Gov 2000: 10. Multiple Regression in Matrix Form

Matthew Blackwell

Fall 2016

Where are we? Where are we going?

- Last few weeks: regression estimation and inference with one and two independent variables, varying effects
- This week: the general regression model with arbitrary covariates
- Next week: what happens when assumptions are wrong

Nunn & Wantchekon

- Are there long-term, persistent effects of slave trade on Africans today?
- Basic idea: compare levels of interpersonal trust (Y_i) across different levels of historical slave exports for a respondent's ethnic group
- Problem: ethnic groups and respondents might differ in their interpersonal trust in ways that correlate with the severity of slave exports
- One solution: try to control for relevant differences between groups via multiple regression

Nunn & Wantchekon

VOL. 101 NO. 7 NUNN AND WANTCHEKON: THE ORIGINS OF MISTRUST IN AFRICA 3231

III. Estimating Equations and Empirical Results

A. *OLS Estimates*

We begin by estimating the relationship between the number of slaves that were taken from an individual's ethnic group and the individual's current level of trust. Our baseline estimating equation is:

$$(1) \quad trust_{i,e,d,c} = \alpha_c + \beta slave\ exports_e + \mathbf{X}'_{i,e,d,c}\boldsymbol{\Gamma} + \mathbf{X}'_{d,c}\boldsymbol{\Omega} + \mathbf{X}'_e\boldsymbol{\Phi} + \varepsilon_{i,e,d,c},$$

- Whaaaaa? Bold letter, quotation marks, what is this?
- Today's goal is to decipher this type of writing

Multiple Regression in R

```
nunn <- foreign::read.dta("../data/Nunn_Wantchekon_AER_2011.dta")
mod <- lm(trust_neighbors ~ exports + age + male + urban_dum
          + malaria_ecology, data = nunn)
summary(mod)
```

```
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     1.5030370  0.0218325   68.84   <2e-16 ***
## exports        -0.0010208  0.0000409  -24.94   <2e-16 ***
## age             0.0050447  0.0004724   10.68   <2e-16 ***
## male            0.0278369  0.0138163    2.01    0.044 *
## urban_dum      -0.2738719  0.0143549  -19.08   <2e-16 ***
## malaria_ecology 0.0194106  0.0008712   22.28   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.978 on 20319 degrees of freedom
##   (1497 observations deleted due to missingness)
## Multiple R-squared:  0.0604, Adjusted R-squared:  0.0602
## F-statistic:  261 on 5 and 20319 DF,  p-value: <2e-16
```

Why matrices and vectors?

Why matrices and vectors?

- Here's one way to write the full multiple regression model:

$$y_i = \beta_0 + x_{i1}\beta_1 + x_{i2}\beta_2 + \cdots + x_{ik}\beta_k + u_i$$

- Notation is going to get needlessly messy as we add variables.
- Matrices are clean, but they are like a foreign language.
- You need to build intuitions over a long period of time.

Quick note about interpretation

$$y_i = \beta_0 + x_{i1}\beta_1 + x_{i2}\beta_2 + \cdots + x_{ik}\beta_k + u_i$$

- In this model, β_1 is the effect of a one-unit change in x_{i1} conditional on all other x_{ij}.
- Jargon "partial effect," "ceteris paribus," "all else equal," "conditional on the covariates," etc
- Notation change: lower-case letters here are random variables.

1/ Matrix algebra review

Vectors

- A vector is just list of numbers (or random variables).
- A $1 \times k$ row vector has these numbers arranged in a row:

$$\mathbf{b} = \begin{bmatrix} b_1 & b_2 & b_3 & \cdots & b_k \end{bmatrix}$$

- A $k \times 1$ column vector arranges the numbers in a column:

$$\mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_k \end{bmatrix}$$

- Convention we'll assume that a vector is column vector and vectors will be written with lowercase bold lettering ($\mathbf{b}$)

Vector examples

- Vector of all covariates for a particular unit i:

$$\mathbf{x}_i = \begin{bmatrix} 1 \\ x_{i1} \\ x_{i2} \\ \vdots \\ x_{ik} \end{bmatrix}$$

- For the Nunn-Wantchekon data, we might have:

$$\mathbf{x}_i = \begin{bmatrix} 1 \\ \text{exports}_i \\ \text{age}_i \\ \text{male}_i \end{bmatrix}$$

Matrices

- A matrix is just a rectangular array of numbers.
- We say that a matrix is $n \times k$ ("n by k") if it has n rows and k columns.
- Uppercase bold denotes a matrix:

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1k} \\ a_{21} & a_{22} & \cdots & a_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nk} \end{bmatrix}$$

- Generic entry: a_{ij} where this is the entry in row i and column j

Examples of matrices

- One example of a matrix that we'll use a lot is the design matrix, which has a column of ones, and then each of the subsequent columns is each independent variable in the regression.

$$\mathbf{X} = \begin{bmatrix} 1 & \text{exports}_1 & \text{age}_1 & \text{male}_1 \\ 1 & \text{exports}_2 & \text{age}_2 & \text{male}_2 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \text{exports}_n & \text{age}_n & \text{male}_n \end{bmatrix}$$

Design matrix in R

```
head(model.matrix(mod), 8)
```

```
##   (Intercept) exports age male urban_dum malaria_ecology
## 1           1     855  40    0         0           28.15
## 2           1     855  25    1         0           28.15
## 3           1     855  38    1         1           28.15
## 4           1     855  37    0         1           28.15
## 5           1     855  31    1         0           28.15
## 6           1     855  45    0         0           28.15
## 7           1     855  20    1         0           28.15
## 8           1     855  31    0         0           28.15
```

```
dim(model.matrix(mod))
```

```
## [1] 20325     6
```

2/ Matrix Operations

Transpose

- The transpose of a matrix $\mathbf{A}$ is the matrix created by switching the rows and columns of the data and is denoted $\mathbf{A}'$.
- kth column of $\mathbf{A}$ becomes the kth row of $\mathbf{A}'$:

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{bmatrix} \quad \mathbf{A}' = \begin{bmatrix} a_{11} & a_{21} & a_{31} \\ a_{12} & a_{22} & a_{32} \end{bmatrix}$$

- If $\mathbf{A}$ is $n \times k$, then $\mathbf{A}'$ will be $k \times n$.
- Also written $\mathbf{A}^{\mathbf{T}}$

Transposing vectors

- Transposing will turn a $k \times 1$ column vector into a $1 \times k$ row vector and vice versa:

$$\mathbf{x}_i = \begin{bmatrix} 1 \\ x_{i1} \\ x_{i2} \\ \vdots \\ x_{ik} \end{bmatrix} \qquad \mathbf{x}_i' = \begin{bmatrix} 1 & x_{i1} & x_{i2} & \cdots & x_{ik} \end{bmatrix}$$

Transposing in R

```r
a <- matrix(1:6, ncol = 3, nrow = 2)
a
```

```
##      [,1] [,2] [,3]
## [1,]    1    3    5
## [2,]    2    4    6
```

```r
t(a)
```

```
##      [,1] [,2]
## [1,]    1    2
## [2,]    3    4
## [3,]    5    6
```

Write matrices as vectors

- A matrix is just a collection of vectors (row or column)
- As a row vector:

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} = \begin{bmatrix} \mathbf{a}_1' \\ \mathbf{a}_2' \end{bmatrix}$$

  with row vectors
  $\mathbf{a}_1' = \begin{bmatrix} a_{11} & a_{12} & a_{13} \end{bmatrix}$ $\mathbf{a}_2' = \begin{bmatrix} a_{21} & a_{22} & a_{23} \end{bmatrix}$
- Or we can define it in terms of column vectors:

$$\mathbf{B} = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \end{bmatrix} = \begin{bmatrix} \mathbf{b_1} & \mathbf{b_2} \end{bmatrix}$$

  where $\mathbf{b_1}$ and $\mathbf{b_2}$ represent the columns of $\mathbf{B}$.
- j subscripts columns of a matrix: $\mathbf{x}_j$
- i and t will be used for rows $\mathbf{x}_i'$.

Design matrix

- Design matrix as a series of row vectors:

$$\mathbf{X} = \begin{bmatrix} 1 & \text{exports}_1 & \text{age}_1 & \text{male}_1 \\ 1 & \text{exports}_2 & \text{age}_2 & \text{male}_2 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \text{exports}_n & \text{age}_n & \text{male}_n \end{bmatrix} = \begin{bmatrix} \mathbf{x}'_1 \\ \mathbf{x}'_2 \\ \vdots \\ \mathbf{x}'_n \end{bmatrix}$$

- Design matrix as a series of column vectors:

$$\mathbf{X} = \begin{bmatrix} \mathbf{1} & \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_k \end{bmatrix}$$

Addition and subtraction

- How do we add or subtract matrices and vectors?
- First, the matrices/vectors need to be comformable, meaning that the dimensions have to be the same.
- Let $\mathbf{A}$ and $\mathbf{B}$ both be 2×2 matrices. Then, let $\mathbf{C} = \mathbf{A} + \mathbf{B}$, where we add each cell together:

$$\begin{aligned}
\mathbf{A} + \mathbf{B} &= \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} \\
&= \begin{bmatrix} a_{11} + b_{11} & a_{12} + b_{12} \\ a_{21} + b_{21} & a_{22} + b_{22} \end{bmatrix} \\
&= \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} \\
&= \mathbf{C}
\end{aligned}$$

Scalar multiplication

- A scalar is just a single number: you can think of it sort of like a 1 by 1 matrix.
- When we multiply a scalar by a matrix, we just multiply each element/cell by that scalar:

$$b\mathbf{A} = b \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = \begin{bmatrix} b \times a_{11} & b \times a_{12} \\ b \times a_{21} & b \times a_{22} \end{bmatrix}$$

3/ Linear model in matrix form

The linear model with new notation

- Remember that we wrote the linear model as the following for all $i \in \{1, \ldots, n\}$:

$$y_i = \beta_0 + x_i\beta_1 + z_i\beta_2 + u_i$$

- Imagine we had an n of 4. We could write out each formula:

$$\begin{aligned}
y_1 &= \beta_0 + x_1\beta_1 + z_1\beta_2 + u_1 \quad \text{(unit 1)} \\
y_2 &= \beta_0 + x_2\beta_1 + z_2\beta_2 + u_2 \quad \text{(unit 2)} \\
y_3 &= \beta_0 + x_3\beta_1 + z_3\beta_2 + u_3 \quad \text{(unit 3)} \\
y_4 &= \beta_0 + x_4\beta_1 + z_4\beta_2 + u_4 \quad \text{(unit 4)}
\end{aligned}$$

The linear model with new notation

$$
\begin{aligned}
y_1 &= \beta_0 + x_1\beta_1 + z_1\beta_2 + u_1 \quad \text{(unit 1)} \\
y_2 &= \beta_0 + x_2\beta_1 + z_2\beta_2 + u_2 \quad \text{(unit 2)} \\
y_3 &= \beta_0 + x_3\beta_1 + z_3\beta_2 + u_3 \quad \text{(unit 3)} \\
y_4 &= \beta_0 + x_4\beta_1 + z_4\beta_2 + u_4 \quad \text{(unit 4)}
\end{aligned}
$$

- We can write this as:

$$
\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \beta_0 + \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \beta_1 + \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} \beta_2 + \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}
$$

- Outcome is a linear combination of the the $\mathbf{x}$, $\mathbf{z}$, and $\mathbf{u}$ vectors

Grouping things into matrices

- Can we write this in a more compact form? Yes! Let $\mathbf{X}$ and $\boldsymbol{\beta}$ be the following:

$$\underset{(4\times3)}{\mathbf{X}} = \begin{bmatrix} 1 & x_1 & z_1 \\ 1 & x_2 & z_2 \\ 1 & x_3 & z_3 \\ 1 & x_4 & z_4 \end{bmatrix} \quad \underset{(3\times1)}{\boldsymbol{\beta}} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix}$$

Matrix multiplication by a vector

- We can write this more compactly as a matrix (post-)multiplied by a vector:

$$\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \beta_0 + \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \beta_1 + \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} \beta_2 = \mathbf{X}\boldsymbol{\beta}$$

- Multiplication of a matrix by a vector is just the linear combination of the columns of the matrix with the vector elements as weights/coefficients.
- And the left-hand side here only uses scalars times vectors, which is easy!

General matrix by vector multiplication

- $\mathbf{A}$ is a $n \times k$ matrix
- $\mathbf{b}$ is a $k \times 1$ column vector
- Columns of $\mathbf{A}$ have to match rows of $\mathbf{b}$
- Let $\mathbf{a}_j$ be the jth column of A. Then we can write:

$$\underset{(n\times 1)}{\mathbf{c}} = \mathbf{Ab} = b_1\mathbf{a}_1 + b_2\mathbf{a}_2 + \cdots + b_k\mathbf{a}_k$$

- $\mathbf{c}$ is linear combination of the columns of $\mathbf{A}$

Back to regression

- $\mathbf{X}$ is the $n \times (k+1)$ design matrix of independent variables
- $\boldsymbol{\beta}$ be the $(k+1) \times 1$ column vector of coefficients.
- $\mathbf{X}\boldsymbol{\beta}$ will be $n \times 1$:

$$\mathbf{X}\boldsymbol{\beta} = \beta_0 + \beta_1\mathbf{x}_1 + \beta_2\mathbf{x}_2 + \cdots + \beta_k\mathbf{x}_k$$

- Thus, we can compactly write the linear model as the following:

$$\underset{(n\times 1)}{\mathbf{y}} = \underset{(n\times 1)}{\mathbf{X}\boldsymbol{\beta}} + \underset{(n\times 1)}{\mathbf{u}}$$

Inner product

- The inner (or dot) product of a two column vectors $\mathbf{a}$ and $\mathbf{b}$ (of equal dimension, $k \times 1$):

$$\langle \mathbf{a}, \mathbf{b} \rangle = \mathbf{a}'\mathbf{b} = a_1 b_1 + a_2 b_2 + \cdots + a_k b_k$$

- If $\mathbf{a}'\mathbf{b} = 0$ we say that the two vectors are orthogonal.
- With $\mathbf{c} = \mathbf{A}\mathbf{b}$, we can write the entries of $\mathbf{c}$ as inner products:

$$c_i = \mathbf{a}_i'\mathbf{b}$$

- If $\mathbf{x}_i'$ is the ith row of $\mathbf{X}$, then we write the linear model as:

$$\begin{aligned} y_i &= \mathbf{x}_i'\boldsymbol{\beta} + u_i \\ &= \beta_0 + x_{i1}\beta_1 + x_{i2}\beta_2 + \cdots + x_{ik}\beta_k + u_i \end{aligned}$$

4/ OLS in matrix form

Matrix multiplication

- What if, instead of a column vector b, we have a matrix $\mathbf{B}$ with dimensions $k \times m$.
- How do we do multiplication like so $\mathbf{C} = \mathbf{AB}$?
- Each column of the new matrix is just matrix by vector multiplication:

$$\mathbf{C} = [\mathbf{c}_1 \quad \mathbf{c}_2 \quad \cdots \quad \mathbf{c}_m] \qquad \mathbf{c}_j = \mathbf{A}\mathbf{b}_j$$

- Thus, each column of $\mathbf{C}$ is a linear combination of the columns of $\mathbf{A}$.

Properties of matrix multiplication

- Matrix multiplication is not commutative: $\mathbf{AB} \neq \mathbf{BA}$
- It is associative and distributive:

$$\mathbf{A}(\mathbf{BC}) = (\mathbf{AB})\mathbf{C}$$
$$\mathbf{A}(\mathbf{B} + \mathbf{C}) = \mathbf{AB} + \mathbf{AC}$$

- The transpose: $(\mathbf{AB})' = \mathbf{B}'\mathbf{A}'$

Square matrices and the diagonal

- A square matrix has equal numbers of rows and columns.
- The identity matrix, $\mathbf{I}_k$ is a $k \times k$ square matrix, with 1s along the diagonal and 0s everywhere else.

$$\mathbf{I}_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- The $k \times k$ identity matrix multiplied by any $m \times k$ matrix returns the matrix:

$$\mathbf{A}\mathbf{I}_k = \mathbf{A}$$

Identity matrix

- To get the diagonal of a matrix in R, use the diag() function:

```
b <- matrix(1:4, nrow = 2, ncol = 2)
b
```

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

```
diag(b)
```

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

- diag() also creates identity matrices in R:

```
diag(3)
```

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

Multiple linear regression in matrix form

- Let $\widehat{\boldsymbol{\beta}}$ be the matrix of estimated regression coefficients and $\hat{\mathbf{y}}$ be the vector of fitted values:

$$\widehat{\boldsymbol{\beta}} = \begin{bmatrix} \widehat{\beta}_0 \\ \widehat{\beta}_1 \\ \vdots \\ \widehat{\beta}_k \end{bmatrix} \qquad \hat{\mathbf{y}} = \mathbf{X}\widehat{\boldsymbol{\beta}}$$

- It might be helpful to see this again more written out:

$$\hat{\mathbf{y}} = \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_n \end{bmatrix} = \mathbf{X}\widehat{\boldsymbol{\beta}} = \begin{bmatrix} 1\widehat{\beta}_0 + x_{11}\widehat{\beta}_1 + x_{12}\widehat{\beta}_2 + \cdots + x_{1k}\widehat{\beta}_k \\ 1\widehat{\beta}_0 + x_{21}\widehat{\beta}_1 + x_{22}\widehat{\beta}_2 + \cdots + x_{2k}\widehat{\beta}_k \\ \vdots \\ 1\widehat{\beta}_0 + x_{n1}\widehat{\beta}_1 + x_{n2}\widehat{\beta}_2 + \cdots + x_{nk}\widehat{\beta}_k \end{bmatrix}$$

Residuals

- We can easily write the residuals in matrix form:

$$\hat{\mathbf{u}} = \mathbf{y} - \mathbf{X}\widehat{\boldsymbol{\beta}}$$

- The norm or length of a vector generalizes Euclidean distance and is just the square root of the squared entries,

$$\|\mathbf{a}\| = \sqrt{a_1^2 + a_2^2 + \cdots + a_k^2}$$

- We can write the norm in terms of inner product: $\|\mathbf{a}\|^2 = \mathbf{a}'\mathbf{a}$
- Thus we can compactly write the sum of the squared residuals as:

$$\begin{aligned} \|\hat{\mathbf{u}}\|^2 &= \hat{\mathbf{u}}'\hat{\mathbf{u}} \\ &= \sum_{i=1}^{n} \hat{u}_i^2 \end{aligned}$$

OLS estimator in matrix form

- OLS still minimizes sum of the squared residuals

$$\arg\min_{\mathbf{b}\in\mathbb{R}^{k+1}} \|\hat{\mathbf{u}}\|^2 = \arg\min_{\mathbf{b}\in\mathbb{R}^{k+1}} \|\mathbf{y} - \mathbf{X}\mathbf{b}\|^2$$

- Take (matrix) derivatives, set equal to 0
- Resulting first order conditions:

$$\mathbf{X}'(\mathbf{y} - \mathbf{X}\widehat{\boldsymbol{\beta}}) = 0$$

- Rearranging:

$$\mathbf{X}'\mathbf{X}\widehat{\boldsymbol{\beta}} = \mathbf{X}'\mathbf{y}$$

- In order to isolate $\widehat{\boldsymbol{\beta}}$, we need to move the $\mathbf{X}'\mathbf{X}$ term to the other side of the equals sign.
- We've learned about matrix multiplication, but what about matrix "division"?

Scalar inverses

- What is division in its simplest form? $\frac{1}{a}$ is the value such that $a\frac{1}{a} = 1$:
- For some algebraic expression: $au = b$, let's solve for u:

$$\frac{1}{a}au = \frac{1}{a}b$$
$$u = \frac{b}{a}$$

- Need a matrix version of this: $\frac{1}{a}$.

Matrix inverses

- **Definition** If it exists, the inverse of square matrix $\mathbf{A}$, denoted $\mathbf{A}^{-1}$, is the matrix such that $\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$.
- We can use the inverse to solve (systems of) equations:

$$\begin{aligned} \mathbf{A}\mathbf{u} &= \mathbf{b} \\ \mathbf{A}^{-1}\mathbf{A}\mathbf{u} &= \mathbf{A}^{-1}\mathbf{b} \\ \mathbf{I}\mathbf{u} &= \mathbf{A}^{-1}\mathbf{b} \\ \mathbf{u} &= \mathbf{A}^{-1}\mathbf{b} \end{aligned}$$

- If the inverse exists, we say that $\mathbf{A}$ is invertible or nonsingular.

Back to OLS

- Let's assume, for now, that the inverse of $\mathbf{X}'\mathbf{X}$ exists (we'll come back to this)
- Then we can write the OLS estimator as the following:

$$\widehat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

- Memorize this: "ex prime ex inverse ex prime y" sear it into your soul.

Understanding check

- Suppose $\mathbf{y}$ is $n \times 1$ and $\mathbf{X}$ is $n \times (k+1)$.
- What are the dimensions of $\mathbf{X}'\mathbf{X}$?
- True/False: $\mathbf{X}'\mathbf{X}$ is symmetric.
  - Note: A square matrix is symmetric if $\mathbf{A} = \mathbf{A}'$.
- What are the dimensions of $(\mathbf{X}'\mathbf{X})^{-1}$?
- What are the dimensions of $\mathbf{X}'\mathbf{y}$?
- What are the dimensions of $\widehat{\boldsymbol{\beta}}$?

Implications of OLS

- We can generalize some mechanical results about OLS.
- The independent variables are orthogonal to the residuals:

$$\mathbf{X}'\hat{\mathbf{u}} = \mathbf{X}'(\mathbf{y} - \mathbf{X}\widehat{\boldsymbol{\beta}}) = 0$$

- The fitted values are orthogonal to the residuals:

$$\hat{\mathbf{y}}'\hat{\mathbf{u}} = (\mathbf{X}\widehat{\boldsymbol{\beta}})'\hat{\mathbf{u}} = \widehat{\boldsymbol{\beta}}'\mathbf{X}'\hat{\mathbf{u}} = 0$$

OLS by hand in R

$$\widehat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

- First we need to get the design matrix and the response:

```
X <- model.matrix(trust_neighbors ~ exports + age + male
                  + urban_dum + malaria_ecology, data = nunn)
dim(X)
```

```
## [1] 20325     6
```

```
## model.frame always puts the response in the first column
y <- model.frame(trust_neighbors ~ exports + age + male
                 + urban_dum + malaria_ecology, data = nunn)[,1]
length(y)
```

```
## [1] 20325
```

OLS by hand in R

$$\widehat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

- Use the solve() for inverses and %*% for matrix multiplication:

```
solve(t(X) %*% X)   %*%   t(X) %*% y
```

```
##      (Intercept)   exports      age    male urban_dum
## [1,]       1.503 -0.001021 0.005045 0.02784   -0.2739
##      malaria_ecology
## [1,]         0.01941
```

```
coef(mod)
```

```
##     (Intercept)         exports             age            male
##        1.503037       -0.001021        0.005045        0.027837
##       urban_dum malaria_ecology
##       -0.273872        0.019411
```

Intuition for the OLS in matrix form

$$\widehat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

- What's the intuition here?
- "Numerator" $\mathbf{X}'\mathbf{y}$: is roughly composed of the covariances between the columns of $\mathbf{X}$ and $\mathbf{y}$
- "Denominator" $\mathbf{X}'\mathbf{X}$ is roughly composed of the sample variances and covariances of variables within $\mathbf{X}$
- Thus, we have something like:

$$\widehat{\boldsymbol{\beta}} \approx (\text{variance of } \mathbf{X})^{-1}(\text{covariance of } \mathbf{X} \text{ \& } \mathbf{y})$$

- This is a rough sketch and isn't strictly true, but it can provide intuition.

5/ OLS inference in matrix form

Random vectors

- A random vector is a vector of random variables:

$$\mathbf{x}_i = \begin{bmatrix} x_{i1} \\ x_{i2} \end{bmatrix}$$

- Here, $\mathbf{x}_i$ is a random vector and x_{i1} and x_{i2} are random variables.
- When we talk about the distribution of $\mathbf{x}_i$, we are talking about the joint distribution of x_{i1} and x_{i2}.

Distribution of random vectors

- Expectation of random vectors:

$$\mathbb{E}[\mathbf{x}_i] = \begin{bmatrix} \mathbb{E}[x_{i1}] \\ \mathbb{E}[x_{i2}] \end{bmatrix}$$

- Variance of random vectors:

$$\mathbb{V}[\mathbf{x}_i] = \begin{bmatrix} \mathbb{V}[x_{i1}] & \text{Cov}[x_{i1}, x_{i2}] \\ \text{Cov}[x_{i1}, x_{i2}] & \mathbb{V}[x_{i2}] \end{bmatrix}$$

- Properties of this variance-covariance matrix:
  - if $\mathbf{a}$ is constant, then $\mathbb{V}[\mathbf{a}'\mathbf{x}_i] = \mathbf{a}'\mathbb{V}[\mathbf{x}_i]\mathbf{a}$.
  - if matrix $\mathbf{A}$ and vector $\mathbf{b}$ are constant, then $\mathbb{V}[\mathbf{A}\mathbf{x}_i + \mathbf{b}] = \mathbf{A}\mathbb{V}[\mathbf{x}_i]\mathbf{A}'$

Most general OLS assumptions

1. Linearity: $y_i = \mathbf{x}_i'\boldsymbol{\beta} + u_i$
2. Random/iid sample: $(y_i, \mathbf{x}_i')$ are a iid sample from the population.
3. No perfect collinearity: $\mathbf{X}$ is an $n \times (k+1)$ matrix with rank $k+1$
4. Zero conditional mean: $\mathbb{E}[u_i|\mathbf{x}_i] = 0$
5. Homoskedasticity: $\mathbb{V}[u_i|\mathbf{x}_i] = \sigma_u^2$
6. Normality: $u_i|\mathbf{x}_i \sim N(0, \sigma_u^2)$

Matrix rank

- **Definition** The rank of a matrix is the maximum number of linearly independent columns.
- **Definition** The columns of a matrix $\mathbf{X}$ are linearly independent if $\mathbf{X}\mathbf{b} = 0$ if and only if $\mathbf{b} = 0$:

$$b_1\mathbf{x}_1 + b_2\mathbf{x}_2 + \cdots + b_k\mathbf{x}_k = 0$$

- Example violation: one column is a linear function of the others.
  - 3 covariates with $\mathbf{x}_1 = \mathbf{x}_2 + \mathbf{x}_3$

$$\begin{aligned} 0 &= b_1\mathbf{x}_1 + b_2\mathbf{x}_2 + b_3\mathbf{x}_3 \\ &= b_1(\mathbf{x}_2 + \mathbf{x}_3) + b_2\mathbf{x}_2 + b_3\mathbf{x}_3 \\ &= (b_1 + b_2)\mathbf{x}_2 + (b_1 + b_3)\mathbf{x}_3 \end{aligned}$$

- ...equals 0 when $b_1 = -b_2 = -b_3 \rightsquigarrow$ not linearly independent!

Rank and matrix inversion

- If $\mathbf{X}$ is $n \times (k+1)$ has rank $k+1$, then all of its columns are linearly independent
  - Generalization of no perfect collinearity to arbitrary k.
- $\mathbf{X}$ has rank $k+1 \rightsquigarrow (\mathbf{X}'\mathbf{X})$ has rank $k+1$
- If a square $(k+1) \times (k+1)$ matrix has rank $k+1$, then it is invertible.
- $\mathbf{X}$ has rank $k+1 \rightsquigarrow (\mathbf{X}'\mathbf{X})^{-1}$ exists and is unique.

Zero conditional mean error

- Combining zero mean conditional error and iid we have:

$$\mathbb{E}[u_i|\mathbf{X}] = \mathbb{E}[u_i|\mathbf{x}_i] = 0$$

- Stacking these into the vector of errors:

$$\mathbb{E}[\mathbf{u}|\mathbf{X}] = \begin{bmatrix} \mathbb{E}[u_1|\mathbf{X}] \\ \mathbb{E}[u_2|\mathbf{X}] \\ \vdots \\ \mathbb{E}[u_n|\mathbf{X}] \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Expectation of OLS

- Useful to write OLS as:

$$\begin{aligned}\widehat{\boldsymbol{\beta}} &= (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y} \\ &= (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'(\mathbf{X}\boldsymbol{\beta} + \mathbf{u}) \\ &= (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{X}\boldsymbol{\beta} + (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{u} \\ &= \boldsymbol{\beta} + (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{u}\end{aligned}$$

- Under assumptions 1-4, OLS is conditionally unbiased for $\boldsymbol{\beta}$:

$$\begin{aligned}\mathbb{E}[\widehat{\boldsymbol{\beta}}|\mathbf{X}] &= \boldsymbol{\beta} + (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbb{E}[\mathbf{u}|\mathbf{X}] \\ &= \boldsymbol{\beta} + (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{0} \\ &= \boldsymbol{\beta}\end{aligned}$$

- Implies that OLS is unconditionally unbiased: $\mathbb{E}[\widehat{\boldsymbol{\beta}}] = \boldsymbol{\beta}$

Variance of OLS

- What about $\mathbb{V}[\widehat{\boldsymbol{\beta}}|\mathbf{X}]$?
- Using some facts about variances and matrices, can derive:

$$\mathbb{V}[\widehat{\boldsymbol{\beta}}|\mathbf{X}] = \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbb{V}[\mathbf{u}|\mathbf{X}]\mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1}$$

- What the covariance matrix of the errors, $\mathbb{V}[\mathbf{u}|\mathbf{X}]$?

$$\mathbb{V}[\mathbf{u}|\mathbf{X}] = \begin{bmatrix} \mathbb{V}[u_1|\mathbf{X}] & \text{cov}[u_1, u_2|\mathbf{X}] & \dots & \text{cov}[u_1, u_n|\mathbf{X}] \\ \text{cov}[u_2, u_1|\mathbf{X}] & \mathbb{V}[u_2|\mathbf{X}] & \dots & \text{cov}[u_2, u_n|\mathbf{X}] \\ \vdots & & \ddots & \\ \text{cov}[u_n, u_1|\mathbf{X}] & \text{cov}[u_n, u_2|\mathbf{X}] & \dots & \mathbb{V}[u_n|\mathbf{X}] \end{bmatrix}$$

- This matrix is symmetric since $\text{cov}(u_i, u_j) = \text{cov}(u_i, u_j)$

Homoskedasicity

- By homoskedasticity and iid, for any units i, s, t:
  - $\mathbb{V}[u_i|\mathbf{X}] = \mathbb{V}[u_i|\mathbf{x}_i] = \sigma_u^2$ (constant variance)
  - cov$[u_s, u_t|\mathbf{X}] = 0$ (uncorrelated errors)
- Then, the covariance matrix of the errors is simply:

$$\mathbb{V}[\mathbf{u}|\mathbf{X}] = \sigma_u^2 \mathbf{I}_n = \begin{bmatrix} \sigma_u^2 & 0 & 0 & \dots & 0 \\ 0 & \sigma_u^2 & 0 & \dots & 0 \\ & & & \vdots & \\ 0 & 0 & 0 & \dots & \sigma_u^2 \end{bmatrix}$$

- Thus, we have the following:

$$\begin{aligned} \mathbb{V}[\widehat{\boldsymbol{\beta}}|\mathbf{X}] &= \left(\mathbf{X}'\mathbf{X}\right)^{-1} \mathbf{X}' \mathbb{V}[\mathbf{u}|\mathbf{X}] \mathbf{X} \left(\mathbf{X}'\mathbf{X}\right)^{-1} \\ &= \left(\mathbf{X}'\mathbf{X}\right)^{-1} \mathbf{X}' (\sigma_u^2 \mathbf{I}_n) \mathbf{X} \left(\mathbf{X}'\mathbf{X}\right)^{-1} \\ &= \sigma_u^2 \left(\mathbf{X}'\mathbf{X}\right)^{-1} \mathbf{X}'\mathbf{X} \left(\mathbf{X}'\mathbf{X}\right)^{-1} \\ &= \sigma^2 \left(\mathbf{X}'\mathbf{X}\right)^{-1} \end{aligned}$$

Sampling variance for OLS estimates

- Under assumptions 1-5, the sampling variance of the OLS estimator can be written in matrix form as the following:

$$\mathbb{V}[\widehat{\boldsymbol{\beta}}|\mathbf{X}] = \sigma_u^2(\mathbf{X}'\mathbf{X})^{-1}$$

- This symmetric matrix looks like this:

$$\begin{bmatrix} \mathbb{V}\left[\widehat{\beta}_0|\mathbf{X}\right] & \text{Cov}\left[\widehat{\beta}_0, \widehat{\beta}_1|\mathbf{X}\right] & \cdots & \text{Cov}\left[\widehat{\beta}_0, \widehat{\beta}_k|\mathbf{X}\right] \\ \text{Cov}\left[\widehat{\beta}_0, \widehat{\beta}_1|\mathbf{X}\right] & \mathbb{V}\left[\widehat{\beta}_1|\mathbf{X}\right] & \cdots & \text{Cov}\left[\widehat{\beta}_1, \widehat{\beta}_k|\mathbf{X}\right] \\ \vdots & \vdots & \ddots & \vdots \\ \text{Cov}\left[\widehat{\beta}_0, \widehat{\beta}_k|\mathbf{X}\right] & \text{Cov}\left[\widehat{\beta}_k, \widehat{\beta}_1|\mathbf{X}\right] & \cdots & \mathbb{V}\left[\widehat{\beta}_k|\mathbf{X}\right] \end{bmatrix}$$

Inference in the general setting

- Under assumption 1-5 in large samples:

$$\frac{\widehat{\beta}_j - \beta_j}{\widehat{\mathsf{se}}[\widehat{\beta}_j]} \sim N(0,1)$$

- In small samples, under assumptions 1-6,

$$\frac{\widehat{\beta}_j - \beta_j}{\widehat{\mathsf{se}}[\widehat{\beta}_j]} \sim t_{n-(k+1)}$$

- Thus, under the null of $H_0 : \beta_j = 0$, we know that

$$\frac{\widehat{\beta}_j}{\widehat{\mathsf{se}}[\widehat{\beta}_j]} \sim t_{n-(k+1)}$$

- Here, the estimated SEs come from:

$$\widehat{\mathbb{V}}[\widehat{\boldsymbol{\beta}}] = \widehat{\sigma}_u^2(\mathbf{X}'\mathbf{X})^{-1}$$

$$\widehat{\sigma}_u^2 = \frac{\hat{\mathbf{u}}'\hat{\mathbf{u}}}{n-(k+1)}$$

Covariance matrix in R

- We can access this estimated covariance matrix, $\widehat{\sigma}_u^2(\mathbf{X}'\mathbf{X})^{-1}$, in R:

```
vcov(mod)
```

```
##                  (Intercept)    exports        age       male
## (Intercept)      0.0004766593  1.164e-07 -7.956e-06 -6.676e-05
## exports          0.0000001164  1.676e-09 -3.659e-10  7.283e-09
## age             -0.0000079562 -3.659e-10  2.231e-07 -7.765e-07
## male            -0.0000667572  7.283e-09 -7.765e-07  1.909e-04
## urban_dum       -0.0000965843 -4.861e-08  7.108e-07 -1.711e-06
## malaria_ecology -0.0000069094 -2.124e-08  2.324e-10 -1.017e-07
##                  urban_dum malaria_ecology
## (Intercept)     -9.658e-05      -6.909e-06
## exports         -4.861e-08      -2.124e-08
## age              7.108e-07       2.324e-10
## male            -1.711e-06      -1.017e-07
## urban_dum        2.061e-04       2.724e-09
## malaria_ecology  2.724e-09       7.590e-07
```

Standard errors from the covariance matrix

- Note that the diagonal are the variances. So the square root of the diagonal is are the standard errors:

```
sqrt(diag(vcov(mod)))
```

```
##     (Intercept)         exports             age            male
##      0.02183253      0.00004094      0.00047237      0.01381627
##       urban_dum malaria_ecology
##      0.01435491      0.00087123
```

```
coef(summary(mod))[, "Std. Error"]
```

```
##     (Intercept)         exports             age            male
##      0.02183253      0.00004094      0.00047237      0.01381627
##       urban_dum malaria_ecology
##      0.01435491      0.00087123
```

Nunn & Wantchekon

III. Estimating Equations and Empirical Results

A. *OLS Estimates*

We begin by estimating the relationship between the number of slaves that were taken from an individual's ethnic group and the individual's current level of trust. Our baseline estimating equation is:

$$(1) \quad trust_{i,e,d,c} = \alpha_c + \beta \, slave\ exports_e + \mathbf{X}'_{i,e,d,c}\mathbf{\Gamma} + \mathbf{X}'_{d,c}\mathbf{\Omega} + \mathbf{X}'_e\mathbf{\Phi} + \varepsilon_{i,e,d,c},$$

Wrapping up

- You have the full power of matrices.
- Key to writing the OLS estimator and discussing higher level concepts in regression and beyond.
- Next week: diagnosing and fixing problems with the linear model.