Statistics 5401
14. Univariate Linear Models

Gary W. Oehlert
School of Statistics
313B Ford Hall
612-625-1557
gary@stat.umn.edu

Linear models relate a *target* or *response* or *dependent* variable y to known *predictor* or *independent* variables x_j, unknown *parameters* β_j, and *random variation*.

$$y = \text{predictable part} + \text{random variation}$$

The predictable part is a function of the predictor variables and the parameters.
Because this is a linear model, the parameters enter the predictable part linearly:

$$\text{predicatable part} = f(x)'\beta$$

where β is the vector of unknown parameters and $f(x)$ is some (vector) function of the predictor variables.
Many well known examples.
Multiple regression.

$$y_i = (x_{i0}\beta_0 + x_{i1}\beta_1 + \ldots + x_{ik}\beta_k) + \{\epsilon_i\}$$

Usually, $x_{i0} \equiv 1$.
Here the part in parentheses is the predictable part, and the part in braces is the unpredictable part.
One-way ANOVA, g-group means.

$$y_{ij} = (\mu_i) + \{\epsilon_{ij}\}$$

or

$$y_{ij} = (\mu + \alpha_i) + \{\epsilon_{ij}\}$$

with $\sum_{i=1}^g \alpha_i = 0$ or a similar restriction.
This can be rewritten as a multiple regression in several ways.
Nested random effects.

$$y_{ijk} = (\mu) + \{A_i + B_{ij} + \epsilon_{ijk}\}$$

Here the only predictable part is the overall mean. The other terms are random, and because all y_{ijk}s with the same i share the same A_i, and all y_{ijk}s with the same i, j share the same B_{ij}, there is correlation among the responses.
Randomized complete block.

$$y_{ij} = (\mu + \alpha_i) + \{B_j + \epsilon_{ij}\}$$

with $\sum_{i=1}^g \alpha_i = 0$ or a similar restriction. This assumes that the block effects B_j are random.
This is a special case of a profile analysis, where we know ahead of time that the correlations are $\sigma_B^2/(\sigma_B^2 + \sigma^2)$.
In particular, the distribution of $\mathbf{C}y$ does not depend on σ_B^2.
Analysis of Covariance. This combines regression and ANOVA-type predictor terms.

$$y_{ij} = (\mu + \alpha_i + x_{ij}\beta) + \{\epsilon_{ij}\}$$

with $\sum_{i=1}^g \alpha_i = 0$ or a similar restriction. This is a model with parallel lines, with the slope β and different intercepts from the different α_is.

There are many more fancier ANOVA-type structures, including factorials, split plots, and so on. All can be written as linear models.

In all cases, if we write all the responses in one vector y, all the parameters in one vector β and all the predicting variables in one matrix $\mathbf{X}$, then

$$y = \mathbf{X}\beta + \epsilon$$

where $\mathbf{X}\beta$ is predictable, and ϵ is not predictable. The elements of ϵ may be correlated.

We can write the predictable part in many ways. That is,

$$\mathbf{X}\beta = \mathbf{X}^\star\beta^\star$$

for lots of different $\mathbf{X}^\star$ and $\beta^\star$ pairs.

In one-way ANOVA, we could write μ_i or $\mu + \alpha_i$.

In regression, we could replace x_{i1} and x_{i2} with $(x_{i1} + x_{i2})$ and $(x_{i1} - x_{i2})$ (and modified coefficients).

In general, the value of the predictable part is well defined, but the expression as independent variables and parameters is pretty arbitrary.

We have a linear model. The parameters enter linearly, and the unpredicatable term is added to the predictable term.

We also want to test *linear hypotheses* about the parameters. Let β be $r \times 1$, and let $\mathbf{L}$ be $f_h \times r$ of full rank. We want to test

$$H_0 : \mathbf{L}\beta = 0$$

versus

$$H_0 : \mathbf{L}\beta \neq 0$$

Important note: if $\mathbf{X}$ is not full rank, then some linear combinations $\ell'\beta$ are not well defined without further restrictions.

For example, consider the one-way model $\mu + \alpha_i$. It doesn't make sense to look at α_1, because we can add 10 to μ and subtract 10 from α_i and not change the predictable part.

We are OK if $\ell = \mathbf{X}'\gamma$, that is, if ℓ is a linear combination of the rows of $\mathbf{X}$. In this case, the linear combination is *estimable*.

In the one-way model, if the coefficient for μ equals the sum of the coefficients for the α_is, then $\ell'\beta$ is estimable in the one-way model.

Examples. Multiple regression with a constant plus four predictors.

$\text{H}_0 : \beta_2 = 0$ has $f_h = 1$ and corresponds to

$$\mathbf{L} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

$\text{H}_0 : \beta_2 = \beta_3 = 0$ has $f_h = 2$ and corresponds to

$$\mathbf{L} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$\text{H}_0 : \beta_2 - \beta_3 = 0$ has $f_h = 1$ and corresponds to

$$\mathbf{L} = \begin{bmatrix} 0 & 0 & 1 & -1 & 0 \end{bmatrix}$$

One-way ANOVA.

$H_0 : \alpha_1 = \alpha_2 = \ldots = \alpha_g = 0$ has $f_h = g - 1$ and corresponds to ($g = 4$ groups)

$$\mathbf{L} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

or

$$\mathbf{L} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

because the sum of the α_is is fixed at zero.

Least Squares. Estimation by least squares finds the estimate $\mathbf{b}$ of β that minimizes the sum of squared differences between the observed data and the fitted values using $\mathbf{b}$. Least squares estimation is also maximum likelihood estimation for independent, normally distributed errors.

The sum of squared differences is often referred to as the *residual sum of squares* RSS, or the *sum of squares for error* SS_E.

Let $\mathbf{b}^0$ be the estimate of β when the null is assumed to be true. That is, the vector that minimizes RSS subject to $\mathbf{Lb} = 0$. Call the RSS under the null RSS(H_0).

Let $\mathbf{b}^1$ be the estimate of β when the alternative is assumed to be true. That is, the vector that minimizes RSS without restrictions. Call the RSS under the alternative RSS(H_1), or SS_E.

RSS(H_0) and RSS(H_1) do not depend on the parameterization we choose (the $\mathbf{b}$'s depend on the parameterization, but not the sums of squares). Thus we can always use the most convenient parameterization.

Define

$$SS_H = RSS(H_0) - RSS(H_1)$$

This is the increase in RSS when going from the null fit to the alternative fit.

Large values of SS_H imply that the alternative fits much better than the null, thus implying that the null should be rejected. Specifically, we look at the ratio SS_H/SS_E and reject for large values. Under the null (with normality)

$$\frac{SS_H}{SS_E} \sim \frac{f_h}{f_e} F_{f_h,\, f_e}$$

Of course, this is just the usual F test with the degrees of freedom multiplying the F distribution instead of scaling sums of squares into mean squares.

The likelihood ratio test is

$$\Lambda = \left(\frac{RSS(H_0)}{RSS(H_1)}\right)^{-n/2} = \left(1 + \frac{SS_H}{SS_E}\right)^{-n/2}$$

For large samples under the null, SS_E should be much bigger than SS_H, so

$$\chi^2_{f_e} \sim -2 \ln \Lambda = n \ln \left(1 + \frac{SS_H}{SS_E}\right) \approx n \frac{SS_H}{SS_E}$$

which agrees asymptotically with the F test.

Here is a shortcut(?). Suppose that the (estimated) variance matrix for $\mathbf{b}$ is $s^2\mathbf{C}$. Then the sum of squares for the hypothesis

$$H_0 : \mathbf{L}\beta = 0$$

is

$$SS_H = (\mathbf{Lb})'(\mathbf{LCL}')^{-1}(\mathbf{Lb})$$

This is very like a Mahalanobis distance.
A regression example. The actual data follow a quadratic, and we'll try to fit a cubic.

```
Cmd> x <- run(20)

Cmd> x2 <- x*x

Cmd> x3 <- x*x*x

Cmd> setseeds(12224,546778)

Cmd> y <- 3 + 2*x - x2/100 + rnorm(20)

Cmd> regress("y=x+x2+x3")
Model used is y=x+x2+x3
                 Coef       StdErr            t
CONSTANT       2.7427       1.1934       2.2981
x              1.8668      0.48019       3.8875
x2            0.02275      0.05246      0.43367
x3         -0.0014094    0.0016447     -0.85696

N: 20, MSE: 1.1937, DF: 16, R^2: 0.99120
Regression F(3,16): 600.39, Durbin-Watson: 1.5525
To see the ANOVA table type 'anova()'

Cmd> anova()
Model used is y=x+x2+x3
WARNING: summaries are sequential
            DF           SS           MS
CONSTANT     1        10126        10126
x            1       2140.9       2140.9
x2           1       8.2267       8.2267
x3           1      0.87661      0.87661
ERROR1      16       19.099       1.1937

Cmd> .87661/1.1937
(1)     0.73436

Cmd> .85696*.85696
(1)     0.73438

Cmd> (8.2267+.87661)
(1)      9.1033

Cmd> 9.1033/2
```

```
(1)       4.5517

Cmd> 4.5517/1.1937
(1)       3.8131

Cmd> 1-cumF(3.813,2,16)
(1)     0.044242

Cmd> COEF
 CONSTANT         x        x2        x3
   2.7427    1.8668   0.02275  -0.00141

Cmd> SS
 CONSTANT         x        x2        x3   ERROR1
    10126    2140.9    8.2267   0.87661   19.099

Cmd> DF
 CONSTANT         x        x2        x3   ERROR1
        1         1         1         1       16

Cmd> XTXINV
              CONSTANT            x           x2           x3
CONSTANT        1.1932      -0.4343     0.042312    -0.001204
x              -0.4343      0.19317    -0.020548   0.00061435
x2            0.042312    -0.020548    0.0023055  -7.1383e-05
x3           -0.001204   0.00061435  -7.1383e-05   2.2661e-06

Cmd> c <- XTXINV[run(3,4),run(3,4)]

Cmd> lb <- COEF[run(3,4)];lb
           x2           x3
      0.02275   -0.0014094

Cmd> c
             x2             x3
x2    0.0023055  -7.1383e-05
x3  -7.1383e-05   2.2661e-06

Cmd> lb'%*%solve(c)%*%lb
             (1)
(1)       9.1034

Cmd> anova("y=x+x2+x3")
Model used is y=x+x2+x3
WARNING: summaries are sequential
                DF            SS           MS
```

```
CONSTANT          1       10126       10126
x                 1      2140.9      2140.9
x2                1      8.2267      8.2267
x3                1     0.87661     0.87661
ERROR1           16      19.099      1.1937

Cmd> anova("y=x+x2+x3",fstats:T)
Model used is y=x+x2+x3
WARNING: summaries are sequential
                  DF          SS          MS           F     P-value
CONSTANT           1       10126       10126  8482.63112           0
x                  1      2140.9      2140.9  1793.55009           0
x2                 1      8.2267      8.2267     6.89193    0.018373
x3                 1     0.87661     0.87661     0.73438     0.40412
ERROR1            16      19.099      1.1937
```

One-way ANOVA with five groups.

```
Cmd> a <- factor(rep(run(5),4))

Cmd> y <- vector(3,1,6,4,5)[a]+rnorm(20)

Cmd> anova("y=a")
Model used is y=a
                 DF          SS          MS
CONSTANT          1      277.99      277.99
a                 4       78.37      19.592
ERROR1           15      10.529     0.70191

Cmd> coefs()
component: CONSTANT
(1)       3.7282
component: a
(1)  0.059304   -3.2705    2.7802  -0.51608   0.94706

Cmd> coefs("a",se:T)
component: coefs
(1)  0.059304   -3.2705    2.7802  -0.51608   0.94706
component: se
(1)   0.37468   0.37468   0.37468   0.37468   0.37468

Cmd> contrast("a",vector(1,1,1,-1.5,-1.5))
component: estimate
(1)      -1.0775
component: ss
(1)      0.61915
component: se
(1)       1.1472
```

Analysis of covariance.

```
Cmd> y <- vector(3,1,6,4,5)[a]+x/2+rnorm(20)

Cmd> anova("y=x+a",pvals:T)
Model used is y=x+a
WARNING: summaries are sequential
             DF        SS        MS    P-value
CONSTANT      1    1685.4    1685.4          0
x             1    157.68    157.68 4.6626e-08
a             4    63.095    15.774 0.00027676
ERROR1       14    19.746    1.4104

Cmd> anova("y=a+x")
Model used is y=a+x
WARNING: summaries are sequential
                DF          SS            MS
CONSTANT         1      1685.4        1685.4
a                4       101.6          25.4
x                1      119.17        119.17
ERROR1          14      19.746        1.4104

Cmd> anova("y=x+a",marginal:T)
Model used is y=x+a
WARNING: SS are Type III sums of squares
                DF          SS            MS
CONSTANT         1      93.258        93.258
x                1      119.17        119.17
a                4      63.095        15.774
ERROR1          14      19.746        1.4104
```

`anova()` creates several variables as side effects.

```
Cmd> SS
  CONSTANT          x          a    ERROR1
    93.258     119.17     63.095    19.746

Cmd> DF
  CONSTANT          x          a    ERROR1
         1          1          4        14

Cmd> RESIDUALS
 (1)      0.72573      0.19843      0.73128      0.72575       1.4661
 (6)      -1.0188     -0.48538      -2.1221      -1.6932      -1.7946
(11)    -0.053961     -0.22043       1.3752       1.2618      0.32385
(16)      0.34705      0.50739     0.015678     -0.29436      0.00465
```

```
Cmd> HII
 (1)         0.34         0.34         0.34         0.34         0.34
 (6)         0.26         0.26         0.26         0.26         0.26
(11)         0.26         0.26         0.26         0.26         0.26
(16)         0.34         0.34         0.34         0.34         0.34
```

```
Cmd> COEF
UNDEFINED
```

`regress()` creates `COEF`, but `anova()` does not.
What else can you extract?

```
Cmd> out <- modelinfo(all:T)

Cmd> compnames(out)
 (1) "xvars"
 (2) "y"
 (3) "parameters"
 (4) "xtxinv"
 (5) "coefs"
 (6) "aliased"
 (7) "scale"
 (8) "colcounts"
 (9) "weights"
(10) "strmodel"
(11) "bitmodel"
(12) "link"
(13) "distrib"
(14) "termnames"
(15) "sigmahat"
```

The **X** matrix.

```
Cmd> print(out$xvars,format:"f5.0")
MATRIX:
 (1,1)     1     1     1     0     0     0
 (2,1)     1     2     0     1     0     0
 (3,1)     1     3     0     0     1     0
 (4,1)     1     4     0     0     0     1
 (5,1)     1     5    -1    -1    -1    -1
...
(16,1)     1    16     1     0     0     0
(17,1)     1    17     0     1     0     0
(18,1)     1    18     0     0     1     0
(19,1)     1    19     0     0     0     1
(20,1)     1    20    -1    -1    -1    -1
```

```
Cmd> out$termnames
(1) "CONSTANT"
(2) "x"
(3) "a"
(4) "ERROR1"

Cmd> out$strmodel
(1) "y=1+x+a"

Cmd> out$colcounts
(1)            1            1            4

Cmd> print(out$xtxinv,format:"f8.3",labels:F)
MATRIX:
   0.226  -0.017  -0.034  -0.017   0.000   0.017
  -0.017   0.002   0.003   0.002  -0.000  -0.002
  -0.034   0.003   0.206  -0.047  -0.050  -0.053
  -0.017   0.002  -0.047   0.202  -0.050  -0.052
   0.000  -0.000  -0.050  -0.050   0.200  -0.050
   0.017  -0.002  -0.053  -0.052  -0.050   0.202
```