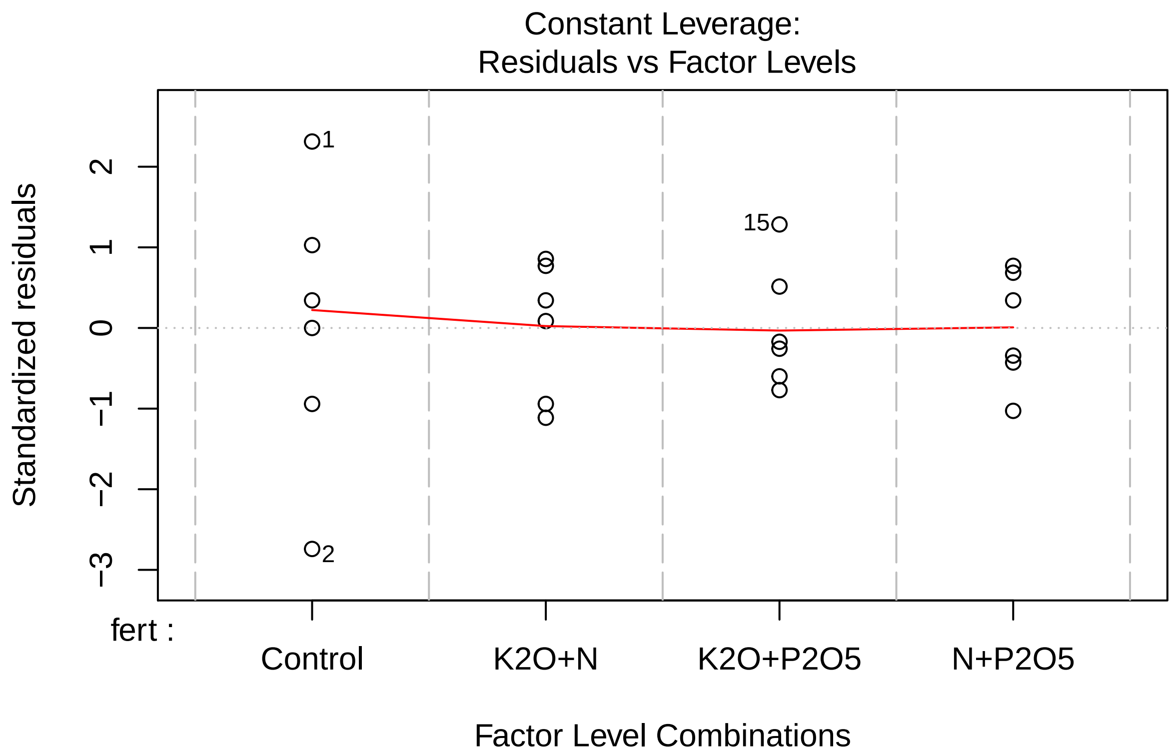
```
##  Control    K2O+N K2O+P2O5   N+P2O5
##       72       95       66       83
```

```
# The following approach will also fit the 0-1 dummy or indicator variables described on page 10 of the
# In R or S-Plus version 8, this is the default version, so will give identical results to the previous

corn.lm2 <- lm(yield ~ fert, contrasts=list(fert=contr.treatment))
plot(corn.lm2)
```




Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(yield ~ fert)
−2
−1
0
1
2
15


Scale−Location
√|Standardized residuals|
Fitted values
lm(yield ~ fert)
0.0
0.5
1.0
1.5
65
70
75
80
85
90
95
2
1
15




```r
anova(corn.lm2)
```

```
## Analysis of Variance Table
##
## Response: yield
##           Df Sum Sq Mean Sq F value   Pr(>F)
## fert       3   2940   980.0  5.9902 0.004387 **
## Residuals 20   3272   163.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(corn.lm2)
```

```
##
## Call:
## lm(formula = yield ~ fert, contrasts = list(fert = contr.treatment))
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -32.00  -7.50   0.50   8.25  27.00
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   72.000      5.222  13.788 1.13e-11 ***
## fert2         23.000      7.385   3.115  0.00546 **
## fert3         -6.000      7.385  -0.812  0.42607
## fert4         11.000      7.385   1.490  0.15194
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.79 on 20 degrees of freedom
```

```
## Multiple R-squared:  0.4733, Adjusted R-squared:  0.3943
## F-statistic:  5.99 on 3 and 20 DF,  p-value: 0.004387
```

```r
# In this "treatment" parameterisation, R has chosen the first or Control group as the reference
# group by creating the following 0-1 indicator variables:

contr.treatment(4)
```

```
##   2 3 4
## 1 0 0 0
## 2 1 0 0
## 3 0 1 0
## 4 0 0 1
```

```r
# If we are not happy with R's default choice of reference group, we could do our own manual
# coding. Here is the manually coded equivalent of what R has done above:

fert1 <- ifelse(fert=="Control",1,0)
fert2 <- ifelse(fert=="K2O+N",1,0)
fert3 <- ifelse(fert=="K2O+P2O5",1,0)
fert4 <- ifelse(fert=="N+P2O5",1,0)
ferts <- cbind(fert1, fert2, fert3, fert4)
ferts
```

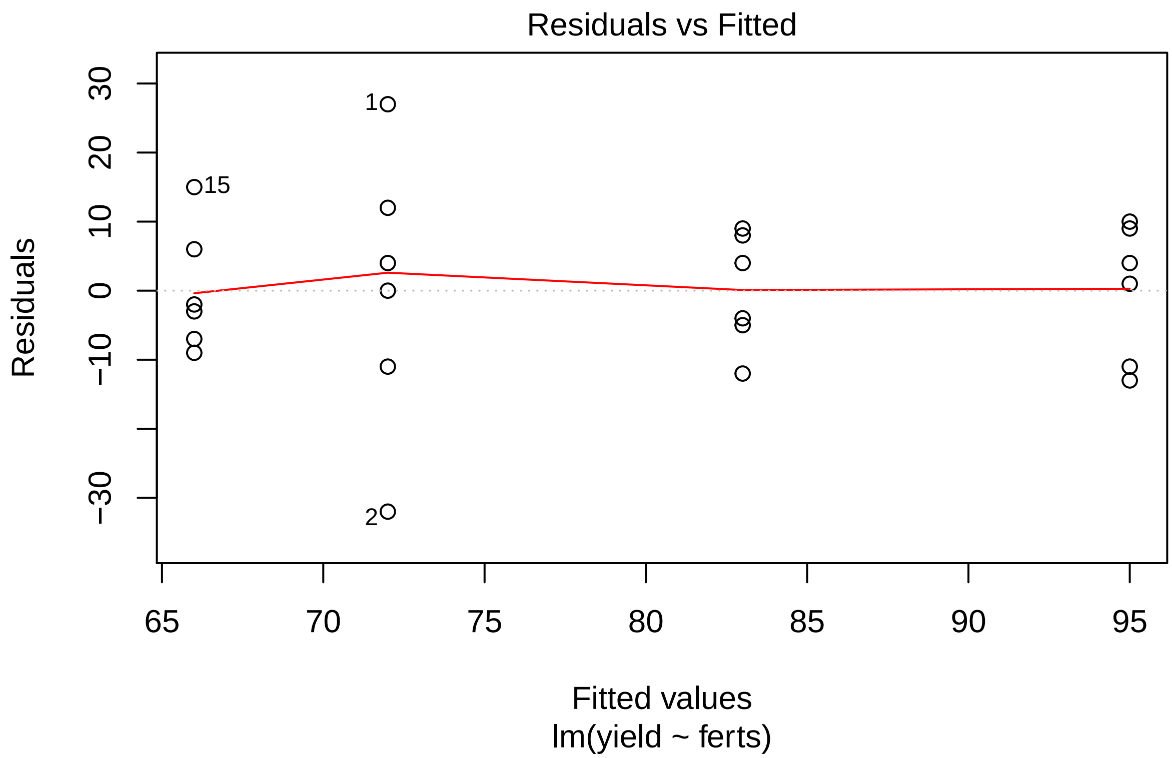
```
##       fert1 fert2 fert3 fert4
##  [1,]     1     0     0     0
##  [2,]     1     0     0     0
##  [3,]     1     0     0     0
##  [4,]     1     0     0     0
##  [5,]     1     0     0     0
##  [6,]     1     0     0     0
##  [7,]     0     1     0     0
##  [8,]     0     1     0     0
##  [9,]     0     1     0     0
## [10,]     0     1     0     0
## [11,]     0     1     0     0
## [12,]     0     1     0     0
## [13,]     0     0     1     0
## [14,]     0     0     1     0
## [15,]     0     0     1     0
## [16,]     0     0     1     0
## [17,]     0     0     1     0
## [18,]     0     0     1     0
## [19,]     0     0     0     1
## [20,]     0     0     0     1
## [21,]     0     0     0     1
## [22,]     0     0     0     1
## [23,]     0     0     0     1
## [24,]     0     0     0     1
```

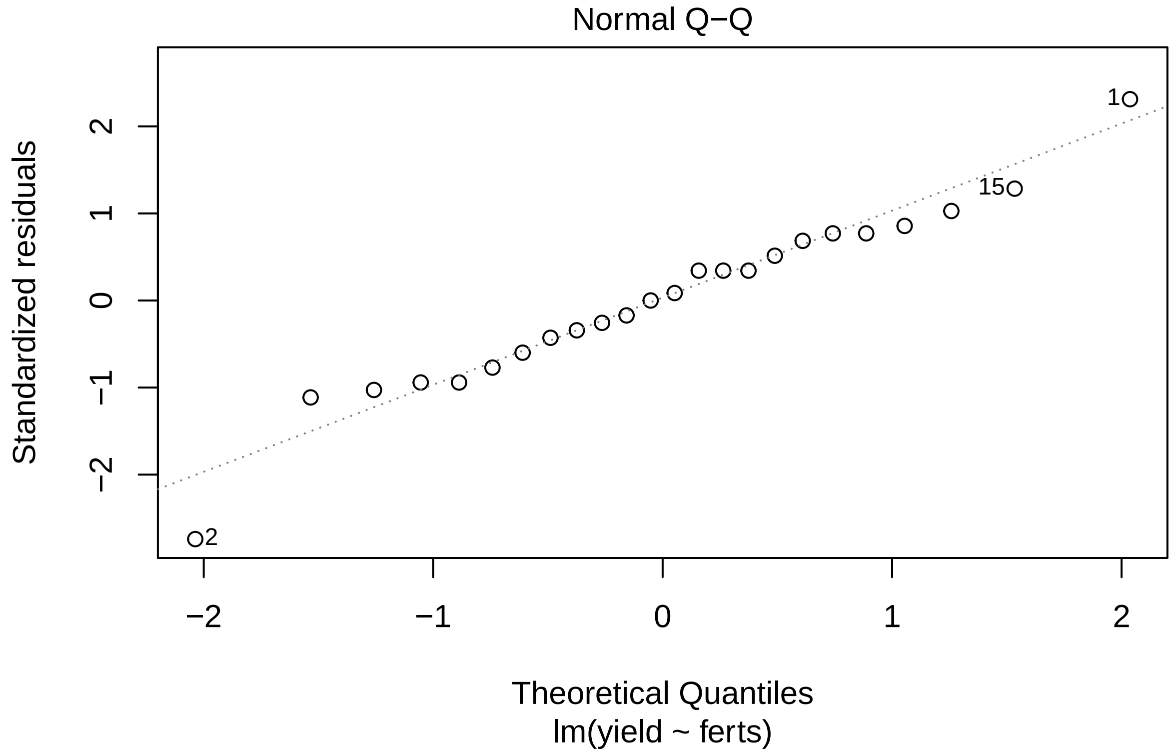
```r
corn.lm2a <- lm(yield ~ ferts)

# Note that unlike S-Plus, R does not give an error message to indicate that this model is
# over-parameterised, but if you examine the cofficients for this model, you find that R has
# simply decided not to fit the last of the parameters (fert4):
```

```r
summary(corn.lm2a)
```

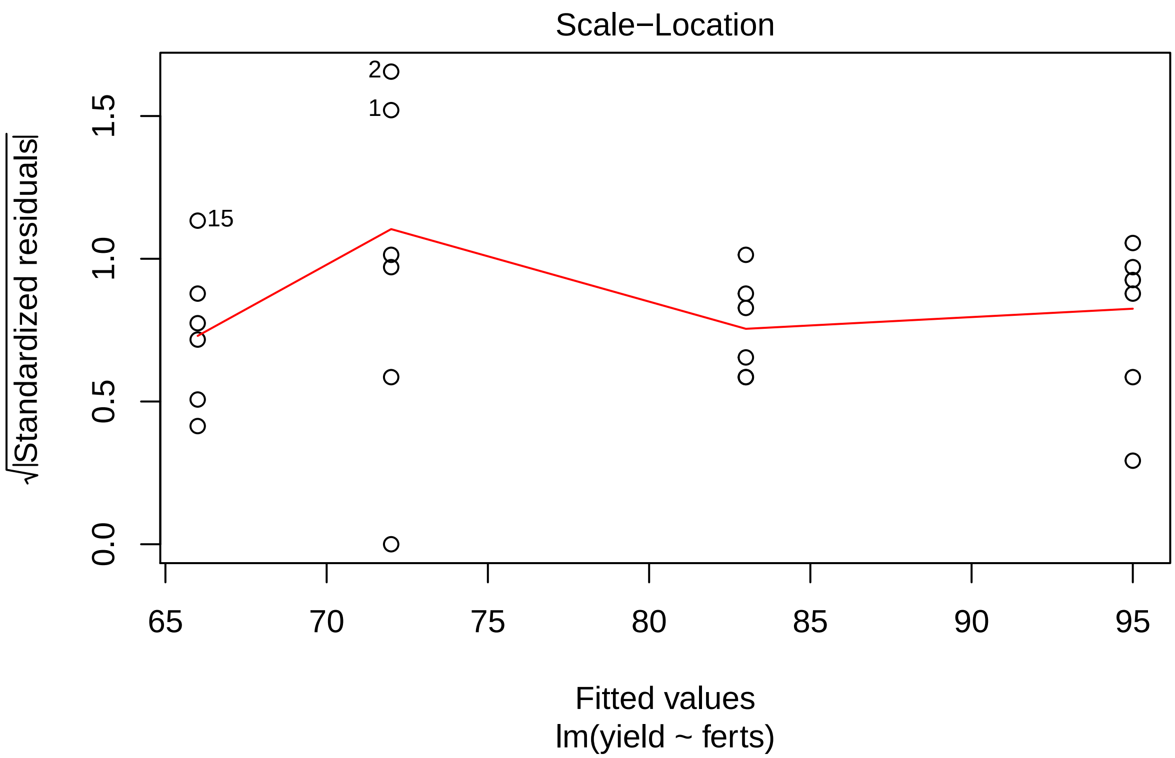
```
##
## Call:
## lm(formula = yield ~ ferts)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -32.00  -7.50   0.50   8.25  27.00
##
## Coefficients: (1 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   83.000      5.222  15.895 8.23e-13 ***
## fertsfert1   -11.000      7.385  -1.490   0.1519
## fertsfert2    12.000      7.385   1.625   0.1198
## fertsfert3   -17.000      7.385  -2.302   0.0322 *
## fertsfert4        NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.79 on 20 degrees of freedom
## Multiple R-squared:  0.4733, Adjusted R-squared:  0.3943
## F-statistic:  5.99 on 3 and 20 DF,  p-value: 0.004387
```

```r
# Using treatment coding, the intercept becomes the mean for the group for which we don't fit
# an indicator variable - this is called the reference group. In this instance, a better choice
# for the reference group is the Control group, so we should delete fert1 rather than fert4:

ferts <- cbind(fert2, fert3, fert4)
corn.lm2a <- lm(yield ~ ferts)
plot(corn.lm2a)
```






```
## hat values (leverages) are all = 0.1666667
##  and there are no factor predictors; no plot no. 5
```




```r
anova(corn.lm2a)
```

```
## Analysis of Variance Table
##
## Response: yield
##           Df Sum Sq Mean Sq F value   Pr(>F)
## ferts      3   2940   980.0  5.9902 0.004387 **
## Residuals 20   3272   163.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(corn.lm2a)
```

```
##
## Call:
## lm(formula = yield ~ ferts)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -32.00  -7.50   0.50   8.25  27.00
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   72.000      5.222  13.788 1.13e-11 ***
## fertsfert2    23.000      7.385   3.115  0.00546 **
## fertsfert3    -6.000      7.385  -0.812  0.42607
## fertsfert4    11.000      7.385   1.490  0.15194
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.79 on 20 degrees of freedom
```

```
## Multiple R-squared:  0.4733, Adjusted R-squared:  0.3943
## F-statistic:  5.99 on 3 and 20 DF,  p-value: 0.004387
```

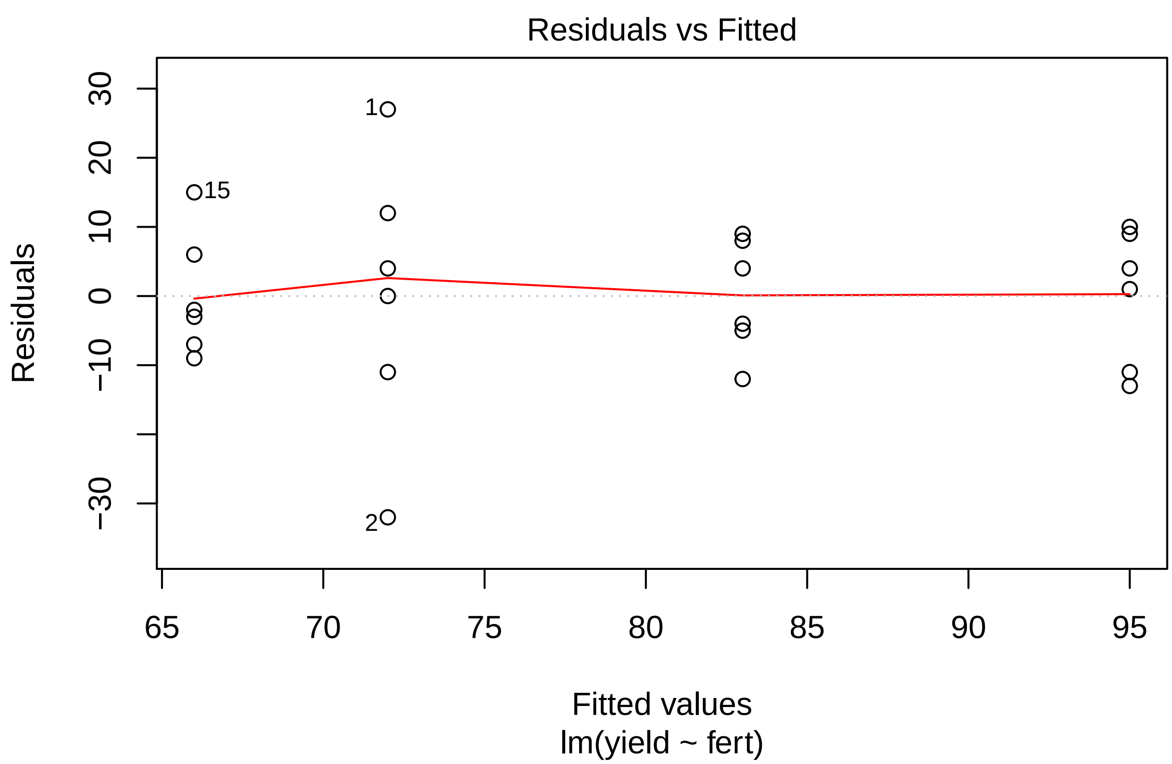
```r
# If we remember the level means calculated earlier, we can see that under the treatment
# parameterisation, the intercept is the mean for the "Control" group and the other parameters
# are the deviations away from this mean to get to the other group means:

lvl.mns
```

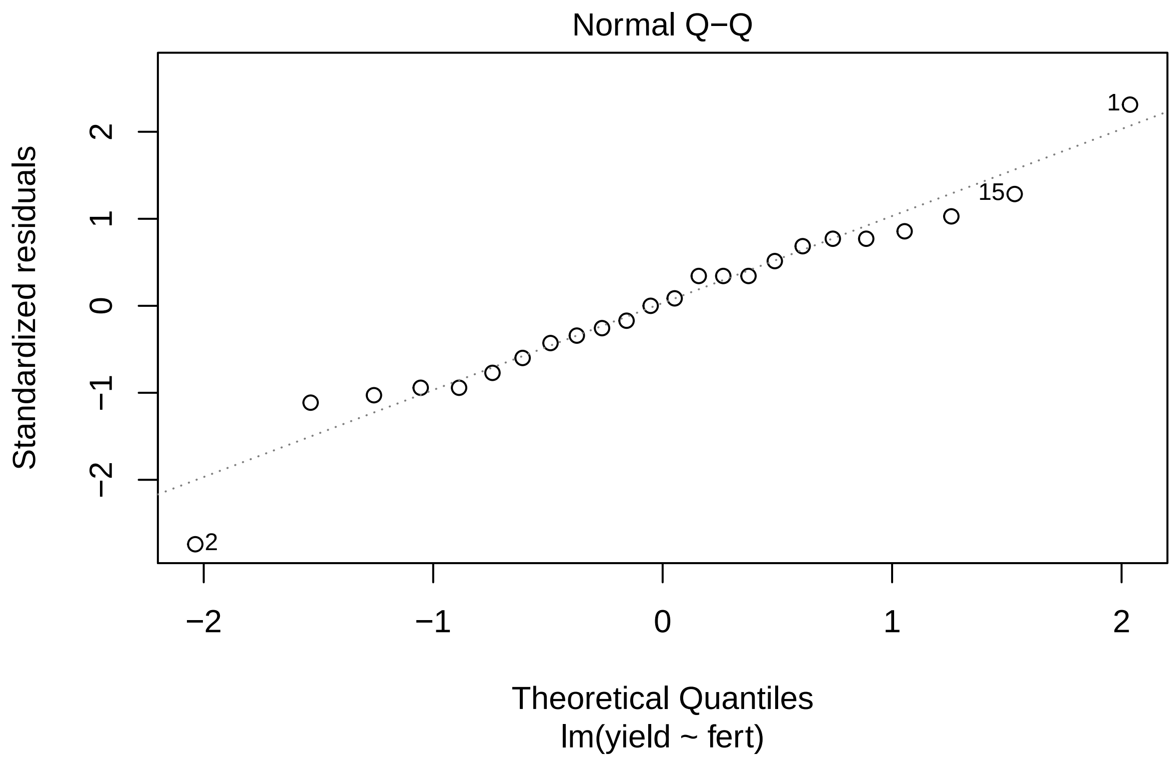
```
##   Control     K20+N K20+P205    N+P205
##        72        95       66        83
```

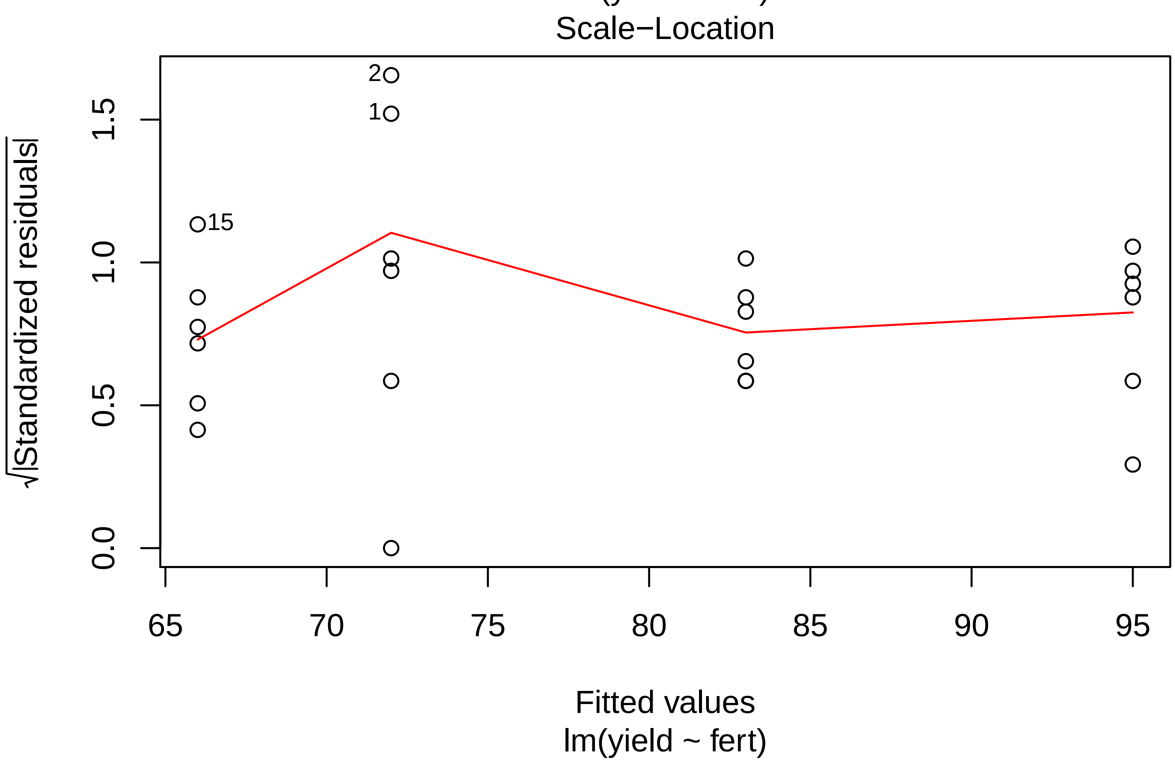
```r
lvl.mns - lvl.mns["Control"]
```

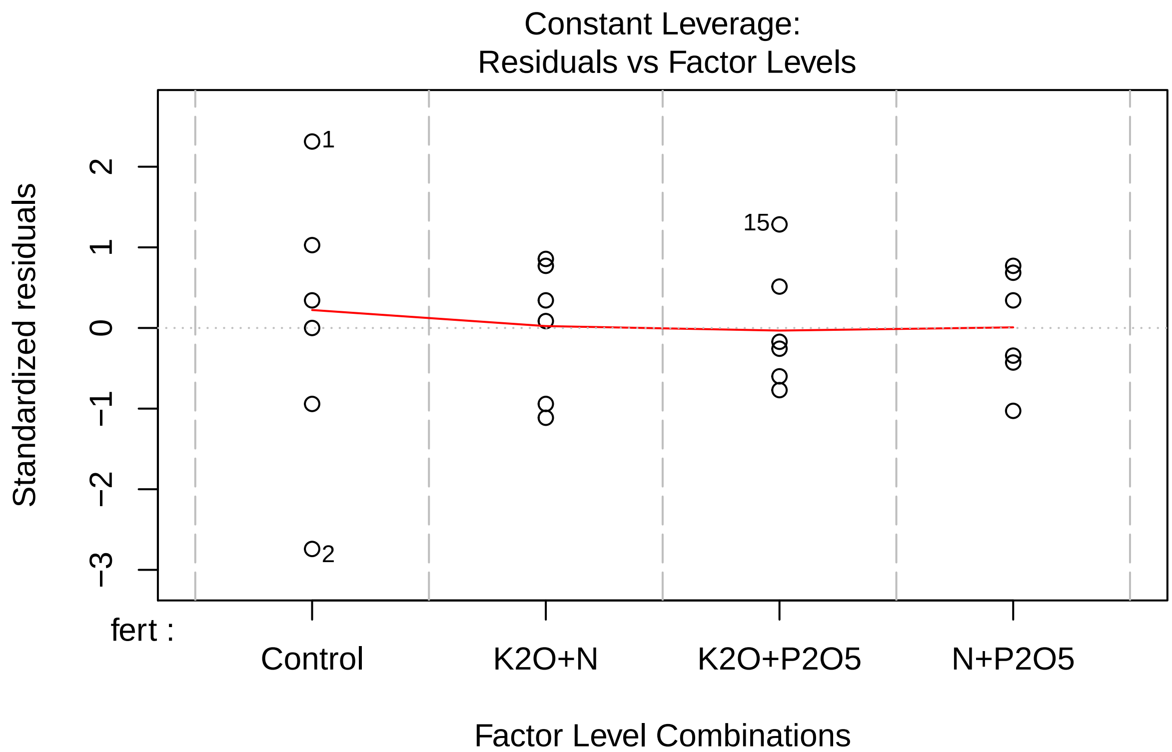
```
##   Control     K20+N K20+P205    N+P205
##         0        23       -6        11
```

```r
# Another approach is the "sum" parameterisation, which also has a sensible interpretation:

corn.lm3 <- lm(yield ~ fert, contrasts=list(fert=contr.sum))
plot(corn.lm3)
```




Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(yield ~ fert)
1
15
2


Scale−Location
√|Standardized residuals|
Fitted values
lm(yield ~ fert)
2
1
15




```r
anova(corn.lm3)
```

```
## Analysis of Variance Table
##
## Response: yield
##           Df Sum Sq Mean Sq F value   Pr(>F)
## fert       3   2940   980.0  5.9902 0.004387 **
## Residuals 20   3272   163.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(corn.lm3)
```

```
##
## Call:
## lm(formula = yield ~ fert, contrasts = list(fert = contr.sum))
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -32.00  -7.50   0.50   8.25  27.00
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   79.000      2.611  30.258  < 2e-16 ***
## fert1         -7.000      4.522  -1.548  0.13732
## fert2         16.000      4.522   3.538  0.00206 **
## fert3        -13.000      4.522  -2.875  0.00937 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.79 on 20 degrees of freedom
```

```
## Multiple R-squared:  0.4733, Adjusted R-squared:  0.3943
## F-statistic:  5.99 on 3 and 20 DF,  p-value: 0.004387
```

```
# Here the intercept is the overall mean and the other parameters are deviations away from the
# overall mean to the first three level means (in alphabetical order).  The "sum"
# parameterisation is equivalent to applying the constraint that the (weighted) deviations for
# the four groups sum to 0, so the deviation for the fourth group can be found by subtraction:

mean(yield)
```

```
## [1] 79
```

```
lvl.mns - mean(yield)
```

```
##  Control    K2O+N K2O+P2O5   N+P2O5
##       -7       16      -13        4
```

```
coef(corn.lm3)
```

```
## (Intercept)       fert1       fert2       fert3
##          79          -7          16         -13
```

```
coef(corn.lm3)[2:4]
```

```
## fert1 fert2 fert3
##    -7    16   -13
```

```
-sum(coef(corn.lm3)[2:4])
```

```
## [1] 4
```

```
# Note that this constraint is sum(taui) = 0 not sum(ni * taui) = 0 from the notes.  This will
# not make a difference if the design is balanced, although if the design is unbalanced the
# contr.sum function cannot be used to give the grand mean interpretation. In this case the
# coefficients for the fourth group should be:
# (-ni[1]/ni[4],-ni[2]/ni[4],-ni[3]/ni[4]) and not (-1,-1,-1).

contr.sum(4)
```

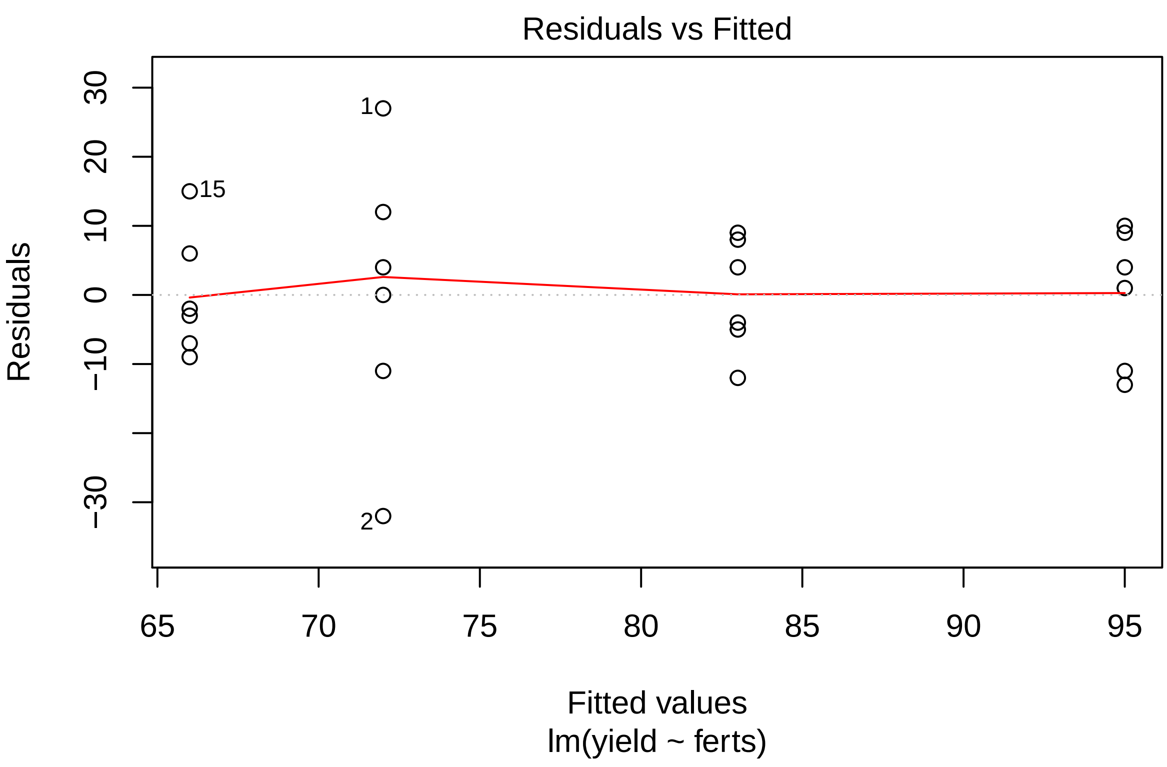
```
##   [,1] [,2] [,3]
## 1    1    0    0
## 2    0    1    0
## 3    0    0    1
## 4   -1   -1   -1
```

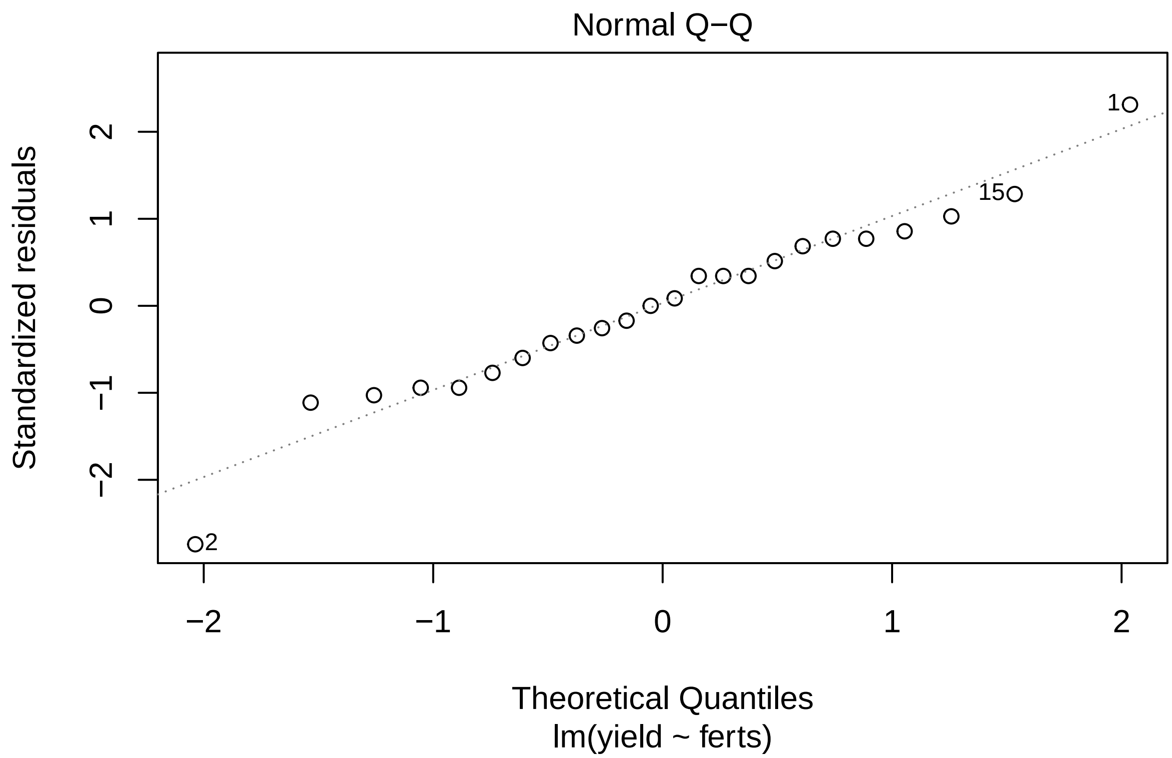
```
# We could also do a manual equivalent of this "sum" parameterisation:

fert1 <- c(rep(1,6), rep(0,12), rep(-1,6))
fert2 <- c(rep(0,6), rep(1,6), rep(0,6), rep(-1,6))
fert3 <- c(rep(0,12), rep(1,6), rep(-1,6))
ferts <- cbind(fert1, fert2, fert3)
ferts
```

```
##       fert1 fert2 fert3
##  [1,]     1     0     0
##  [2,]     1     0     0
##  [3,]     1     0     0
##  [4,]     1     0     0
##  [5,]     1     0     0
##  [6,]     1     0     0
```

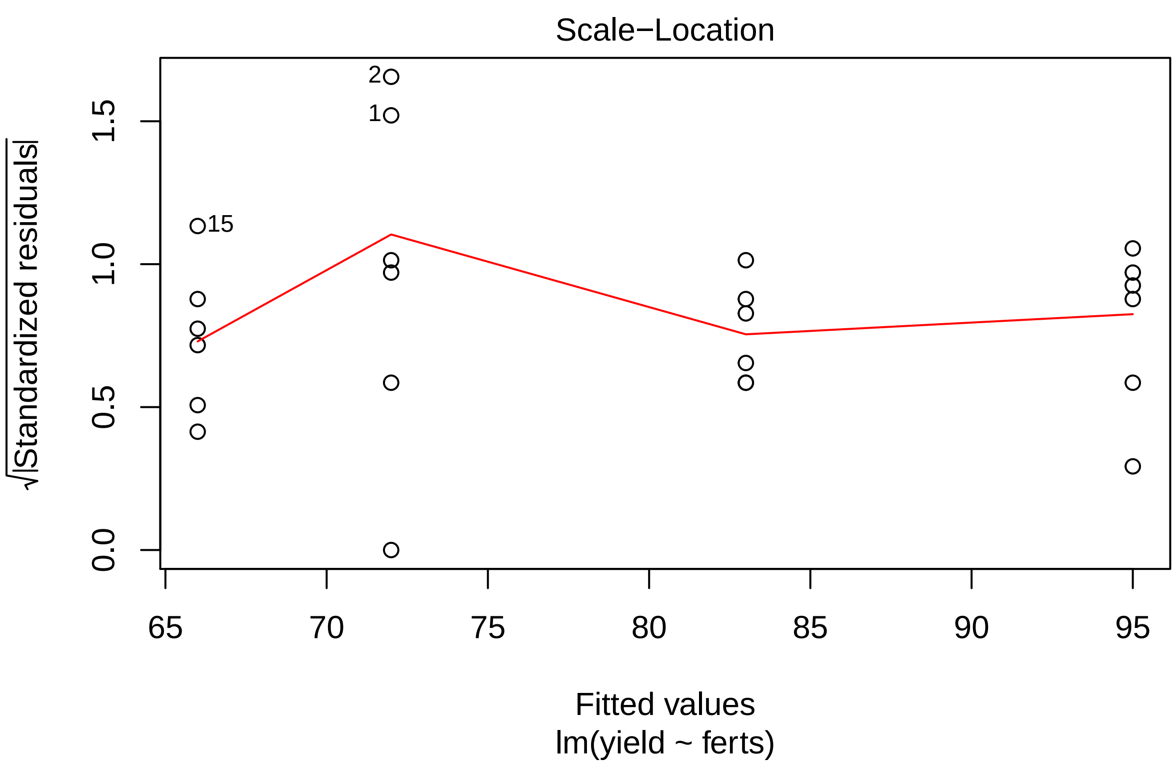
```
##  [7,]    0    1    0
##  [8,]    0    1    0
##  [9,]    0    1    0
## [10,]    0    1    0
## [11,]    0    1    0
## [12,]    0    1    0
## [13,]    0    0    1
## [14,]    0    0    1
## [15,]    0    0    1
## [16,]    0    0    1
## [17,]    0    0    1
## [18,]    0    0    1
## [19,]   -1   -1   -1
## [20,]   -1   -1   -1
## [21,]   -1   -1   -1
## [22,]   -1   -1   -1
## [23,]   -1   -1   -1
## [24,]   -1   -1   -1
```

```r
corn.lm3a <- lm(yield ~ ferts)
plot(corn.lm3a)
```






```
## hat values (leverages) are all = 0.1666667
##  and there are no factor predictors; no plot no. 5
```




```
anova(corn.lm3a)
```

```
## Analysis of Variance Table
```

```
##
## Response: yield
##           Df Sum Sq Mean Sq F value   Pr(>F)
## ferts      3   2940   980.0  5.9902 0.004387 **
## Residuals 20   3272   163.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(corn.lm3a)
```

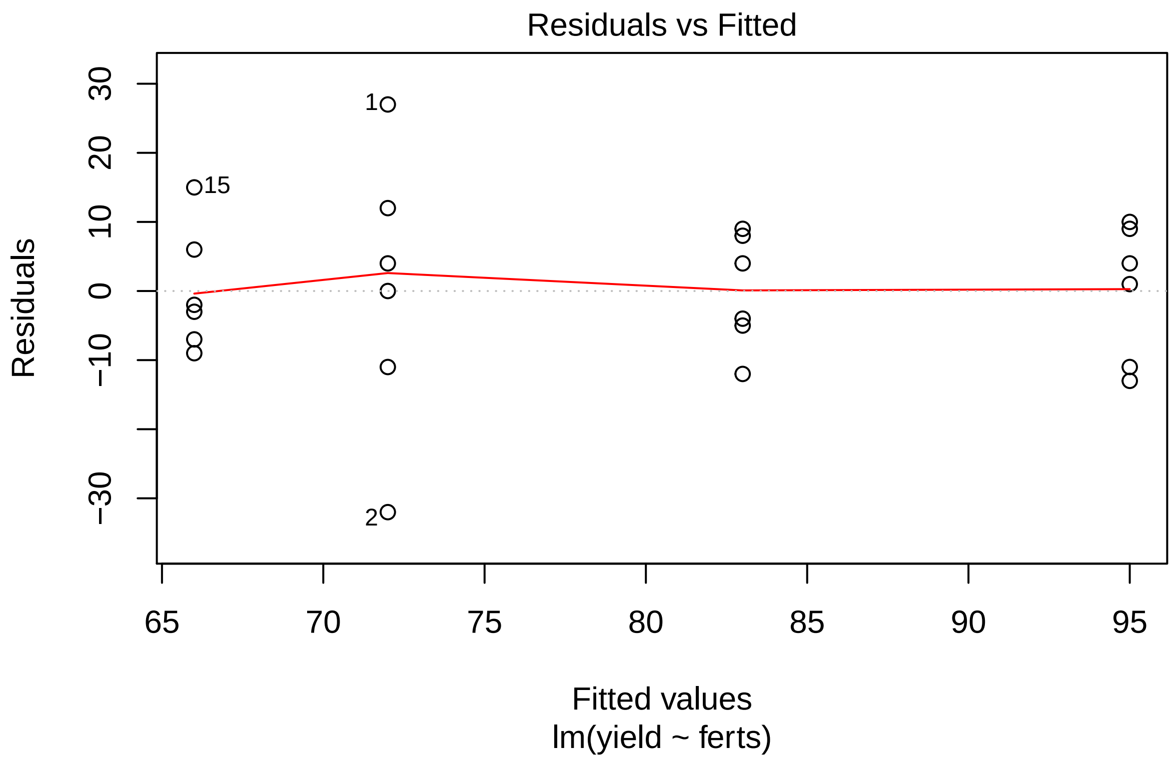
```
##
## Call:
## lm(formula = yield ~ ferts)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -32.00  -7.50   0.50   8.25  27.00
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   79.000      2.611  30.258  < 2e-16 ***
## fertsfert1    -7.000      4.522  -1.548  0.13732
## fertsfert2    16.000      4.522   3.538  0.00206 **
## fertsfert3   -13.000      4.522  -2.875  0.00937 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.79 on 20 degrees of freedom
## Multiple R-squared:  0.4733, Adjusted R-squared:  0.3943
## F-statistic:  5.99 on 3 and 20 DF,  p-value: 0.004387
```

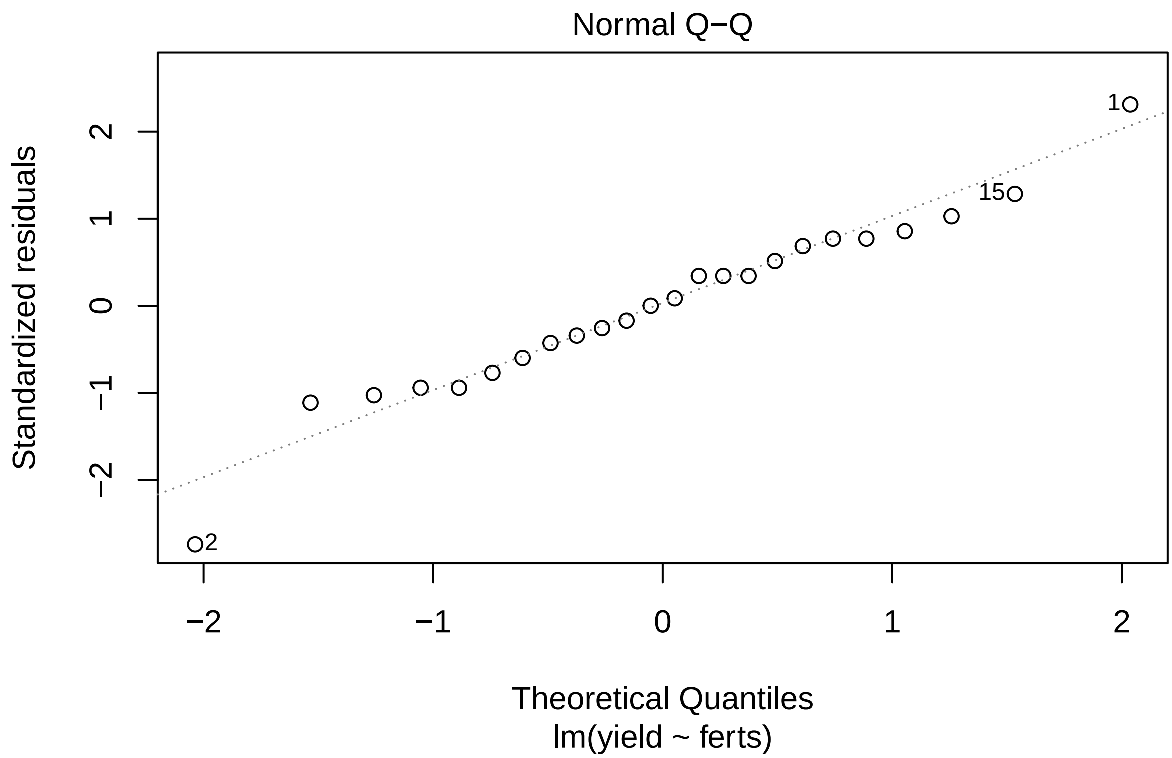
```
# By default, R gives explicit parameters for all but the last of the four groups.  If we are
# not happy with this approach, we can force R to use a different group as the reference group:

fert2 <- c(rep(-1,6), rep(1,6), rep(0,12))
fert3 <- c(rep(-1,6), rep(0,6), rep(1,6), rep(0,6))
fert4 <- c(rep(-1,6), rep(0,12), rep(1,6))
ferts <- cbind(fert2, fert3, fert4)
ferts
```

```
##       fert2 fert3 fert4
##  [1,]    -1    -1    -1
##  [2,]    -1    -1    -1
##  [3,]    -1    -1    -1
##  [4,]    -1    -1    -1
##  [5,]    -1    -1    -1
##  [6,]    -1    -1    -1
##  [7,]     1     0     0
##  [8,]     1     0     0
##  [9,]     1     0     0
## [10,]     1     0     0
## [11,]     1     0     0
## [12,]     1     0     0
## [13,]     0     1     0
## [14,]     0     1     0
## [15,]     0     1     0
```

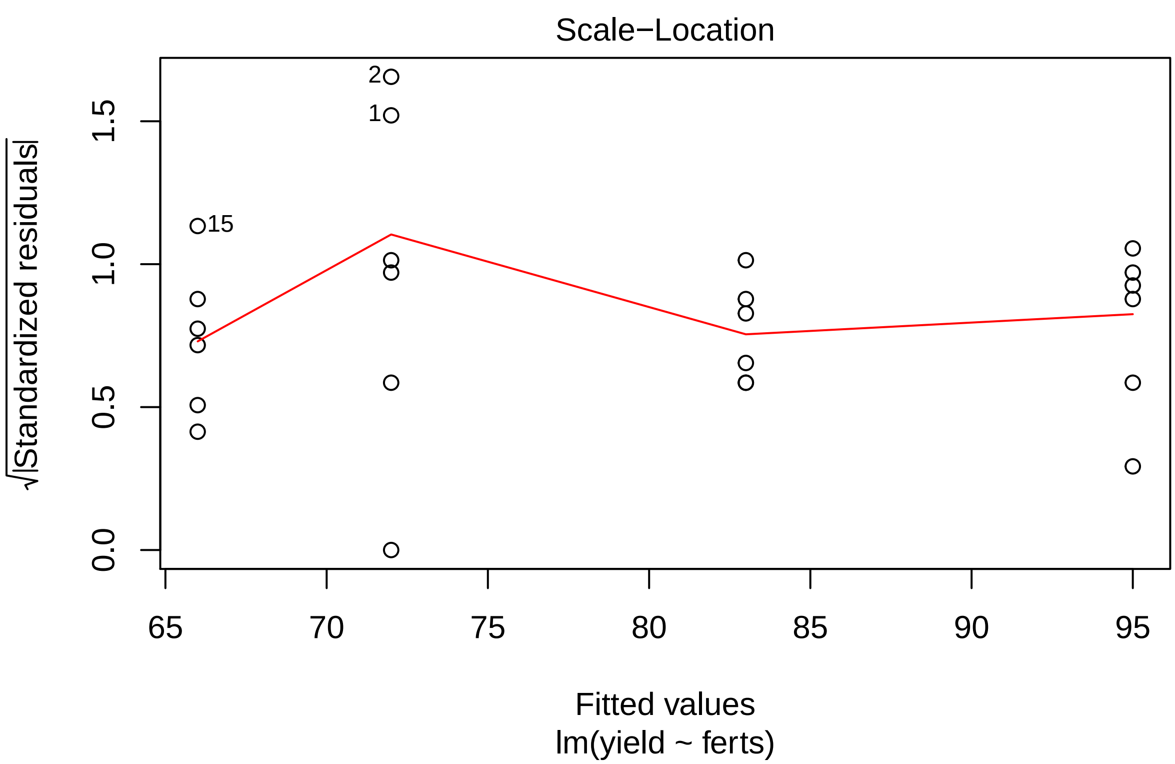
```
## [16,]    0    1    0
## [17,]    0    1    0
## [18,]    0    1    0
## [19,]    0    0    1
## [20,]    0    0    1
## [21,]    0    0    1
## [22,]    0    0    1
## [23,]    0    0    1
## [24,]    0    0    1
```

```
corn.lm3b <- lm(yield ~ ferts)
plot(corn.lm3b)
```






```
## hat values (leverages) are all = 0.1666667
##  and there are no factor predictors; no plot no. 5
```



```r
anova(corn.lm3b)
```

```
## Analysis of Variance Table
##
## Response: yield
##           Df Sum Sq Mean Sq F value   Pr(>F)
## ferts      3   2940   980.0  5.9902 0.004387 **
## Residuals 20   3272   163.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(corn.lm3b)
```

```
##
## Call:
## lm(formula = yield ~ ferts)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -32.00  -7.50   0.50   8.25  27.00
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   79.000      2.611  30.258  < 2e-16 ***
## fertsfert2    16.000      4.522   3.538  0.00206 **
## fertsfert3   -13.000      4.522  -2.875  0.00937 **
## fertsfert4     4.000      4.522   0.885  0.38692
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.79 on 20 degrees of freedom
## Multiple R-squared:  0.4733, Adjusted R-squared:  0.3943
## F-statistic:  5.99 on 3 and 20 DF,  p-value: 0.004387
```

```
# Note that all the above models produce identical plots and ANOVA tables - they are basically
# all the same model,  all of which use essentially the same information coded in different ways.
# The differences between the models lie in how we use the different parameters to recover
# information about the level means.

# Is this model really appropriate? The outliers in the control group appear to be inflating
# the variance of that group, which may cause problems as we are assuming that the variance is
# constant, ie. the same for all four groups. The rule of thumb for assessing this is on page 4
# of the brick, the ratio of the largest to smallest group variance should not be greater than
# the number of groups:

lvl.vars <- tapply(yield, fert, var)
lvl.vars
```

```
##  Control    K20+N K20+P205   N+P205
##    406.8     97.6     80.8     69.2
```

```
max(lvl.vars)
```

```
## [1] 406.8
```

```
min(lvl.vars)
```

```
## [1] 69.2
```

```
max(lvl.vars)/min(lvl.vars)
```

```
## [1] 5.878613
```

```
# There are only 4 groups, so we have a problem (one which is simply ignored in the analysis in
# the brick).  If we do decide to include these outliers, we should certainly qualify any
# conclusions we make based on this analysis, as an important underlying assumption has been
# violated. Here the presence of two outliers in different directions may just inflate the
# overall estimate of the error variance, which would inflate the standard errors and therefore
# reduce the precision of any contrasts between the treatment groups, making our inference
# more conservative.

anova(corn.lm2)
```

```
## Analysis of Variance Table
##
## Response: yield
##           Df Sum Sq Mean Sq F value   Pr(>F)
## fert       3   2940   980.0  5.9902 0.004387 **
## Residuals 20   3272   163.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# The ANOVA table indicates a strong difference between the groups, but not which fertilizer
# treatments produce the best yields. Here are some alternative ways of doing the analysis
# shown at the top of page 8 of the brick:
```

```r
summary(corn.lm2)
```

```
##
## Call:
## lm(formula = yield ~ fert, contrasts = list(fert = contr.treatment))
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -32.00  -7.50   0.50   8.25  27.00
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   72.000      5.222  13.788 1.13e-11 ***
## fert2         23.000      7.385   3.115  0.00546 **
## fert3         -6.000      7.385  -0.812  0.42607
## fert4         11.000      7.385   1.490  0.15194
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.79 on 20 degrees of freedom
## Multiple R-squared:  0.4733, Adjusted R-squared:  0.3943
## F-statistic:  5.99 on 3 and 20 DF,  p-value: 0.004387
```

```r
# The basic results appear to be that fertilizer 1 "K20 + N", produces significantly
# higher corn yields than the "Control" group (which is presumably some default
# fertilizer regime or possibly no added fertilizer at all), whilst the other two
# groups do not produce significantly different yields from the "Control" group.

# These conclusions do not change, even if we apply an appropriate Bonferroni correction
# to account for the fact that are performing a series of 3 comparisons (i.e. we have
# to somehow overcome the problem of multiple comparisions):

names(summary(corn.lm2))
```

```
##  [1] "call"          "terms"         "residuals"     "coefficients"
##  [5] "aliased"       "sigma"         "df"            "r.squared"
##  [9] "adj.r.squared" "fstatistic"    "cov.unscaled"
```

```r
summary(corn.lm2)$sigma
```

```
## [1] 12.79062
```

```r
se <- summary(corn.lm2)$sigma*sqrt(1/6 + 1/6)
se
```

```
## [1] 7.384669
```

```r
est <- coef(corn.lm2)[2:4]
est
```

```
## fert2 fert3 fert4
##    23    -6    11
```

```r
lower <- est - qt(1 - 0.05/(2*3), 20)*se
lower
```

```
##      fert2      fert3      fert4
##   3.706922 -25.293078  -8.293078
```

```r
upper <- est + qt(1 - 0.05/(2*3), 20)*se
upper
```

```
##    fert2    fert3    fert4
## 42.29308 13.29308 30.29308
```

```r
cbind(lower, est, upper)
```

```
##            lower est    upper
## fert2   3.706922  23 42.29308
## fert3 -25.293078  -6 13.29308
## fert4  -8.293078  11 30.29308
```

```r
# Different research questions would call for different analyses and therefore different
# interpretations of the results. Here is the analysis at the bottom of page 7 of the brick:

ni <- tapply(yield, fert, length)
ni
```

```
##  Control    K2O+N K2O+P2O5   N+P2O5
##        6        6        6        6
```

```r
h <- c(-1, 1/3, 1/3, 1/3)
h
```

```
## [1] -1.0000000  0.3333333  0.3333333  0.3333333
```

```r
est <- t(h) %*% lvl.mns
est
```

```
##          [,1]
## [1,] 9.333333
```

```r
MSE <- sum((yield-fitted(corn.lm2))^2)/corn.lm2$df.residual
MSE
```

```
## [1] 163.6
```

```r
se <- sqrt(MSE)*sqrt(sum((h^2)/ni))
lower <- est - qt(0.975, corn.lm2$df.residual)*se
upper <- est + qt(0.975, corn.lm2$df.residual)*se
c(lower, est, upper)
```

```
## [1] -3.244102  9.333333 21.910768
```

```r
# Finally, here's the analysis from the bottom of page 8. Please read the discussion of
# these results in the brick.

h <- c(0, 0.5, -1, 0.5)
est <- t(h) %*% lvl.mns
se <- sqrt(MSE)*sqrt(sum((h^2)/ni))
lower <- est - qt(0.975, corn.lm2$df.residual)*se
upper <- est + qt(0.975, corn.lm2$df.residual)*se
c(lower, est, upper)
```

```
## [1]  9.659615 23.000000 36.340385
```