Regression Review for Multi-Level Modeling

Paul E. Johnson

1 Department of Political Science

2 Center for Research Methods and Data Analysis, University of Kansas

2015
The Linear model

Assume

\[ y_i = \beta_0 + \beta_1 x_i + \varepsilon_i, \ i = 1, \ldots, N \]  \hspace{1cm} (1)

where

- \( \beta_0 \) and \( \beta_1 \) are unknown real numbers, and
- \( \varepsilon_i \) is a random “disturbance”. Typically, we assume
  - Expected value is 0 \( E[\varepsilon_i] = 0 \) and
  - Homogeneous variance, \( Var[\varepsilon_i^2] \) is the same value for all \( i \). Usually, we call that \( \sigma_\varepsilon^2 \).
Most people don’t think about subscript $i$

Sub $i$ means the claim is true for any observation, so

- It is true for case number 1

$$y_1 = \beta_0 + \beta_1 x_1 + \varepsilon_1$$

- And case number 2

$$y_2 = \beta_0 + \beta_1 x_2 + \varepsilon_2$$

- And case number 467

$$y_{467} = \beta_0 + \beta_1 x_{467} + \varepsilon_{467}$$
The unknowns that must be estimated are $\beta_0$, $\beta_1$, AND $\sigma^2_{\varepsilon}$

Estimates $\hat{\beta}_0$, $\hat{\beta}_1$ of coefficients have hats

The **predicted value** (estimate of $E[y_i|x_i]$) also has a hat

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$$

One might consider the residuals, $y_i - \hat{y}_i$ as estimates of the error term $\varepsilon_i$
Everybody is interested in $\hat{\beta}_0$ and $\hat{\beta}_1$

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i, \quad i = 1, \ldots, N$$

- A 1 unit increase in $x_i$ is associated with a $\beta_1$ change in the expected value of $y_i$
We should be more interested in the error term

\[ y_i = \beta_0 + \beta_1 x_i + \varepsilon_i, \quad i = 1, \ldots, N \]

What do assume?

- The probability distribution of the error term has to be centered on 0.
- To repeat, the expected value of \( \varepsilon_i \) is 0, \( E[\varepsilon_i] = 0 \).
- What if \( E[\varepsilon_i] = 2 \)? Not a problem.
  - Squish those 2 units into the \( \beta_0 \)
  - and re-hypothesize a new error term \( \varepsilon_i - 2 \)
We should be more interested in the error term

We also assume,

- the draws from the $\varepsilon_i$ random data generator are homogeneous, they are from the SAME random number generating process.
- That’s usually summarized as “homogeneous variance”, or homoskedasticity. The error variance of case 1 is the same as case 2, is the same as case 467:
  - $\text{Var}[\varepsilon_i] = \sigma^2_\varepsilon$ (note same for all $i$)
Where do estimates come from?

- Ordinary Least Squares.
  - Adjust the values to minimize the sum of squared errors:

  \[
  \text{Choose } \hat{\beta}_j \text{ as Minimizers of } S(\hat{\beta}_0, \hat{\beta}_1) = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2
  \]

- The estimated variance of the error term, \( \hat{\sigma}_\varepsilon^2 \), often referred to as the Mean Squared Error (MSE)
- \( \sqrt{\sigma}_\varepsilon^2 \) is the Root MSE (RMSE), or sometimes “residual standard error” or just “sigma”
Theory: \( y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \varepsilon_i \)

Predicted value: \( \hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_{1i} + \hat{\beta}_2 x_{2i} \)

Objective function. Choose \( \hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2 \) to minimize

\[
S(\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2) = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2
\]

First Order Conditions:
Calculate partial derivative for each parameter, set them equal to 0 (finding the “bottom of the bowl”).

\[
\frac{\partial S}{\partial \beta_0} = -2 \sum (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_{1i} - \hat{\beta}_2 x_{2i}) = 0
\]

\[
\frac{\partial S}{\partial \beta_1} = -2 \sum (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_{1i} - \hat{\beta}_2 x_{2i}) x_{1i} = 0
\]

\[
\frac{\partial S}{\partial \beta_2} = -2 \sum (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_{1i} - \hat{\beta}_2 x_{2i}) x_{2i} = 0
\]
2 is a nice round number ...

These imply the **normal equations** (one for each estimated parameter):

\[
\begin{align*}
\sum y_i &= N\hat{\beta}_0 + (\sum x_1i)\hat{\beta}_1 + (\sum X_2i)\hat{\beta}_2 \\
\sum y_ix_1i &= (\sum x_1i)\hat{\beta}_0 + (\sum x_1^2i)\hat{\beta}_1 + (\sum x_1ix_2i)\hat{\beta}_2 \\
\sum y_ix_2i &= (\sum x_2i)\hat{\beta}_0 + (\sum x_1ix_2i)\hat{\beta}_1 + (\sum x_2^2i)\hat{\beta}_2
\end{align*}
\]

- One equation per parameter
- Note those sums are “just numbers” that come out of the data
- We can “do the math thing” to solve for the \(\hat{\beta}_j\), but it will take a lot of time. I’ve seen the solutions written out when there are 3 parameters to estimate, but never when there are more than 3.
- Estimate a regression with 10 variables, write out 10 equations? (ugh...)
The Design Matrix Has Many Columns

- Consider a multiple regression with a lot of numeric predictors

\[ y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \beta_3 X_{3i} + \beta_4 X_{4i} + \beta_5 X_{5i} + \beta_6 X_{6i} + \beta_7 X_{7i} + \varepsilon_i \]

- The “design matrix” has a column of 1’s plus a column for each variables.

\[
X = \begin{bmatrix}
\text{intercept} & X_1 & X_2 & X_3 & X_4 & X_5 & X_6 & X_7 \\
1 & 19 & 1 & 0.1 & 1 & 0 & 22 & 155 \\
1 & 22 & 2 & 1.1 & 0 & 1 & 42 & 199 \\
... & ... & ... & ... & ... & ... & ... & ... \\
1 & 8 & 4 & 0.2 & 1 & 1 & 77 & 77
\end{bmatrix}
\]

- **Design matrix**: numeric representation of all variables for which coefficient estimates are sought.
The Design Matrix Has Many Columns

- In R, run a multiple regression, something large like this, where X1, X2, and X3 are numeric, and X4 is categorical

\[
m1 \leftarrow \text{lm}(y \sim X1 + X2 + X3 + X4, \text{data} = \text{dat})
\]

- The regression program should manufactures a design matrix that has dummy variables for several categories of variable X4

- R’s model.matrix function can be used to review the design matrix, we see something like

\[
X = \begin{bmatrix}
\text{intercept} & X1 & X2 & X3 & X4cat1 & X4cat2 & X4cat3 & X4cat4 \\
1 & 19 & 1 & 0.1 & 1 & 0 & 0 & 0 \\
1 & 22 & 2 & 1.1 & 0 & 1 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
1 & 8 & 4 & 0.2 & 1 & 0 & 0 & 0 \\
\end{bmatrix}
\]

(4)
Regression Review for Multi-Level Modeling

Multiple Regression

Regression Assumptions

- Your challenge: Estimate each $\beta_j$ and $\sigma^2_e$

1. Specification:

$$y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \beta_3 X_3 + \beta_4 X_{4cat1i} + \beta_5 X_{4cat2i} + \beta_6 X_{4cat3i} + \beta_7 X_{4cat4i} + e_i$$

2. Error is an unmeasured variable with pleasant properties

   1. $E[e_i] = 0$, so that observed $y_i$ is scattered above and below “true” value
   2. homogeneous (same for all $i$) variance

   $$E[e_i] = 0, E[e_i^2] = \sigma^2_e.$$
Matrix Algebra

- There is a separate presentation about this named “matrices”
Regression in Matrices

\[ Y = X\beta + e \]

<table>
<thead>
<tr>
<th>dep. var</th>
<th>indep var</th>
<th>slopes</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ y = \begin{bmatrix} y_1 \ y_2 \ \vdots \ y_N \end{bmatrix} ]</td>
<td>[ X = \begin{bmatrix} 1 &amp; x_{11} &amp; x_{21} \ 1 &amp; x_{12} &amp; x_{22} \ \vdots &amp; \vdots &amp; \vdots \ 1 &amp; x_{1N} &amp; x_{2N} \end{bmatrix} ]</td>
<td>[ \hat\beta = \begin{bmatrix} \hat\beta_0 \ \hat\beta_1 \ \hat\beta_2 \end{bmatrix} ]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>residuals</th>
<th>predicted values</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ \hat{e} = y - \hat{y} = \begin{bmatrix} y_1 \ y_2 \ \vdots \ y_N \end{bmatrix} - \begin{bmatrix} \hat{y}_1 \ \hat{y}_2 \ \vdots \ \hat{y}_N \end{bmatrix} ]</td>
<td>[ \hat{y} = \begin{bmatrix} \hat{y}_1 \ \hat{y}_2 \ \vdots \ \hat{y}_N \end{bmatrix} = X\hat{\beta} ]</td>
</tr>
</tbody>
</table>
Regression in Matrices ...

- $y$ is a single column, the same as an $(N \times 1)$ matrix
- $X$ is an $(N \times p)$ rectangular matrix
- $e$ is $(N \times 1)$
- With 2 predictors,

$$y = X\beta + e$$

is short for:

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & x_{21} \\ 1 & x_{12} & x_{22} \\ \vdots & \vdots & \vdots \\ 1 & x_{1N} & x_{2N} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{bmatrix}$$
The assumption about the error term

\[ \text{Var}(e) = \begin{bmatrix} \sigma^2_e & 0 & 0 & 0 & 0 \\ 0 & \sigma^2_e & 0 & 0 \\ 0 & 0 & \sigma^2_e & 0 \\ 0 & 0 & 0 & \cdot & 0 \\ 0 & 0 & 0 & \cdot & \sigma^2_e \end{bmatrix} = \sigma^2_e \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \cdot & 0 \\ 0 & 0 & \cdot & 0 \\ 0 & 0 & \cdot & 0 \\ 0 & 0 & \cdot & 1 \end{bmatrix} \]
Estimation as Minimization (of the Sum of Squared residuals)

- In scalar math, we’d write the objective as the minimization of a sum-of-squared errors

\[
S(\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_{p-1}) = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2
\]

- Various styles using matrices.

1. Multiplication of residual vectors

\[
(y - \hat{y})^T (y - \hat{y})
\]

which is

\[
(y - X\hat{\beta})^T (y - X\hat{\beta})
\]

2. Norm notation
Estimation as Minimization (of the Sum of Squared residuals) ...

**Norm:** a measure of a vector’s magnitude

\[
\frac{1}{2} \| y - \hat{y} \|^2_2
\]

There are many different “norms” that might measure a vector’s “magnitude.” The L2 norm is used here, which uses the Pythagorean theorem

\[
\| x \|_2 = \sqrt{x_1^2 + x_2^2 + x_3^2 + \ldots + x_N^2}
\]
Estimation: Derivation of the "Normal Equations"

- For any guess about
  \[
  \beta = \begin{bmatrix}
  \beta_0 \\
  \beta_1 \\
  \vdots \\
  \beta_p
  \end{bmatrix}
  \]
  we can calculate \( X\beta \)

- So imagine putting in guesses of \( \beta \) that make the sum of squared errors as small as possible

- The sum of squared errors
  \[
  S(\beta) = (y - \hat{y})^T(y - \hat{y}) = (y - X\hat{\beta})(y - X\hat{\beta})
  \]
  \( (6) \)

The ordinary least squares solution is the value of \( \beta \) which minimizes that,

\[
\hat{\beta}^{OLS} = \argmin_{\beta} (y - \hat{y})(y - \hat{y})
\]
The first order conditions have one row for each coefficient being estimated. When we write out the FOC, one after another, we have the normal equations

\[
\frac{\partial S}{\partial \beta_0} = 0 \\
\frac{\partial S}{\partial \beta_1} = 0 \\
\frac{\partial S}{\partial \beta_2} = 0 \\
\frac{\partial S}{\partial \beta_3} = 0 \\
\vdots
\]

First Order Conditions with 2 predictors:
Matrix View of Multiple Regression

Estimation: Derivation of the "Normal Equations" ...

\[
\frac{\partial S}{\partial \beta_0} = -2 \sum (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_1 i - \hat{\beta}_2 x_2 i) = 0 \\
\frac{\partial S}{\partial \beta_1} = -2 \sum (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_1 i - \hat{\beta}_2 x_2 i) x_1 i = 0 \\
\frac{\partial S}{\partial \beta_2} = -2 \sum (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_1 i - \hat{\beta}_2 x_2 i) x_2 i = 0 \\
\]

The Normal Equations for a regression with an intercept and 2 predictors

\[
\sum y_i = N \hat{\beta}_0 + (\sum x_1 i) \hat{\beta}_1 + (\sum X_2 i) \hat{\beta}_2 \\
\sum y_i x_1 i = (\sum x_1 i) \hat{\beta}_0 + (\sum x_1^2 i) \hat{\beta}_1 + (\sum x_1 i x_2 i) \hat{\beta}_2 \\
\sum y_i x_2 i = (\sum x_2 i) \hat{\beta}_0 + (\sum x_1 i x_2 i) \hat{\beta}_1 + (\sum x_2^2 i) \hat{\beta}_2 \\
\]
Estimation: Derivation of the "Normal Equations" ...

- Re-group to see how we will view that as a matrix equation with 3 rows:

\[
\begin{bmatrix}
\sum y_i \\
\sum y_i x_1 \\
\sum y_i x_2 \\
\end{bmatrix}
= \begin{bmatrix}
N & \sum x_1 & \sum x_2 \\
\sum x_1 & \sum x_1^2 & \sum x_1 x_2 \\
\sum x_2 & \sum x_1 x_2 & \sum x_2^2 \\
\end{bmatrix}
\begin{bmatrix}
\hat{\beta}_0 \\
\hat{\beta}_1 \\
\hat{\beta}_2 \\
\end{bmatrix}
\]  

Which is

\[X^T y = (X^T X) \hat{\beta}\]  

(9)
Often described as a Case for Matrix Inversion

- If we could calculate \((X^TX)^{-1}\), we would multiply both sides of the previous:

\[
(X^TX)^{-1}X^Ty = (X^TX)^{-1}(X^TX) \hat{\beta}
\]

\[
(X^TX)^{-1}X^Ty = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \hat{\beta}
\]

\[
(X^TX)^{-1}X^Ty = \hat{\beta}
\]

- I was taught in the 1970s that computer programs calculate regression in that way.
- Today, that would be considered poor numerical linear algebra.
  - Even forming \(X^TX\) involves rounding error that gives us gray hair.
- Today, these calculations are usually done by decomposing \(X\) into numerically more-stable submatrices. See Wood, S (2006). *Generalized Additive Models.*
Partitioned Predictor Matrix

- Reminder. Treating all of the $X$’s as a big block, the estimate is the solution to the normal equation

$$
\begin{bmatrix}
\hat{\beta}_0 \\
\vdots \\
\hat{\beta}_p
\end{bmatrix} = X^T y
$$

- It is allowed, however, to separate the predictors into 2 groups, say $X_1$ and $X_2$, and think of the regression model as

$$y = X_1 \beta_1 + X_2 \beta_2 + \varepsilon$$

- The **normal equation** for the partitioned problem is

$$
\begin{bmatrix}
X_1^T X_1 & X_1^T X_2 \\
X_2^T X_1 & X_2^T X_2
\end{bmatrix}
\begin{bmatrix}
\beta_1 \\
\beta_2
\end{bmatrix} =
\begin{bmatrix}
X_1^T y \\
X_2^T y
\end{bmatrix}
$$
The coefficient estimate vector $\hat{\beta}$ from the original model is just the two separate pieces stacked together, $[\beta_1, \beta_2]$.

Note that if the 2 blocks of predictors are completely uncorrelated—Orthogonal, then $X_1^T X_2$ will be a big block of 0s. The two bits can be solved separately.

If the blocks are not orthogonal, then we could think of solving this as 2 simultaneous equations:

\[
X_1^T X_1 \beta_1 + X_1^T X_2 \beta_2 = X_1^T y \\
X_2^T X_1 \beta_1 + X_2^T X_2 \beta_2 = X_2^T y
\]

Put a bookmark on this and remember it when we come to Henderson’s Mixed Model Equations (MME)
Evaluate Variance

- Important to figure out how precise our estimated slopes might be
- Want to conduct hypothesis tests, one popular way is the ratio $\hat{\beta}_j / \text{s.e.}(\hat{\beta}_j)$
- The standard error is the square root of the estimated variance of $\hat{\beta}$, which is now our topic.
- The “true” variance/covariance matrix of the estimator $\hat{\beta}$ is found by solving

$$ \text{Var}(\hat{\beta}) = E[(\hat{\beta} - \beta)^T(\hat{\beta} - \beta) | X] $$

which boils down to a very simple:

$$ \text{Var}(\hat{\beta}) = \sigma^2_e (X^T X)^{-1} \quad (10) $$

- That’s a theoretical quantity, because $\sigma^2_e$ is an unknown (to be estimated).
Evaluate Variance ...

- Question: How did we end up with a simple expression like (10) to begin with?
- Answer: Aggressive application of several simplifying assumptions.
- There's a nice description of it in Green's Econometric Analysis, 6th ed (p. 150). We want

\[
Var(\hat{\beta}) = E[(\hat{\beta} - \beta)(\hat{\beta} - \beta)|X]
\]

- Insert \((X^T X)^{-1} X^T y\) in place of \(\hat{\beta}\). Because the \(X\) is treated as fixed, this whole thing simplifies to

\[
(X^T X)^{-1} X^T \text{Var}(\varepsilon) X (X^T X)^{-1}
\]

(11)
Evaluate Variance …

- The ordinary least squared setup implies

\[
\text{Var}(\varepsilon) = \sigma^2_\varepsilon I = \sigma^2_\varepsilon \begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 \\
0 & 0 & \ddots & 0 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\]

- Put that in the middle of (11), and look how the formula simplifies

\[
\text{Var}(\hat{\beta}) = (X^T X)^{-1} X^T \sigma^2_\varepsilon I X (X^T X)^{-1} \\
= \sigma^2_\varepsilon (X^T X)^{-1} X^T X (X^T X)^{-1} \\
= \sigma^2_\varepsilon (X^T X)^{-1} (X^T X)(X^T X)^{-1} \\
= \sigma^2_\varepsilon (X^T X)^{-1} \\
\] (12)
Evaluate Variance ...

- The residuals from the fitted model, $y - \hat{y}$, are the central element in this.
- The maximum likelihood estimate of the variance is simply the sum of squared residuals divided by $N$

$$\hat{\sigma}_e^2^{MLE} = \frac{1}{N} (y - \hat{y})^T (y - \hat{y})$$

- There are some obvious shortcomings in this.
  - Intuition: as you add in more predictors, $(y - \hat{y})$ always gets smaller. You always “improve” the model by adding nearly irrelevant predictors.
  - Difference between $R^2$ and adjusted $R^2$ flows from this
  - Theoretical problem: The MLE is “biased”. That means, it is, on average, incorrect.
Evaluate Variance ...

- The alternative estimator, is a “restricted maximum likelihood estimator”. We are used to this, it is a bias corrected version.

\[
\hat{\sigma}_e^2^{REML} = MSE = \frac{1}{N - p} (y - \hat{y})^T (y - \hat{y})
\]  

(13)

- The correction is changing the denominator from \( N \) to \( (N - p) \).

- This is important in the debate in multi-level modeling, since MLM is mostly about estimating variances.

- We use an estimated variance of \( \hat{\beta} \) (two hats! Estimated variance of Estimated coefficients!)

\[
\text{Var}(\hat{\beta}) = \hat{\sigma}_e^2 (X^T X)^{-1}
\]

(14)

- The square root of the diagonal elements of \( \text{Var}(\hat{\beta}) \) is commonly called the “standard error” of \( \hat{\beta} \). It is the denominator in t-tests.
OLS relies on the simplifying assumptions to derive the variance formula and to make it work with estimated error variance.

What if there are covariances among the observations?

$$\text{Var}(\varepsilon) = \begin{bmatrix}
\sigma_1^2 & \sigma_{12} & \sigma_{13} & \cdots & \sigma_{1N} \\
\sigma_{21} & \sigma_2^2 & & & \\
\sigma_{31} & \sigma_{32} & \sigma_3^2 & & \\
& & \ddots & \ddots & \\
\sigma_{N1} & \sigma_{N2} & & \sigma_N^2 \\
\end{bmatrix}$$

errors of one case co-vary with errors from another case.
About the Error Variance matrix ...

- Because that’s a variance matrix, one simplification is immediate. It is necessary that this is symmetric.

\[
\text{Var}(\varepsilon) = \begin{bmatrix}
\sigma_1^2 & \sigma_{21} & \sigma_{31} & \cdots & \sigma_{N1} \\
\sigma_{21} & \sigma_2^2 & \sigma_{32} & \cdots & \sigma_{N2} \\
\sigma_{31} & \sigma_{32} & \sigma_3^2 & \cdots & \sigma_{N3} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
\sigma_{N1} & \sigma_{N2} & \sigma_{N3} & \cdots & \sigma_N^2
\end{bmatrix}
\]

The matrix has \(N \times N\) elements, but we are only freely able to set the diagonal and one triangle. That means there are, at most, \(N(1 + N)/2\) parameters.

- In the end, we find that it is probably asking to much to expect precise estimates of all \(N(1 + N)/2\) parameters.

- The best we can do is develop some simple, interesting frameworks and estimate a few coefficients in them.
While I was thinking about that, I thought of this other Thing

Is your error term \( \{ \varepsilon_1, \varepsilon_2, \ldots, \varepsilon_N \} \) best thought of as

1. one variable with \( N \) random realizations, or
2. \( N \) random variables, each 1 draw from one of \( N \) random processes.

- \( N \) draws from the same process is the standard idea.
- When the assumption of “homoskedasticity” is violated, we definitely lean toward the 2nd interpretation.
- Sometimes people will talk about the thing we usually call the error term as \( N \) variates to emphasize this.
- There is a fine line between univariate and multivariate regression, then.
  - Univariate: a single column of outcomes modeled as 1 variable
While I was thinking about that, I thought of this other Thing ...

- Multivariate: several separate random processes under consideration within the same framework

\[ \varepsilon = \{ \varepsilon_1, \varepsilon_2, \ldots, \varepsilon_N \}^T \] can be a multivariate draw from an \( N \) – dimensional Multivariate Normal Distribution. This is the most familiar \( N \) – dimensional distribution. There are 2 parameters, expected values and variance, in \( MVN(\mu, \Sigma) \).

\[
\mu = \begin{bmatrix}
\mu_1 \\
\mu_2 \\
\mu_3 \\
\vdots \\
\mu_N
\end{bmatrix}
\quad \text{and} \quad
\Sigma = \begin{bmatrix}
\sigma_1^2 & \sigma_{21} & \sigma_{31} & \ldots & \sigma_{N1} \\
\sigma_{21} & \sigma_2^2 & & \sigma_{N2} \\
\sigma_{31} & \sigma_{23} & \sigma_3^2 & \sigma_{N3} \\
\vdots & \vdots & \vdots & \ddots \\
\sigma_{N1} & \sigma_{N2} & \sigma_{N3} & \ldots & \sigma_N^2
\end{bmatrix}
\]
While I was thinking about that, I thought of this other Thing ...

- The full error variance matrix is overwhelming. We can layer on different assumptions, keeping the variance model as simple as possible.
- In OLS, you assert a very simple

\[
Var(\varepsilon) = \begin{bmatrix}
\sigma^2 & 0 & \cdots & 0 \\
0 & \sigma^2 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & \sigma^2 
\end{bmatrix}
\]
While I was thinking about that, I thought of this other Thing ...

- Heteroskedasticity: $\varepsilon$’s are drawn from different distributions, but they are all uncorrelated with each other

$$
\text{Var}(\varepsilon) = \begin{bmatrix}
\sigma_1^2 & 0 & \cdots & 0 \\
0 & \sigma_2^2 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & & & \sigma_N^2
\end{bmatrix}
$$

That’s known as “heteroskedasticity”, the errors still have EV of 0 but the variances differ.
It is a little difficult to visualize GLS estimation, but it is easier to see what’s going on in its simpler cousin, WLS.

In a weighted least squares problem, we face a simpler challenge. Covariances = 0, but variances are heterogeneous

\[
\text{Var}(\varepsilon) = V = \begin{bmatrix}
\sigma_1^2 & 0 & 0 & 0 \\
0 & \sigma_2^2 & 0 & 0 \\
0 & 0 & \ddots & 0 \\
0 & 0 & 0 & \sigma_N^2
\end{bmatrix}
\]

**Notation Alert**

Sometimes this matrix is referred to as $\Sigma$ or $\Omega$ or some other Greek letter. I don’t mind writing out $\text{Var}(\varepsilon)$ when we have room to do it.
Weighted Least Squares …

- \( \hat{\beta}_{WLS} \) is a minimizer of the weighted sum of squared errors

\[
S(\beta) = (y - X\beta)^T V^{-1} (y - X\beta)
\]

which simplifies to a sum of \( N \) weighted squared terms:

\[
S(\beta) = \sum_{i}^{N} \left( \frac{1}{\sigma_i^2} \right) (y_i - X_i\beta)^2
\]

- Some authors will keep the weights inside the square. Let \( w_i = 1/\sigma_i \).

\[
S(\beta) = \sum_{i}^{N} (w_i(y_i - X_i\beta))^2 = \sum_{i}^{N} (w_i y_i - w_i X_i\beta)^2
\]

Reconceptualize the exercise. We multiply the data columns by a weighting factor, creating \( y_i^* = w_i y \) and \( X_i^* = w_i X_i \) which are fed to the OLS estimator.
Weighted Least Squares ...

- The R regression program `lm` includes a weight argument, to which we provide a column $1/\sigma_i^2$.
- The term “Feasible Weighted Least Squares” arises because we usually have to calculate estimates for the elements of $V$ and then insert them into the formula.
- Sometimes you’ll get confused in this literature because some authors prefer to factor out a common element $\sigma^2_\epsilon$. We could say

$$V = \sigma^2_\epsilon \begin{bmatrix}
\omega_1 & \omega_2 \\
\omega_1 & \omega_N \\
\omega_2 & \omega_N 
\end{bmatrix}, \text{ where } \sum \omega_i = N, \text{ or } V = \sigma^2_\epsilon \Omega$$

- It is necessary that the trace of $\Omega$ must be $N$ in order for these calculations to remain consistent with the OLS result for homogeneous variances.
Weighted Least Squares ...

- The benefit of this notation is not apparent until you start to read the Bates and DebRoy papers on maximum likelihood estimation of random effects models.

- Suppose we use OLS and ignore the problem of heterogeneous variance. The variance of the OLS estimator is:

\[
\text{Var}(\hat{\beta}_{OLS}) = [X^T X]^{-1} [X^T V X] [X^T X]^{-1}
\]  \hspace{1cm} (15)

- This all supposes the weights are “known”, but usually they have to be guessed or estimated. If we happen to guess the var/covar matrix is \( V^* \), and use that in place of the true matrix \( V \), then the variance of that estimator ends up depending on both the wrongly supposed variance matrix and the true variance matrix (See Greene, Econometric Analysis, 6ed, p. 168).

\[
\text{Var}(\hat{\beta}) = [X^T V^*^{-1} X]^{-1} X^T V^*^{-1} V V^*^{-1} X [X^T V^*^{-1} X]^{-1}
\]  \hspace{1cm} (16)

- White’s “robust” estimator of the variance is based on this setup
A More Elaborate Error Variance: Time Series AR(1) errors

- Time series analysis was the first area in stats to work on the implications of correlations across rows in a data set.
- The simple auto-correlated error model assumes that the error today is equal to a freshly drawn error for each row plus some left over error from yesterday
  \[ \varepsilon_i = v_i + \rho \varepsilon_{i-1} \]
- That is known as an $AR(1)$ model, it leads to a variance matrix that depends on only 2 separate parameters, $\sigma_v^2$ and $\rho$.

\[
\text{Var}(\varepsilon) = \frac{\sigma_v^2}{(1 - \rho^2)} \\
\begin{bmatrix}
1 & \rho & \rho^2 & \ldots & \rho^{N-1} \\
\rho & 1 & \rho & \ldots & \rho^{N-2} \\
\rho^2 & \rho & 1 & \ldots & \rho^{N-3} \\
\vdots & \vdots & \ddots & \ddots & \rho \\
\rho^{N-1} & \rho^{N-2} & \rho^{N-3} & \ldots & 1
\end{bmatrix}
\]

- How to estimate that? GLS!
How does that get Estimated? Generalized Least Squares


- GLS. The objective changes from

$$(y - \hat{y})^T (y - \hat{y})$$

to this

$$(y - \hat{y})^T \text{Var}(e)^{-1} (y - \hat{y})$$

- The error variance plays the role of a weight, so high variance cases are given less weight.
How does that get Estimated? Generalized Least Squares

- Let capital sigma, $\Sigma = Var(\varepsilon)$.

- Aitken’s GLS estimator is surprisingly simple, the error matrix is placed into the middle of the usual answer in two places.

$$\hat{\beta}^{GLS} = (X^T\Sigma^{-1}X)^{-1}(X^T\Sigma^{-1}y)$$

- The variance of those estimates is

$$\text{Var}(\hat{\beta}^{GLS}) = \left(\frac{(y - X\hat{\beta})^T\Sigma^{-1}(y - X\hat{\beta})}{N}\right) (X^T\Sigma^{-1}X)^{-1}$$

- That first thing on the right is a variance adjusted error sum of squares, you’ll see it written out more simply as

$$\text{Var}(\hat{\beta}^{GLS}) = \hat{\sigma}_\varepsilon^2 (X^T\Sigma^{-1}X)^{-1}$$
Two challenges

1. Computation
   - Success requires the creation and inversion of an $N \times N$ variance matrix.
   - On a modern “large” data set, that can be computationally intensive. We would look for decompositions and abstractions to move this along.

2. Usually there are parameters in $\Sigma$ that must be estimated. This leads to an iterative fitting process. The usual description is as follows
   - Estimate $\hat{\beta}$
   - Estimate $\text{Var}(e)$ from the residuals
   - Re-estimate $\hat{\beta}$, etc.
Matrix View of Multiple Regression

Cholesky Root application

- Cholesky decomposition (the “square root” of a matrix): $V = R^T R$, where $R$ is an upper triangular matrix. Replace $V$ in the sum of squares with $R^T R$

- $S(\beta) = (y - X \beta)^T (R^T R)^{-1} (y - X \beta) = (y - X \beta)^T R^{-1} R^{-T} (y - X \beta)$, where $R^{-T}$ means $(R^T)^{-1}$.

- Which rearranges as $S(\beta) = (R^{-T} y - R^{-T} X \beta)^T (R^{-T} y - R^{-T} X \beta)$.

- Basically, this is creating new weighted observations $y^* = R^{-T} y$ and $X^* = R^{-T} X$

  - and then we run the regression to minimize $S(\beta) = (y^* - X^* \beta)^T (y^* - X^* \beta)$

- The key thing about this, as far at MLM is concerned, is that once $V$ is estimated, we can insert that into the calculation of $\hat{\beta}_{GLS}$

- If we estimate $V$, then $\hat{\beta}_{GLS}$ is a known thing, so we do not need to think of maximum likelihood estimates for $(V, \beta)$. Rather, we just think of maximizing $(V, \hat{\beta}(V))$. $V$ is the only variable we need to concentrate on. This is a “profiled Likelihood” function.